박사 학위논문

Ph. D. Dissertation

공공 유비쿼터스 컴퓨팅 환경을 위한 효율적이며 확장성이 개선된 인증 프레임워크 연구

A Scalable Privacy-Preserving Authentication Framework for the Public Ubiquitous Computing Environments

김 장 성 (金 章 成 Kim, Jangseong)

정보통신공학과

Department of Information and Communications Engineering

KAIST

2011

공공 유비쿼터스 컴퓨팅 환경을 위한 효율적이며 확장성이 개선된 인증 프레임워크 연구

A Scalable Privacy-Preserving Authentication Framework for the Public Ubiquitous Computing Environments

A Scalable Privacy-Preserving Authentication Framework for the Public Ubiquitous Computing Environments

Advisor : Professor Kim, Kwangjo

by

Kim, Jangseong

Department of Information and Communications Engineering

KAIST

A thesis submitted to the faculty of KAIST in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Information and Communications Engineering . The study was conducted in accordance with Code of Research Ethics[1].

2011. 05. 25.

Approved by

Professor Kim, Kwangjo

[Advisor]

[1]Declaration of Ethical Conduct in Research: I, as a graduate student of KAIST, hereby declare that I have not committed any acts that may damage the credibility of my research. These include, but are not limited to: falsification, thesis written by someone else, distortion of research findings or plagiarism. I affirm that my thesis contains honest conclusions based on my own careful research under the guidance of my thesis advisor.

공공 유비쿼터스 컴퓨팅 환경을 위한 효율적이며 확장성이 개선된 인증 프레임워크 연구

김 장 성

위 논문은 한국과학기술원 박사학위논문으로
학위논문심사위원회에서 심사 통과하였음.

2011년 05월 25일

심사위원장 김 광 조 (인)

심사위원 김 대 영 (인)

심사위원 이 동 만 (인)

심사위원 이 병 천 (인)

심사위원 최 두 호 (인)

DICE
20068053

김 장 성. Kim, Jangseong. A Scalable Privacy-Preserving Authentication Framework for the Public Ubiquitous Computing Environments. 공공 유비쿼터스 컴퓨팅 환경을 위한 효율적이며 확장성이 개선된 인증 프레임워크 연구. Department of Information and Communications Engineering . 2011. 76p. Advisor Prof. Kim, Kwangjo. Text in English.

ABSTRACT

After the concept of ubiquitous computing environment was introduce by M. Weiser in 1991, many researcher proposed their idea to address challenge issues of ubiquitous computing environments. Recently, marvelous advance of fundamental technologies for ubiquitous computing environments (*i.e.*, sensor network and 4G network) enable us to expect the ubiquitous computing environments will be come true in near future. Based on these research outputs and the advance of fundamental technologies, some of the ubiquitous computing environments (*e.g.*, Navitime in Japan, open urban testbed in Finland, and child-care & safety service in Korea) have been integrated in our real life. From now, we call the ubiquitous computing environments integrated in the real life as the public ubiquitous computing environments.

Compared to the ordinary ubiquitous computing environment, the public ubiquitous computing environments have the following properties: multiple administrative domains, complicated trust model, numerous subscribers of a service, heterogeneous devices, and numerous authentication requests. Although these properties causes the security and privacy problems, the previous work cannot solve these problems: the leakage of the stored subscription information to enforce proper access control, privacy violation due to traceability of the end-user, and low scalability due to the heavy computational overhead and communication cost.

To address these problems, we propose a scalable privacy-preserving authentication framework, consisting of three privacy-preserving authentication protocols: anonymous authentication protocol, privacy-preserving membership verification protocol, and anonymous re-authentication protocol. All protocols reduce the processing delay time and communication overhead while preserving the privacy of the end-user. The privacy-preserving membership verification protocol allows the service provider to adjust its performance to satisfy his/her desire performance in the various services. In addition, the processing delay time does not be affected by the number of the service subscriber. Also, the protocol can preserve the privacy of the end-user based on computationally hard problem.

Through a set of the selected numbers and standard version of membership verification, the anonymous authentication protocol enhances non-linkability and removes the privacy concern regarding to the abuse of the stored service subscription information in the anonymous authentication protocol.

The anonymous re-authentication protocol employing the lightweight version is used to provide an efficient anonymous re-authentication. If the end-user has authenticated in the previous session, the end-user can share a secret information with the wireless sensor network. The proposed anonymous re-authentication verifies whether the end-user has the secret information or not.

In order to illustrate how to employ these authentication protocols to build privacy-preserving protocols for the public ubiquitous computing environments, we provide a privacy-preserving secure service discovery protocol as one example. In addition, we demonstrate that the proposed protocols address the security and privacy problems of the possible services, child-care and safety service, in the public ubiquitous computing environments.

Contents

List of Tables

List of Figures

Chapter 1. Introduction

1.1 Motivation

After M. Weiser proposed the concept of ubiquitous computing environments [1], many papers are proposed to address the research challenges of ubiquitous computing environments. For service discovery, which allows the end-user to find an alternative service, Czerwinski *et al.* [2] proposed a scheme which was called "Secure Service Discovery Service". Also, Zhu *et al.* [3] introduced another secure service discovery protocol to preserve the privacy of the end-user while protecting the privacy of the service provider. In order to provide anonymous authentication preserving the privacy of the end-user, many researchers proposed their idea [4, 5, 7, 8, 9, 10, 11, 12, 15, 16, 13, 14]. Keyword search techniques on the encrypted data [17, 18, 19, 20] are proposed to share the data while preventing the other entities having no access control from obtain the data. Various key establishment schemes [21, 22, 23, 24, 25, 26, 27, 28, 29] for wireless sensor network, one of core technologies to establish ubiquitous computing environment, have been introduce to address security issues.

Based on these research results and recent advance of the technologies (*e.g.*, Bluetooth, Wi-Fi, sensor network, smart phone), the ubiquitous computing environments have being integrated in our real life. For instance, in Japan a pedestrian navigation through a mobile phone, called as Navitime, is provided to almost 2 million Japanese citizens [30]. "iHospital system" [31] is deployed to support a hospital scheduling and awareness system in Denmark through location tracking, a context-awareness system, large interactive displays, and mobile phones. Using iHospital system, the hospital can adjust the schedule changes and maintain a shared overview of the current surgeries and personnel, and provide to communicate withe the people involved. In Finland, "Open Urban Testbed for Ubiquitous Computing" provides open horizontal resources for building functional prototypes of future ubiquitous city [32]. The testbed consists of panOULU WLAN, panOULU BT, panOULU WSN, large public displays, middleware resources, and monitoring tools where panOULU WLAN provides wireless network through a public-private partnership, panOULU BT is a cluster of Bluetooth APs around downtown Oulu, panLULU WSN is a cluster of wireless sensor network AP's around downtown Oulu. Note that wireless sensor network AP provides open and free connectivity to low-power sensor nodes in wireless sensor network. Also, in Korea, mobile service providers (*i.e.*, SKT, KT, and LGT) and Gangnam province in Seoul [33] provide a child's location information to parents' mobile phone per every 1 hour and send an alarm message if a child is out of his/her safety zone, predetermined by the parent. From now, we call the ubiquitous computing environment integrated in our real life as the public ubiquitous computing environment. Compared to the ubiquitous computing environments which have been researched in the open literature, the public ubiquitous computing environments has the following differences:

Multiple administrative domains: The ubiquitous computing environment, had been researched in the open literature, focuses on single administrative domain such as ubiquitous campus or smart home. However, in our real life, multiple administrative domains coexist. ISPs (Internet Service Providers), mobile service providers (*i.e.*, SKT, KT, and LGT) are typical examples of multiple administrative domains.

Complicated trust model: Due to the multiple administrative domains, the trust model in the public ubiquitous computing environment will be complicated. While the end-users and service providers need to trust only one authentication server and multiple directory servers managed by the authentication server in the ordinary ubiquitous computing environments, multiple directory servers and authentication servers coexist. Note that the authentication server is used to provide anonymous authentication, which protects the end-user's privacy in the public ubiquitous computing environment. As a result, the end-users and service providers may have the privacy concern related to the abuse of the subscription information, which is stored in the authentication server and directory server to enforce access control.

Numerous subscribers of a service: As the scale of the public ubiquitous computing environments is considered a city, the number of end-users subscribing the server may be the whole number of citizen in the city. According to the experimental result in panOULU WLAN [32], "about 20000 WLAN devices use of panOULU WLAN every month so that 25-40% of them are visitors", the privacy-preserving protocols in the literature may be suffered from the scalability issue.

Heterogeneous devices: As the public ubiquitous computing environments is designed to cover a city, the devices should be diverse to support various services satisfying the needs of the whole citizens.

Numerous authentication requests: The number of the end-users subscribing the wireless sensor network increases as the coverage of the public ubiquitous computing environment has been expanded to a city. The public ubiquitous computing environments include high-bandwidth network (*e.g.*, 3GPP, 4G, and WLAN) and low-bandwidth network (*e.g.*, wireless sensor network). As the numerous authentication requests cannot be handled in low-bandwidth network, the network cannot be operated to satisfy its original goal.

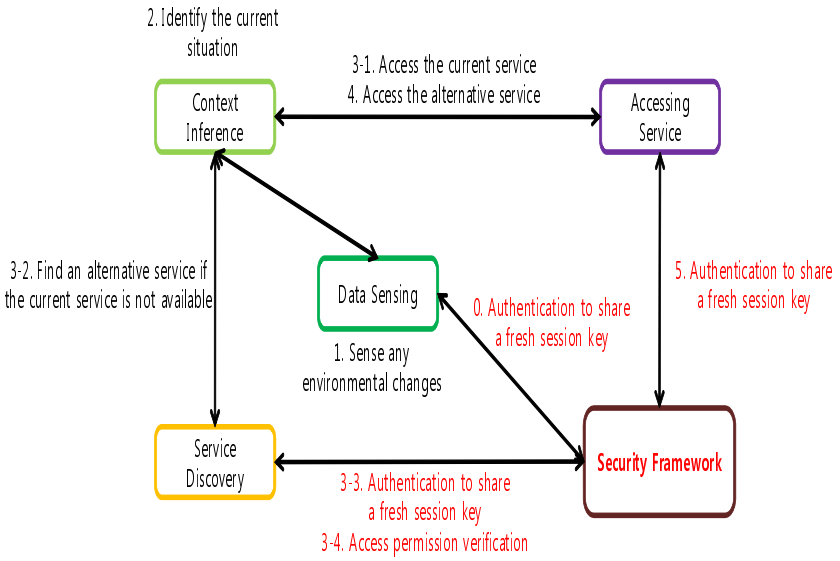

Figure 1.1: Abstract activities for the public ubiquitous computing environments

The activities of the end-users for the public ubiquitous computing environments can be summarized as Figure 1.1. The authentication of the end-users are required in three parts: authentication to obtain the sensing data from the wireless sensor network, authentication to find an alternative service during service discovery, authentication to access the target service. Therefore, the authentica-

tion framework for the public ubiquitous computing environment should support these authentications which preserves the privacy of the end-users and service providers. Then, the authentication framework should support the following security requirements:

Mutual authentication: Mutual authentication is required since each end-user and service provider want to identify whether the communicating party is legitimate entity or not. When mutual authentication is not provided, an adversary can impersonate a specific end-user or service provider.

Anonymity: On the one hand, mutual authentication provides a functionality that an end-user and service provider identify each other, on the other hand, it also enables an adversary to track the end-user or service provider. In the public ubiquitous computing environments, the end-user may provide his/her service to the other end-users. This is why we should protect privacy of the end-user and service provider. A typical approach to protect the privacy of an end-user and service provider is to provide anonymous communication through an authentication server.

Accountability: Although anonymity can protect the privacy of an end-user and service provider, it also can help a malicious user, who has overheard the authorized credential of another user, access several services without permission. Hence, proper accountability is required.

Differentiated access control: As a service provider may want to provide differentiated service to his/her subscribers, the system in the public ubiquitous computing environment should support differentiated access control. However, anonymous communication through an authentication server allows the authentication server to access the subscription information of a specific service provider, which causes the privacy concern for the service provider that the administrator of the authentication server illegally extracts the subscription information (*e.g.*, mobile user tracking and information leakage of the subscribers). This concern is critical issue in the public ubiquitous computing environments because the multiple administrative domains exist in the public ubiquitous computing environments. Thus, we need to remove the privacy concern regarding the abuse of subscription information in authentication server while supporting differentiated access control with anonymity.

Efficient keyword search on encrypted data: When the authentication server verifies the access permission of an end-user on a service, the authentication server may obtain the subscription information of the service. Keyword search on encryption data (or privacy-preserving keyword search) is one of possible solutions to address this problem. Because the approach can allow the server to verify the access permission on the specific service without exposing the subscription information. However, the existing approach is not efficient in view of computational overhead. In the public ubiquitous computing environments, the service subscriber may be all citizens in a city. That's why efficient keyword search on encrypted data is required.

Lightweightness: As the public ubiquitous computing environments is considered to cover a city, the devices should be diverse to support various services satisfying the needs of the whole citizens. That's why the cryptographic protocols which should be run on several devices should be lightweight with respect to communication, computation cost, and storage overheads.

1.2 Challenges

Due to the characteristics of the public ubiquitous computing environments, the privacy-preserving authentication protocol, which is proposed to support the activities of the end-user, should consider the following:

Challenges in Privacy protection

1. **Semi-honest directory server**: Since the directory servers are owned by multiple administrate domains, the directory server should be regarded as semi-honest entity. In other words, the directory server performs its original functionalities (*i.e.*, storing the access information of the service and providing the access information to the end-user having proper access permission) while the server may collect the private information of the end-user or service provider.

2. **Semi-honest authentication server**: Since the authentication servers are owned by multiple administrative domains, one authentication server cannot want to provide the subscription information, which is required to enforce proper access control, to another authentication server. While preventing another authentication server from obtaining the subscription information, one authentication server wants to allow another authentication server to authenticate the end-user belonging to its authentication server.

3. **Enhanced non-linkability**: The adversary can easily eavesdrop any communication since the environments are open to all entities. If the adversary can identify the end-user or service provider, the adversary can monitor the activities of the target entity (*i.e.*, end-user or service provider) and track the location of the target entity. Here, non-linkability means that, for insiders(*i.e.*, service provider and other users) and outsiders, 1) neither of them could ascribe any session to a particular end-user, and 2) neither of them could link two different sessions to the same user [34].

Challenges in Scalability

1. **Lower processing delay time**: As the average number of the service subscriber in panOULU WLAN is about 20000, lower processing delay time is required in the directory server and authentication server so that the servers can support more end-users within the certain time period.

2. **Lower communication overhead**: As the various devices can be used in the network having limited bandwidth (*i.e.*, sensor network), lower communication overhead is preferred to support more end-users within the certain time period. Communication overhead can be determined by the size of the message to be transmitted and the number of rounds to achieve the protocol.

Based on these observations, Figure 1.2 illustrates the challenges to be addressed in the authentication protocols for the public ubiquitous computing environments.

However, the existing authentication protocols, authentication to obtain the sensing data from the wireless sensor network, authentication to find an alternative service during service discovery, authentication to access the target service, in the ubiquitous computing environments do not consider the above challenges of the public ubiquitous computing environments.

During service discovery protocol, an end-user, Alice, may expose her private information to the directory server, which is introduced to simplify trust management and support better scalability [2, 3]. However, the adversary can obtain the information from the directory server by compromising the server in the public ubiquitous computing environment since the server may not be protected from the physical access of the adversary. In addition, the authentication server or service provider can identify whether two different authentication requests are sent by Alice or not [2]. Hence, Alice can not prevent the adversary from obtaining the private information of Alice.

Although various anonymous authentication schemes based on one-time pseudonym [13, 14, 15, 16] have been proposed, these schemes have the following problems: there is no concrete protocol [13]; the

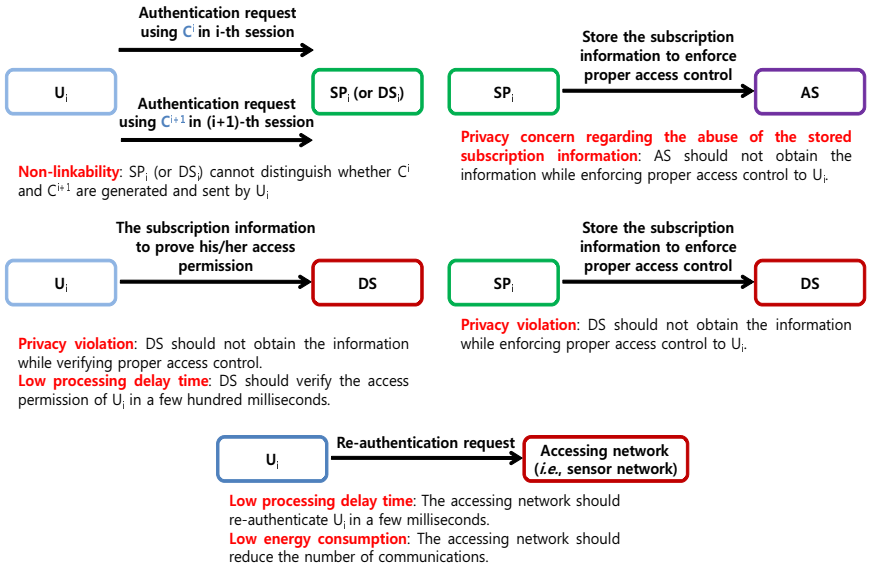

Figure 1.2: Challenges of the authentication protocols for the public ubiquitous computing environments

adversary can reuse the authorized pseudonym by overhearing the authentication request of the end-user [14]; the service provider can link up two different sessions to the same end-user so that the service provider can track the activity of the end-user [15, 16]. When Alice authenticates herself to the service provider through these approaches, the privacy of Alice may not be protected.

To verify access permission of Alice, the authentication server should have a capability to verify whether Alice is a legitimate subscriber of the service provider. In the mean time, the authentication server should not obtain the subscription information of the service provider in order to remove the privacy concern on abuse of the subscription information. If Alice should register herself with each service provider, the management cost of each service provider will be increased to prevent the private information exposure of Alice from the adversary. In order to address this issue, the keyword searches on the encrypted data [17, 18, 19, 20] are introduced in the open literature. However, access control does not considered [20]. The computational overhead employing proper access control [20] increases as the number of the end-user subscribing the target service grows.

When Alice uses indoor location supporting service over wireless sensor network, the nearby sensor nodes, which are deployed to monitor any change of the nearby environment and required to forward the received authentication request of Alice, should consume their energy. As the number of the subscribers for this indoor location service increases, the expected lifetime of the deployed sensor nodes will dramatically reduce. A more serious problem is that the sensor nodes cannot perform its operation due to the numerous message forwarding of the authentication request. Because the existing key management schemes [21, 22, 23, 24, 25, 26, 27, 28, 29] require the nearby nodes to forward the authentication requests to the other nodes or base station. If the device hold by Alice can directly communicate with the base station, which collects the sensed data from the deployed sensor nodes and provide the data to the end-user having proper access permission, we can avoid this problem. However, due to the heterogeneous device in the ubiquitous computing environment, we cannot sure whether the device can directly communicate with the base station. As a result, we should provide the efficient re-authentication of the end-user over the sensor network.

1.3 Contributions

In this thesis, we propose scalable privacy-preserving authentication framework, addressing the above problems, for the public ubiquitous computing environments. In order to support three kind of authentications, shown in Figure 1.1, we propose three privacy-preserving authentication protocols: anonymous authentication protocol, privacy-preserving membership verification, anonymous re-authentication protocol.

We introduce two versions of privacy-preserving membership verification to identify whether the end-user has access permission on the service or to find an alternative services which are specified by the encrypted keywords. Using the standard version of our membership verification, we can remove privacy concern about abuse of subscription information, stored to allow an authentication server to check access permission of an end-user, in authentication server. As the computational overhead in the verifier has reduced to a few hundred milliseconds, the number of membership verification done by the verifier within the certain time period increases. The processing delay time can be adjusted for satisfying the desire performance.

The lightweight version of privacy-preserving membership verification can prove that the end-user has authenticated himself/herself in the previous session by showing the knowledge related to the shared secret information. In order to verify whether the end-user has the shared information, the verifier (*i.e.*, sensor node) only requires 1 modular addition and 1 modular exponentiation rather than sending the authentication request to the authentication server or base station. Therefore, the lightweight version of membership verification can be used to provide an efficient re-authentication in low-bandwidth network such as wireless sensor network by reducing communication overhead and processing delay time in the sensor node.

Anonymous authentication protocol based on one-time pseudonym, enhances non-linkability by removing the relationship between the previous pseudonym and current pseudonym. While providing the enhanced level of security, the scheme reduces the computational overhead to authenticate the end-user in the authentication server. This authentication protocol is proposed to provide anonymous authentication to access the target service.

Anonymous re-authentication protocol, based on the lightweight version of membership verification, is proposed to provide an efficient re-authentication of the end-users in the wireless sensor network by reducing the communication overhead to authenticate the end-user. Based on the computationally hard problem, the proposed re-authentication protocol can preserve the privacy of the end-users.

We employ these scalable privacy-preserving authentication protocols to provide an actual service in the public ubiquitous computing environments, called as child-care and safety service while preserving the privacy of the end-user. Through performance and security analysis, we demonstrate that the service can provide the privacy protection of the end-user while supporting better scalability.

To illustrate how to apply the proposed protocols to build a privacy-preserving protocol for the public ubiquitous computing environments, we introduce privacy-preserving secure service discovery protocol. Using the privacy-preserving keyword search, we can remove the privacy concern about the abuse of the stored service subscription information in the authentication server and directory server. Compared to the existing approaches, the proposed service discovery protocol requires a constant processing time although the the number of the service subscribers increases.

From these results, we believe that the proposed authentication protocols can address security

and privacy issues in the public ubiquitous computing environments.

Since these privacy-preserving authentication protocols require lower the processing delay time and less communication overhead compared to the existing approaches, we believe that these protocols can address scalability issue in the public ubiquitous computing environment. In chapter 5, we explain the performance of each protocol in detail.

1.4 Assumption and notations

We assume that an end-user can control the source addresses of the outgoing Medium Access Control (MAC) frames since this assumption is a prerequisite for anonymous communications. A detailed method for this modification is covered by Gruteser *et al.* [35]. Table 1.1 illustrates the notations used throughout this paper. Note that the PK_{BGN} and SK_{BGN} are only used in membership verification. In order to support proper level of security in membership verification, the key size should be the greater than or equal to 512 bit. The reason why we choose that size of PK_{BGN} will be explained in the chapter 3. However, the size of PK_{Entity} is 160 bit which implies the same level of security as RSA-1024.

The AS issues SID, a polynomial $f(x)$ with degree t, access key ak_i, $E[i+r, PK_{BGN}, G_1]$, $Cert_A$ and ID_i to service provider. The SP generates several polynomials, issues an identifier per each polynomial, distributes necessary information for service access (*i.e.*, SID, polynomial degree p, access key ak_i, $E[i + r, BGN_{pk}]$ and user account w_i) to his/her subscribers and stores requisite information for membership verification (*i.e.*, SID list, polynomial identifiers and corresponding polynomials) to database server in an encrypted form. Note that $E[i + r, BGN_{pk}]$ is polynomial evaluation result of the corresponding access key ak_i. Using the received information, the end-user or SP can generate their MT. The SP stores SID list and polynomials for membership verification to the AS in an encrypted form. If there is any change in the stored information, the SP updates the stored information.

A secure tunnel (*e.g.*, IPsec ESP mode [36]) between a directory server and the authentication server exists. PK_{AS}, ID_{AS} and PK_{BGN} are assumed to be known to all entities.

Each sensor node (or mobile node) has SK_{BGN}, R_{BS}, and PK_{BGN} when the node is distributed. The mobile node, owned by the child, performs power adaptation for delivering a message directly to its neighbor cluster head via omnidirectional or directional antenna. Network topology is established using K_{Init} during network initialization.

1.5 Organization

We briefly outline the structure of this thesis as follows:

Chapter 2 introduces the previous work on three re-presentive services: child-care and safety service, keyword search in the intelligent lecture assistance, and re-authentication in the campus navigation over the sensor network. In the child-care and safety service, we explain the existing service provided by the mobile service providers such as SKT, KT, and LGT. Also, we introduce the existing approach by Takata *et al.* [37] in the open literature.

Chapter 3 introduces two versions of privacy-preserving membership verification to allow the verifier (*i.e.*, authentication server and directory server) to check whether the end-user has proper access permission on a service while preserving the privacy of the end-user. Also, the proposed approach can allow the end-users to find an alternative services which are specified by the encrypted keywords.

Table 1.1: Notations

AS / BS	Authentication server / Base Station
$Credential$	A ticket for authentication
DS / HS	Directory Server / Home server
ID_A	An identifier of entity A
KD / MO	Kid's device / Mobile Operator
DMV	A ticket for Discovery Membership Verification
MT	A trapdoor for indicating subscribers of the target SP
n	An allowable access frequency of an end-user to a service
PK_A	A public key of entity A under Elliptic Curve Cryptography
PK_{BGN}	A public key under BGN encryption [44] owned by AS
S	A set of selected numbers where $\mid S \mid \geq 2n$
SID	A service type identifier describing a selected subset of the available service pool and including an polynomial identifier for membership test
SK_A	A private key of entity A
SK_{BGN}	A private key under BGN encryption [44], which is owned by AS and distributed to DS for membership test
SN	Sink node
SP	Service provider or service access point
U	Mobile user
$C^i, i = 0, 1, \cdots$	A series of authorized credentials
$j^i, i = 1, 2, \cdots$	A series of the number selections by an end-user
$Anchor$	An initial credential C^0
$Cert_A$	A certificate which binds entity A with PK_A
$E\{m, K_A\}$	A message m is encrypted by a symmetric key K_A
$E[m, PK_A]$	A message m is encrypted by public key of entity A using Elliptic Curve Cryptography
$D[m, SK_A]$	A message m is signed by private key of entity A using Elliptic Curve Cryptography
$E[m, PK_{BGN}, \mathbb{G}$ or $\mathbb{G}_1]$	A message m is encrypted by the public key PK_{BGN} on cyclic group $\mathbb{G}$ or $\mathbb{G}_1$ and the ciphertext is $g^m h^r$ or $g_1^m h_1^r$ where $g_1 = e(g, g)$ and $h_1 = e(e, h)$
$H(m)$	Hashing a message m
$K_{A,B}$	A shared secret key between entities A and B
$R_A^i, i = 1, 2, \cdots$	A series of nonce generated by entity A which is usually a 64-bit pseudo random number.

When the end-user has enough resource to compute the heavy computational operation such as pairing, the standard version of our membership verification is useful. The lightweight version of membership verification is proposed to provide an efficient re-authentication of the end-user. When the end-user has authenticated himself/herself with the service provider, the end-user can share a secret information with the service provider. Based on this fact, the lightweight version of membership verification

check whether the end-user knows the shared secret information or not.

Using these privacy-preserving membership verifications, we build anonymous authentication protocol and anonymous re-authentication protocol to support various authentications in the public ubiquitous computing environments.

Chapter 4 employs the proposed authentication protocols, explained in the chapter 3, to the child-care and safety service, which is one of the possible applications in the public ubiquitous computing environments. We discuss the system model to address security and privacy problems of this service, necessary assumptions, and the sketch of this service. In the sketch of child-care and safety service, we explain the issue to be addressed in each phase and how to apply the proposed schemes. To provide how to implement each service, we illustrate the actual process for child-care and safety service. Through evaluation of the child-care and safety service, we show that the service can address security and privacy issues for the public ubiquitous computing environments.

To illustrate how to apply the proposed authentication protocols to build the privacy-preserving protocols in the public ubiquitous computing environments, we demonstrate privacy-preserving secure service discovery protocol. Using the standard version of membership verification, the proposed protocol prevents the directory server from obtaining the subscription information of the service subscriber while verifying proper access permission on the requested service. Compared to the previous approach [3], the proposed protocol reduces the communication overhead so that the proposed protocol can support better scalability.

Chapter 5 evaluate three privacy-preserving authentication protocols. The standard version of membership verification is implemented under Intel®Core ™2 2.13GHz CPU, 1GB RAM and Microsoft Windows XP Professional Service Pack 3. In order to illustrate the the efficiency of the lightweight version, we compute the expected processing time. Also, we explain why the proposed membership verifications are secure based on several observations.

Also, we evaluate anonymous authentication protocol, applying the standard version of membership verification. By comparing the computational overhead, communication overhead, and storage cost with the previous work [15, 16], the protocol can provide better scalability. Using reasoning based approach, we show that the protocol can address the security and privacy problems in the public ubiquitous computing environment while supporting various security-related features.

We show that anonymous re-authentication protocol for the end-user over wireless sensor network can address the security and privacy problems in the public ubiquitous computing environment. Using the lightweight version of membership verification, the proposed protocol reduces the communication overhead and the processing delay time to authenticate the end-user. Based on the security of the lightweight version of membership verification, the proposed protocol can preserve the privacy of the end-user.

Chapter 6 conclude this dissertation with the summary and future research.

Chapter 2. Related work

2.1 Security and privacy concerns in the public ubiquitous computing environments

2.1.1 Navitime in Japan

In Japan, a pedestrian navigation through a mobile phone, called as Navitime, is provided to almost 2 million Japanese citizens [30]. To estimate the end-user's location, Navitime used the GPS based location determination technique by incorporating map-matching techniques to obtain an accuracy of 3 meters. If the end-user exists in indoor spaces or urban canyons, cell-tower positioning is used. Compared to the other mobile phone-based navigation services such as Google Mobile Maps and smart2go, Navitime provides the integrated support for various modes of transportation: walking, driving, and riding trains, buses, taxis, and airplanes. Using the various interactive guides (*e.g.*, maps, itineraries, voice prompt, vibration alerts, and progress bars), the end-user can arrive his/her final destination.

In location-based services, a simultaneous effort to both protect the location privacy and disclose location information to the others socially. To satisfy the need to re-use and recall the previous location information, the 20 most recent location search queries are stored in the server of the mobile service provider. By encrypting the personally identifiable information by the end-user, the stored queries are not linked to the personally identifiable information. The end-user can delete the historical data. As the service is based on the mobile phone, the anonymous authentication cannot be provided during accessing the service.

2.1.2 iHospital system in Denmark

"iHospital system" [31] is deployed to support a hospital scheduling and awareness system in Denmark through location tracking, a context-awareness system, large interactive displays, and mobile phones. Using the iHospital system, the hospital can adjust the schedule changes and maintain a shared overview of the current surgeries and personnel, and provide to communicate withe the people involved.

The system consists of AwareMedia, location and context-awareness infrastructure, and AwarePhone. AwareMedia is an application providing information about the status in the different operating rooms. Location and context-awareness infrastructure is used to monitor the location of people in the hospital using Bluetooth. AwarePhone is a program running on Symbian mobile phone which provides the status of the surgeries in the operating rooms. Based on the knowledge obtained from the system deployment, we conclude that the performance and scalability issues should be addressed when the system for the pervasive computing environment is deployed. Moreover, security and privacy should be considered in order to deploy the system in the real world.

2.1.3 Open Urban Testbed in Finland

"Open Urban Testbed for Ubiquitous Computing" provides open horizontal resources for building functional prototypes of future ubiquitous city [32]. The testbed consists of panOULU WLAN, panOULU BT, panOULU WSN, large public displays, middleware resources, and monitoring tools where panOULU WLAN provides wireless network through a public-private partnership, panOULU BT is a cluster of Bluetooth APs around downtown Oulu, panLULU WSN is a cluster of wireless sensor network AP's around downtown Oulu. Note that wireless sensor network AP provides open and free connectivity to the low-power sensor nodes in wireless sensor network.

The panOULU WLAN provides open (no authentication or registration) and free (no payment) wireless Internet access to the equipped a WLAN device. According to the experimental result in panOULU WLAN [32], "about 20000 WLAN devices use of panOULU WLAN every month so that 25-40% of them are visitors", scalability issue of privacy-preserving protocols in the literature should be re-considered.

This testbed was used in UMA (Unlicensed Mobile Access) technology pilot, allowing to access GSM core services over the unlicensed wireless network (*i.e.*, panOULU WLAN) if it is available. Also, the testbed was used in panOULU Luotsi service prototype, providing location-based information to the users of the panOULU WLAN. The location of the end-user is estimated by identifying the WLAN AP to which the end-user's wireless device is currently connected.

2.1.4 Child-care and safety service in Korea

In Korea, mobile service providers (*i.e.*, SKT, KTF, and LGT) provide child-care and safety service. The service providers notify a child's location information to parents' mobile phone per every 1 hour and send an alarm message if a child is out of his/her safety zone, predetermined by the parent. Also, Gangnam province in Seoul [33] provides a similar service to an end-user.

From now, we call this service as child-care and safety service. Then, we can define the service using three components such as end-user, service provider, and device. The end-user requests the service provider to send the location information of his/her child having a proper device where the location is determined by various techniques such as A-GPS [38], ultrasound [39] and RSSI based on RF [40, 41]. Figure 2.1 illustrates the detailed activities among an end-user, service provider and child. When the end-user becomes a legitimate subscriber of a child-care & safety service, the user should store the private information (*i.e.*, his/her mobile phone number, living space, frequent visiting place, and *etc.*) to the service provider. Then, the child who has the proper device can report his/her location to the end-user through the service provider. Using the received location information, the service provider analyzes the risk of the child. If any emergency situation has been occurred or periodic reporting is required, the service provider sends the analyzed result to the end-user.

However, this approach has the following disadvantages: the child's location is not accurate due to A-GPS (Network-assisted Global Positioning System) [38], typical method of location determination technology; Private information such as safety zone and mobile phone should be stored in a server of mobile operator; An end-user cannot control over the child-care and safety service although the user does not want to observe the child's location during some time period.

In the open literature, Takata *et al.* [37] proposed a dangerous location aware system for assisting kids' safety care. To assist for kid's safety care, they assume that each kid has proper mobile devices communicating with a server in his/her home and a public alerting service notifying several dangerous

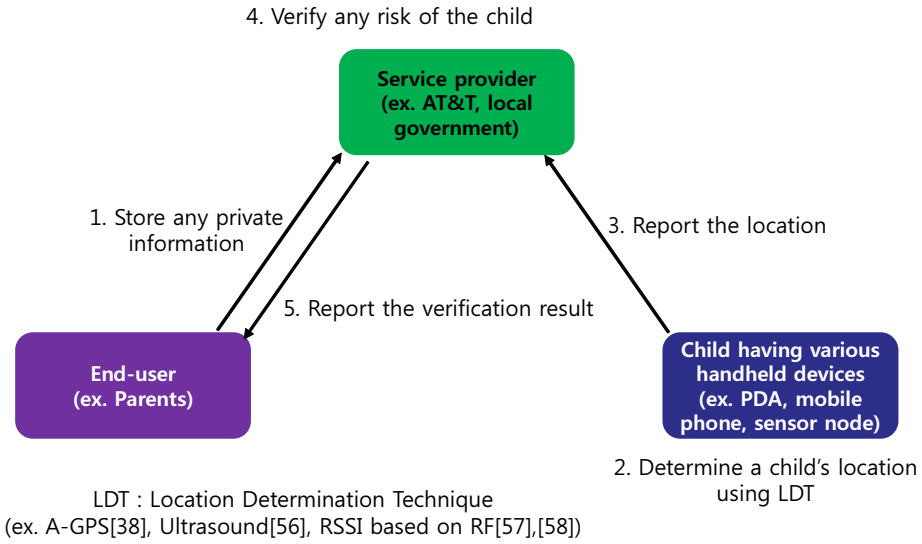

Figure 2.1: Existing model for child-care and safety service

location with real-time traffic exists. Compared to the commercial services, the system can preserve the privacy of an end-user by storing any private information in his/her home server and determining the kid's location in the kid's device. However, direct communication between the kid's device and home server in his/her home is expensive and impractical since the devices should support various networking technologies as any changes of the nearby environment.

2.2 Privacy-preserving authentication protocols

2.2.1 Membership verification through keyword search

In order to enforce proper access control on keyword search, the subscription information should be encrypted. Typical approach presenting the subscription information on the specific service was presented in the polynomial form so that the searching of the set can be converted to the evaluation of the given polynomial [42, 43]. The existing approach [20] in the open literature, being able to employ access control to keyword search on the encrypted data, requires homomorphic operations in addition and multiplication. Among the homomorphic encryptions [45], the encryption scheme proposed by Boneh *et al.* in 2005 [44] can support the unlimited additive homomorphic operations and one multiplicative homomorphic operation. That's why we employ the encryption scheme proposed by Boneh *et al.* to hide the subscription information during keyword search.

BGN encryption

As mentioned in the introduction, we need to get rid of privacy concern regarding the abuse of subscription information in authentication server. In other words, the authentication server should verify whether an end-user is one of subscribers without information leakage of the subscription information. To address this issue, we employ the previous approaches [42, 43], converting the searching of the sets to an evaluation of polynomial representations of a given set.

In 2005, Boneh *et al.* proposed new encryption scheme [44] which supports additively homomorphic operation and one multiplicative operation on encrypted data. Before describing the scheme, we explain the notation used in the scheme and how to construct the bilinear groups.

The homomorphic encryption scheme proposed by Boneh *et al.* uses the following notation:

1. $\mathbb{G}$ and $\mathbb{G}_{\mathbb{1}}$ are two (multiplicative) cyclic groups of finite order n.
2. g is a generator of $\mathbb{G}$.
3. e is a bilinear map e: $\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_{\mathbb{1}}$. Namely, for all u, $v \in \mathbb{G}$ and a, $b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$.
4. $e(g, g)$ is a generator of $\mathbb{G}_{\mathbb{1}}$.

If $\mathbb{G}$ is a bilinear group, we can say that a group $\mathbb{G}_{\mathbb{1}}$ and a bilinear map as above also exist.

Let $n > 3$ be a given square-free integer that is not divisible by 3. Then, a bilinear group $\mathbb{G}$ of order n can be constructed as follows:

1. Find the smallest positive integer $l \in \mathbb{Z}$ such that $p = ln - 1$ is prime and $p = 2 mod 3$.
2. Consider the group of points on the (super-singular) elliptic curve $y^2 = x^3 + 1$ defined over $\mathbb{F}$. Since $p = 2 mod 3$, the curve has $p + 1 = ln$ points in $\mathbb{F}$. Therefore the group of points on the curve has a subgroup of order n which denote by $\mathbb{G}$.
3. Let $\mathbb{G}_{\mathbb{1}}$ be the subgroup of $\mathbb{F}^*$ of order n. The modified Weil pairing on the curve [46] gives a bilinear map e: $\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_{\mathbb{1}}$ with the required properties.

The homomorphic encryption scheme consists of the three algorithms as follows:

1. KeyGen(τ): Let $n = q_1 q_2$. Pick two random generators g and u from the group $\mathbb{G}$ and set $h = u^{q_2}$. Then h is a random generator of the subgroup of $\mathbb{G}$ of order q_1. The public key PK_{BGN} is $(n, \mathbb{G}, \mathbb{G}_{\mathbb{1}}, e, g, h)$. The private key SK_{BGN} is q_1.
2. Encrypt(PK_{BGN}, m): When the message space consists of integers in the set $\{0, 1, \ldots, T\}$ with $T < q_2$ and r is a random number from 0 to $n-1$, the encryption is computed as $C = g^m h^r \in \mathbb{G}$. C is the encrypted message of m.
3. Decrypt(SK_{BGN}, C): The decryption is computed as $C^{q_1} = (g^m, h^r)^{q_1} = (g^{q_1})^m$.

Then, the encryption of $m_1 + m_2$ and the encryption of $m_1 m_2$ can be computed as $g^{m_1} h^{r_1} \cdot g^{m_2} h^{r_2}$ and $e(g^{m_1} h^{r_1}, g^{m_2} h^{r_2})$ where e is a bilinear mapping from $\mathbb{G} \times \mathbb{G}$ to $\mathbb{G}_{\mathbb{1}}$ and the encryption of m_i is $g^{m_i} h^{r_i}$ in BGN scheme. Also, the expected decryption time using the lambda method proposed by Pollard is $\tilde{O}(\sqrt{|m|})$ [44] although the authentication server has the private key $vk = q_1$.

Since the encryption of m_i is $g^{m_i} h^{r_i}$, the encryption of $m_1 + m_2$ is $g^{m_1 + m_2} h^{r'} = g^{m_1} g^{m_2} h^{r'}$. By assigning $r' = r_1 \cdot r_2$, the encryption result of $m_1 + m_2$ can be expressed as $g^{m_1} h^{r_1} \cdot g^{m_2} h^{r_2}$. To allow one multiplication of two encrypted messages, we use the bilinear map. When $g_1 = e(g, g)$ and $h_1 = e(g, h)$, the encryption of $m_1 m_2$ is $g_1^{m_1 m_2} h_1^{r''}$.

$$e(g^{m_1} h^{r_1}, g^{m_2} h^{r_2}) h_1^{r'} = \begin{cases} e(g^{m_1 + q_2 r_1},\ g^{m_2 + q_2 r_2}) h_1^{r'} \\ e(g, g)^{(m_1 + q_2 r_1) \cdot (m_2 + q_2 r_2)} h_1^{r'} \\ g_1^{(m_1 m_2 + \alpha q_2 r_2 m_1 + \alpha q_2 r_1 m_2 + \alpha^2 q_2^2 r_2 r_1)} h_1^{r'} \\ g_1^{m_1 m_2} h_1^{r' + r_2 m_1 + r_1 m_2 + \alpha q_2 r_2 r_1} \\ g_1^{m_1 m_2} h_1^{r''} \end{cases}$$

Note that $h_1 = e(g,h) = e(g,g^{\alpha q_2}) = e(g,g)^{\alpha q_2} = g_1^{\alpha q_2}$. Also, we can assign $r'' = r' + r_2 m_1 + r_1 m_2 + \alpha q_2 r_2 r_1$ since r'' is a random number in $\mathbb{Z}_n$.

Keywords search on the encrypted data

A keyword search on encrypted data is introduced to share audit log and email on a public server while minimizing information leakage. Previous protocols [17, 18, 19] have three common entities in their system models: a data provider, public server, and data retriever. The data provider generates shared information and stores it on a public server in an encrypted form. Only an entity having a proper trapdoor (*i.e.*, access permission) can retrieve the stored information. This approach can remove privacy concern regarding the abuse of subscription information stored in authentication server. However, no access control is provided [20] and the server can link two different sessions to the same group using the relationship between the stored data and the submitted trapdoor.

To provide access control only, [20] proposed an idea to convert the searching of the sets to an evaluation of polynomial representations of a given set [42, 43] using BGN encryption [44]. Interestingly, this approach can address the second problem due to non-deterministic property of BGN encryption which will be explained in the following section. However, the proposed approach is not efficient in view of computational overhead. Denote S_1 and S_2 by a set of access keys and a set of keywords, respectively. Then, the data retriever should compute $|S_1|+|S_2|+1$ exponent multiplications and BGN encryptions [44] per each query. Also, the server should compute $|S_1|+|S_2|+1$ pairing operations and $2 \cdot |S_1| + |S_2| + 1$ exponent multiplications per each query. Figure 2.2 illustrates the evaluation result proposed by Yau *et al.*.

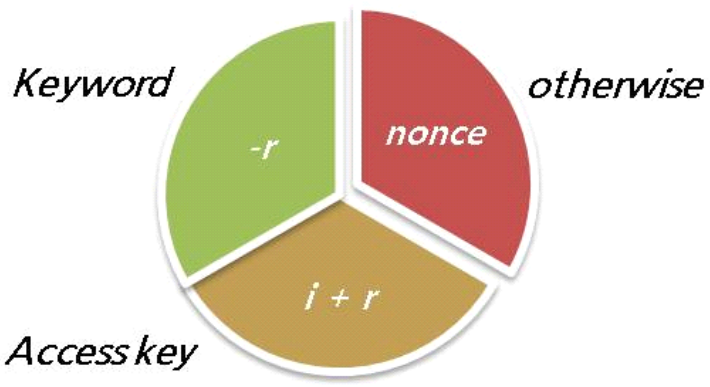

Figure 2.2: Evaluation result of the polynomial f(x) in [20]

When the encrypted data is the the subscription information regarding a service, the size of S_1 and S_2 are the number of the subscribers in the service. If the data is the keywords specifying the service, the size of S_1 (or S_2) is the number of the subscribers in the service (or the number of the keywords specifying the service). Usually, the keywords specifying the service is less than 15. As a result, we need more lightweight approach for keyword search on the encrypted data than the existing approach [20].

2.2.2 Anonymous authentication

There are many approaches to solve user privacy and security challenges in ubiquitous computing environment [4, 5, 7, 8, 9, 10, 11, 12, 15, 16, 13, 14]. However, most of these results fall in the scope of establishing general security framework and identifying general security requirements, without providing concrete security protocols. Some work [4, 5, 7, 9, 10, 13] focused on designing specific security infrastructures to protect user context privacy like location information from service providers. Creese *et al.* [8] and Wu *et al.* [11] revised authentication and privacy requirements and Zugenmaier *et al.* [12] showed that the use of a combination of devices using incompatible anonymous mechanisms can compromise the anonymity, which is achieved when each device is used separately.

Jendricke *et al.* [13] introduced an identity management system for ubiquitous computing environment. A user can issue multiple identities and use them depending on the applications. Using these virtual identities, the scheme can protect user privacy while providing access control and user authentication. However, there is no concrete protocol. He *et al.* [14] presented a simple anonymous ID scheme for ubiquitous computing environment. However, this scheme cannot prevent the double spending problem, a kind of replay attack, since there is no verification about the actual holder anonymous ID based on Chaum's blind signature technique [47]. In 2005, Ren *et al.* [15, 16] proposed new scheme supporting a part of the requirements for ubiquitous computing environment. While the scheme prevents double spending problem by combining two cryptographic primitives, blind signature and hash chain, the scheme reduces the number of signature verifications on the authentication server side. Additionally, the scheme provides non-linkability and differentiated service access control and does not rely on underlying system infrastructure such as the "lighthouse" or "mist routers" [5]. However, a mobile user should store all hash chains of his/her anchor to avoid repetitions of the same hash computations and perform one public key operation whenever the user sends a service access request message. Moreover, the service providers may have privacy concern regarding the abuse of their subscription information which is stored in the authentication server to enforce proper access control.

Gruteser and Grunwald [35] offered a method for hiding user's MAC address with anonymous IDs so that the user cannot be tracked in a wireless LAN environment.

2.2.3 Re-authentication of the end-user over wireless sensor network

To allow the end-users to access the deployed sensor network, the end-users and the nearby sensor nodes should share a secret key. In the following, we classify the existing key establishment in the sensor network and introduce the analytical model to illustrate the efficiency of the proposed protocol.

Figure 4.3 shows the typical system model. In this model, the sensor network consists of a base station, several gateways, multiple cluster heads and many sensor nodes. A sensor node, having a battery power, gathers the nearby interesting event (*i.e.*, environmental information, location information for indoor location supporting application, and living human in disaster area), and sends the information to a cluster head. Then, the cluster head aggregates the received information and forwards it to the base station via a gateway. Since the sensor nodes in the same cluster report very similar data compared with other nodes in the different cluster, data aggregation technique is required to extend the lifetime of the sensor network. Also, the base station, which has more computation of power, battery and storage resource compared to the other entities, sends its query or response to the cluster head and sensor node via the gateway. By introducing gateway, we can reduce energy consumption of the intermediate nodes between the target cluster head and base station. In addition, we can reduce

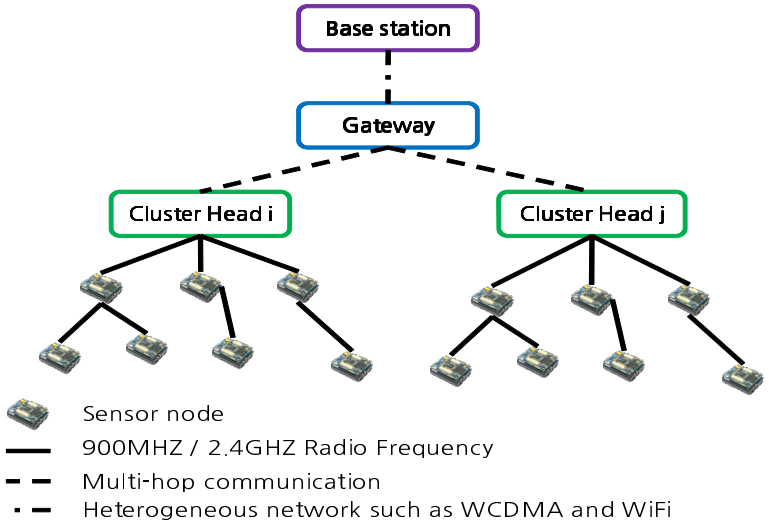

Figure 2.3: System model

transmission delay and packet loss due to congestion close to the base station. That's why many prototype systems support gateway [48, 49]. Hence, this system model can increase lifetime of the sensor network by reducing the number of packet retransmissions. We assume that the channel between the gateway and base station is secure.

User authentication through the deployed sensor nodes

We can classify the existing key establishment schemes [21, 22, 23, 24, 25, 26, 27, 28, 29] for wireless sensor network into random key pre-distribution based approach, master key based approach and trusted party based approach. In 2002, Eschenauer *et al.* proposed a random key pre-distribution [21]. While the proposed scheme is rather simple, it has several weaknesses: tradeoff between network connectivity and storage requirement, low resilience against node capture and scalability. A big storage capacity is required to provide fully connected network topology. When the whole sensor network consists of 10,000 sensor nodes, each sensor node should store 250 keys, about 6K bytes in case of 256-bit key, in the memory to provide 99.8% network connectivity. Reduced the storage requirement, less keys in the memory, indicates a low probability to find the shared key(s). To enhance resilience against node capture and reduce storage requirements several methods using deployment knowledge or symmetric polynomial are introduced [22, 23, 24, 25, 26, 27]. But, some problems still remain. Cooperation among the compromised sensor nodes is not considered in the analysis for resilience against node capture [50]. Random key pre-distribution approach needs more neighbor nodes for better network connectivity. This means that the whole network size should be increased or each node should increase its transmission range which causes more frequent packet collision and communication cost [51].

LEAP (Localized Encryption and Authentication Protocol) is a representative scheme of master key based approach [28]. Using the shared master key and identifiers of its neighbors, each sensor node generates pairwise keys with its neighbors. Compared to random key pre-distribution approach, LEAP provides fully connected network topology with less storage requirements, about 4K bytes when 256-bit key is used. After generating all pairwise keys, each sensor node should erase the shared master

key. But, the problem in this approach is that the adversary can obtain the master key before erasing and generate all pairwise keys in the entire network. In 2005, Hartung *et al.* showed that anyone can get all data within 1 minute using chip-debugging method [52].

Typical example of trusted party based approach is HIKES (HIerarchical Key Establishment Scheme) [29]. In 2007, Ibriq *et al.* proposed HIKES to provide robustness against well-known routing attacks while supporting the authentication and key distribution efficiently. Compared to the above two schemes [21, 28], HIKES needs less storage, about 3K bytes when 256-bit key is used, and enhances resiliency against node compromise. Also, as the network size increases from 1000 to 9000, according to the simulation result in [29], the energy consumption of a cluster head on key management is 3% to 20% while the cluster head in LEACH-type scheme dissipates 13% to 82% energy on key management. That's why we believe that trusted party based approach is more suitable than the other approaches. However, the adversary can reuse the stored key escrow table to guess pairwise keys of neighbors of the compromised node. Also, the adversary gets identifiers of sensor nodes. Still, HIKES requires a large amount of communications for authenticating cluster members, although it aggregates authentication message at the cluster heads.

Analytical model for sensor network

In 2004 Polastre *et al.* [53] proposed an analytical model for a real world monitoring application to present efficiency of their medium access control protocol and validated the analytical model by performing several microbenchmarks. In addition, as the analytical model is independent of the medium access control, the model is useful to analyze the operation of a wireless sensor network application. To illustrate a sensor node's lifetime, the model in [53] computes overall energy consumption of the target sensor node as the follow equation: $E = E_{rx} + E_{tx} + E_{listen} + E_d + E_{sleep}$ where E_{rx}, E_{tx}, E_{listen}, E_d, and E_{sleep} are energy consumption caused by receiving a message, sending a message, listening a message, sampling data, and sleeping.

Note that E_{listen} and E_{sleep} are determined by medium access control protocol. Also, E_d is related to purpose of the target application and is not required for kid's safety care service. Hence, we only focus on E_{rx} and E_{tx}. The energy consumed by transmitting, E_{tx}, is the message length times the message transmission rate.

$$E_{tx} = r \times |m| \times t_{txb} \times c_{txb} \times V, \tag{2.1}$$

where r is message transmission rate, $|m|$ is a length of message m, t_{txb} is time for transmitting 1 byte, c_{txb} is current consumption for transmitting 1 byte, and V is voltage.

Also, the energy consumed by receiving, E_{rx}, is the number of received messages times the message length times the message transmission rate.

$$E_{rx} = z \times r \times |m| \times t_{rxb} \times c_{rxb} \times V, \tag{2.2}$$

where z is the number of received messages from its neighbors, t_{rxb} is time for receiving 1 byte, and c_{rxb} is current consumption for receiving 1 byte.

Note that when we use Mica2 mote having CC1000 transceiver as a sensor node, t_{txb}, c_{txb}, t_{rxb}, c_{rxb}, voltage, and capacity of battery are $416E-6$, $20mA$, $416E-6$, $15mA$, 3, and 2500mAh.

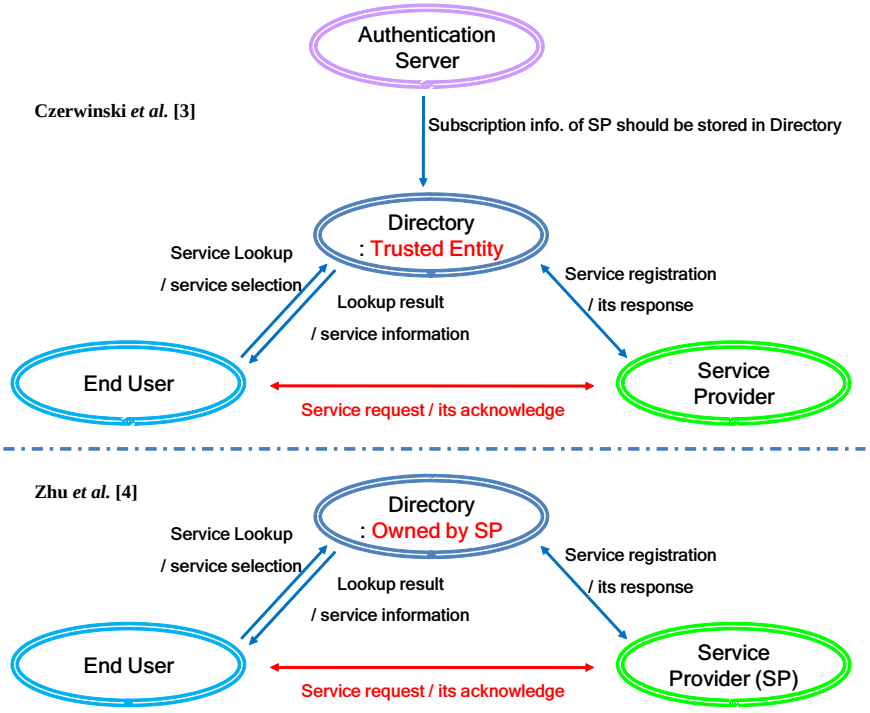

Figure 2.4: System architecture of the existing approaches

2.3 Secure Service Discovery

To address these privacy and security issues, Czerwinski *et al.* [2] proposed a scheme which was called "Secure Service Discovery Service". The scheme consisted of end-user, discovery server, and service provider. Through directory server, regarded as trusted entities, an end-user and service provider authenticated each other. However, they should expose their own identities and service access information during service lookup and service announcement.

In 2006, Zhu *et al.* [3] proposed the PrudentExposure model for a secure service discovery protocol. This approach can preserve the privacy of end-users and service providers based on a Bloom filter. Also, end-users should bind themselves to a nearby agent and transfer all their identities to the agent via a secure channel. However, this approach has several limitations. **First**, additional communication cost is incurred to bind and transfer end-users' identities to an agent. **Second**, privacy leakage occurs among insiders even though the model is designed to preserve sensitive information for an end-user and service provider. **Third**, the end-user should perform two public key encryptions (or decryptions) and one signing operation *whenever* he/she sends a lookup message. Although the agent near the end-user should perform this computation, we need to take into account the cost of removing privacy concern that the nearby agent can identify the user's service selection and obtain all messages between the end-user and service provider as a computational cost of the end-user. **Finally**, each service provider should have his/her own directory server.

Figure 2.4 depicts the system architectures of the previous approaches [2, 3].

Chapter 3. Privacy-Preserving Authentication Framework

In this chapter, we explain the system model for the public ubiquitous computing environments and introduce the scalable privacy-preserving authentication framework consisting of three privacy-preserving authentication protocols.

3.1 Overview

The conceptual system model for the public ubiquitous computing environments consists of four entities: authentication server, end-users, directory servers, and service providers. The authentication server signs or verifies the end-user's pseudonym or service provider's pseudonym. The pseudonym is used to hide the identity of end-user or service provider. The directory server stores the information related to the available services and provides it to the end-users. Since the directory server should be available to all end-users and service providers, the adversary can compromise the directory server in order to obtain the private information of the end-user (or service provider). That's why we regard the directory server as semi-honest entity. In short, the directory server performs its functionality while the directory server collects the private information of the end-user (or service provider). The service providers register their service to the directory server and provide his/her service to the end-users.

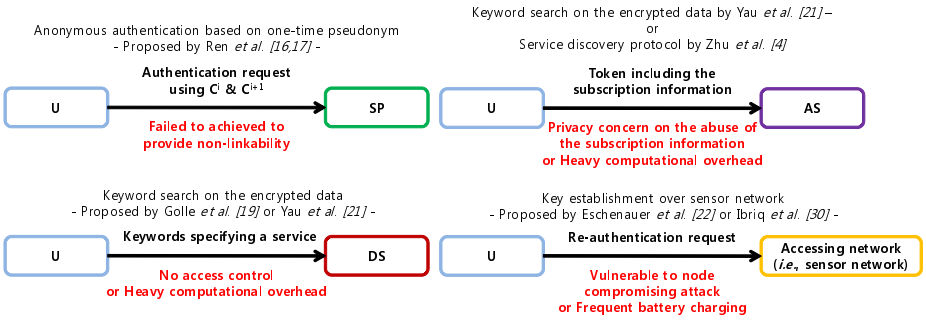

Figure 3.1: The problems of the existing approaches

Figure 3.1 summarizes the problems of the existing approaches in the public ubiquitous computing environments. Due to these problems, we can depict the expected problems in the conceptual model as Figure 3.2.

To address these expected problems, we introduce a scalable privacy-preserving authentication framework. The framework consists of three privacy-preserving authentication protocols removing privacy issues while enhancing scalability issues. The reason why the framework require three authentication protocols is to support three kind of authentications in the public ubiquitous computing environments. To remove the privacy issues, the authentication protocols satisfy the following security requirements: mutual authentication, anonymity, non-linkability, accountability, differentiated access control under semi-honest entity, confidentiality, and integrity as shown in Figure 3.3.

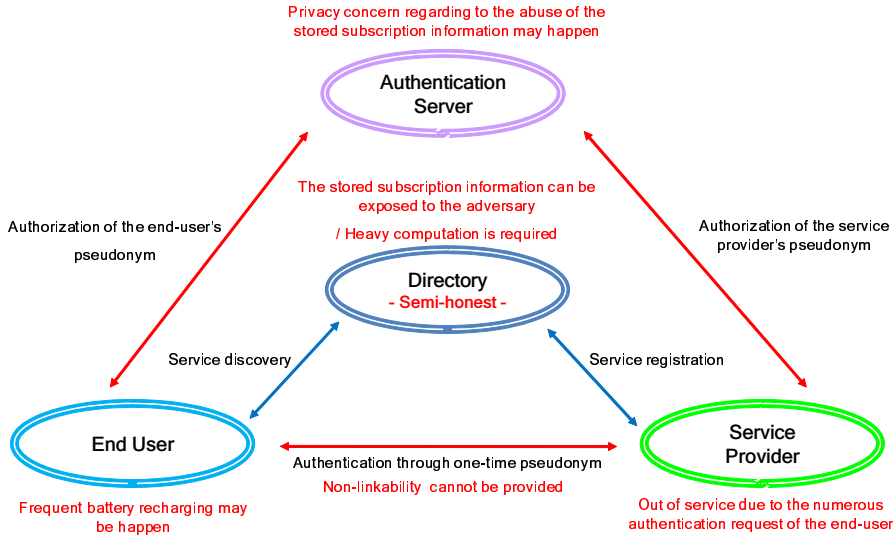

Figure 3.2: Conceptual model with the problems of the existing approaches

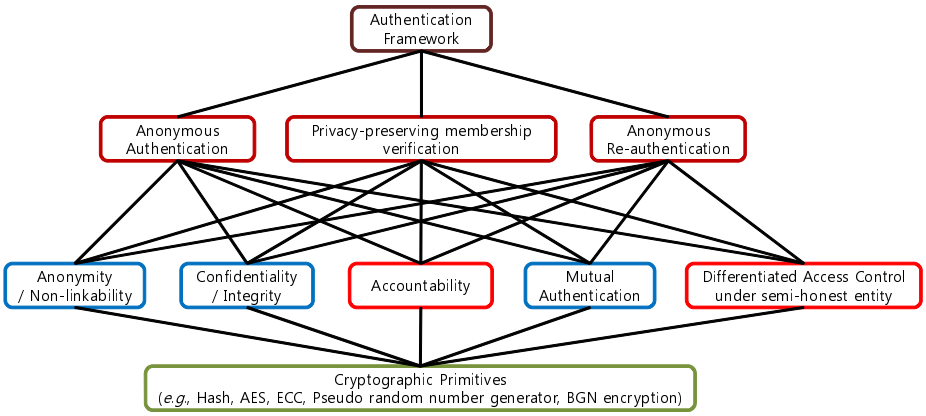

Figure 3.3: The proposed authentication framework

First protocol, called as privacy-preserving membership verification, allows the end-user to authenticate himself/herself to the other entities while protecting his/her privacy. Using our membership verification, the other entities can verify whether the end-user is a subscriber of the requested service. Also, the end-user can find an alternative service from the directory server where the information related to the service (*i.e.*, the keywords, a list of the service subscribers, and access information) is encrypted to protect the information from the adversary. We employ the standard version of the proposed membership verification to enforce proper access control in anonymous authentication and keyword search on the encrypted data in the privacy-preserving secure service discovery protocol. The lightweight version of the proposed membership verification is used to simplify the re-authentication process due to mobility of the end-user.

Second protocol, called as anonymous authentication protocol based on one-time pseudonym, allows the end-user to authenticate himself/herself to the other entities while preserving his/her privacy.

Third protocol, called as anonymous re-authentication protocol, is proposed to provide an efficient re-authentication of the end-users to obtain the sensing data from the deployed wireless sensor network.

3.2 Privacy-preserving membership verification

The goal of the membership verification scheme is to check whether the end-user is a subscriber of the requested service. We convert membership verification to set search by evaluating of a polynomial representing a given set [42, 43], where the set contains the service subscriber list. In other words, our membership verification can allow the verifier (*i.e.*, directory server or authentication server) to authenticate the end-user.

Compared to membership verification cost of the previous work [20], which relies on the number of access keys and subscribers, our membership verification cost is reduced to the number of subscribers. Figure 3.4 compares the polynomial evaluation result of the previous work [20] with ours. In this point, we argue that our membership verification is an efficient approach. Note that access keys in the membership verification are used to derive K_{rk} which is used to encrypt the access information of the target service and can be derived from the list of hidden encryption keys $g^{r-ak_1}, \cdots, g^{r-ak_p}$ using each subscriber's access key ak_i where i is an index of each end-user.

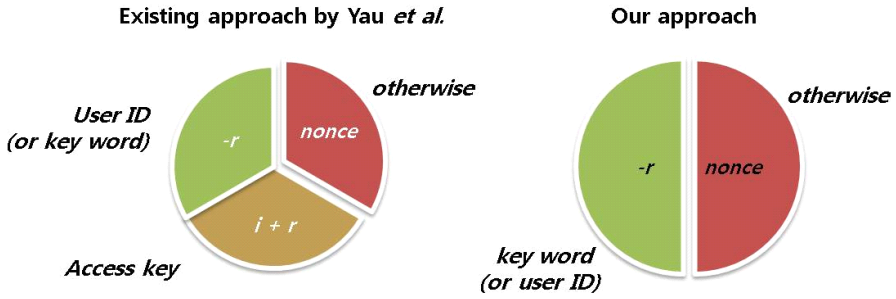

Figure 3.4: Comparison of polynomial evaluation result

We propose two membership verification schemes so that the various devices from the sensor node to the notebook can perform the proposed membership verification. The standard version is useful if the devices owned by the end-user can afford to compute one pairing operation less than a few hundred milliseconds. However, the lightweight version is proposed to provide our membership verification for the resource constrained devices such as the sensor nodes. By removing the heavy computational overhead, the lightweight version of the proposed membership verification can be used to re-authenticate the legal end-users.

3.2.1 Standard version

Figure 3.5 depicts the abstract model of standard version. When the end-user becomes a subscriber of the service provider, the service provider issues $E[i + r, PK_{BGN}, G_1$ and the proper information regarding to the keyword specifying his/her service (or identifier in the subscribers of the service). The service provider registers $E[a_1, PK_{BGN}, G]|| \ E[a_2, PK_{BGN}, G]|| \cdots ||E[a_{n-1}, PK_{BGN}, G]$ to the verifier so that the verifier to check whether the end-user is a subscriber of the requested service. In the polynomial generation, we explain how the polynomial $f(x)$ should be defined. Also, the reason why the proposed protocol can prove that the end-user is a subscriber of the requested service is discussed.

The end-user submits $E[i+r, PK_{BGN}, G_1]|| \ E[1, PK_{BGN}, G_1]|| \ E[w_i, PK_{BGN}, G]|| \ E[(w_i)^2, PK_{BGN}, G]|| \cdots ||E[(w_i)^{n-1}, PK_{BGN}, G] \ ||E[(w_i)^n, PK_{BGN}, G_1]$ to the verifier. Using this information, the verifier

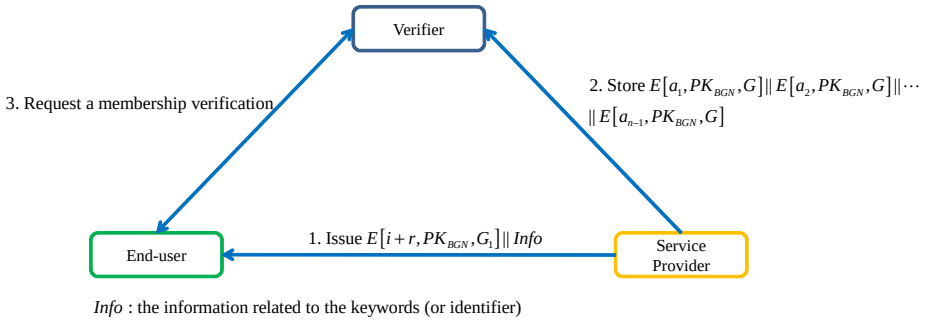

Figure 3.5: Abstract model of our membership verification

performs our membership verification scheme. The detailed evaluation algorithm is explained in the polynomial evaluation.

Polynomial generation

For a set $S_1 = w_1, w_2, \cdots, w_p$, a polynomial with degree t, $f(x)$ for S_1 is defined as the following:

$$f(x) = \begin{cases} -r & x = w_i \in S_1 \\ -r' & x = w_i \notin S_1, \end{cases}$$

where r and $r'(r' \neq r)$ are random integers. Here, $w_i \in \mathbb{Z}_T$ is a user account of i^{th} subscriber if $f(x)$ is used for membership test in AS (or DS). Figure 3.6 presents an example of generating a polynomial.

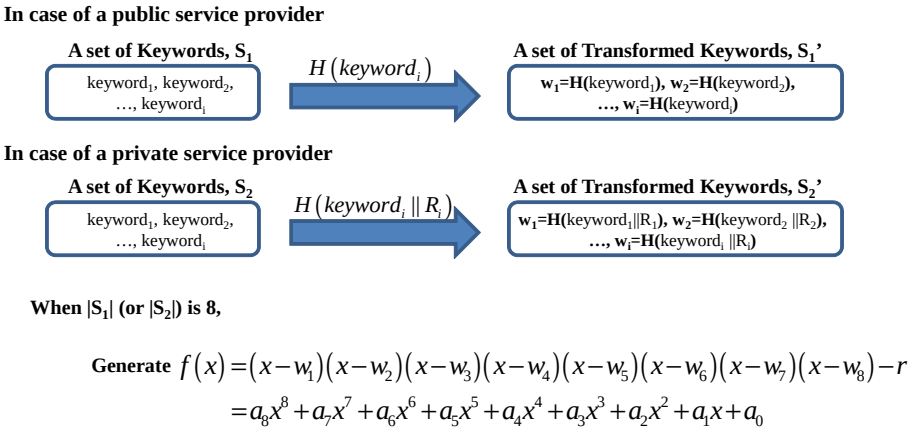

Figure 3.6: An example of polynomial generation

For membership verification, an end-user submits w_i and $i+r$ to the membership verifier. Then, the membership verifier can check whether the end-user belongs to one of the service subscribers by computing $i+r+f(w_i)$. Only if $1 \leq i+r+f(w_i) \leq p$, the user is one of the service subscriber where p is the number of subscribers of target service.

However, we want to prevent the AS and non-subscribers of the service from obtaining the detailed information of the polynomial $f(x)$. That's why the SP encrypts the polynomial $f(x)$' coefficients, $a_0, a_1, a_2, \cdots, a_n$, with public key PK_{BGN} on cyclic group G and $i+r$ with public key PK_{BGN} on cyclic group G_1. Also, the end-user encrypts $1, w_i^1, w_i^2, \cdots, w_i^t$ with public key PK_{BGN} since BGN encryption supports only one multiplicative operation on encrypted data.

Polynomial evaluation

The membership verifier performs the following steps:

1. Set $z = 1$.
2. Compute $C = \prod_{v=1}^{t-1} e(E[a_v, PK_{BGN}, \mathbb{G}], E[(w_j)^v, PK_{BGN}, \mathbb{G}])$.
3. Compute $C' = C \cdot E[a_0, PK_{BGN}, \mathbb{G}_{\mathbb{1}}] \cdot E[(w_j)^t, PK_{BGN}, \mathbb{G}_{\mathbb{1}}] \cdot E[i + r, PK_{BGN}, \mathbb{G}_{\mathbb{1}}]$
4. Repeat the following steps until $z \leq p$.
   (a) If $C'^{(SK_{BGN})} = e(g, g)^{(z \cdot SK_{BGN})}$, return z.
   (b) z = z + 1
5. Return 0.

Using $f(x) = a_t \cdot x^t + a_{t-1} \cdot x^{t-1} + \cdots + a_1 \cdot x + a_0$ and the homomorphic properties of the BGN encryption scheme, we can change $\prod_{v=1}^{t-1} e(E[a_v, PK_{BGN}, \mathbb{G}], E[(w_j)^v, PK_{BGN}, \mathbb{G}])$ to C in the above procedure. By assuming that a_t and a_0 are both 1, C' in the above step (3) is the same as $E[i + r, PK_{BGN}, \mathbb{G}_{\mathbb{1}}] \cdot E[f(w_j), PK_{BGN}, \mathbb{G}_{\mathbb{1}}] = E[(i + r) + f(w_j), PK_{BGN}, \mathbb{G}_{\mathbb{1}}]$.

3.2.2 Lightweight version

The goal of lightweight version is to provide an efficient re-authentication due to the mobility of the end-user. If the end-user has authenticated himself/herself in the previous location, we can simplify the re-authentication process in order to reduce the overhead in the re-authentication process. When the device owned by the end-user has the limited resources, the computation cost of the standard version is too heavy. Moreover, the communication cost is determined by the degree of the given polynomial $f(x)$. From these observations, we propose the lightweight version of our membership verification.

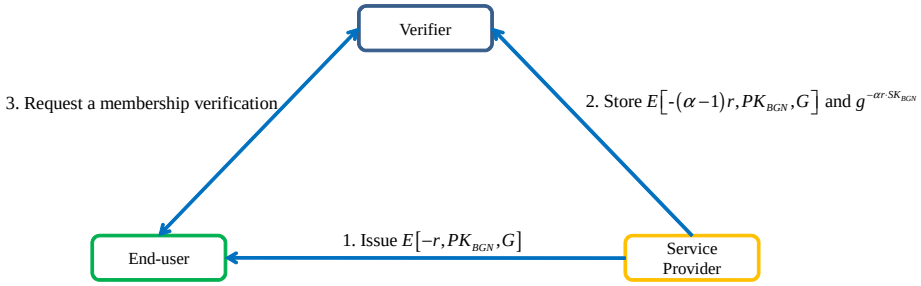

Figure 3.7: Abstract model of our lightweight version

Figure 3.7 depicts the abstract model of lightweight version. When the end-user has authenticated himself/herself to the service provider in the previous location, the service provider issues $E[-r, PK_{BGN}, G]$ to the end-user. Also, the service provider stores $E[-(\alpha - 1)r, PK_{BGN}, G]$ and $g^{-\alpha r \cdot SK_{BGN}}$ to the verifier. In the lightweight polynomial generation, we explain how the polynomial $f(x)$ is defined.

For membership verification, the end-user submits w_i to the verifier. Then, the verifier can check whether the end-user is a subscriber of the requested service through the algorithm in the lightweight

polynomial evaluation. Since w_i is encrypted by PK_{BGN}, the end-user can prevent the adversary from linking up two different sessions to the end-user by multiplying h^r to w_i. Here r is a random number from 0 to $n-1$.

As the lightweight version of the proposed protocol only requires additive homomorphic operation, the other homomorphic encryption schemes [45] can be used. To reduce the storage cost of the security library, we employ the BGN encryption [44].

Lightweight polynomial generation

For a set $S_1 = w_1, w_2, \cdots, w_p$, a polynomial with degree t, $f(x)$ is defined as $f(x) = x \times E[-(\alpha - 1)r, PK_{BGN}, G]$ where α, r, and $r'(r' \neq r)$ are random integers. Here, $w_i = E[-r, PK_{BGN}, \mathbb{G}] = g^{-r}h^{R_i}$ is a user account of i^{th} subscriber issued by the service provider. Note that all integers are 0 to $2^{160} - 1$. If the end-user exists in the set, the evaluation result of the given polynomial $f(x)$ is a fixed value $E[-\alpha r, PK_{BGN}, \mathbb{G}]$. Therefore, we can verify whether the end-user exists in the subscriber list.

Lightweight polynomial evaluation

For membership verification, an end-user submits w_i to the membership verifier. Then, the membership verifier can check whether the user belongs to one of the service subscribers by evaluating $f(w_i)$. Only if the result is $E[-\alpha r, PK_{BGN}, \mathbb{G}]$, the end-user is a legitimate subscriber of the service. However, the evaluated value will be different due to the non-deterministic property of BGN encryption [44]. That's why the membership verifier performs the following steps:

1. Compute $C = f(w_i)$.
2. Compare $C^{(SK_{BGN})}$ with the stored $g^{(-\alpha r \cdot SK_{BGN})}$.
   (a) If the result is the same, return TRUE.
   (b) Otherwise, return FALSE.

3.3 Anonymous authentication protocol

The proposed protocol consists of three main phases: entity registration, entity authentication and service access. In the entity registration phase, the mobile users registers his/her credentials with the authentication server. In the entity authentication phase, the authentication server verifies whether the service requestor has proper legitimacy of the service when the requestor sends his/her service access request to the service providers. Only if the requestor has the legitimacy of the service, the service provider and the user can share a fresh session key to protect their communication channel. In service access phase, the user accesses the service anonymously.

In order to support one-time usage of the authorized pseudonym, we employ a set of the selected number. When the end-user receives his/her initial credential C^0 (or pseudonym) from the authentication server, the end-user generates randomly $2n$ bit array S. Note that n is the access frequency of the end-user, which indicates the end-user can generate his/her pseudonym n times.

Note that if the *i-th* element of the array is 1, it indicates that the number i has been selected. As the array is randomly generated by the end-user, the number of the elements having 0 value is almost n. From this point, our approach allows n time usage of the authorized credential.

Figure 3.8 illustrates the relationship between the authorized credentials. Since all credentials are derived from the initial credential C^0, it is difficult to link up two different credentials to the same end-user.

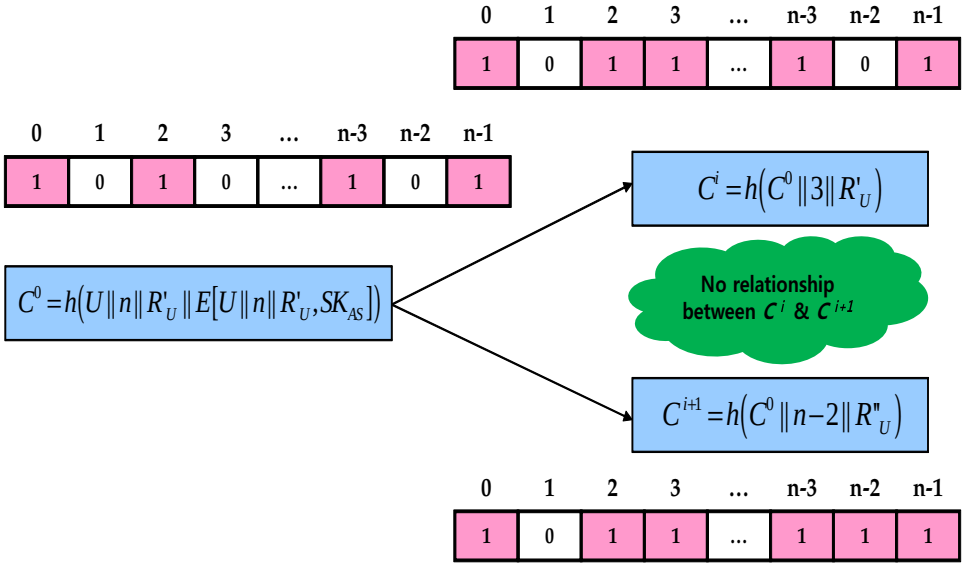

Figure 3.8: Relationship between the authorized pseudonyms

To support the one-time usage of the authorized credentials, the set will be shared with authentication server. Whenever the end-user sends an authentication request using the different credential, the end-user and authentication server update their shared set. Based on this property, we can enhanced level of security. Even if the adversary has C^{i-1}, the adversary cannot impersonate the target end-user due to the shared set of the selected numbers.

As we provide anonymous authentication based on one-time pseudonym, the end-user may repudiate his/her service access. When the end-user repudiates his/her service access, the authentication server requests the end-user to submit $Cert$ $||ID_{User}||n||R'$ $||D[ID_{User}||n||R', SK_{User}])$. Using the certificate of the end-user, the authentication server derives the public key of the end-user and verifies whether the end-user knows the private key. Only if the verification result is correct, the authentication server computes C^0 using the received information and compares the result with the stored C^0. That's why the initial credential C^0 should include $ID_{User}||n||R'$ $||D[ID_{User}||n||R', SK_{User}])$.

Also, we apply our membership verification technique to enforce proper access control in the authentication server. Since a service provider defines several own polynomials presenting subset of subscribers and each subscriber can generate his/her trapdoor MT, presenting his/her access permission on the service, the authentication server verifies whether a requestor belongs to the service subscribers. However, the authentication server cannot obtain the subscription information of the target service, as the polynomials and trapdoors are encrypted by BGN encryption scheme. Moreover, compared to the previous set search [20], our membership verification needs less pairing operations and modular exponentiations. Note that the existing approach in [20] is used to identify whether a requestor knows proper keywords specifying the requested document and has valid access permission on the document.

3.3.1 Entity registration

Entity registration phase consists of three procedures: credential generation, trapdoor generation and credential authorization. In the credential generation procedure, the mobile user generates his/her own credential. As the credential contains a user's signature which consists of access frequency n, his/her identity and a fresh nonce, the proposed protocol can provide non-repudiation. In the trapdoor

generation procedure, the user his/her own trapdoor indicating the user has proper access permission on the service. In the credential authorization procedure, the user register his/her credential with the authentication server. Only if the user is a legal entity having proper access permission on the service, the authentication server authorizes the received credential. To hide the relationship between the authorized credential and the mobile user's real identity we apply blind signature technique.

Credential generation: The mobile user generates two fresh nonces and signs his/her identity together with one fresh nonce R' using own private key SK_{User}. Then, the mobile user computes an anchor value C^0 with the signature. Note that the procedure can be done off-line. We summarize it as:

1. Generate two fresh nonces: R' and R''
2. Sign user's own ID with a fresh nonce $R' and n$:

$$D[ID_{User}||n||R', SK_{User}]$$

3. Compute the anchor value C^0 of credential chain as:

$$C^0 = H(ID_{User}||n||R'||D[ID_{User}||n||R', SK_{User}])$$

4. Blind C^0 as $C_{User} = E[R'', PK_{AS}] \times C^0$

Trapdoor generation: When the user becomes one of the service subscribers, the user can receive necessary information for service access (*i.e.*, SID, polynomial degree p, access key ak_i, $E[i + r, BGN_{pk}]$ and user account w_i) from the service provider. Using this information the user can generate his/her trapdoor, $MT = \{E[(i + r), BGN_{pk}]|| \ \{E[(w_i)^1, BGN_{pk}], \cdots, E[(w_i)^p, BGN_{pk}]\}\}$ where pk is public key of the authentication server. Due to the characteristic of BGN encryption, the expected decryption time using Pollard's lambda method is $\tilde{O}(\sqrt{|m|})$ [44], the user can hide his/her user account from the authentication server although the server has the private key $vk = q_1$ to message decryption.

Credential authorization: After credential generation procedure and trapdoor generation procedure, the mobile user sends his/her own identity, blinded credential C_{User}, SID, trapdoor MT and his/her certificate to the authentication server. To verify whether the user is a legal entity having proper access permission on the service, the authentication server verifies the received certificate with PK_{AS} and checks whether the user is a subscriber of the requested service through membership verification procedure. The authentication server signs on the received credential C_{User} with SK_{AS} and sends response message to the user only if the user has valid certificate issued by the authentication server and the membership verification procedure returns 1. After receiving the response message from the authentication server, the user verifies the received ID_{User} and ID_{AS}. When the verification result is successful, the user can obtain a valid authorized credential $E[C^0, SK_{AS}]$ by unblinding the received C_{Signed}. Otherwise, the user discards the response message and retries this procedure. Figure 3.9 illustrates this procedure. As the authentication server is trusted third party and does not store any information obtained in this procedure, the server cannot identify the relationship between the user's identity and the authorized credential in entity authentication phase.

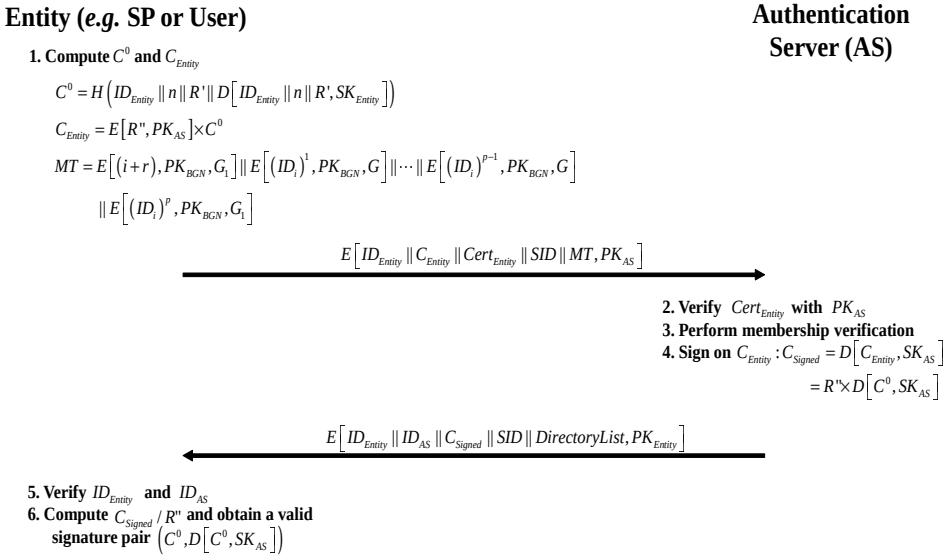

Figure 3.9: Authorization of credential information

3.3.2 Entity authentication

Entity authentication phase consists of three procedures: access request, credential verification and key establishment. In the service access request procedure, a mobile user sends his/her access request to the service provider. Since the relationship between the authorized credential and the mobile user's real identity is hidden through blind signature technique, the service provider forwards the request to the authentication server. Only if the mobile user has the legitimacy of the target service, the mobile user generates correct C^0, C^{i-1} and S^i in the entity registration phase. Also, the authentication server stores C^0, C^{i-1} and S^i to the database server only if the authentication server verifies that the credential is authorized. In this point, the authentication server can authenticate mobile user. In the credential verification procedure, the authentication server verify whether the mobile user is a legal entity having proper access permission on the service and forwards the verification result to the service provider. In the key establishment procedure, the service provider and the mobile user can establish a fresh session key to protect their communication channel.

This processes are shown in Figure 3.10.

Access request: When the mobile user sends an *i-th* access request to the service provider, the mobile user generates a fresh nonce R^i and selects one random number j between 0 to $l-1$. Next, the user checks that j is not in the list S by comparing *j-th* index of S is 0. If j is in S, the mobile user should select unused random number. Then, the user generates one-time credential as $C^i = H(C^0||j^i||R^i)$. Also, both entities mobile user and AS share a secret key $K_{U,AS}$ by computing as:

$$K_{U,AS} = \begin{cases} H(C^0||PK_{AS}||R^1||j^1||SID) & \text{if } i = 1 \\ H(C^0||C^{i-1}||SID) & \text{otherwise} \end{cases}$$

Note that both entities (*i.e.*, mobile user and authentication server) can easily generate a fresh session key $K_{U,AS}$ since they shared the requisite information (*i.e.*, the anchor value and S) and the mobile user discloses her previous credential information C^{i-1}.

The service provider forwards the received request message to the authentication server with a

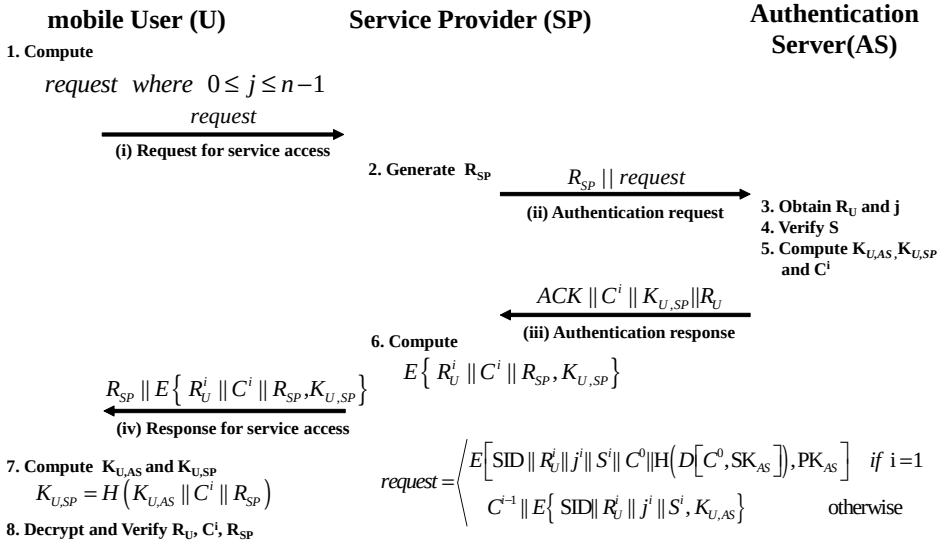

Figure 3.10: Entity verification when the user sends the *i-th* access request

fresh nonce. After decrypting the request message, the AS checks duplication and validation of the secret information, C^{i-1} and S.

Credential verification: In order to verify the received authentication request, the following two cases are considered.

1. First access request: After decrypting the *request* message using SK_{AS}, the authentication server computes $H(D[C^0, SK_{AS}])$ and compares the result with the received one. Only if the result is same, the server believes that the entity has an authorized credential. Then, the server computes $C^1 = H(C^0||j^1||R^1)$ and stores SID, S^1, C^0 and C^1 to the database server. Otherwise, the server discards it.

2. *i-th* request: The authentication server sends a query message, finding C^0, S^{i-1} and SID, to the database server using the received C^{i-1} and decrypts the received message with $K_{U,AS}$. Next, it verifies that the user has the same set of the selected numbers by checking *j-th* index of the stored S^{i-1} is 0 and comparing the derived S^i, flipping *j-th* index of the stored S^{i-1}, with the received S^i. Only if the verification result is correct, the authentication server believes that the user has legitimacy of the service and stores the received information C^i and S^i to the database server. Otherwise, the authentication server discards it. If there are several verification failures on the series of the authorized credentials, the server can request the user to change his/her credential or notify that there is an impersonation attack. When the user is a legal entity with proper access permission, C^i and S^i are stored as the authorized credential and the selected number list, respectively. As a result, the authentication server can verify whether the user has authorized credential using the received C^{i-1}.

3.3.3 Key establishment

After verifying validation and duplication of the request message, the authentication server computes $K_{U,SP}$ which is used to secure communication between the service provider and the mobile user. Next, the authentication server sends C^i, $K_{U,SP}$ and R_U^i to the service provider. The service provider

encrypts received information with a fresh nonce R_{SP} by using $K_{U,SP}$. Through this activity, the proposed protocol provides explicit key authentication between service provider and mobile user.

After computing $K_{U,AS}$ and $K_{U,SP}$, the mobile user decrypts the received access acknowledgement and verifies C^i, R_U^i and R_{SP}. If the verification result is correct, the mobile user can access the target service of the service provider. Otherwise, the mobile user resend the access request to the service provider.

3.3.4 Extension for out-of-order requests

Sometimes the mobile user might want to request multiple services simultaneously. If the multiple concurrent sessions are handled by a single server, the access request messages may arrive out of order at the authentication server due to unexpected network problems. To deal with this problem we adapt a sliding-window-based extension to the credential verification and key generation procedure on the authentication server side. We assume that the database server has the stored credential list, the selected number list, the nonce list and the encrypted message to deal with our-of-order requests. There are two cases to deal with out-of-order requests:

1. When the authentication server cannot find C^{i-1} in the authorized credential and the stored credential list
   (a) Store C^{i-1} and $E\{R_U^i||j^i||C^0, K_{U,AS}\}$ to the stored credential list and the encrypted message respectively.
2. When the authentication server find C^{i-1} in the authorized credential and the stored credential
   (a) Send a query message to the DS by setting C^{i-1} and SID as searching condition for getting proper S^{i-1} and C^0.
   (b) Compute $K_{U,AS}$ and decrypt the received message.
   (c) Flip the *j-th* index of the stored S^{i-1} only if the index is set as 0. Otherwise, discard it.
   (d) Update C^i in the authorized credential and generate C^{i+1}. Next, search the generated credential in the stored credential list. If the C^{i+1} are found in the stored credential list then repeat 2.(a)- 2.(d) steps until the searching has failed or the stored credential list has empty.

Figure 3.11 shows our approach handling the out-of-order requests.

Service access phase

Through the entity authentication phase, the service provider and the mobile user can establish a fresh session key when the user is one of subscribers in the service. Using the shared key, the user can access the service while hiding his/her identity.

3.4 Anonymous re-authentication protocol for an end-user over wireless sensor network

The proposed protocol consists of four phases: token authorization, entity registration, entity authentication, and token update phase. In the following, we describe our protocol in detail.

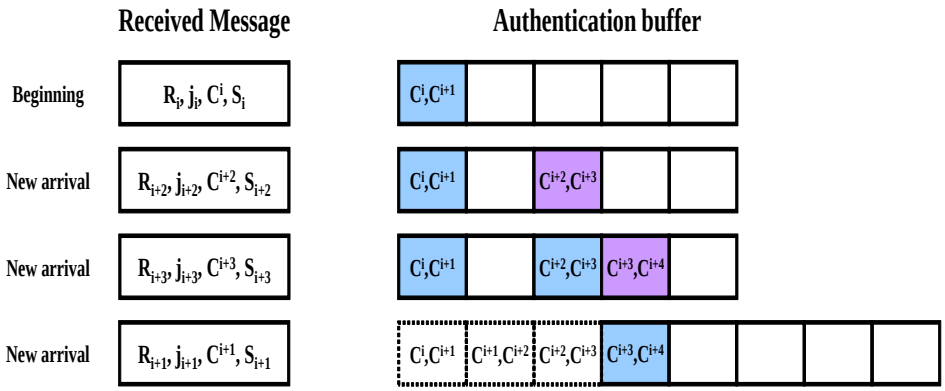

Figure 3.11: Illustration of our extension for out-of-order request

3.4.1 Token authorization

When an end-user becomes a subscriber of the target service provider, the end-user obtains proper authenticated token w_i from the service provider. To issue the token w_i, the service provider selects two random integers R_i and r $(0 \leq r \leq 2^{160} - 1)$. Using two random numbers, the service provider computes $w_i = E[-r, PK_{BGN}, \mathbb{G}] = g^{-r} \cdot h^{R_i}$ for each subscriber. Since anonymous communication between the end-user and service provider is out of scope, we simply assume that the communication between the end-user and service provider is anonymous and secure. In addition, the ciphertext of the same plaintext will be different due to non-deterministic property of the BGN encryption [44]. Figure 3.12 shows token authorization phase.

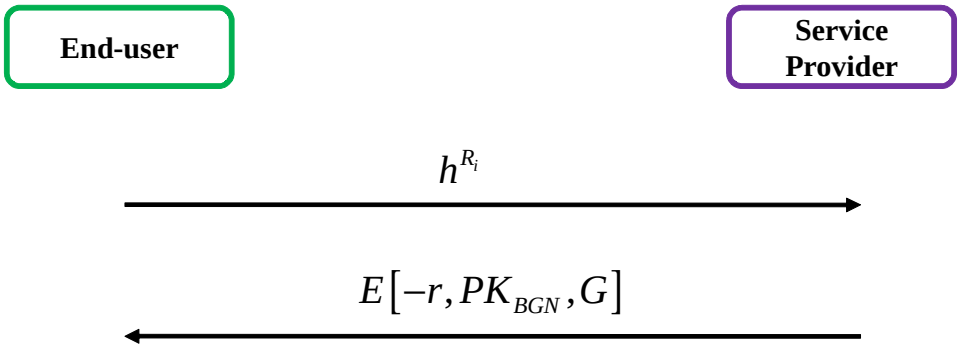

Figure 3.12: Token authorization phase

3.4.2 Entity registration

After token authorization phase, the end-user should register him/her with the base station in order to obtain nonce R_{BS}. The base station verifies whether the end-user is a legitimate subscriber of the target service through membership verification. Only if the verification result is true, the base station sends R_{BS} to the end-user.

3.4.3 Entity authentication

In the entity authentication phase, the end-user having the mobile node and cluster head establish $K_{U,CH} = H(R_{BS} + 1||R_U)$. Using this key, the communication between the end-user and cluster head can be secure. Since R_{BS} is only known to the cluster head and end-user, they can share a secret key $K_{U,CH}$. We employ HMAC in order to support message integrity. Figure 3.13 depicts entity authentication phase.

Compared to the previous authentication protocols [21, 22, 23, 24, 25, 26, 27, 28, 29], each cluster head in our re-authentication protocol should compute only two modular exponentiations, four hash operations, and two symmetric operations. Without any help of the nearby sensor nodes, the cluster head can verify whether the end-user is a legitimate subscriber through membership verification. As the computation cost in wireless sensor network is less than communication cost, our re-authentication protocol is more efficient than the previous protocols.

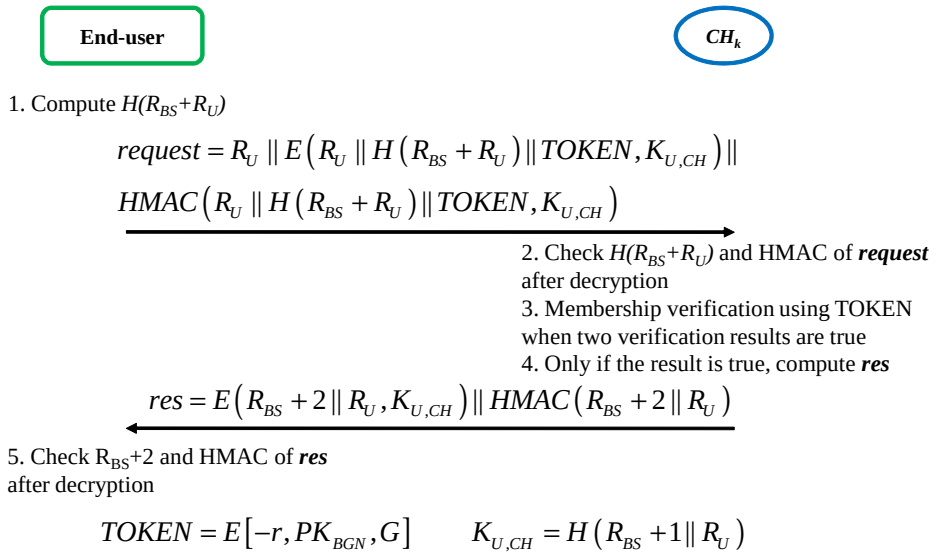

Figure 3.13: Entity re-authentication phase

To reduce the communication cost of entity authentication, the end-user can send the x-coordinate of $E[-r, PK_{BGN}, \mathbb{G}]$ to the nearby cluster head. Since PK_{BGN} is a point on cyclic group G, the encrypted result is also one of points on cyclic group G where a point consists of x-coordinate and y-coordinate. To prevent abuse of the eavesdropped token by the adversary, the cluster head sends the token to the base station. Then, the base station requests the end-user to update the token.

3.4.4 Token update

To prevent illegal activity tracking from the adversary, the end-user should change his/her authorized token after entity authentication phase. Due to non-deterministic property of BGN encryption [44], we can enable the end-user to change the previous token while allowing successive membership verification. Recall that the message encryption m_i in BGN encryption is $g^{m_i}h^{r_i}$ where r_i is a random number. When the end-user multiplies h^{α} to the authorized token, the new token is changed to $g^{m_i}h^{\alpha}h^{r_i}$. In membership verification procedure, however, $h^{\alpha}h^{r_i}$ will be converted to 1. Therefore, the end-user can update his/her token while allowing successive membership verification. After updating

the token, the end-user should report h^{α} to the service provider to provide accountability.

Chapter 4. Case Studies

In this chapter, we employ the proposed authentication framework to provide one of the possible services, child-care and safety service, in the public ubiquitous computing environments and build the privacy-preserving secure service discovery protocol. By showing that the service can preserve the privacy of the end-user while requiring less computational overhead and communication cost, we demonstrate that the proposed framework can be used to address security and privacy issues in the public ubiquitous computing environments. Also, we illustrate how to employ the proposed framework to build the privacy-preserving protocols, useful to address security and privacy issues in the public ubiquitous computing environments.

4.1 Child-care and safety service

According to the survey result for cognition against social safety [54] in Figure 4.1, Compared to 10 years ago, 61.4% people in Korea realize that safety level changes to unsafe. Also, as a heinous crime against kid increases, many parents worry about their kid's safety during commuting to a school or playground near their home. Moreover, 54.1% people expect that the safety level of our society becomes deteriorated in near future. When the children plays on the schoolyard, the teacher in charge cannot observe all activities of the children in his/her class. If the teacher can receive the status of the children (*i.e.*, high fever, heart failure, and dangerous location), the teacher can preserve the safety of the children. From this observation, child-care and safety service should be provided in ubiquitous elementary school.

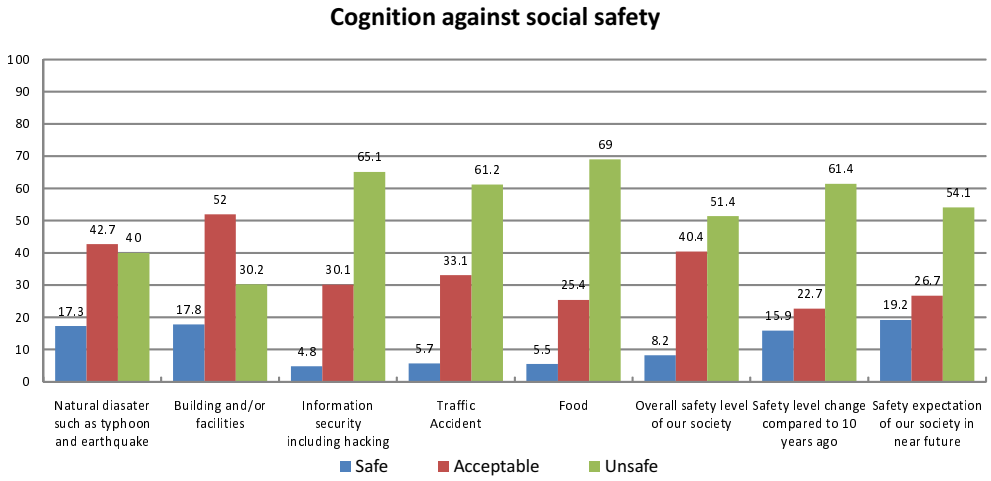

Figure 4.1: Cognition against social safety [54]

The goal of this service is to provide child-care and safety from any dangerous situation of the children. In order to provide this service without disturbing proper mobility of the children, wireless communication is required. Although the current status information of the children should be sent

to the corresponding teacher and parents, the information does not exposed to the adversary for the privacy of the children. As the children cannot identify whether the handheld device is expensive and fragile, the expensive device such as notebook and i-Pad cannot be used in this environment. The various networking technologies such as sensor network, Wi-Fi, 3GPP, and LAN are required to establish ubiquitous computing environment. The sensor network is deployed to monitor any change of the nearby environment. The parents can obtain the weather information from the device hold by their children using the sensor network so that the children are able to avoid the rain. In addition, the sensor network can detect any smoke near to the school and send fire alarm to the teachers. That's why the sensor network is deployed in the ubiquitous elementary school.

4.1.1 System architecture

Conceptual model

The following conceptual model consists of four components such as end-user (*i.e.*, a parent having his/her child), location information provider, service provider, and device. When the end-user becomes one of the legitimate subscribers of the child-care & safety service provider, the end-user obtains an authorized credential for the service from the service provider. Whenever the user wants to receive location information of his/her child, the end-user should register the service with the location information provider and request periodical location report of the child. Then, the location information provider forwards the information to the end-user where the information is determined by various techniques (*i.e.*, A-GPS [38], ultrasound [39] and RSSI based on RF [40, 41]). By sending a periodical location query to the base station, the end-user can obtain the location of his/her child and verify the risk of the child.

The location information of the child is encrypted by the shared key between the child and end-user so that the location service provider cannot obtain the exact location information. When the end-user wants to stop the service, the user can deregister the service. Therefore, this conceptual model can provide the end-user with a capability controlling over the service. Figure 4.2 depicts these activities.

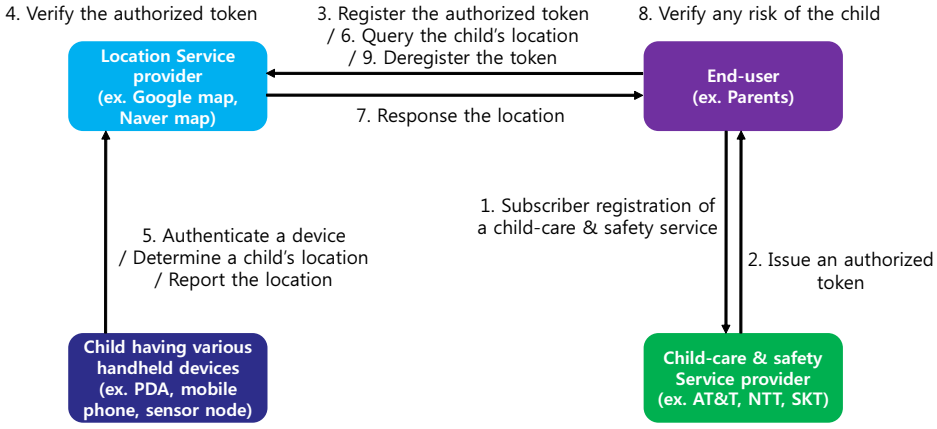

Figure 4.2: Our abstract model for child-care & safety service

Compared to the previous model, this approach has the following advantages. We can prevent the service provider from obtaining the private information of an end-user (*i.e.*, location of his/her kid, safety zone and frequent visiting place). Because the service provider can only issue an authorized

credential used to preserve the anonymity of the end-user. Whenever the user wants the child-care & safety service, the end-user can register and deregister the service.

As the end-user may have various devices and want to experience by a seamless location supporting service, we should support various location determination techniques to deal with mobility of the end-user.

System architecture over sensor network

We choose the sensor network to be our location determination technique due to the following observations. As sensor network will be deployed to monitor nearby environmental condition in u-City, our system model can reuse the existing infrastructure. In addition, a sensor node can support various cryptography primitives such as symmetric key encryption, asymmetric key encryption including pairing computations with low cost compared to PDA, mobile phone, and wireless access point. While location determination based on sensor network can be used for indoor or outdoor, it is more accurate compared to the other techniques such as A-GPS [38] and ultrasound [39]. When the location determination technique is changed to RSSI based on Wi-Fi or UWB, our system model does not require any modification.

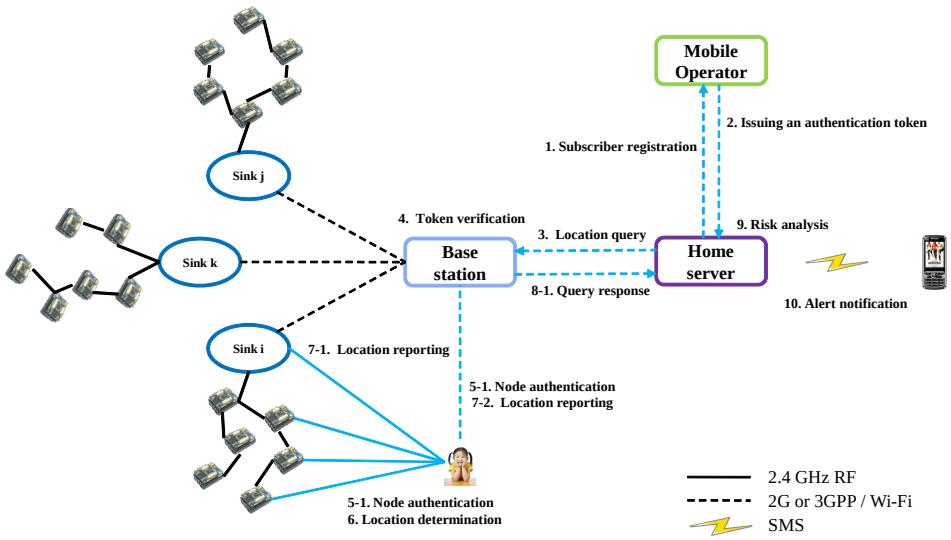

Figure 4.3: Our system model

Figure 4.3 shows our concrete system model. In this model, a sensor network consists of sink nodes, sensor nodes, and a base station. A sensor node, having a battery power, gathers the nearby environmental information and sends the information to a sink node. Then, the sink node, having a permanent power and a capability of direct communication with the base station, aggregates the received information and forwards it to a base station. Thus, the administrator in the base station only requires recharging a node's battery or redeploying another sensor node whenever the battery of the sensor node is exhausted. In addition, this approach can reduce unnecessary energy consumption of the intermediate nodes between a child's device and base station, caused by forwarding authentication or service requests to the base station. As a result, our system model can increase lifetime of the sensor network.

The base station verifies whether the end-user is one of legitimate subscribers or not. To support

this verification, we adopt our anonymous authentication technique based on blind signature which is discussed in Chapter 4. Only if the end-user is legitimate and registers location query of his/her child, the base station stores the received authorized credential and the child's location in its database where the location information is encrypted with a shared key between the child's device and end-user's home server. Note that the base station may be owned by the mobile operator or local government as the sensor network will be deployed in near future. To prevent the base station from obtaining the location information of the device, we employ anonymous authentication as the device authentication. As a result, the base station can only where an end-user's device connects with the sensor network. Still, the base station cannot identify the end-user.

Also, by increasing transmission power, a device of an end-user's kid can directly communicate with the sink node via another radio frequency, which is not used by the communication between the sensor nodes and the nearby sink node. To support scalability, we should consider that the inhabitants in a city can access the sensor network using several devices. If we do not reflect this situation, the sensor network cannot achieve its primary goal such as monitoring the nearby environment but forwarding the received authentication request. As the child's device takes a role of location determination and identify his/her location from the periodic or event message of the legitimate sensor nodes, the sensor nodes do not need any additional cost such as computations or communication. Thus, this system model can maximize the usage of sensor network.

Finally, our system model includes a home server of an end-user to preserve the end-user's privacy as the system proposed by Takata *et al.* [37]. Using the location information received from the base station, the home server takes a role of identifying whether the end-user's child is in safety zone. The home server periodically sends a query message including the registered credential in location query phase. Only if the query message includes the registered credential and the message is encrypted with the shared key between the base station and end-user, the home server can receive the child's location information from the base station.

System architecture over Wi-Fi network

Since the various Wi-Fi based smart phones are popular and the mobile operators have their own Wi-Fi network, we illustrate an example scenario with Wi-Fi based mobile device in Figure 4.4.

Compared to the system architecture in Figure 4.3, the only differences are that the mobile operator takes a role of location service provider and the device cannot directly communicate with the mobile operator. Because we should prevent the mobile operator from identifying the end-user although the mobile operator can identify the device owned by a child.

4.1.2 Brief sketch

This service consists of subscriber registration, location query, device authentication, location determination, query response and location information transmission.

Subscriber registration

In the subscriber registration phase, an end-user generates an authentication token and sends the token to a mobile operator providing child-care and safety service. Only if the end-user is a subscriber of the mobile operator, the mobile operator authorizes the received authentication token. Through the

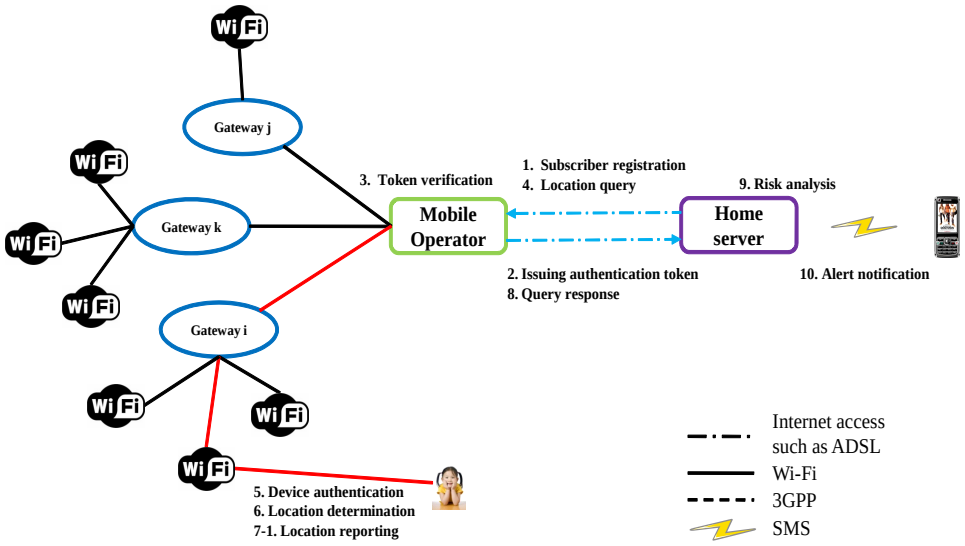

Figure 4.4: An example scenario with Wi-Fi based mobile device

entity registration phase in 3.3, we can preserve the privacy of the end-user while verifying the access permission on this service.

Location query

In the location query phase, the end-user registers his/her authorized token with the base station in order to obtain the child's location. By registering or deregistering the token with the base station, the end-user can control this service so that the mobile operator or base station cannot track the child's location illegally. In order to verify the authorized token, the base station performs the entity authentication phase in 3.3. Only if the end-user registers the token, the device owned by the child can share a secret key with the base station.

Device authentication

Using the shared key with the base station, the device owned by the child can authenticate with the base station and share a fresh session key with the nearby sensor nodes (or access points). To provide better scalability in the wireless sensor network, the re-authentication protocol for an end-user can be used. When the device supports a capability of accessing Wi-Fi or 3GPP network, the device directly sends the token to the base station. Since the bandwidth of Wi-Fi and 3GPP network is higher than the sensor network, this approach can reduce the processing delay and support better scalability.

Although the base station has the shared key with the child's device, the base station cannot identify the owner of the device. In location query phase, the base station only verifies the legitimacy of the received token. When the token is authorized by the mobile operator, the base station stores the shared key and authorized token in its database. Therefore, we believe that our protocol can support privacy of the end-user by hiding the relationship between the device and end-user.

Location determination

Using triangulation based on RSSI [40] from three or more legitimate sensor nodes, the device can determine its location within 3 meters. After obtaining the location information, the device owned by the child send the information to the base station with the help of the nearby sensor nodes (or access points). When the device can directly access the base station through Wi-Fi and 3GPP, the result message is directly sent to the base station. As a result, the nearby sink node can handle more messages from the child's device by reducing the processing time of the received message in the nearby sink node. That's why we believe that our approach can support better scalability. The base station stores the received information if the message, sent by the device including the location of the child, has the authorized one-time pseudonym. The verification of the pseudonym is explained in 3.3.

Query response

To prevent the base station from identifying the end-user, the end-user should continuously send the query regarding his/her child's location to the base station. Only if the received token is valid and has stored in the database, the base station sends the stored location information of the child. The information is encrypted by the shared key with the end-user.

Location information transmission

After receiving query response from the base station, the home server can identify the child's location. If the location is a dangerous area, the home server notifies an alerting message to the end-user's mobile phone. When the end-user wants to observe kid's location periodically, the home server can send the location information to the end-user's mobile phone.

4.1.3 Implementation

In this section, we explain how to employ our privacy-preserving authentication protocols in the child-care and safety service. We apply our authentication protocol based on one-time pseudonym to the anonymous authentication.

Our protocol for child-care and safety service consists of subscriber registration, location query, device authentication, location determination, query response and location information transmission. Before describing each phase in detail, we need some assumptions.

The end-user distributes a fresh session key to his/her home server and kid's device to authenticate themselves to the base station.

The base station has a public key of the mobile operator and its certificate to verify the authorized token of an end-user. In addition, the base station distributes a fresh session key K_{Init}, used to support message integrity in device authentication phase, to all sink nodes in the sensor network. Although this approach is vulnerable to node compromise attack, an adversary having K_{Init} cannot deplete the batteries of other sensor nodes due to direct communication between a sink node and base station.

All sensor nodes broadcast their location information in periodic data reporting message. Although this approach increases additional two bytes of transmitting message, it can enable an administrator of the base station to identify which sensor nodes should be recharged. In addition, this approach can reduce communication cost of a child's device to determine his/her location.

The end-user's child has a device equipped with communicating the sensor nodes such as Mica2, Mica2dot, and Telosb. Moreover, the device communicates with its nearby sink node using the different radio frequency compared to the used radio frequency in the sensor network. This assumption can reduce a possibility of message collision between the legitimate sensor nodes and end-users' child devices.

Subscriber registration

In subscriber registration phase, an end-user generates an authentication token and sends the token to a mobile operator providing child-care and safety service. Only if the end-user is a subscriber of the mobile operator, the mobile operator authorizes the received authentication token. We illustrate this procedure in Figure 4.5. To provide anonymous authentication for preserving an end-user's privacy, we adopt our privacy-preserving authentication protocol which is proposed in the previous chapter. The proposed protocol provides novel properties (*i.e.*, enhanced security level, accountability, and non-linkability) while reducing the cost of communications and computation.

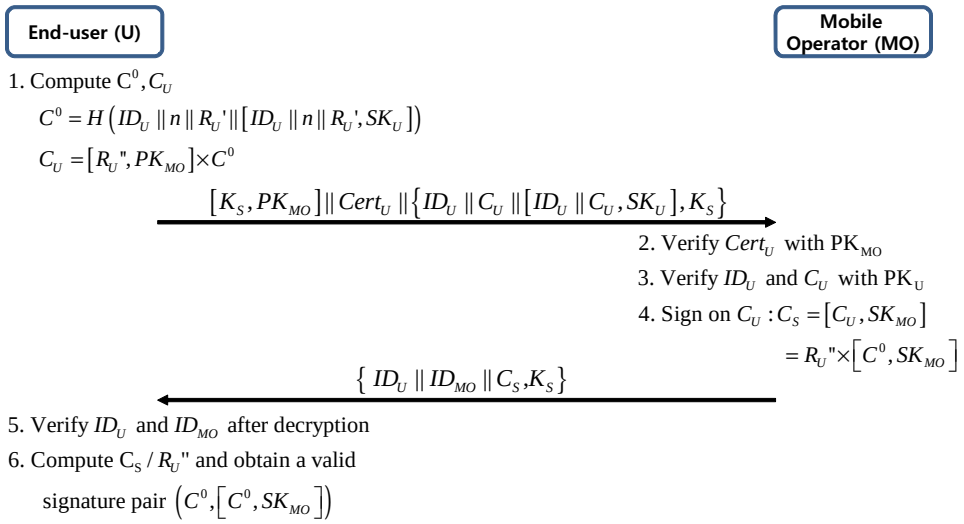

Figure 4.5: Subscriber registration phase

The end-user generates two fresh nonces and signs his/her identity together with one fresh nonce R' using own private key SK_U. Then, the end-user computes an anchor value C^0 using the signature. Note that this procedure can be done off-line.

When a mobile operator receives a request for subscriber registration, the mobile operator verifies the received certificate $Cert_U$, end-user's identity ID_U, and anchor value C^0 using SK_{MO} and PK_U. Only if the request has proper private key SK_U and certificate $Cert_U$, the mobile operator computes $C_S = E\{C_U, SK_{MO}\}$ and sends $E\{ID_U||ID_{MO}||C_S, K_S\}$ to the end-user. After then, the end-user verifies the received ID_U and ID_{AS} and compute C_S/R_U" to obtain a valid signature pair $(C^0, D[C^0, SK_{MO}])$.

As the legitimate end-user can provide proper $Cert_U$ and the knowledge of SK_U to the service provider, the mobile operator can authenticate the end-user. However, an adversary cannot obtain K_S, C^0, and the knowledge of SK_U due to the message encryption PK_{MO}. Also, the end-user can authenticate the mobile operator with K_S. The key K_S is shared with the legitimate mobile operator

having SK_{MO}. Hence, we believe that our protocol can support mutual authentication between the end-user and service provider.

Location query

In location query phase, the end-user registers his/her authorized token with the base station. In addition, the end-user can control child-care and safety service by registering or deregistering his/her token to the base station. This approach can remove the end-user's concern about illegal tracking by the mobile operator or base station. Figure 4.6 depicts location query phase.

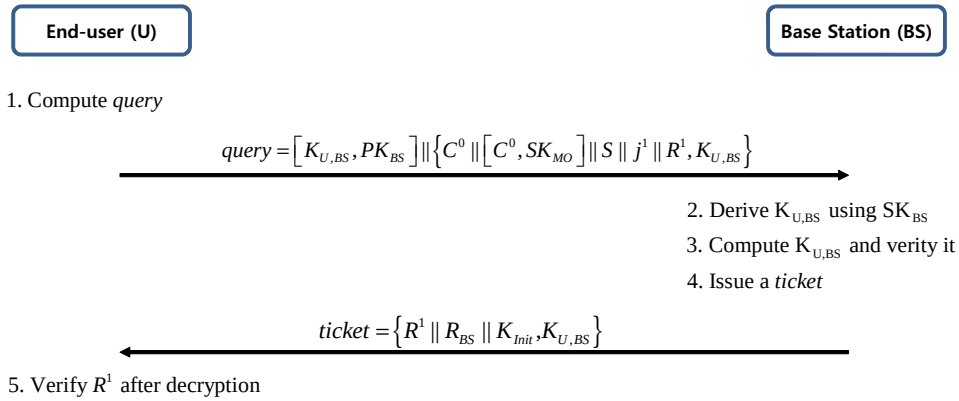

Figure 4.6: Location query phase

For location query, the end-user randomly generates a fresh nonce R^1 and a set of selected numbers S, expressed as l-bit array. If the i-th value of S is 1, it indicates that i is already selected. Also, the end-user selects one random number j^i between 0 to $l-1$ until j^1-th value of S is 0. Next, the end-user computes one-time credential $C^1 = H(C^0||j^1||R^1)$ and a session key $K_{U,BS} = H(C^0||PK_{BS}||R^1||j^1)$. Then, the end-user sends a query message $E[K_{U,BS}, PK_{BS}]$ $||\{C^0||E[C^0, SK_{MO}]$ $||S||j^1||R^1, K_{U,BS}\}$ to the base station.

After decrypting the received query message with SK_{BS}, the base station derives $K_{U,BS}$ and obtain necessary information (*i.e.*, C^0, R^1, and j^1) to compare a computed $K_{U,BS}$ with the derived one. Only if the verification result is correct, the base station computes $C^1 = H(C^0||j^1||R^1)$, sends a ticket $E\{R^1||R_{BS}||K_{Init}, K_{U,BS}\}$ to the end-user, and stores the derived information (*i.e.*, C^0, R^1, j^1, C^1, R_{BS}, and $K_{U,BS}$) in its database.

After decrypting the received ticket, the end-user verifies whether the derived R^1 is the same as stored R^1. If the verification result is correct, the end-user stores K_{Init} to his/her child's device. Otherwise, the end-user retries this phase. After registration of location query, the end-user can be ready to receive the location information of the child. Whenever the child's device performs location determination procedure, the device sends location information to the base station via its nearby sink node.

Device authentication

Device authentication is required to share a fresh session key, K_S, between child's device and its nearby sink node. Using the key K_S, the device can securely communicate with its nearby sink node and authenticate itself to the sink node without any participation of the base station.

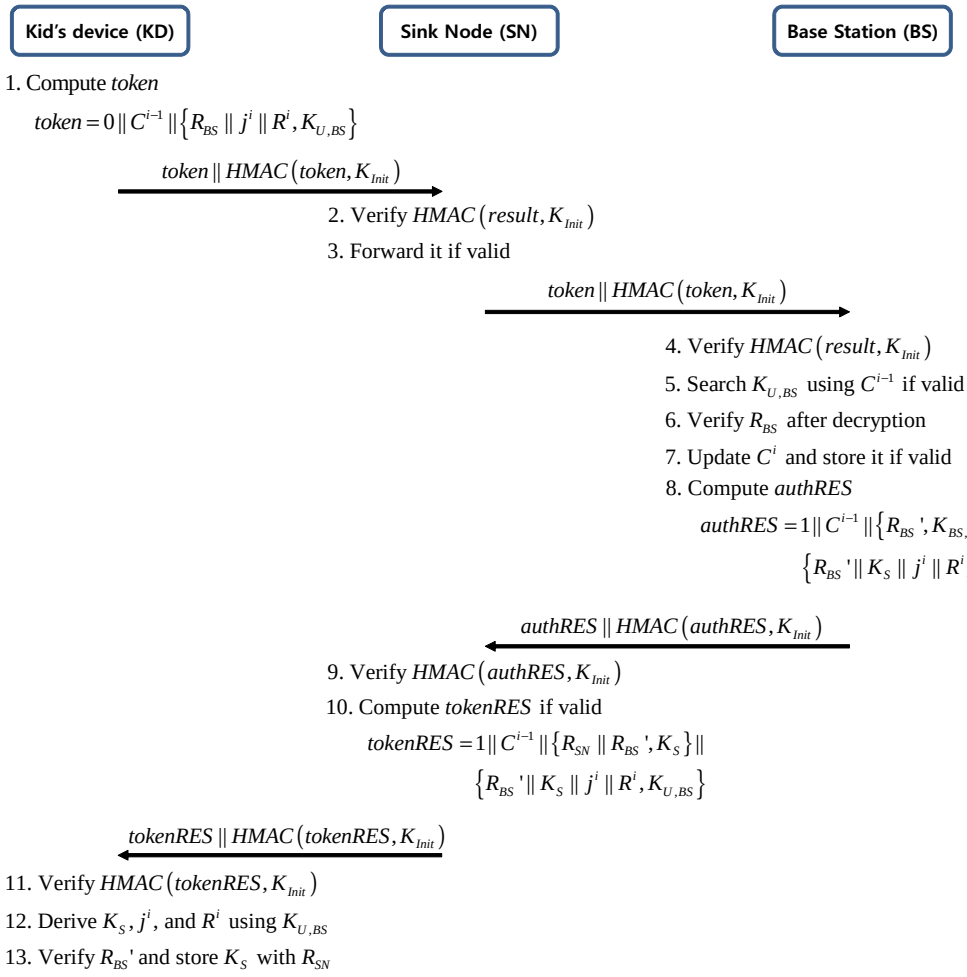

Figure 4.7: Device authentication phase

In device authentication, a device owned by the child sends its authorized credential with necessary information for next authorized credential to the base station via its nearby sink node. The nearby sink node forwards the received message to the base station only if HMAC (keyed-Hash Message Authentication Code) of the received message is valid.

Then, the base station checks integrity of the received authentication token. Only if verification result is correct, the base station searches $K_{U,BS}$ in its database using C^{i-1} where $i = 2, 3, \cdots, n$. When the base station find the shared key $K_{U,BS}$, the base station can decrypt the token and verifies R_{BS}. If verification is correct, the base station authenticates the device, updates $C^i = H(C^0||j^i||R^i)$, and stores C^i in its database. Note that i indicates i-th authentication request. Otherwise, the base station drops the received token. Note that the base station has C^0, R^1, j^1, C^1, R_{BS}, and $K_{U,BS}$ in its database after location query phase. Also, this information is only known to the end-user and base station. As a result, the base station can identify that the device is legal and authorized. After storing updated C^i, the base station computes a response message of the received token, called as *authRES*, and forwards *authRES* to the nearby sink node of the device. Note that K_S is $H(R'_{BS}||K_{CK})$ where K_{CK} is a shared key between the nearby sink node and its members consisting of a cluster for data aggregation.

The nearby sink node verifies integrity of the received message, computes a response message of

the received token, called as *tokenRES*, and sends it to the device.

After receiving *tokenRES* and verifying its integrity, the device derives K_S, j^i, and R^i using $K_{U,BS}$. Only if the received R_{BS}' from the base station and the received R_{BS}' from the nearby sink node are the same, the device stores K_S and R_{SN}. We illustrate this procedure in Figure 4.7.

When the device supports a capability of accessing Wi-Fi or 3GPP network, the device directly sends the token to the base station. Only the difference with the procedure in Figure 4.7 is the message $E\{R'_{BS}, K_S, j^i, R^i, K_{U,BS}\}$ directly sent by the base station to the device. Since the bandwidth of Wi-Fi and 3GPP network is higher than the sensor network, this approach can reduce the processing delay and support better scalability.

Although the base station has the shared key K_S with the child's device, the base station cannot identify the owner of the device. In location query phase, the base station only verifies the legitimacy of the received token. When the token is authorized by the mobile operator, the base station stores K_S and C^0 in its database. Therefore, we believe that our protocol can support privacy of the end-user by hiding the relationship between the device and end-user.

Location determination

Since the device owned by a child has K_{Init} and K_S, the device can distinguish whether the nearby sensor nodes are members of the sensor network belonging to the base station.

Using triangulation based on RSSI [40] from three or more legitimate sensor nodes, the device can determine its location within 3 meters. As our interest is not location determination technique, a detailed method is out of scope in our paper. After identifying the location of the device, the device broadcasts a result message $R_{SN}||C^{i-1}||E\{ZONE||R_{KD}\ ||R_{BS}, K_{HS,KD}\}$ with its HMAC to its nearby sink nodes.

The nearby sink node verifies integrity of the received message and checks whether the device has proper R_{SN} and C^{i-1}. Only if the device has valid R_{SN} and C^{i-1}, the sink node forwards the received message to the base station.

The base station verifies integrity of the received message, searches $K_{U,BS}$ in its database using C^{i-1}, decrypts the message, and verifies R_{BS} using $K_{U,BS}$. If verification result is correct, the base station stores $E\{ZONE||R_{KD}||R_{BS}, K_{HS,KD}\}$ in its database. Otherwise, the base station drops the received message. Figure 4.8 illustrates this procedure. When the device can directly access the base station through Wi-Fi and 3GPP, the result message is directly sent to the base station. As a result, the nearby sink node can handle more messages from the child's device by reducing the processing time of the received message in the nearby sink node. That's why we believe that our protocol can support better scalability.

Query response

In query response phase, an end-user sends a location request $C^{i-1}||E\{REQ||\ C^{i-1}||R_{BS}\ ||R', K_{U,BS}\}$ to the base station via his/her home server, where R' is a fresh nonce and REQ is message type.

The base station stores location information of the child, $K_{U,BS}$, C^i, j^i, R^i, and C^0, in its database. Using the stored C^{i-1}, the base station can find the necessary information (*i.e.*, $K_{U,BS}$ and R_{BS}) and compare the received R_{BS} with the stored R_{BS}. Only if the result is correct and location information is received from the device, the base station issues and sends a response $R'||R'_{BS}||E\{ZONE||R_{KID}, K_{HS,KD}\}$ to the end-user.

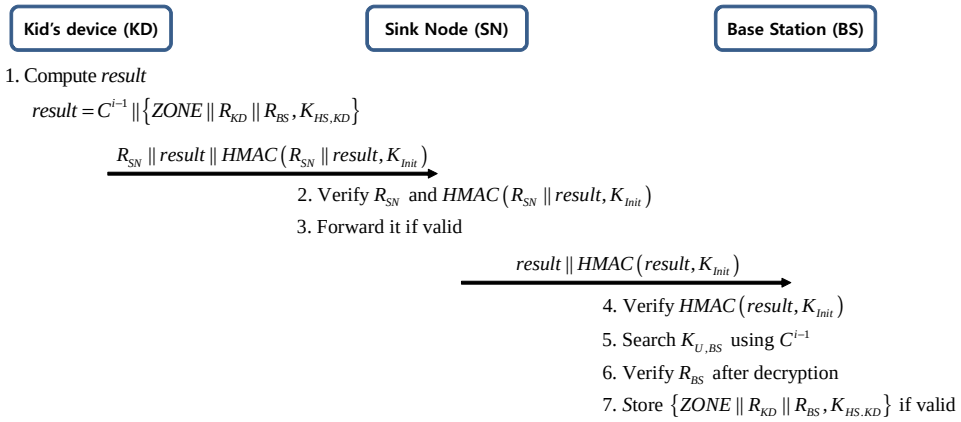

Figure 4.8: Location determination phase

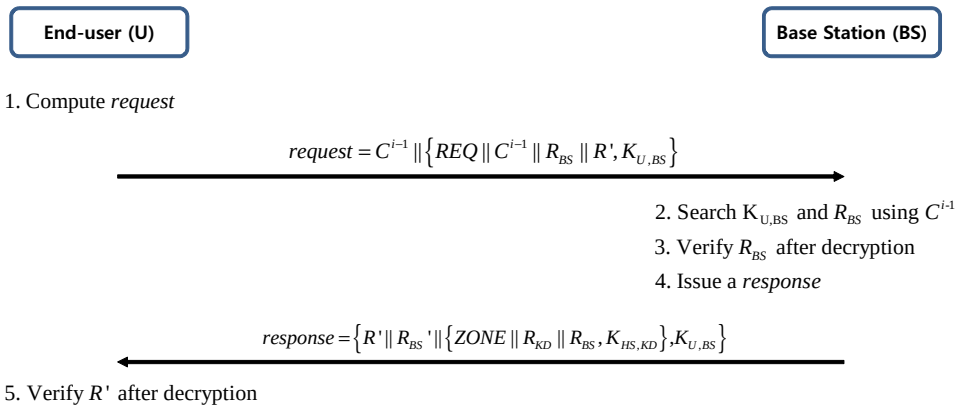

Figure 4.9: Query response phase

The end-user's home server decrypts the received response and verifies R' with $K_{U,BS}$. Only if the verification result is correct, the home server can start to identify the child's location. Otherwise, the home server retries query response phase. Figure 4.9 shows this procedure.

Location information transmission

After receiving query response from the base station, the home server can identify the child's location. If the location is a dangerous area, the home server notifies an alerting message to the end-user's mobile phone. When the end-user wants to observe kid's location periodically, the home server can send the location information to the end-user's mobile phone.

4.1.4 Analysis

Compared to the previous approaches by mobile service providers such as SKT, KT, and LGT, the service provider does not store any private information of the end-user (*e.g.*, emergency contact number, name, and frequent visiting location). While taking a service, the end-user can hide his/her identity from the service provider. From this point, we believe that our protocol can protect the privacy of the end-user compared to the previous approaches by mobile service providers. In addition,

the kid's device only needs to communicate with the nearby sensor node (or access point). In order to authenticate the device owned by the kid, the device only requires one symmetric key operation and two hash operations while the nearby sensor node only requires two hash operations. Based on these observations, we believe that our protocol is lightweight.

Since the existing approaches are based on GPS, the performance comparison between our protocol and the previous approaches is not meaningful. That's why we only show the properties comparison in Table 4.1. The existing services in Korea employs Assisted GPS to determine an end-user's location whether the end-user exists indoor or outdoor. Although the services have 10m accuracy in outdoor environment, accuracy of the services is changed to 2km in indoor environment. Since the signal strength of GPS is too weak for location determination in indoor environment, accuracy of GPS does not influence accuracy of the service in an indoor environment.

In our protocol, a child's device only requires RF communication module (or WiFi module) while the device in the previous work should include A-GPS and any communication module (*e.g.*, 3G network and WiFi). Compared to the device cost in the previous work, the device with WiFi or RF module in our protocol is 2 or 20 times more inexpensive, respectively. As the number of the service subscribers increases, the device cost for the service subscribers will increase linearly. Although our protocol requires the infrastructures such as wireless sensor network or WiFi network, this deployment cost is relatively smaller than the reduced device cost for the service subscribers. From this point, we believe that our approach is useful in u-City, which is a typical example of ubiquitous computing environments.

Table 4.1: Properties comparison

Properties	**Existing services in Korea**	**Takada *et al.* [37]**	**Ours**
Risk analysis	By service provider	By end-user	By end-user
LDT †	GPS	GPS	RSSI based on RF
Accuracy in indoor	$\sim$ 2km	$\sim$ 2km	$\sim$ 3m
Accuracy in outdoor	$\sim$ 10m	$\sim$ 10km	$\sim$ 3m
Supporting area	Outdoor	Outdoor	Indoor & outdoor ‡
Direct communication	Not required	Required	Not required
Infrastructure	Required	Not required	Required
Privacy protection	X	O	O
Controllability of an end-user	X	O	O

† : Location Determination Technique

‡ : An area supporting RF communication

By deploying several access points (*i.e.*, sink node in our protocol), our protocol can provide the child-care & safety service in outdoor environment. Although this approach incurs additional deployment cost, the cost may be less than the all subscribers' expenditures for their GPS modules of in the outdoor environment. Also, in u-City, these access points may be deployed to support seamless

connection for the subscribers.

In performance analysis, we illustrate the efficiency of our protocol. Specially, our protocol supports scalability by reducing computation and communication overhead of the existing infrastructure. During device authentication and location determination phases, a sink node requires only one symmetric key operation and 8 hash operations. However, the deployed sensor nodes do not require any computation and communication cost. Also, in security analysis, we indicate that our protocol can satisfy the security requirements of the child-care & safety service in ubiquitous computing environment.

Performance analysis

Storage cost: A device owned by an end-user's kid is only required to save R_{KD}, R_{BS}, C^0, R^i, j^i, n, S, shared key between the end-user and base station $K_{U,BS}$, shared key between the end-user's home server and device $K_{HS,KD}$, and shared key between its nearby sink node and device K_S. Since C^l, where l is $1, \cdots, n$, can be generated directly from the anchor value, it means that our protocol does not require to store all credential information.

Computational cost: We present computational cost of each phase in Table 4.4(a). If an "off-line" term exists, the computation can be done prior to the session. For instance, an end-user should require online computation (one public key operation, two symmetric key operations, and one hash operation) and off-line computation (two public key operations and one signature generation) in subscriber registration phase. Since a kid's device in our protocol requires only symmetric key operations and hash operations, we believe that our protocol can support various devices with the limited resources in ubiquitous computing environment.

Table 4.2: Computational overhead in each phase

Phase	Entity	Public Key Oper.	Sign Oper.	Symmetric Key Oper.	Hash Oper.
Subscriber	U	2(off-line)+1	1(off-line)	2	1
Registration	MO	1	2	2	1
Location	U	1(off-line)	0	1	1
Query	BS	1	1	1	1
	KD	0	0	3	5
DA†	SN	0	0	1	6
	BS	0	0	3	5
	KD	0	0	1	2
LD‡	SN	0	0	0	2
	BS	0	0	0	2
Location	U	0	0	3	0
Response	BS	0	0	2	2

† : Device Authentication

‡ : Location Determination

Communications cost: In our protocol, for device authentication a device of an end-user's kid requires only two rounds, which are minimum to achieve authenticated key establishment protocol. In

Table 4.3: Communication cost of a kid's device

Phase	Energy	Communication cost (μJ)	Communication cost (bytes)
Device authentication	E_{tx}	0.99840000	72
	E_{rx}	1.08160000	104
Location determination	E_{tx}	1.10933333	80
	E_{rx}	0	0
Device authentication	E_{tx}	1.23433333	89
based on B-MAC [53]	E_{rx}	1.25840000	121
Location determination	E_{tx}	1.3450666	97
based on B-MAC [53]	E_{rx}	0	0

addition, the device requires one round to report the kid's location. From this, we believe that our protocol is lightweight in the number of communication rounds.

To compute lifetime of a kid's device, we assume that AES-128 and SHA-1 are used as symmetric encryption scheme and hash function. Also, we assume that n is 80, z is zero and message transmission rate r is every 30 minutes (*i.e.*, 1/(30*60)). Based on the analytical model in [53], we can compute total energy consumption of a kid's device for regular location report and illustrate the result in Table 4.3. Note that E_{tx} is independent of neighbor size, since the kid's device can directly communicate with a nearby sink node. Also, we only consider the necessary message for our protocol in the above computation. When we adopt B-MAC [53], one of efficient Medium Access Controls for wireless sensor network in the literature, additional 17 bytes (*i.e.*, 8 bytes for preamble, 2 bytes for synchronization, 5 bytes for header, and 2 bytes for CRC) are required and we present total energy consumption of the kid's device for regular location report in Table 4.3.

Since the kid's device consumes $3.18933333\mu J$ without B-MAC (or 3.83760000 μJ with B-MAC), our protocol satisfies lightweightness.

Security analysis

Mutual authentication: In our protocol, an end-user including the device owned by the end-user's child authenticates him/her to the base station (or mobile operator) using the authorized credential, so that the base station (or mobile operator) knows that the user is legal and authorized. The base station (or mobile operator) authenticates himself/herself to the user through its own public key and by showing his/her knowledge of the corresponding private key.

Privacy protection: Since any private information (*i.e.*, safety zone, mobile phone number, and location information) are stored in personal home server, the mobile operator only issues an end-user's authorized credential for anonymous authentication, and the base station verifies the authorized credential with PK_{MO}, an administrator in mobile operator cannot obtain any private information during location query, location determination, and query response.

Moreover, an administrator in base station cannot distinguish who requests the child-care and safety service due to anonymous authentication. Note that the base station can guess the child's nearby location since the information is delivered to the base station via nearby sink node. However, the base station cannot find any relationship between the location information and end-user. Hence,

our service can preserve an end-user's privacy.

Also, privacy of the end-user's child can be protected as the child can control over his/her location information transmission.

Confidentiality and integrity: All communications are encrypted with a receiver's public key or symmetric key, which is shared between a sender and receiver. Thus, confidentiality is provided in our service. Also, the sender and receiver can derive a secret key for HMAC using the shared key. In this point, integrity can be easily provided in our service.

Scalability: Since our protocol can support anonymous authentication based on the authorized credential, our protocol may be vulnerable to battery exhaustion attack, sending continuous or periodic authentication requests (or location reporting). As the base station can only verify an end-user's authentication request, the intermediate nodes between an adversary and base station should forward the received message to their parent node. That's why battery exhaustion attack is possible. However, we reduce the effective bounds of battery exhaustion attack to one of the adversary's nearby clusters based on the following observations. ***First***, an outsider cannot generate a valid authentication request since the outsider does not know K_{Init}. ***Second***, a sink node, having a permanent power for easiness of network management, can directly communicate with the base station to minimize a hop distance between the adversary and base station although the adversary has K_{Init} by compromising one of sensor nodes. ***Third***, we assume that a kid's device communicates with its nearby sink node via the different radio frequency with one used in communication between a sensor node and sink node. Therefore, we believe that our protocol can provide scalability.

Lightweightness: In our protocol, a child's device only needs four symmetric key operations, three HMAC operations, and one hash operation for device authentication and location determination. Also, the device does not require communicating with the child's home server. Hence, our protocol is believed to be lightweight than the previous approach [37].

4.2 Privacy-preserving secure service discovery protocol

4.2.1 System architecture

Before describing our protocol in detail, we illustrates our system architecture and activities of each entity in Figure 4.10.

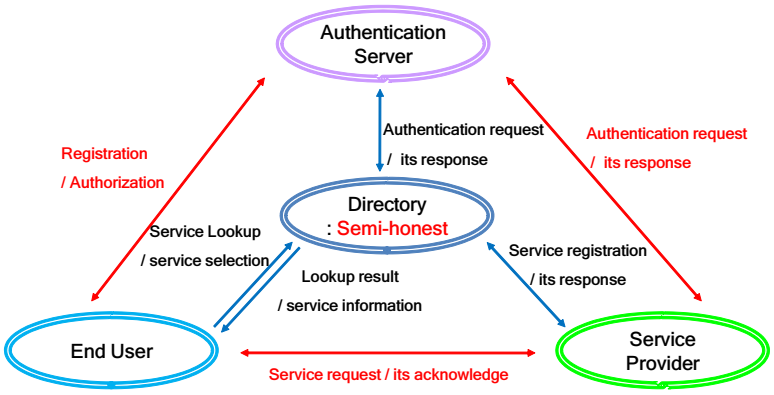

Figure 4.10: System architecture for ubiquitous computing environment

Compared to the previous model shown in the Chapter 2, the directory server is introduced to

store or provide proper service access information for simplified trust management and scalability. The DS can authenticate itself and shares a fresh session key with the AS though entity registration phase and entity authentication phase. Using the shared session key, the DS can make the communication with the AS to be secure. From now, we assume that the communication between the DS and AS is secure.

4.2.2 Brief sketch

This privacy-preserving secure service discovery protocol consists of entity registration, entity authentication, service registration, service discovery, and service access.

Entity registration

In the entity registration, the entity (*i.e.*, end-user and service provider) generates their authentication tokens and requests the authentication server to authorize the received token. Only if the entity has proper access permission on the requested service or providing a service, the authentication server authorizes the received token. Through the entity registration phase in 3.3, we can preserve the privacy of the end-user while verifying access permission on the requested service.

Entity authentication

Through the entity authentication phase, the entity (*i.e.*, end-user and service provider) authenticates himself/herself to the directory server or service provider in order to access the service. In order to preserve the privacy of the end-user and service provider, the anonymous authentication is required. The entity authentication phase 3.3 can provide the privacy of the entity while requiring only one public key operation in the first authentication request. Except the first authentication request, the entity only requires symmetric key operation in order to reduce the processing delay time. After the entity authentication phase, the entity can share a secret key with the directory server in the service discovery phase and the service provider in the service access phase.

Service registration

In order to allow the directory server to forward the service access information to the legitimate end-user, the directory server should obtain the service access information (*e.g.*, service type, service name, service description, SID list, and network address) from the service provider. That's why the service registration phase is required. However, the service provider may want to preserve his/her privacy during the service registration. We should provide anonymous authentication through 3.3.

In addition, the service provider submits the necessary information to allow the directory server to verify the access permission of the end-user, one of his/her legitimate subscribers. This problem can be easily addressed by the standard version of the membership verification 3.2.1 in Chapter 3. Even if the directory server has SK_{BGN}, the directory server cannot obtain the subscription information based on the security analysis in Chapter 4.

Service discovery

During the service discovery phase, the service provider may want to hide his/her identity from the other entities having no access permission. By allowing the legitimate end-user having proper ac-

cess permission to find the target service and hiding the identity of the service provider from the directory server and adversary, we can address the above issue. Through the entity authentication, the service provider can hide his/her identity from the directory server including the adversary. In order to allow the legitimate end-user having proper access permission to find the target service, the directory server should check the access permission of the end-user. Then, the directory server may obtain the subscription information. That's why the privacy-preserving membership verification 3.2.1 is required to find the alternative by specifying encrypted keywords.

Service access

Through the service discovery phase, the end-user can find an alternative service and access the service. In order to provide anonymous authentication during accessing the service, the entity authentication phase 3.3 is useful.

4.2.3 Implementation

The proposed protocol consists of four phases: entity registration, service registration, discovery, and service access. Through entity registration phase, an entity (*e.g.*, end-user or SP) obtains his/her authorized credential preserving the privacy of the entity. The AS verifies the legitimacy of the entity using the received MT. In service registration phase, a service provider stores his/her service and proper information, which are required to verify the legitimacy of the end-user, to the DS. Using his/her credential and DMV, the end-user can find the registered service and obtain proper information to access the service in discovery phase.

Entity registration

Only if the entity is a legal entity having proper permission to access a service or provide his/her service, the AS authorizes the received credential. To hide the relationship between the authorized credential and the entity's real identity we apply blind signature technique. Figure 4.11 depicts entity registration phase.

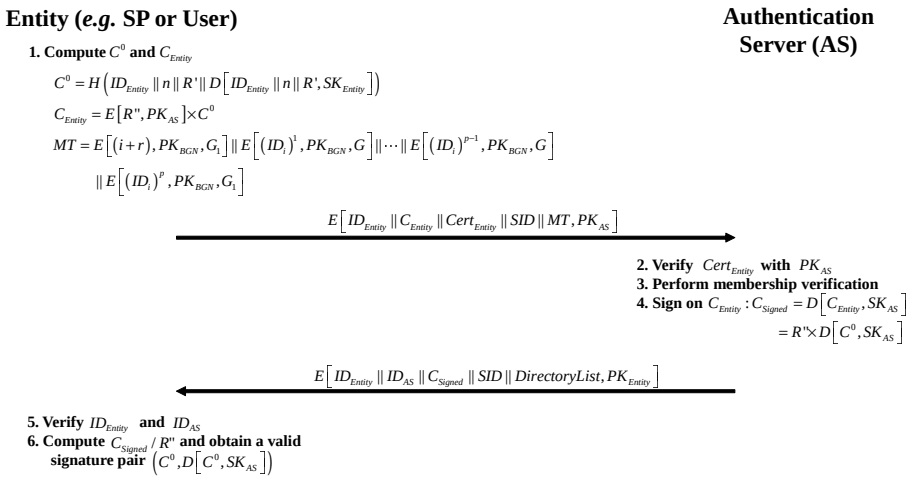

Figure 4.11: Entity registration

To verify whether the entity is a legitimate one and has proper permission to access or provide a service, the AS verifies the received certificate with PK_{AS} and performs our membership verification. If the result of the entity's membership verification is non-zero integer less than p, the AS sends proper information to the entity. Otherwise, the AS discards the received message. Note that $DirectoryList$ indicates a list of the accessible and legitimate DS.

Entity authentication

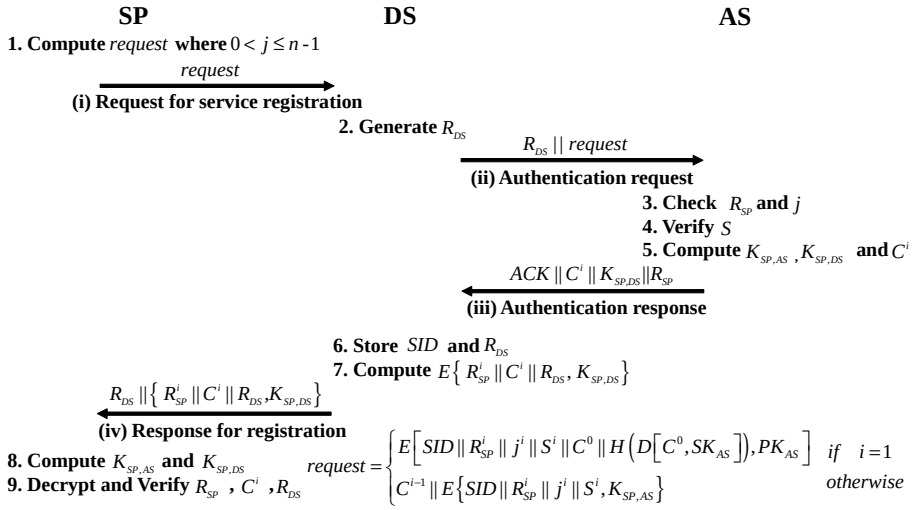

Figure 4.12: Entity authentication in service registration

In the entity authentication phase, each entity establishes $K_{Entity,AS}$ and $K_{Entity,DS} = H(K_{Entity,AS} ||C^i||R_{DS})$.

$$K_{Entity,AS} = \begin{cases} H(C^0||PK_{AS}||R^1||j^1||SID) & \text{if } i = 1 \\ H(C^0||C^{i-1}||SID) & \text{otherwise} \end{cases}$$

To provide accountability of the authorized credential, we adopt a set of selected numbers S, which is l-bit array. In the first access request, each entity generates the set randomly. Whenever sending an i^{th} authentication request, each entity generates a fresh nonce R^i_{Entity} and selects one random number j between 0 to $l-1$ until $j-th$ value of S is 0. Since the set is only known to the entity and AS, the adversary without knowing S cannot generate the authentication request. Therefore, we believe that our protocol can enhance security level. Note that $C^i = H(C^0||j^i||R^i)$. For entity authentication, the AS performs the following verification procedure:

1. 1^{st} request: After decrypting the request message, the AS computes $H(D[C^0 , SK_{AS}])$ and compares the result with the received $H(D[C^0, SK_{AS}])$. Only if the result is same, the AS believes that the entity has an authorized credential and computes $C1 = H(C^0||j^1||R^1)$ and stores SID, S^1, C^0, and C^1 in the database. Otherwise, the AS discards the request.

2. i^{th} request: The AS finds C^0, $S^(i-1)$ and SID in the database using the received $C^(i-1)$ and decrypts the received message with $K_{Entity,AS}$. Next, the AS verifies that the entity has the same set of selected numbers and j^i is not in the set. Only if the result is correct, the AS stores the received C^i and S^i. Otherwise, the AS discards it. If the entity is a legal one with

proper access permission, C^i and S^i are stored in the database. As a result, the AS can verify whether the entity has an authorized credential using the received $C^(i-1)$.

After this verification, the AS sends a response message to the DS. Then, the DS stores proper information (*i.e.*, SID and R_{DS}) and gives a response for entity authentication request to the entity. After verifying the response message, the entity computes $K_{Entity,AS}$ and $K_{Entity,DS}$. Figure 4.12 illustrates this phase when the entity is a SP.

Service registration

The service registration phase consists of entity authentication and registration. Through the entity authentication phase, an SP can anonymously authenticate himself/herself to a DS and establish the shared keys, $K_{SP,AS}$ and $K_{SP,DS}$. Using $K_{SP,DS}$, the SP registers an encrypted service access information (*e.g.*, service type, service name, service description, SID list, and network address) by $K_{rk} = H(g^r)$, encrypted coefficients of $f(x)$ with PK_{BGN}, and a list of hidden encryption keys $\{g^{r_m/ak_1}, \cdots, g^{r_m/ak_p}\}$ with the directory, where r, g, and p are a random number, a generator of cyclic group G with order $n = q_1 q_2$, and the number of access keys, respectively.

Also, the SP may expose polynomial identifiers, SID list, and service type to the DS when the SP wants to serve all or partial end-users enrolled in the AS. Because the exposed access information allows an end-user without any prior knowledge about nearby environment to obtain an accessible service list, this approach can support an end-user's mobility.

Service discovery

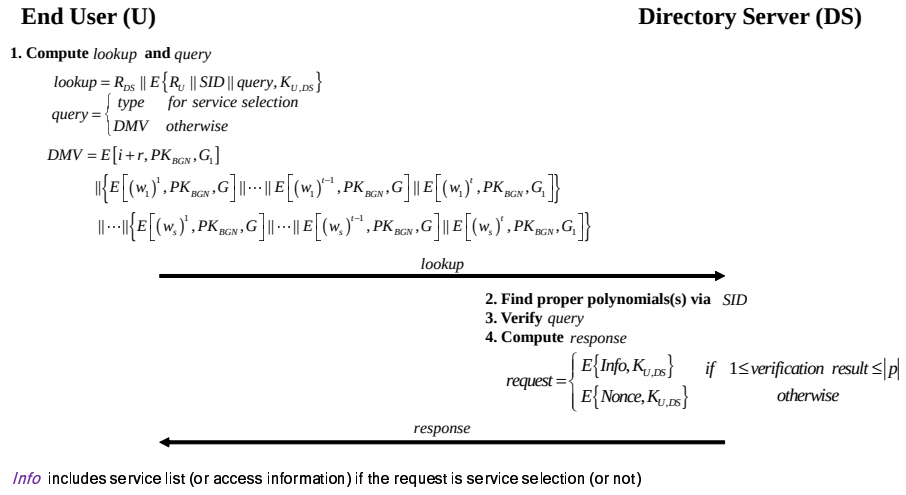

Figure 4.13: Service lookup and its response in discovery phase

The discovery phase consists of three sub-steps, entity authentication, service lookup, and service selection. As the entity authentication phase has already been explained, we will skip the explanation.

Service lookup: Although our keyword search is an efficient method compared to the previous approach [20], we should reduce a searching space to address scalability issue by reducing the processing time of an end-user's service lookup request in a DS or entity authentication request in the AS. Also, the end-user without having any prior knowledge may know that the same or similar type of

the alternative services. That's why we use the service type as searching condition in service lookup request.

When the lookup is used to find alternative services, query is type indicating a type of the alternative services. If lookup is used to obtain service access information, query is DMV. Then, the DS finds the shared key using R_{DS}, decrypts the lookup message, and checks whether the stored SID is the same as the received SID. If the comparison is correct, the directory server performs the following steps:

1. To find alternative services: Using the type, the DS can search alternative services as some service providers expose partial access information of their services in the service registration. If any matched services exist, the DS encrypts the stored service list using $K_{U,DS}$ and sends the resulting ciphertext to the user.

2. To obtain the service access information: The DS performs membership verification procedure by evaluating the given DMV. When the verification result is not zero, the DS sends the stored access information, Em, K_{rk}, and the matched hidden key g^{r-ak_i}.

Figure 4.13 depicts this phase in detail.

Service selection: After the service lookup phase, the end-user may obtain a service list having the submitted service type. Then, the end-user selects one service from the list and notifies the selection to the DS. If a proper access control is enforced against the selected service, he/she should submit the proper DMV to the DS. As the detailed procedure is the same as the service lookup phase for obtaining service access information, we do not explain the procedure.

Service access

To preserve an end-user's privacy during the service access phase, an anonymous authentication protocol is required. Our protocol can support this without additional computational cost since the end-user already has the authorized credentials after the entity registration phase. The detailed procedure is similar to the entity authentication phase. The difference is that the entities participating in the authentication process are the end-user, SP, and AS.

4.2.4 Analysis

Performance analysis

Storage overhead: Each end-user should store $E[(i+r), PK_{BGN}, \mathbb{G}_1]$, access key, PK_{BGN}, and one 5-tuples (C^0, R^i, j^i, N, S) for service discovery and service access request. The service provider needs to save a set of polynomials $f_1(x), f_2(x), \cdots, f_k(x)$ presenting a subset of own subscriber and one 5-tuple (C^0, R^i, j^i, N, S) for service registration.

Computational overhead: Table 4.4(a) shows the computational overhead of the proposed protocol. Note that p is the number of access keys in S_1, t is the number of keywords in S_2, s is the number of keywords specifying the target service and $1/N$ indicates that one operation is needed during N sessions. During discovery phase, the user should compute (one secret key encryption, two secret key decryptions, and four hash operations) per each discovery request while computing ($1/N$ public key encryption, one secret key encryption, $s \times t$ BGN encryptions, and $s \times (t-1)$ modular exponentiations) before the session. Compared with the previous protocol in [3], where the end-user should

Table 4.4: Performance and security analysis

(a) Computational overhead in each phase

	Entity registration		**Service registration**			**Discovery**		
	U or SP	AS	SP	DS	AS	U	DS	AS
Public key Oper.	$(1)^{\dagger}$ + 1	1	$(1/N)^{\dagger}$	0	1/N	$(1/N)^{\dagger}$	0	1/N
Signature Oper.	1	1	0	0	1/N	0	0	1/N
Hash Oper.	1	0	4	0	4	4	0	4
Secret Key Oper.	0	0	$(1)^{\dagger}$ + 2	1	1	$(1)^{\dagger}$ + 3	1	1
BGN Enc. [44]	$(p)^{\dagger}$	0	$(t)^{\dagger}$	0	0	$(s \cdot t)^{\dagger}$	0	0
Pairing Oper.	0	p+1	0	0	0	0	s·(t-1)	0
Exponent Oper.	0	p+2	$(t)^{\dagger}$	0	0	s·(t-1)	s·(t-1)+5	0

† : Precomputation Oper. : Operation Enc. : Encryption

(b) Security-related features comparison

	Proposed protocol	Zhu *et al.* [3]	Czerwinski *et al.* [2]
Mutual authentication	Yes	Yes	Yes
Confidentiality & Integrity	Yes	Yes	Yes
Anonymity & non-linkability	Yes	Yes to outsiders	No
Accountability	Yes	No	No
Directory server	Not trusted	Trusted	Trusted
Access control	Yes	Easy to obtain	Yes
Scalability	Good	Not too bad	Good
Enhanced security of level	Yes	No	No
The abuse of subscription information	None	Yes	Yes

compute two public key encryptions and one signature generation, the proposed protocol needs less computational cost.

Communication overhead: This protocol needs four rounds during service discovery. Previous protocol in [3] also requires four rounds. To discuss with the size of communication message, let assume that SHA-1, AES-128, and ECC-160 are used as hash function, symmetric encryption scheme, and asymmetric encryption scheme, respectively. Also, service identifier is 8 bit, N is 80, the given polynomial identifier is 24 bit, and degree of polynomial is 4. Although the previous protocol [3] requires 1104 bits, which is less than our protocol (*i.e.*, 1920 bits in i^{th} request or 2016 bits in 1^{st} request), the previous protocol does not consider that communication cost during agreeing with the same hash functions to be used service match in advance. Also, certificate exchange between an end-user and directory server is not considered. To this point, the proposed protocol is reasonable in communication overhead.

Security analysis

The proposed protocol provides the following security-related features. In Table 4.4(b), we compare the security-related features of our protocol with previous work.

Mutual authentication: The end-user authenticates the AS through a public key of the AS and knowledge of the corresponding private key. Also, the AS authenticates the end-user using an authorized credential of the end-user.

Anonymity: Our protocol protects privacy of an end-user against insiders and outsiders. As the each user authenticates herself to insiders (*i.e.*, SP and DS) using authorized credentials, the insiders cannot predict who sends the service access request or lookup request. Here, insiders other than the user cannot find any relationship among the authorized credentials, in that the credentials are derived from initial credential C^0 using one-way hash function. Outsiders also cannot identify who sends the messages since all communication messages are encrypted using a shared session key.

Non-linkability: Non-linkability means that, for insiders (*i.e.*, SP and DS) and outsiders, 1) neither of them can ascribe any session to a particular end-user, and 2) neither of them can link two different sessions to the same user [34]. More precisely, non-linkability needs to prevent insiders and outsiders from obtaining an end-user's private information. Our protocol can achieve non-linkability with respect to both insiders and outsiders. First, the information to distinguish each user is never transmitted in a plaintext form. As a result, outsiders cannot associate a session with a particular user and ascribe two sessions to the same user. Second, outsiders and insiders cannot find any relationship between the exposed credentials due to the one-way hash function. Third, as the given DMV is non-deterministic, the DS cannot link two different sessions to the same user. Finally, all communications are protected by a fresh session key.

Accountability: The credentials are authorized only when the end-user is explicitly authenticated and has proper access permission on the service. By adopting a set of selected numbers, the proposed protocol can provide a one-time usage of the authorized credentials to prevent an attacker from reusing the authorized credentials. Also, the proposed protocol can provide good accounting capability by incorporating an accounting function.

Data confidentiality and integrity: All communications are protected by a shared session key or the receiver's public key. In this point, our protocol supports data confidentiality. Although we do not explain explicitly how to generate a key for integrity check, end-user, SP, DS, and AS can derive the key using the shared information such as a fresh session key (or the receiver's public key) and exchanged nonce. By applying HMAC with the derived key, the proposed protocol can support data integrity.

Enhanced level of security: Every access request message contains S, randomly generated by the end-user and delivered only to the AS, to prove the actual holder of the message. Thus, an adversary is required to present S even if the adversary knows the target user's initial credential. In this point, the proposed scheme enhances the level of security.

No additional key management: Two of the following entities the end-user, SP, DS, and AS can generate $K_{U,AS}$, $K_{U,DS}$, $K_{U,SP}$, $K_{SP,AS}$ and $K_{SP,DS}$ during the entity authentication phase. Also, the shared key can be derived from the stored 5-tuple (C^0, R^i, j^i, N, S). Thus, there is no additional key management overhead when replacing the reduced public key operations with symmetric key operations. Moreover, there is no revocation cost of the shared key since the key is used during only one session.

Less the abuse of subscription information: In our protocol, the administrator can only iden-

tify the subscribers of the target service provider when he/she monitors all registration requests. However, a proper operation policy for AS can prevent illegal tracking of all registration requests. In this way, the proposed approach reduces the privacy concern of each service provider regarding the abuse of subscription information.

Chapter 5. Evaluation

In this chapter, we evaluate the proposed authentication framework. As three privacy-preserving authentication protocols in the proposed authentication framework are used to support three kind of authentications in the public ubiquitous computing environments, we analyze the performance and security of each protocol. In order to prove that each protocol can preserve the privacy of the end-user, we show that each protocol is based on the computationally hard problem so that the adversary cannot violate the privacy of the end-user and service provider unless the adversary solves the problem in polynomial time.

Privacy protection

Privacy-preserving membership verification: As the directory server and authentication server are operated as semi-honest entity, they cannot obtain the subscription information, encrypted by BGN encryption [44] and stored to enforce proper access control. Only if they decrypt the encrypted information in polynomial time, they can obtain the information. However, they require 2^{80} time complexity to decrypt the subscription information.

Anonymous authentication protocol: Since the adversary cannot obtain any useful information as the information is encrypted by the shared key with the authentication server. Even if the adversary obtain the authorized credential, he/she cannot obtain the necessary information due to the one-way hash function. In other words, the adversary can violate the privacy of the end-user only if the adversary can break the security of the used cryptographic primitives such as ECC, RSA, BGN encryption, and AES. Also, the adversary should guess the set of the selected numbers in order to impersonate the target end-user. As the set is randomly generated, the adversary can guess the correct set with the probability $1/2^{80}$.

Anonymous re-authentication protocol: The shared information with the service provider is encrypted by BGN encryption [44] so that the adversary requires 2^{80} time complexity to decrypt the information. Therefore the adversary cannot obtain the information in polynomial time.

Since less processing delay time and communication overhead allows the directory server and authentication server to support better scalability, we demonstrate that each protocol requires less computational overhead and communication cost than the previous approaches.

Scalability

Privacy-preserving membership verification: Although the number of the service subscriber increases, the proposed membership verification allows the directory server or authentication server to verify the access permission of the end-user within a constant time. Based on the desire performance, the service provider can adjust the processing delay time from 55 millisecond to a few hundred millisecond. Here, the term "constant time" means that the processing delay time does not be affected by the number of the service subscriber. As the communication overhead is determined by the degree of the given polynomial $f(x)$, the overhead can be adjusted by the service provider.

Anonymous authentication protocol: Since the authentication server only requires one public key operation per each user during n times, we can reduce the processing delay time to authenticate the end-user. During verification process checking access permission of the end-user, the authentication server only requires a constant time even if the number of the service subscribers increases. Compared

to the previous work, the proposed protocol reduce the communication overhead while support 2^{80} computational complexity. By throwing the additional computational complexity, we can lower the communication overhead by reducing 160 bit per each authentication request.

Anonymous re-authentication: The proposed protocol reduces the number of the communication between the gateway and base station. Also, the protocol reduces the number of the communication between the gateway and the nearby sensor nodes. Through these communications reduced by the proposed protocol, the processing delay time to authenticate the end-user is lowered. Although the proposed protocol requires 1 modular addition and 1 modular exponentiation compared to the previous work, the verification time on the sensor node only is less than 9.97 ms.

5.1 Privacy-preserving membership verification

5.1.1 Performance analysis

Standard version

To obtain the processing time of membership verification, we generate a polynomial $f(x)$ with different degree i and implement our polynomial evaluation method using pairing based cryptography library [55] under Intel®Core ™2 2.13GHz CPU, 1GB RAM and Microsoft Windows XP Professional Service Pack 3.

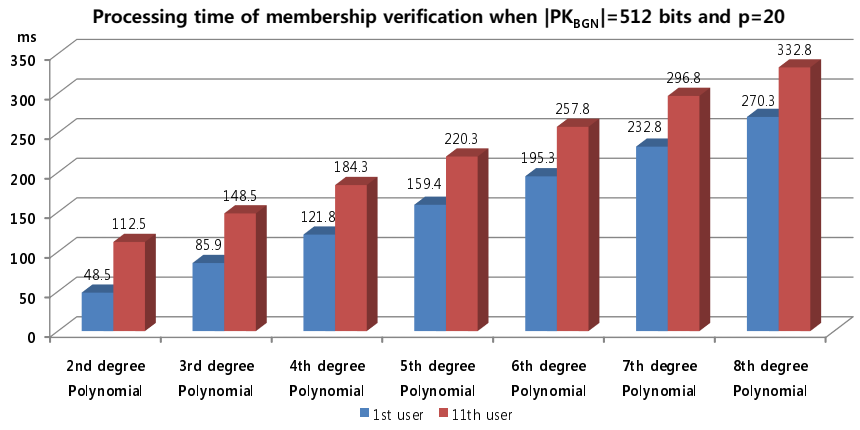

Figure 5.1: Processing time of membership verification when $|PK_{BGN}| = 512$ bits and $p = 20$

Figure 5.1 shows the processing time of the membership verification when $|PK_{BGN}| = 512$ bits and the number of subscribers is 20. Rather than showing membership verification request of all subscribers of the target service, we present the request of $1st$ (or $11th$) end-user. As increasing the degree of a given polynomial $f(x)$, the number of pairings in step (2) and exponent multiplications in step (4) will be increased. That's why the processing time of membership verification is increased linearly. In addition, the communication cost increases linearly with the degree of a given polynomial $f(x)$. Therefore, we suggest that each SP should divide his own subscribers to several subsets based on access privilege and desired performance of our membership verification.

Assume that Alice provides a printing service based on two different privileges such as heavy and light user. In each privilege, 30 legitimate users are subscribing the printing service. If 3 degree of poly-

nomial $f(x)$ is sufficient to satisfy the performance level desired by Alice, for each privilege Alice generates 10 polynomials where the degree of polynomial $f(x)$ is 3 and each polynomial has 10 different subscribers. For membership verification, each end-user should submit $E[(i+r), PK_{BGN}, \mathbb{G}_1]||E[(w_1), PK_{BGN}, \mathbb{G}]$ $||E[(w_1)^2, PK_{BGN}, \mathbb{G}_1]$ where i is the index value of an end-user among the target service subscribers and r is a random number. Then, the message size of submitted membership information is $3 \times |PK_{BGN}|$. However, we can reduce the message size by sending the x-coordinate of membership information to $1.5 \times |PK_{BGN}|$. Since PK_{BGN} is a point on cyclic group G, the encrypted result is also one of points on cyclic group $\mathbb{G}$ or $\mathbb{G}_1$ where a point consists of x-coordinate and y-coordinate. Therefore, we can reduce the communication cost and processing delay to the desired performance of the service provider. Figure 5.2 depicts that the change of storage cost by dividing the given subscribes to several subsets is small. Moreover, this approach will help us to support differentiated access control.

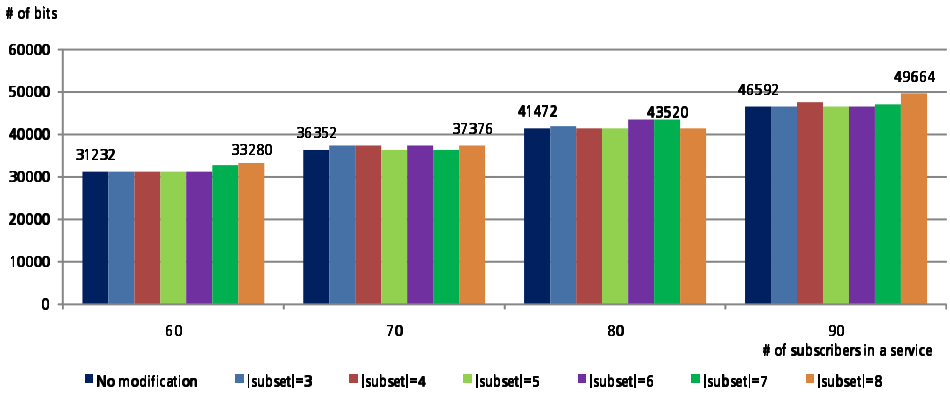

Figure 5.2: Comparison result of storage cost by dividing the given subscribers to several subsets

We believe that our membership verification supports better scalability by reducing the processing time to verify the service subscriber in the directory server as shown in Figure 5.3.

Lightweight version

Compared to the standard version of our membership verification, the lightweight version only requires one exponent addition and one exponent multiplication. By reducing pairing computation in the standard version, we can allow the lightweight version of our membership verification to be used for the resource limited devices in ubiquitous computing environment. As processing time of one modular exponentiation on Elliptic Curve $F_{2^{233}}$ provided by T. Takagi *et al.* is 4.981 ms on MicaZ [56], our membership verification only takes less than 9.97 ms. When we employ assembly implementation, the processing time can be reduce to 3.18 ms. Even if we require additional storage cost for implementing Elliptic Curve Cryptography, we can save much more energy dissipation due to send the received authentication request. From this point, we believe that our membership verification can satisfy the lightweightness requirement.

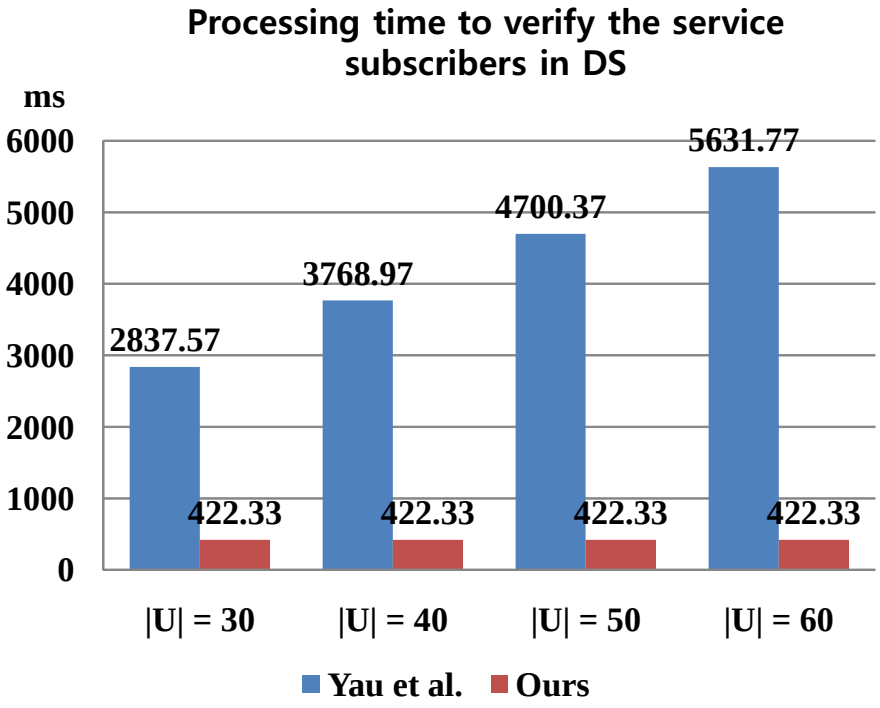

Figure 5.3: Processing time comparison between ours and Yau *et al.*'s approach [20]

5.1.2 Security analysis

Standard version

Based on the following observations, we can hide subscription information of a target service from a computationally-bounded adversary having a ciphertext, PK_{BGN}, and SK_{BGN}. First, the expected decryption time of the given ciphertext, encrypted by PK_{BGN}, using Pollard's lambda method is $\tilde{O}(\sqrt{|T|})$ when the adversary has the private key $SK_{BGN} = q_1$ [44]. As a result, the adversary cannot obtain any useful information in polynomial time.

Second, when the size of public key is larger than 512 bits, our membership verification is resilient against the MOV or Frey-Ruck attack [57, 58], which is a method to break the discrete log problem in $\mathbb{F}_q$ after breaking the discrete log problem in $\mathbb{F}_{q^k}$ using finite field discrete logarithm algorithms such as index calculus. Since the size of public key is larger than 512 bits and k in BGN encryption is 2, the ciphertext in $\mathbb{F}_{q^k}$ is large enough, index calculus method is infeasible in $\mathbb{F}_{q^k}$.

Hence, here comes our third observation. Even if the adversary can decrypt the given ciphertext in polynomial time, the adversary obtains keywords specifying the target service and index of the target user. However, the information is only useful to identify presence information of the target service and the service type since an end-user, who passed membership verification, can obtain the encrypted information regarding service access only. Moreover, the service access information is encrypted with a random key $K_{rk} = H(g^{r_m})$ that can be derived from the list of hidden encryption keys $\{g^{r_m/ak_1}, \cdots, g^{r_m/ak_p}\}$ using each subscriber's access key ak_i where i is an index of each end-user. As a result, the adversary without a random key, K_{rk}, can only obtain presence information of the target service. Whether the service provider is public or private, leakage of presence information is not a problem. Since presence information of the target service such as cyber lecture in ubiquitous institutions is not

important, the decryption of given ciphertexts is not valuable. In the case of private service provider implying that presence information is important, he/she can hide a relationship between his/her service and the given keywords by concatenating a nonce like the idea in Figure 3.6. Note that the private service provider should distribute the nonce, R^i, when each end-user becomes a subscriber of the target service.

Lightweight version

Based on the following observations, we can hide subscription information of a target service from a computationally-bounded adversary having a ciphertext, PK_{BGN}, and SK_{BGN}. First, the expected decryption time of the given ciphertext, encrypted by PK_{BGN}, using Pollard's lambda method is $\tilde{O}(\sqrt{|T|})$ when the adversary has the private key $SK_{BGN} = q_1$ [44]. Since we assume that the message size $|T|$ is 160, the expected decryption time is 2^{80}. As a result, the adversary cannot obtain any useful information in polynomial time.

Second, when the size of public key is larger than 512 bits, our membership verification is resilient against the MOV or Frey-Ruck attack [57, 58], which is a method to break the discrete log problem in $\mathbb{F}_q$ after breaking the discrete log problem in $\mathbb{F}_{q^k}$ using finite field discrete logarithm algorithms such as index calculus. Since the size of public key is larger than 512 bits and k in BGN encryption is 2, the ciphertext in $\mathbb{F}_{q^k}$ is large enough, index calculus method is infeasible in $\mathbb{F}_{q^k}$.

5.2 Anonymous authentication protocol

5.2.1 Performance analysis

Storage overhead: A mobile user is only required to save C^0, R^i, j^i, n, S and $E[i+r, BGN_{pk}]$. Since all credential information except the anchor value can be generated directly from the anchor value, our proposed scheme does not require to store all credential information. Although a mobile user in [15, 16] should store C^0 to avoid repeated hash operation, all credential information should be stored. In this point, the proposed protocol requires less storage capability. Additionally, the proposed protocol is more flexible in the view of access frequency since the information which should be stored is fixed even if the user's access frequency is increased.

Computational overhead: Except first access request, encrypted with a public key of the authentication server, all messages between each mobile user and the authentication server are encrypted using a shared symmetric key. Therefore, the proposed protocol is computationally efficient in entity authentication since symmetric key operation is lightweight than public key operation. However, the proposed protocol requires more computation overhead (*i.e.*, pairing computations and exponent multiplications) in entity registration phase, ridding of privacy concern regarding the abuse of subscription information in a service provider side. If a user performs this job in his/her desktop, the user can use the authorized credentials in resource constrained devices to authenticate himself/herself to a service provider. We compare computation overhead of the proposed protocol with the scheme in [15, 16] in Table 5.1 and 5.2. Note that in Table 5.2 and Table 5.1 if we do not append the term "off-line", then the communication entity such as mobile user, authentication server and service provider, needs on-line computation. Also, "p" is the degree of polynomial $f(x)$, which is used to verify the user's service subscription.

	Ren *et al.*[15, 16]		**Proposed protocol**	
	User	**AS**	**User**	**AS**
Public key Operation	2 (off-line)	1	2 (off-line)	2
Signature Gen.	1 (off-line)	1	1 (off-line)	1
Hash Operation	n+1 (off-line)	0	1	0
Nonce Gen.	2	0	2	0
BGN Encryption [44]	0	0	p+1 (off-line)	0
Pairing Operation	0	0	0	p+1
Exponent Multiplication	0	0	0	p+2

Table 5.1: Computation overheads comparison of entity registration

	Ren *et al.*[15, 16]			**Proposed protocol**		
	User	**SP**	**AS**	**User**	**SP**	**AS**
Public key Operation	1 (off-line)	0	1	1/n (off-line)	0	1/n
Signature Gen.	0	0	1/n	0	0	1/n
Symmetric Key Operation	3	3	0	1(off-line) + 1	1	1
Hash Operation	2	2	0	3	0	3
Nonce Gen.	1	1	0	2	1	0

Table 5.2: Computation overheads comparison of entity authentication

Communication overhead: The proposed protocol only requires two rounds to achieve the authenticated key establishment. Note that two rounds in authenticated key establishment protocol are minimum rounds. To compare the message size in the proposed protocol with the scheme in [15, 16], we assume that SHA-1 and AES−128 are used as hash function and symmetric encryption scheme in the both scheme. Also, we assume that service identifier is 8 bits, n is 80 and polynomial identifier is 24 bits. Then, SID in [15, 16] is 8 bits while SID in the proposed protocol is 32 bits. Here, 8 bits service identifier is sufficient when we define the service type as hierarchical structure. Finally, we assume that the user in the scheme [15, 16] sends $H(D[C^n, SK_{AS}])$ to the authentication server to reduce message size of the first service access request although the user sends $D[C^n, SK_{AS}]$ to the authentication server. Then, the difference size of all messages during 80 sessions is:

$$Difference = \begin{cases} 79 \times (536 - x) - 8 & \text{if RSA is used} \\ 79 \times (152 - x) + 248 & \text{if ECC is used,} \end{cases}$$

where x is $|PK_{AS}|$.

When we adopt RSA-1024 (or ECC-160) as asymmetric encryption scheme, the difference is $-38,560$ (or -384) bits. Note that ECC is abbreviation of Elliptic Curve Cryptography. Also, if the message size is greater than the size of PK_{AS}, we apply hybrid encryption. This result indicates that the proposed protocol can reduce 482 (or 5) bits per each session while providing additional 2^{80} computational complexity. Moreover, the proposed protocol can adjust this communication cost by increasing or decreasing the user's access frequency. As a result, the proposed protocol is efficient from the point of communication overhead.

5.2.2 Security analysis

Before describing the security-related features provided by the proposed protocol, we compare the proposed protocol with other similar approaches whose goal is to provide anonymous interaction between the mobile user and the service provider in table 5.3. The service provider in the proposed protocol can't link two different sessions to the same user unless the user exposes his/her the anchor value. However, the service provider in the scheme proposed by Ren *et al.* [15, 16] can link two different sessions to the same user even if the user keeps his/her the anchor value secure.

Now, we describe security-related features of the proposed protocol in detail.

Mutual authentication: As, in the proposed protocol, the mobile user authenticates himself/herself to the authentication server using own authorized credential, the authentication server knows that the mobile user has proper legitimacy of the service. The authentication server also authenticates itself to the mobile user through its public key and its private key.

Anonymity: The proposed protocol protects the end-user's privacy against insiders and outsiders. As each user authenticates himself/herself to insiders (*i.e.*, service provider and other users) using the authorized credentials, the insiders cannot predict who sends the service access request or lookup request. Here, the insiders except the user cannot find any relationship among the authorized credentials in that the credentials are derived from the initial credential C^0 using oneway hash function. Outsiders also cannot identify who sends the messages since all communication messages are encrypted with the shared session key.

Non-linkability: Based on the definition in [34], non-linkability needs to prevent insiders and outsiders from obtaining any information related to an end-user (*i.e.*, authorized credential and/or

	Proposed protocol	Ren *et al.*[15, 16]	He *et al.*[14]
Concrete protocol	Yes	Yes	Yes
Mutual authentication	Yes	Yes	Yes
Anonymity	Yes	Yes	Yes
Non-linkability	Yes to outsiders, yes to SP	Yes to outsiders, partially yes to SP	No
Pseudo non-transferability	Almost yes	Almost yes	No
Data confidentiality	Yes	Yes	Easy to obtain
Message integrity	Easy to obtain	Yes	Yes
Differentiated service access control	Yes	Yes	No
Privacy concern about abuse of subscription information	None	Yes	Not considered

Table 5.3: Comparison of security-related features

membership test). In our protocol, non-linkability is achieved with respect to both insiders and outsiders. First, the information to distinguish each user is never transmitted in the plaintext form. As a result, the outsiders cannot associate a session with a particular user and ascribe two sessions to the same user. Second, the outsiders and the insiders cannot find any relationship between the exposed credentials to generate a session key due to one-way hash function. Third, as encryption result of the message m is non-deterministic, the outsiders cannot link two different sessions to the same user. Finally, all communications are protected by fresh session key.

Accountability and pseudo non-transferability: In the proposed protocol the credentials, generated by a mobile user, are authorized only when the mobile user is explicitly authenticated. By adopting a set of selected numbers, the proposed protocol can provide one-time usage of the authorized credentials. Hence, it can prevent the attacker from reusing the authorized credential. Also, proposed protocol can provide good accounting capability by incorporating an accounting function. Furthermore, the scheme provides similar non-transferability from the service point of view. Since the credentials are delegated among users only, no harm is done to the service provider in the sense that the authorized user is responsible for all the received services using her own credentials. This property greatly reduces the problem of service abuse about which the service providers are worried.

Data confidentiality and integrity: All communications except between service provider and authentication server are protected by the shared session key or the receiver's public key. Hence, data confidentiality and integrity can be easily achieved using symmetric cryptography.

Differentiated service access control: We assume that each service provider defines SID list, generates polynomial list for membership test, assigns his/her subscribers with proper polynomial list and distributes necessary information related to service subscription (*i.e.*, access key, identifier and polynomial identifier) to his/her subscribers before entity registration phase. Also, the end-users obtain their authorized credentials only if they are legitimate users with appropriate access permission on the SID. If each SID is defined using hierarchical organization, verification of the access permission against SID is converted to the membership test against the service using the submitted polynomial identifier. Hence, the proposed protocol can provide differentiated service access control based on SID.

Enhanced the level of security: Every access request message contains S, randomly generated by the mobile user and only delivered to the authentication server, used to prove the actual holder of the message. To impersonate the target user, the adversary should present S even if the adversary knows the user's anchor value. Therefore, the proposed scheme enhances the level of security.

No additional key management: Two entities among each mobile user, service provider and authentication server can generate a shared symmetric key (*e.g.*,$K_{U,AS}$, $K_{U,SP}$ and $K_{SP,AS}$) during entity authentication phase. Also, the shared key can be derived from the stored 5-tuple (C^0, R^i, j^i, n, S). Thus, there is no additional key management overhead by replacing the reduced public key encryptions (or decryptions) with the symmetric key encryptions (or decryptions). Moreover, there is no revocation cost of the shared key since the key is used only one session.

Less privacy concern regarding the abuse of subscription information: Although the authentication server is regarded as a trusted third party, the service provider may worry that the administrator of the authentication server illegally extracts the subscription information of the service provider (*e.g.*, end-user tracking and information leakage of the subscribers). In the proposed protocol, however, the administrator can only identify the subscribers of the target service provider when he/she monitors all entity registration requests. Also, a proper operation policy for authentication server can prevent illegal tracking of all registration requests. In this way, our approach reduces privacy concern of each service provider about the abuse of subscription information.

5.3 Re-authentication protocol for an end-user over wireless sensor network

5.3.1 Performance analysis

Table 5.4: Computational overhead

Each phase	Entity	Exponent Operation	BGN Encryption [44]	Hash Operation	Symmetric Key Operation
Token	U	1	0	0	0
Authorization	SP	0	1	0	0
Entity	U	0	0	4	2
Authentication	CH	2	0	4	2

Storage overhead: While an end-user should store $E[-r, PK_{BGN}, G]$, each cluster head requires to store $E[-(\alpha-1)\cdot r, PK_{BGN}, \mathbb{G}]$, $g^{-(\alpha)\cdot r \cdot SK_{BGN}}$, SK_{BGN}, $K_{U,CH}$, and R_{BS}. Let assume that $\mid PK_{BGN} \mid$, $\mid SK_{BGN} \mid$, $K_{U,CH}$, and R_{BS} are 512 bit, 512 bit, 128 bit, and 64 bit, respectively. Then, the end-user and cluster head should store 512 bit and 1728 bit.

Computational overhead: Table 5.4 illustrates the computational overhead of the proposed protocol. In entity authentication phase, the mobile node owned by an end-user should compute four hash operations and two symmetric key operations. The nearby cluster head requires two modular exponentiations, four hash operations, and two symmetric key operations.

Communication overhead: The nearby cluster head only requires one transmission and receiving message for entity re-authentication. However, the previous approaches [21, 22, 23, 24, 25, 26, 27,

28, 29] require several message exchanges among the neighbors of the cluster head. As the number of mobile nodes in the sensor network increases, the energy consumption due to the message exchanges will sharply increase. Hence, the proposed re-authentication protocol is efficient than the other protocols.

5.3.2 Security analysis

Table 5.5: Security-related features comparison

Property	Ours	LEAP [28]	HIKES [29]
Mutual Authentication	Yes	Yes	Yes
Non-linkability	Yes	Not considered	Not considered
Data confidentiality	Yes	Yes	Yes
Integrity	Yes	Yes	Yes
Resilience against node compromise	Yes	No	No
Accountability	Yes	Not considered	Not considered

Our protocol provides the following security-related features. In Table 5.5, we compare the security-related features of our protocol with the previous work [28, 29].

Mutual authentication: The end-user authenticates a nearby cluster head through the shared information (*i.e.*, R_{BS} and $E[-r, PK_{BGN}, \mathbb{G}]$) with the base station and service provider. Only if the end-user is a legitimate subscriber, the base station and service provider issue the shared information. Also, the nearby cluster head authenticates the end-user through membership verification and R_{BS}. From this point, we believe that our re-authentication protocol can support mutual authentication.

Non-linkability: Non-linkability means that, for insiders (*i.e.*, SP and DS) and outsiders, 1) neither of them can ascribe any session to a particular end-user, and 2) neither of them can link two different sessions to the same user [34]. More precisely, non-linkability needs to prevent insiders and outsiders from obtaining an end-user's private information. Our protocol can achieve non-linkability with respect to both insiders and outsiders. First, the information to distinguish each user is never transmitted in a plaintext form. As a result, outsiders cannot associate a session with a particular user and ascribe two sessions to the same user. Second, as the given $E[-r, PK_{BGN}, \mathbb{G}]$ is non-deterministic, outsiders and insiders including the cluster head cannot link two different sessions to the same user. Finally, all communications are protected by a fresh session key.

Accountability: Since the authorized token of an end-user is generated by the selected random number h^{R^i} only known to the service provider, the adversary cannot know the random number. Also, the end-user report h^{R^i} to the service provider after updating his/her authorized token.

Data confidentiality and integrity: All communications are protected by the shared key among participants. Also, we use HMAC to provide message integrity. As a result, the proposed protocol can achieve confidentiality and integrity requirements.

Resilience against node compromise: The major advantage of node capture is the acquisition of valid keys since the adversary can launch various attacks using those keys. In our protocol, the adversary can obtain $E[-(\alpha-1)\cdot r, PK_{BGN}, \mathbb{G}]$, $g^{-(\alpha)\cdot r\cdot SK_{BGN}}$, SK_{BGN}, $K_{U,CH}$, and R_{BS} by

compromising the nearby cluster head. Although the adversary can impersonate the nearby cluster head using this information, he/she cannot distinguish two different the end-users. Also, the adversary cannot generate different $E[-r, PK_{BGN}, \mathbb{G}]$ unless participating the authentication process. From this point, our re-authentication protocol has resilience against node compromise, while the existing protocols may be vulnerable against node compromise.

Scalability: By reducing the message exchanges to authenticate the mobile user, we can support more mobile users than the previous protocols [21, 22, 23, 24, 25, 26, 27, 28, 29].

Chapter 6. Conclusion

In this chapter, we conclude the dissertation by summarizing the findings obtained from this work and and providing future research directions in the authentication for the public ubiquitous computing environment.

6.1 Summary

After M. Weiser proposed the concept of ubiquitous computing environments [1], many papers are proposed to address the research challenges of ubiquitous computing environments. For service discovery, which allows the end-user to find an alternative service, Czerwinski *et al.* [2] proposed a scheme which was called "Secure Service Discovery Service". Also, Zhu *et al.* [3] introduced another secure service discovery protocol to preserve the privacy of the end-user while protecting the privacy of the service provider. To preserve the privacy of the end-user, many researchers proposed their idea [4, 5, 7, 8, 9, 10, 11, 12, 15, 16, 13, 14]. Keyword search techniques [17, 18, 19, 20] on the encrypted data are proposed to share the data while preventing the other entities having no access control from obtain the data. Various key establishment schemes [21, 22, 23, 24, 25, 26, 27, 28, 29] for wireless sensor network, one of core technologies to establish ubiquitous computing environment, have been introduce to address security issues.

Based on these research results and recent advance of the technologies (*e.g.)*, Bluetooth, Wi-Fi, sensor network, smart phone), the ubiquitous computing environments have being integrated in our real life (*i.e.*, Navitime in Japan [30], Open Urban Testbed for Ubiquitous Computing in Finland [32]). From now, we call the ubiquitous computing environment integrated in our real life as the public ubiquitous computing environment. Then, the public ubiquitous computing environments have the following properties: multiple administrative domains, complicated trust model, numerous subscribers of a service, heterogeneous devices, and numerous authentication requests. Due to these properties, the privacy-preserving protocols for the public ubiquitous computing environment should consider the following:

Challenges in Privacy protection

1. **Semi-honest directory server**: Since the directory servers owned by multiple administrate domains, the directory server should be regarded as semi-honest entity. In other words, the directory server performs its original functionalities (*i.e.*, storing the access information of the service and providing the access information to the end-user having proper access permission) while the server may collect the private information of the end-user or service provider.

2. **Semi-honest authentication server**: Since the authentication servers owned by multiple administrative domains, one authentication server cannot want to provide the subscription information, which is required to enforce proper access control, to another authentication server. While preventing the subscription information from another authentication server, one authentication server want to allow another authentication server to authenticate the end-user belonging to its authentication server.

3. **Enhanced non-linkability**: The adversary can easily eavesdrop any communication since the environments are open to all entities. If the adversary can identify the end-user or service provider, the adversary can monitor the activities of the target entity (*i.e.*, end-user or service provider) and track the location of the target entity. Here, non-linkability means that, for insiders(*i.e.*, service provider and other users) and outsiders, 1) neither of them could ascribe any session to a particular end-user, and 2) neither of them could link two different sessions to the same user [34].

Challenges in Scalability

1. **Lower processing delay time**: As the average number of the service subscriber in panOULU WLAN is about 20000, lower processing delay time is required in processing time the directory server and authentication server so that the servers can support more end-users within the certain time period.

2. **lower communication overhead**: As the various devices can be used in the network having limited bandwidth (*i.e.*, sensor network), lower communication overhead is preferred to support more end-users within the certain time period. Communication overhead can be determined by the size of the message and the number of rounds to achieve the protocol.

If we apply the previous work to the privacy-preserving authentication protocols for the public ubiquitous computing environments, we suffer the following problems as shown in Figure 6.1.

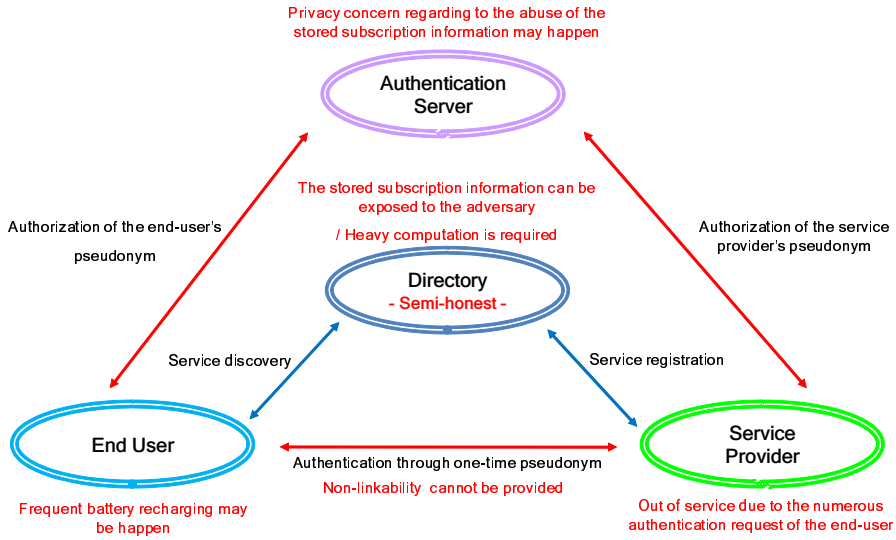

Figure 6.1: Conceptual model with the problems of the existing approaches

To address these problems, we suggest scalable privacy-preserving authentication framework consisting of three privacy-preserving authentication protocols. First protocol is proposed to address privacy violation in the directory server and authentication server. The protocol allows the directory server and authentication server to check whether the end-user has proper access permission on the requested service. In the meantime, the directory server and authentication server cannot obtain the subscription information to enforce access control since the information is encrypted by BGN encryption [44]. If we assure that the size of the message space is larger than 2^{160}, the expected decryption

time is 2^{80} [44]. As a result, the directory sever and authentication server cannot obtain the information even if they has the private key SK_{BGN}.

Second protocol is proposed to provide anonymous authentication during accessing the target service. The protocol can enhance non-linkability and remove privacy violation of the end-user in the authentication server. Compared to the previous work [15, 16], the proposed protocol can reduce computational overhead and communication cost while enhancing non-linkability. The detailed procedure is explained in the Chapter 3.

Third protocol is introduced to provide an efficient anonymous re-authentication of the end-user over wireless sensor network. Although the public ubiquitous computing environments may support the whole citizens in a city, the wireless sensor network has low-bandwidth indicating that the number of the message transmitted by the network may be lowered than the high-bandwidth network (*i.e.*, 3GPP and WLAN). That's why the efficient anonymous re-authentication protocol for end-user over wireless sensor network is considered. In order to reduce the number of the message transmission, the lightweight version of membership verification is applied. Although the end-user submits the shared secret information to the verifier (*i.e.*, the nearby sensor node or access point), the information is encrypted by BGN encryption [44]. Therefore, the proposed anonymous re-authentication protocol can address the security and privacy issues in the public ubiquitous computing environments.

To demonstrate how to apply the proposed privacy-preserving protocols to build child-care and safety service, regarded as one of the possible applications in the public ubiquitous computing environments, we discuss the system model, assumption, and the sketch of each service in the Chapter 4. In the sketch of the service, we explain the issue to be addressed in each phase and how to apply the proposed protocols. To provide how to implement the service, we illustrate the actual process for child-care and safety service. Through evaluation of the child-care and safety service, we show that the service can address security and privacy issues for the ubiquitous computing environments.

Also, we show that the privacy-preserving secure service discovery protocol, used to demonstrate how to build privacy-preserving protocols for the public ubiquitous computing environments, can reduce communication overhead while preventing the directory server from obtaining the subscription information. However, the communication cost for membership verification increases linearly as the degree of polynomial $f(x)$. To relieve this limitation we suggest that each service provider should divide all the subscribers to several subsets, which is also useful to reduce processing delay of service discovery and support differentiated access control by assigning a different privilege to each subset. If the service has 10 keywords and 30 subscribers, the processing delay time to verify proper access control is 422.33 ms. Compared to the previous approach [3], this delay is added to address the security and privacy issues of secure service discovery protocol for the public ubiquitous computing environments. Also, the added delay does not be affected by the number of the subscribers. However, the communication overhead is less than the previous approach [3]. In addition, the proposed protocol provides more useful security-related features compared with the previous approach in [3].

To prove whether the proposed privacy-preserving protocols can address the security and privacy issues, we evaluate the proposed privacy-preserving authentication protocols, anonymous authentication protocol, membership verification, and re-authentication protocol for the end-user over wireless sensor network in the Chapter 5. The standard version of membership verification only requires a constant processing delay time in the directory server (or authentication server) where the time can be adjusted by the service provider to satisfy the desire performance of his/her service. In addition, the lightweight version of membership verification only requires 1 modular addition and 1 modular ex-

ponentiation. From these evaluation results in the performance aspect, we believe that the proposed privacy-preserving authentication protocols can address scalability issue by reducing lower processing delay time and communication overhead. The detailed discussion is done in the Chapter 5.

Compared to the previous anonymous authentication protocol [15, 16], the proposed anonymous protocol reduces the processing delay time by replacing the public key operation with the symmetric key operation in the authentication server. Although the proposed protocol requires additional processing time to verify the access permission of the end-user, the time can be adjusted to a few hundred ms (121.8 $\sim$ 184.3 ms) when the number of the service subscriber is 20 and the set of the service subscriber is divided by 4. However, the previous protocol [15, 16] assumes that the subscription information should be known to the authentication server while the proposed anonymous authentication protocol does not require this assumption.

The anonymous re-authentication protocol for the end-user over wireless sensor network applies the lightweight version of membership verification. As processing time of one multiplication on Elliptic Curve $F_{2^{233}}$ provided by T. Takagi *et al.* is 4.981 ms on MicaZ [56], our membership verification only takes less than 9.97 ms. When we employ assembly implementation, the processing time can be reduce to 3.18 ms.

6.2 Future research

Although we implement membership verification and one-time pseudonym approach in order to illustrate the performance of each privacy-preserving protocol in the proposed authentication framework, we does not cover how to generate and reuse SID. We simply assume that SID is generated through hierarchical structure in the public ubiquitous computing environment. However, SID is used to identify the stored polynomial including subscription information on the service. In near future, we will introduce new technique to generate SID to satisfy the characteristics of the public ubiquitous computing environments.

Although we evaluate the lightweight version of membership verification using [44], the other homomorphic encryption schemes can be used. If we use the other homomorphic encryption schemes, We expect that the message size can be reduced. For rigorous analysis, we need to implement the other schemes and compare the result in the view of processing time and message size.

In the public ubiquitous computing environments, temporal registration is required to deal with the temporal visit of the end-user in the other administrative domain. However, temporal registration incurs the management overhead in the authentication server. The end-user should register himself/herself whenever the end-user visits the other administrative domain. Using the proposed privacy-preserving membership verification, we may reduce the management overhead incurred by temporal registration. In near future, we will introduce a concept to address this issue.

In the public ubiquitous computing environments, vehicular network, shown in Figure 6.2, may be one of core networking technologies. Since the anonymous authentication in the vehicular network requires the heavy computations such as pairing computation, indicating that the processing delay time will be increased as the number of the service subscribers increases, the existing approach cannot be used in the public ubiquitous computing environments. Due to the mobility of the vehicular, we should consider anonymous re-authentication. To address these problems, we may extend the proposed authentication framework. In the vehicular network, we should support two kind of authentications: authentication in the vehicular-to-infrastructure communication and authentication in

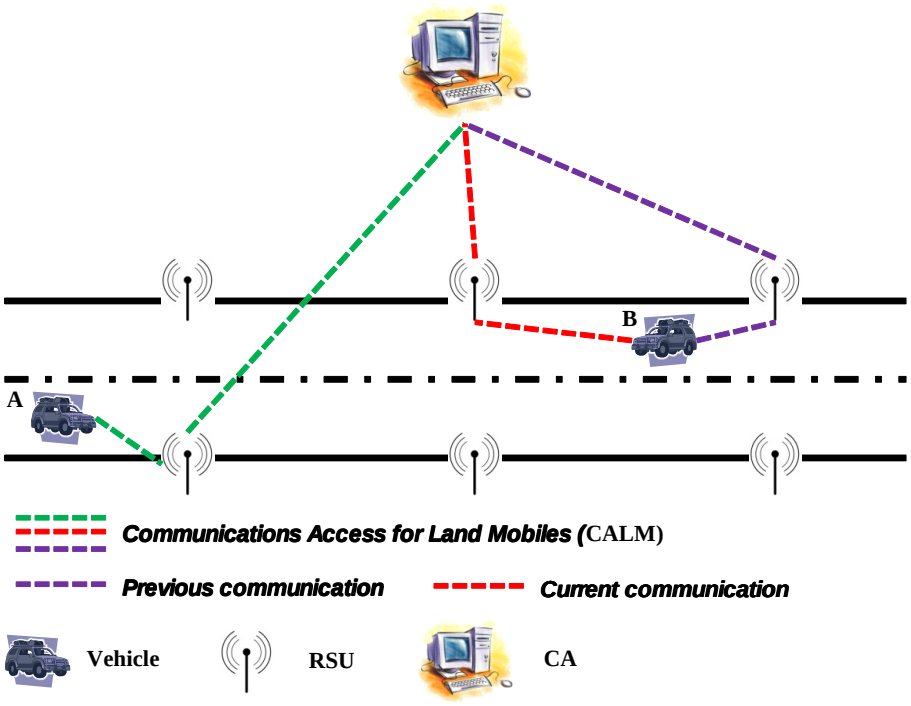

Figure 6.2: System architecture of vehicular network

the vehicular-to-vehicular communication. In order to support the anonymous authentication in the vehicular-to-infrastructure, we should reduce the processing delay time and the communication between the road-side-unit and the authentication server. When we support the anonymous authentication in the vehicular-to-vehicular, we can apply anonymous re-authentication technique to reduce the processing delay time and communication between the road-side-unit and the authentication server.

Summary

A Scalable Privacy-Preserving Authentication Framework for the Public Ubiquitous Computing Environments

1991년 M. Weiser가 유비쿼터스 컴퓨팅 환경에 대한 개념을 소개한 이후로, 많은 연구자들이 유비쿼터스 컴퓨팅 환경에 필요한 이슈를 해결하기 위한 다양한 방법들을 제시하였다. 최근 유비쿼터스 컴퓨팅 환경을 구축하는데 필요로 하는 4세대 이동통신 및 센서네트워크와 같은 기술들의 진보는 우리로 하여금 가까운 미래에 유비쿼터스 환경이 현실이 될 것으로 기대감을 준다. 다양한 연구 결과 및 기술적 진보를 바탕으로 하여금 일본의 Navitime, 핀란드의 개방형 도시 테스트베드, 한국의 자녀 안심서비스와 같은 유비쿼터스 컴퓨팅 환경이 실생활에 조금씩 다가오고 있다. 실생활에 들어온 유비쿼터스 컴퓨팅 환경을 공공 유비쿼터스 컴퓨팅 환경이라 부르고자 한다.

기존의 유비쿼터스 컴퓨팅 환경에 비해 공공 유비쿼터스 컴퓨팅 환경은 다수의 관리 도메인, 복잡해진 신뢰 모델, 다수의 서비스 가입자 및 사용자 인증, 다양한 사용자 단말과 같은 특징을 가진다. 이로 인해 접근 제어를 위해 저장되는 서비스 가입자 정보의 외부 누출, 사용자 추적 가능성으로 인한 프라이버시 침해, 높은 연산 및 통신 비용으로 인한 낮은 확장성과 같은 새로운 문제들이 야기되지만 기존의 연구 결과는 이를 해결할 수 없다.

해당 문제를 해결하기 위해 확장성 및 사용자 프라이버시가 개선된 인증 프레임워크를 제안하였다. 제안된 인증 프레임워크는 서비스 접근과정에서 인증을 위한 익명 인증 프로토콜, 서비스 검색과정에서 익명 몇 접근 제어를 위한 멤버십 검증 기법, 센세네트워크에서 데이터 획득 과정에서 인증을 위한 익명 재인증 기법으로 구성되어 있다. 제안된 프로토콜은 사용자 프라이버시를 보호하면서도 프로세싱 딜레이 시간 및 통신 비용을 감소시켜 확장성이 개선되었다. 특히, 제안된 메버쉽 검증 기법은 서비스 제공자가 다양한 서비스에서 필요로 하는 성능을 만족하기 위해 적절하게 인증 기법의 성능을 조절할 수 있게 해주면서도 서비스 가입 인원의 증가에 영향을 받지 않는다. 사용자의 프라이버시는 소인수 분해와 같은 계산 복잡한 문제 기반으로 보호된다.

난수들의 집합 및 멤버쉽 검증 기법을 통해 익명 인증은 non-linkability를 개선하였으며, 사용자 접근 권한 검증을 위해 인증 서버에 저장된 서비스 가입자 정보 누출에 관한 프라이버시 문제를 해결하였다.

익명 재인증 기법은 경량화된 멤버쉽 검증 기법을 통해 효과적이면서도 익명 인증을 제공한다. 기존에 인증된 사용자는 센서네트워크와 비밀 정보를 공유할 수 있는 점을 이용해서 해당 정보를 알고 있는가 없는가를 확인하는 방식으로 동작한다.

사용자의 프라이버시를 보호하는 안전한 서비스 검색 프로토콜을 통해 제안된 인증 프레임워크를 구성하는 인증 프로토콜들이 실제 프라이버시를 보호하는 프로토콜에 어떻게 적용되는지를 보여주었다. 또한, 공공 유비쿼터스 컴퓨팅 환경에서 제공될 것으로 기대되는 자녀안심서비스에 제안된 인증 프로토콜을 적용해 새로운 보안 및 프라이버시 문제 해결 여부를 보여주었다.

References

[1] M. Weiser, "The Computer for the 21st Century", Scientific of American, vol. 265, Sep., 1991.

[2] S. Czerwinski, B. Y. Zhao, T. Hodes, A. Joseph, and R. Katz, "An Architecture for a Secure Service Discovery Service", Proc. of Fifth annual International Conf. on Mobile Computing and Networks (MobiCom '99), Aug. 1999, pp.24-35.

[3] F. Zhu, M. Mutka and L. Ni, "A Private, Secure, and User-Centric Information Exposure Model for Service Discovery Protocols", IEEE Transactions on Mobile Computing, vol. 5, no. 4, Apr. 2006, pp.418-429.

[4] J. Al-Muhtadi, A. Ranganathan, R. Campbell and M. Mickunas, "A Flexible, Privacy-Preserving Authentication Framework for Ubiquitous Computing Environments", Proc. 22nd International Conference on Distributed Computing Systems (ICDCS), 2002, pp. 771–776.

[5] J. Al-Muhtadi, R. Campbell, A. Kapadia, D. Mickunas, and S. Yi, "Routing Through the Mist: Privacy Preserving Communication in Ubiquitous Computing", Proc. ICDCS, Vienna, Austria, 2002, pp. 65–74.

[6] J. Al-Muhtadi, A. Ranganathan, R. Campbell and M. Mickunas, "Cerberus: A Context-Aware Security Scheme for Smart Spaces", Proc. the First IEEE International Conference on Pervasive Computing and Communications (PerCom), 2003, pp. 489–496.

[7] M. Burnside et al., "Proxy-Based Security Protocols in Networked Mobile Devices", Proc. ACM SAC, Madrid, Spain, 2002, pp. 265–272.

[8] S. Creese, M. Goldsmith, B. Roscoe, and I. Zakiuddin, "Authentication for Pervasive Computing", Proc. Security in Pervasive Computing 2003, 2004, vol. 2802, pp.116–129.

[9] M. Langheinrich, "A Privacy Awareness System for Ubiquitous Computing Environments", Proc. UbiComp, 2002, vol. 2498, pp. 237–245.

[10] K. Nahanishi, J. Nakazawa, and H. Tokuda, "LEXP: Preserving User Privacy and Certifying Location Information", Proc. 2nd Workshop Security Ubicomp, 2003.

[11] M. Wu and A. Friday, "Integrating Privacy Enhancing Services in Ubiquitous Computing Environments", Workshop on Security in Ubiquitous Computing, 4th International Ubicomp, 2002.

[12] A. Zugenmaier and A. Hohl, "Anonymity for Users of Ubiquitous Computing", Proc. Security Workshop in Ubicomp, Seattle, Washington, Oct. 2003.

[13] U. Jendricke, M. Kreutzer and A. Zugenmaier, "Pervasive Privacy with Identity Management", in Proc. 1st Workshop Security, Ubicomp, 2002.

[14] Q. He, D. Wu and P. Khosla, "Quest for Personal Control over Mobile Location Privacy", IEEE Commun. Mag., vol. 42, no. 5, pp. 130–136, May 2004.

[15] K. Ren, W. Lou, "Privacy Enhanced Access Control in Ubiquitous computing Environments", 2nd International Conference of Broadband Networks 2005, Vol. 1, pp. 356–365, 3-7 Oct. 2005.

[16] K. Ren, W. Lou, K. Kim and R. Deng, "A Novel Privacy Preserving Authentication and Access Control Scheme for Pervasive Computing Environments", IEEE Transactions on Vehicular Technology, vol. 55, no. 4, pp. 1373–1384, July 2006.

[17] D. Boneh, G. D. Crescenzo, R. Ostrovsky, and G. Persiano, "Public key encryption with keyword search", Advances in Cryptology (EUROCRYPT '04), LNCS 3027, pp. 506-522, 2004.

[18] P. Golle, J. Staddon and B. Waters, "Secure Conjunctive Search over Encrypted Data", Proc. Applied Cryptography and Network Security (ACNS '05), LNCS 3089, pp. 31-45, Jun. 8-11, China.

[19] J. Baek, R. Safavi-Naini and W. Susilo, "Public Key Encryption with Keyword Search Revisited", Cryptology ePrint Archive, Report 2005/191.

[20] S. S. Yau and Y. Yin, "Controlled Privacy Preserving Keyword Search", Proc. ACM Symposium on Information, Computer & Communication Security (ASIACCS '08), pp. 321-324 , Mar. 18-20, 2008, Tokyo, Japan. ETRI Journal, Vol. 32, No. 5, October 2010, pp. 704-712.

[21] L. Eschenauer and V. D. Gligor, "A Key Management Scheme for Distributed Sensor Networks", in Proceedings of the 9th ACM conference on Computer and Communications Security (CCS), Washington, DC, USA, November 18-22, 2002, pp.41-47.

[22] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks", in IEEE Symposium on Security and Privacy, Berkeley, California, May 11-14, 2003, pp.197-213.

[23] W. Du, J. Deng, Y. S. Han and P. K. Varshney, "A Pairwise Key Predistribution Scheme for Wireless Sensor Networks", in Proceedings of the 10th ACM conference on CCS, Washington, DC, USA, October 27-31, 2003, pp.42-51.

[24] D. Liu and P. Ning, "Establishing Pairwise Keys in Distributed Sensor Networks", in Proceedings of the 10th ACM conference on CCS, Washington DC, USA, October 27-31, 2003, pp.42-61.

[25] J. Hwang and Y. Kim, "Revisiting Random Key Predistribution Schemes for Wireless Sensor Networks", in Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks, Washington DC, USA, 2004, pp.43-52.

[26] W. Du, J. Deng, Y. S. Han, S. Chen and P. K. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge", in Proceedings of the 23rd conference on the IEEE INFOCOM 2004, Hong Kong, China, March 7-11, 2004.

[27] M. G. Sadi, D. S. Kim and J. S. Park, "GBR: Grid Based Random Key Predistribution for Wireless Sensor Network", in Proceedings of the 11th Annual IEEE International Conference on Parallel and Distributed Systems (ICPADS '05), Vol. 2, July 20-22, 2005, pp.310-314.

[28] S. Zhu, S. Setia and S. Jajodia, "LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks", in Proceedings of the 10th Annual Conference on CCS, Washington DC, USA, October 27-31, 2003, pp.62-72.

[29] J. Ibriq and I. Mahgoub, "A Hierarchical Key Management Scheme for Wireless Sensor Networks", in Proceedings of 21st International Conference on Advanced Networking and Applications (AINA 2007), May 21-23, 2007, Niagara Falls, Canada, pp.210-219.

[30] M. Arikawa, S. Konomi, and K. Ohnishi, "Navitime: Supporting Pedestrian Navigation in the Real World", IEEE Pervasive Computing, Vol. 6, No. 3. (2007), pp. 21-29.

[31] T.R. Hansen, J.E. Bardram, and M. Soegaard, "Moving Out of the Lab: Deploying Pervasive Technologies in a Hospital", IEEE Pervasive Computing, Vol. 5, No. 3, (2006), pp. 24-31.

[32] T. Ojala, "Open Urban Testbed for Ubiquitous Computing", in Proceedings of 2010 International Conference on Communications and Mobile Computing (CMC 2010), April 12-14, 2010 Shenzhen, China, pp. 442-447.

[33] u-Safe Gangnam Home Page, `http://usafe.gangnam.go.kr/u-safe_01.html`

[34] S. Xu and M. Yung, "K-anonymous Secret Handshakes with Reusable Credentials", in Proc. ACM Conf. CCS, pp. 158–167, 2004.

[35] M. Gruteser and D. Grunwald, "Enhancing Location Privacy in Wireless LAN Through Disposable Interface Identifiers: A Quantitative Analysis", Mobile Networks and Applications, vol. 10, no. 3, pp. 315–325, 2003.

[36] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol", IETF RFC 2401, 1998.

[37] K. Takata, J. Ma and B. O. Apduhan, "A Dangerous Location Aware System for Assisting Kids Safety Care", In 20th International Conference on Advanced Information Networking and Applications, Vienna, Austria, April 18-20, 2006; pp. 657–662.

[38] Assisted GPS, Wikipedia, `http://en.wikipedia.org/wiki/Assisted_GPS` (accessed 12/17/09).

[39] N. Priyantha, A. Chakraborty and H. Balakrishnan, "The Cricket Location-Support System", in Proceedings of the 6th annual international conference on Mobile computing and networking, Boston, Massachusetts, USA, August 6-11 2000, pp. 32–43

[40] P. Bahl and V. N. Padmanabhan, "RADAR: An in-building RF-based user location and tracking system", In Proceedings of IEEE Infocom 2000, Tel Aviv, Israel, March 26-30, 2000; pp.775–784.

[41] G. V. Zàruba, M. Huber, F. A. Kamangar and I. Chlamtac, "Indoor location tracking using RSSI readings from a single Wi-Fi access point", Wireless Networking, 2007, 13, 221–235.

[42] M. J. Freedman, K. Nissim and B. Pinkas, "Efficient Private Matching and Set Intersection", Advances in Cryptography (EUROCRYPT '04), LNCS 3027, pp. 1-19, 2004.

[43] L. Kissner and D. X. Song, "Privacy-preserving Set Operations", Proc. 25th Annual International Cryptology Conference (CRYPTO '05), LNCS 3621, pp. 241-257, 2005.

[44] D. Boneh, E.-J. Goh and K. Nissim, "Evaluating 2-dnf formulas on ciphertexts", Theory of Cryptography (TCC 2005), LNCS 3378, pp. 325-341, 2005.

[45] Homomorphic encryption, http://en.wikipedia.org/wiki/Homomorphic_encryption (accessed 05/04/11).

[46] D. Boneh and M. Franklin, "Identify based encryption from the Weil pairing", SIAM Journal of Computing, Vol. 32, No. 3, pp.586-615 (Extended abstract in Proceedings of Crypto 2001).

[47] D. Chaum, "Untraceable Electronic Mail, Return Address, and Digital Pseudonyms", Communications of the ACM, vol. 24, no. 2, pp. 84–88, 1981.

[48] R. Beckwith, D. Teibel, and P. Bowen, "Pervasive Computing and Proactive Agriculture", in Adjunct Proceedings Pervasive Computing and Proactive Agriculture, 2004, Vienna, Austria, April 2004.

[49] C. Kappler and G. Riegel, "A Real-World, Simple Wireless Sensor Network for Monitoring Electrical Energy Consumption", in Proceedings of First European Workshop on Wireless Sensor Networks, Berlin, Germany, January 2004, pp. 339-352.

[50] T. Moore, "A Collusion Attack on Pairwise Key Predistribution Schemes for Distributed Sensor Networks", in Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOMW '06), March 13-17, 2006, pp.251-255.

[51] R. M. S. Silva, N. S. A. Pereira and M. S. Nunes, "Applicability Drawbacks of Probabilistic Key Management Schemes for Real World Applications of Wireless Sensor Networks", in Proceedings of the Third International Conference on Wireless and Mobile Communications (ICWMC '07), March 4-9, 2007, pp.51.

[52] C. Hartung, J. Balasalle and R. Han, "Node Compromise in Sensor Networks: The Need for Secure Systems", Technical Report CU-CS-990-05, January, 2005.

[53] J. Polastre, J. Hill and D. Culler, "Versatile Low Power Media Access for Wireless Sensor Networks", in Proceedings of the 2nd International Conference on Embedded Networked Sensor Systems, Baltimore, MD, USA, November 3-5, 2004; pp. 95–107.

[54] Cognition against social safety in 2008, http://www.kosis.kr/ (accessed 12/17/09), written in Korean. (accessed 12/17/09), written in Korean.

[55] B. Lynn, Pairing Based Cryptography, http://crypto.stanford.edu/pbc/.

[56] M. Shirase, Y. Miyazaki, T. Takagi, D. Han, and D. Choi, "Efficient Implementation of Pairing Based Cryptography on a Sensor Node", IEICE Transactions on Information and Systems, Vol E92.D, No. 5, pp 909-917, 2009.

[57] A. Menezes, T. Okamoto, and S. Vanstone, "Reducing elliptic curve logarithms to logarithms in a finite field", In STOC '91: Proceedings of the twenty-third annual ACM symposium on Theory of computing, pp. 80-89, New York, NY, USA, 1991. ACM Press.

[58] G. Frey and H. Ruck, "A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves", Math. of Computaions, 62:865-874, 1994.

Acknowledgement

이 논문을 완성하기까지 주위의 모든 분들로부터 수많은 도움을 받았습니다. 김광조 교수님께서는 틈틈히 연구 상황을 확인해 주셔서 체계적인 연구방향을 세울 수 있었습니다. 게다가 성철이형, 진이형에게 연구실 생활 및 연구 주제 등 많은 부분에서 격려 및 조언을 받았습니다. 또한, 바쁘신 와중에도 학위논문심사를 위해 참석하셔서 진심어린 조언을 주신 김대영 교수님, 이동만 교수님, 이병천 교수님, 최두호 박사님께 감사드립니다. 연구실 선·후배인 현록이형, 규석이형, Liem, Duc, Divyan, 노한영, 민혜누나, 신승목, 최임성, 유명한, 임준현, 정도영, 박이재, Made 모두에게 감사합니다.

끝으로 오늘의 제가 있을 수 있도록 사랑으로 키워 주신 어머니, 무뚝뚝하지만 항상 그 자리에 계셔주신 아버지에게 감사드립니다. 저의 이 작은 결실이 그분들께 조금이나마 보답이 되기를 바랍니다.

Curriculum Vitae

Name : Jangseong Kim

Date of Birth : November 12, 1979

Birthplace : 69-3, Dongchonangil, Dong-gu, Daegu, 701-801 KOREA

Domicile : 301-6, Ansimro, Dong-gu, Daegu, 701-801 KOREA

Address : 291 Daehak-ro, Yuseong-gu, Daejeon 305-701, KOREA

E-mail : jskim.withkals@kaist.ac.kr

Educations

1998. 3. – 2006. 2.	Computer Engineering, Kyungpook National University (B.S.)
2006. 2. – 2009. 2.	Engineering, Information and Communications University (MS/Ph. D integrated course)
2009. 3. –	Information and Communications Engineering, KAIST (MS/Ph. D integrated course)

Career

2005. 3. – 2005. 8.	Development of Location Supporting System for missing children in Wireless Sensor Network
2005. 5. – 2005. 7.	Development of Certificate verification in mobile phone
2007. 6. – 2007. 8.	Computer Architecture, Graduate Teaching Assistant, Information and Communications Univ.
2007. 9. – 2007. 12.	Computer Architecture, Undergraduate Teaching Assistant, Information and Communications Univ.
2008. 6. – 2008. 8.	Computer Architecture, Graduate Teaching Assistant, Information and Communications Univ.
2008. 11. – 2009. 2.	I^2R's Postgraduate Attachment program, Singapore.
2009. 9. – 2009. 12.	Freshmen design course, Undergraduate Teaching Assistant, KAIST.
2010. 9. – 2010. 12.	Cyber security, Graduate Teaching Assistant, KAIST.

Publications

1. **Jangseong Kim**, Zeen Kim and Kwangjo Kim, *A Lightweight Privacy Preserving Authentication and Access Control Scheme for Ubiquitous Computing Environment*, The 10th International Conference on Information Security and Cryptology (ICISC '07), LNCS 4817, pp.37-48, Nov. 29-30, 2007, Seoul, Korea.

2. **Jangseong Kim** and Kwangjo Kim, *A Lightweight, Privacy Preserving and Secure Service Discovery Protocol in Ubiquitous Computing Environment*, The 9th International Workshop on Information Security Applications (WISA '08), Short paper presentations, Sep. 23-25, 2008, Jeju Island, Korea.

3. **Jangseong Kim**, Joonsang Baek, Kwangjo Kim and Jianying Zhou, *A Privacy-Preserving Secure Service Discovery Protocol for Ubiquitous Computing Environments*, 7th European Workshop on PUBLIC KEY SERVICES, APPLICATIONS AND INFRASTRUCTURES (EuroPKI '10), Sep. 23-24, 2010, Athens, Greece.

4. Zeen Kim, **Jangseong Kim** and Kwangjo Kim, *Key Predistribution Scheme for Wireless Sensor Networks with Higher Connectivity*, Symposium on Cryptography and Information Security (SCIS '07), Abstracts pp.235, Jan. 23-26, 2007, Sasebo, Japan.

5. Zeen Kim, **Jangseong Kim**, Youngdoo Kang, Kwangjo Kim, Dai I. Kim and Choong Heui Jeong, *Guideline of Cyber Security Policy for Digital I&C Systems in Nuclear Power Plant*, Transactions of the Korean Nuclear Society Autumn Meeting, Oct. 25-26, 2007, Pyeongchang, Korea.

6. 김진, **김장성**, 강영두, 정충희, 김광조, *원자력 발전소 디지털 시스템이 보안 요구사항*, 한국정보보호학회 동계학술대회, vol. 17, no.2, pp.248-251, 2007년 12월 1일, 상명대학교, 서울.

7. Hanyoung Noh, **Jangseong Kim**, Chanyeob Yeun and Kwangjo Kim, *New Polymorphic Worm Detection based on Instruction Distribution and Signature*, Symposium on Cryptography and Information Security (SCIS '08), Jan. 22-25, 2008, Miyajaki, Japan.

8. **김장성**, 권미영, 김이형, 곽민혜, 한규석, 김광조, *감시정찰 센서네트워크 및 주요 시설물 관리에서의 키 관리 기법 비교*, 한국정보보호학회 충청지부 학술발표회 논문집, pp.75-38, 2008년 10월 17일, 배재대학교, 대전.

9. 노한영, **김장성**, 김광조, *포렌식을 고려한 휴대폰 개인정보 보호 기법*, 한국정보보호학회 동계학술대회, pp.66-69, 2008년 12월 6일, 고려대학교, 서울.

10. Minhea Kwak, **Jangseong Kim** and Kwangjo Kim, *Desynchronization and Cloning resistant light-weight RFID authentication protocol using integer arithmetic for low-cost tags*, Symposium on Cryptography and Information Security (SCIS '09), Jan. 20-23, 2009, Otsu, Japan.

11. Myunghan Yoo, **Jangseong Kim** and Kwangjo Kim, *A Secure Clustering Scheme over an Energy-aware Routing Protocol for Monitoring Critical Conditions*, Symposium on Cryptography and Information Security (SCIS '09), Jan. 20-23, 2009, Otsu, Japan.

12. Dang Nguyen Duc, **Jangseong Kim** and Kwangjo Kim, *Scalable Grouping-proof Protocol for RFID Tags*, Symposium on Cryptography and Information Security (SCIS '10), Jan. 19-22, Takamatsu, Japan.

13. **Jangseong Kim**, Myunghan Yoo and Kwangjo Kim, *A privacy-preserving kid's safety care service based on sensor network in u-City*, Symposium on Cryptography and Information Security (SCIS '10), Jan. 19-22, Takamatsu, Japan.

14. Xiaofeng Chen, Fangguo Zhang, Haibo Tian, Qianhong Wu, Yi Mu, **Jangseong Kim** and Kwangjo Kim, *Three-round Abuse-free Optimistic Contract Signing With Everlasting Secrecy*, Financial Cryptography and Data Security (FC '10), LNCS 6052, pp. 304-311, Jan. 25-28, Tenerife, Canary Islands, Spain.

15. Kyusuk Han, **Jangseong Kim**, Kwangjo Kim and Taeshik Shon, *Efficient Sensor Node Authentication via 3GPP Mobile Communication Networks*, 17th ACM Conference on Computer and Communications Security (ACMCCS '10), Oct. 4-8, 2010, pp.687-689, Chicago, IL, USA

16. **Jangseong Kim**, Kwangjo Kim, Taeshik Shon and Jonghyuk Park, *A Scalable and Privacy-preserving child-care & safety service in ubiquitous computing environment*, will be published in a journal of Mathematical and Computer Modelling, Elsevier (SCIE).

17. **Jangseong Kim**, Taeshik Shon and Kwangjo Kim, *Location-aware and privacy-preserving approach for child-care and safety in ubiquitous computing environment*, IEICE Communications, Vol. E94-B, No. 3, pp.686-689, Mar 2011.

18. **Jangseong Kim** and Kwangjo Kim, *A Scalable and Robust Hierarchical Key Establishment Scheme for Mission-Critical Applications over Sensor Networks*, will be published in Journal of Telecommunication Systems (JTS), Springer (SCIE).

19. Yi Jae Park, Doyoung Chung, Made Harta Dwijaksara, **Jangseong Kim** and Kwangjo Kim, *An Enhanced Security Policy Framework for Android*, 2011 Symposium on Cryptography and Information Security (SCIS 2011), Jan. 25-28, 2011, Kokura, Japan.

20. Doyoung Chung, Made Harta Dwijaksara, Yi Jae Park, **Jangseong Kim** and Kwangjo Kim, *An Efficient and Privacy Preserving Authentication Protocol for HAN*, 2011 Symposium on Cryptography and Information Security (SCIS 2011), Jan. 25-28, 2011, Kokura, Japan.

21. Made Harta Dwijaksara, Doyoung Chung, Yi Jae Park, **Jangseong Kim** and Kwangjo Kim, *Secure, Fast Rebuilding and Energy Efficient Routing Protocol for Mission Critical Application over Wireless Sensor Networks*, 2011 Symposium on Cryptography and Information Security (SCIS 2011), Jan. 25-28, 2011, Kokura, Japan.

22. 손태식, 박용석, 한규석, 김광조, **김장성**, Patent, *이동통신망을 이용한 싱크 인증 시스템 및 방법*, 출원번호 10-2009-0114725.

23. 손태식, 박용석, 한순섭, 김광조, 한규석, **김장성**, Patent, *센서 네트워크에서 센서 노드 인증 방법 및 장치*, 출원번호 10-2009-0057778.

24. 손태식, 박용석, 한순섭, 김광조, 한규석, **김장성**, Patent, *센서 네트워크에서 노드와 싱크간의 상호 인증 시스템 및 방법*, 출원번호 10-2009-0057175.