박 사 학 위 논 문

Doctoral Thesis

센서 네트워크와 이동 통신망을 위한 보안 기술 연구

Advanced Security Schemes in Sensor Networks and Mobile Networks

한 규석 (韓 圭奭 Han, Kyusuk)

정보통신공학과

Department of Information and Communication Engineering

한 국 과 학 기 술 원

Korea Advanced Institute of Science and Technology

2010

센서 네트워크와 이동 통신망을 위한 보안 기술 연구

Advanced Security Schemes in Sensor Networks and Mobile Networks

Advanced Security Schemes in Sensor Networks and Mobile Networks

Advisor : Professor Kwangjo Kim

by

Han, Kyusuk
Department of Information and Communication Engineering
Korea Advanced Institute of Science and Technology

A thesis submitted to the faculty of the Korea Advanced Institute of Science and Technology in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of Information and Communication Engineering

Daejeon, Korea
2010. 5. 24.
Approved by

Professor Kwangjo Kim
Advisor

센서 네트워크와 이동 통신망을 위한 보안 기술 연구

한 규석

위 논문은 한국과학기술원 박사학위논문으로 학위논문심사 위원회에서 심사 통과하였음.

2010년 5월 24일

심사위원장 김 광 조 (인)

심사위원 김 대 영 (인)

심사위원 이 병 천 (인)

심사위원 최 두 호 (인)

심사위원 한 영 남 (인)

DICE
20055186

한 규석. Han, Kyusuk. Advanced Security Schemes in Sensor Networks and Mobile Networks. 센서 네트워크와 이동 통신망을 위한 보안 기술 연구. Department of Information and Communication Engineering . 2010. 113p. Advisor Prof. Kwangjo Kim. Text in English.

Abstract

Recent advance of wireless sensor network and mobile communication network technologies bring several new issues such as the mobility of sensor nodes, the deployment of PKI. Moreover, the convergence of such different networks are one of rising issues.

Mobility of sensor node in Wireless Sensor Networks (WSN) brings security issues such as re-authentication and tracing the node movement. However, current security research on WSN are insufficient to support such environments since their designs only considered static environments. In this thesis, we propose efficient node authentication and key exchange protocols that reduces the overhead in node re-authentication and also provides untraceability of mobile nodes. We propose not only symmetric key based authentication protocol, but also the asymmetric key based authentication protocol for supporting the advanced sensor technologies. Compared with previous protocols, our protocol has only a third of communication and computational overhead. We expect our protocol to be an efficient solution that increases the lifetime of sensor network. We also propose the sensor node authentication protocol in the 3G-WSN network that the sensor network is integrated with mobile network.

For the deployment of PKI into the next generation mobile network, we propose the security protocols for the application security architecture, Voice over IP application, and the lawful interception protocol deploy the ID based cryptosystem. In each chapter we analyze the efficiency and the security of our protocols by comparing with previous protocols.

Contents

List of Tables

List of Figures

1. Introduction

This chapter provides an introduction and organization of this thesis. The contributions of the thesis are also described.

1.1 Advanced Wireless Sensor Network Issues

Wireless Sensor Network (WSN) is the network that consists of lightweight devices with short-ranged wireless communication and battery-powered. The devices have the sensors that gather environmental information. After sensing this information, the devices send it to the networks. We define such devices as sensor node, and the core parts of the network as sinks or the base station as in Figure 1.1 (a).

Authenticated key distribution in WSN is one of the fundamental security problems. Employing the security protocols in other computer networks is insufficient due to the limited resources of lightweight devices. Thus, the most important issues in security researches on WSN are designing resource-efficient security protocol. Several approaches such as key pre-distribution, pairwise key agreement, group key based key agreement, and hierarchical key management schemes were introduced for the efficient authenticated key distribution.

Zigbee [16] specifies the key pre-distribution method that stores the master secret between two entities for commercial application. It requires management of large number of keys, which is hard to be scalable. The pairwise key agreement protocols based on the random key pre-distribution that enables the share the pairwise key from the pre-distributed key pool are proposed in [24], [13] and [22]. For the group key based key agreement, Zhu *et al.* [84] showed the efficient key distribution model with cluster key that reduces the overhead of the base station. Recently, the hierarchical key management schemes that the sensor nodes establish the hierarchy for the key distribution are proposed by [3] and [35]. Abraham and Ramanatha [3] showed the hierarchical architecture that construct the sensor networks. Ibriq [35] showed the concept of partial key escrowing that delegate the authentication and key distribution to sensor nodes.

1.1.1 Privacy and Mobility of Sensor Node

The advance of WSN brings several new concepts such as node mobility. In early stage, it was already expected that the sensor network must be dynamic that sensors may fail or new sensors may be added, and experience changes in their position, reachability, available energy, and even task details [23]. After that, Wireless Sensor and Actor Network (WSAN) is introduced that is the extension of WSN with node mobility [4, 43, 17]. In such environments, the network combines static sensor nodes and the mobile sensor nodes. In the thesis, we define such environments as *Dynamic Wireless Sensor Networks* as in Figure 1.1, b).

However, most previous authenticated key management protocols that only considered static environments are not sufficient to be applied to the advanced WSN with the mobile nodes. It is obvious that the forthcoming WSN will be the combined network of static sensor network and the mobile sensor and actor networks. For example, Wireless Sensor and Actor Network (WSAN) brings the concept of mobility as the extension of WSN [43, 17] . It is obvious that the wireless sensor network will be the combined network of static sensor network, and the mobile sensor and actor networks. In such environments, handling a large overhead from frequent node re-authentication requests due to the continuous node movements and the threats of tracing the node movement are important security issues.

Thus, the efficient re-authentication and untraceability are the important security requirements in WSN with mobile nodes. Even Fantacci *et al.* [25] argued the possible presence of mobile node, and proposed the authentication protocol supporting node mobility that does not require any sink or base station for authentication and key distribution, their model still occurs the large communication overhead in node re-authentication.

1.1.2 Deploying PKI in Wireless Sensor Network

Although Public key infrastructure (PKI) enables the strong and advanced security services, most previous studies focused on the symmetric key crypto-system based approach due to the insufficient computational resources for PKI of the sensor nodes. However, many efforts that enables PKI for sensor networks such as TinyPK [83] and TinyECC [46] are continually proposed.

In order to reduce the communication overhead from the key establishment, Huang *et al.* [33] proposed self-organizing algorithm by using Elliptic Curve Cryptography (ECC) [46]. Once the certificates are issued to nodes, nodes can self establish the pairwise key with exchanging the certificates with any nodes. Although the public key based security architecture requires more computational power and resources, efficient applications for

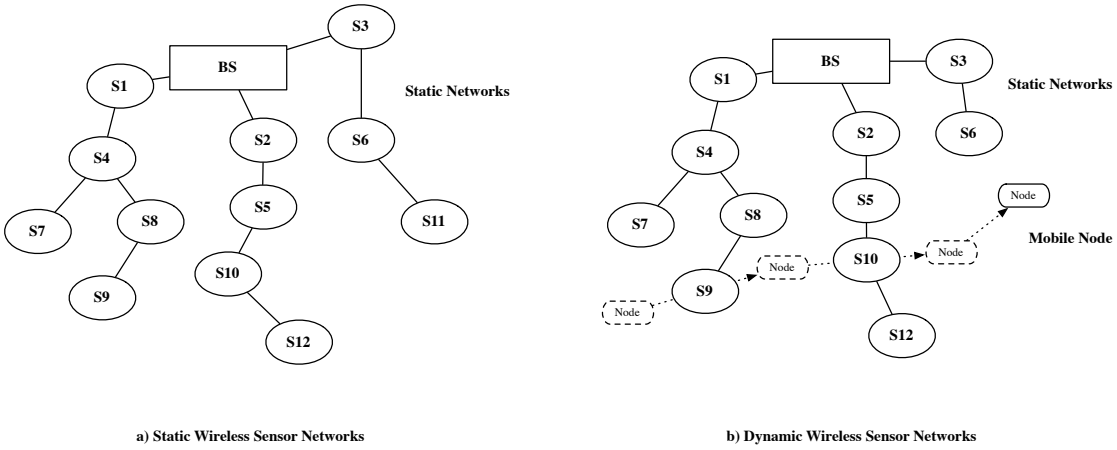


Figure 1.1: Wireless Sensor Networks (a) Static Networks (b) Dynamic Networks

the sensor networks will be available in the near future with light weight implementation such as TinkPK [83] and TinyECC [46].

1.2 Next Generation Mobile Network

1.2.1 Integration between Different Network

Recently, the convergence of the various communication technologies such as Third Generation (3G) mobile communication networks, Wireless Sensor Networks (WSNs), Wireless Local Area Network (WLAN), and Mobile WiMAX are one of the emerging trends of the next generation ubiquitous network (NGUN), and there are several efforts for their consolidation.

For example, Third Generation Partnership Project (3GPP) specifies interworking architecture between UMTS and GSM [77]. Kim *et al.* [41] specifies the interworking architecture between Wi-Fi based wireless communication devices and home area network connectivity devices. In recent years, there are several attempts to integrate WSNs and 3G mobile networks in order to provide various ubiquitous convergence applications and services [14]. In converged communication environments, the mobile and wireless communication devices will have a wide range of communication capabilities as illustrated in Fig. 1.2.

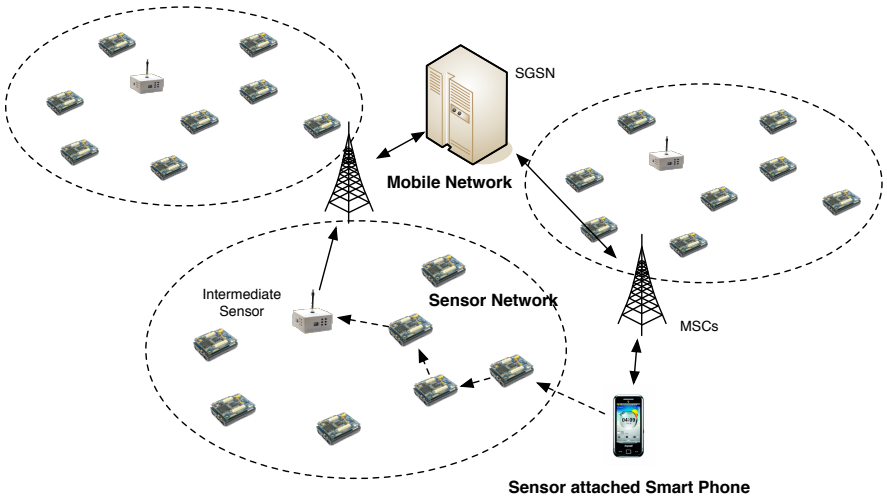


Figure 1.2: In 3G-WSN networks, the sensor attached smart phone can communicate to both sensor networks and mobile networks.

While most smartphones have the capabilities of 3G mobile communication, Bluetooth, and Wi-Fi, there is also the trial of consolidating the Zigbee/RF4CE [16] module into a Universal Subscriber Identity Module (USIM) and microSD [14, 59]. In addition, a few mobile gadgets show the multi-functionalities supporting 3G, WiFi, Bluetooth, Zigbee/RF4CE into one device as a type of being called smartbook in CES 2010 [27].

However, such integration works have been mainly progressing around the 3G mobile networks for simply connecting the sensor networks to the wide area networks (WANs) to provide basic services based on gathered information in WSN. Deploying the 3G mobile networks for the intermediate connections between WSNs and WANs could reduce the communication overhead of WSNs. However there still exist some limitations and inefficiency since the gaps of capabilities such as bandwidth, range, and speed between the 3G mobile network and the WSNs are significantly huge. In addition, due to the lack of consideration on the secure interworking in 3G mobile networks and WSNs (3G-WSN) integrated environments, most security studies remain on the WSN-only environments. Although there is an effort to overcome such drawbacks and to provide high capabilities in the 3G-WSN networks, such design is based on own architecture [63] rather than Zigbee.

1.2.2 Public Key Management in the Mobile Environment

Recent developments of mobile communication technologies [76] require the deployment of the public key based security architecture for more advanced applications beyond the symmetric key based security architectures [71, 74]. However, the large computational overhead of PKI brings the public key management issue. Storing public keys and security computation in USIM with 5-40 Mhz clock speed could be the bottleneck in advanced security services. Since the public key based security operations such as signature generations require much higher computational overhead, the operations depend on the computational power of USIM. Even more advanced USIM technologies with high capability are shown by telecommunication manufacturers [54], the capabilities of USIM are still weaker than the mobile equipments (ME). Storing public key pairs in ME weaken the strength of key storage as claimed in [75], while storing keys in USIM has the computational overhead problem due to the security computations operated in USIM.

Although several researches such as 'key-insulated' encryption [20] and signature scheme [21, 56] are proposed to be resilient against the key leakage, their designs did not consider mobile environments that the key losses occasionally happen. Moreover, their designs are related to the specific protocols and insufficient to support the various applications in mobile network. Deploying ID-based cryptosystem (IDBC) [11, 66] enables the simplified key management, and we show the IDBC based security architecture for the next generation mobile networks.

1.2.3 Secure Voice over IP

Internet Phone such as Skype is becoming more popular with widely deployed broadband Internet. Internet Phone is based on Voice over Internet Protocol (VoIP). Currently Session Initiation Protocol (SIP) based VoIP application is commonly presented [61]. Since SIP based VoIP is based on TCP/IP, the same security services in the computer networks are used for the security of VoIP. For example, HTTP digest authentication between VoIP user and servers, SSL/TLS among servers and S/MIME for the message authentication are currently recommended as the standard solutions. However, such generic security applications are not optimized for VoIP.

Thus, there are several studies for reducing the overall communication overheads for the security. To reduce user's overhead, RFC 4474 [58] defines the VoIP server of user side signs the SIP message. For example, users send their SIP messages to the VoIP server while the server signs the messages. Users do not provide the security of SIP message. However, there are high overheads in the server with large number of SIP transactions.

To reduce the server's overhead, Kong et al. [42] proposed the scheme that users create their own public key pairs and the servers share the information of the public key. Such conventional public key computations also require the public key management overhead. To reduce such overhead, Ring et al. [60] proposed the authentication and key agreement protocol for the VoIP employing IDBC. However, IDBC requires larger computation overhead than generic PKI, and such overhead could result the call set-up delay.

1.2.4 Lawful Interception of Secure Communication based on Recent Security Primitives

The lawful interception (LI) is inevitably required for protecting the national security or for detecting the criminal evidence, but should be allowed under strict guidelines and regulations. Several technical specifications for the LI such as [7, 70, 72, 73] are designed to satisfy such restrictions. For the LI on the secure communications, such regulations state that the network service providers should provide the proper decryption method for the request of law enforcement agency (LEA). Thus, the secret keys of network service subscribers are escrowed and provided for the request of the LEA. After the permission of the LI is expired, it should be disabled that the LEA uses the secret key. Also, the network subscribers should not recognize whether they are under surveillance [70].

While the current security architecture of mobile communication networks is widely based on the symmetric key cryptosystem that share secret keys between subscribers and the service provider [71], the advance of the communication technologies brought the IP based communication such as Voice over IP [40, 64]. Also, the communication is not limited to the two-pass communications such as the voice and video conversation, but include the one-way data communications such as SMS/MMS and e-mail services.

In such environments, escrowing the symmetric session key is not sufficient for supporting the LI of the advanced security services such as the digital signature. Thus escrowing the asymmetric key is necessary to support the LI of the secure one-way communications. For example, using the private key of the receiver can only decrypt the secure e-mail that has been encrypted by using the public key, the private key should be provided to the LEA.

Since the public key has much longer lifetime than the symmetric session key, it cannot be technically prevented from the LEA illegally eavesdrop the communication if the public key is not updated. Thus, the existing key escrow models focus on limiting the capability of the LEA [51, 67, 36, 82, 26, 1]. However, those works have the problem that subscribers should participate in escrowing the public key pairs in order to limit the

warrant bound of LI using their models. Such processes conflict with the LI requirements that the subscribers never be noticed or recognize whether their communications are under surveillance.

Moreover, there is lack of consideration of the LI using the ID-based cryptosystem (IDBC) [66]. Studies on IDBC are mostly introduced after the interest on the key escrowing model is moved to the industry. Also, the inherent property that the key escrowing is initially available stunted the interest on the key escrowing of IDBC. By using IDBC, the LEA could self-generate the private key of each user from the escrowed master key. However, the inherent property of IDBC for the LI occurs two significant shortcomings: One is that the LEA can also generate any keys without legal permission until the master key is updated due to every subscriber's private key is generated from the master key. The other is that the update of a single private key of a subscriber is infeasible. Thus, the update of the public key pair in IDBC occurs large communication and computational cost.

1.3 Organization

We briefly outline the structure of this thesis as follows:

Chapter 2 introduces the previous work on the sensor network security and 3GPP based mobile network security. We first briefly introduce the ZigBee standard and the previous security protocols in WSN. We then introduce the security architecture for 3rd party application support and the interworking between 3GPP and non-3GPP networks.

Chapter 3 introduces the efficient mobile node authentication and key agreement schemes in dynamic wireless sensor network. We propose two protocols: symmetric cryptosystem based protocol and hybrid protocol that deploys both symmetric key cryptosystem and public key cryptosystem. We also introduce the concept of '*Neighbor Sink List*' (NSL) that enables the deployment of our protocol in real environments. Using NSL, each sink share the neighbor sink information easily, and re-authenticate the mobile node that is once authenticated in the network. We also show the efficiency by comparing with previous work and analyze the security.

Chapter 4 introduces an efficient and secure authentication and key exchange protocol between sensor nodes and the smartphone with sensors in the consolidation of WSNs and 3G mobile network (3G-WSN). Our protocols are applicable to the standard ar-

chitecture such as IEEE 802.15.4 based Zigbee and 3GPP mobile network architectures, and integrated the sensor network into the 3G mobile network as an application based on the standard Generic Authentication Architecture (GAA) [78, 74], and minimized the communication and computation overheads in the sensor network for mutual authentication between a sensor attached smartphone and a sensor node.

Chapter 5 introduces the security protocols for the next generation mobile networks deploying public key cryptosystem. We first argue the key management problem of public key cryptosystem, and then we introduce the ID-based cryptosystem (IDBC). We improve the 'key-insulated' model [20, 21, 56] and show 'Trust Delegation' model resilient against not only the key exposure but also the key loss, and to provide the secure and efficient public key management for the next generation mobile networks. Also, we propose the secure Voice over IP applying the IDBC in order to reduce the call setup delay in the secure communication. We also show the implementation result and the design analysis of our design. Finally, we propose a new robust and feasible key escrow model for securing communications based on IDBC that not only overcome the shortcomings of the previous key escrowing models for the lawful interception in the mobile networks, but also enable efficient update of a single private key that overcome the inherent threat of IDBC. Our new model also demonstrates the efficiency in the public key management. We first describe the information on the lawful interception architecture and the key escrowing models, and propose the new key escrow protocol based on IDBC. We show the correctness of our protocol by comparing with previous work.

2. Previous Work in Sensor Networks and Mobile Networks

In this chapter, we describe the previous work in sensor networks and mobile networks. Section 2.1 shows the brief sensor network standard and security researches. We also briefly describe the security standards and studies on mobile network in Section 2.2.

2.1 Security Studies on Wireless Sensor Networks

In this section, we describe the previous work on the authenticated key agreement in wireless sensor networks.

2.1.1 Brief Overview of ZigBee Standard

ZigBee global specification [16] is the current *de-facto* standard for operating low-cost, low-power devices in a wireless sensor network. The ZigBee stack builds upon the IEEE 802.15.4 standard that specifies the characteristics of the physical and medium-access control layer for wireless low-rate personal area networks (WPANs). The radio transceiver of a ZigBee device supports data transmission at a rate of 250 Kb/s if the radio is operated in the 2.4 GHz frequency band.

The higher layers comprise the network (NWK) layer, an application support (APS) layer, the security service provider (SSP), the ZigBee device object (ZDO) and the application objects. The NWK layer is in charge of organizing a multi-hop network and routing data packets over it. The SSP unit provides a security service including ensuring freshness of data, message integrity, network and node level authentication, and encryption. The APS layer is responsible for binding together devices based on their service needs in order to exchange application messages between them. The ZDO entity provides the service to discover other devices and application objects in the network. Figure 2.1 shows the security architecture by ZigBee Aliance.

Zigbee [16] specifies the key agreement architecture that pre-distribute keys. In their architecture, each node pre-installs their unique keys, such as the master key (MK) and the link key (LK), that are shared to other entities and the network key (NK) that is

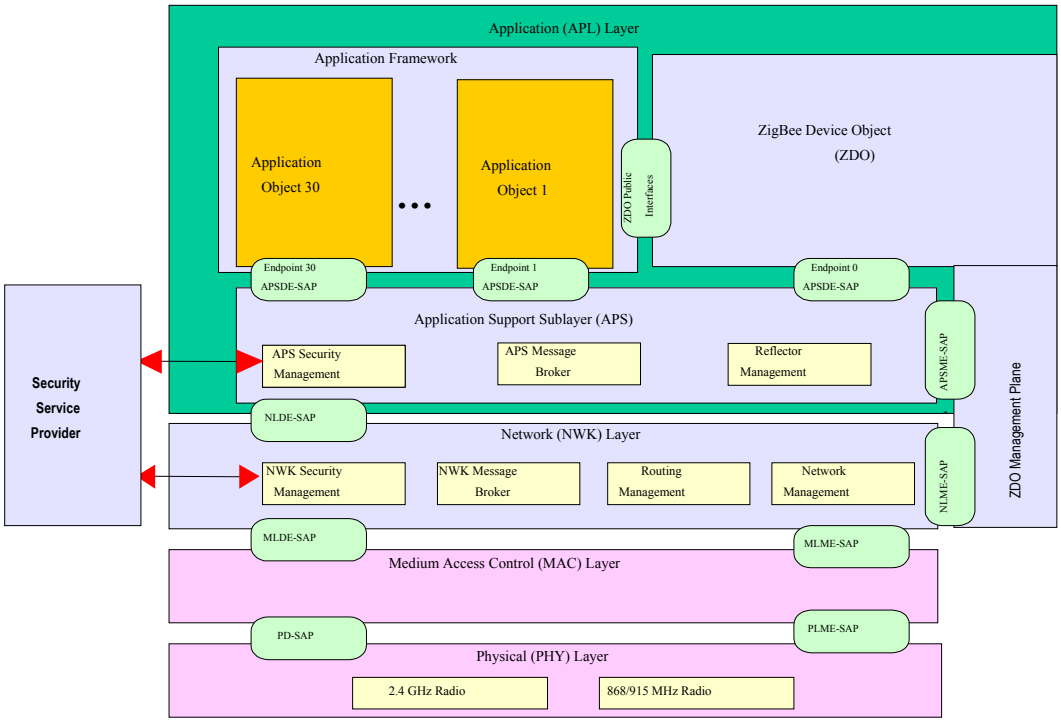


Figure 2.1: ZigBee Security Architecture by ZigBee Aliance

shared with entire network by manufacturer. In order to support node mobility using the unique key, each node has to have keys as many as the number of nodes. Figure 2.2 shows the required keys in Zigbee. Seven keys (three MKs, three LKs, and a NK) for the secure communication in the network should be shared with only four nodes. Thus, deploying Zigbee in the large-scale networks requires quite large storage for the key management.

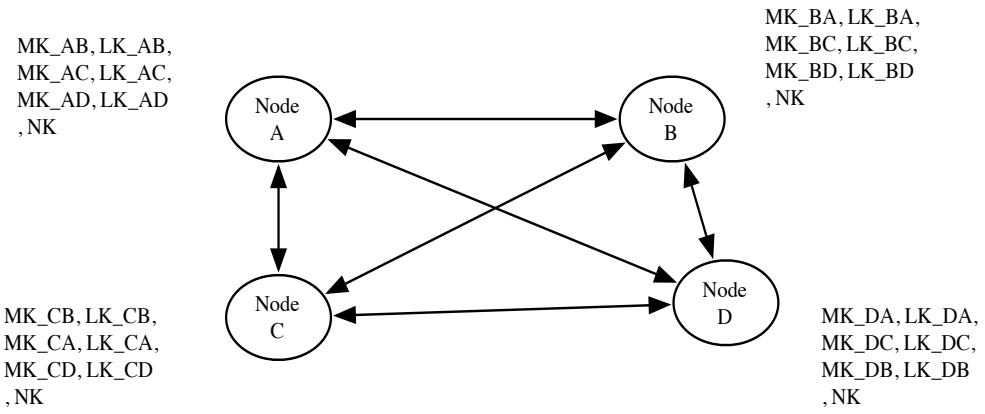


Figure 2.2: Each node has to store seven keys in order to support mobile nodes in the network with four sensor nodes under Zigbee [16].

2.1.2 Authenticated Key Agreement Protocols For Wireless Sensor Network

However, it is hard to assume that the every key is pre-installed in the sensor-attached smart phone, which requires large storage for the keys in the large-scale sensor network. Thus, many active researches such as [3, 13, 24, 25, 33, 35, 47, 84, 85] are continued in order to provide efficient authentication.

In 2002, Eschenauer and Gligor [24] proposed the pairwise key agreement protocols based on the random key pre-distribution that enables sharing the pairwise key from the pre-distributed key pool. In the initial stage, each node stores m keys selected in a key pool. After the nodes are deployed, each node shares the key information to neighbor nodes. When the shared keys are found, the node establishes the secure links between sinks that share the keys. After the link is established, nodes generate the pairwise key with the sink that has no shared information via the secure link. Later, Chan *et al.* [13] improved model that generates the pairwise key from the multiple number of shared key,

and Liu and Ning [47] proposed the model that the pairwise key is not directly distributed but derived by a bivariate polynomial. However, the networks cannot be entirely connected due probabilistic methods. The probability of failure increases in the case of irregular deployment of sensor nodes or unpredictable interruptions.

Zhu *et al.* [84, 85] introduced the group key based key agreement model that minimized threats of compromised nodes. Every node has a unique key, pairwise keys with neighbor nodes, a cluster key shared with all neighbor nodes, and the global key that shared with entire network. However, they assumed the networks are static.

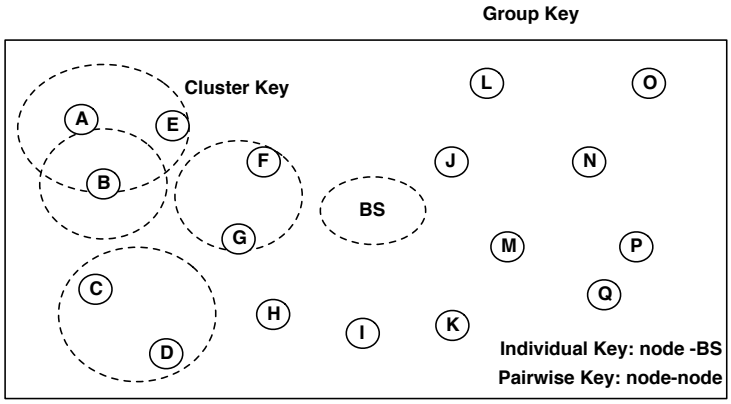


Figure 2.3: Zhu *et al.*'s Model. [84, 85]

Hierarchical Key Establishment Model

In 2006, Abraham and Ramanatha [3] proposed an authentication and initial shared key establishment model in hierarchical clustered networks. In 2006, Ibriq and Mahgoub [35] proposed an efficient hierarchical key establishment model with 'partial key escrow table'. Using the key escrow table, a sink can self-generate the shared key for the attached nodes. Figure 2.4 shows the brief model of [35]. The intermediate Sink 1 stores the partial key escrow table that stores the partial information of nodes. After the requests from nodes are received, Sink 1 requests the authentication ticket to the base station. After receiving the ticket, Sink 1 authenticates and share keys with nodes. However, any sinks have to maintain the information of every node in the table to support the node mobility.

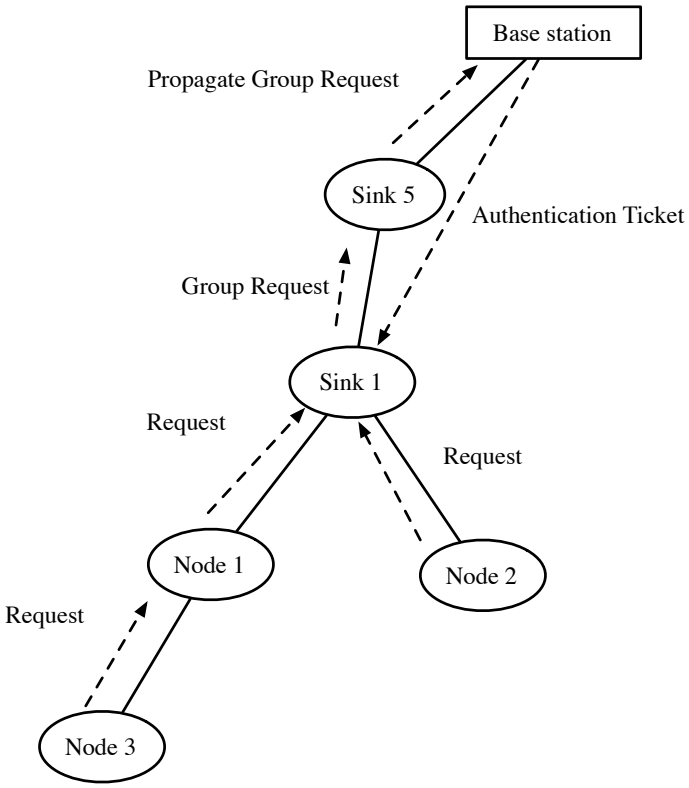


Figure 2.4: Ibriq and Mahgoub' Model [35]:

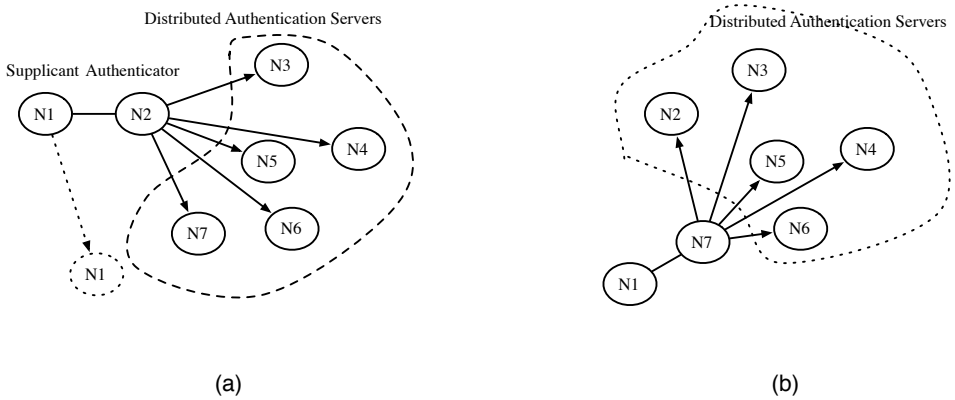


Figure 2.5: Fantacci's Distributed Node Authentication Model (a) Initial authentication by N_2 (b) N_1 reauthenticated by N_7

Distributed Authentication Model

Fantacci *et al.* [25] proposed the distributed node authentication model that does not require the base station as centralized authenticator. Figure 2.5 shows the brief model that there is no centralized authenticator. Every node shares the partial of the authentication information of each node based on Shamir's Secret Sharing Scheme [65] that enables the node mobility support. When a node requests to be authenticated to other node, the node N_2 is authenticator, while other nodes such as N_5 and N_6 are distributed authentication server. However, this model brings the large overhead on each node due to the involvement in the authentication process. Since the node has to participate in the authentication procedures as authenticator or an authentication server, the computational and communication overhead can be significantly increased with frequent authentication requests. Once a node N_1 is authenticated by N_2 as in Figure 2.5 (a), N_1 requests the reauthentication to N_7 as in Figure 2.5 (b). In the figure, N_3, N_4, N_5, and N_6 are involved in both authentication processes as authentication servers.

PKI based Authenticated Key Agreement Model

Although Public key infrastructure (PKI) brings the strong and advanced security services, most studies focused on the symmetric key crypto-system based approach due to the insufficient computational resources for PKI of the sensor nodes. However, many ef-

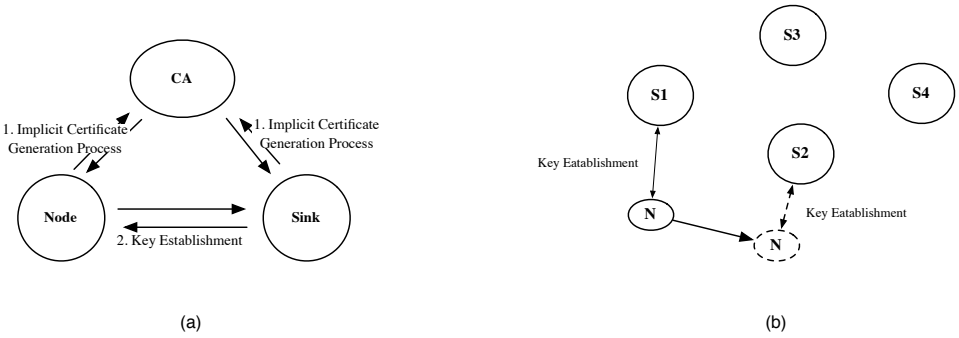


Figure 2.6: (a) Brief processes of Huang's Key Agreement Model (b) Applying Huang's model in Dynamic Sensor Network

forts that enables PKI for sensor networks such as TinyPK [83] and TinyECC [46] are continually proposed.

Huang *et al.* proposed self-organizing algorithm by using Elliptic Curve Cryptography (ECC) [33]. Huang's model has two phases; *Implicit certificate generation process* and *Hybrid key establishment process*. Once the certificates are issued to nodes, nodes can self establish the pairwise key with exchanging the certificates with any sinks. Brief processes are shown in Figure 2.6 (a). Although Huang *et al.* did not address that their protocol could be applicable to the dynamic sensor networks, their protocol can support node reauthentication. After the certificate is issued to the node N, N is authenticated by a sink S_1. When N moves and requests the reauthentication to other sink S_2, S_2 can easily authenticates N again as in Figure 2.6 (b).

However, their model has two critical problems. One is that all sensor nodes must contact CA to obtain their certificates. They require direct contact between each sensor node and CA, and it does not be considered as practical. The other is that every node has to be capable of ECC computation. Even though PKI based applications for the sensor networks will be available in near future with efficient implementation, the public key based security architecture still requires more advanced computational power and resources. A sensor node that is only capable of more lightweight cryptosystem such as AES or SHA-1 may not be able to join to such networks.

2.2 Security Studies in Mobile Network

2.2.1 Brief Overview of 3GPP based Mobile Network

The Universal Mobile Telecommunication System (UMTS) mobile network consists of User Equipment (UE), UMTS Terrestrial Radio Access Network (UTRAN), and Core Network (CN) as in Figure 2.7. UE is the user area that handles all user interfaces, and consists of Mobile Equipment (ME) and USIM that contains the unique identity of user such as phone number and the master key.

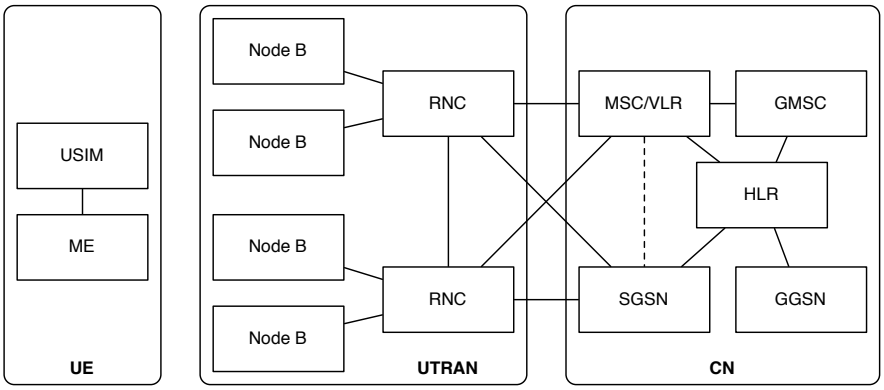


Figure 2.7: UMTS Network Architecture

UTRAN is the service network area that consists of Radio Network Controllers (RNCs) that control mobile accesses and Node B, which handles the functions related to mobile access. CN is the core of the mobile network that is comprised of Home Location Register (HLR), Authentication Center (AuC), Visiting Location Register (VLR), Serving GPRS Support Node (SGSN), Gateway MSC(GMSC), and Gateway GPRS Support Node (GGSN).

HLR is the core of the mobile network that all subscribers are assigned to specific HLRs based on their phone numbers. When someone attempts to call a subscriber, the caller's switch queries an appropriate HLR with a request for the current location of targeted subscriber's. MSCs act as telephony switches and deliver circuit-switched traffic in a GSM based network. MSCs are responsible for performing between base stations, assist in billing operations and can provide a function as gateways to both wired and neighboring mobile networks. With the assistance of a VLR, MSCs can identify and store information about currently associated subscribers. For data connections, these operations are per-

formed by the Serving GPRS Support Node (SGSN). GMSC and GGSN are required to connect to outer networks.

2.2.2 Security Architecture of 3GPP based Mobile Network

In 3GPP based mobile network, the master seed key for the security is stored in USIM which is considered as a temper resistant security hardware module and secret information may be preferred to store in USIM. Widely employed symmetric key based security architectures use authentication and key agreement (AKA) protocol [71] that generates the session keys (ie. the cipher key (CK) and the integrity key (IK)) from the master seed key stored in USIM. In brief AKA process, when ME is visiting the foreign networks, the serving network (SN) requests the authentication information to HLR. Then HLR sends the authentication information to SN, and SN and ME mutually generate CK and IK. Similar processes are shown in Figure 2.13. AKA protocol is also used for supporting third party network application services such as mobile banking and multimedia services, specified in the generic authentication architecture (GAA) [78, 74].

2.2.3 Security Architecture for 3rd-Party Application

Basically, a mobile device consists of two main components. One is a mobile equipment (ME) and the other is a universal subscriber identity module (USIM). ME is the device such as mobile phones, those attaches USIM that contains the unique identity of user such as phone number and the master seed key. Since USIM is considered as a tamper resistant security hardware module and secret information such as user identity and secret key are stored in USIM. Therefore, 3GPP specifies the security architecture that the seed key is securely stored in USIM, and the session keys (the cipher key and the integrity key) are generated from the seed key using the UMTS-AKA algorithm and transferred to ME [71]. Then ME uses the session keys for the secure communication.

3GPP also specifies the generic authentication architecture (GAA) to support the third party network applications such as mobile banking and various multimedia services. Currently, widely employed GAA is the symmetric key based generic bootstrapping architecture (GBA) [74] as in Figure 2.9. The architecture consists of four essential entities such as the Bootstrapping Function (BSF), the network application function (NAF), ME and Home Subscriber Server (HSS). For the third party service, ME can be communicated with NAF that can be used with any specific application protocol necessary. HSS has the initial key shared with USIM and sends the authentication information to BSF. BSFs are located in each domain and send the received authentication information to NAF. GBA

employs the UMTS-AKA algorithm for mutual authentication between the mobile equipment and BSF in the network.

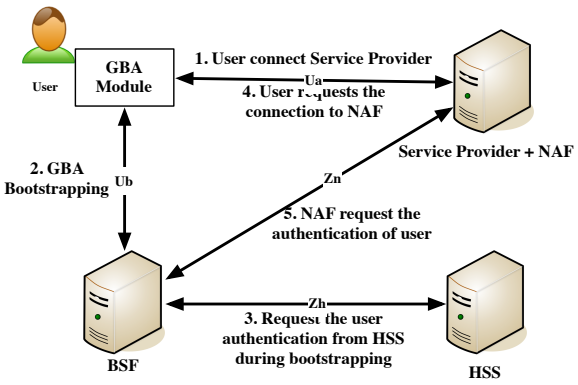


Figure 2.8: 3GPP Generic Bootstrapping Architecture. A subscriber proceed GBA for the service from NAF.

The architecture consists of four essential entities such as BSF, NAF, ME and Home Subscriber Server (HSS). HSS has the initial key shared with USIM and sends the Authentication Vectors (AV) to BSF. BSFs are located in each domain and send the received AV to NAF. NAF provides the third party services to ME. Figure 2.8 shows overview of GBA process. In the figure, Ub, Ua, Zh and Zn denote the reference points. Ub provides mutual authentication between ME and the Bootstrapping Server Function (BSF), and used for ME to bootstrap the session key based on 3GPP AKA infrastructure. Ua carries out the application protocol, which is secured using the keys agreed between BSF and ME. Zh is used for BSF to fetch the required authentication information and all GBA user security settings from the HSS. Zn is used by NAF to fetch the key agreed during a previous HTTP Digest AKA protocol run over Ub from ME to the BSF, and to fetch application-specific user security settings from the BSF. For the asymmetric key based security architecture, 3GPP specifies the *Support for Subscriber Certificate* (SSC) [75] that specifies the role of NAF as the PKI portal issuing the certificates of ME .

2.2.4 PKI Support for Advanced Security Service

The advance of mobile network brings the request for the deployment of PKI that enables the more various security applications such as the digital signature.

Thus, 3GPP also specifies the asymmetric key based security architecture [75] to sup-

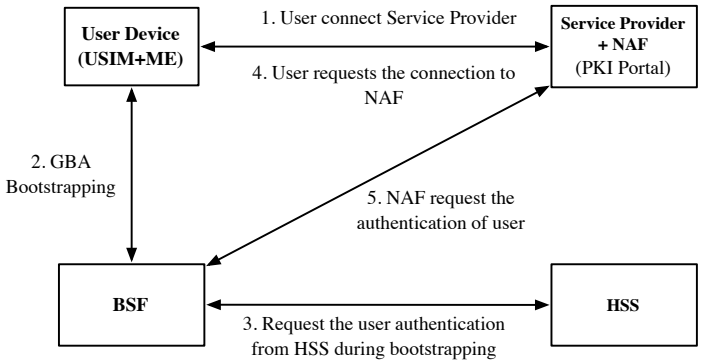


Figure 2.9: The process of Generic Authentication Architecture. 3GPP TS 33.221 specifies that NAF has the role of the PKI portal.

port the certificate service. In the architecture, a NAF acts as the PKI portal that issues the certificates of ME as in Figure 2.9. In order to establish the secure channel between the PKI portal and ME, the BSF should have the shared secret key with NAF and ME. That means that the PKI support in [75] is only available along with GBA (Section 2.2.3).

One of important issues on deploying public key based cryptosystem is key management problem [75]. Only the key owner should know the private key, and the certificate are securely stored. There can be two cases on storing the certificate: one is storing the certificate in the USIM as in Figure 2.10 (a) and the other is storing the certificate in ME as in Figure 2.10 (b). Storing the certificate and computing the security operations in the USIM will be significant overhead to USIM. Instead, storing the certificate and computing the security operations in ME are better for the overall performance. Nevertheless, the storing the certificate in ME increases the potential threats of the leakage of the secret keys.

Although storing the certificate in the USIM provides enhanced security, the USIM has weaker resources and performance. Typical hardware configuration of the USIM is about 5-40 MHz clock speed, a few kilobytes of memories. Also, the communication between USIM and ME is based on ISO 7816-3 based I/O interface (T=0, 1) provides half-duplex communication between USIM and ME with 9600 baud - 115 kbps communication speed. Even the recent developments [54] are prepared to provides the full-duplex I/O interface, multi-application service, flash memory and browser based service, plenties of time are need for such technologies are deployed, and mobile devices will be more powerful at that time. Recent Javacard platform [68] is designed to support the better hardware that has

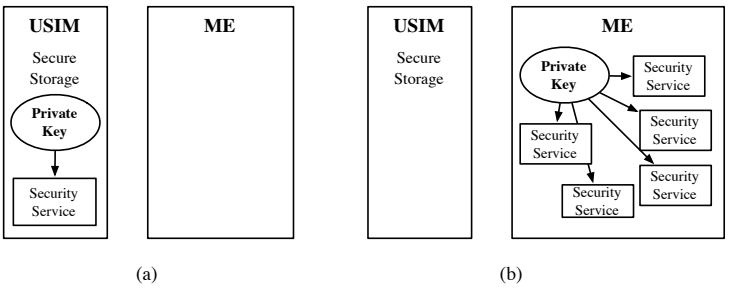


Figure 2.10: (a) Storing public key in USIM gives security strength. (b) Storing public key in ME enables many applications.

a 32-bit processor,128 kilobytes of EEPROM, 512 kilobytes of ROM, and 24 kilobytes of RAM, while the performance of the recent smart phone is equivalent to entry-level mobile computer. In order to compare the performance between USIM and ME, we refer [62] that the computation time for encryption with RSA 1024 is 5-25 ms in USIM with 5-40 MHz clock speed, and 1 ms in Intel Celeron 450 MHz. Nowadays, the performance of smart phones show the better performance than Celeron 450MHz. Thus, we can consider ME significantly outperforms the USIM.

While storing the certificate in ME could overcome such constraints, and it increase the potential security threat of key leakage instead. Alternatively, we can use only short-lived certificates for enrolling subscriber. Even if new user may access the old user's private key, he/she should fail to masquerade as the old user in authorization transactions when the subscriber certificates expired. However, the use of the short-lived certificates requires the more frequent communication between PKI portal and the user for update the certificates. Also, the risk lives until the expiration of the certificates. Key pair generation should protect disclosure/cloning of private key, because the key pair generation is important for the secrecy of the private key.

2.2.5 Interworking between 3GPP and non-3GPP Network

While the several different networks are integrated, 3GPP provides the security architecture and *Extensible Authentication Protocol - Authentication and Key Agreement* (EAP-AKA) [6] for secure interworking between 3GPP and non-3GPP [77] as shown in Fig. 2.11. Followings are required security features for the interworking between 3GPP and non-3GPP networks.

Network access security(I): This security features provide users with secure access to services. Radio access protection is a non-3GPP access specific and outside the scope.

Network domain security(II): This security features can enable nodes to securely exchange signaling data and protection against attacks on the wireline network.

Non-3GPP domain security(III): This security features are a non-3GPP access specific and outside the scope.

Application domain security(IV): This security features can enable applications in user and in provider domain to securely exchange messages.

User domain security(V): This security features can securely access the mobile station.

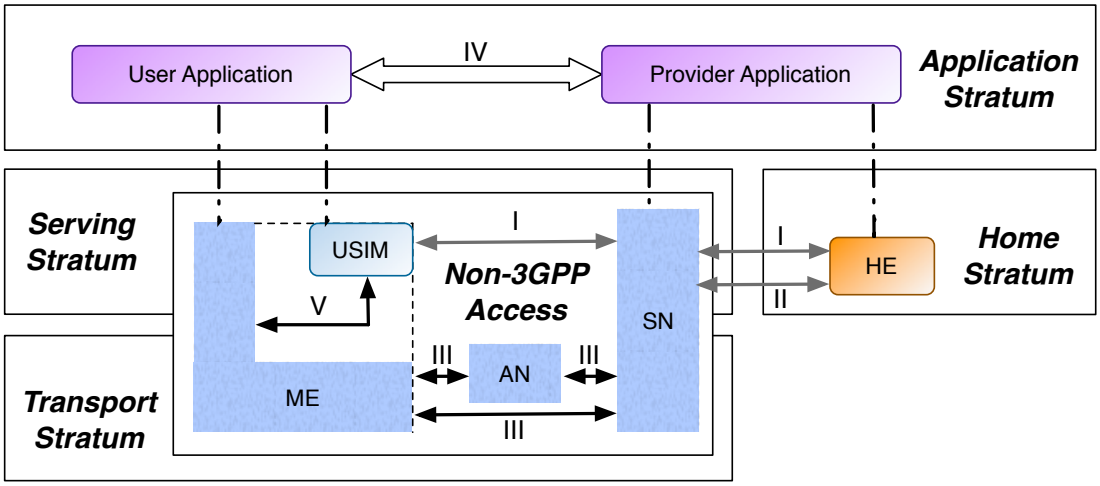


Figure 2.11: Security architecture for accesses to non-3GPP

Fig. 2.12 shows how the SAE/LTE architecture accesses non-3GPP. Refer to Fig. 2.12 non-3GPP consists of trusted non-3GPP such as WiMax and untrusted non-3GPP such as WLAN.

The AAA server performs mutual authentication between 3GPP and non-3GPP as well as accesses HSS through *Wx* interface to get subscriber's information such as IMSI and Authentication Vector (AV). Therefore, the AAA server performs important roles in interworking between 3GPP and non-3GPP interworking. *Ta* interface which was connected with trusted non-3GPP transmits authentication, authorization, and accounting informa-

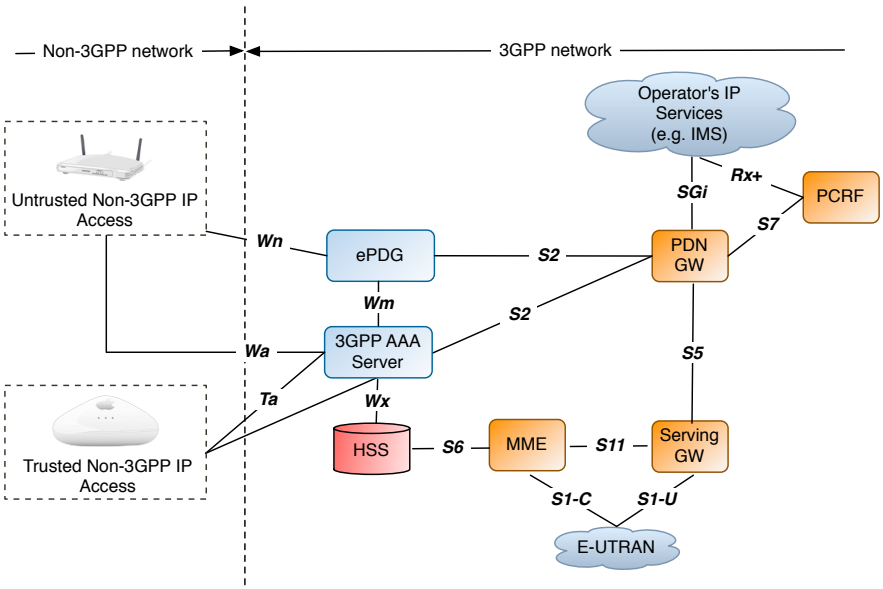


Figure 2.12: Architecture of interworking between 3GPP and non-3GPP

tion to the AAA server. Trusted non-3GPP transmits subscriber's information to PDN GW through *S2* interface.

In order to access untrusted non-3GPP, evolved Packet Data Gateway (ePDU) is added in 3GPP network. All traffics that are generated by untrusted non-3GPP are concentrated on the ePDU. Therefore, the ePDU establishes secure tunnel using IPsec and then securely sends subscriber information. Moreover, *Wm* interface transmits subscriber-related information from AAA server to ePDU [45, 77].

Figure 2.13 shows the procedure of EAP-AKA that provides mutual authentication between the User Equipment (UE) and the Authentication, Authorization, Accounting (AAA) server. Thus, EAP-AKA enables authentication and key agreement procedure between 3GPP and WLAN. Similarly, EAP-SIM [30] is also provided to authenticate a user for WLAN access via the SIM card using GSM networks. Not only such standardization, there are several on-going researches such as [53, 80] enhancing the security of EAP-AKA, and EAP-SIM.

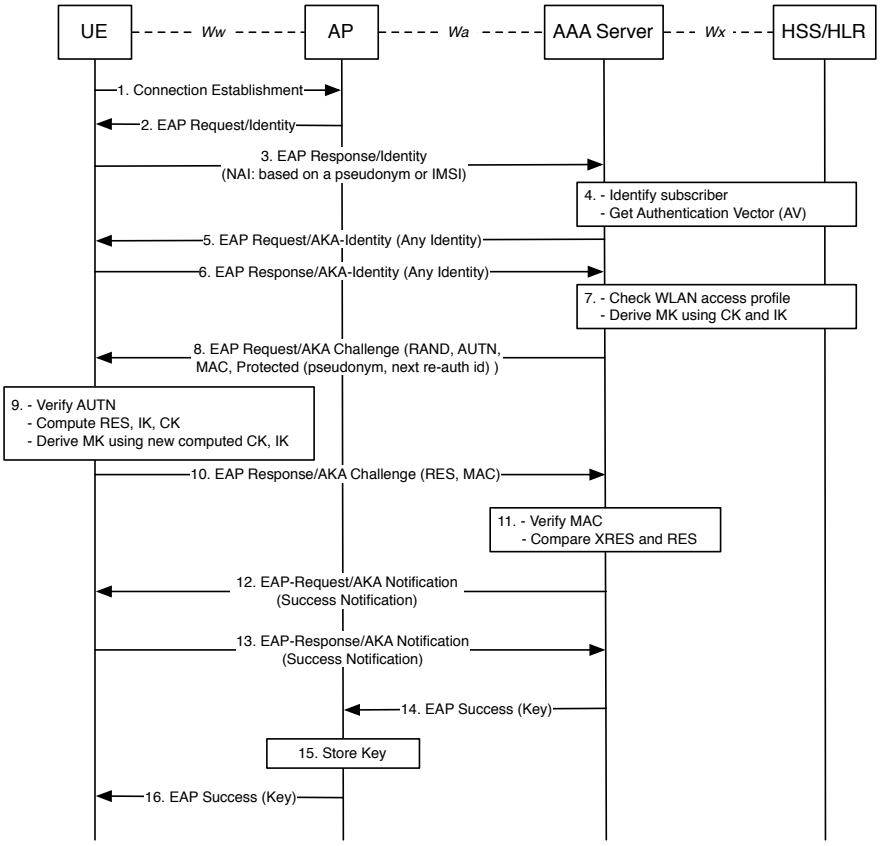


Figure 2.13: Procedures of EAP-AKA

3. Efficient Sensor Node Authentication in Dynamic Wireless Sensor Network

In this chapter, we propose efficient node re-authentication and key exchange protocols that reduce communication and computational overhead for node re-authentication. After claiming the security issues in WSN with mobile nodes, we argue the insufficiency of applying current authentication and key distribution researches to such environments. And then we propose the efficient untraceable re-authentication and key distribution protocol that can reduce the communication overhead between a sink and the base station. Applying our protocol, a node previously authenticated by a sink can be efficiently re-authenticated when the node changed the position with less communication and computational overhead; also the node movement is untraceable.

Chapter 3 organized as follows: Section 3.1 briefly claims drawbacks of previous authentication and key distribution protocols supporting mobility support in WSN and identifies the security requirements. Then, We propose the efficient mobile node re-authentication protocol in Section 3.2, and analyze the performance and security of our protocol in Section 3.3. We then extend our protocol deploying public key cryptosystem for the advanced sensor network environments in Section 3.4 and analyze the protocol in Section 3.5.

3.1 Issues of Mobile Node Authentication in WSN

In this section, we claim the security problems on the node mobility in WSN and limits of previous authentication and key agreement models. At first, we show a sensor network model with mobile nodes as in Figure 3.1. We define a static sensor node as Sink, a mobile node as Node, and the base station that is the core network. The node has linear movements in the network. The base station and sinks are static as same as Ibriq and Mahgoub's model [35]. Sinks act as the gateway that link nodes to the base station, and the base station is a kind of headquarter that manages entire networks. When a node initially joins the network, the node connects to a sink in the network and is authenticated by the sink with help of the base station. After that the node moves and reconnects to other sink. We assume that the sink that re-authenticates the node is the neighbor sink of the sink that previously authenticated the node. The re-authentication processes fre-

quently happen since the node continually moves in the network.

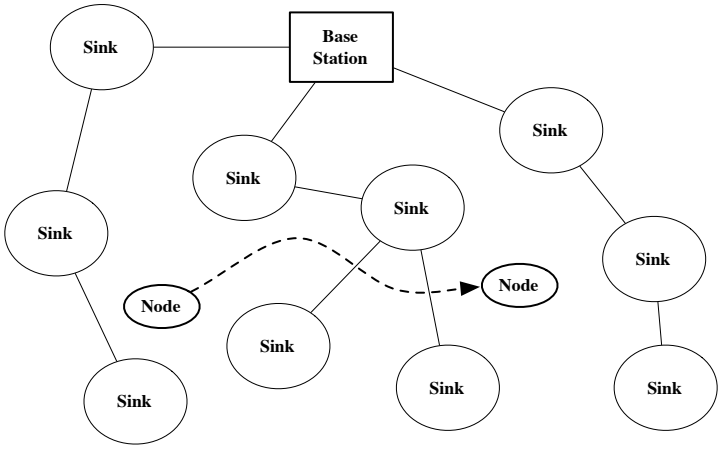


Figure 3.1: System Model of Dynamic Wireless Sensor Network

In practical scenarios, re-authentication happens when a node lost connection to the sink or moved and connected to other sink. For the former case, the node can be easily re-authenticated to the same sink when the connection becomes available again. For the latter case, the node request the re-authentication to other sink that is the near to the previously attached sink.

3.1.1 Drawbacks of Previous Protocols Supporting Mobile Node

Re-authentication

Since the sensor has low powered battery and low-end processor with short-range wireless communication, the reducing communication and computational overheads is important to increasing the lifetime of the sensor. However, the mobile sensor node may occur the large overhead for the security computation due to the frequent requests of node re-authentication. When a node connects to a sink, the sink has to authenticate the node. After that, the node will connect to other sink after the movement; the new sink has to authenticate the node again. If the node has continuous movement, the authentication process will also occur repeatedly. It is obvious that the frequent re-authentication processes are the significant factors that drain the resources in battery-based sensor nodes.

However, the current authentication and key distribution protocols are insufficient to

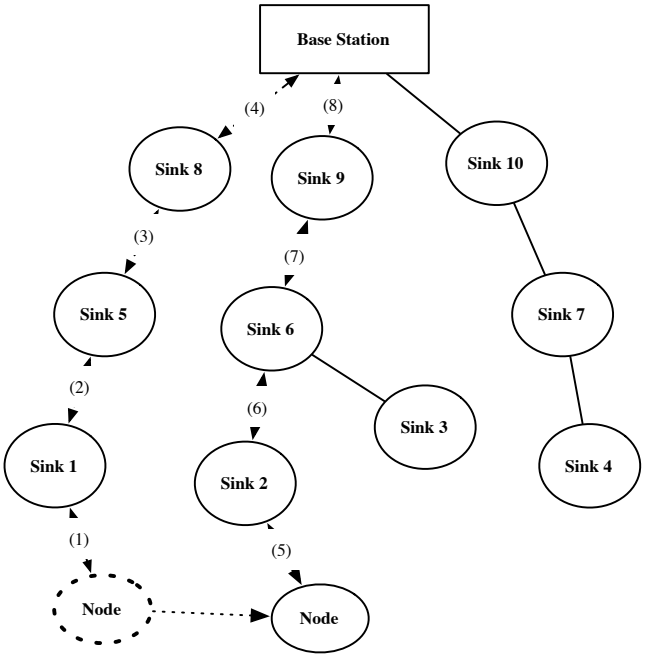


Figure 3.2: Communication Pass: Initial Authentication (1)-(2)-(3)-(4). Re-authentication (5)-(6)-(7)-(8). The unbroken line denotes the static connection, and the dotted line denotes the movement of the node

be applied in such environment with lack of consideration of the node mobility. Using the current protocols such as [35], the communication pass (1)-(2)-(3)-(4) is required for the initial authentication and key distribution in Figure 3.2. When the node moves and reconnects to sink 2, the communication pass (5)-(6)-(7)-(8) is required for authentication and key distribution that have the similar communication overhead to the initial authentication. Such overhead will be the problem in the environment that the frequent movements of the large number of nodes are happened. Thus, the less computational and communication overheads in re-authentication are very urgent requirement for the node mobility support in the WSN.

Tracing Node Movements

Considering the mobility of sensor nodes, the tracking of the node movement is one of possible attacks. For example, when the mobile nodes are deployed in battlefields, the tracking by enemies is significant threats for the networks. Also, tracking node movement

is the threat on the privacy. Thus, the authentication and key agreement protocols should provide the privacy of the mobile node, while current protocols do not consider the mobility of the node.

3.1.2 Security and Privacy Requirements

We define the security requirements as follows. We assume that when the node N communicates with a sink S_2 after disconnection to the sink S_1, S_1 cannot receive any message between N and S_2. S_2 is one of neighbor sinks of S_1.

Re-authentication An authenticated node N and S_2 should be able to identify each other with less communication and computational overhead than initial authentication.

Untraceability In re-authentication of N, S_2 only identifies that N was previously connected to S_1, and never traces the direction of N.

In addition to the requirements of 're-authentication' and 'untraceability', we also define the fundamental security requirements as follows.

Confidentiality When N and S_1 are operating initial authentication, nobody can know the communication packet between N and S_1, between S_1 and BS. For re-authentication between N and S_2, nobody except S_1 can know the communication information, while S_1 out of communication range.

Message Authentication Any malicious adversaries should not be able to forge the communication packet.

Key Freshness N and S should be able to verify that the key is generated during the current session.

Node/Sink Resiliency Even N, S_1 or S_2 are compromised by a malicious adversary, they should not be able to affect to the entire network.

'Confidentiality', 'message authentication', and 'key freshness' are important requirements against the attacks such as the replay attack or man-in-the-middle attack. 'Node/Sink resiliency' is practical threat that the sensor nodes are generally deployed in the environment where the administration is unavailable.

3.2 Protocol 1: Untraceable Mobile Node Re-authentication Scheme

In this section, we propose our novel authentication and key distribution scheme that provides efficient mobile node re-authentication and untraceablity. In section 3.2.1, we briefly overview the overall process of proposed protocol. In section 3.2.2, we introduce the concept of 'authentication ticket' that enables the fast re-authentication. After that, we show our efficient node re-authentication protocol in following section 3.2.3.

3.2.1 Overview of Proposed Protocol

We briefly describe the procedure of our proposed protocol as in Figure 3.3. Assume that there are a base station BS, a sink S_1, a neighbor sink S_2, and a mobile node N in the network. We define the neighbor sink as the sink that is in the 1 hop communication range. S_1 periodically broadcasts HELLO in Phase 0. When S_2 receives HELLO, S_2 initiates the neighbor relationship if S_1 is a newly discovered sink. After the pairwise key between S_1 and S_2 has been exchanged in Phase 1, S_1 and S_2 exchange the authentication key that is used to verify the authenticated user in Phase 2. Phase 1 and Phase 2 are only required during establishing the static sensor network. Establishing the static sensor network can follow the any previous protocols such as [35].

When N firstly joins the network, N may be connected to S_1 in the network as in Figure 3.3. After receiving HELLO of S_1, N initiates the initial authentication with S_1 in Phase 3. After N is authenticated S_1, N only needs the re-authentication in Phase 4 when N continually moves and request the authentication again. The authentication process in Phase 3 is only necessary when the re-authentication fails due to the certain case that the neighbor sink is not available.

3.2.2 Authentication Ticket

We define 'Authentication Ticket' that is used for the node re-authentication. When a node requests authentication to a sink, the sink generates the authentication ticket and sends it to the node. The authentication ticket is verified by the authentication key that is given to neighbor sinks. Using the authentication ticket, the node movement is untraceable. Verification of the authentication ticket is available to neighbor sinks of the sink that issued the ticket. We adopt the idea of 'cluster key' in [85] that shared to neighbor sinks. The main difference is that the cluster key in [85] is used for broadcast communication in

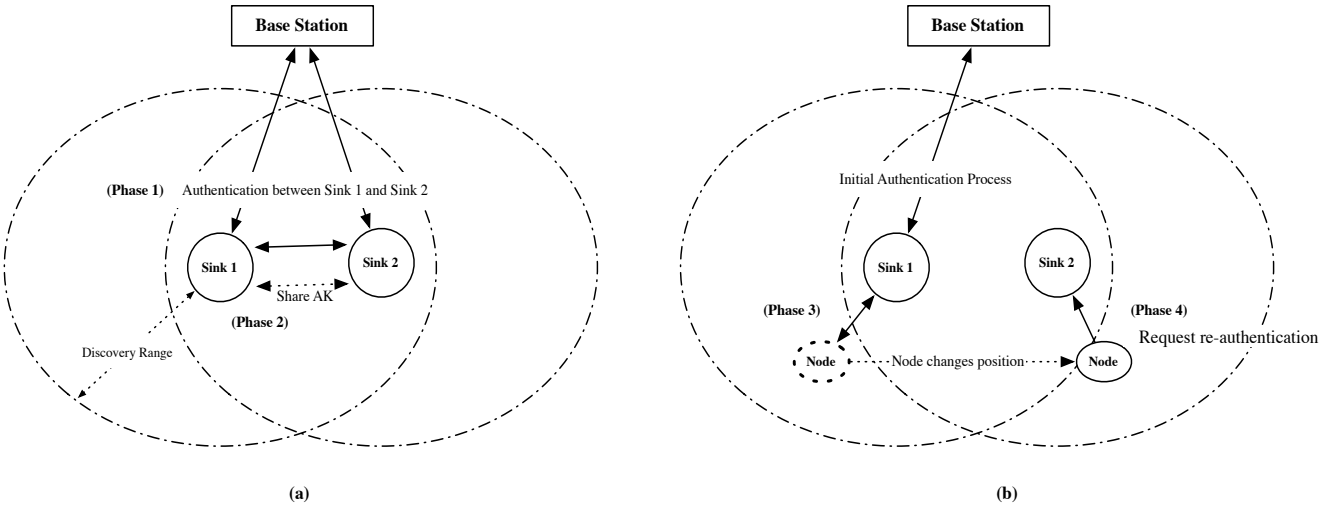


Figure 3.3: Protocol Overview: In receiving HELLO of Sink 2 (S_2), (a) Sink 1 (S_1) mutually authenticate Sink 2 (Phase 1), and share the authentication key (Phase 2). (b) Node is initially authenticated by Sink 1 (Phase 3), and requests re-authentication to Sink 2.

the cluster, while the key in our protocol is used for verifying the authentication ticket. Thus, we rename the key as 'authentication key' due to the different use in the protocol. Figure 3.4 shows that neighbor sinks of Sink 1 (S_1) shares the authentication key AK_{S_1}.

3.2.3 Protocol Description

The protocol consists of five phases as follows:

Phase 0 The common neighbor discovery

Phase 1 Neighbor sink relationship set up,

Phase 2 Neighbor group authentication key share,

Phase 3 Initial node authentication, and

Phase 4 Node re-authentication.

The notations used in the protocol are defined in Table 3.1. Key IK_N is the integrity key derived from K_N, where $IK_N = KDF(K_N)$. KDF is an one-way key derivation function. We can also use a hash function for KDF.

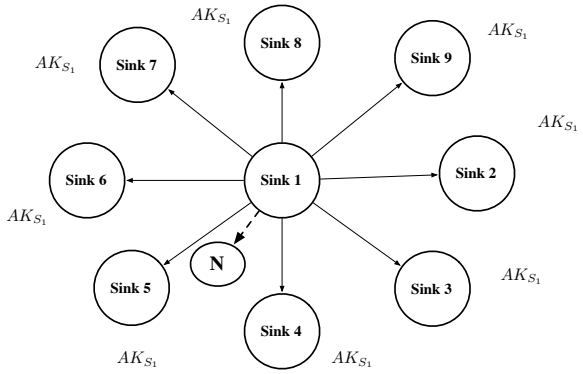


Figure 3.4: Sink 1 shares AK_{S_1} to neighbor sinks. When N is authenticated by Sink 1, any neighbor sinks can re-authenticate N.

Table 3.1: Notations

Term	Description	Term	Description
BS	Base Station	$E_t\{m\}$	Encrypt arbitrary message m using t
$h\{m\}$	Hash arbitrary message m	$MAC_t(m)$	Message Authentication Code using t
TS	Time stamp	K_N	Pre-shared key between N and BS
IK_N	IK derived from K_N	K_S	Pre-shared key between S and BS
IK_S	IK derived from K_S	SK	Shared session key between sinks
SIK	IK derived from SK	AK_S	Group Authentication Key of Sink
AIK_S	IK derived from AK_S	NK	Shared session key between S and N
NIK	IK derived from NK	IK	Integrity Key

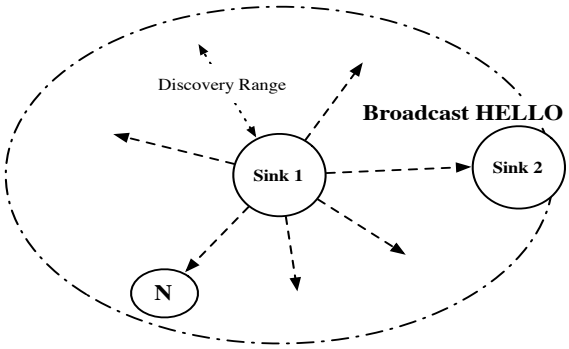


Figure 3.5: Neighbor Discovery (Phase 0): Sink periodically broadcast HELLO.

Phase 0: Neighbor Discovery

A sink S_1 periodically generates a random nonce R_0 . S_1 also generates $u_0 = E_{K_{S_1}}\{R_0||TS_0\}$ and $v_0 = MAC_{IK_{S_1}}(S_1||\text{HELLO}||u_0)$, where TS_0 is time stamp. u_0 and v_0 are included in the HELLO message as in Figure 3.5. Then S_1 broadcasts u_0 and v_0 as follows:

$$S_1 \rightarrow Broadcast : S_1||HELLO||u_0||v_0 \tag{3.1}$$

Phase 0 is the periodical common procedure. When a sink receives HELLO, the sink initiates Phase 1 or Phase 2. When a node receives HELLO, the node initiates Phase 3 or Phase 4.

Phase 1: Neighbor Sink Relationship Set Up

Assume another sink S_2 receives HELLO message. S_2 checks the sender of HELLO whether S_1 is known or not. If S_2 already knows S_1, S_2 discards the message. Otherwise, S_2 requests the setting up the neighbor relationship as follows:

P-1.a. S_2 randomly selects R_1 and generates $u_1 = E_{K_{S_2}}\{R_1||u_0\}$, $v_1 = MAC_{IK_{S_2}}(S_2||BS||S_1||u_1||v_0)$.

$$S_2 \rightarrow BS : S_2||BS||S_1||u_1||v_1||v_0 \tag{3.2}$$

P-1.b. After verifying v_1, BS decrypts u_1 and retrieves R_1 and u_0. Then, BS verifies v_0 and decrypts u_0. Finally, BS retrieves R_0 and TS_0 . BS generates and sends

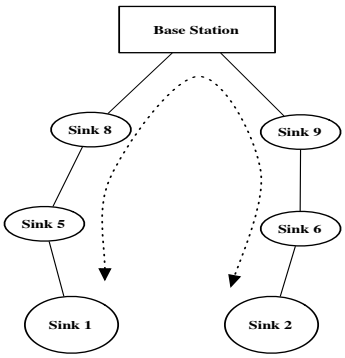


Figure 3.6: Setting up Neighbor Sink Relationship (Phase 1): Sink 1 and Sink 2 share the pairwise key.

u_4, v_4, and v_3 to S_2 where,$u_3 = E_{K_{S_1}}\{R_1||h(TS_0)\}$, $v_3 = MAC_{IK_{S1}}(BS||S_1||u_3)$, $u_4 = E_{K_2}\{R_1||u_3\}$ and $v_4 = MAC_{IK_2}(BS||S_2||R_1||u_4||v_3)$

$$BS \rightarrow S_2 : BS||S_2||S_1||u_4||v_4||v_3 \tag{3.3}$$

P-1.c. After verifying v_4, S_2 decrypts u_4, and retrieves R_1 and u_3. S_2 generates $K_{S_1S_2} = KDF(0||R_0||R_1)$ and $IK_{S_1S_2} = KDF(1||R_0||R_1)$ with R_0 and R_1. $K_{S_1S_2}$ is encryption key and $IK_{S_1S_2}$ is integrity key between S_1 and S_2. Then S_2 generates $v_5 = MAC_{IK_{S_1S_2}}(S_2||S_1||R_0||R_1)$ and sends u_3, v_3, and v_5 to S_1.

$$S_2 \rightarrow S_1 : S_2||S_1||u_3||v_3||v_5 \tag{3.4}$$

P-1.d. After verifying v_3, S_1 decrypts u_3 and retrieves R_1. S_1 also generates $K_{S_1S_2}$ and $IK_{S_1S_2}$. Then S_1 verifies v_5. S_1 generates $v_6 = MAC_{IK_{S_1S_2}}(S_1||S_2||ACK||R_0||R_1)$ and sends v_6 with ACK to S_2.

$$S_1 \rightarrow S_2 : S_1||S_2||ACK||v_6 \tag{3.5}$$

P-1.e. S_2 verifies v_6 and shares pairwise keys $K_{S_1S_2}$ and $IK_{S_1S_2}$.

Phase 2: Neighbor Group Authentication Key Share

Phase 2 can be operated solely or after Phase 1 is completed. In Phase 2, S_1 initiates following procedures.

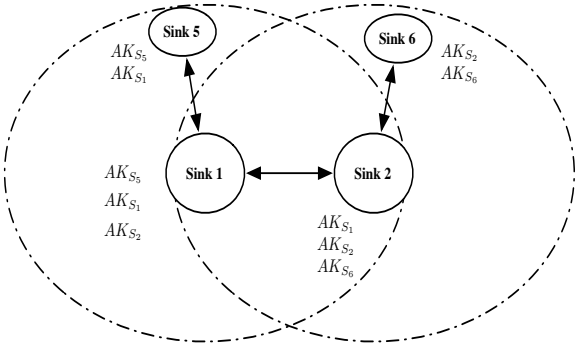


Figure 3.7: Neighbor Group Authentication Key Share (Phase 2): Sinks share neighbor sink's authentication keys.

P-2.a. S_1 randomly selects two nonces $ASEED_{S_1}$ and R_1. Then S_1 generates $u_1 = E_{K_{S_1S_2}}\{ASEED_{S_1}||R_1\}$ and $v_1 = MAC_{IK_{S_1S_2}}(S_1||S_2||u_1)$.

$$S_1 \rightarrow S_2 : S_1||S_2||u_1||v_1 \tag{3.6}$$

P-2.b. After verifying v_1, S_2 decrypts u_1, and retrieves $ASEED_{S_1}$ and R_1. Then S_2 generates $AK_{S_1} = KDF(0||ASEED_{S_1})$ and $AIK_{S_1} = KDF(1||ASEED_{S_1})$. S_2 also generates $v_2 = MAC_{AIK_{S_1}}(S_2\ ||S_1||ACK||AR_1)$ using AIK_{S_1}.

$$S_2 \rightarrow S_1 : S_2||S_1||ACK||v_2 \tag{3.7}$$

P-2.c. S_1 verifies v_2.

After the Phase 2 is completed, sinks share their neighbor sink's authentication keys as in Figure 3.7.

Phase 3: Initial Node Authentication

When N receives HELLO that S_1 broadcasted in Phase 0 and is not yet authenticated by any sink, N proceeds followings.

P-3.a. Node N randomly selects R_1 and generates u_1 and v_1, where $u_1 = E_{K_N}\{R_1||u_0||v_0\}$ and $v_1 = MAC_{IK_N}(N_1||S_1||u_1)$.

$$N \rightarrow S_1 : N||S_1||u_1||v_1 \tag{3.8}$$

P-3.b. S_1 generates $v_2 = MAC_{IK_{S_1}}(S_1||BS||N||u_1||v_1)$.

$$S_1 \rightarrow BS : S_1||BS||N||u_1||v_1||v_2 \tag{3.9}$$

P-3.c. After verifying v_2 and v_1, BS decrypts u_1, and retrieves R_0, u_0 and v_0. After verifying v_0, BS decrypts u_0, and retrieves R_0 and TS. BS checks the validity of TS and generates $u_3 = E_{K_N}\{R_0\}$, $v_3 = MAC_{IK_N}(BS||N||S_1||u_3)$, $u_4 = E_{K_{S_1}}\{R_1||u_3||v_3\}$ and $v_4 = MAC_{IK_{S_1}}(BS||S_1||N||R_0||u_4)$.

$$BS \rightarrow S_1 : BS||S_1||N||u_4||v_4 \tag{3.10}$$

P-3.d. After verifying v_4, S_1 decrypts u_4, and retrieves R_1, u_3 and v_3. Then S_1 generates $NK_N = KDF(R_0||R_1)$. S_1 generates $t = E_{AK_{S_1}}\{TS||R_1||NK_N\}$ and $w = MAC_{AIK_{S_1}}(N||t)$. Next, S_1 also generates u_5 and v_5, where $u_5 = E_{NK_N}\{TS||t||w\}$ and $v_5 = MAC_{NIK_N}(S_1||N||R_0||u_5)$.

$$S_1 \rightarrow N : S_1||N||u_3||v_3||u_5||v_5 \tag{3.11}$$

P-3.e. After verifying v_3, N decrypts u_3 and retrieves R_0. Then N also generates NK_N and verifies v_5. N decrypts u_5 and retrieves TS, t and w. N generates v_6, where $v_6 = MAC_{NK_N}(N||S_1||ACK||R_0||R_1)$.

$$N \rightarrow S_1 : N||S_1||ACK||v_6 \tag{3.12}$$

P-3.f. S_1 verifies v_6.

Phase 4: Node Re-authentication

When previously authenticated N receives HELLO that S_2 broadcasted in Phase 0, N proceeds followings.

P-4.a. N generates $v_1 = MAC_{NIK_N}(N||S_2||t||w||v_0)$.

$$N \rightarrow S_2 : N||S_2||t||w||v_1 \tag{3.13}$$

P-4.b. S_2 verifies w and decrypts t. S_2 retrieves R_1, NK_N and TS. Using NK_N, S_2 verifies v_1. Then S_2 generates $NK' = KDF(R_1||R_0)$, also generates $t' = E_{AK_{S_2}}\{R_1||NK'_N\}$ and $w' = MAC_{AIK_{S_2}}(N||t')$. S_2 generates v_2, u_3 and v_3, where $v_2 = h(NK'_N||R_0)$, $u_3 = E_{NK_N}\{R_0||v_2||t'||w'\}$ and $v_3 = MAC_{NIK_N}(S_2||N||u_3)$.

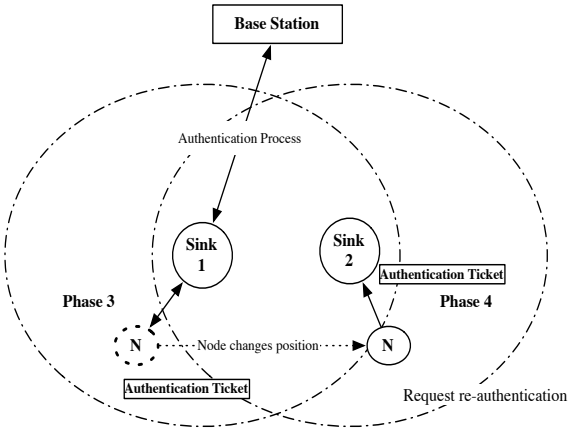


Figure 3.8: Phase 3: Node requests initial authentication to Sink 1. Phase 4: Node requests re-authentication to Sink 2

$$S_2 \rightarrow N : S_2||N||u_3||v_3 \tag{3.14}$$

P-4.c. After verifying v_3, N decrypts u_3 and retrieves R_0, v_2 , t' and w'. Then N generates NK'_N and verifies v_2. N generates $v_4 = MAC_{NIK'_N}(N||S_2||ACK||R_0||R_1)$.

$$N \rightarrow S_2 : N||S_2||ACK||v_3 \tag{3.15}$$

P-4.d. After verifying v_4, S_2 authenticates N.

Brief procedures of Phase 3 and Phase 4 are shown in Figure 3.8.

3.3 Analysis of Protocol 1

In this section, we show the performance and security analysis of our protocol. Section 3.3.1 shows the security analysis for the requirements and known attacks in WSN and Section 3.3.2 shows the comparison to the previous protocols.

3.3.1 Security Analysis

We show the security analysis of our protocol that holds the requirements defined in Section 3.1.2: 'Re-authentication', 'Untraceability', 'Confidentiality', 'Message Integrity', 'Key Freshness', and 'Node/Sink Resiliency'. Then, we analyze the security of our protocol against known attacks.

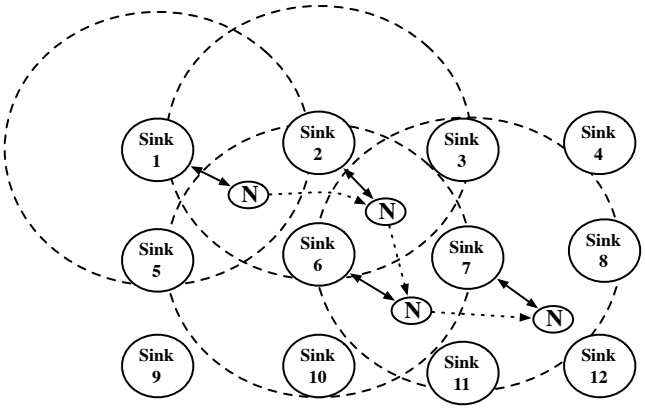


Figure 3.9: When N move in the networks, sinks re-authenticate N without knowing the node's direction

Re-Authentication

After a node N is initially authenticated by a sink S_1 in phase 3, the node receives the authentication ticket (t, w) and v_1, where $t = E_{AK_{S_1}}\{TS||R_1||NK_N\}$, $w = MAC_{AIK_{S_1}}(N||t)$ and $v_1 = MAC_{NIK_N}(N||S_2||t||w||v_0)$. When N moves and requests re-authentication to the neighbor sink S_2, S_2 can verifies (t, w) since the authentication key of S_1, AK_{S_1} is shared to S_2. N can authenticates S_2 with u_3 and v_3 with NK_N. Finally, S_2 authenticates N after verification of v_4. In the re-authentication phase, the base station is not involved.

Untraceability

A sink S_1 issues the authentication ticket (t, w) to a node N. However, S_1 does not know the next move of N. N can be re-authenticated by any neighbor sinks of S_1. For the re-authenticated sink S_2, S_2 only knows that N was previously authenticated by S_1, but never knows the direction N ahead. Sinks only know N was previously authenticated by neighbor sinks, but never predict N's next direction as in Figure 3.9.

Confidentiality

Any sinks and nodes pre-share secret keys only with the base station. For the Neighbor discovery phase, the neighbor discovery message is encrypted using K_S that is only shared between a sink and the base station. For setting up the neighbor group and node authentication, adversary requires shared secret key to know the information. For the node re-authentication, the responses u_3 and v_3 are encrypted using NK_N that is known to S_1. However, we assume that the re-authentication happens, where S_1 cannot involve in the communication from out-of-reach.

Message Authentication

In our protocol, every packet is protected by 4 bytes MAC. The outside adversary should be able to forge the message to success the attack. The security of the MAC depends on the security of the hash function. 4 bytes of MAC size is recommended for practical application in [38], since only 40 forgery attempts per second available on 19.2kb/s channel while 2^{31} trial requires for successful forgery. However, the performance of communication channel is increasing, the size of MAC should be increased in future application. Recently the efficient implementation of hash functions is introduced in [44]. Thus, our protocol is secure against the Man-in-the-Middle attack while the adversary has no efficient way to forge MAC even when the part of the network is compromised by the attacker.

Key Freshness

In Phase 0, the sink S_1 periodically generates random nonce R_0. Thus, S_1 can verify that the requests of authentication are from the directly linked sinks or nodes. In Phase 1, two entities generates the random nonces that both entities can check the freshness. In Phase 2, S_1 also generates random nonce R_1 for the freshness check. In Phase 3 and 4, the node also generates random nonce R_1 to check the freshness.

Node/Sink Resiliency

We can define two kinds of threat of sink capture: the sink missing case and the compromised sink case. When a sink S_1 is just missing, the node will lose the connection S_1 and find other sink such as S_2. Thus, we only need to consider the compromised sink case.

When the sink is compromised, we can assume that the keys in the sink is leaked. However, even the group authentication key is leaked, the effect is only to the neighbor sinks. The compromised sink can self-attach the fake nodes that will request re-authenti

cation without initial authentication. For this case, we add $h(K_N||R_1)$ in the authentication ticket that is sent to the sink when the node requests re-authentication. For suspected node, the sink can check if the node is genuine with help of the base station. Also, we need to define the security policy for the extreme abnormality in deploying sensor network application When the node is compromised, we can define that the compromised node may try to know the information of the sinks or impersonate other nodes. However, the compromised node fail in both cases, since the node does not share any information in the protocol. Thus, our protocol can be practically secure against the security against selective forwarding and acknowledgement spoofing with the node and sink resiliency.

Security against known attacks

We analyze the security of our protocol against the attacks identified in [39]. Since the static parts in the networks could follow the previous models such as [35], we only focus on the security of node re-authentication in this section.

The sinkhole attack against our protocol fails without knowing the keys. An adversary A may capture the authentication ticket (t, w) that N initially sent to S_2, and A send (t, w) to S_2 or other sink S_5 that is also a neighbor sink of S_1. However, A fails in such attack without knowing AK_{S_1}. Wormhole attack on our protocol fails since the adversary cannot send the confirmation message. Spoofed, altered or replayed routing information attack also fail with our knowing encrypted nonce in our protocol. To succeed in the replay attack, the adversary have to be able to re-use the intercepted packet. We don't consider relaying through the attackers as successful attack. Sybil attacks also fails from verification of identity of nodes through sinks and the base station. And for HELLO flood attacks, we can apply the global key shared to all entities in the network that many researches such as [35] and [85] used for the efficient message broadcast and DoS attack protection.

3.3.2 Performance Analysis

For the performance analysis, we compared the number of communication passes, the required message sizes, and the number of computation of the protocol. We do not count the overhead in Phase 0, since Phase 0 does not initiate the protocol. The node just ignores Phase 0 when the node receives HELLO from the sink that already authenticated the node.

Table 3.2: Comparison of Required Communication Pass for Re-authentication

	Fantacci *et al.*'s [25]	Ibriq and Mahgoub's [35]	Proposed
Node	2	$2n$	$2n$
Sink	$2t+1$	$2t$	1
Base station	–	2	–

Communication Pass

We compared the required number of communication passes with Fantacci *et al.*'s model [25], and Ibriq and Mahgoub's model [35]. The reason is that [25] considers the node mobility is considered in that does not require sinks and the base station, and [35] shows the efficient key distribution in static networks. Table 3.2 shows the comparison of communication passes for node re-authentication, where n denotes the number of nodes and t denotes the number of sinks. Since nodes act as the authentication server (the base station) and the authenticator (the sink), all the communications in [25] are operated among nodes.

Comparison of required number of communication pass in initial authentication is as same as the previous models. In re-authenticaion of the nodes, we show our novel protocol has much efficiency in re-authentication than any other protocols compared with [25] and [35]. Since our protocol does not require the communication with the base station in re-authentication.

In practical application, we can deploy the network that all nodes directly connect to any sinks (i.e. $n = 1$). In that case, the communication passes in our protocol are just three passes (*challenge-response-confirmation*).

Message Size

We compared Abraham and Ramanatha's model [3], and [35] for the required message size for authentication. Based on the results in [3], we approximately compared the message sizes based on the message size with MAC size as 4 bytes, the time stamp as 8 bytes, nonce as 8 bytes, and key size as 16 bytes. We also set the source and target IDs as 1 byte, respectively.

Table 3.3 and Table 3.4 show the message sizes in initial authentication and the message sizes in re-authentication with 2 hops between sink and base station, respectively. Table 3.3 shows that the performance for the initial authentication is similar to other pro-

Table 3.3: Comparison of Required Message Size for Initial Authentication (Bytes)

	Abraham's [3]	Ibriq and Mahgoub's [35]	Proposed
Node to Sink	46	68	56
Sink to Sink	70	76	62
Sink to Base station	70	76	66
Base station to Node	92	188	180
Total message size	278	408	302

Table 3.4: Comparison of Required Message Size for Re-authentication (Bytes)

	Abraham's [3]	Ibriq and Mahgoub's [35]	Proposed
Node to Sink	46	68	44
Sink to Sink	70	76	-
Sink to Base station	70	76	-
Base station to Node	92	188	64
Total message size	278	408	108

tocols. In initial authentication (Phase 3), Abraham and Ramanatha's model [3] showed the best result that 30 bytes less message sizes than our protocol. However, As the Table 3.4 shows, our protocol achieves about a third overall message sizes than other protocols. Even we increase the size of each parameter, our protocol is still much efficient than any other protocols in node re-authentication.

For the comparison in multi hop environments, Figure 3.10 and Figure 3.11 show the message size of initial authentication (Phase 3) and re-authentication (Phase 4) in our protocol and the comparison between other protocols, respectively. When the hop distances between the sinks that the node is attached and the base station increases, the required message size and the communication pass also increase.

Computation

Now, we compare the computational overhead of initial authentication (Phase 3) and re-authentication (Phase 4). In total, 10 times of encryption/decryption and 14 times of MAC generation/verification are required for initial authentication, while 4 times of encryption/decryption and 10 times of MAC generation/verification are required for re-auth entication. For node specific operation, 3 times of encryption/decryption for initial au-

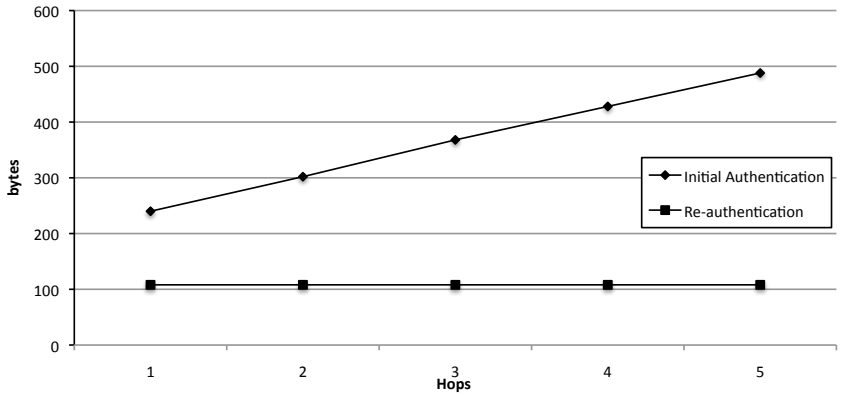


Figure 3.10: Comparison of message sizes with initial authentication and re-authentication per hop distance from sink to the base station increases

Table 3.5: Comparison of Computation between Initial Authentication and Re-authentication (times)

	Initial Authentication	Re-authentication.
Encryption/Decryption in Total	10	4
Encryption/Decryption by Node	3	1
MAC Generation/Verification in Total	14	10
MAC Generation/Verfication by Node	4	4

thentication, 1 time of encryption/decryption are required. Both cases require 4 times of MAC generation/verification. Since the computation of MAC does not have significant overhead, comparing the computation of encryption and decryption, we can achieve 2 - 3 times efficient computation. The comparison of computation is shown in Table 3.5. We do not measure the computation time of each operation that depends on the encryption and hash algorithms in this chapter. Note that we can apply TinySEC [38] and TinyHash [44] for the implementation.

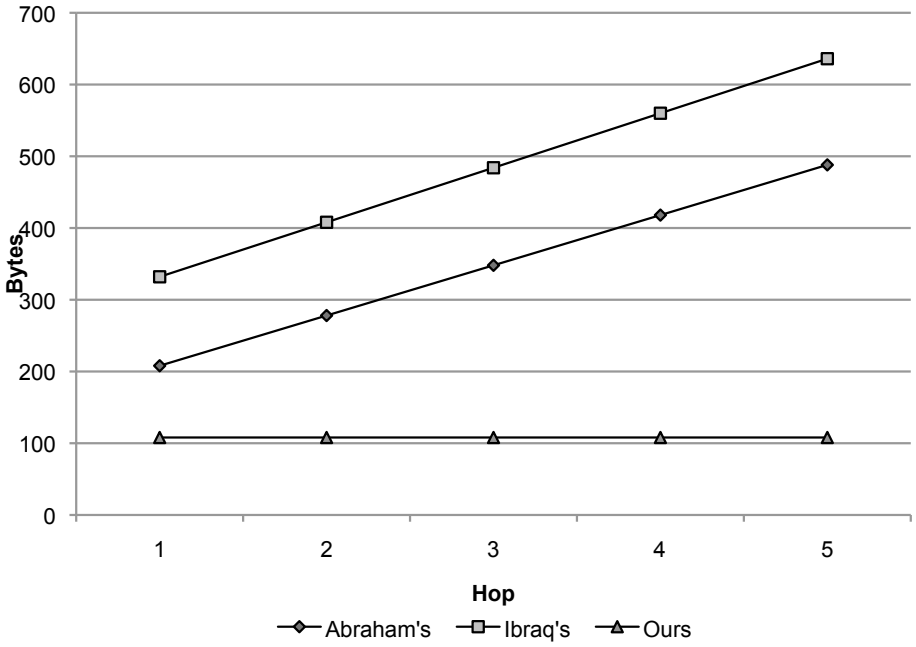


Figure 3.11: Comparison of message sizes with [3] and [35] per hop distance between a sink and a base station

3.4 Protocol 2: PKI based Mobile Node authentication

In this section, we propose the protocol that combines symmetric cryptosystem and public key cryptosystem for the less overhead. Our protocol enables the reduced authentication process for the mobile node that is once authenticated in the network. We also introduce the concept of '*Neighbor Sink List*' (NSL) that enables the deployment of our protocol in real environments. Using NSL, each sink share the neighbor sink information easily, and re-authenticate the mobile node that is once authenticated in the network.

3.4.1 System Overview

We show the dynamic WSN that consists of mobile sensor node N, a static sink node S, and the base station BS as previously shown in Figure 1.1 (b). N continually moves in the network and requests the authentication, whenever it reconnects to a sink S. BS shares the secret keys with N and S, individually. We assume that the sink S has more computational power with better resources than the mobile node N.

Table 3.6: Notations used in the protocol

Notaition	Description	Notation	Description
pk_n	Public Key of n	TS	Timestamp
sk_n	Private Key of n	$\|\|$	Concatenation
e_{pk_n}	Public key encryption using pk_n	h	Hash Function
MAC	Message Authentication Function	$cert$	Certificate
$sign_n$	Digital Signature signed by n	IK	Integrity key

The initial configuration of the static networks can follow the previous design such as Ibriq's protocol [35]. We also provide the procedure for a sink to find the neighbor sinks in section 3.4.2. Every sink nodes S_i, where $i = 0, 1, ...t$ and t is the number of sink in the network, periodically sends HELLO as in section 3.4.2. When a mobile sensor node N firstly joins to the network, N proceed the initial authentication as in section 3.4.2. Once N is authenticated by any S_i, N only operates the reduced procedures as in section 3.4.3 when N reconnects to other sink S_j. Figure 3.12 shows the comparison between initial authentication and reauthentication. When N is initially authenticated by S_1, the communication pass is $N - S_1 - S_3 - S_4 - BS$. On the other hand, when N requests reauthentication to S_2, the communication pass is only $N - S_2 - S_1$.

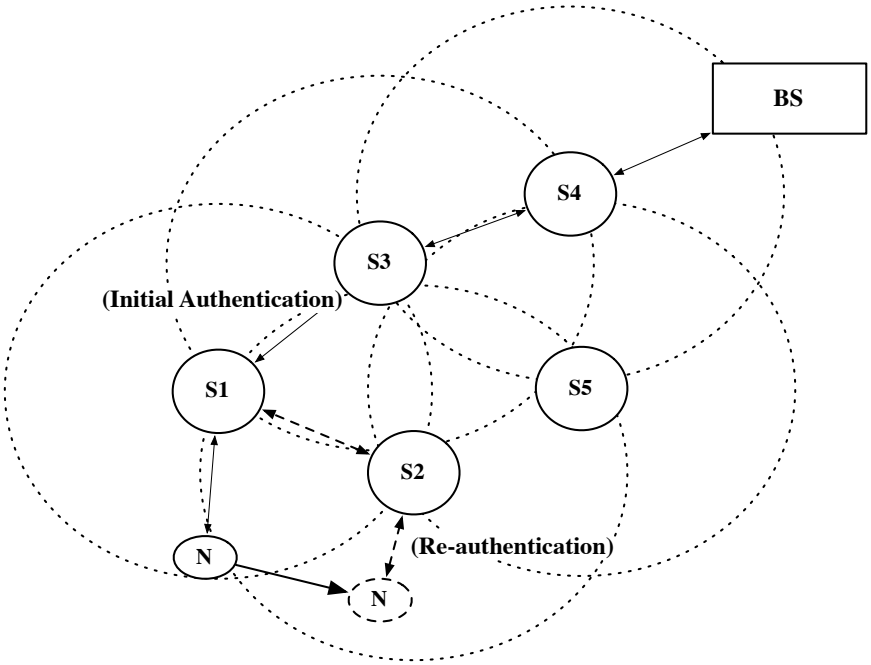


Figure 3.12: Node Initial Authentication and Reauthentication: The communication pass for the initial authentication by S_1 is N-S_1-S_3-S4. The communication pass for the reauthentication by S_2 is only N-S_2-S_1.

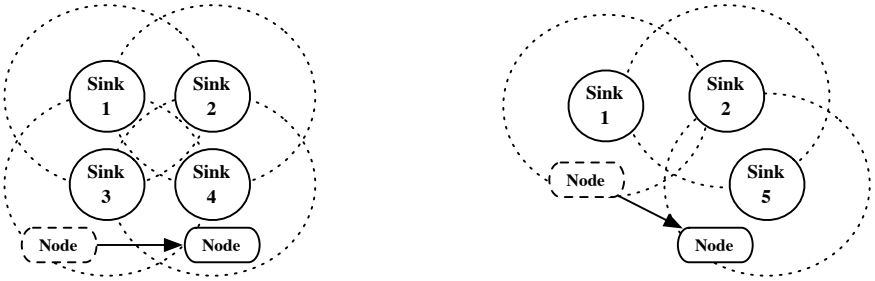
We define the notations in this section in Table 3.6. IK and CK are the integrity key and cipher key, and derived from a shared key K. When we describes $A \rightarrow B : A||B||C||m$, it means that A sends m to B indicating that C is also a participant.

Neighbor Sink List

Assume static sink nodes are distributed as in Figure 3.13 (a). In this case, the node authenticated by S_3 can be reauthenticated from the neighbor sinks S_1 or S_4 wherever it moves. However, the sinks may not be well distributed in the real environments.

In Figure 3.13 (b), a node that authenticated by S_1 cannot directly reauthenticated by S_5 since S_5 is not a neighbor sink of S_1. However, we also see that both S_1 and S_5 has the common neighbor sink S_2 that may link S_1 and S_5 for the reauthentication of N.

Thus, we define the concept of '*Neighbor Sink List*' (NSL) that stores the neighbor



(a) Ideal Environments **(b) Real Environments**

Figure 3.13: Static sensor node distribution in (a) Ideal environments (b) Real Environments

sink's information such as ID, shared secret key and public key. We assume that each sink has own NSL. Let the NSL stored in S_1 as $NSL_{S_1} = m||sign_{S_1}(h(m))$, where $m = S_1||S_2||.....||S_k$ and $sign_{S_1}$ is the signature of S_1 as in Table 3.6. Figure 3.14 shows that S_1 and S_5 has common sink, S_2 who can link the two sinks. The signature scheme can be flexibly chosen. When TinyECC is used, Elliptic Curve Digital Signature Algorithm (ECDSA) can be properly used. Whenever a new neighbor sink is found, the sink updates its own NSL.

Sink 1

Sink ID	Public key
Sink 2	pk_s1
Sink 3	pk_s2
Sink 4	pk_s3

Sink 5

Sink ID	Public key
Sink 2	pk_s1
Sink 7	pk_s2
Sink 9	pk_s3

Figure 3.14: Sink 1 and Sink 5 find the neighbor sink Sink 2 in their Neighbor Sink Lists

3.4.2 Pre-Phases of Our Protocol

Our protocol has three pre-phases: *Neighbor Discovery*, *Neighbor Sink List Set Up* and *Initial Mobile Node Authentication* as follows.

Pre-Phase 0: Neighbor Discovery

A sink S_1 generates $v_0 = sign_{sk_1}(h(HELLO||TS))$, where $HELLO$ is generic HELLO message and TS is time stamp. S_1 broadcasts v_0 with $HELLO$, TS, pk_1, and $cert_{S_1}$. Because pk_1 and $cert_{S_1}$ are only required for the Pre-Phase1, any nodes that receive $HELLO$ may ignore pk_1 and $cert_{S_1}$.

Pre-Phase 1: Neighbor Sink List Set Up

When a sink S_2 receives HELLO of S_1 that has no previous relationship with S_2, S_2 operates as follows:

P-1.a After verifying $cert_{S_1}$ using pk_{BS}, S_2 verifies v_0 using pk_1. Then S_2 randomly selects R_1, generates $u_1 = e_{pk_1}\{S_2||S_1||R_1||h(R_1)\}$, and returns u_1 to S_1 as follows.

$$S_2 \rightarrow S_1 : S_2||S_1||u_1 \tag{3.16}$$

P-1.b S_1 also verifies pk_2 and retrieves R_1 after decryption of u_1. S_1 then randomly selects R_2 and generates $K_{S_1S_2} = KDF(R_1||R_2)$. KDF is a key derivation function such as hash function. S_1 generates $u_2 = e_{pk_2}\{S_1||S_2||\ R_2||h(K_{S_1S_2}\ ||R_2)\}$ and sends u_2 to S_2 as follows.

$$S_1 \rightarrow S_2 : S_1||S_2||u_2 \tag{3.17}$$

P-1.c S_2 decrypts u_2 and retrieves R_2. S_2 also generates $K_{S_1S_2}$ and check $h(K_{S_1S_2}||R_2)$ for freshness check. S_2 generates $v_3 = MAC_{K_{S_1S_2}}(S_2||S_1||\ ACK||R_1||R_2)$ and send ACK and v_3 to S_1 as follows.

$$S_2 \rightarrow S_1 : S_2||S_1||ACK||v_3 \tag{3.18}$$

P-1.d S_1 verifies v_3 and updates NSL_{S_1} as in section 3.4.1.

As a result, S_1 shares a key $K_{S_1S_2}$ with S_2. The integrity key $IK_{S_1S_2}$ and the cipher key $CK_{S_1S_2}$ are derived from $K_{S_1S_2}$. If a sink receives $HELLO$ of sinks in the NSL, ignore this phase.

Pre-Phase 2: Initial Mobile Node Authentication

P-2.a When a node N receives v_0 from S_1, N randomly selects R_1 and generates $u_1 = e_{CK_{N,BS}}\{R_1||v_0\}$, $v_1 = MAC_{IK_{N,BS}}\ (N||S_1||u_1)$, where $CK_{N,BS}$ and $IK_{N,BS}$ are shared cipher key and integrity key between N and BS, and sends u_1 and v_1 to S_1 as follows.

$$N \rightarrow S_1 : N||S_1||u_1||v_1 \tag{3.19}$$

P-2.b After receiving u_1 and v_1, S_1 randomly selects R_2, and generates $u_2 = e_{CK_{S_1,BS}}\{u_1||R_2\}$ and $v_2 = MAC_{IK_{S_1,BS}}(S_1||BS\ ||N||u_2||v_1)$, where $CK_{S_1,BS}$ and $IK_{S_1,BS}$ are shared cipher key and integrity key between S_1 and BS. Then S_1 sends u_2, v_1 and v_2 to BS as follows.

$$S_1 \rightarrow BS : S_1||BS||N||u_2||v_1||v_2 \tag{3.20}$$

P-2.c After receiving u_2, v_1 and v_2, BS verifies v_2 and decrypts u_2. BS then retrieves R_2 and verifies v_1. BS also retrieves R_1, TS and v_0, and verifies v_0 checking if TS is valid. BS then generates $u_3 = e_{CK_{N,BS}}\{R_2\}$, $v_3 = MAC_{IK_{N,BS}}(BS||N||S_1||u_3)$, $u_4 = e_{CK_{S_1,BS}}\{R_1||u_3||v_3\}$ and $v_4 = MAC_{IK_{S_1,BS}}\ (BS||S_1||N||R_2||u_4)$, and sends u_4 and v_4 to S_1 as follows.

$$BS \rightarrow S_1 : BS||S_1||u_4||v_4 \tag{3.21}$$

P-2.d S_1 verifies v_4 and decrypts u_4. After retrieving R_1, u_3 and v_3, S_1 derives $K_{N,S_1} = KDF(R_1||R_2)$ that will be the shared session key between S_1 and N. CK_{N,S_1} and IK_{N,S_1} are cipher key and integrity key derived from K_{N,S_1} individually. S_1 then generates $u_5 = e_{CK_{N,S_1}}\{NSL_{S_1}\}$ and $v_5 = MAC_{IK_{N,S_1}}(S_1||N||R_1||u_5)$. S_1 then sends u_3, v_3, u_5 and v_5 to N as follows.

$$S_1 \rightarrow N : S_1||N||u_3||v_3||u_5||v_5 \tag{3.22}$$

P-2.e After verifying v_3, N decrypts u_3 and retrieves R_2. N derives K_{N,S_1} using R_2. N can verify v_5 and decrypt u_5. After decrypting u_5, N obtains NSL_{S_1}. N then generates $v_6 = MAC_{IK_{N,S_1}}(N||S_1||ACK||R_2||R_1)$ and sends ACK and v_6 to S_1 as follows.

$$N \rightarrow S_1 : N||S_1||ACK||v_6 \tag{3.23}$$

P-2.f After S_1 verifies v_6, S_1 authenticates N.

3.4.3 Reconnecting Mobile Node Authentication

When an authenticated node N moves and reconnects to other sink S_2, N and S_2 proceeds as follows. Assume N receives v_0 from S_2.

RA.a N randomly selects R_1, and generates $u_1 = e_{CK_{N,S_1}}\{R_1||v_0\}$ and $v_1 = MAC_{IK_{N,S_1}}(N||S_2||v_1||v_0)$. N then sends NSL_{S_1}, u_1 and v_1 to S_2 as follows.

$$N \rightarrow S_2 : N||S_2||NSL_{S_1}||u_1||v_1 \tag{3.24}$$

RA.b S_2 verifies NSL_{S_1} and checks whether S_1 is the neighbor sink of S_2 with the public key pk_{S_1}. If S_1 is the neighbor sink of S_2, S_2 generates $u_2 = e_{CK_{S_1,S_2}}\{N||R_2||u_1||v_1\}$ and $v_2 = MAC_{IK_{S_1,S_2}}(S_2||S_1||u_2)$, where CK_{S_1,S_2} and IK_{S_1,S_2} are derived from K_{S_1,S_2}, and sends u_2 and v_2 to S_1 as follows.

$$S_2 \rightarrow S_1 : S_2||S_1||u_2||v_2 \tag{3.25}$$

RA.c After verifying v_2, S_1 decrypts u_2, and retrieves N, R_2, u_1, and v_1. S_1 then finds CK_{N,S_1} and IK_{N,S_1} to verify v_1 and decrypt u_1. S_1 generates u_3, v_3, u_4 and v_4, where $u_3 = e_{CK_{N,S_1}}\{R_2\}$, $v_3 = MAC_{IK_{N,S_1}}(S_2||R_1||u_3)$, $u_4 = e_{CK_{S_1,S_2}}\{R_1||u_3||v_3\}$ and $v_4 = MAC_{IK_{S_1,S_2}}(S_1||S_2||N||\ u_4||R_2)$. And then S_1 sends u_4 and v_4 to S_2.

$$S_1 \rightarrow S_2 : S_1||S_2||u_4||v_4 \tag{3.26}$$

RA.d S_2 verifies v_4 and decrypts u_4. S_2 then retrieves R_2, u_3 and v_3. S_2 derives new session key $K_{N,S_2} = KDF(R_1||R_2)$, and also derives CK_{N,S_2} and IK_{N,S_2} individually. S_2 generates $v_5 = MAC_{IK_{N,S_2}}(S_2||N||u_3||\ v_3||\ R_1||R_2)$. S_2 then sends u_3, v_3 and v_5 to N.

$$S_2 \rightarrow N : S_2||N||u_3||v_3||v_5 \tag{3.27}$$

RA.e N verifies v_3 and decrypts u_3. N generates K_{N,S_2} and verifies v_5 using IK_{N,S_2}. N then generates $v_6 = MAC_{IK_{N,S_2}}(N||S_2||ACK||R_2||R_1)$ and sends ACK and v_6 to S_2 as follows.

$$N \rightarrow S_2 : N||S_2||ACK||v_6 \tag{3.28}$$

In real environments, the distribution of sensor nodes may not be fine-grained as in Figure 3.13 (b). When N requests the connection to S_5, S_5 may not be able to authenticate N properly because S_5 is not the neighbor sink of S_1. Figure 3.14 illustrates that S_5 is not the neighbor sink of S_1, but they have the common neighbor S_2. In this case, S_5 can authenticates N via S_2 modifying step **RA.b** as follows:

RA.b-1 S_5 checks whether S_1 is the neighbor of S_5. If S_1 is not the neighbor of S_5, S_5 finds the common neighbor sink from NSL_{S_1} and NSL_{S_5}. When S_5 has the common neighbor sink S_2 with S_1 as in Figure 3.14, S_5 randomly selects R_2 and generates $u_2 = e_{CK_{S_2,S_5}}\{N||R_2||NSL_{S_1}||u_1||v_1\}$ and $v_2 = MAC_{IK_{S_2,S_5}}(S_5||S_2||u_2)$. S_2 then sends u_2 and v_2 to S_2.

$$S_5 \rightarrow S_2 : S_5||S_2||u_2||v_2 \tag{3.29}$$

RA.b-2 S_2 verifies v_2 and decrypts u_2. S_2 then verify NSL_{S_1} and checks whether S_1 is the neighbor sink of S_2. If S_1 is the neighbor sink of S_2, S_2 generates $u_3 = e_{CK_{S_1,S_2}}\{N||R_2||u_1||v_1\}$ and $v_3 = MAC_{IK_{S_1,S_2}}(S_2||S_1||u_3)$, and sends u_3 and v_3 to S_1 as follows.

$$S_2 \rightarrow S_1 : S_2||S_1||u_3||v_3 \tag{3.30}$$

Also, step **RA.d** is also modified that S_1 sends the authenticating information to S_5 via S_2. We omit the details of modified step, since the procedure is similar to the step **RA.b-1** and **RA.b-2**.

3.5 Analysis of Protocol 2

In this section, we analyze our protocol. We show the security analysis in section 3.5.1 and performance analysis with comparing previous protocols in section 3.5.2.

3.5.1 Security Analysis

We show the security analysis of our protocol that holds the requirements: '*re-authentication*', '*confidentiality*', '*message Integrity*', and '*key freshness*'. After that we analyze the security of our protocol against known attacks.

Re-Authentication

When the node N requests the re-authentication to a sink S_2, S_2 verifies the NSL_{S_1} of S_1 that N sent to S_2. Since NSL_{S_1} is signed by S_1, any adversary cannot compromise it. In case of an malicious node N' intercepts NSL_{S_1} and tries to be authenticated by S_2, S_2 may check if N' is previously authenticated by S_1 in step 3 in section 3.4.3. Attackers fail to compromise the communication without knowing any shared secret between entities. Although Previously authenticated sink knows both random numbers R_1 and R_2 that derive the shared key, the sink cannot intercept the communication in out-of-range.

Confidentiality

We assumed that any sinks and nodes pre-share secret keys only with the base station. For the neighbor sink list set-up phase (Pre-phase 0), the attacker requires private keys to know the transmitting information. For the initial node authentication (pre-phase 2), the attacker should know the shared secret keys between sinks/nodes and the base station. For the re-authentication, the message is encrypted by the shared session key CK.

Message Integrity

It is recommended for the practical application that the size of MAC is 4 bytes, since only 40 forgery attempts per second available on 19.2kb/s channel while 2^{31} trial requires for successful forgery [38]. However, the performance of communication channel is increasing, the size of MAC should be increased in future application. Recently TinyHash [44] is introduced the efficient implementation of hash functions.

Key Freshness

Whenever a node requests the re-authentication to sinks, the node randomly selects nonce R_1 while sinks randomly selects nonce R_2. R_1 and R_2 are only valid in the session. Also, the node can verify the freshness of the communication from v_3, the sink can verify it from v_5.

Security against known attacks

We analyze the security of our protocol against the attacks that Karlof and Wargner [39] identified. Since the static parts in the networks could follow the previous models such as [3, 35], we only focus on the security of node re-authentication in this section.

The 'sinkhole attack' against our protocol fails without knowing the keys, and the 'wormhole attack' on our protocol fails since the adversary cannot send the confirmation message. Spoofed, altered or replayed routing information attack also fails with our knowing encrypted nonce in our protocol. To succeed in the replay attack, the adversary have to be able to re-use the intercepted packet. In order to succeed in the 'man-in-the-middle attack', the adversary should be be able to forge the communication. We don't consider relaying through the attackers as successful attack. The 'sybil attacks' also fails from verification of identity of nodes through sinks and the base station. And for HELLO flood attacks, we can apply the global key shared to all entities in the network that many researches such as [35] and [85] used for the efficient message broadcast and DoS attack protection. We consider the security against selective forwarding and acknowledgement spoofing are very limited due to using the timestamp and the neighbor sink list.

3.5.2 Performance Analysis

Also, we analyze performance of our protocol based on Meulenaer *et al.*'s experiments [18]. Meulenaer *et al.* shows the overall energy costs of ECC-160 point multiplcation, ECDSA-160 sign and verification. Using MICAz, ECC-160 point multiplication requires 55mJ, while ECDSA-160 sign and verification require 52mJ and 63mJ respectively. Using TelosB, ECC-160 point multiplication requires 17mJ, while ECDSA-160 sign and verification require 15mJ and 199mJ respectively. We estimates the overall computational energy cost based on Meulenaer *et al.*'s experiment results. We assume the symmetric key based computation uses AES-128, and one-way computations including key derivation, hash and MAC use SHA-1.

Table 3.7: Energy cost Comparison using MICAz based on [18]

Protocol	Computation	Communication
Huang's protocol [33]	440 mJ	0.919mJ
Our protocol	63 mJ	1.597 mJ

We compared our protocol to Huang's protocol [33] in re-authentication case as follows: Comparing communication complexity, our protocol requires 2496 bits or 312 bytes for total communications, while Huang's protocol requires 1437 bits or 180 bytes with assumption that the node ID and random number are 64 bits each, and the modulus for ECC is 160 bits. We also assumed the hash size is 160 bits and maximum number of neighbor

Table 3.8: Energy cost Comparison using TelosB based on [18]

Protocol	Computation	Communication
Huang's protocol [33]	136 mJ	1.106mJ
Our protocol	19 mJ	1.921 mJ

sinks is 6. Following the experimental results in [18], our protocol requires about 1.597 mJ for total energy costs, while Huang's protocol requires about 0.919 mJ using MICAz. Our protocol also requires about 1.921 mJ, while Huang's protocol requires about 1.106 mJ using TelosB.

However, comparing computation complexity, our protocol requires two AES-128 encryption/decryption and five SHA-1 computations for a mobile node, and one ECDSA-160 verification, seven AES-128 encryption/decryption and seven SHA-1 computations for a sink. On the other hand, in Huang's protocol, a mobile node requires four ECC multiplications, two AES-128 encryption/decryption and two SHA-1 computations, while a sink requires four ECC multiplication, one AES-128 encryption/decryption and three SHA-1 computation. Huang's protocol requires that each node spend about 220 mJ for ECC point multiplication (total 440 mJ), while our protocol requires only 63 mJ in sink side when MICAz is used. Using TelosB, 68 mJ per node (total 136 mJ) is required in Huang's protocol, while only 19 mJ is required in our protocol.

Table 3.7 and Table 3.8 show computation and communication energy cost of Huang's protocol and our protocol using MICAz and TelosB individually. We show the comparison of energy cost when a mobile node continually move in the network in Figure 3.15.

As a result, even though our protocol requires more communication energy cost than Huang's protocol, our protocol shows about 8 times more energy efficiency with less ECC computation. Compare to other symmetric key based protocol such as Ibriq's protocol [35]'s protocol, our protocol shows less communication overhead that Ibriq's protocol requires 480 bytes, while our protocol requires about 160 or 240 bytes constantly when the hop distance between the mobile node and the base station is 3, with MAC size as 4 bytes, the time stamp as 8 bytes, nonce as 8 bytes, the key size as 16 bytes and the source and target IDs as 1 byte individually. Although our protocol requires one ECDSA-160 verification, it does not be the significant in our assumption that the sink has enough computational power. We show the comparison of message size in Figure 3.16.

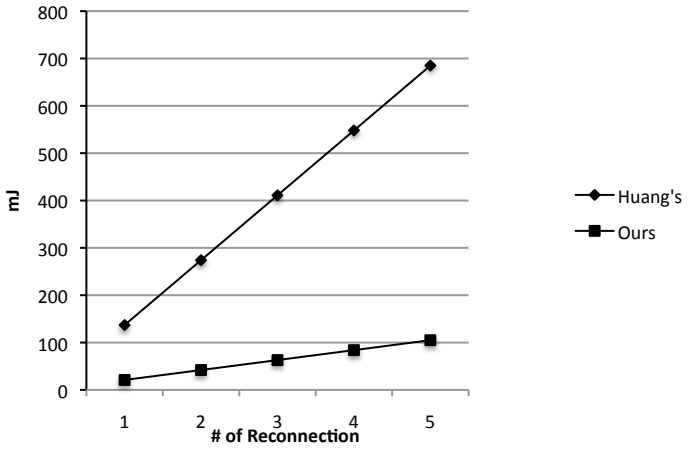


Figure 3.15: Comparison of energy costs for the number of re-authentication of mobile node using TelosB [18]: Huang's protocol [33] and Proposed protocol.

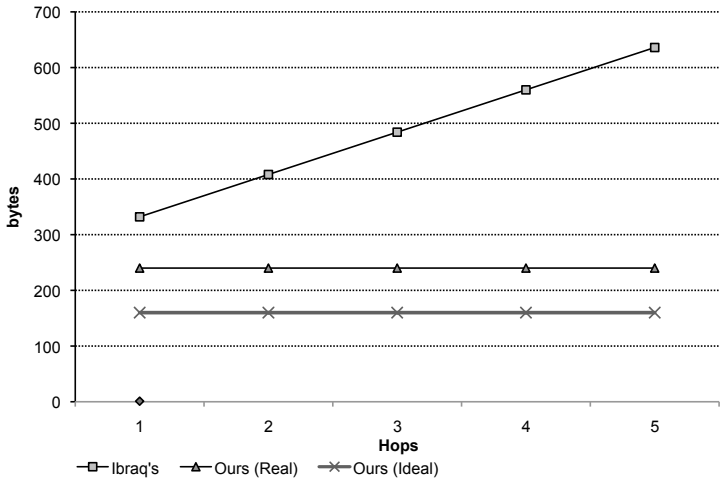


Figure 3.16: Comparison of message size required for the re-authentication: Ibriq's protocol [35], proposed protocol in the ideal environment and in the real environment

4. Efficient Sensor Node Authentication via Mobile Network

In this chapter, we propose an efficient and secure authentication and key exchange protocol between sensor nodes and the smartphone with sensors in order to bring more benefits from the consolidation of WSNs and 3G mobile network (3G-WSN) based on the standard architecture. Since efficient resource management is one of the most important requirements in WSNs, our approach concentrates on how to minimize the energy consumption and inefficient message transmission.

The followings are our main contribution in this chapter:

- Design the protocol applicable to the standard architecture such as IEEE 802.15.4 based Zigbee and 3GPP mobile network architectures,
- Integrate the sensor network into the 3G mobile network as an application based on the standard Generic Authentication Architecture (GAA) [78, 74],
- Minimize the communication and computation overheads in the sensor network for mutual authentication between a sensor attached smartphone and a sensor node.

Chapter 4 is organized as follows: In Section 4.1, we show the issues in 3G-WSN network. We propose our protocol in Section 4.2. Section 4.3 shows the analysis of our proposed protocol and the comparison with previous models.

4.1 Issues in 3G-WSN Networks

There are several efforts of integrating 3G mobile network and WSN such as an example of Figure. 4.1. In the scenario, the mobile network is deployed at the intermediate part in the network. While the communication is through WSNs at the end points, the intermediate communication is through the mobile network.

However, such applications have several limitations since there is no clear consideration on the security interworking between two different networks. Although a few studies deploying EAP into WSN such as [5] exist, there are the significant performance gaps between WSN and mobile network as in Table 4.1, which occur the degradation of the

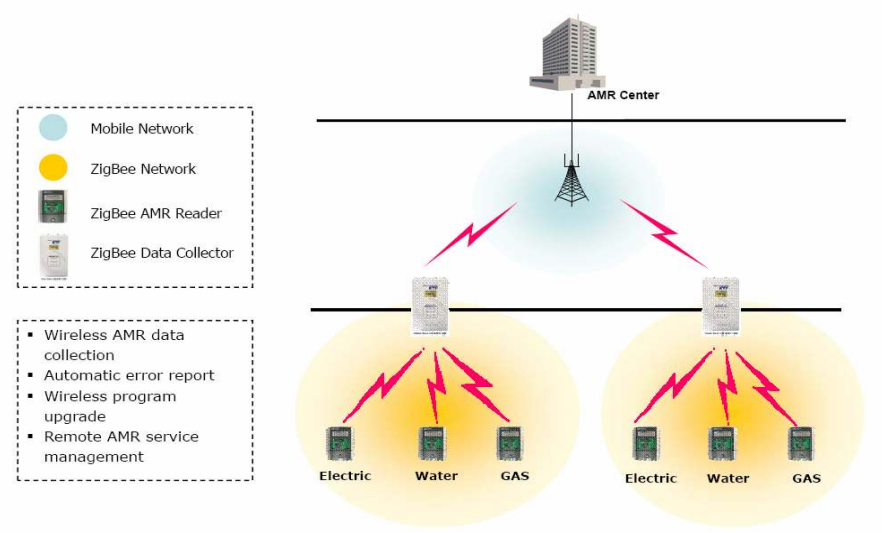


Figure 4.1: An Example Application of 3G-WSN Integrated Network [14]

overall network performance due to the weaker capability of the sensor network. In such environments, applying the interworking mechanism in Section 2.2.5 is rather insufficient.

In the previous 3G-WSN network application as in Figure 4.2, the communication through mobile network enables the solid communication capability than WSN-only environments, though the communication through sensor network becomes the bottleneck of the overall communication.

Table 4.1: Comparison between WSN and Mobile Network

Type	WSN	Mobile Network
Speed	250kbps (1Mbps)	75Mbps (300Mbps)
Tech.	Zigbee	Long Term Evolution
Standard	IEEE 802.15.4	3GPP
Coverage	30 - 50 m	3 - 5 km

Thus, our main motivation is to overcome such bottlenecks and maximize the synergy of interworking between 3G and WSN networks by concentrating on the most procedures for the authentication of the sensor nodes into the mobile network communication. Figure

4.3 shows our proposed model that the sensor attached smart phone communicates to the authentication server via mobile network, and directly communicates to the sensor. In the architecture, the sensor network can be a kind of third party application in the mobile network applying the generic authentication architecture [74].

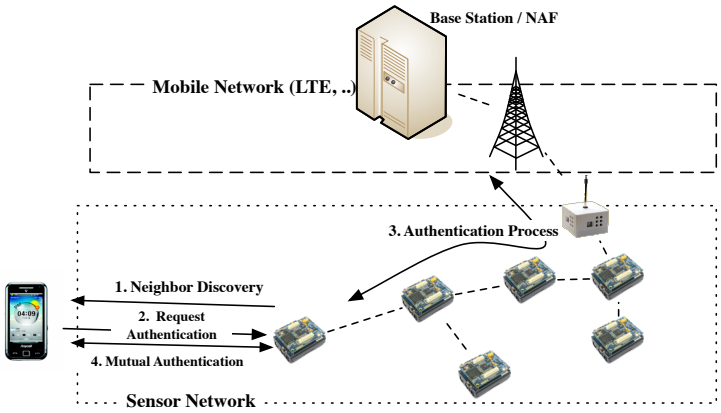


Figure 4.2: Previous 3G-WSN models integrate sensor network as one of network.

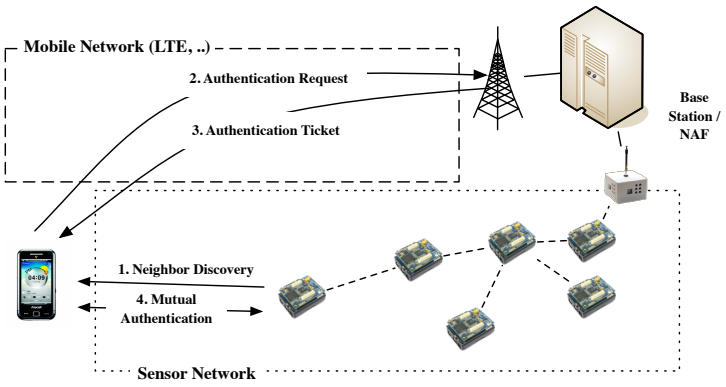


Figure 4.3: Proposed model integrate sensor network as one of application into mobile network.

4.2 Proposed Protocol

4.2.1 System model

We define the system model of our protocol in Figure 4.4. The sensor attached smartphone as a mobile device (MD) that has GAA module, Zigbee module. GAA module is used for the generic bootstrapping architecture and the communication in the mobile network [74], and Zigbee module is used for the communication in the sensor network. USIM stores the secret information including the seed key. The network consists of mobile network entities such as BSF and NAF, and the sensor network entity such as sinks. We assume a sensor network that consists of a base station and sensor nodes (sinks). When sinks are deployed, each sink shares a unique key with the base station. The establishment of the sensor network is done using any previous security protocols such as [35, 85] which is out of scope.

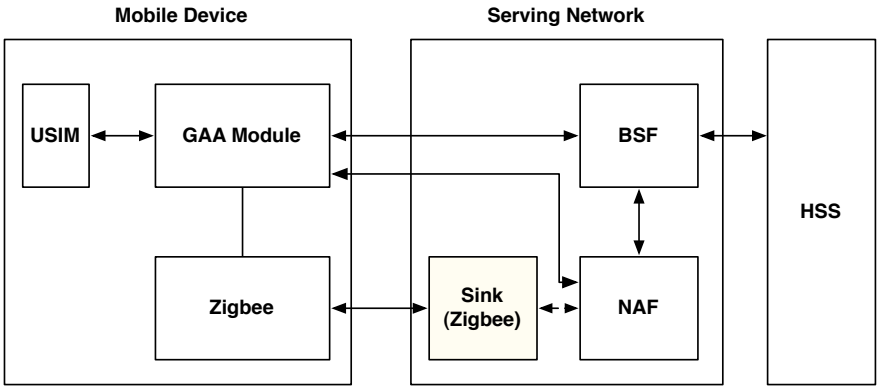


Figure 4.4: The system model of our protocol

4.2.2 Protocol Description

The protocol is mainly divided into two parts: Phase 1 is operated in the mobile network, and Phase 2 is operated in the sensor network. We show the notations and the message types used in the protocol in Table 4.2 and Table 4.3 respectively. APPREQ and APPRES are transmitted in Phase 1 using GAA. $AUTHREQ$, $AUTHRES$, $AUTHCON$ are the messages transmitted in Phase 2 via the sensor network.

Table 4.2: Notations

Type	Description
MD	Mobile Device, a sensor attached phone
S_i	Sensor node, a sink i
NAF	Network Application Function in GBA
BSF	Bootstrapping Server Function in GBA
$MAC_k(m)$	MAC of a message m using key k
$e_k\{m\}$	Encrypt m using k
$h(m)$	Hash output of m
TS	Timestamp
CK_i	Cipher key of an entity i
IK_i	Integrity key of an entity i

Table 4.3: Message Type used in the Protocol

Type	Description	Network
$HELLO$	HELLO message	WSN
$APPREQ$	Request of service	3G
$APPRES$	Response of service	3G
$AUTHREQ$	Request of authentication	WSN
$AUTHRES$	Response of authentication	WSN
$AUTHCON$	Confirmation of authentication	WSN

Neighbor Discovery Phase

Neighbor discovery is the periodical operation by the sensors. Every sensor broadcast HELLO message to find the neighbor sensors. A sink S_1 periodically broadcasts HELLO with generating u_0 and v_0, where $u_0 = e_{CK_{S_1}}\ \{R_0||TS\}$ and $v_0 = MAC_{IK_{S_1}}\ (u_0)$ as shown in Fig. 4.5. R_0 is a random nonce selected by S_1, and TS is a timestamp.

$$S_1 \rightarrow Broadcast : HELLO||S_1||u_0||v_0 \tag{4.1}$$

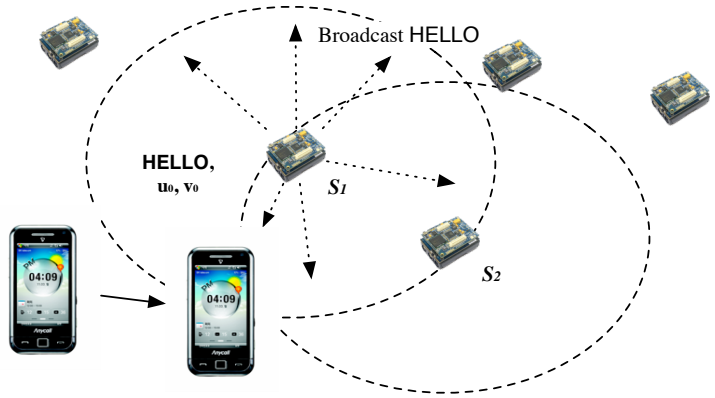


Figure 4.5: Neighbor Discovery: Each Sink such as S_1 periodically broadcasts HELLO.

When MD receives the HELLO message from S_1 already authenticated, MD ignores this phase. Thus, the energy cost and message size of this phase is not considered for the performance analysis of this protocol.

Phase 1: Authentication using Generic Authentication Architecture

If MD is firstly joining the network, MD has to operates the generic bootstrapping architecture (GBA) [74] to have the shared key CK_{MD} and IK_{MD} between MD and the serving network. When unauthenticated MD receives HELLO from S_1, MD requests the authentication of S_1 to the NAF. MD generates u_1 using CK_{MD} and v_1 using IK_{MD}, where $u_1 = e_{CK_{MD}}\{S_1||u_0||v_0\}$ and $v_1 = MAC_{IK_{MD}}(MD||u_1)$. After that MD send u_1 and v_1 to NAF.

$$MD \rightarrow NAF : APPREQ||MD||u_1||v_1 \tag{4.2}$$

If the NAF has no information of MD, the NAF asks the BSF about MD and obtains CK_{MD} and IK_{MD} from GBA process. After that NAF generates u_2 and v_2, where $u_2 =$

$e_{CK_{S_1}}\{h(R_0||CK_{MD})||\ h(R_0||IK_{MD})\}$ and $v_2 = MAC_{IK_{S_1}}\ (R_0||u_2)$. After that the NAF also generates u_3 and v_3, where $u_3 = e_{CK_{MD}}\{R_0||\ TS||h(R_0||CK_{S_1})||h(R_0||IK_{S_1})||\ u_2||v_2\}$ and $v_3 = MAC_{IK_{MD}}(APPRES||u_3)$. And, the NAF sends u_3 and v_3 to MD.

$$NAF \rightarrow MD : APPRES||MD||u_3||v_3 \tag{4.3}$$

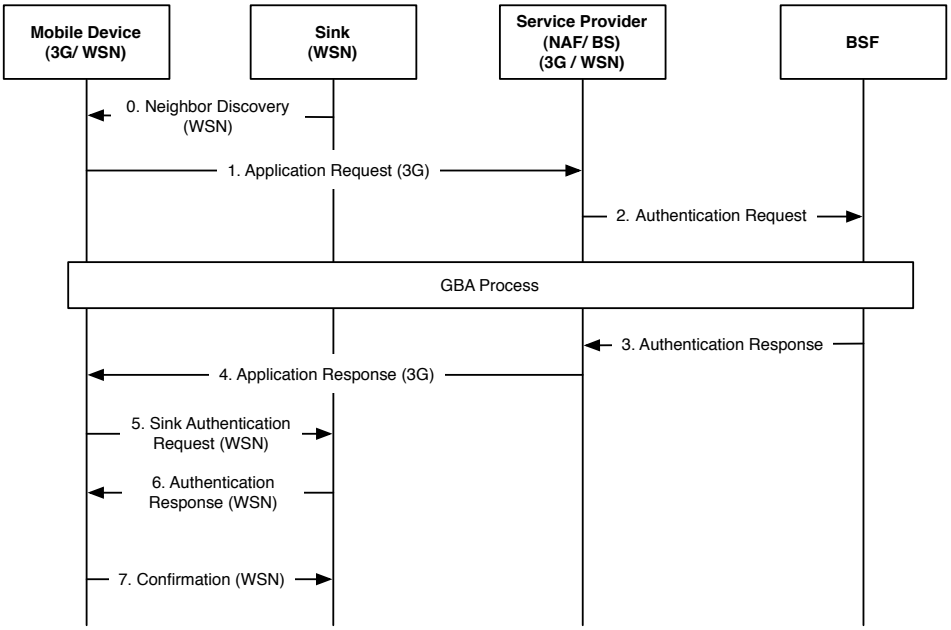


Figure 4.6: Overall Message Flow in the Protocol

After verifying v_3 and decrypting u_3, MD retrieves R_0, $h(R_0||CK_{S_1})$ and $h(R_0||IK_{S_1})$. Then MD generates the shared session key between MD and S_1, CK_{S_1MD} and IK_{S_1MD} as follows:

$$CK_{S_1MD} = KDF(h(R_0||CK_{S_1})||h(R_0||CK_{MD})) \tag{4.4}$$

$$IK_{S_1MD} = KDF(h(R_0||IK_{S_1})||h(R_0||IK_{MD})) \tag{4.5}$$

Phase 2: Mutual Authentication between MD and Sensor

After the authentication process between MD and NAF, MD generates the shard session keys CK_{S_1MD} and IK_{S_1MD}. Using IK_{S_1MD}, MD computes v_4, where $v_4 = MAC_{IK_{S_1MD}}$

$(AUTHREQ||MD||S_1||R_0||u_2||v_2)$ and send v_4 with u_2 and v_2 to S_1 as follows.

$$MD \rightarrow S_1 : AUTHREQ||MD||S_1||u_2||v_2||v_4 \tag{4.6}$$

When S_1 receives u_2, v_2 and v_4, S_1 checks the validity of v_2 at first. After that S_1 decrypts u_2 and retrieves $h(R_0||CK_{MD})$ and $h(R_0||IK_{MD})$. S_1 generates IK_{S_1MD} with $h(R_0||IK_{MD})$ and verifies v_4. Finally, S_1 generates v_5 as the response to MD, where $v_5 = MAC_{IK_{S_1MD}}(AUTH\ RES||S_1||MD||R_0)$ and sends it to MD as follows:

$$S_1 \rightarrow MD : AUTHRES||S1||MD||v_5 \tag{4.7}$$

After MD verifies v_5, MD generates v_6 for the confirmation of the authentication response, where $v_6 = MAC_{IK_{S_1MD}}\ (AUTHCON||MD||S_1||R_0 + 1)$ and sends it to S_1 as follows:

$$MD \rightarrow S_1 : AUTHCON||MD||S_1||v_6 \tag{4.8}$$

R_0+1, denoted the update of R_0 with addition, is used for the freshness check, and can be substituted with other methods. S_1 completes the authentication of MD by checking the validity of v_6.

4.3 Analysis

In this section, we show the analysis of the proposed protocol. At first, we show the security analysis of our proposed protocol, and then show the efficiency of our proposed design by comparing with the previous models.

4.3.1 Security of Proposed Protocol

We analyze the security of our protocol that provides the confidentiality, authentication, and security against several known attacks.

Security Against Key Compromise

The share session keys are initially generated using the master seed key stored in USIM. From the seed key, the CK_{MD} and IK_{MD} are shared between MD and NAF using GAA under the mobile network. We do not consider security of such standardized operations [71, 74], and only focus on the security of the communication under WSN.

Since the transmitted key generating informations are encrypted, an adversary $\mathcal{A}$ fails to know such information.

The sink S_1 periodically generates random nonce R_0. Thus, S_1 can verify that the requests of authentication are from the directly linked sinks or nodes. The shared session keys CK and IK are generated using R_0.

Assume the node S_1 is compromised, the attacker may try to know the value of CK_{MD} and IK_{MD} in order to impersonate MD. However, $\mathcal{A}$ is only able to generate the shared session key between MD and S_1 using the only known informations of MD are $h(R_0||CK_{MD})$ and $h(R_0||IK_{MD})$. $\mathcal{A}$ cannot know CK_{MD} from $h(R_0||CK_{MD}$ due to the *one-wayness* of cryptographic hash function. Also, a malicious MD only receives $h(R_0||CK_{S_1})$ and $h(R_0||IK_{S_1})$, and MD cannot predict the CK_{S_1} and IK_{S_1}) as in Figure 4.7.

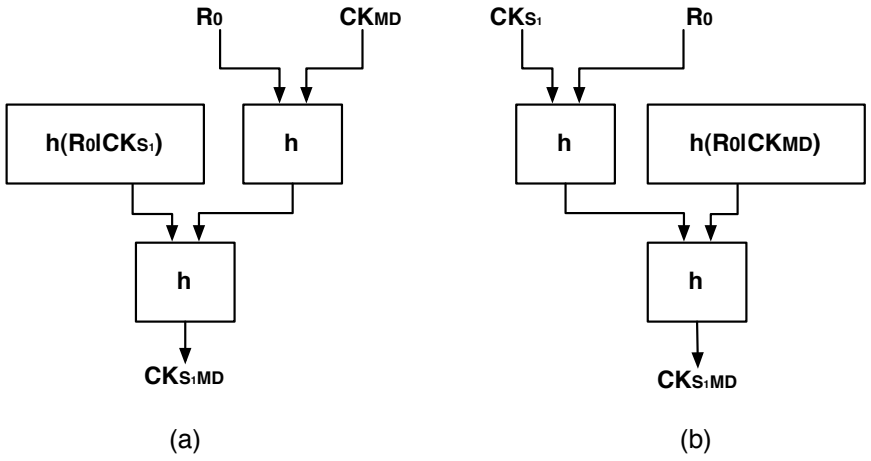


Figure 4.7: (a)Key generation in MD (b) Key Generation in Sensor

Security Against Message Forgery

The security of the MAC depends on the security of the hash function such as SHA-1 [55]. 20 bytes of MAC is used for the theoretical security of hash function, 4 bytes of MAC is recommended for the practical application in [38] though, since only 40 forgery attempts per second available on 19.2kb/s channel while 2^{31} trial requires for successful forgery. However, the performance of communication channel is increasing as in Table 4.1, the size of MAC should be increased in future application. Recently the efficient implementation of hash functions is introduced in [44].

In our protocol, every packet is protected by the Message Authentication Code (MAC). The outside adversary $\mathcal{A}$ should be able to forge the MAC to success the attack. Thus,

our protocol is secure against the Man-in-the-Middle attack while the adversary has no efficient way to forge MAC.

Security against known attacks

Since the most parts of the proposed protocol are operated in the mobile networks, most attacks on the sensor network [39] do not affect on the proposed protocol. Thus we only consider the security of Phase 2 that the direct authentication process between MD and S_1.

The replay attack fails in the protocol due to the random nonce used in the packet in each session. Wormhole attack on our protocol fails since the adversary cannot send the confirmation message. Spoofed, altered or replayed routing information attack also fail with our knowing encrypted nonce in our protocol. The sinkhole attack against our protocol fails without knowing the keys. Sybil attacks also fails from verification of identity of nodes.

4.3.2 Performance Comparison

We defined the three cases of the network as follows: (Case 1) The wireless sensor network environment that communications like raw data sensing, control and data transmission are operated by sensor nodes as shown in Figure 4.8 (a), (Case 2) Previous 3G-WSN network that integrates the mobile network and the sensor network in the intermediate communication as in Figure 4.8 (b). (Case 3) Proposed 3G-WSN network that the sensor network is integrated as one of applications of 3G network as in Figure 4.8 (c).

Case 1: Wireless Sensor Network

Case 1 is the WSN that the communication is under the sensor network operated by sensor nodes. In this case, the longer hop distance between MD and BS (Base Station) invokes more energy consumption. For example, the previous protocol [35] spends aapproximately-bout 350 μJ for the transmission during authentication and key agreement procedure in each hop. The total energy cost is approximately 700 μJ in 1 hop cases, and is increased depending on the hop distance. When the hop distances are five times longer, the energy cost is also increased to approximately 3800 μJ for the authentication of MD as in Table 4.4.

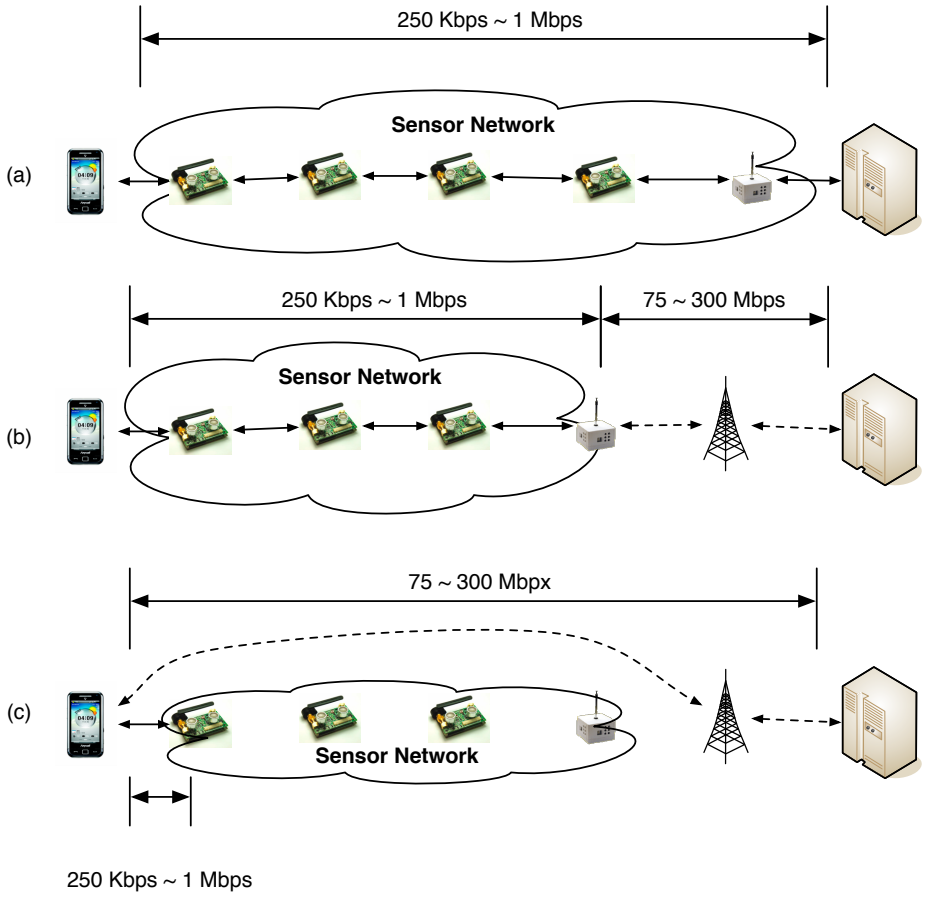


Figure 4.8: (a) Case 1: Communication in WSN. (b) Case 2: Communication in Previous 3G-WSN. (c) Our Proposed Model

Case 2: Previous 3G-WSN Network

Case 2 is the 3G-WSN network that the mobile network is integrated to the sensor network as the intermediate network [14]. Such integration provides the more efficiency in the authentication process, since the parts of communication passes of Case 1 are substituted to the mobile network that has better capabilities. In Table 4.1, the data rate of general 3G mobile networks is about 300 times faster than Zigbee-based WSN, so it can provide stable and robust communication environments, too. However, the same protocol as in Case 1 is used in the WSN parts due to the simple integration between heterogeneous network. Also, as shown in figure 4.8 (b), the energy consumption in the sensor networks still exists because at least three nodes are connected via a sensor network. Moreover, there can be additional overheads in the integrating the two different networks. For instance, the integration of 3G network and WLAN is via EAP. Thus, it can't fully take advantage of the outstanding capability of 3G mobile network due to the unreliable link status of Zigbee.

Our Proposed 3G-WSN Model

On the other hand, our proposed model integrates sensor network to the mobile network as an application. Since the information for the mutual authentication between MD and S_1 is transmitted under the mobile network in Phase 1 of the protocol, the communication in the sensor network is only necessary for the direct communication in 1 hop between MD and S_1 in Phase 2.

Comparison of Cases

For measuring the approximate communication overheads in each design, we defined the message size with MAC size as 4 bytes, the time stamp as 8 bytes, nonce as 8 bytes, and key size as 16 bytes as shown in [3]. And, We set the source and target IDs as 1 byte, respectively. For our protocol, we also set the message types as 1 byte. Comparing above three cases, the energy cost for the transmitting the messages are estimated based on the experimental results in [18], which used the MICAz running at 7.37 MHz and TelosB at 4 MHz for application data rates of respectively 108 kbps and 75 kbps. Based on the such results, our proposed protocol shows approximately 169 μJ in the authentication between MD and a sink, concentrating the most communication to the mobile network.

Table 4.4 shows the more detailed comparison of three cases for the authentication of MD. Our protocol shows the significant efficiency than other previous models. Since

Table 4.4: Comparison

Protocol	Case 1	Case 2	Proposed
Comm. Type	WSN	3G -WSN	3G -WSN
WSN	Network	Network	Application
Interworking	N/A	Undetermined	GAA [74]
Num. of Nodes	5	> 5	1
Energy (μJ)	707	>707	172
Tot. Msg. (bytes)	744	> 592	33
Tot. Eng. (μJ)	3,869	$> 2{,}288$	172

only two nodes are involved in the communication under the sensor network in Phase 2, overall message size is small and static. Also, the energy cost for transmission is also dropped by about 90 percent than previous protocol. The computation overhead is not considered for the performance analysis, since such overhead is negligibly lower than in the communication.

Therefore, the separated communication suited application's purpose in 3G network and WSN enables us to use the maximized benefits of the consolidated network, the more applicable architecture.

Concentrating the Communication to Mobile Network

Most procedures in our propose protocol is operated under the mobile network. Although the energy cost is significantly reduced compare to the previous models, there are the another overheads in the communicating under the mobile network. However, the only resource constrained entity in the mobile network is the MD that is the sensor attached smartphone. Moreover, the smartphone is daily recharged in average use, while the sensor nodes in WSN have one-lifetime depending on the lifetime of the battery attached. Thus, shifting the power consumption to the smartphone and the mobile network enables the significantly longer lifetime in the sensor network.

5. Deploying ID-based Cryptosystem for Advanced Security of Next Generation Mobile Network

In this chapter, we introduce the advanced security architecture and application for the next generation mobile network applying ID-based cryptosystem. We first briefly introduce the ID-based cryptosystem in Section 5.1. We then propose the generic authentication architecture using IDBC in Section 5.2. We also propose the secure voice over IP and the lawful interception using IDBC in Section 5.3 and Section 5.4, respectively.

5.1 Brief Overview on ID-based Cryptosystem

The concept of ID-based cryptosystem (IDBC) is based on properties of pairing and firstly shown by Shamir [66] in 1984. The practical IDBC based models began to be widely studied after Boneh and Franklin [11] proposed the encryption schemes in 2002. We summarize some concept of bilinear pairings on elliptic curves in this section.

Let G_1 and G_2 be additive and multiplicative groups of the same large prime order q, respectively. Let P be a generator of G_1.

A pairing is a map $e : G_1 \times G_1 \rightarrow G_2$ which satisfies the following properties:

Bilinearity For all $P, Q, R \in G_1$, we have $e(P, Q + R) = e(P, Q) \cdot e(P, R)$; and $e(P + Q, R) = e(P, R) \cdot e(Q, R)$. Furthermore, for all a, $b \in Z_q$: $e(aP, bQ) = e(P, Q)^{ab}$.

Non-degeneracy $e(P, P) \neq 1_{G_2}$, where 1_{G_2} is the identity element of G_2. This also can be interpreted as: if $e(P, Q) = 1$ for all $Q \in G_1$, then $P = O$.

Computability There exists an efficient algorithm to compute $e(P, Q)$ for and $P, Q \in G_1$.

For bilinear pairings on elliptic curves, group G_1 is usually a subgroup of the points on an elliptic curve over a finite field, *i.e.* $E(F_q)$, and G_2 is a subgroup of the multiplicative group of a related finite field, *i.e.* F_{q^r} where r is known as the embedding degree or the security multiplier. Choosing the value of r affects the pairing computation efficiency. The higher the value of r is, the less efficient the pairing computation will be.

In order to construct the bilinear pairing, we can use the Weil pairing [50] or the Tate pairing [28] on an elliptic curve over a finite field. Usually, the supersingular elliptic curves with distortion maps [81] applied to both Weil and Tate pairings provide pairings with non-degeneracy. However, for ordinary curves, another technique introduced by Boneh *et al.* [11] called trace maps needs to be used in order to provide the non-degenerate property. The first efficient algorithm to compute pairings in polynomial time was introduced in [52] and can be used for computing both Weil and Tate pairings. The Tate pairing implementation achieve better efficiency compared with that of the Weil pairing.

5.1.1 Security Problems and Assumptions

Frequently, cryptographic primitives rely on mathematical hard problems. We define some computational and decisional problems in group G_1 in this section.

Bilinear Diffie-Hellman(BDH) We say that a randomized algorithm $\mathcal{IG}$ is a bilinear Diffie-Hellman (BDH) parameter generater, if:

1. $\mathcal{IG}$ takes as input a security parameter $k \geq 1$
2. $\mathcal{IG}$ runs in polynomial time in k, and
3. $\mathcal{IG}$ outputs a prime number q, the description of groups G_1, G_2 of the same prime order q as well as a pairing map $e : G_1 \times G_1 \rightarrow G_2$.

We denote the output of $\mathcal{IG}$ as $\mathcal{IG}(1^k)$ which includes $< G_1, G_2, e >$. We assume that the computational complexity of $\mathcal{IG}$ is $O(k^n)$. Also the computational complexity in groups G_1, G_2, and pairings e at most $O(k^{n_1})$, $O(k^{n_2})$, and $O(k^e)$, respectively. We have $n, n_1, n_2, e \in N$ are order of the polynomial time algorithm.

Discrete Logarithm Problem (DLP) . Let G be a group of prime order q which was output by $\mathcal{IG}$, and P be a random generator of G. The Discrete Logarithm Problem in G is defined as follows: Given $< P, aP >$ with uniformly random choice of $a \in Z_q^*$, find a.

Computational Diffie-Hellman Problem (CDHP) . Let G be a group of prime order q which was output by $\mathcal{IG}$, and P be a random generator of G. The Computational Diffie-Hellman Problem in G is defined as follows: Given $< P, aP, bP >$ with uniformly random choice of $a, b \in Z_q^*$, compute $abP \in G$.

CDH Assumption . A probabilistic algorithm $\mathcal{A}$ is said to be (t, ϵ)-break-CDH in a cyclic group G if $\mathcal{A}$ runs at most time t, computes the Diffie-Hellman function $DH_{P,q}$

$(aP, bP) = abP$, with input (P, q) and (aP, bP), with a probability of at least ϵ, where the probability is over the coins of $\mathcal{A}$ and (a, b) is chosen uniformly from $Z_q \times Z_q$. The group G^* is a (t, ϵ)-CDH group if no algorithm (t, ϵ)-break-CDH in this group.

The CDH assumption, intuitively, implies that there is no polynomial algorithm A has non-negligible advantage (in k) in solving the CDHP for G generated by $\mathcal{IG}(1^k)$.

Decisional Diffie-Hellman Problem (DDHP) . Let G be a group of prime order q which was output by $\mathcal{IG}$, and P be a random generator of G. The Decisional Diffie-Hellman Problem in G is defined as follows: Given $< P, aP, bP, cP >$ with uniformly random choice of $a, b, c \in Z_q^*$, decide if $abP = cP$.

In group G, DDHP is easy as shown in [37]. This can be easily seen by observing that given $< P, aP, bP, cP > \in G$, we have: $c = ab \mod q \equiv e(P, cP) \equiv e(aP, bP)$.

Gap Diffie-Hellman Problem (GDHP) . Let G be a group of prime order q which was output by $\mathcal{IG}$. The Gap Diffie-Hellman Problem in G is to solve the CDHP in G given that there exists an efficient polynomial time algorithm which solves the DDHP in G.

Details about GDH groups can be found in [11, 12, 37].

Using the properties of IDBC, we can easily construct key exchange or signature protocol. Our TD model is based on the property of bilinearity, and the security of temporary private key is based on the computational hardness of elliptic curve discrete logarithm problem (ECDLP). The details of ID based cryptography are explained in [49].

5.1.2 Inherent Key Escrowing Property under ID-based Cryptosystem

The public key distribution in IDBC is as follows: For a user A, the key generation center (KGC) generates a master key s, where $s \in Z_P^*$, and compute $H(ID_A)$ with A's unique ID, ID_A, and a hash function $H : Z_P^* \to G$, and computes $sH(ID_A)$ with s and $H(ID_A)$, which is the A's private key. On the other hand, $H(ID_A)$ is used as A's public key. Extract s from $H(ID_A)$ and $sH(ID_A)$ has the same computational complexity as solving Discrete Logarithm Problem (DLP) [9].

Using the master key s securely stored in the KGC enables generation of all user's private keys. In general, KGC can be also the escrow agency. Thus, IDBC has the inherent

key escrowing property by storing the master key in the KGC without any additional key escrowing process.

5.2 Advanced Security Architecture in Next Generation Mobile Network

In this section, we improve the 'key-insulated' model [20, 21, 56] and show 'Trust Delegation' model that is resilient against not only the key exposure but also the key loss. We also provide the secure and efficient public key management for the next generation mobile networks. Our trust delegation model is based on the *ID-based cryptosystem* (IDBC) [10, 49], and achieve great benefit regarding the efficiency of public key management. Compared with the current architecture [75], our model does not require the involvement of symmetric key base architecture [74], and has only one third of transaction that helps the resilience against DoS attacks to mobile networks[79].

5.2.1 Trust Delegation Concept

There were the several ideas against the key exposure problem. In 2002, Dodis *et al.* proposed the 'key insulated public key cryptosystem' for the encryption[20] and 'Key insulated signature' (KIS) scheme for the signature generation [21]. Later, Ohtake *et al.* showed more efficient KIS scheme and showed the application that a large-scale multi-receiver authentication system in which a signer communicates with a huge number of receivers [56]. In such KIS schemes, the 'master' private key remains in the secure storage, and the 'temporary' private key generated from the master key is actually used for the security applications.

However, such KIS schemes have no consideration for the mobile network. In the mobile network, the losses of data including keys in ME occasionally happen, while the KIS schemes are focusing on resilience against the leakage of the master key. Since the old temporary key is required to generated the updated temporary key [21, 56], the loss of key in ME disables the key update. For example, TSK_3 and later keys cannot be generated in case TSK_2 is lost in Figure 5.2 (a). Also, KIS schemes use N number of the temporary private keys, and each key is used during a constant time period t. After $t \times N$ times later, the large overhead for reconstructing the temporary key set is required [21].

Moreover, KIS schemes are deeply related to specific security protocol. For instance, Ohtake *et al.*'s KIS scheme [56] is based on Abe-Okamoto proxy signature scheme [2].

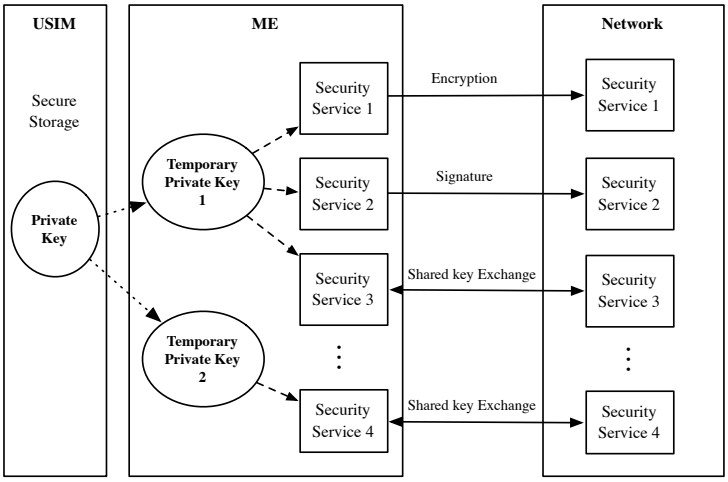


Figure 5.1: Trust Delegation Model enables the various security applications such as encryption, signature generation, and the shared key exchange using multiple temporary private key.

Thus, Simultaneous deployment of both KIS scheme and the encryption scheme requests separated process to generate the temporary keys, which can be the potential security threat.

Instead KIS schemes, our design is to provide the 'common' architecture that supports practical mobile networks. Since the design criteria is rather different to the KIS schemes, we introduce the alternative model of 'Trust Delegation' (TD) employing the ID-based cryptosystem (IDBC) (refer section 5.1.) that the user's identity is used as the public key and private key as shown in section 5.2.2. Because the old temporary key is not required to update the new temporary key as in Figure 5.2 (b), our TD model is not only resilient against the loss of key, but also provides simultaneous invocation of multiple distinct temporary private keys. Also, TD model enables the various security services computed in the mobile device while the private key is securely stored in USIM. Figure 5.1 depict the brief TD model.

5.2.2 Basic Scheme

In this section, we explain our proposed trust delegation model for the mobile networks. Section 5.2.2 shows the private key distribution in initial setup that users obtain their private key. Section 5.2.2 shows the session key setup between peer users. We propose

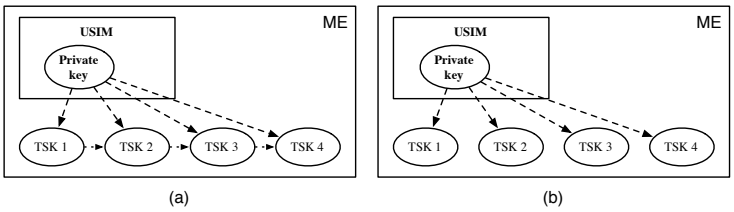


Figure 5.2: (a) Temporary private keys are linked in KIS model (b) Each temporary key has no link in 'Trust Delegation' model

the enhanced generic authentication architecture in Section 5.2.3. We define notations in Table 5.1, and the message format to request to USIM in Table 5.2.

Initial Setup: Private Key Distribution

Assume the Key Generation Center (KGC) that is a trusted entity that distributes the private keys to users. KGC generates a random integer $s \in Zp^*$, which will be the master secret of KGC. Each subscriber owns the unique identity ID. KGC distributes the private key $sk_{ID} = s \cdot H(ID)$ for each subscriber, where the hash function $H : Z_p^* \rightarrow G_1$. The symbol '$\cdot$' denotes the scalar point multiplication over Elliptic curve. The private key sk_{ID} is initially distributed in off-line environment. In practical application, users obtain sk_{ID} stored in USIM when they subscribe mobile services.

Session Key Establishment Between Peer Entities

Assume two users A and B try to establish their secure communication. Each entity has a device ME equipped with USIM U. Then, A initiates the session key establishment in ME_A and proceed following steps:

P.1. ME_A generates a timestamp TS_A. And then ME_A sends $REQ1$, ID of B and TS_A to U_A.

P.2. U_A generates a random nonce r_A and generates e_A and sig_A, and return them to ME_A, where $e_A = e_{pk_B}(r_A)$ and $sig_A = sign_{sk_A}(e_A||TS_A)$.

P.3. ME_A sends REQ, e_A, TS_A, and sig_A to ME_B.

P.4. ME_B sends $REQ2$, e_A, TS_A, and sig_A to U_B.

Table 5.1: Notations

Notation	Description
ID	Identity of User.
r_{ID}	Random nonce generated by user ID
TS	Timestamp
sk_{ID}	Private key of ID, $s \cdot H(ID)$
pk_{ID}	Public key of ID, $H(ID)$
tsk_{ID}	Temporal private key of ID
$sign_K(m)$	Sign a message m using private key K
$tsig_{ID}$	Signature of IDusing tsk
sig_{ID}	Signature of ID
e_{pk_ID}	Encryption using Public Key pk_{ID}
U_{ID}	USIM of an identity ID
ME_{ID}	Mobile Equipment of ID
REQ	Trust delegated key request
RES	Trust delegated key response
REQ	Registration request of r_{ID} to server

Table 5.2: USIM Request Message Type

Type	Input			Output	
$REQ1$	ID, TS	-	-	sig	e_{ID}
$REQ2$	ID, TS	sig	e_{ID}	sig, tsk	e_{ID}
$REQ3$	-	sig	e_{ID}	tsk	-

P.5. After verifying sig_A with the A's public key pk_A generated by A's ID, U_B decrypts e_A and obtain r_A. U_B then generates a random nonce r_B and compute e_B, sig_B and tsk_B, where $e_B = e_{pk_A}(r_B)$, $sig_B = sign_{sk_B}(r_B||TS_A)$ and $tsk_B = t \cdot sk_B$, respectively. We can compute $t = (r_A \oplus r_B)$, where $\oplus$ denotes the arbitrary operation of two inputs.

P.6. U_B returns e_B, sig_B and tsk_B to ME_B.

P.7. ME_B sends RES, e_B, and sig_B to ME_A.

P.8. ME_A sends $REQ3$, e_B and sig_B to U_A.

P.9. U_A verifies sig_B and decrypts e_B to obtain r_B. U_A then generates $tsk_A = t \cdot sk_A$. After that U_A returns tsk_A to ME_A.

After that ME_A stores tsk_A and ME_B stores tsk_B. With tsk_A and tsk_B, ME_A and ME_B can operate secure computation without revealing original sk_A and sk_B. Overall procedures are shown in Figure 5.3.

After the authentication procedures are completed, ME_A generates the shared session key $K_A = e(tsk_A, pk_B)$, while ME_B generates $K_B = e(pk_A, tsk_B)$. The correctness of $K_A = K_B$ is as $K_A = e(tsk_A, pk_B) = e(t \cdot sH(A), H(B)) = e(H(A), H(B))^{t \cdot s} = e(H(A), t \cdot sH(B)) = e(pk_A, tsk_B) = K_B$.

For the general mobile communication networks, the subscribers request the communication to the mobile access point that is linked to the servers. For the practical application of trust delegation model, we apply our design to the 3GPP generic authentication architecture [75, 74] in section 5.2.3.

5.2.3 Enhanced Generic Authentication Architecture with Trust Delegation

In this section, we show the enhanced design of GAA that applies PKI. Scheme 3 consists of three phases: *temporary private key generation*, *bootstrapping procedure*, and *service request to NAF*.

Phase 1: Temporary Private Key Generation

TD.1. ME_A sends ME_REQ, TS_A to the U_A, where ME_REQ is the request of tsk_A and TS_A is the timestamp generated by ME_A.

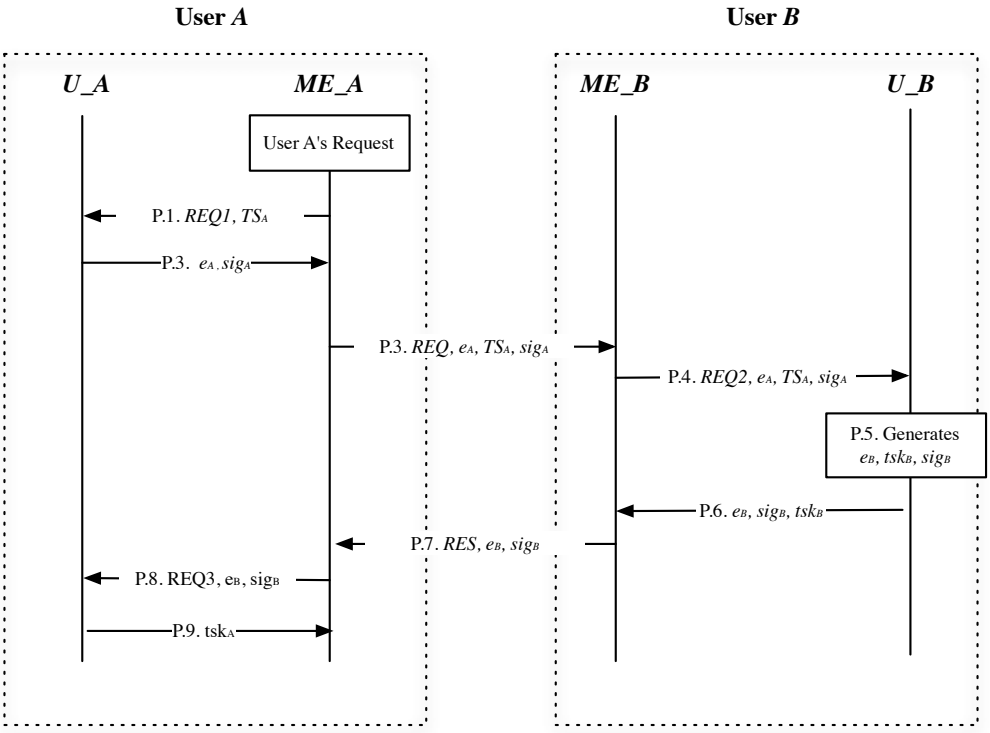


Figure 5.3: Scheme 1: Session Key Establishment Between Peer Entities A and B

TD.2. U_A generates r_A and computes e_A as in Section 5.2.2. U_A then returns e_A, tsk_A and sig_A to ME_A, where $tsk_A = r_A \cdot sk_A$ and $sig_A = sign_{sk_A}(e_A || \ TS_A)$.

After the phase 1 is completed, ME_A stores tsk_A, e_A, TS_A and sig_A.

Phase 2: Bootstrapping Procedure

If there is no shared information with NAF, ME_A has to contact BSF.

GB.1. ME_A sends ID of A, $tsig_A$, e_A, TS_A, and sig_A with bootstrapping request (BSF_REQ) to BSF, where $tsig_A = sign_{tsk_A}(BSF\_REQ)$.

GB.2. BSF generates $pk_A = H(A)$ for the verification of sig_A. After verifying sig_A, BSF can retrieve r_A by decrypting e_A and check the validity of tsk_A. If tsk_A is valid, BSF can verify $tsig_A$. After the successful verification, BSF stores r_A and TS_A with the ID of ME_A, and sends the response BSF_RES with corresponding signature back.

Phase 3: Service Request to NAF

After Phase 2, ME_A requests the service to NAF, then NAF authenticates ME_A as following procedures.

NF.1. ME_A sends NAF_REQ, $APPL\_ID$ and $tsig'_A$ to the NAF, where NAF_REQ is the request of application service and $APPL\_ID$ is the application ID. $tsig'_A$ is the signature where $tsig'_A = sign_{tsk_A}(NAF\_REQ \ ||APPL\_ID)$.

NF.2. If NAF has already authorized tsk_A, NAF instantly verifies $tsig'_A$. In other case, NAF requests BSF the authentication information of A. We assume that NAF and BSF have the secure channel.

NF.3. BSF returns r_A and TS_A those are used for NAF to verify $tsig'_A$. NAF stores r_A until TS_A is expired.

NF.4. NAF generates pk_A and verifies $tsig'_A$. When $tsig'_A$ is valid, NAF authenticates ME_A and provides its service to ME_A. Overall procedures are shown in Figure 5.4.

5.2.4 More Simplified Enhanced Generic Authentication Architecture with Trust Delegation

Since IDBC does not request the public key management, we can also reduce the BSF involvement for the authentication procedures. Thus, we can simplify the step **S-NF.1** and **S-NF.2** as follows. The overall simplified procedures are shown in Figure 5.5.

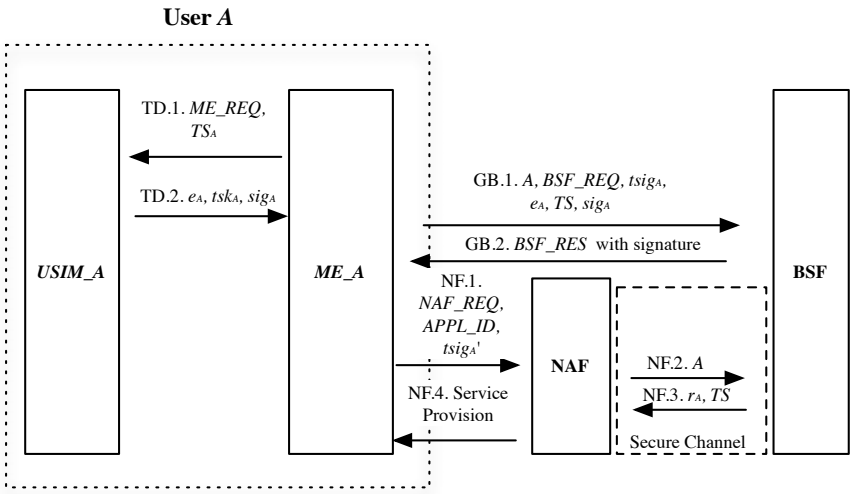


Figure 5.4: Public key based GAA with Trust Delegation. BSF involved for the compatibility

S-NF.1. ME sends NAF_REQ, ID of A, e_A, TS_A, sig_A, $APPL\_ID$, and $tsig_A$ to NAF for requesting the service $APPL\_ID$.

S-NF.3. After generating pk_A, NAF verifies sig_A and compute $tsig_A$ by decrypting e_A in sequence. When $tsig_A$ is valid, NAF authenticates ME_A and provides its service to ME_A.

5.2.5 Design Analysis

In this section, we briefly analyze our proposed model and compare with 3GPP generic authentication architecture. Section 5.2.5 shows the security analysis, and section 5.2.5 shows the performance analysis.

Security Analysis

For the analysis of our design, we define the attack scenarios as follows: the impersonation by malicious adversaries, the private key leakage by a compromised ME, and the weaken strength from temporary private key.

For the impersonation, an adversary may resend REQ, e_A, TS_A, and sig_A (**P.3**) or RES, e_B, and sig_B (**P.7**). However, the adversary should be able to manipulate the fake

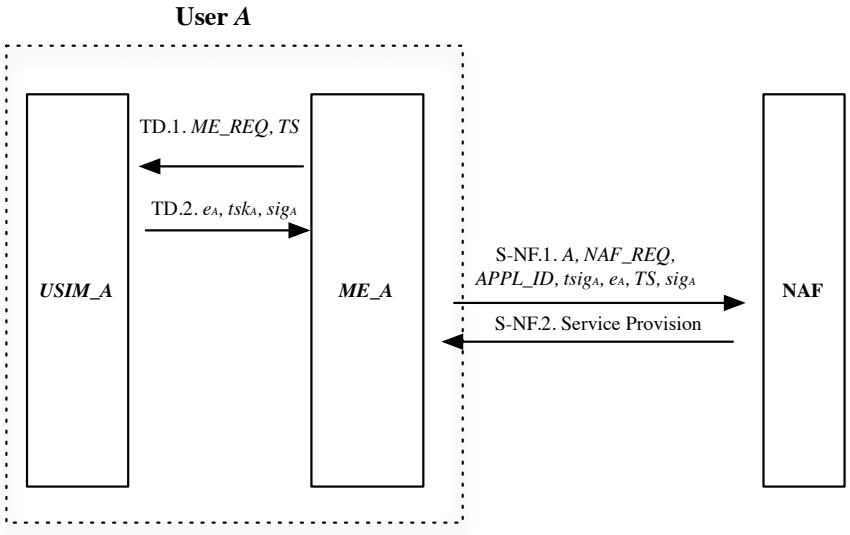


Figure 5.5: NAF directly authenticates the mobile device for request of services without BSF

sig_A and sig_B without knowing sk_A or sk_B.

For the case of compromised ME, since the tsk, e_A and $H(A)$ is known to ME_A, the compromised ME_A tries to compute $sH(A)$ with e_A, $H(A)$ and $r_A \cdot sH(A)$. However, compromised ME_A fails to retrieve the original private key without any information of r_A. Recall the hard problems in section 5.1, we can know such trial has the same success probability of solving DLP. Even though compromised ME_A sends tsk_A to other adversary, the adversary fails to impersonate after TS is expired. Thus, even ME is compromised, sk is still secure in USIM and the effect on tsk in ME is limited. Since we already assume that the private key in the USIM is stored in secure, the security of the USIM is considered as the security of the security storage of the USIM and the out of focus in this thesis.

Also, comparing the session keys derived from the initial private key sk and the temporary private key tsk, a session key using sk is computed as $e(sk_A, pk_B)$, while a session key using tsk is computed as $e(t \cdot sk_A, pk_B)$. It is trivial that the both have the same security strength.

Finally, if tsk is erased in ME, new tsk can be simply generated by choosing new random nonce r'_A. Thus, our design is resilient to not only key exposure problem, but

also key erase problem.

Performance Analysis

Computational Overhead Our design reduces the overall computational overhead in USIM comparing the case that the private key is stored in USIM (Figure 2.10 (a)) that all public key based security computations are operated in USIM. In Scheme 1 in section 5.2.2, the computations in the USIM are one hash computation to generate the public key, one point multiplication to generate the temporary private key and the signature generation of random nonce for the temporary private key. Because USIM is the only trusted entity, signature generation and verification in USIM are inevitable. The computational overhead of hash function generation is negligible.

Finally, our design does not require public key based security computation after the initial signature generation and verification, while the private key still remains in the secure storage. We do not count the computational overheads in ME that has the large computational power.

Transaction Overhead Our model shows about a half number of transaction than current 3GPP security architecture [74, 75], because our design is fully based on asymmetric key cryptosystem. Applying the IDBC, our design reduces the number of transaction to 4 rounds when we let NAF authenticate ME for itself (Section 5.2.4). The design supporting PKI [75] still requires the support of GBA [74] for the certificate management that requires 13 rounds of transaction, while our design does not have the overhead for PKI certificate management that eventually follows the use of the GBA. Thus, our model is resilient to the DoS attack that makes HSS or BSF unavailable [79].

5.3 Design and Implementation of One-way Key Agreement Model for Enhancing VoIP Internet Phone Security

The explosive usage of Internet based communication technologies has become more prevalent in recent years. Specifically, Session Initiation Protocol (SIP) based Voice over Internet Protocol (VoIP) is commonly presented for the Internet Phone. Current standards recommend Secure/Multipurpose Internet Mail Extensions (S/MIME) for securing the communication. However, S/MIME is likely to be too heavy for resource-constrained handsets

due to accompanying Public Key Infrastructure (PKI). The key agreement process is required for the shared session key that encrypts the conversation using the Internet phones. Nowadays, MIKEY [69] becomes one of the recommended solutions for the standard VoIP implementation to provide the shared link key generation. In the standard, the key is called as Traffic Encryption Key (TEK), and generated from shared TEK Generating Key (TGK). For generating TGK, several algorithms such as Diffie-Hellman key agreement or pre-sharing TGK are also recommended. Skype uses the proprietary key agreement model that each user generates 256-bit session key by exchanging their Identify Certificate and contributing 128 random bits in their security application [15][34]. Several studies such as [58][42] are proposed to provide more light-weight solution, they have the overhead of public key management though.

Ring et al.'s model [60] proposed the efficient model by applying identity based cryptography (IDBC) that uses the user's identity as the private key. Applying IDBC, their model does not require the public key management, and enables simplified process for the session key generation. However, the heavy cryptographic pairing computation results the call setup delay in generating TGK.

Our contributions in section 5.3 are as follows:

- We employed Okamoto *et al.*'s algorithm [57] that is the one-way key agreement model for reducing the call setup dely,
- also, combined with Hess's signature algorithm [31] for reducing overall SIP message sizes and the delay from initiating Secure Real-time Transport Protocol (SRTP).
- and, implemented the VoIP security service based on an open source Internet Phone called 'KPhone'.

Applying our protocol, the session key of caller side can be solely generated in the caller side. Thus, SRTP can be immediately initiated as soon as the response from the receiver arrived. Our novel design reduces delaying for the key generation and provides the explicit mutual authentication. In addition, the proposed approach reduces computational and communication overheads from public key management, signing number of messages by server and the SIP message sizes.

Section 5.3 is organized as follows: Section 5.3.1 shows the method to reduce the call setup delay in the secure communication using ID-based cryptosystem. We show the protocol in section 5.3.2 and the implementation result and the design analysis of our design in section 5.3.3.

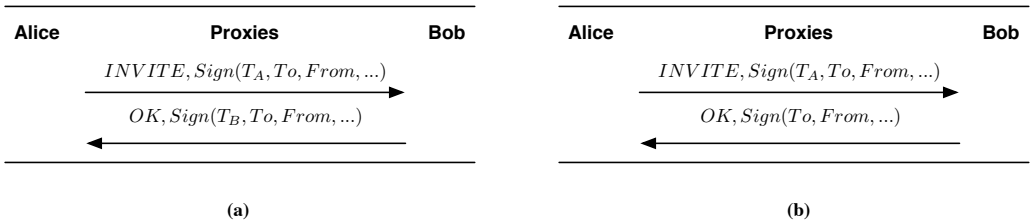


Figure 5.6: Key Agreement Model for SIP: (a) Ring et al.'s Model [60] (b) Our Propose Model

5.3.1 Reducing Call Setup Delay

In this section, we compare two approaches for key agreement of TGK between two entities. At first, assume two entities, Alice and Bob who exchange the key, where Alice requests the key exchange to Bob. Using two-pass method including Diffie-Hellman key agreement and Skype, Alice and Bob mutually exchange key generating information. Ring et al.'s model [60] is based on the two-pass key agreement protocol that is shown in Fig. 5.6 (a). Alice and Bob exchange key generation information T_A and T_B. Alice computes K_{AB} with T_B and Bob computes K_{BA} with T_A, where $K_{AB} = K_{BA}$. Using one-way method, Alice can request key agreement and send encrypted message using the session key to Bob at the same time, since only Alice sends T_A. In this model, the communication is required only once. To reduce the delay from computing the session key used for SRTP encryption, we use the one-way key agreement model. The example is shown in Fig. 5.6 (b).

The comparison of our one-way key agreement and two-pass key agreement [60] employing in VoIP is shown in Fig. 5.7. Using one-way key agreement, Alice can pre-compute the session key when she sends the INVITE message to Bob. When Alice and Bob agree with the session key and send SRTP transaction, they can reduce the delay, which is shown in two-pass model. In two-pass model, Alice can compute the session key after Bob responds with OK message. Thus, our model enables the immediate SRTP initiation after OK message is received while two-pass key agreement models have the delay from key generation.

5.3.2 Proposed Design

In this section, we describe the protocol design that we implemented in KPhone and analyze the security of the design. For the one-way key agreement protocol, we apply the

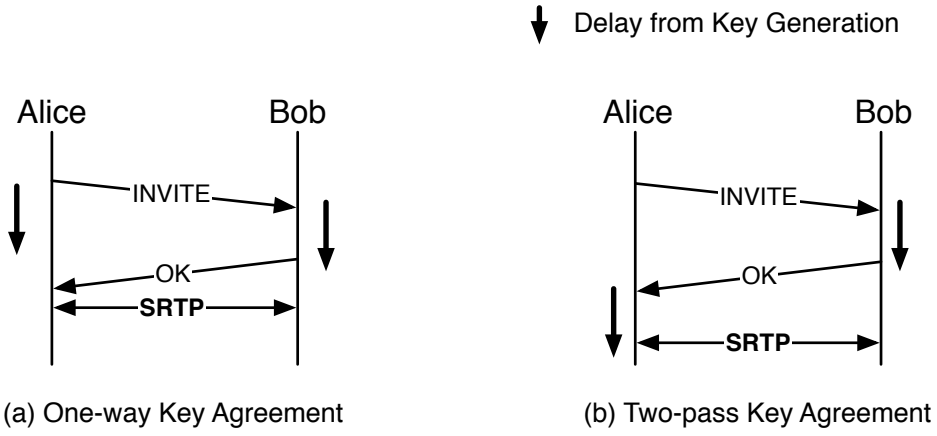


Figure 5.7: Comparison of (a) Ring *et al.*'s and (b) the Proposed Model

scheme 1 in [57] that is based on ID-based cryptosystem. We assume a caller Alice, a receiver Bob, and a server in a certain VoIP service. In order to generate SIP message, Alice generates r, t, v, and u, where $r = e(P_1, P)^k$, $t = H^*(r) \cdot H(ID_{Alice})$, $v = h(m, t)$, and $u = v \cdot d_{Alice} + k \cdot P_1$. Here $h : \{0,1\}^* \times G_1 \rightarrow (Z/lZ)^\times$, $H : \{0,1\}^* \rightarrow G_1$, and others follow [6]. k is randomly selected by Alice. To generate t, r should be transformed from elliptic curve to finite fields. H^* is a map-to-point hash function, where $H^* : G_2 \rightarrow \{0,1\}$. To compute with $H(ID_{Alice})$, the transformation is necessary. $e : G_1 \times G_1 \rightarrow G_2$. G_1 is a cyclic additive group, generated by P with order q. G_2 is a cyclic multiplicative group with the same prime order q. d_{Alice} denotes Alice's private key, $d_{Alice} = sH(ID_{Alice})$. m is the SIP message that are fixed SIP headers including the sender's address, the receiver's address, message generated time and other necessary information. Session Description Protocol (SDP) information are not signed. Alice also generates the session key as follows.

$$k_{AB} = e(d_{Alice}, H(ID_{Bob}))H^*(r) \oplus e(d_{Alice}, H(ID_{Bob})) \tag{5.1}$$

Then, Alice sends u, v to Bob, where $(u, v) \in (G, (Z/lZ))$. After receiving (u, v), Bob generates the following.

$$\begin{aligned} t &= H^*(r)H(ID_{Alice}) \\ &= H^*(e(u, P) \cdot e(H(ID_{Alice}), -sP)^v) \cdot H(ID_{Alice}) \end{aligned} \tag{5.2}$$

After that, Bob verifies u and v with computing Equation (5.2) using m and t. After that Bob generates the session key as follows.

$$k_{BA} = e(t, d_{Bob}) \oplus e(H(ID_{Alice}), d_{Bob}) \tag{5.3}$$

The results of equation (5.1) and (5.3) are same. Correctness of $k_{AB} = k_{BA}$ follows [57]. $\oplus$ is the additive operation in G_2. When a hash function $H' : G_2 \rightarrow \{0,1\}$ is used, $\oplus$ can be XOR operation in $k_{BA} = H'(e(t, d_{Bob})) \oplus H'(e(H(ID_{Alice}), d_{Bob}))$. K_{AB} is used as TGK, and TEK is generated using [8].

Thus, t is used for both SIP message signature and the key generation to reduce the additional communication only for the key generation.

5.3.3 Implementation and Analysis

Implementation

We implemented the one-way key agreement model based on the open source VoIP client, 'KPhone' (http://sourceforge.net/projects/kphone) for user terminal, 'SIP Express Router' of iptel.org (http://www.iptel.org/ser) for the SIP gateway that includes SIP registrar, SIP proxy, SIP redirection and SIP location server. We implemented our signing and key agreement protocol in 'S-INVITE' of call setup phase as in Fig. 5.8. Compare to previous designs, the caller does not have to wait the response '200 OK' for the key generation due to the one-way key agreement in our protocol.

The caller only sign the first indicator and fixed SIP header parts, since other parts such as SDP can be continually changed during the communication. The signature is attached after SDP parts as in Fig 5.9.

After receiving 'S-INVITE', the receiver generates the public key from the caller's ID '*109@220.69.191.100*' as in Fig. 5.10. Since ID of an entity is used to generate the public key in ID-based cryptosystem, there is no need to verify the public key of each entity. The receiver also finds u, v, and recovers t. The receiver also collects the fixed parameters from the initial SIP message and recovered t and generates the hashed output of two parts. And then the receiver compares the hashed output with the received v for integrity check as in Fig. 5.11. Finally, the receiver generates the TGK first and the traffic encryption key (TEK) with the method of [8] from the value t as in Fig. 5.12. Since caller already generated 163 bits TGK and 163 bits TEK, the receiver can directly use TEK for the secure communication.

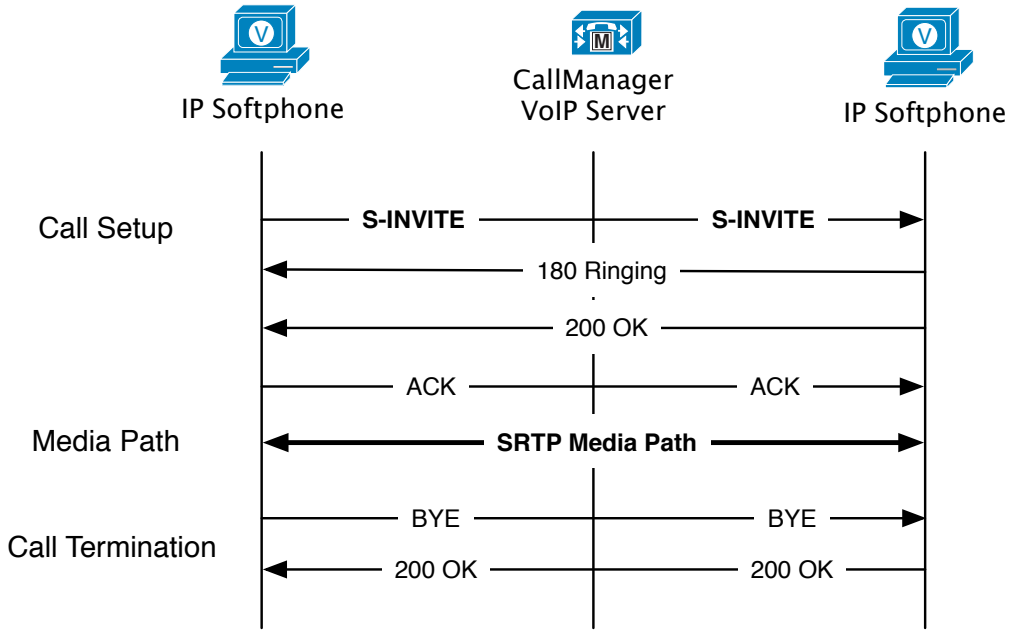


Figure 5.8: The overall Calling process. We implemented our protocol as 'S-INVITE' in call setup phase.

Performance Analysis

Our design requires one exponentiation operation in G_2, two hash operations, two multiplications in G_1 for the signature generation, and one exponentiation operation in G_2, two pairing operations, and one multiplication operation for the verification. When the several messages are sent by the same identity, the sender can reduce one pairing operation by pre-computing $e(H(ID), -sP)$. For the key agreement, one pairing operation of the sender, one multiplication over elliptic curve, one exponentiation operation, and two pairing operations of the receiver are required. The computation time for hash operation takes under 600 microseconds, and the generation of signature and key agreement takes approximately 26 milliseconds on the Intel Core2 Duo 2.0 GHz. Recent studies on optimal pairing implementations [19][29] show that the pairing computation takes approximately 3 seconds in smart card, and 14.5ms in Core2 Duo 1.66GHz. The open source cryptographic pairing library [48] showed the similar results.

Thus we can estimate that the paring computation in current smart phones would take approximately 150 - 300 milliseconds. Compare to Ring *et al.*'s and Diffie-Hellman

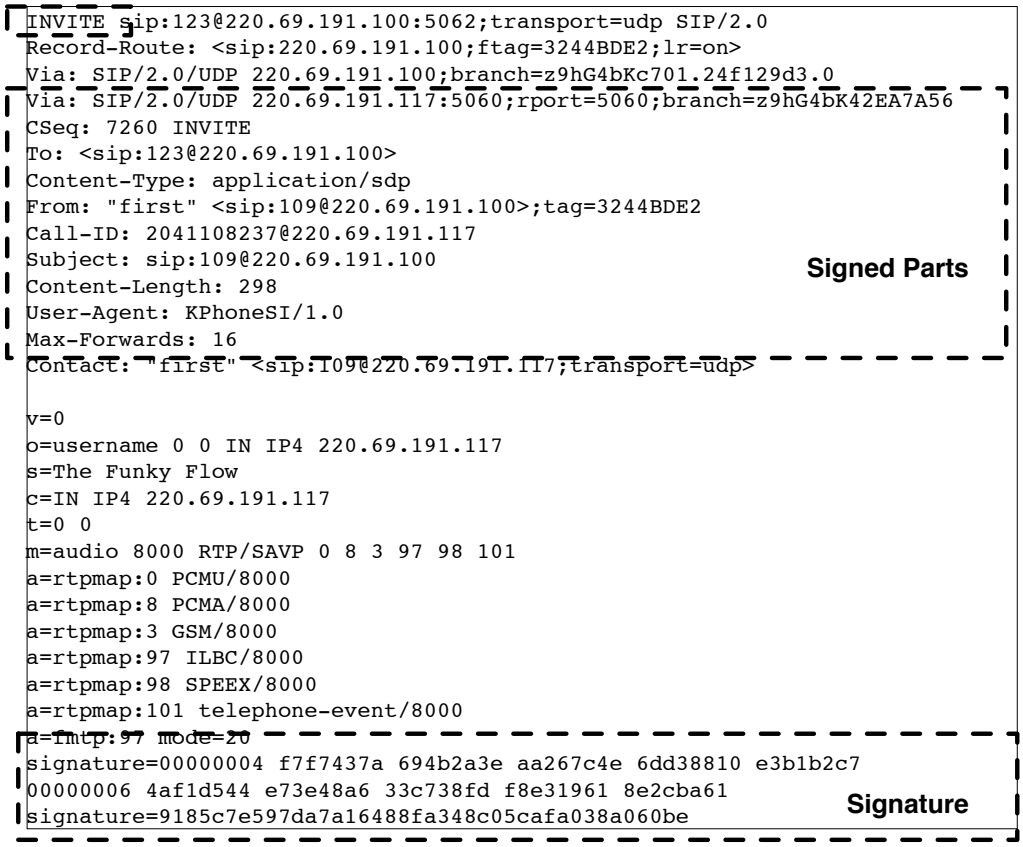


Figure 5.9: Generate Signature of SIP Message. (First 'INVITE' and SIP header parts)

Key agreement protocol, our protocol can immediately initiate SRTP encryption without the delay from verifying the signature of the receiver and generating TGK in caller side and. In this sense, our proposed design enables practical adaptation of IDBC in smart phones than previous models because VoIP usually spends one direction latency of about 150-200ms. Moreover, overall computation takes under 1 - 2 seconds that is suitable for practical VoIP service without occurring the noticeable delay.

Security Analysis

We analyze our design that holds security requirements defined in [57] as follows:

Known-key Security The caller Alice randomly choose P_1 and k in each session in order to generate r, where $r = e(P_1, P)^k$. The leakage of P_1 or k doesn't affect the previous

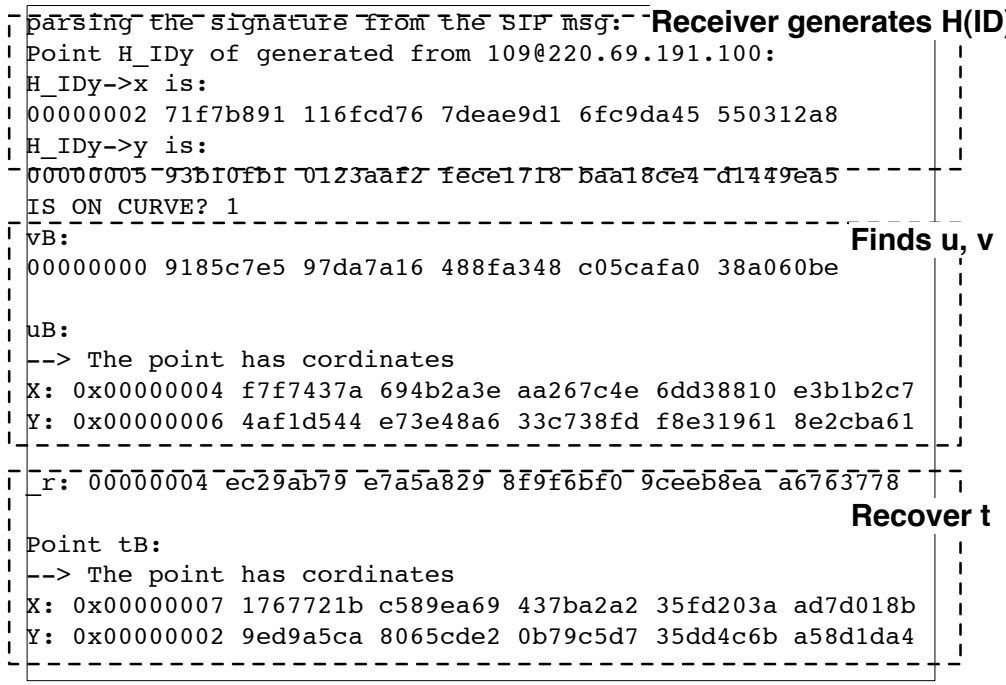


Figure 5.10: After receiving S-INVITE, Receiver generates H(ID) from caller ID, finds u, v and recovers t

session. Although an adversary $\mathcal{A}$ obtains P_1 and k, $\mathcal{A}$ cannot know the TGK used in the previous session.

Unknown Key-share In order to generate the session key, Bob firstly generates t. t is used to verify the signature of Alice. Also, Alice self-generates the session key without any information from Bob. Therefore, any other entities except Alice and Bob cannot exchange the key. To succeed the attack, the adversary should be able to generate the signature of Alice or know the private key of Bob.

Key Control Since Alice selects the key generating parameter, and the process is done in one-way, Bob cannot control the session key, also it is difficult for Alice to precompute the random integer r and the generator P_1 to control t.

Attacks to Sender When Alice's private key is leaked, the adversary can impersonate Alice, since r is known to Alice, while it is not possible to impersonate other entity.

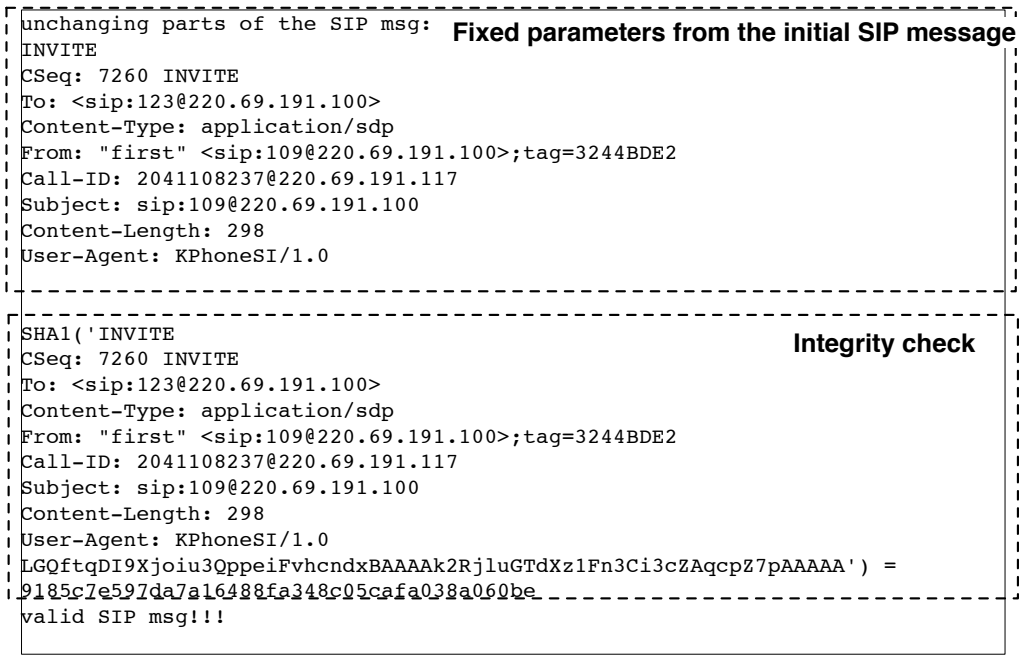


Figure 5.11: Integrity check of received message

Sender's forward security is guaranteed, since k and P_1 are randomly selected in each session by Alice.

Random number compromise The random integer r is easily known from (u, v). However it is difficult know Alice and Bob's private keys or session key from public parameters P, sP, and r. To attack the session key, the knowledge of Alice or Bob's private key is necessary. The success of attack with P, sP and r is the same as the success of attack on the signature.

5.4 Lawful Interception of Secure Communication based on ID Based Cryptosystem

In this section we design a new robust and feasible key escrow model for securing communications based on ID based cryptosystem (IDBC) that not only overcome the shortcomings of the previous key escrowing models for the lawful interception (LI) in the mobile networks, but also enable efficient update of a single private key that overcome the in-

Figure 5.12: Key Agreement with recovered t

herent threat of IDBC. Our new model also demonstrates the efficiency in the public key management.

Section 5.4 consists of six sections. Section 5.4.1 briefly shows the network architecture for LI and the concept of IDBC. Section 5.4.2 illustrates the existing key escrow model and addresses shortcomings of previous key escrow models. Section 5.4.3 describes proposed new scalable and efficient key escrow model. Section 5.4.4 analyzes of our protocol and compare with previous protocols.

5.4.1 Related Work

In this section, we show the brief of current standard LI architectures in mobile network, and describe the key escrowing models for the LI of secure communications.

Mobile Network Architecture for Lawful Interception

While the generic LI architecture is largely specified by ANSI, ETSI, 3GPP and etc., we briefly introduce the specification by 3GPP due to the similarity of the architectures. 3GPP specifies the requirements of the LI [70], the architectures and the functions [72], and the handover interface (HI) between the LEA and the mobile service operators (MO) [73].

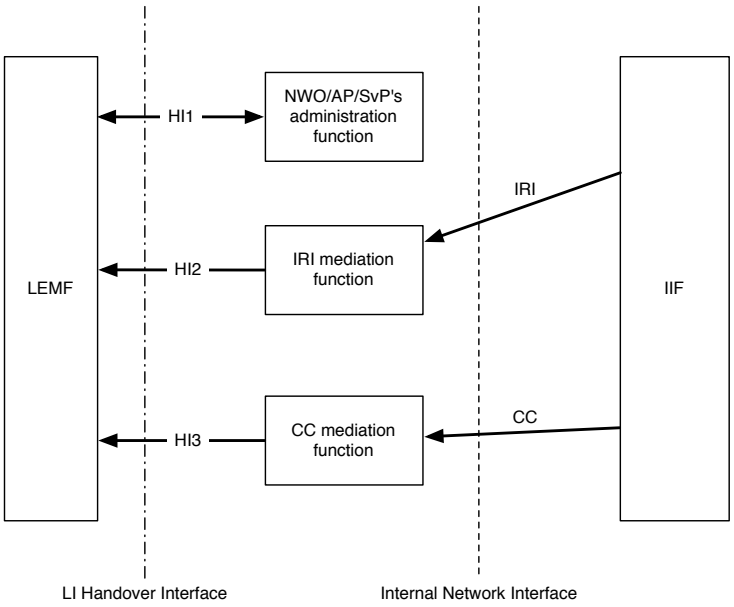


Figure 5.13: Architecture for the Lawful Interception by 3GPP

Fig. 5.13 shows the several HIs that link the law enforcement management function (LEMF) of the LEA to the internal network interception function (IIF) of the MO. We can assume the HI as the secure channel. Each HI is defined to send the following informations:

- HI1: administrative information
- HI2: intercept related information (IRI)
- HI3: the content of communication (CC)

The administrative function in HI1 includes the network management function. Both IRI and CC are sent via the IIF of the MO. The LEA manages the LEMF that gathers and analyzes both IRI and CC. IRI is coded using ASN.1 and Basic Encoding Rules, and transmitted from IIF of the MO to LEMF via HI2.

When the LEA requests the LI of the secure communication to the MO via HI1, the MO may provide the proper decryption method (the escrowed keys) via HI2 and the encrypted communication via HI3.

Key Escrowing for the Lawful Interception of the Secure Communication

In the current symmetric key based security architecture [72], the MO also has the role of the escrow agency (EA) that provides the session keys with short lifetimes for the LEA. Thus most studies on the key escrowing are for the public key infrastructure (PKI) based secure one-way communications such as secure e-mail.

The key escrow model for the PKI based secure communication consists of the EA and the LEA: the EA stores user's private keys, the LEA requests the private keys for the purpose of the LI under the legal permission of the court. Brief LI procedures are as follows: When the users initiate the secure communication $\mathcal{C}$, the LEA are granted LI of $\mathcal{C}$. Than the LEA request the escrowed key to the EA, and the EA provides the key to the LEA. Finally the LEA disclosure the information of $\mathcal{C}$.

Since there are the potential vulnerability that the malicious behavior of the EA or the LEA, most studies focused on the limiting the capability of the EA and the LEA. Micali [51] proposed that a user divides his private key into several pieces and register to several EAs in order to limit the capability of the single EA. Therefore, the initial key can only be recovered when all EAs agree on the key recovery. Shamir [67] proposed the partial key escrow method that requires sufficient time consumption to protect the incident misuse from the malicious EA. However, the such method requests the large overhead that conflicts with the LI requirements [70]. Also, Jefferies *et al.* [36] proposed the warrant bound to limit the duration of the lawful interception of the LEA in order to prevent the malicious behavior of the LEA. Verheul *et al.* [82] proposed fraud detectability while Frankel *et al.* [26] introduced compliance certification.

Abe and Kanda's Key Escrow Model

In 2002, Abe and Kanda [1] defined the requirements and the properties for the key escrowing and proposed PKI based key escrow algorithm for the one-way communication that allows the limited permission period. Their protocol consists of the registration phase, the communication phase and the disclosure phase as follows:

Registration A user u generates public key pairs (x_{ui}, y_{ui}) for $i = 0, ..., t$ and sends to the EA. $x_{ui} \in_R \mathcal{Z}_q$ is the private key randomly chosen by u and $y_{ui} := g^{x_{ui}}$ is the corresponding public key, where g is the generator of $\mathcal{G}_q$, a multiplicative subgroup of order q in Z_p . After verifying the keys, the EA stores everything received.

Communication u initiates the secure communication $\mathcal{C}_{u\tau}$ using $x_{u\tau}$ or $y_{u\tau}$, where τ is the target term wherein monitoring is approved.

Disclosure the LEA is granted LI of u and disclose $\mathcal{C}_{u\tau}$ within the warrant (the user u and the term τ).

However, each subscriber has to participate in the key escrowing that conflict with the LI requirements that the subscriber shall not recognize whether they are under surveillance.

5.4.2 Shortcoming on Previous Key Escrow Models

Conflicting with Requirements of Lawful Interception

In some previous key escrow models [1], any subscribers self-generate their public key pairs and register to the key escrow agency. However, the subscriber's participation in the key escrow procedures fundamentally conflicts with the requirements of the LI that the subscribers never recognize whether they are under surveillance and their communications are intercepted.

Warrant Bound of Law Enforcement Agency

In order to provide the proper decryption method, the mobile service providers escrow subscriber's key to the key escrow agency and send escrowed keys for the request of the LEA. Providing the symmetric session key for the secure two-pass communication such as voice conversation has less complication due to the short lifetime of the key that expires after the session is closed.

On the contrary, the private key should be sent to the LEA for the LI of the one-way communication such as secure e-mail. Due to the lifetime of the public key pair (the public key and the private key) is much longer than that of the symmetric session key, the LEA might be able to illegally eavesdrop the subscriber's communication after the permission is expired, if the public key pair is not updated. For example, the permission terms on the LI may be at least several days while the life of public key is about a year in general. Even though a few models such as [1] overcome such a problem, they require the participation of the subscriber that conflict with the requirements of the LI. Moreover, [1] only supports the LI of one-way communication.

Overhead for the Network

Existing mobile networks such as 3GPP, the security architecture based on the symmetric key cryptosystem is widely adopted in the market due to the performance efficiency. Thus, implementing the previous public key escrow models such as [1, 51, 67, 82] requires large

overheads on the public key management and the large key storage from the non-standard architecture for each key escrow model. Thus, the complex network facilities that increase the overall cost of the networks are required.

Applying IDBC, such overheads from the public key management are not required, due to the LEA can self-generate the private key of each subscriber with the escrowed master key.

Security Threats on Inherient Property of IDBC

However, only depending on the inherent property of IDBC has the potential security threat that the LEA can illegally eavesdrop all the communication in the network. When the EA provides the master key s to the LEA for the LI of A, the LEA can generates the private key of A, $sH(ID_A)$ using the publicly known hash function $H : Z_P^* \rightarrow G$ and A's ID, ID_A. However, the LEA can also compute $sH(ID_C)$ to eavesdrop C without legal permission if the domain master key s is not updated. Moreover, the key update of a single subscriber is not available in IDBC. Once the private key of a subscriber is compromised or known to the LEA, all keys of all subscribers must be updated.

Although several studies such as [32] prevents key escrowing, they cannot be used for the LI from the requirements [70] since they disabled the key escrow property of IDBC.

5.4.3 Proposed Key Escrow Model

In this section we propose our key escrow model for IDBC based secure communication that overcome the shortcomings shown in Section 5.4.2. We define following entities in our model:

Law Enforcement Agency (LEA) : The LEA requests the content of the communication and receives the intercept related information and content from MO under the law.

Mobile Service Provider (MO) : MO offers the mobile communication service including the encryption to subscribers, and provides the proper decryption method and interception related information for the request of the LEA.

Key Generation Center (KGC) : KGC provides keys for encryption to MO, subscriber and the LEA. It also provides the subscriber's key for the request of LEA. (Note that KGC is also the escrow agency.)

Subscriber : Subscriber uses the mobile communication service, and receives the encryption key from KGC.

We assume that the LEA may illegally intercept the secure communication of the subscriber over the warrant bound.

Security Requirements

The requirements of key escrowing are defined in [1]. Designing the new key escrow model based on the IDBC, We inherent the requirements and define following requirements.

- **Non Subscriber participation**: It shall not be recognized by a subscriber in escrowing and providing subscriber's key to the LEA.
- **Warrant Bounds**: It shall be available to limit the duration of the permission for the lawful interception by the LEA.
- **Key Escrow Efficiency**: It shall not consume large overhead for providing the key to the LEA.
- **Off-line KGC**: When the law enforcement agency obtains the private key or the necessary information for decryption, it should be able to intercept the communication without help of key escrow agency.

Overall Key Escrow Protocol

In this section, we show the key escrow model that enables the LEA to intercept any kinds of secure communications between two subscribers A and B. Section 5.4.3 shows the LI for the two-pass communication, and section 5.4.3 illustrates the LI for the one-way communication. The symbol '**I**' denotes the interception procedures while the symbol '**S**' denotes the communication procedures between subscribers.

For the pre-procedure, the KGC initially operates the key distribution process as in section 5.1.2. Thus, we assume that A already stores $sH(ID_A)$ as the private key and the $H(ID_A)$ as the public key, while B stores $sH(ID_B)$ and $H(ID_B)$. We also assume the shared key k_A between MO and A, and k_B between MO and B exist.

We assume that the LEA requests the LI of A to the MO.

LI For Two-pass Communication

Let A initiates the secure communication with B and the LEA are on the surveillance of A.

I.1. The LEA requests the KGC and the MO for the lawful interception of B via HI1.

S.1. A generates the random integer r_A and the corresponding signature $sign_A(r_A)$. A encrypts them with the shared key with MO, k_A and sends u_1 to the MO.

$$u_1 = e_{k_A}(A||B||r_A||sign_A(r_A)) \tag{5.4}$$

The symbol of $e_k(M)$ denotes encryption function and $sign$ is a signature function. Suffixes of each function denote the owner of the key used for the encryption or signing. For example, e_{k_A} denotes the encryption with the shared key between A and MO and $sign_A$ denotes signature with A's private key. $||$ denotes concatenation.

S.2. After decrypting u_1, MO verifies r_A with the signature $sign_A(r_A)$. And then MO encrypts them using k_B and sends u_2 to B.

$$u_2 = e_{k_B}(A||B||r_A||sign_A(r_A)) \tag{5.5}$$

If MO includes the signature, MO sends u_2^* to B.

$$u_2^* = e_{k_B}(A||B||r_A||sign_A(r_A)||sign_{MO}(r_A||sign_A(r_A))) \tag{5.6}$$

S.3. After decrypting u_2, B verifies r_A with $sign_A(r_A$, and selects another random nonce r_B. Then B generates the signature of r_B, $sign_B(r_B)$, and sends u_3 to the MO. And then, B computes $v = devf(r_A, r_B)$, where $devf$ is a function from the input r_A and r_B, implies the general computation including $+$ or $\times$.

$$u_3 = e_{k_B}(B||A||r_B||sign_B(r_B)) \tag{5.7}$$

S.4. MO decrypts u_3 and verifies r_B with $sign_B(r_B)$. Then MO generates u_4 and sends it to A.

$$u_4 = e_{k_A}(B||A||r_B||sign_B(r_B)) \tag{5.8}$$

S.5. A computes $v = devf(r_A, r_B)$. MO also computes v.

I.2. MO sends v with A's ID and the request of the LI to the KGC.

I.3. The KGC sends $vsH(ID_A)$ to the LEA via HI2. $vsH(ID_A)$ denotes multiplication of v and $sH(ID_A)$.

I.4. MO sends the IRI via HI2 and the CC to the LEA via HI3, as in Section 5.4.1.

We assume a key agreement protocol between A and B as following: A computes $k_{AB} = e(vsH(ID_A), H(ID_B))$, while B computes $k_{BA} = e(H(ID_A), vsH(ID_B))$, where $e : G_1 \times G_1 \rightarrow G_2$ is bilinear pairing function. The correctness of two equations can be shown from following equation.

$$k_{AB} = e(vsH(ID_A), H(ID_B)) = e(H(ID_A), H(ID_B))^{vs} = e(H(ID_A), vsH(ID_B)) = k_{BA}$$

The LEA can compute $H(ID_A)$ and $H(ID_B)$ with public hash function $H : Z \rightarrow P$, and each subscriber's identity ID_A and ID_B. Also with $vsH(ID_A)$, the LEA can compute k_{AB} for decrypting the secure communication between A and B. k_{AB} is used as the session key between A and B.

Fig. 5.14 depicts overall process of the LI for two-pass communication.

LI For One-way Communication

In this section, we show the model for one-way communication such as e-mail. Let A generates an e-mail message M_A and securely sends it to B. Most steps are as same as the case for two-pass communication, and we only describe the differences.

S.1'. A generates the random integer r_A and the corresponding signature $sign_A(r_A)$. A also encrypts the message M_A with the temporary public key of B, $r_A H(ID_B)$, which is denoted as $Enc_B(M_A)$. In this case, only r_A is used due to the one-way communication from A to B. After that A encrypts them with the shared key with MO, k_A and sends u_1 to the MO.

$$u_1 = e_{k_A}(A||B||Enc_B(M_A)||r_A||sign_A(r_A)) \tag{5.9}$$

S.2'. After decrypting u_1, MO verifies r_A with the signature $sign_A(r_A)$. Then MO encrypts them and send B the following.

$$u_2 = e_{k_B}(A||B||Enc_B(M_A)||r_A||sign_A(r_A)) \tag{5.10}$$

S.3'. B decrypts u_2 and verifies r_A with $sign_A(r_A)$. After that B generates $r_A sH(ID_B)$ and decrypts $Enc_B(M_A)$.

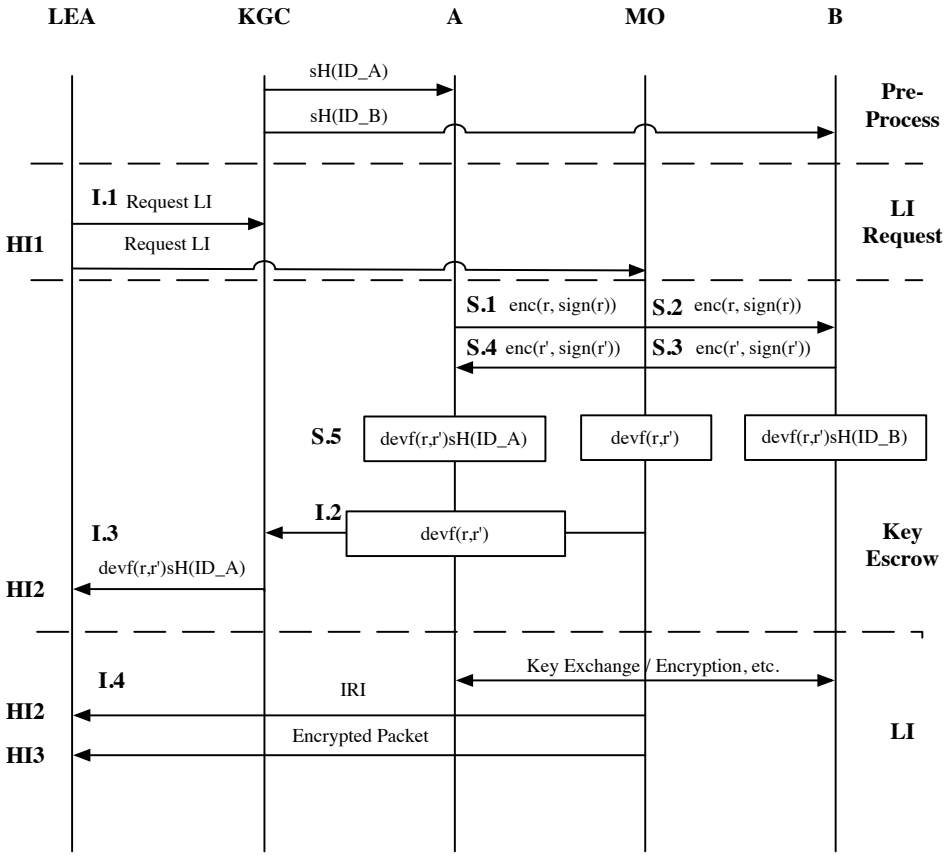


Figure 5.14: LI Procedures for Two-pass Communication

I.2'. MO sends r_A with B's ID and the request of lawful interception to the KGC.

I.3'. The KGC sends $r_A sH(ID_B)$ to the LEA via HI2.

I.4'. MO sends the IRI and the CC to the LEA via HI2 and HI3.

The LEA with $r_A sH(ID_B)$ cannot extract s even though the LEA obtains $sH(ID_A)$ or $sH(ID_B)$ with r_A from the subscriber A or B from the computational hardness of EC DLP [9]. In case of A receives the secure e-mail from any entities, the KGC sends $rsH(ID_A)$ to the LEA in **I.3'**, where r is a random nonce.

5.4.4 Design Analysis

Security Analysis

In this section, we briefly show that our model satisfies the requirements of the lawful interception as follows.

Non Subscriber Participation : In our protocol, subscribers do not participate in the key escrowing and recognize whether their communication is under surveillance.

Warrant bounds : The nonce r_A and r_B are randomly selected in each session to prevent the replay attack due to checking the freshness of nonce. The private key of the subscriber provided to the LEA is also different in each session. Consequently, the LEA fails to eavesdrop the communications in the unauthorized session.

Key Escrow Efficiency While the generic PKI based key escrow models require that the KGC stores the large number of public key pairs, the KGC in our protocol only stores one master key.

Off-line KGC : The key escrow agency only involved in the LI during the key escrowing. After that the LEA could directly intercepts the secure communication via the mobile network operator [73], and reveal the information under surveillance.

Our protocol also guarantees key escrow requirements in [1] such as the 'admissibility' that the LEA verify the message from the subscriber, the 'fraud detectability' that the LEA can verify the signature of random r_A and r_B for checking the freshness, and the 'sender Authentication' that the LEA authenticate the sender from the public key of $H(ID)$.

We also show the security strength of our protocol when the LEA illegally eavesdrops the secure communication. If the LEA tries to intercept the communication without permission, then the LEA will not receive any support from MO and KGC. We could consider followings attack scenarios: The LEA attempts the unauthorized interception (eavesdropping) without the legal permission. The LEA attempts the interception using v after the permission is expired. The LEA colludes with the user A or the user B to retrieve the master key.

Case 1: The LEA attempts the unauthorized interception without any legal permission

Assume that the LEA intercepts the encrypted communication between the users A and B. The eavesdropping processes are as follows:

1. The user A sends the random number r_A and the signature to the server.
2. The server verifies r_A and the signature of A, and sends r_A and the signature to B.
3. After verifying r_A, B generates the random number r_B and the signature of B, and sends them to the server.
4. The server verifies r_B and the signature of B and sends them to A.
5. The LEA attempts to eavesdrop the secure communication.

Let the key agreement protocol between A and B. A computes $e(vsH(ID_A), H(ID_B))$ while B computes $e(H(ID_A), vsH(ID_B))$. In this case, the LEA has no information of the secret parameter s that is necessary to compute $e(vsH(ID_A)$ and $H(ID_B))$. Thus, the LEA cannot know any information of the session key between A and B and cannot decrypt the encrypted packet from the illegal eavesdropping. The LEA also fails on the attack without knowing r_As for the one-way communication.

Case 2: The LEA attempts the interception using v after the permission is expired.

Assume that the LEA tries the unauthorized interception with expired $vsH(ID_A)$ as follows: A and B begin another secure communication with a new session.

1. User A sends the random number n_A and the signature to the server.
2. The server verifies n_A and the signature from A and sends them to B.

3. After verifying n_A, B generates the random number n_B and the signature, and sends them to the server.

4. The server verifies n_B and the signature, and sends them to A.

5. The LEA attempts to eavesdrop the communication with the expired $vsH(ID_A)$.

Assume the key agreement protocol between A and B as follows: Both A and B computes $v' = devf(n_A, n_B)$. After that A computes $e(v'sH(ID_A), H(ID_B))$ while B computes $e(H(ID_A), v'sH(ID_B))$. In this case, the LEA cannot know $v'sH(ID_A)$ from $vsH(ID_a)$. Thus, the LEA has no information of the session between A and B, and cannot decrypt the encrypted packet from the packet sniffing. The security of $v'sH(ID_A)$ is based on the computational infeasibility of ECDLP [9].

Case 3: The LEA colludes with the user A or the user B.

Assume the LEA has $vsH(ID_A)$ and get v and $sH(ID_A)$ from the colluded user. The LEA may try to retrieve s from $vsH(ID_A)$. However, knowing s from $vsH(ID_A)$, v, and $sH(ID_A)$ (or $sH(ID_A)$) has the same computational infeasibility of ECDLP.

5.4.5 Comparisons

In this section, we compare our proposed protocol with the symmetric key based model, Abe-Kanda's model [1]. The symmetric key based model only partially satisfies 'Warrant Bound' with short lifetime of the key. Abe-Kanda's model does not satisfy 'Non subscriber participation' due to the subscriber self generates n number of partial public keys and register them to the escrow agencies. Our protocol is more efficient than previous models because our model requires only one key, optionally one additional symmetric key, whereas [1] requires $t+1$ number of secret keys, where t is the threshold. Moreover, our protocol provides the LI of both two-pass communication and one-way communication, whereas the symmetric key based model only provides the LI of two-pass communication, and [1] only provides the LI of the one-way communication. Finally, our protocol can be widely used in combination of other key agreement protocols such as Diffie-Hellman protocols, whereas [1] only used with their own protocol.

Table 5.3 shows the comparison of our proposed protocols with the symmetric key based model and Abe-kanda's model.

Table 5.3: Comparison of Key Escrow Models

	Symmetric Key Model	Abe-Kanda [1]	Proposed
Warrant bounds	X	O	O
One-way Comm.	X	O	O
Two-pass Comm.	O	X	O
Non Subscriber Participation	O	X	O
Efficiency (Number of Keys)	1	$t+1$	1
Scalability	X	X	O

6. Conclusion and Further Work

In the thesis, we designed security protocols for the advanced wireless sensor networks and mobile networks. We designed the efficient sensor node authentication protocols for the dynamic sensor network environments. While secure and efficient interworking of several different networks is the important issue in the next generation convergence network, we also designed the converged protocol that efficiently authenticates the sensor node via mobile networks. For the request of the public key based security architecture in the mobile networks, we deployed the ID based cryptosystem (IDBC) into the security architecture of the next generation mobile networks. We designed the IDBC based application security architecture and the security protocol for Voice over IP and the lawful interception.

The contributions of the thesis are as follows:

- We claimed the drawbacks of previous authentication protocols supporting mobile nodes in WSN, and identified following requirements: efficient node re-authentication and untraceability. And then, we proposed our novel efficient node authentication and key distribution protocol that provides re-authentication and untraceability. Also, we analyzed our protocol by comparing with the previous protocols. Our protocol requires only three passes of communication with one third of communication message sizes compared with previous protocols in node re-authentication. The computational overhead of node re-authentication of a single mobile node achieves about 2-3 times efficiency than that of initial node authentication.

- For the forthcoming advanced sensor technologies, we also we proposed our novel efficient node authenticated key agreement protocol for dynamic WSN combining symmetric key base model with PKI based model. We introduced the concept of '*Neighbor Sink List*' for the real environments that the nodes are irregularly distributed. Analyzing our protocol, we achieved about 8 times more energy efficiency on computation than PKI based approaches and about 50 percents of energy efficiency on communication than symmetric key cryptosystem based approaches.

- We proposed an efficient authentication and key exchange protocol for the 3G-WSN network by integrating WSN into 3G network as the application. While most communications are operated under the mobile network, the communication in the sensor

network is minimized than previous work. When the hop distance between end-to-end nodes are five in the sensor network, energy cost in the sensor network applying our proposed design is estimated to be dropped by about 90 percent than previous models.

- We described public key management issues in the mobile networks and proposed 'trust delegation' concept based on IDBC that enables multiple security applications simultaneously, and is resilient against not only the key exposure but also the key loss. Reducing the number of transactions as well as involved entities such as HSS and BSF, our design is resilient to the DoS attack targeting HSS or BSF.

- We proposed the efficient authentication and key agreement model for VoIP security service and shows the implementation based on the open source VoIP client. Since the cryptographic pairing computation in ID based cryptography (IDBC) is still heavy, the verification of the response message and the key generation in the caller side is the cause of the call setup delay. Our proposed model enables the practical application of IDBC in the VoIP by deploying one-way key agreement model. By Pre-computing the secret key in the caller side, our proposed model significantly reduces the call setup delay. Through the performance, design and security analysis, we can see that the proposed approach decreases not only the cost for the public key management but also additional process for the key generation with using the parameter for the signature verification and reducing the delay for the initiating SRTP media path.

- We proposed the robust key escrow protocol that enables the lawful interception for the secure communication based on IDBC. Providing the warrant bound, our protocol overcomes the security threats from the inherent key escrow property of IDBC, also satisfies the requirements of the requirements in [70] whereas the most previous key escrow models do not meet the requirements. Based on the comparison of key escrow models, we can show that the proposed protocol provides the scalable and efficient key escrowing for both two-pass and one-way secure communication in mobile networks.

Our future plan is gaining the energy efficiency of sensor network in the initial authentication process of our protocol. Thus, We expect our proposed protocol will be the efficient security solution supporting the mobile node in WSN. We also consider deploying the proposed architecture to real 3G-WSN interworking network environments such as Home

Area Network and Neighborhood Area Network for implementing Smartgrid Testbed in Korea.

요 약 문

센서 네트워크와 이동 통신망을 위한 보안 기술 연구

모바일 환경 기술의 대표적인 기술인 무선 센서 네트워크 (Wireless Sensor Network, WSN) 기술과 이동통신망 기술은 최근들어 급격히 발전하고 있으며, 이종 환경 간의 융합 또한 많은 관심 속에 연구가 진행되고 있다. WSN은 베터리를 통해 전원 공급을 받고, 제한된 자원을 이용하는 센서 노드로 구성되어 있으며, 일반적으로 센서 노드 간의 Ad-Hoc 네트워크를 구성한다. 대표적인 이동통신망 기술은 3GPP (3rd Generation Partnership Project) 표준은 차세대 모바일 통신 시스템을 위해 SAE (System Architecture Evolution)/LTE (Long Term Evolution)구조를 개발하고 있다.

모바일 환경에서 공통적으로 제한된 자원을 갖고 있는 이동 통신 전화와 센서 노드의 에너지 효율성 확보는 매우 중요한 문제로 간주되고 있으며, 보안 기술 역시 에너지 효율성을 매우 중요한 요구 사항으로 하여, 효율적인 인증 및 키 교환 기술 등에 대한 연구를 진행하고 있다.

최근의 WSN의 발전에 따라 무선 센서/액터 네트워크 (Wireless Sensor & Actor Network, WSAN) 와 같은 이동성을 가진 노드가 나타나고 있으며, 다양한 응용 환경이 고려되고 있으나, 기존의 보안 기술 연구는 노드의 이동성에 대한 고려가 미비하여 이러한 경우 많은 에너지 소모가 예상된다.

따라서, 본 논문 연구를 통해 이동성을 가진 노드의 연속된 노드의 인증과 키 합의 과정에서 이미 인증 과정을 거친 노드의 경우 이후 인증 과정에서는 이전에 인증 정보를 활용하여 인증 단계를 단축시켰으며, 기존 기술에 비해 80 % 정도의 에너지 효율성을 얻을 수 있었다. 또한, WSN과 이동통신망의 융합 환경에서 센서 노드의 에너지 효율성을 대폭 증가 시키는 방안 역시 제시하였다.

한편, 차세대 이동통신망에서 광범위한 보안 문제에 대응하기 위한 공개키 기반 기술로서, 효율적인 공개키 관리를 가능하게 하는 ID 기반 암호 기술을 적용한 애플리케이션 보안 구조를 제안하였으며, 역시 통화 연결 지연 시간을 대폭 감소시킨 Voice over IP 보안 프로토콜 및 통화 감청 기술을 제시하였다.

본 논문 연구를 통해 점차 증가할 모바일 환경에서 다양한 보안 문제에 대한 대응이 가능할 것으로 예상한다.

감 사 의 글

본 논문을 완성하는데 정말 많은 분들의 도움이 있었습니다. 먼저 오랜 시간 동안 저를 믿고 지원해주신 부모님께 깊은 감사를 드리며, 저의 연구의 기반을 마련해주신 지도 교수이신 김광조 교수님께도 감사의 말씀을 드립니다. 그리고, 논문 심사를 해주신 한국과학기술원의 김대영 교수님, 한영남 교수님, 중부대학교에 계신 우리 연구실 출신의 이병천 교수님, 또 한국전자통신연구원의 최두호 박사님께도 감사의 말씀을 드립니다. 그리고, 같은 연구실에서 오랜 시간을 함께 지낸 Duc, 이현록, Divyan, 김진, 김장성과 제 박사과정 중 졸업한 Liem, 박재민, 이상신, 서영준, 허성철, 지성배, 윤성준, 곽민혜, 노한영, 박혜원, 신승목, 유명한, 최임성에게도 감사의 말을 전합니다. 연구실의 행정 업무를 지원해준 목선혜씨, 박현경씨, 홍지연씨에게도 감사의 말을 전합니다. 지금까지 제 졸업에 직간접적으로 많은 도움을 주신 Kalifa University에 계신 윤찬엽 교수님, 그리고 많은 연구를 같이 해오고 앞으로도 할 삼성전자에 계신 손태식 박사님, 서울산업대학교의 박종혁 교수님께도 감사의 말씀을 드립니다. 또, 오랜 시간 함께 해왔고, 앞으로도 함께 할 사랑하는 혜란이에게도 감사의 인사를 전합니다. 마지막으로, 제가 태어난 대한민국에 감사의 말을 전합니다.

References

[1] Masayuki Abe and Masayuki Kanda. A key escrow scheme with time-limited monitoring for one-way communication. *British Computer Society 2002, The Computer Journal*, 45(6):661–671, 2002.

[2] Masayuki Abe and Tatsuaki Okamoto. Delegation chains secure up to constant length. *IEICE Trans. Fundamentals*, E85-A(1):110–116, 2002.

[3] Jibi Abraham and K S Ramanatha. An efficient protocol for authentication and initial shared key establishment in clustered wireless sensor networks. *Proceeding of Third IFIP/IEEE International Conference on Wireless and Optical Communications Networks*, 2006.

[4] Ian F. Akyildiz and Ismail H. Kasimoglu. Wireless sensor and actor networks: research challenges. *Ad Hoc Networks*, 2(4):351 – 367, 2004.

[5] M. Abdul Alim and Behcet Sarikaya. EAP-Sens: A security architecture for wireless sensor networks. *WICON '08: Proceedings of the 4th Annual International Conference on Wireless Internet, Maui, Hawaii, USA*, November 17-19 2008.

[6] J. Arkko and H. Haverinen. Extensible authentication protocol method for 3rd generation authentication and key agreement (EAP-AKA),. Technical report, draft-arkko-eap-aka-kdf-10, November 2008.

[7] F. Baker, B. Foster, and C. Sharp. Cisco architecture for lawful intercept in ip networks. *RFC 3924*, 2004.

[8] M. Baugher, R. Canetti, L. Dondeti, and F. Lindholm. Multicast security (MSEC) group key management architecture. *IETF RFC 4046*, 2005.

[9] Ian F Blake, Gadiel Seroussi, and Nigel Paul Smart. *Elliptic Curves in Cryptography*. London Mathematical Society 265, Cambridge University Press, 1999.

[10] Dan Boneh and Matthew Franklin. Identity-Based Encryption from the Weil Pairing Advances in Cryptology. *Proceedings of CRYPTO 2001*, 2001.

[11] Dan Boneh and Matthew Franklin. Identity based encryption from the weil pairing. *SIAM Computing*, 32(3):586–615, 2003.

[12] Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the weil pairing. *J. of Cryptology*, 17(4):297–319, 2004.

[13] Haowen Chan, Adrian Perrig, and Dawn Song. Random key predistribution schemes for sensor networks. *in IEEE Symposium on Security and Privacy, Berkeley, California*, pages 197–213, May 11-14 2003.

[14] Inyoung Cho. Zigbee chip core technology and application area (in korean). *Technical Report, RadioPulse, Inc.*, 2009.

[15] Luca De Cicco, Saverio Mascolo, and Vittorio Palmisano. A mathematical model of the Skype VoIP congestion control algorithm. *IEEE Transactions on Automatic Control*, 55(3):790–795, 2010.

[16] William C Craig. Zigbee:Wireless control that simply works. *Zigbee Alliance*, 2005.

[17] Shantanu Das, Hai Liu, Ajith Kamath, Amiya Nayak, and Ivan Stojmenovic. Localized movement control for fault tolerance of mobile robot networks. *in IFIP International Federation for Information Processing, Wireless Sensor and Actor Networks, eds. L. Orozco-Barbosa, Olivares, T., Casado, R., Bermudez, A., (Boston:Springer)*, 248, 2007.

[18] Giacomo de Meulenaer, François Gosset, François-Xavier Standaert, and Olivier Pereira. On the Energy Cost of Communications and Cryptography in Wireless Sensor Networks. In *(extended version), IEEE International Workshop on Security and Privacy in Wireless and Mobile Computing, Networking and Communications (SecPriWiMob'2008)*, pages 580–585, 10 2008.

[19] Augusto Jun Devegili, Michael Scott, and Ricardo Dahab. Implementing cryptographic pairings over Barreto-Naehrig Curves. *Pairing-Based Cryptography – Pairing 2007, LNCS.*, 4575(2009):197–207, 2007.

[20] Yevgeniy Dodis, Jonathan Katz, Shouhuai Xu, and Moti Yung. Key-insulated public key cryptosystems. In *Theory and Application of Cryptographic Techniques*, pages 65–82, 2002.

[21] Yevgeniy Dodis, Jonathan Katz, Shouhuai Xu, and Moti Yung. Strong key-insulated signature schemes. *Proc. of PKC'03*, 2003.

[22] Wenliang Du, Jing Deng, Yunghsiang S. Han, and Pramod K. Varshney. A pairwise key pre-distribution scheme for wireless sensor networks. *in Proceedings of the 10th ACM conference on Computer and Communications Security (CCS), Washington. DC. USA*, pages 42–51, Oct 27-31 2003.

[23] Jeremy Elson and Deborah Estrin. Random, ephemeral transaction identifiers in dynamic sensor networks. *21st International Conference on Distributed Computing Systems*, page pp. 0459, 2001.

[24] Laurent Eschenauer and Virgil 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*, pages 41–47, Nov, 18-22 2002.

[25] Romano Fantacci, Francesco Chiti, and Leonardo Maccari. Fast distributed bidirectional authentication for wireless sensor networks. *John Wiley & Sons, Security and Communication Networks*, 1:17–24, 2008.

[26] Yair Frankel and Moti Yung. Escrow encryption systems visited: Attacks, analysis and designs. *In Proc. Advances in Cryptology-CRYPTO '95*, LNCS. 963:222–235, August 27-31 1995.

[27] Freescale. *Smart Application Blueprint for Rapid Engineering (SABRE) Platform for Smartbooks*. Freescale, release 1.0 edition, Dec 2009.

[28] Gerhard Frey, Michael Muller, and Hans Georg Ruck. The tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. *IEEE TRANSACTIONS ON INFORMATION THEORY*, 45(5):1717–1719, 1999.

[29] Darrel Hankerson, Alfred Menezes, and Michael Scott. Software implementation of pairings. *in Identity-Based Cryptography, edited by Marc Joye and Gregory Neven.*, May 2008.

[30] H. Haverinen and J.Salowey. EAP-SIM authentication. Technical report, draft-arkko-pppext-eap-sim-12, IETE, October 2003.

[31] Florian Hess. Efficient identity based signature schemes based on pairings. *9th Annual International Workshop, SAC 2002*, 2595(2003):310–324, 2002.

[32] Mengbo Hou, Qiuliang Xu, and Shanqing Guo. Secure key-escrowless explicit authenticated key agreement protocol in ID-based public key cryptography. *Journal of Information Assurance and Security*, 5:130–137, 2010.

[33] Qiang Huang, Johnas Cukier, Hisashi Kobayashi, Bede Liu, and Jinyun Zhang. Fast authenticated key establishment protocols for self-organizing sensor networks. *WSNA '03: Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, 2003.

[34] Te Yuan Huang, Polly Huang, Kuan Ta Chen, and Po Jung Wang. Could Skype be more satisfying? a QoE-centric study of the FEC mechanism in an internet-scale VoIP system. *IEEE Network*, 24(2):42–48, 2010.

[35] Jamil Ibriq and Imad Mahgoub. A hierarchical key establishment scheme for wireless sensor networks. *Proceedings of 21st International Conference on Advanced Networking and applications (AINA'07)*, pages 210–219, 2007.

[36] Nigel Jefferies, Chris Mitchell, and Michael Walker. A proposed architecture for trusted third party services. *In Proc. Cryptography: Policy and Algorithms, Brisbane, Australia*, LNCS. 1029:98–104, July 3-5 1995.

[37] Antoine Joux and Kim Nguyen. Separating decision diffie-hellman from diffie-hellman in cryptographic groups. Cryptology ePrint Archive, 2001.

[38] Chris Karlof, Naveen Sastry, and David Wagner. TinySec: a link layer security architecture for wireless sensor networks. *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, 2004.

[39] Chris Karlof and David Wagner. Secure routing in wireless sensor networks. *In Proc. of SNPA'03, Anchorage, Alaska*, pages 113–127, May 2003.

[40] Balamurugan Karpagavinayagam, Radu State, and Olivier Festor. Monitoring architecture for lawful interception in VoIP networks. *Second International Conference on Internet Monitoring and Protection (ICIMP 2007)*, July 2007.

[41] Sang Tae Kim and Hyun Deok Kim. Interoperation of home and mobile network devices employing a Jini-agent system. *LNCS, Vol 4096*, pages 1158–1165, 2006.

[42] Lei Kong, Vijay Arvind Balasubramaniyan, and Mustaque Ahamad. A lightweight scheme for securely and reliably locating SIP users. *The 1st IEEE Workshop on VoIP Management and Security (VoIP MaSe 2006)*, 2006.

[43] Sita S. Krishnakumar and Randal T. Abler. Intelligent actor mobility in wireless sensor and actor networks. *in IFIP International Federation for Information Processing,*

Wireless Sensor and Actor Networks, eds. L. Orozco-Barbosa, Olivares, T., Casado, R., Bermudez, A., (Boston:Springer), pages 13–22, 2007.

[44] Hang Rok Lee, Yong Je Choi, and Ho Won Kim. Implementation of tinyhash based on hash algorithm for sensor network. *Proceedings of World Academy of Science, Engineering and Technology*, 10:135–139, December 2005.

[45] Pierre Lescuyer and Thierry Lucidarme. *Evolved Packet System (EPS): The LTE and SAE Evolution of 3G UMTS*. J.Wiley & Sons, 2008.

[46] An Liu and Peng Ning. TinyECC: A configurable library for elliptic curve cryptography in wireless sensor networks. *2008 International Conference on Information Processing in Sensor Networks*, 2008.

[47] Donggang Liu and Peng Ning. Establishing pairwise keys in distributed sensor networks. *Proc. of the 10th ACM Conference on Computer and Communication (CCS)*, 2003.

[48] Ben Lynn. On the implementation of pairing-based cryptosystems. *Ph.d Thesis, Stanford University*, 2007.

[49] Luther Martin. *Introduction to Identity-Based Encryption*. Number ISBN-13: 978-1-59693-238-8 in Information Security and Privacy. Artech House, Inc., 685 Canton Street, Norwood, MA 02062, 2008.

[50] Alfred Menezes, Scott Vanstone, and Tatsuaki Okamoto. 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*, pages 80–89, New York, NY, USA, 1991. ACM.

[51] Silvio Micali. Fair public key cryptosystems. *In Proc. Advances in Cryptology-CRYPTO '92, Santa Babara, CA*, LNCS. 740:113–138, August 16-20 1992.

[52] Victor S. Miller. Short programs for functions on curves. *Unpublished manuscript*, 1986.

[53] Hyeran Mun, Kyusuk Han, and Kwangjo Kim. 3G-WLAN interworking: Security analysis and new authentication and key agreement based on EAP-AKA. *In Proc. of 2009 IEEE Wireless Telecommunications Symposium (WTS 2009), Prague, Czech Republic.*, April 22-25 2009.

[54] Jun Che Na. Next generation usim technologies. *TTA Journal (written in Korean)*, 116:80–85, Apr. 2008.

[55] NIST. Secure Hash Standard. *Federal Information Processing Standard, FIPS-180-1*, April 1995.

[56] Go Ohtake, Goichiro Hanaoka, and Kazuto Ogawa. An efficient strong key-insulated signature scheme and its application. *5th European PKI Workshop, NTNU, Trondheim, Norway*, June 16-17 2008.

[57] Takeshi Okamoto, Raylin Tso, and Eiji Okamoto. One-way and two-party authenticated ID-based key agreement protocols using pairing. *Modeling Decisions for Artificial Intel ligence, Second International Conference, MDAI 2005, Tsukuba, Japan*, 3558(2005):122–133, 2005.

[58] J. Peterson and C. Jennings. Enhancements for authenticated identity management in the session initiation protocol (SIP). *RFC4474*, August 2006.

[59] RadioPulse. Exhibition in IIC China 2010. The International IC-China Conference & Exhibition, 2010.

[60] Jared Ring, Kim-Kwang Raymond Choo, Ernest Foo, and Mark Looi. A new authentication mechanism and key agreement protocol for SIP using identity-based cryptography. *Proceedings AusCERT Asia Pacific Information Technology Security Conference*, 2006.

[61] J. Rosenberg, H. Schulzrinne, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. Sip: Session initiation protocol. *RFC 3261*, June 2002.

[62] RSA Laboratories. *RSAES-OAEP Encryption Scheme - Algorithm specification and supporting documentation*, 2000.

[63] Wolfgang Schott, Alexander Gluhak, Mirko Presser, Urs Hunkeler, and Rahim Tafazolli. e-SENSE protocol stack architecture for wireless sensor networks. *Proceedings of the 16th Mobile and Wireless Communications Summit*, July 2007.

[64] Jan Seedorf. Lawful interception in P2P-based VoIP systems. *Principles, Systems and Applications of IP Telecommunications. Services and Security for Next Generation Networks: Second International Conference, IPTComm 2008, Heidelberg, Germany. Revised Selected Papers*, pages 217–235, July 2008.

[65] Adi Shamir. How to share a secret. *Communications of the ACM*, 22(11):612–613, 1979.

[66] Adi Shamir. Identity-based cryptosystems and signature schemes. *Advances in Cryptology: Proceedings of CRYPTO 84*, LNCS. 196:47–53, 1984.

[67] Adi Shamir. Partial key escrow: A new approach to software key escrow. *In Key Escrow Conf., Washington, DC.*, September 15 1995.

[68] Sun Microsystems, Inc. *Runtime Environment Specification*, java cardtm platform, version 3.0.1 connected edition edition, 2009.

[69] Peter Thermos and Ari Takanen. Securing VoIP Networks: Threats, Vulnerabilities, and Countermeasures. *Addison-Wesley Professional, USA,*, August 2007.

[70] Third Generation Partnership (3GPP). *TS 33.106 v8.1.0 Lawful Interception Requirements*, Mar 2008.

[71] Third Generation Partnership (3GPP). *TS 33.102 v9.1.0 3G Security: Security Architecture (Release 9)*, Dec. 18 2009.

[72] Third Generation Partnership (3GPP). *TS 33.107 v8.8.0 Lawful interception architecture and functions*, Jun 2009.

[73] Third Generation Partnership (3GPP). *TS 33.108 v8.7.0 Handover interface for Lawful Interception (LI)*, Jun 2009.

[74] Third Generation Partnership (3GPP). *TS 33.220 v9.2.0 Generic Authentication Architecture(GAA); Generic Bootstrapping Architecture (Release 9)*, Dec. 18 2009.

[75] Third Generation Partnership (3GPP). *TS 33.221 v9.0.0 Generic Authentication Architecture(GAA); Support for Subscriber Certificates (Release 9)*, Dec. 31 2009.

[76] Third Generation Partnership (3GPP). *TS 33.401 v9.2.0 3GPP System Architecture Evolution (SAE); Security Architecture (Release 9)*, Dec. 18 2009.

[77] Third Generation Partnership Project (3GPP). *TS 33.402 v9.3.0 Architecture Enhancements for non-3GPP accesses (Release 9)"*, September 2010.

[78] Third Generation Partnetship (3GPP). *TR 33.919 v9.0.0 3G Security; Generic Authentication Architecture(GAA); System Descrption (Release 8)*, Dec 2009.

[79] Patrick Traynor, Michael Lin, Machigar Ongtang, Vikhyath Rao, Trent Jaeger, Patrick McDaniel, and Thomas La Porta. On cellular botnets: measuring the impact of malicious devices on a cellular network core. In *CCS '09: Proceedings of the 16th ACM conference on Computer and communications security*, pages 223–234, New York, NY, USA, 2009. ACM.

[80] Huei Ru Tseng, Rong Hong Jan, and Wuu Yang. An improved dynamic user authentication scheme for wireless sensor networks. *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*, pages 986–990, 2007.

[81] Eric R. Verheul. Evidence that xtr is more secure than supersingular elliptic curve cryptosystems. In *J. Cryptology*, pages 195–210. Springer-Verlag, 2001.

[82] Eric R. Verheul and Henk C. A. van Tilborg. Binding elgamal: A fraud-detectable alternative to key escrow proposals. *In Proc. Advances in Cryptology - EUROCRYPT '97*, LNCS. 1233:119–133, May 11-15 1997.

[83] Ronald Watro, Derrick Kong, Sue fen Cuti, Charles Gardiner, Charles Lynn, and Peter Kruus. TinyPK: Securing sensor networks with public key technology. *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, pages 59–64, 2004.

[84] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. LEAP: efficient security mechanisms for large-scale distributed sensor networks. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 62–72, New York, NY, USA, 2003. ACM.

[85] Sencun Zhu, Sanjeev Setia, and Sushil Jajodia. LEAP+: Efficient security mechanisms for large-scale distributed sensor networks. *ACM Trans. Sen. Netw.*, 2(4):500–528, 2006.

이 력 서

이 름 : 한 규 석

생 년 월 일 : 1978년 1월 23일

본 적 지 : 서울특별시 종로구 효자동 53-2

주 소 : 서울특별시 강남구 도곡동 도곡2차 아이파크 101동 1002호

E-mail 주 소 : hankyusuk@kaist.ac.kr

학 력

1993. 3. – 1996. 2. 보성고등학교

1996. 3. – 2001. 2. 홍익대학교 기계공학 (B.S.)

2001. 6. – 2004. 6. 한국정보통신대학교 대학원 공학부 전산학 전공(M.S.)

2005. 2. – 2010. 6. 한국과학기술원 정보통신공학 (Ph. D)

경 력

2001. 6. – 2004. 2. Graduate Research Assistant, *Cultivation of Top Level IT Security Manpower*, The Ministry of Information and Communications (MIC)

2002. 2. – 2003. 2. Graduate Research Assistant, *A Middleware Architecture for Virtual Community Service*, Electronics and Telecommunications Research Institute(ETRI)

2003. 7. – 2004. 4. Graduate Research Assistant, *Ubiquitous System Security Technology: Protecting Digital Contents from Illegal Use*, NITZ Co.

2003. 5. – 2003. 8. 싱가포르 I^2R 인턴

2004. 4. – 2004. 12. Graduate Research Assistant, *A Group-Aware Middleware Infrastructure for Active Surroundings*, Electronics and Telecommunications Research Institute(ETRI)

2004. 3. – 2009. 8. Gifted Students Teaching Assistant, *Education Program for Gifted Students*, Education Research Center for the Gifted in IT, Information and Communications University

2004. 1. – 2005. 12 Graduate Research Assistant, *Link Layer Security Algorithm Development and Standardization*, Electronics and Telecommunications Research Institute(ETRI)

2005. 7. – 2008. 6. Graduate Research Assistant, *ICU-Samsung Research Center*, Samsung Electronics.

2005. 7. – 2005. 12. Graduate Research Assistant, *UHF RFID and USN System Technology Development*, Electronics and Telecommunications Research Institute(ETRI)

2005. 3. – 2006. 2. Graduate Research Assistant, *A Study on the Security for Special Digital Signature*, Hannam University

2006. 8. – 2006. 12. Graduate Research Assistant, *A Study on the Ubiquitous-Web Platform Security*, KT

2008. 10. – 2008. 12. Graduate Research Assistant, *서비스 사업자 식별 체계 국제표준 기술 및 상호 접속 보안 요구사항 연구*, Electronics and Telecommunications Research Institute(ETRI)

2009. 1. – 2009. 9. Graduate Research Assistant, *USN Distributed Authentication Technology Development*, Samsung Electronics.

2009. 8. – 2009. 12. Teaching Assistant of KAIST Freshman Design Course

2010. 3. – Graduate Research Assistant, *iPhone OS Software Security Research*, NSRI

학 회 활 동

1. **Kyusuk Han**, Kwangjo Kim, and Taeshik Shon, *Efficient Sensor Node Authentication in 3G-WSN Integrated Networks*, [Submitted to ACM CCS '10 (4/17)]
2. Jangseong Kim, **Kyusuk Han**, Kwangjo Kim, *A Scalable and Robust Hierarchical Key Establishment Scheme for Mission-Critical Applications over Sensor Networks*, submitted to 18th IEEE International Conference on Network Protocols, Oct, 5-8, 2010, Kyoto, Japan.

3. **Kyusuk Han**, Hyeran Mun, Chan Yeob Yeun, and Kwangjo Kim, *Design of Intrusion Detection System Preventing Insider Attack*, The 10th International Workshop on Information Security Applications (WISA 2009), Aug. 24-26, 2009, Busan, Korea.
4. Hyeran Mun, **Kyusuk Han**, and Kwangjo Kim, *3G-WLAN Interworking: Security Analysis and New Authentication and Key Agreement based on EAP-AKA*, 2009 IEEE Wireless Telecommunication Symposium (WTS 2009), Apr. 22-25, 2009, Prague, Czech Republic
5. **Kyusuk Han**, Chan Yeob Yoon, and Kwangjo Kim, *New Key Escrow Model for the Lawful Interception in 3GPP*, 2009 IEEE International Conference on Consumer Electronics (ICCE2009), Jan. 12-14, 2009, Las Vegas, US
6. **Kyusuk Han**, Chan Yeob Yoon, Kwangjo Kim, *New Design of Generic Authentication Architecture Using ID-based Cryptosystem in 3GPP*, Triangle Symposium on Advanced ICT 2008 (TriSAI 2008), Oct. 6-9, 2008, Daejeon, Korea.
7. Hyewon Park, **Kyusuk Han**, Chan Yeob Yeun and Kwangjo Kim, *Improving Choi et al.'s ID-based Authenticated Group Key Agreement Scheme at PKC2004*, Proc. Of SCIS 2008, Jan. 22-25, 2008, Miyajaki, Japan.
8. Hyeran Mun, **Kyusuk Han**, Chan Yeob Yeun and Kwangjo Kim, *Yet Another Intrusion Detection System against Insider Attacks*, Proc. Of SCIS 2008, Jan. 22-25, 2008, Miyajaki, Japan.
9. **Kyusuk Han**, Chan Yeob Yeun and and Kwangjo Kim, *Design of secure VoIP using ID-Based Cryptosystem*, Proc. Of SCIS 2008, Jan. 22-25, 2008, Miyajaki, Japan.
10. Chan Yeob Yeun, **Kyusuk Han** and Kwangjo Kim, *New Novel Approaches for Securing VoIP Applications*, The Sixth International Workshop for Applied PKC IWAP, Dec 3-4, 2007, Perth, Australia.
11. **Kyusuk Han** and Kwangjo Kim, *Enhancing Privacy and Authentication for Location Based Service using Trusted Authority*, The 2nd Joint Workshop on Information Security, pp.219 - 232, Aug. 6-7, 2007, Tokyo, Japan
12. Dang N. Duc, **Kyusuk Han**, Zeen Kim and Kwangjo Kim, *A New Transitive Signature Scheme based on RSA-based Security Assumptions*, Proc. of Industrial and Short-Papers Track in ACNS2005, pp.165-175, Jun. 7 10, 2005, NY, USA.
13. Kui Ren, Hyunrok Lee, **Kyusuk Han**, Jaemin Park and Kwangjo Kim, *An Enhanced Lightweight Authentication Protocol for Access Control in Wireless LANs*, Proc. of ICON2004, IEEE, Nov. 16-19, 2004, Hilton, Singapore.

14. **Kyusuk Han**, Fangguo Zhang and Kwangjo Kim, *A Secure Testment Revealing Protocol*, Proc. of SCIS2003, vol 1/2, pp 399 404, Jan.26 29, 2003, Itaya, Japan.

15. 문 혜 란, **한 규 석**, 김 광 조, *3G-WLAN 상호연동: EAP-AKA에 기반을 둔 새로운 인증 및 키 합의 프로토콜*, CISC-W'08 Proceedings, Dec. 6, 2008 고려대학교, 서울

16. 김 장 성, 권 미 영, 김 이 형, 곽 민 혜, **한 규 석**, 김 광 조, *감시정찰 센서네트워크 및 주요 시설물 관리에서의 키 관리 기법 비교*, 2008년도 한국정보보호학회 충청지부 학술발표회 논문집, pp.75-83, 2008.10.17, 배재대학교,대전.

17. **한 규 석**, 윤 찬 엽, 김 광 조, *허가된 범위 외의 도청을 방지하는 암호 통신 복호화 키 제공 방식*, CISC-S'08 Proceedings vol.18, no.1, pp.212-215, 2008.6.26,순천향대학교, 천안.

18. **한 규 석**, 지 성 배, 김 광 조, *ID 기반 암호 기술을 이용한 VoIP 보안 서비스 설계 및 구현*, CISC-W'07 Proceedings vol.17, no.2, pp.540-543, 2007.12.1,상명대학교, 서울.

19. **한 규 석**, 김 광 조, *접근 권한이 있는 내부 사용자에 의한 정보 유출 방지를 위한 기술적 보안 방안 연구*, 2007년도 한국정보보호학회 하계학술발표대회, pp. 487-490, 2007. 6.22, 단국대학교 천안캠퍼스 제3과학관.

20. **한 규 석**, 서 영 준, 윤 성 준, 김 광 조, *Enhancing Security for Vertical Handoff in SARAH under the Heterogeneous Networks*, 2006년도 정보보호학술발표회논문집, pp. 159-166, 2006.9.29-30, 목원대학교, 대전.

21. **한 규 석**, 김 광 조, *신뢰 기관을 통한 위치 정보 기반 서비스의 프라이버시 보호 및 인증 기법*, CISC-S'06 Proceedings vol.16, no.1, p.623-p.626, 2006.6.30-7.1, 부경대학교, 부산.

22. **한 규 석**, 이송원, 김광조, 인소란, *On the design of secure DRM in ubiquitous environment*, 2004년 한국정보보호학회 영남지부 학술대회, 2004.2.20, 경일대학교, 경산.

연 구 업 적

1. **Kyusuk Han**, Taeshik Shon, Member, and Kwangjo Kim, *Efficient Mobile Sensor Authentication In Smart Home and WPAN*, In submission to IEEE Transaction on Consumer Electronics [Submitted on 3/1, Expected notification date is May]

2. **Kyusuk Han**, Hyeran Mun, Taeshik Shon, Chan Yeob Yeun and Kwangjo Kim, *Secure and Efficient Public Key Management in Next Generation Mobile Networks*, In submission to ACM Computer Communications Review [Submitted on 3/1, Expected notification date 7/1]

3. **Kyusuk Han**, Chan Yeob Yeun, Taeshik Shon, Jonghyuk Park, and Kwangjo Kim, *A Scalable and Efficient Key Escrow Model for Lawful Interceptions of IDBC based Secure Communication*, In submission to Special Issue on "Ubiquitous Computing for Communications and Broadcasting", Accepted to International Journal of Communication Systems, John Wiley & Sons, Ltd.

4. **Kyusuk Han**, Kwangjo Kim, and Taeshik Shon, *Untraceable Mobile Node Authentication in WSN*, Sensors 2010, 10(5)

5. **Kyusuk Han**, Kwangjo Kim, Taeshik Shon, *Efficient Authenticated Key Agreement Protocols for Dynamic Wireless Sensor Networks*, In submission to Ad Hoc & Sensor Wireless Networks [Submitted on 12/31, 2009]

6. Chanyeob Yoon, **Kyusuk Han**, Vo Duc Liem, and Kwangjo Kim, *Secure authenticated group key agreement protocol in the MANET environment*, Information Security Technical Report 13, pp.158-164, Elsevier, 2008.

7. Songwon Lee, **Kyusuk Han**, Seok-kyu Kang, Kwangjo Kim and So Ran Ine, *Threshold Password-Based Authentication Using Bilinear Pairings*, Proc. of European PKI, LNCS 3093, pp.350-363, June 25-26, 2004, Springer-Verlag, Samos island, Greece.

8. [Patent] 손 태 식, 박 용 석, **한 규 석**, 김 광 조, 김 장 성, *이동통신망을 이용한 싱크 인증 시스템 및 방법*, 출원번호 10-2009-0114725

9. [Patent] **한 규 석**, 김 장 성, 김 광 조, *센서 네트워크에서 센서 노드 인증 방법 및 장치*, 특허출원 (제P2009-0057778호)

10. [Patent] **한 규 석**, 김 장 성, 김 광 조 *센서 네트워크에서 노드와 싱크간의 상호 인증 시스템 및 방법*, 특허출원 (제P2009-0057175호)

11. [Patent] 박 영 준, 안 민 영, 신 국, 김 광 조, **한 규 석**, *통신 시스템에서 인증 방법*, 2007.8.9. 특허출원 (제2007-0094224호)

12. [Patent] 박 영 준, 안 민 영, 신 국, 김 광 조, **한 규 석**, *통신 시스템에서 암호 통신 방법*, 2007.9.17. 특허출원 (제2007-0094224호)