DOI: 10.2298/CSIS120917037D

Key Management Approach for Secure Mobile Open IPTV Service

Inshil Doh[1], Jiyoung Lim[2]*, and Kijoon Chae[1]

[1]Ewha Womans University,
{isdoh1, kjchae}@ewha.ac.kr
[2]Korean Bible University
jylim@bible.ac.kr
*Correspondent Author

Abstract. In mobile open Internet Protocol TV (IPTV) which is one of the major attracting technologies recently, the security is a key issue for reliable service, because the mobility and the openness in IPTV could cause much more vulnerabilities to various attacks compared with traditional IPTV services. In this paper, we propose an energy-efficient and secure channel group key establishment and rekeying management scheme for mobile open IPTV services. Our scheme provides the data authentication between an Evolved Node B (eNB) or a Base Station and the mobile devices for the security enhancement and efficiently rekeys the group key when the membership changes. Additionally, it proposes a pairwise key establishment mechanism for open IPTV services through eNBs. Our proposal can cope with the security vulnerability in mobile open IPTV services and guarantee the secure group key rekeying in addition to decreasing the storage and communication overhead.

Keywords: group key; pairwise key; channel; security; rekeying; authentication; mobile open IPTV

1. Introduction

IPTV is a system through which television services are delivered using the Internet protocol suite over a packet-switched network. It has attracted a lot of interest as many intelligent devices appear and support IPTV related functions. Secure IP multicast may be used to support the secure transmission of IP packets to groups of receivers in IPTV services but neglects access control and network management. Key distribution solutions for secure group communications usually apply key refreshing techniques upon a group change (member join or leave) in order to impose both perfect forward and backward secrecy [1,2].

Recently, with the advance of mobile devices technology, users would want to receive their services through mobile devices anywhere, and mobility is additionally required for IPTV service. However, the use of wireless environment has many risks and weaknesses when it is compared with the existing wired networks. There are two approaches for mobile IPTV security technologies as in Fig. 1. One is adding the mobility to IPTV, and the other is adding IP technologies to mobile TV such as DMB, DVB, and so on. In our work, we are focusing on adding the mobility to the fixed IPTV technology.

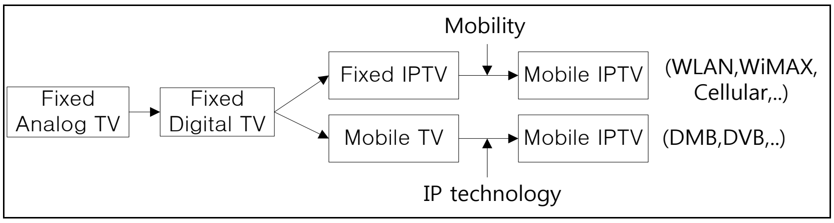


Fig. 1. Mobile IPTV Technology Approaches

In addition to mobile IPTV, in open IPTV, consumers refuse to be passive content users, but instead want to have influence as content providers and choose the contents they want [3]. Open IPTV is one of the application areas of Machine-to-Machine (M2M) communication services, which is a major paradigm shift. As the major standardization organization, the Open IPTV Forum [4] is developing an end-to-end solution to provide personalized IPTV services in a managed or non-managed network.

Because IP networks are open to anyone, IPTV based on IP network has the attacks such as unauthorized access and watching, illegal copy and circulation, and so on. To solve these problems, current broadcasting systems adopt encryption technologies such as the Condition Access System (CAS) [5] and Digital Rights Management (DRM) [6]. However, when mobility is considered, security technologies for traditional IPTV are not proper to adopt. For the security requirements for mobile IPTV, group management for users who subscribe the membership and watch the channel is essential. Fundamental of group management is the group keys.

In mobile IPTV, the users join in and leave the service often while moving. Every time the users join in, they need to be provided the group keys, but they are not supposed to know the previous contents, so the rekeying is required. When they leave the service, the keys need to be rekeyed for the leaving nodes not to get the service any more. This frequent rekeying makes the security vulnerable. Especially, in the open IPTV, where each user can be the service provider, it is much more complicated. If key management system becomes vulnerable due to its poor security, there is possibility that the security of the whole communication system becomes insecure. Other related works have not considered the frequent membership changes or the openness. Therefore, we propose a channel group key management mechanism based on Pre-distribution and local Collaboration-based Group Rekeying (PCGR) and an automatic group key rekeying mechanism considering membership changes and device mobility [7]. In addition, for the users to communicate for open IPTV, pairwise keys between them need to be established. In this paper, we additionally propose a pairwise key management for efficient mobile open IPTV service. Our contributions are as follows.

- Our proposal basically supports data authentication functionality through eNBs by verifying the information received in the rekeying process.
- By considering the frequency of membership change, our mechanism increases the efficiency of channel group key rekeying with low communication, computation, and storage overhead.
- Device communication for each pair of users who participate in the open IPTV service is also described in our work.

The remainder of this paper is organized as follows. Section 2 describes the related works for IPTV and group key management schemes. We also briefly describe the PCGR which we partly adopt in our mechanism. Section 3 presents the previous proposed group key management mechanism which provides data authentication and automatic rekeying among IPTV users. Open IPTV service between devices is presented in section 4. Section 5 evaluates the effectiveness of advanced mechanism and analyzes the security issues. Finally, we conclude our paper in Section 6.

2. Related Works

Major researches on secure IPTV service are described in this section. In addition, in considering the mobility of devices and group communication security, group key management mechanisms including PCGR that we partially adopted in our work are presented in this section.

2.1. IPTV Security

As IPTV brings a lot changes in industrial and technological aspect, security becomes a key issue to solve for the service. To prevent the unauthorized watching of IPTV, user authentication and access control are required. CAS [5] and DRM [6], the major technologies for IPTV security, are frequently adopted [8]. They differ from each other in terms of how they are applied; however, they also complement each other at the same time. CAS is the core technology for securely transferring content encrypted with a the private key preloaded for each user, and it is used for content protection in traditional digital and satellite TV, as well as IPTV, etc. The structure of CAS is shown in Fig. 2. At the head-end, control word (CW) is used to initialize the generation of a pseudo random sequence number. The pseudo random sequence number is generated by a pseudo random sequence generator for scrambling and descrambling of video programs. The CW for each subscriber is encrypted with the authorization key (AK) of the corresponding channel and the encrypted CW forms an entitlement control message (ECM). The AK is also encrypted using the private key (DK) and the encrypted AK forms an entitlement management message (EMM). The ECM, EMM and the scrambled program are re-multiplexed in a new transport stream which is broadcast in the form of a radio frequency signal. The subscriber management system is used to administer the issue of or update of the smart card for a subscriber, which contains the DK and other account information. At the receiver end, the receiver can descramble the program according to the reverse steps of the head-end with the cooperation of the smart card and Set-Top-Box (STB).

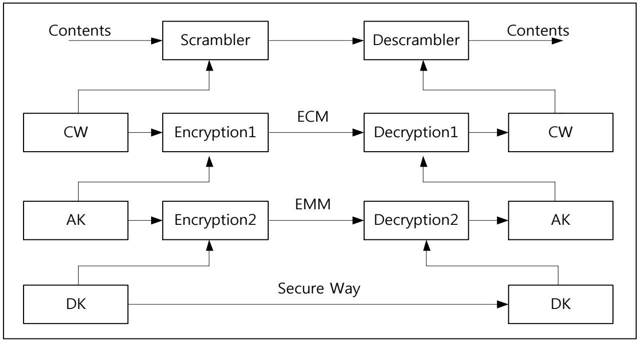


Fig. 2. Traditional CAS System

DRM [6] is a technology designed to prevent the unauthorized use and duplication of digital media. Technically, this technology is based on the encryption of the data. The key used for decryption is itself encrypted and bundled with the permissions. The encrypted data and corresponding licenses are typically associated with each other using unique content identifiers. In order for a single receiver to access and use the license, and thus the content key and content, the encrypted license should be known only to the sender and the intended receiver. This can conveniently be done by deploying a public-key infrastructure and surrounding trust system.

Several researchers [9], [10], [11] have considered the security problem for IPTV services. However, most of them deal with traditional IPTV which is static and cannot be applied for mobile IPTV services. For open IPTV network security, in our previous work, we proposed a secure user authentication and key distribution based on Kerberos for open IPTV security. We also proposed a contents sharing mechanism in home network [3].

2.2. Group Key Management Mechanisms

Group key management has been researched a lot, and the mechanisms can be classified into three categories.

In centralized key management schemes, a group manager generates group keys and distributes the key to authenticated group members and manages key material and lists. Blundo, C.et al. proposed a mechanism in which a server chooses a t-degree polynomial randomly and distributes them to neighbor nodes and the member nodes substitute the polynomial with their IDs; hence, all the nodes share one group key [12]. Wang, Y. and Ramamurthy, B. proposed four safe group communication methods [13]. Information for group key rekeying is unicasted to each node. This creates a heavy overload when group size grows. Broadcasting is proposed to solve the overhead problem. The broadcasting mechanism requires heavier overhead when groups are generated; however, rekeying cost is relatively low. Overlapping is also proposed to prevent flooding attack. Finally, group information predistribution minimizes group generation time. Karuturi, N.N.et al. provide a generalized framework for centralized GKM along with a formal security model and definitions for the security properties that dynamic groups demand. A lot of researches have been done for centralized group key management. However, in mobile communication environment, parent-child relationship changes constantly because of devices movements. Even if centralized management is very stable and secure, it is not proper for adopting in mobile network.

In distributed key management, multiple key managers generate group keys and distribute them to authentic members. Zhang, W. and Cao, G. proposed a mechanism (PCGR) that predistributes key related information and generates group keys[14]. When group key rekeying is required, nodes cooperate and a new group key is computed. This scheme is applied in our proposal and will be more described in subsection 2.3. Huang, J. H.et al. proposed a level key infrastructure for multicast and group communication that uses level keys to provide an infrastructure that lowers the cost of nodes joining and leaving [15]. This scheme has a drawback in that process delay increases even when many nodes are changed. Zhu, S. et al. proposed a key management protocol for sensor network designed to support in-network processing, while at the same time restricting the security impact of a compromised node [16]. This mechanism is safer, because it uses four different kinds of keys. However, key update consumes much overhead. Adusumilli, P., Zou, X. and Ramamurthy, B. proposed a Distributed Group Key Distribution (DGKD) protocol which does not require existence of central trusted entities such as group controller or subgroup controllers [17]. Aparna, R. and Amberker, B.B. proposed a key management scheme for managing multiple groups. They uses a combination of key-based and secret share-based approach for managing the keys and showed that it is possible for members belonging to two or more groups to derive the group keys with less storage [18]. Kim, Y., Perrig, A, and Tsudik, G. investigated a novel group key agreement approach which blends key trees with Diffie-Hellman key exchange [19]. It yielded a secure protocol suite called Tree-based Group Diffie-Hellman (TGDH) that is both simple and fault-tolerant.

Contributed management mechanisms rekey the group keys through nodes' cooperation without specific key managers. Yu, Z. and Guan,Y. propose a group key management mechanism [20] in which basic matrix G and secret matrices A,B are assigned to each sensor node; each matrix is used to generate group keys among nodes in the same groups and different groups, respectively. The advantage of this mechanism is that the probability of generating group keys is high. However, when the grid size is large, much energy is wasted and when the grid size is small, group keys may not be generated.

2.3. PCGR

This scheme was designed based on the idea that future group keys are generated by neighbors that can collaborate to protect the communication and appropriately use the preloaded keys [14].A detailed description is provided, since our proposal partly adopts this scheme.

Setup server constructs a unique univariate t-degree g-polynomial g(x), and g(0) is the initial group key (Fig.3(a)). After a device has been deployed and has discovered its neighbors, it randomly picks a bivariate e-polynomial and generates g'-polynomial (Fig.3(b)). The encryption polynomial is generated as follows.

$$e_u(c,y) = \sum_{j=0}^{\mu} B_j y^j \tag{1}$$

Encryption is conducted as,

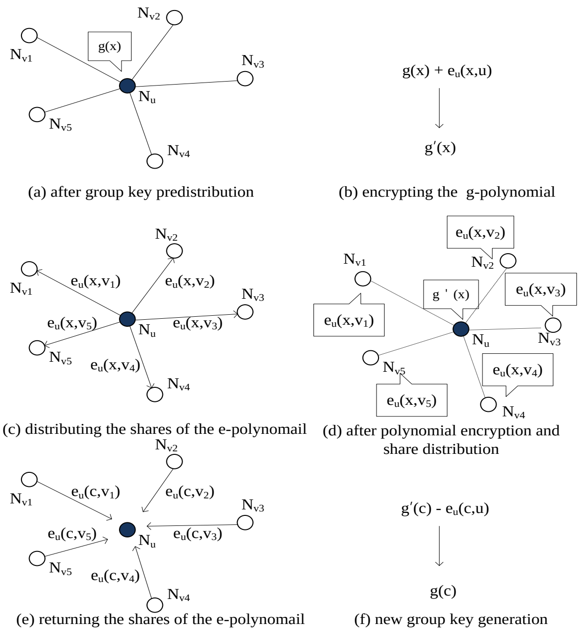
$$g'(x) = g(x) + e_u(x,u). \tag{2}$$

After distributing the shares of $e_u(x, y)$ to its n neighbors as in Fig. 3(c), N_u removes $e_u(x,y)$ and g(x), but keeps g'(x). Fig. 3(d) illustrates the final distribution of g'(x) and $e_u(x,v_i)$.

Every device maintains a timer for rekeying. When the time expires, each innocent device N_u increases its c by one, and returns share $e_{vi}(c,u)$to each trusted neighbor, N_{vi}. Meanwhile, as shown in Fig.3 (e), N_u receives a share $e_u(c, v_i)$ from each trusted neighbor N_{vi}. Having received μ+1 shares, N_u can reconstruct a unique μ-degree polynomial as

$$\sum_{j=0}^{\mu} (v_i)^j B_j = e_u(c,v_i)(0 \le i \le \mu). \tag{3}$$

Finally, N_u computes the new group key g(c) = g'(c) - $e_u(c,u)$ as in Fig.3(f).This scheme has the advantage that even if some devices are attacked, the new group key is not revealed. However, major drawback of this scheme is that any node in the network can initiate the group key rekeying, causing heavy overhead.



(a) after group key predistribution

(b) encrypting the g-polynomial

(c) distributing the shares of the e-polynomail

(d) after polynomial encryption and share distribution

(e) returning the shares of the e-polynomail

(f) new group key generation

Fig. 3. PCGR: Polynomial Encryption, Share Distribution, and Key Updating

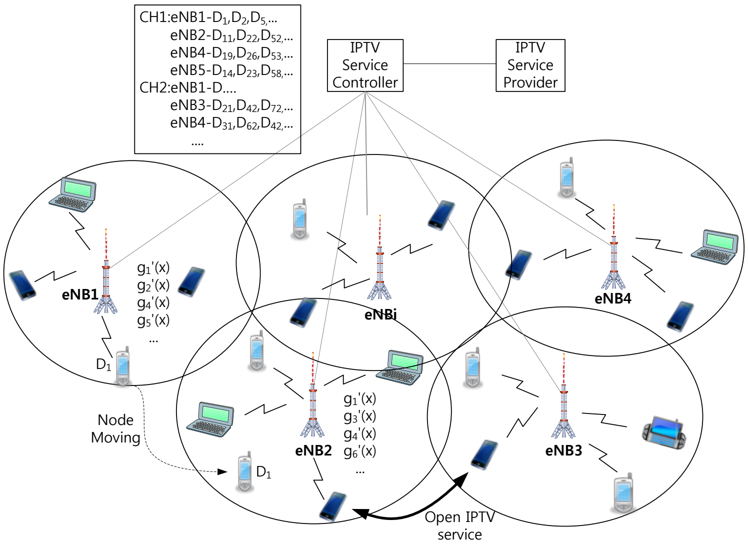


Fig. 4. System Environment including the Device Movement and Open IPTV Service of Our Proposal

3. Proposed Key Management for Secure Mobile IPTV Services

3.1. System Architecture and Basic Assumptions

Based on the cellular network where devices receive their data through eNBs, mobile devices are provided IPTV services through eNBs and ISC in our proposal. Devices watching the same channel share a group key which is used to encrypt the contents delivered through the eNB, which means individual key is assigned for each channel. Basic key materials (polynomial coefficient values) are assigned to eNBs and devices according to the rekeying cycle. Group keys are shared among devices which receive the IPTV service.

Before each device belongs to its own eNB, pairwise keys between the eNB and the device are preassigned. Routing is not considered in our work. As in fig. 4, in our proposal, devices move from one cell to another, and devices can communicate with each other when they subscribe in the open IPTV service.

3.2. Group Key Initialization

For IPTV service, there are many channels for the users to select, and the contents delivered through the channel need to be secured. We define the devices which subscribe and receive the contents from a channel as a group. For each group, group keys for encrypting the contents are required. The most important issue here is how to generate group keys and how to update them efficiently for secure IPTV service. Rekeying is required according not only to the rekeying cycle but also to the membership change. We also need to consider the members mobility. For these objectives, we partially adopt the PCGR for group key generation and rekeying for securing the contents.

Each channel requires individual encryption key for securing the IPTV contents. Because of device mobility, CAS is not proper for securing the contents because it is designed to be installed in STB for traditional IPTV service. We adopted a part of basic PCGR and modified it for channel key generation and rekeying when required. ISC generates the channel key polynomials, $g_i(x)$ for channel i and $e_i(x,y)$ for each $g_i(x)$, and distributes the information to each eNB under the channel service. eNBs receive as many g(x)s as the number of channels that the members of eNB belong to. For each channel i, ISC also generates $e_i(x,y)$ as

$$e_i(x,y) = a_i(x,y) \times d_i(x,y) + q_i(x,y). \qquad (4)$$

With encryption polynomials as above, eNBs can verify the shares from devices to filter the false shares and decide which device is illegally receiving the IPTV service. Using the e-polynomial (i.e., $e_i(x,y)$), eNB encrypts the g-polynomial (i.e., $g_i(x)$) to get its g' polynomial (denoted as $g_i'(x)$). The encryption can be conducted as follows:

$$g_i'(x) = g_i(x) + e_i(x, i) \tag{5}$$

After receiving $g_i(x)$ and $e_i(x, i)$, eNB sends $g_i(0)$ to the member nodes. Next, eNB distributes the shares of $e_i(x, y)$ to its member devices D_{v_i} (i= 0, ⋯ , n−1). Specifically, each device D_{v_i} receives share $e_i(x, v_i)$. eNB unicasts this message to each device, including the individual encryption polynomial, $d_u(x)$, and $q_u(x)$, after encrypting the message with $g_i(0)$.

$eNB_i \Rightarrow D_v$: $E_{g_i(0)}\{e_i(x,ID_{Dv})\}$

($1 \le v \le n$, n is the number of devices in the group)

After transmission, eNB removes $e_i(x,y)$ and $a_i(x,y)$ that has been used to generate $e_i(x,y)$ for security, but keeps $g_i'(x)$. After group initialization, the following information is retained.

eNB : $g_i'(x)$, $d_i(x,y)$, $q_i(x,y)$

Device v: $e_i(x,ID_{Dv})$

Fig.5 shows the group key initialization processes.

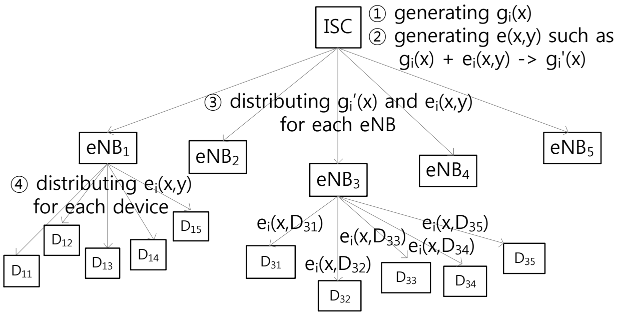

ISC
① generating gi(x)
② generating e(x,y) such as gi(x) + ei(x,y) -> gi'(x)
③ distributing gi'(x) and ei(x,y) for each eNB
eNB1
eNB2
eNB3
eNB4
eNB5
④ distributing ei(x,y) for each device
D11
D12
D13
D14
D15
ei(x,D31)
ei(x,D32)
ei(x,D33)
ei(x,D34)
ei(x,D35)
D31
D32
D33
D34
D35

Fig. 5. Group Key Initialization Flow among ISC, eNB, and Devices

3.3. Group Key Update on Rekeying Cycle

Channel group keys are periodically renewed according to the following processes on rekeying cycle.

(1) On rekeying time, ISC sends the group key rekeying command to eNBs which have subscribers of the channel i.

(2) eNB sends out the request_share message with random number asking group key shares of $e_i(x,y)$ to its member devices.

(3) Devices receiving this message reply with the result value after computing the encryption polynomial. The value is encrypted with current group key $g_i(c)$ after being computed by substituting x with random number r, y with the ID of the device, ID_{Dv}.

$D_v \Rightarrow eNB{:}\,E_{g_i(c)}\{e_i(r,ID_{Dv})\}$(1≤*v*≤*n*, *n* is the number of devices in the cell)

(4) After receiving the key shares, eNB first verifies the values. Because the encryption polynomial was generated as $a_i(x,y) \times d_i(x,y) + q_i(x,y) = e_i(x,y)$,the return value is verified if $e_i(r,ID_{Dv}) \bmod d_i(r,ID_{Dv}) = q_i(r,ID_{Dv}) \bmod d_i(r,ID_{Dv})$.

If the result is true, eNB considers that the device is authenticated. After gathering μ+1 key shares from devices, $e_i(r, ID_{eNB})$ is computed and a new group key is generated, as follows.

$$g_i(r) = g_i'(r) - e_i(r, ID_{eNB}) \tag{6}$$

If $g_i(x)$ is a t-degree polynomial, at least t+1 key shares from neighbor devices are needed to compute the new group key, $g_i(r)$.

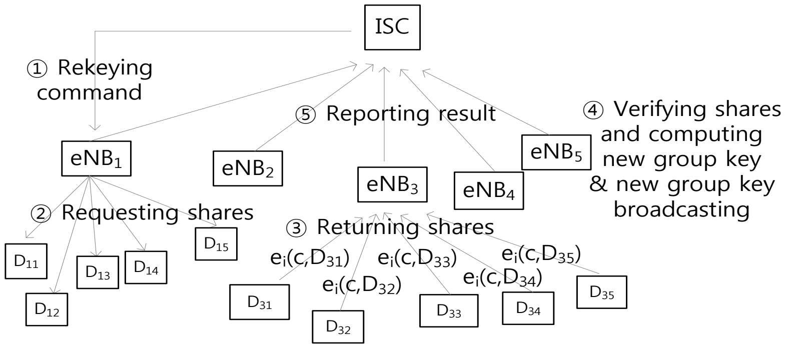


Fig. 6. Group Key Rekeying for All Devices

(5) eNB broadcasts new group key $g_i(r)$ after encrypting it with current group key $g_i(c)$ to its member devices in cell *i*.

$eNB \Rightarrow D_v{:}\ E_{g_i(c)}\{g_i(r)\}$

(1≤*v*≤*n*, *n* is the number of devices under the channel service)

When a device fails to verify itself, it may be assumed to be an illegal watcher. The eNB notifies this to ISC to recheck the subscription. If it is illegal, another rekeying is induced. The result of rekeying whether it is successful or not is encrypted with the pairwise key between the eNB and the ISC and delivered to ISC. Fig.6 depicts the processes.

Terminologies for our proposal are described in Table 1.

Table 1.Terminologies for Our Proposal

Terminologies	Description
Di	Device node i
ID_{D_v}	ID of device D_v
$g_i(x)$	Group key polynomial for channel *i*
$e_i(x,y)$	Group key encryption polynomial of channel *i*
$g_i'(x)$	Encrypted group key polynomial of channel *i* using $e_i(x,y)$
$a_i(x,y),d_i(x,y),q_i(x,y)$	Polynomials of group *i* for generating $e_i(x,y)$
r	Random number *r*
$g_i(c)$	Group key of channel *i* in current session
$g_i(r)$	New group key of channel *i*
$w_i(x)$	Polynomial to exclude a leaving node
$f_i(r)$	Polynomial made with $w_i(x)$ and $f_i(r)$ for isolating the leaving device
TH	Threshold value for group key rekeying
THstd	Standard TH to start group key rekeying process

3.4. Group Key Rekeying Triggering

When membership changes occur, eNBs report this to ISC to check if group key rekeying is required or not.

Device Leaving from the Service Group When some devices don't want to receive the channel service anymore, group key rekeying is required for forward secrecy, which means leaving device should not get the future contents anymore. eNB notifies member leaving to ISC and then ISC checks if normal rekeying process is required. If normal group key rekeying is not required, temporary group key is adopted. For this, instead of encrypting the new group key with the old group key, ISC generates $f_i(x)$ as follows to isolate the device from the service group.

$$f_i(x) = g_i(x) \times w_i(x), \tag{7}$$

where, $w_i(x)=(x- x_1)(x- x_2)\ldots(x- x_{k-1})(x-ID_{D_x})$

(k is the number of devices in group)

D_x is the leaving node, which means that when a leaving device inputs its ID in the formula, $w_i(x)$ becomes zero and the node cannot compute the new group key. Other devices divide $F_i(r)$ by $w_i(ID_{Di})$ and get $f_i(r)$. They can take part in the new group session having obtained this new group key as follows.

$$eNB \Rightarrow D_v\text{: } Eg_i(c)\{f_i(x)||w_i(x)\} \tag{8}$$

($1 \le v \le n$, n is the number of devices in a cell)

$g_i(r)=f_i(r)/w_i(ID_{Dv})$

New Device Join in the Service Group For backward secrecy, group keys need to be rekeyed when new nodes join the IPTV service group. When eNB reports ISC that a new device will be added to the service group, ISC checks the rekeying condition, and decides whether rekey the whole group key or just adopt temporary key for newly joining users. If the ISC decides the latter one, it prepares a polynomial as in (7) and (8), and sends the newly generated group key and $e_i(x, ID_{new})$ encrypted with the pairwise key between eNB and the new device to individual new joining nodes. When eNB confirms that the new node is authentic one with the help of ISC, it unicasts the new group key and $e_i(x, ID_{new})$ encrypted with the pairwise key between eNB and the new device to the newly joining node. After receiving this information, the new node sends the confirm message encrypted with the new group key to eNB. This message can be decrypted by all original members. They can also confirm the new member has joined the group.

As described in previous subsection, group key rekeying is composed of many steps and could cause serious computation and communication overhead if group key rekeying is started on every membership changes. When some nodes frequently change the subscription or when some nodes just join or leave the service group right after the periodic rekeying, the efficiency is decreased. To deal with this situation, after getting the membership change report from the eNBs, the ISC checks whether normal group key rekeying is required or not. The threshold value for deciding to start rekeying process or not is computed as follows.

$$TH =\alpha\cdot(Acc\_users/Tot\_uers)(1+\beta\cdot(Spent\_Time/Rekeying\_Time)), \qquad (9)$$

where Acc_users is the accumulated number of users who have changed their membership by leaving from or joining in the service group, and Tot_users is the number of total users who are subscribing the channel service. Spent_Time is the time since the latest rekeying time, and Rekeying_Time is the normal rekeying time period. It means that more than certain number of users changed their memberships and certain amount of time has spent after the periodic group key rekeying. α and β are the system parameters and can be adjusted between 0 and 1according to system environment. When α is big, the number of membership changing users is more importantly considered, while even if β is big, it cannot trigger group key rekeying if there is no membership change at all. Basically, the number of membership changing users is much more important in normal situations. The overview of our proposed system flow is shown in Fig. 7.

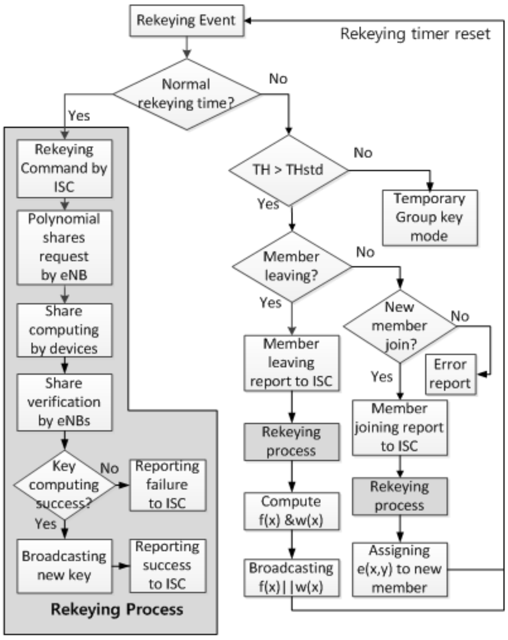


Fig. 7. Proposed Group Key Rekeying System Flows

3.5. Other Considerations

Service Member Mobility Management Different from the traditional IPTV service, in mobile IPTV, devices can move from one cell to another. They still want seamless service while they move. The important point here is that we need to supply them with the same quality of service while providing the new group key even though their locations change. Based on the IPTV systems, handoffs may occur or not. We don’t consider the location update process here, and what we want to focus is that the mobile device is still the member of IPTV service, which means we don’t need group key rekeying. The device just moved in a cell still has the old polynomial share from the old eNB. The device can join the rekeying process because new group key is encrypted with the old group key if membership change does not occur. At the first group key rekeying time in the new cell, the device gets its own polynomial share from the eNB of the new cell. With the share, the device can contribute its own share in the next rekeying process.

eNB Cooperation for Rekeying When the number of devices that receive the channel service is less than μ+1, the eNB cannot gather enough shares and hence cannot compute new group key for its own cell. In this case, more than one eNB need to cooperate and exchange the shares with each other.

When the cell is isolated and the eNB has difficulty for finding the other eNBs with which it can cooperate, the eNB notifies this to ISC and ISC can send the new group key for the eNB and the related mobile devices.

Group key polynomial update In traditional CAS, AK is regenerated by the system parameter for security purpose. In our system, ISC generates new $g_i(x)$ for each channel i according to the membership changes and the number of subscribers. When membership changes occur often, group key rekeying frequency is influenced more by the member leaving or joining events than by rekeying cycle. And in this situation, the lifetime of $g_i(x)$ for channel i is getting shorter, which means ISC needs to changes the $g(x)$ more often.

4. Key Management for Securing Mobile Open IPTV Services

For mobile open IPTV service, each devices need to subscribe the service not only to receive the contents but also to provide the contents of themselves. For secure communication between the devices, they need pairwise keys with each other. These pairwise keys can be generated by eNBs or by ISC according to the locations of the devices.

Pairwise key establishment between subscribers in the same cell When a device wants to subscribe the open IPTV service, it needs to request the service with the contents list it has for the IPTV Service Provider can manage the contents list. After requesting the service, the device can get the list from the ISP and can provide contents to or receive contents from other devices.

When a device requests some contents from a device in the same cell, this is notified to the eNB, and the eNB generates the pairwise key for the pair of devices and distribute the key encrypted with respective symmetric keys. With this key, the two devices can communicate with each other as in Fig. 8.

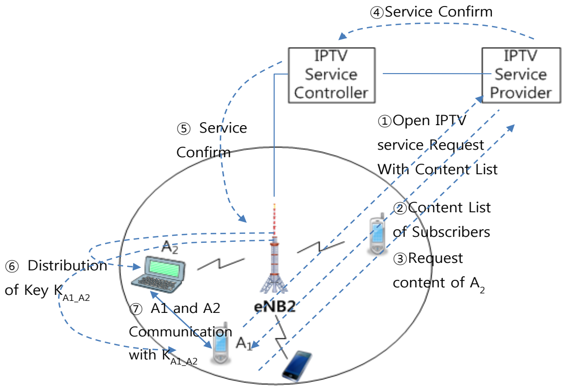

④Service Confirm
IPTV Service Controller
IPTV Service Provider
⑤ Service Confirm
①Open IPTV service Request With Content List
②Content List of Subscribers
③Request content of A_2
⑥ Distribution of Key K_{A1_A2}
A_2
eNB2
⑦ A1 and A2 Communication with K_{A1_A2}
A_1

Fig. 8. Pairwise key establishment between devices for open IPTV in the same cell

Pairwise key establishment between subscribers in different cells When a device wants to communicate with the other device from different cell, eNBs cannot generate the pairwise key between them. In this case, ISC generates the pairwise key and distributes it to each eNBs of the respective cells. After getting the pairwise key between two devices, each eNBs encrypt the pairwise keys and redistribute it to each device. The pair can communicate with each other with the key in secure manner. As mentioned in the assumption, routing is out of scope in our work. The steps are as follows.

(1) When a user A_1 wants to get the open IPTV service, which means he or she wants to get any content from the other user, s/he needs to send the request message encrypted with pairwise key between the device and the eNB to the regional eNB, and this message is delivered to ISC and then to ISP. With the request message the contents list of the device can be reported to the ISP for the other users to request the content from the device.

(2) After checking the authenticity of the device, ISP sends the confirm message and the content list it manages to the requesting user.

(3) When A_1 decides some contents from the list, it requests the contents to the ISP. This message is also encrypted with the pairwise key between A_1 and eNB2.

(4) After receiving content lists from A_1, ISP checks if the content holders are in the same cell or not, and delivers the information to ISC.

(5) If they are in the same cells, ISP gives the right to generate pairwise keys to the eNB as in Fig. 8. If they are located in different cells, ISC generates the pairwise keys for A_1 and B_1 and delivers the keys to each eNBs to redistribute them to individual devices. These keys are also encrypted with pairwise keys between eNB and the devices. Finally, the devices can exchange the contents in secure manner. This process is in shown in Fig. 9.

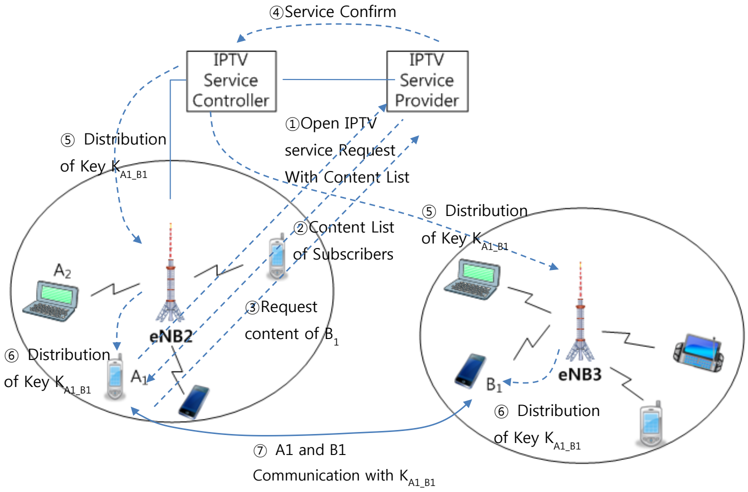

④Service Confirm
IPTV Service Controller
IPTV Service Provider
①Open IPTV service Request With Content List
⑤ Distribution of Key K_{A1_B1}
⑤ Distribution of Key K_{A1_B1}
②Content List of Subscribers
③Request content of B_1
A_2
eNB2
⑥ Distribution of Key K_{A1_B1}
A_1
eNB3
B_1
⑥ Distribution of Key K_{A1_B1}
⑦ A1 and B1 Communication with K_{A1_B1}

Fig. 9. Pairwise key establishment between devices for open IPTV in different cells

5. Performance Analysis

In this section, we analyze communication, computation, and storage overhead of our proposal. There is no proper existing work that we can compare with in the same structure in the mobile open IPTV domain where group memberships change very often while each pairs of users in the group can communicate with each other, i.e., open IPTV, so we compared our proposal with Blundo's mechanism and the basic PCGR mechanism to show how much ours can decrease the overhead efficiently.

5.1. Overhead Analysis

Communication Overhead Our proposal has less overhead compared to other approaches. In centralized scheme such as LKH [21] or SKDC [22], central controller sends a new key to each trusted node individually. In ours, each eNB computes the new group key and reports this to ISC to confirm. In addition, eNB broadcasts the new group key encrypted with the old group key to further decrease the communication messages. In PCGR, the overhead increases with the number of nodes that distribute the group keys. In our mechanism, the total messages for rekeying is two broadcast messages, one for share request, the other one for new key broadcasting, and one unicast message of each device for sending the key share to respective eNB. Because each device has energy constraint owing to the mobility, decreasing the communication overhead of mobile device is very important for mobile IPTV service. Because two broadcast messages are delivered to all devices, the devices check if the message is for itself or not, and can ignore it when the message is not for itself. Especially, when the number of devices increases, communication overhead in centralized or PCGR rises in accordance with the increasing number of devices, while our proposal only requires as many unicast messages as the number of additional devices no matter how many devices exist. It means that our proposal has advantage in large scale network. In temporary group key method, until normal group key rekeying is triggered, the very small number of messages is required, and this further decreases the communication overhead.

Every node in PCGR has to gather the key shares from neighbor nodes, as well as returning the share of its own to every neighbor node, and every one of them needs to compute the group key for itself. In our proposal, eNBs request key shares to their member nodes periodically or on membership change, and the neighbor nodes reply to this request. After eNB verifies the shares from member nodes, it computes the new group key and rebroadcasts it. When the number increases, our mechanism takes less time than [12] or [14], whose rekeying time increases in proportion to the number of nodes as in Fig. 10. This is very efficient when the scale of the network spans.

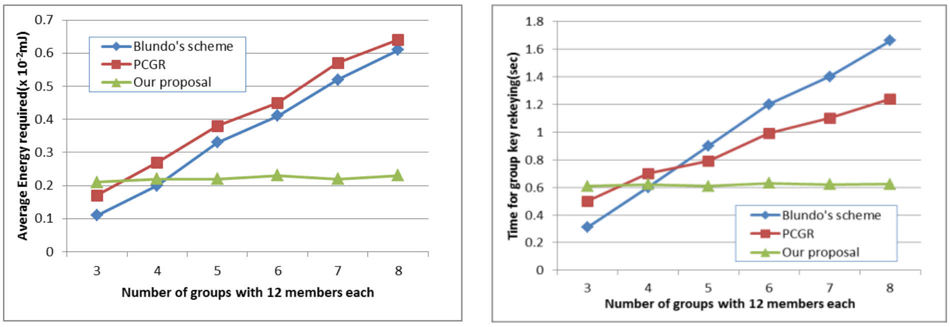


Fig. 10. Group key rekeying overhead (a) Energy consumption (b) Time overhead

In open IPTV, the subscribers communicate with each other for content sharing with or without the help of the other devices such as eNB. Fig. 11(a) shows that the communication time for a pair of users in open IPTV service. Communication time varies according to where each subscriber is located. The overhead when they are located in the same cell is getting shorter, and if they are in their direct M2M communication with each other, it is drastically short. In Fig. 11(b), we can see that the number of packets for rekeying is getting smaller as the rekeying period gets longer, which further decreases the overhead.

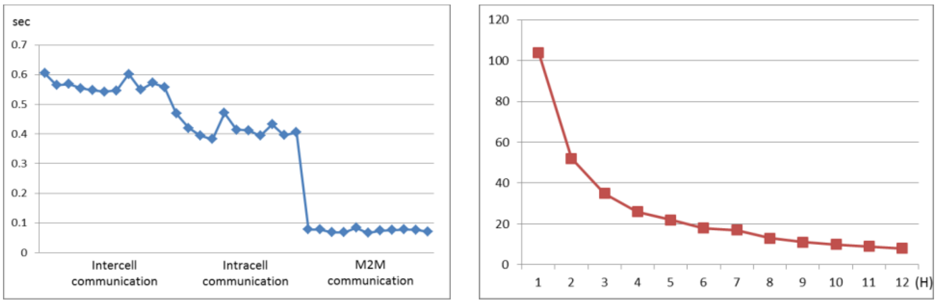


Fig. 11. Communication and rekeying overhead (a) Time for communication between devices as a function of distance (b) Number of packets as a function of group key rekeying frequency in 24 hours.

Computation Overhead eNBs need relatively more computation than that of the devices. eNBs need one decryption and one encryption and key computation for one rekeying process. Decryption is for getting the share from the devices and encryption is required when distributing the new group key. Share verification does not cost much because only simple mod operation is required for each share. Another important cost is for key computation. After receivingμ+*1* or more shares, the eNB needs to solve a (μ +*1*)-variable linear equation system to compute *e(c,u)*, and the computational complexity of using Gaussian elimination to solve the equation system is *O(μ ^3^)* multiplication/divisions. Devices require only one encryption and one decryption. Encryption is required before sending the share of each device, and decryption is for getting the new computed group key delivered by eNB. For the encryption and decryption, any kind of light encryption/decryption algorithm can be used. Our proposal especially requires less computational overhead for devices, which is proper for mobile IPTV service. Still, using temporary group key, much computation overhead can be decreased at the expense of temporal security degradation.

Storage Overhead In our proposal, each node stores as many e-polynomials as the number of channel groups it is being serviced. Each eNB stores as many *g'(x)* as the number of channels that the devices in its own cell are subscribing to. The number of channels that the eNB needs to support may differ from the number of devices in the cell. If many devices with small number of channels exist, the storage performance degrades. If the length of the coefficient is *L* and the number of channels a device watching is *n*, the node needs *L*n*(t+1)* bits. In the same sense, the storage requirement for an eNB is *L*n'*(t+1)* when *n'* is the number of channels that the eNB needs to relay. In basic PCGR, the node in each group requires *g'(x)*, which is *L*(t+1)* bits, and for shares from the neighbor nodes, it needs *n*L*(t+1)*.When the number of nodes is *N*, and the storage overhead is *N*L*(t+1)(n+1)*.

5.2. Security Analysis

Security Level When temporary group key is adopted, security level becomes temporarily low and these keys cannot be used very long. Because of the energy efficiency, when small number of devices change their memberships, temporary group keys are used as in subsection 3.4. In that case, as membership change ratio increases, the security vulnerability decreases. So THstd setup is very import to keep the security level at moderate level.

Access Control Basically, every node needs to register to get the IPTV service at the initial stage, and all communication is secured using group keys depending on each channel. When membership changes, new group keys are generated and distributed for secure and proper service for authentic users.

Intrusion Detection Security system should detect when devices or eNB are attacked by adversaries. Our proposal can identify if the devices are compromised by verifying the shares. Of course, in PCGR, group key rekeying nodes are not determined and if compromised nodes are requested the secret share, they return the information they just have. However, if in that case, they can be clever enough not to make any response for not being detected by their neighbor nodes (PCGR - selective reply). Then we cannot detect if they are compromised. Our proposal detects the compromised nodes, because the eNB randomly selects the devices to answer the requests and checks the authenticity of the device as in 3.3. Compromised nodes can be detected much better via our proposal. In our proposal, however, the success ratio also drops less than 80%, when there are more than 40% adversaries, which is not normal situation.

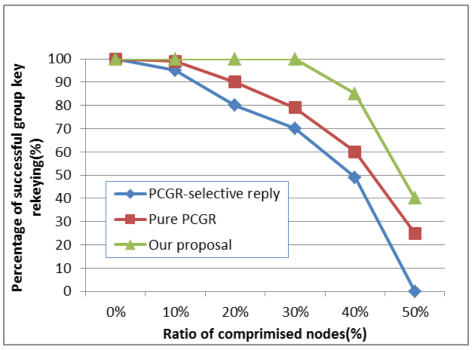


Fig. 12. Percentage of successful group key rekeying

Forward and Backward Secrecy Our proposal guarantees forward and backward secrecy. For backward secrecy, when some devices newly join the channel group, the eNBs report this to ISC; they respond this situation with adopting temporary group keys until normal group rekeying is triggered. For forward secrecy, when a node leaves the group, the temporary group key is also rekeyed, and the leaving node cannot decrypt the messages generated after it leaves.

Availability By filtering the wrong shares from neighbor sensor nodes, clusterheads can get the authentic shares and generate a proper new group key. With this filtering process, we can prevent wrong group key rekeying and wasting unnecessary system resources. In open IPTV, we can further prevent replay attack by adopting timestamp in the packet to protect availability.

Man-In-The-Middle (MITM) Attack In MITM attack, the attacker intercepts the messages between two endpoints and forges or modified them. In our proposal, every pair of devices share pairwise keys with each other, and even the adversaries capture the information in the middle, they cannot forge or modify the data. Even if an attacker captures encrypted data, it is very hard to decrypt them without knowing the pairwise keys.

6. Conclusion

In this paper, we proposed an enhanced group key management mechanism for securing mobile open IPTV. When mobility and openness are added to IPTV technology, key management for traditional IPTV is not proper to apply. Especially, when memberships change often, key updates are required more often. Our proposal basically supports device mobility and membership changes in providing security to IPTV service. Based on the assigning channel group keys to each channel service in cellular environment and updating the keys considering the membership status and user mobility, we additionally enhance the mechanism considering the group key rekeying conditions based on the threshold. Our proposal also provides secure open IPTV service communication by establishing pairwise keys between devices. For our future work, we are planning to simulate our proposal and additionally analyze the security aspects.

Acknowledgment. This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012R1A1A3019459). We thank Li Shi for her contribution in the experiment.

References

1. A. Pinto, M. Ricardo, “On performance of group key distribution techniques when applied to IPTV services,” Computer Communications, Elsevier, 2011.
2. A. N. El-Kassar, R.A. Haraty, “ElGamal Public-Key Cryptosystem in Multiplicative Groups of Quotient Rings of Polynomials over Finite Fields,” Vol.2, ComSIS, June, 2005.
3. I. Doh, J. Lim, K. Chae, “An Improved Security Approach based on Kerberos for M2M Open IPTV System,” In Proceedings of the NeoFusion, Sep., 2012.
4. M. Cedervall, U. Horn, Y. Hu, I. M. Lvars and T. Nasstrom, Open IPTV forum - Toward an open IPTV standard, Ericsson Review, no.3, 2007.
5. F. K. Tu, C. S. Laih and H. H. Tung, “On key distribution management for conditional access system on pay-TV system,” IEEE Trans. Consumer Electron., vol. 45, no. 1, pp. 151-158, Feb., 1999.
6. F. Hartung, S. Kesici, D. Catrein, “DRM protected dynamic adaptive HTTP streaming,” 2nd annual ACM conference on Multimedia systems, pp. 23–25, Feb., 2011.
7. I. Doh, J. Lim, M. Y. Chung, “Group Key Management for Secure Mobile IPTV Service,” In Proceedings of the IMIS, July, 2012.
8. S. O. Hwang, “Content and service protection for IPTV,” IEEE Trans. Broadcasting, vol. 55, no. 2, June, 2009.
9. D. Proserpio, D. Diaz-Sanchez, F. Almenárez, A. Marín, and R. S. Guerrero, “Achieving IPTV Service Portability through Delegation,” IEEE Trans. Consumer Electron., vol. 57, no. 2, pp.492-498, May, 2011.
10. D. Diaz-Sanchez, A. Marín, F. Almenarez and A. Cortes, “Sharing conditional access modules through the home network for Pay TV Access,” IEEE Trans. Consumer Electron., vol. 55, no. 1, pp.88-96, Feb., 2009.
11. D. Diaz-Sanchez, F. Sanvido, D. Proserpio and A. Marín, “DLNA, DVB-CA and DVB-CPCM integration for commercial content management,” IEEE Trans. Consumer Electron., vol. 56, no. 1, pp.79-87, Feb., 2010.
12. Carlo Blundo, Alfredo De Santis, Amir Herzbeerg, Shay Kutten, Ugo Vaccaro, Moti Yung, “Perfectly-Secure Key Distribution for Dynamic Conferences,” Information and Computation, 1995.

13. Y. Wang, B. Ramamurthy, Y. Xue, “Group Rekeying Schemes for Secure Group Communication in Wireless Sensor Networks,” In Proceedings of the IEEE International Conference on Communications 2007.
14. W. Zhang, G. Cao, “Group Rekeying for Filtering False Data in Sensor Networks: A Predistribution and Local Collaboration Based Approach,” IEEE Infocom 2005.
15. J. H. Huang, J. Buckingham, R. Han, “A Level Key Infrastructure for Secure and Efficient Group Communication in Wireless Sensor Networks,” In Proceedings of the International Conference on Security and Privacy for Emerging Areas in Communications Networks 2005.
16. S. Zhu, S. Setia, S. Jahodia, “LEAP+: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks,” ACM Transactions on Sensor Networks 2006.
17. P. Adusumilli, X. Zou, B. Ramamurthy, “DGKD: Distributed Group Key Distribution with Authentication Capability,” In Proceedings of the IEEE Workshop on Information Assurance and Security 2005.
18. R. Aparna, B.B. Amberker, “Key management scheme for multiple simultaneous secure group communication,” In Proceedings of the IEEE Internet Multimedia Services Architecture and Applications (IMSAA) 2009.
19. Y. Kim, A. Perrig, G. Tsudik, “Tree-based group key agreement,” ACM Transactions on Information and System Security (TISSEC) 2004.
20. Z. Yu, Y. Guan, “A Robust Group-based Key Management Scheme for Wireless Sensor Networks,” In Proceedings of the IEEE Communications Society 2005.
21. D. Wallner, E. Harder, and R. Agee, “Key Management for Multicast: Issues and Architectures,” RFC 2627, June, 1999.
22. H. Hugh, C. Muckenhirn, and T. Rivers, “Group Key Management Protocol Architecture,” RFC 2093, Internet Engineering Task Force, Mar., 1997.

Inshil Doh received the B.S. and M.S. degrees in Computer Science at Ewha Womans University, Korea, in 1993 and 1995, respectively, and received the Ph.D. degree in Computer Science and Engineering from Ewha Womans University in 2007. From 1995-1998, she worked in Samsung SDS of Korea to develop a marketing system. She was a research professor of Ewha Womans University in 2009~2010 and of Sungkyunkwan University in 2011. She is currently an assistant professor of Computer Science and Engineering at Ewha Womans University, Seoul. Her research interests include wireless network, sensor network security, and M2M network security.

Jiyoung Lim received the B.S. and M.S degrees in Computer Science at Ewha Womans University, Korea, in 1994 and 1996, respectively and received the Ph.D. degree in Computer Science and Engineering from Ewha Womans University in 2001. She is currently an associate professor of Computer Software at Korean Bible University, Seoul, Korea. Her research interests include wireless/sensor network security, and M2M network security.

Kijoon Chae received the B.S. degree in mathematics from Yonsei University in 1982, an M.S. degree in computer science from Syracuse University in 1984, and a Ph.D degree in Electrical and computer engineering from North Carolina State University in 1990. He is currently a professor of Computer Science and Engineering at Ewha Womans University, Seoul, Korea. His research interests include network security, sensor network, network protocol design and performance evaluation.

Received: September 17, 2012; Accepted: March 22, 2013