Chapter 6

Key Management in Mobile WiMAX Networks

Mohammad-Mehdi Gilanian-Sadeghi,
Borhanuddin Mohd Ali and Jamalul-Lail Ab Manan

Additional information is available at the end of the chapter

http://dx.doi.org/10.5772/56154

1. Introduction

Wireless networks, because of their many advantages in comparison with the wired ones, have become the predominant technology for deployment of communications infrastructure. WiMAX (Worldwide Interoperability for Microwave Access), which is an industry branding for IEEE 802.16 Wireless Metropolitan Area Network (MAN) sets of standard [1, 2], provides wireless access to mobile devices with a range of Quality of Service (QoS) guarantees for various types of applications. There are diverse versions of IEEE 802.16 standard, but IEEE 802.16e [3] also known as Mobile WiMAX, is the most well-known version, though newer versions have also been formulated.

As for the security model of IEEE 802.16, it has been designed to guarantee authentication, confidentiality and integrity. Among the series of IEEE 802.16 standards, the IEEE 802.16d [4] was defined for fixed wireless access. It uses Privacy Key Management Version 1 (PKMv1) to define, manage and distribute the security keys, but there are several security issues in PKMv1. Hence, in IEEE 802.16e, an enhanced key management scheme called Privacy Key Management Version 2 (PKMv2) was introduced to mitigate the security shortcomings of PKMv1. The PKMv2 uses Extensible Authentication Protocol (EAP) [5] and RSA algorithm [6] as authentication methods. The authentication mechanism ensures that when a Mobile Station (MS) enters a Base Station (BS) coverage area, it should perform authentication and authorization in order to obtain the security keys that will protect data more securely.

The rest of this chapter is organized as follows. Section 2 elaborates the main concept of WiMAX architecture which focuses mainly on the security parts. Section 3 reviews key management protocols in Mobile WiMAX. Finally, section 4 presents our conclusion and suggestions for future works.

2. WiMAX architecture

2.1. Protocol stack

The protocol stack of IEEE 802.16 standard consists of two main layers: Medium Access Control (MAC) layer and Physical (PHY) layer [2]. The MAC layer is subdivided into three sub-layers [7], namely it (CS), Common Part Sub-layer (CPS) and Security Sub-layer (SS) as shown in Figure 1.

The service specific convergence sub-layer communicates with higher layers and receives packets from them and then do some specific functions like packet/frame classification and header suppression. Next, it encapsulates these packets into MAC Service Data Unit (MAC SDU) format, and then distributes MAC SDUs to common part sub-layer. Asynchronous Transfer Mode (ATM) convergence and packet convergence sub-layers are two types of service specific convergence sub-layer. The ATM convergence sub-layer is used for ATM networks, and the packet convergence sub-layer is used for packet services like Ethernet, IPv4 and IPv6.

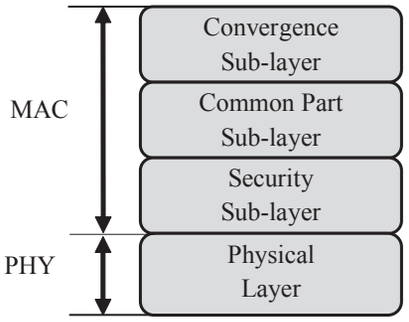

Figure 1. Protocol stack of IEEE 802.16 [2]

The main part of the IEEE 802.16 standard is common part sub-layer which is responsible for bandwidth allocation, connection management, scheduling, connection control, automatic repeat request and QoS enforcement.

The security sub-layer is responsible for providing authentication, authorization and secured key exchange. It is also used for encryption and decryption of data from the MAC layer to PHY layer and vice versa. Two main protocols of security sub-layer are [3]:

1. Encapsulation Protocol, which is used for ciphering operations on data in the networks,
2. PKM protocol, which is used for secure key distribution between BS and MSs, and also it enables the BS to enforce conditional access to network services.

The PHY layer receives MAC frames and then transmits them through coding and modulation of radio frequency signals. It supports Frequency Division Duplexing (FDD) and Time Division Multiplexing (TDM).

2.2. Security sub-layer

The architecture of security sub-layer is shown in Figure 2. As mentioned previously, the security sub-layer provides security services for the standard, and it has been made based on two main components; an encapsulation protocol and a key management protocol [3]. The encapsulation protocol introduces the encryption and authentication methods as cryptographic suites which is a pair of encryption and authentication algorithms.

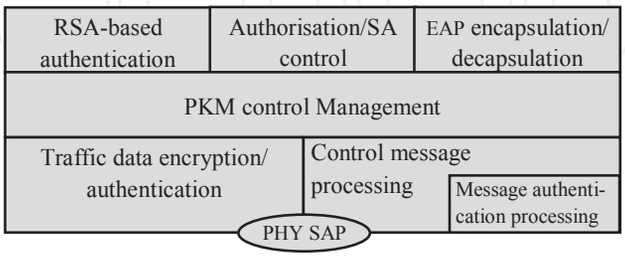

Figure 2. Security sub-layer architecture [3]

Initially, WiMAX security was introduced in the security sub-layer of IEEE 802.16 standard [1]. After releasing the initial versions of the IEEE 802.16 standard, a number of articles such as in [8-10] criticized the security weaknesses of the standard, after which some security improvements were added in IEEE 802.16e [3] and IEEE 802.16m [11]. The security functions regarding key managements have been addressed by PKM protocol. In IEEE 802.16d [4], the key management is based on PKMv1 while IEEE 802.16e uses PKMv2, which is an enhanced version of PKMv1.

Generally, PKM protocol is responsible for authorization, authentication, key exchange and data encryption in the networks between the MSs and BS. In the subsequent sections, we focus our attention on PKMv2, because it is stronger than PKMv1 in terms of security. Recently, the PKMv3 [11] was launched with IEEE 802.16m standard, however, since this protocol is still new and only a few works are being done on it, it is not discussed further in this chapter.

The PKMv2 is used by MSs to get authorization and security keys from the BS, and also to guarantee continuous and uninterrupted re-authorization/re-authentication and refreshing of the security keys. The PKMv2 applies EAP protocol together with RSA algorithm or a mixed function starting with RSA followed by EAP. As shown in Figure 3, in EAP of PKMv2, the root of the security keys is Master Session Key (MSK), and the other keys such as Key Encryption Key (KEK) are obtained from the MSK.

The procedure of security keys generation using the EAP method is shown in Figure 3. In this Figure, the result of EAP authentication protocol is MSK. Then both the MS and BS make a Pairwise Master Key (PMK) by removing some bits of the MSK using a number of functions such as Dot16KDF [12], and also they generate an Authorization Key (AK) from the PMK. After making the AK, the BS and MS will establish the Key Encryption Key (KEK) from the AK. The BS and MS use a 3-way handshake to drive Traffic Encryption Key (TEK) which is used to encrypt data in the network between the BS and MSs. The Multicast Broadcast Service (MBS)

is then applied when there are several MSs whereby the MBS is used to send the messages to them. In this case, both BS and MS need to generate and use some group keys.

IEEE 802.16 supports multicast for applications such as pay-tv and videoconferencing. In order to establish a secure multicast over IEEE 802.16, main components of the standard must be used, namely Multicast Broadcast Service (MBS) and Multicast and Broadcast Rekeying Algorithm (MBRA). We will explain how this is done in the next section.

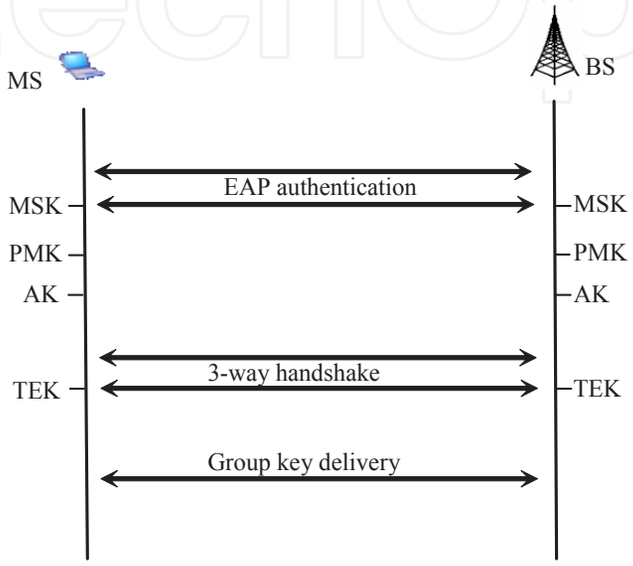

Figure 3. Key generation at initial network entry [12]

3. Key management protocols

There are three types of Key management protocols, viz, centralized, hierarchical and distributed key management [13]. WiMAX Network uses a centralized key management where there is a single manager (BS) that executes key management procedure for all its members (MSs). Though some key management protocols have been proposed for WiMAX, their protocols still remained inefficient.

Generally, key management establishes a set of group keys for its members [14], and the main function of it is to update the group keys, this is called rekeying algorithm [15]. The key management protocols have to face several challenges, but the most outstanding challenges among them are on performance and security, as shown in Figure 4. Under performance are issues such as operational efficiency, scalability and 1-affects-n phenomenon [16, 17]. Operational efficiency is the most important parameter in performance measure and is measured typically by storage, communications and computational costs respectively. In measuring the performance of key management, the storage costs refer to the number of keys stored by the BS and MSs; the communications costs refer to the

number of transmitted group keys upon a rekeying algorithm, and the computational costs refer to the cost of ciphering operations in order to get the updated group keys. Scalability means the capability of key management protocol to handle a large group of members, and also its ability to manage highly dynamic membership changes. The 1-affects-n phenomenon is estimated from the number of members affected by rekeying operations. Moreover, a key management should support forward secrecy, which means that the MSs that leave a BS cannot read future messages; and also it must guarantee backward secrecy, which means that a new MS cannot read previous messages [9].

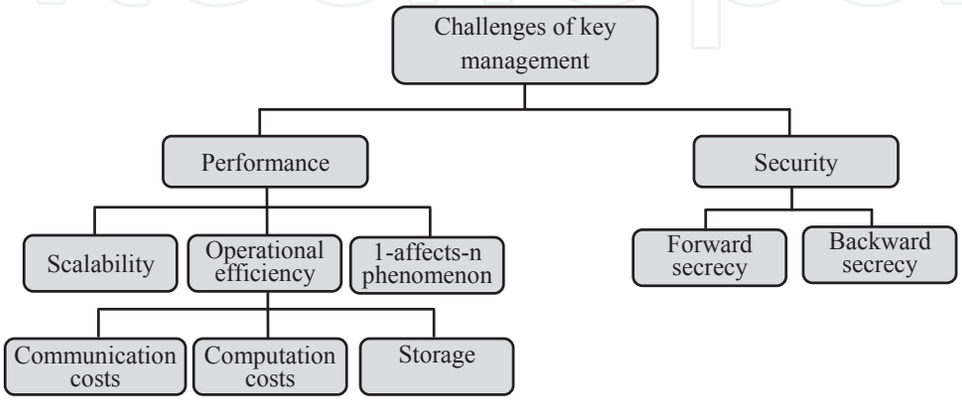

Figure 4. Key management's challenges

3.1. Multicast and broadcast service

Multicast and Broadcast Service (MBS) of IEEE 802.16e is a new feature for broadband wireless standards [3]. It is a mechanism that allows a BS to distribute the same set of data to several MSs concurrently. As highlighted before, first the MSs need to be authenticated by the BS using PKMv2 [3]. After that, the Group Key Encryption Key (GKEK) and the Group Traffic Encryption Key (GTEK) are established. IEEE 802.16e introduced the MBRA as a basic rekeying algorithm to generate, update and distribute the GKEKs and GTEKs upon member changes. The MBRA uses the GTEK which is shared among all MSs to encapsulate the data traffic. The BS generates the GKEK and the key is used to encapsulate the GTEK. The GKEK is also encapsulated by the KEK of each MS. Each MS has a unique KEK which is obtained from the AK. Although, the MBRA of MBS is quite well designed, it still suffers from efficiency and scalability problem and it does not address backward and forward secrecy [8, 18]. To explain this point, in the MBRA algorithm, the BS should unicast n messages, where n is the number of MS, with the aim of updating the group keys, which unfortunately would cause weak scalability due to the increased number of unicast messages. Moreover, when there are high numbers of MSs, and the effect of sending high volume of unicast/broadcast messages would increase communication costs, and consequently this will result in poor efficiency.

Rekeying algorithms in WiMAX networks need to execute using one of the following three events:

1. when a new MS joins the BS,
2. when the life time of both GTEK and GKEK expire,
3. when an MS leaves the BS.

The MBRA algorithm of Mobile WiMAX which is a simple rekeying will only happen at the expiration time of GTEK or GKEK. As shown in Figure 5, from time to time, the BS broadcasts message (1) encrypted by GKEK to all MSs in order to update the GTEK as well as sending a unicast message (2) to all MSs which has been encapsulated by the KEK of each MS as shown by the equations below:

$$BS \Rightarrow all\ MS : \{GTEK\}_{GKEK} \tag{1}$$

$$BS \rightarrow each\ MS : \{GKEK\}_{KEK} \tag{2}$$

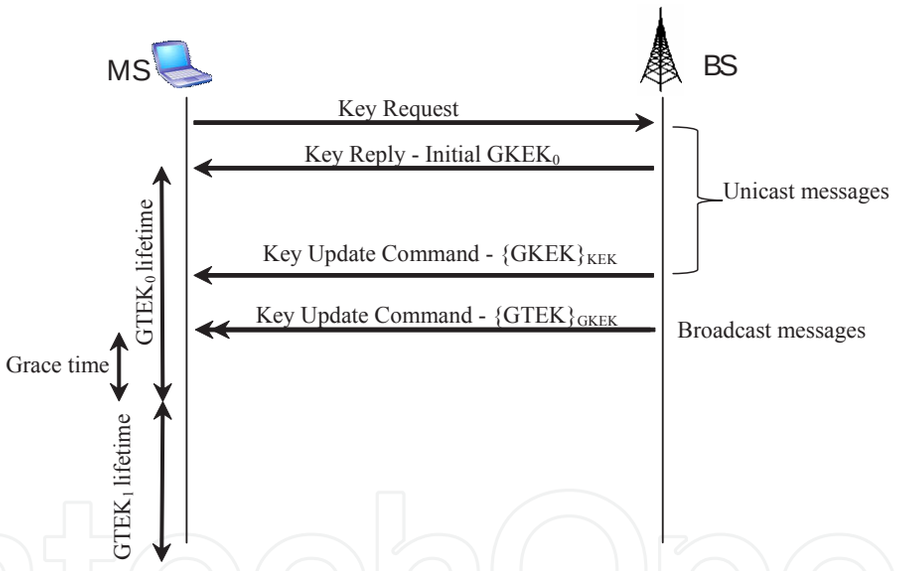

Figure 5. MBRA messages [19]

The nomenclatures are listed as in Table 1.

X⇒Y	X broadcasts a message to Y
X→Y	X unicasts a message to Y
$[X]_Y$	X encrypted by using key Y
MS_{SGi}	The collection of all MSs within $subgroup_i$

Table 1. Nomenclature of key management

3.2. Rekeying algorithms

As mentioned earlier, in the MBRA algorithm, the number of unicast messages on rekeying increase with the number of MS, and hence this method is neither scalable nor efficient. In addition, it does not address forward and backward secrecy, which consequently would lead to this method being vulnerable to attacks [8, 9].

Researchers in [18] performed a detailed analysis of the MBRA algorithm and identified its deficiencies. They proposed an improved scheme to address the deficiencies identified. Even though their method showed some improvements on the MBRA, but they suffer a downside in that, the BS needs to send n (n being the number of MS) unicast messages upon every membership changes, which consequently resulted in the drastic drop in network efficiency for a large number of MSs. In addition, the proposed method also sends some plaintexts for message broadcasting, which could cause critical security breaches [19]. In fact, despite some improvements to the MBRA rekeying, the proposed method suffers from some security issues such as Denial of Service (DoS) [19] as well as poor scalability and efficiency. In addition, it does not address the 1-affects-n phenomenon [17], very well.

The authors in [20] proposed a new group key management protocol called Group-Based Key Distribution Algorithm (GKDA) in which the security keys are distributed into subgroups. The MBS group is first divided into N subgroups; hence, N GKEKs for the subgroups are used instead of one GKEK being shared among all MSs. By doing so, only that GKEK which is used for a certain subgroup needs to be updated whenever any membership change (e.g. leave event) occurs in that subgroup. The GKEK is encapsulated by each MS's KEK in the subgroup, and then unicast to each MS. Although the GKDA provides forward and backward secrecy, it is still not scalable and efficient enough, because when the number of MS in each subgroup grows bigger, the number of unicast messages to update GKEKs grows likewise. Nevertheless, GKDA is still better than MBRA in terms of reducing the number of unicast messages needed to perform updates of the group keys. In GKDA, the GTEK update mode is more lengthy because it consists of N GTEKs which are encapsulated by N GKEKs, and thus it consumes more energy to send the messages. Moreover, the scheme does not have a good support for 1-affects-n phenomenon.

In [21], the authors proposed an algorithm called Efficient sub-Linear rekeying Algorithm with Perfect Secrecy (ELAPSE) in order to address the problems of MBRA algorithm. Although this method solves the forward and backward secrecy problems, it suffers from some weaknesses in terms of scalability and efficiency. In ELAPSE, when member join or leave events happen frequently within a large group, the overall performance will degrade due to communication and computational costs. This method is based on key hierarchy and sub-grouping of the MSs in the cell by means of a binary tree. ELASPE divides the number of MS into N=log (n,2) subgroups where n is the number of MS, and each subgroup keeps a set of hierarchical keys named Sub Group KEKs (SGKEKs) instead of a single GKEK. The number of subgroups (N) is defined in advance by the administrator depending on the application's requirements, i.e. the number of subgroups is permanent. The result is weak performance in terms of efficiency and scalability. We illustrate this issue by way of an example as shown in Figure 6, which shows a binary tree with four subgroups. All MSs maintain similar GTEK, and each MS in each

subgroup saves a set of SGKEKs; for example, the MSs in $subgroup_1$ store three group keys $SGKEK_1$, $SGKEK_2$ and $SGKEK_{1234}$. The $SGKEK_{1234}$ is similar with the GKEK in MBRA. In this case, GKEK is not delivered to each MS by unicast message, instead it is distributed among the subgroups via broadcast messages.

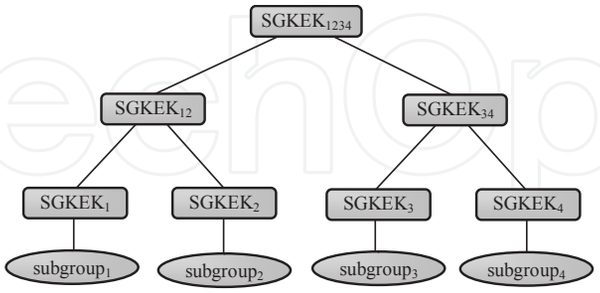

Figure 6. Key Hierarchy with four subgroups [21]

When there is no new member joining or leaving, and the lifetime of GTEK expires, the BS broadcasts a new GTEK encapsulated by $SGKEK_{1234}$ to all MSs represented as message (3) below.

$$BS \Rightarrow all\ MSs : \{GTEK\}_{SGKEK_{1234}} \tag{3}$$

Upon a member join event, i.e. when a new MS enters into the BS coverage area, and $subgroup_2$ has the lowest number of members, then the BS assigns it to $subgroup_2$. The BS unicasts message (4) below to the new MS and all MSs in $subgroup_2$ in order to update the group keys. Message (4) is then encapsulated by KEK of each MS, and contains all new group keys from $subgroup_2$ to the root of binary tree.

$$BS \rightarrow MS_{SG2}\ \&\ new\ MS : \{GTEK, SGKEK_{1234}, SGKEK_{12}, SGKEK_2\}_{KEK} \tag{4}$$

In order to update the group keys as well as to provide the backward secrecy, the BS needs to send two broadcasts i.e. messages (5) and (6) below, to all MSs excluding $subgroup_2$.

$$BS \Rightarrow MS_{SG3}, MS_{SG4} : \{GTEK, SGKEK_{1234}\}_{SGKEK_{34}} \tag{5}$$

$$BS \Rightarrow MS_{SG1} : \{GTEK, SGKEK_{1234}, SGKEK_{12}\}_{SGKEK_1} \tag{6}$$

Upon member leave event, i.e. when a MS leaves the BS coverage area, the process of the group key updating is similar to member join event. For instance, when one MS of $subgroup_2$ leaves

http://dx.doi.org/10.5772/56154

the BS, then the BS should unicast message (7) to all remaining MSs in subgroup$_2$. It also needs to broadcast two messages, i.e. messages (8) and (9), to all MSs except subgroup$_2$.

$$BS \rightarrow MS_{SG2} : \{GTEK, SGKEK_{1234}, SGKEK_{12}, SGKEK_2\}_{KEK} \tag{7}$$

$$BS \Rightarrow MS_{SG3}, MS_{SG4} : \{GTEK, SGKEK_{1234}\}_{SGKEK_{34}} \tag{8}$$

$$BS \Rightarrow MS_{SG1} : \{GTEK, SGKEK_{1234}, SGKEK_{12}\}_{SGKEK_1} \tag{9}$$

Authors in [22] suggested an improved version of ELASPE called ELAPSE+ using cross layering concept. They assigned fast moving MSs such as cars to specific subgroups, and made the size of those specific subgroups to be smaller than the other subgroups. This is because, the fast moving MSs pass through the BS's cell length faster, and therefore they would experience high number join or leave events, which gives rise to the need to update more group keys. Although ELAPSE+ improves the performance of ELAPSE by reducing the amount of rekeying messages needed to send unicast and broadcast messages, it still inherits the drawback of handling static numbers of subgroups, subsequently resulting in weak efficiency and scalability.

The authors in [23] proposed a hybrid key management scheme to improve the performance of ELAPSE and ELASPE+ upon rekeying by reducing message passing. This scheme uses the architecture of LORE [23] within a subgroup of ELAPSE. In this way, when a MS enters a BS coverage area, the BS assigns it to a subgroup and also provides a Subgroup Forward Key Set (SGFSet) and Subgroup Backward Key Set (SGBSet). These key sets are created by simple Pseudo-Random Generator (PRNG) and keep the ordering of nodes inside a subgroup similar to LORE. Hence, if there are k MSs in a subgroup, then there are k numbers of Subgroup Forward Key (SGFK) and k numbers of Subgroup Backward Key (SGBK). In this way, for each MS i there are two sets of keys as follows:

$$SGFSet = \{SGFK_m \mid 1 \le m \le i\} \tag{10}$$

$$SGBSet = \{SGBK_m \mid i \le m \le k\} \tag{11}$$

Figure 7 shows the revised version of ELAPSE. Here, a node i in subgroup$_2$ has three keys $SGKEK_{1234,}$ $SGKEK_{12}$ and $SGKEK_2$ as well as a two-key set $SGFSet^i{}_2$ and $SGBSet^i{}_2$. Upon member join or leave event, the rekeying algorithm updates SGKEKs and GTEK, but there is

no change in SGFSet and SGBSet sets. After a predefined time T, both SGFSet and SGBSet will be renewed.

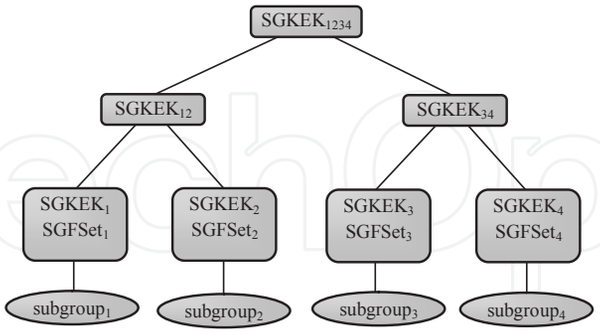

Figure 7. A revised version of ELAPSE [23]

It should be noted here that this improvement in communication costs over ELAPSE, comes at high computational and storage costs in the revised version of ELAPSE. Moreover, this scheme gives rise to security issues such as collusion resistance [23] which means two or more MSs must not get secret keys that they are not allowed to know, and this could be done by exchanging their respective secret keys.

The authors in [24] improved ELAPSE by using a n-ary tree (where n>2) to improve the efficiency of key management. Even though the proposed method shows some improvements on the efficiency of ELAPSE, the method still suffers from the limitations associated with fixed number of subgroups. In this method, the tree depth becomes large when the number of MS increases, and this is the main issue with a binary tree. Therefore, they suggested that by using n-ary, the efficiency of group key updating algorithm will improve. Figure 8 shows a 3-ary tree with 9 subgroups.

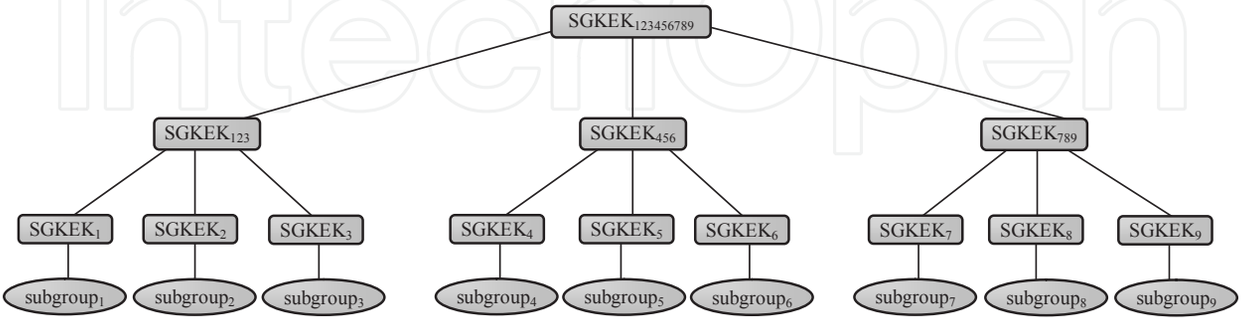

Figure 8. A 3-ary tree [24]

The number of group keys in n-ary tree and the tree depth are given by equation (12) and (13) respectively.

http://dx.doi.org/10.5772/56154

$$k = \sum_{i=0}^{d} n^i \tag{12}$$

$$d = \left\lceil \log_n^N \right\rceil \tag{13}$$

By using n-ary tree, the BS needs to keep more group keys compared with ELAPSE method. So, in terms of storage costs n-ary tree does not perform very well, even though the communication costs is considerably decreased due to the reduction in communication overheads upon group keys updating. The authors made detailed analysis to find the optimal value of n in order to minimize the total energy consumption of the rekeying algorithm. They assumed that transmission and reception energy are equal to total energy consumption of the networks, whereby the energy consumption refers to the length of broadcast or unicast messages. Finally, they came out with an optimal value of n=4, meaning that 4-ary tree would give the best performance in terms of energy consumption.

It should be highlighted here that basically the methods in [20, 21, 23, 24] are based on ELAPSE in that they all use tree structures, and therefore the problem associated with ELAPSE as highlighted before still remains.

The authors in [19] proposed a new method of improving MBRA using asymmetric algorithms. The idea of this method is to establish a common encryption key which is shared among all MSs, but every MS has a different decryption key. This means that the BS can encrypt the messages including the group keys, and only the valid MS can decrypt the messages. In this way, the proposed method provides backward and forward secrecy. In terms of operational efficiency, this method needs to perform more computations because of the use of asymmetric cryptography, and hence this makes the MSs to expense more energy which is not good for mobile devices. Nevertheless, one advantage of the proposed method is that it sends less unicast/broadcast messages, and hence the overall communication cost is low. In this way, upon member changing, the BS sends one broadcast message, but on normal key refresh, it needs to send n unicast messages, where n is the number of MS and also the BS should send two broadcast messages. The proposed method managed to address the backward and forward secrecy issue of the MBRA algorithm. However, it has poor response to scalability, since upon group key updating after the expiration time, it has to send n unicast messages. Moreover, the method needs to make numerous modifications to the standard, which it is not practical for implementation in real environment.

In [25] the Scalable Rekeying Algorithm (SRA) is proposed, which is based on complete binary tree [26], and is implemented by linear linked list data structure. The SRA method improves the scalability for ELAPSE and it can also improve the other methods [20, 21, 23, 24] which have similar setups. As mentioned earlier, ELAPSE divides the MSs into N subgroups. In this way, each subgroup keeps a set of group keys. In fact, ELAPSE employs a fixed number of subgroups, consequently upon group key updating, the ELAPSE shows poor scalability. In

addition, the method consumes more bandwidth because of the sending of high number of unicast messages.

The SRA method establishes the number of subgroups according to the number of current MS in the cell. Figure 9 shows a sample of node within linear linked list, where "#MS" field indicates all MSs in a certain subgroup. The group key for that subgroup is "Group-key". L1 and L2 are two pointer fields in the node, where L1 points to the MSs of that subgroup and L2 points to the next node (subgroup).

Figure 9. A node of linear linked list

The SRA method uses log (n,2) in order to subgroup the MSs, and whereby according to the current number of MS, it increases or decreases the number of subgroups.

As highlighted before, in Mobile WiMAX, group key updating happens on three events:

1. Upon the expiry lifetime of GTEK/GKEK,
2. Upon member join event,
3. Upon member leave event.

For the first event (i.e. upon the lifetime of GTEK or GKEK expiry), the SRA and ELAPSE methods apply similar functions. However, in SRA method, on member join/ leave, it is necessary to add/delete a subgroup at a certain time to increase or decrease the number of subgroups based on log(n,2).

Assuming that in the first step there is one subgroup as shown in Figure 10, it means that there is a node in the linear linked list. Figure 10 shows a linear linked list structure corresponding to complete binary tree. For the rest of the chapter, the tree is not drawn for the sake of simplicity.

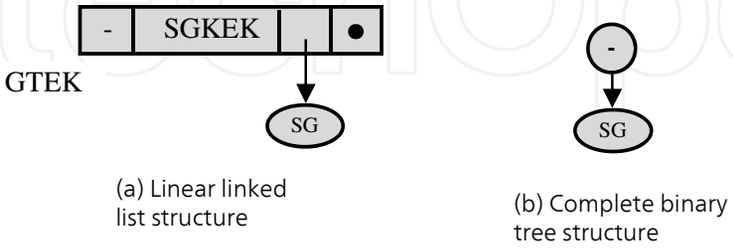

Figure 10. The creation of one subgroup

As the number of MS reaches three, a new subgroup should be added, based on log(3,2)=2. Thus, subgroup SG breaks into 2 subgroups, SG_1 and SG_2. Subsequently, the MSs of SG partition into two different sets, and afterward they are inserted separately into 2 subgroups,

SG_1 and SG_2. In the properties of complete binary tree, if a node is at an index i, the left child is at index 2*i, and the right child is at index 2*i+1. We use these properties of the tree to manage the subgroups. In this way, SG_1 is at index 2 and SG_2 is at index 3. In Figure 11, two subgroups are shown with 1 and 2 MSs in '#MS' field respectively.

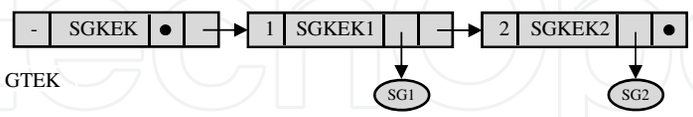

Figure 11. The creation of two subgroups

The BS unicasts two messages i.e. messages (14) and (15) to all MSs with the purpose of updating the group keys. In this way, the BS unicasts $SGKEK_1$ and $SGKEK_2$ to SG_1 and SG_2 respectively.

$$BS \rightarrow MS_{SG1} : \{GTEK, SGKEK_1\}_{KEK} \tag{14}$$

$$BS \rightarrow MS_{SG2} : \{GTEK, SGKEK_2\}_{KEK} \tag{15}$$

As the number of MS increases beyond 5, a new subgroup is added, based on log(n,2). In this case, the new node is added to the left side of the tree; the left hand side's children of the tree are regarded as 2 new subgroups. Next, the MSs of SG_1 divides into 2 parts and then they are associated to 2 new subgroups, viz, SG_{11} and SG_{12} as shown in Figure 12.

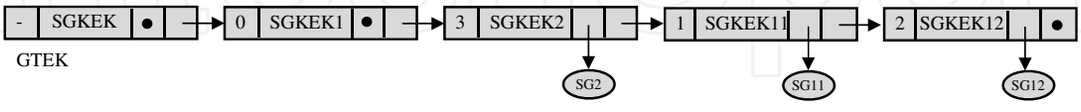

Figure 12. The creation of three subgroups

To update the group keys after inserting one new subgroup, the BS should unicast two messages i.e. messages (16) and (17) to SG_{11} and SG_{12} respectively. Assuming that the BS adds the new MS to SG_{11}, then the new group keys should be unicast to the MS by means of message (16).

$$BS \rightarrow MS_{SG11} \,\&\, newMS : \{GTEK, SGKEK, SGKEK_1, SGKEK_{11}\}_{KEK} \tag{16}$$

$$BS \rightarrow MS_{SG12} : \{GTEK, SGKEK, SGKEK_1, SGKEK_{12}\}_{KEK} \tag{17}$$

If the number of MS is 11 or less, they join three subgroups (Figure 12), but if they exceed 11, one new subgroup must be added. The procedure to add a new subgroup is similar to our explanation for Figure 12. Here, SG_2 divides into two subgroups i.e. SG_{21} and SG_{22}. The entire number of MSs in each subgroup is labeled in '#MS' field of Figure 13, when the 12th MS enters into the BS coverage area.

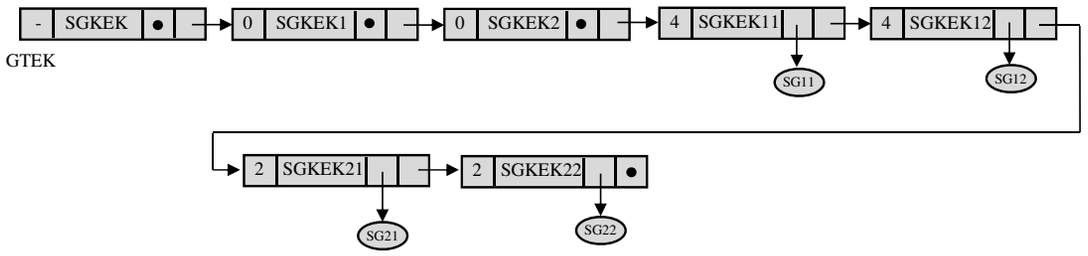

Figure 13. Linear linked list showing the creation of four subgroups

Suppose a few MSs leave a cell, the total number of MS will decrease. As a result, the number of subgroups based on log(n,2) should decrease as well. When the number of MS drops to less than 12, SG_{21} and SG_{22} combine together into one subgroup, i.e. SG_{2b}. Next, the whole MSs in SG_{21} and SG_{22} add into SG_{2b}. When the number of MS stands at 11 the subgroups that exist becomes as shown in Figure 14. The BS unicasts message (18) including 3 new group keys to every MS in SG_{2b} to update the group keys.

$$BS \rightarrow MS_{SG2b} : \{GTEK, SGKEK, SGKEK_{2b}\}_{KEK} \tag{18}$$

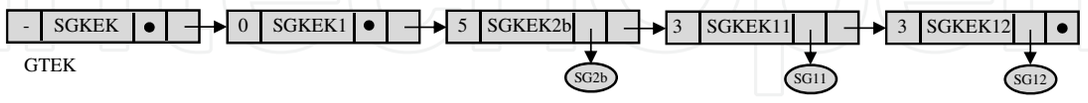

Figure 14. Linear linked list showing the creation of three subgroups

In the forthcoming, the SRA method is compared and analyzed against MBRA [3] and ELAPSE [21]. The MBRA unicasts n messages to all current MSs as well as new MS upon member joining, and upon member leaving, it unicasts n-1 messages (since 1 MS leaves the cell). As mentioned earlier, ELAPSE creates a permanent number of subgroups, therefore when the number of MS in a cell grows, the number of transmitted unicast messages increases likewise. The entire number of transmitted unicast messages in ELAPSE is (n/N); in fact, it is (n/N)+1

for member joining and (n/N)-1 for member leaving. In SRA, the number of MS in each subgroup is n/log(n,2), and therefore the number of unicast messages is likewise n/(log(n,2)) on member joining/leaving. The comparison among the MBRA, ELAPSE and SAR is shown in Table 2, where ELAPSE4 means four subgroups, and ELAPSE8 means 8 subgroups.

Methods	Unicast Messages	
	Join	Leave
MBRA	$O(n)+1$	$O(n)-1$
ELAPSE4	$O\left(n/4\right)+1$	$O\left(n/4\right)-1$
ELAPSE8	$O\left(n/8\right)+1$	$O\left(n/8\right)-1$
SAR	$O\left(\frac{n}{\log_2^n}\right)+1$	$O\left(\frac{n}{\log_2^n}\right)-1$

Table 2. Comparative analysis [25]

Figure 15a & b show the comparison among the rekeying algorithms in terms of unicast messages. Here, y-axis represents the number of unicast messages, and x-axis is the number of MS. As shown in Figure 15a, in the MBRA, the number of unicast messages increase with growing number of MS, and clearly it does not address the question of scalability. Figure 15b shows the analysis among the tree-based rekeying algorithms only. This is also the case with ELAPSE, where the number of unicast messages increases with the number of MS in each subgroup. On the other hand, in SRA the number of unicast messages is less than in ELAPSE, which means that it provides a good scalability even at high number of MS.

Figure 16 shows a magnified view of Figure 15, for the number of MS between 200 to 400 in Figure 16a, and 500 to 700 in Figure 16b, respectively. It is clear from the Figure that in SRA method, as the number of MS increases the number of transmitted unicast messages increases with a much lesser degree than for ELAPSE. In other words, the difference between the number of unicast messages between SRA and ELAPSE widens. For example, when there are 400 MSs (Figure 16a), the difference between the number of transmitted unicast messages in the SRA and ELAPSE8 is around 5, but when there are 700 MSs (Figure 16b), this difference is around 10. This shows that SRA method has a good scalability performance especially at high number of MS in the cell.

Figure 17 depicts a summarised comparison between SRA and ELAPSE. Clearly, SRA reduces the number of unicast messages upon implementing rekeying algorithm, and therefore it has better scalability compared with ELAPSE. Even though ELAPSE8 shows comparable performance with SRA especially at lower number of MS, the number of MS in a subgroup has to be defined in advance and neither it is dynamic.

Finally, Table 3 summarizes the main characteristics of the rekeying algorithms which have been highlighted in this chapter.

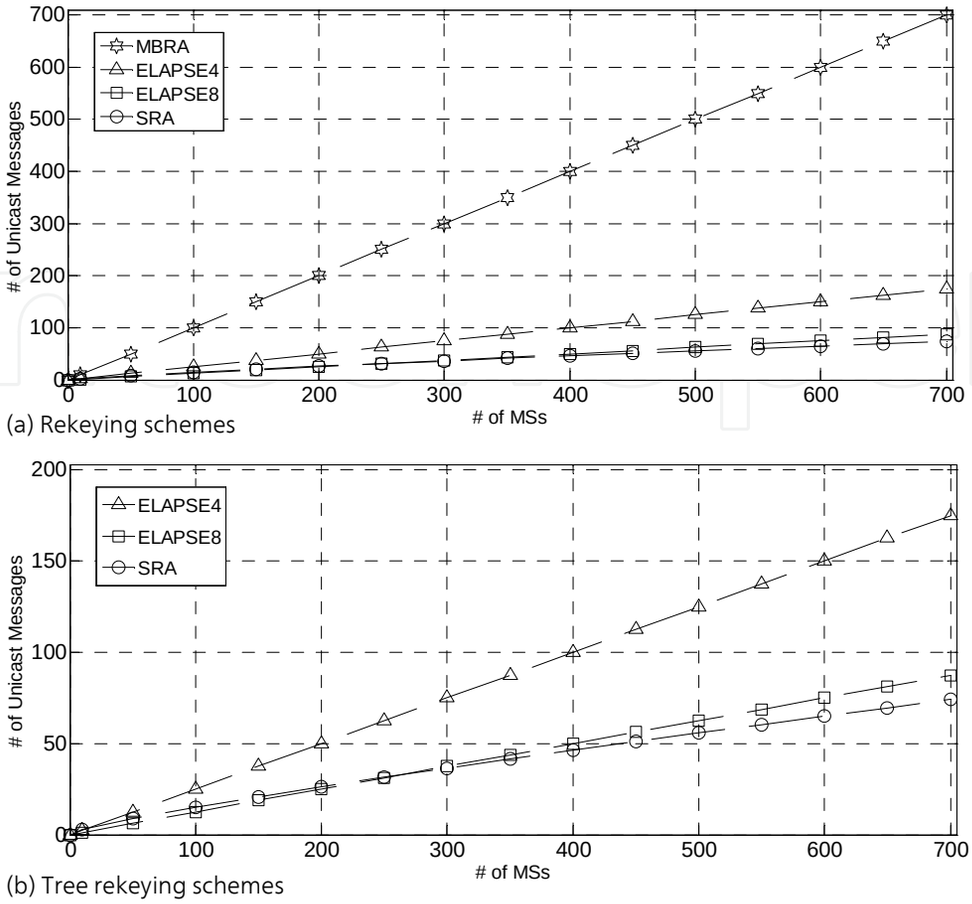

Figure 15. Unicast messages

Scheme	*Forward/Backward Secrecy*	*Scalability & 1-affects-n*	Operational efficiency*
MBRA[3]	not supported	Very weak	Non optimal
Xu et al.[18]	Supported	weak	Non optimal
GKDA[20]	supported	good	Non optimal
Chakraborty et al.[23]	supported	good	Non optimal
Kambourakis et al.[19]	supported	good	Non optimal
ELAPSE [21]	supported	good	Non optimal
Brown et al.[24]	supported	good	Non optimal
SRA[25]	supported	Very good	Near optimal

*This shows the trade-off among communication, computational overheads.

Table 3. Summary of the main performance parameters of rekeying algorithms

http://dx.doi.org/10.5772/56154

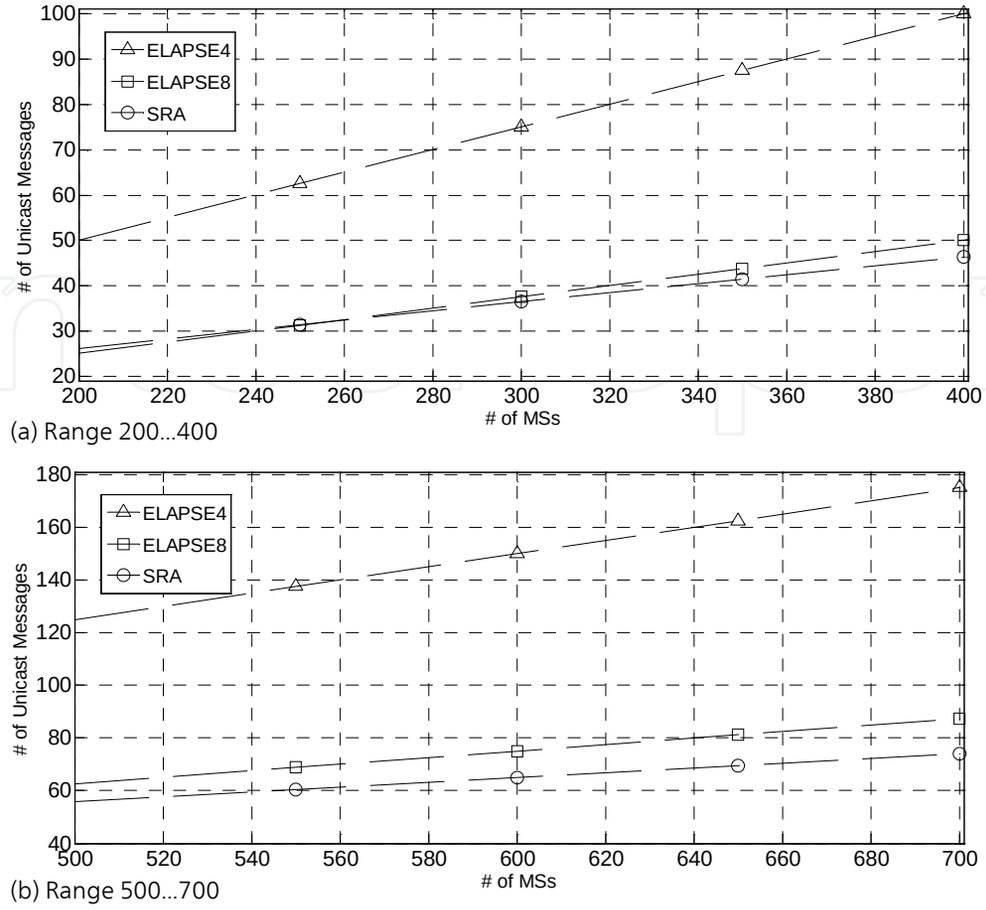

Figure 16. Unicast messages in tree rekeying

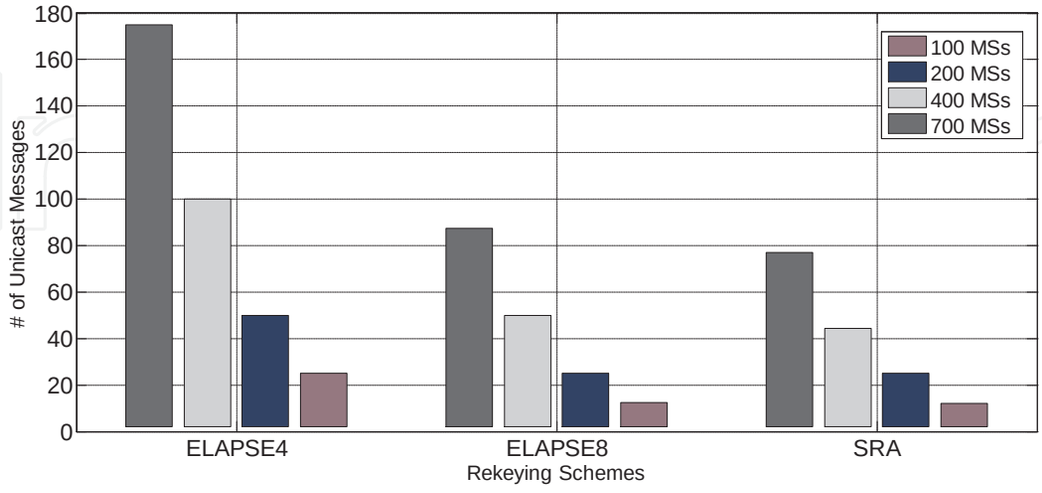

Figure 17. Unicast messages in ELAPSE and SRA

4. Conclusions

In this chapter, we reviewed the MBRA rekeying algorithm of the IEEE 802.16e and analyzed several rekeying algorithms for Mobile WiMAX. We reviewed the rekeying algorithms with emphasis on performance and security particularly their effects on operational efficiency, scalability and 1-affects-n phenomenon as well as backward and forward secrecy. We showed that SRA rekeying algorithm is a strong algorithm from the scalability aspect, because it establishes the number of subgroups dynamically and hence strikes a good balance between the number of MS in each subgroup and the total number of subgroups. The overall result is a reduction in the number of unicast messages on rekeying which produce better scalability and efficiency. The future work will focus on reducing energy consumption in the MSs upon rekeying, by broadcasting the group keys to only the selected MSs that need them, rather than sending to all MSs.

Acknowledgements

This work is supported by Universiti Putra Malaysia and Ministry of Science, Technology and Innovation under the Science-fund (no. 01-01-04-SF1417).

Author details

Mohammad-Mehdi Gilanian-Sadeghi[1], Borhanuddin Mohd Ali[1] and
Jamalul-Lail Ab Manan[2]

1 Department of Computer and Communication Systems Engineering, Faculty of Engineering, Universiti Putra Malaysia, UPM Serdang, Selangor, Malaysia

2 Strategic Advanced Research, MIMOS Berhad, Malaysia

References

[1] "IEEE Std 802.16, IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Broadband Wireless Access Systems ", ed: IEEE Press, 2004.

[2] "IEEE Std 802.16, IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Broadband Wireless Access Systems and Revision of IEEE Std 802.16-2004," ed: IEEE Press, 2009.

[3] "IEEE Std 802.16e, IEEE Standard for Local and metropolitan area networks, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems, Amendment

2: Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands and Amendment and Corrigendum to IEEE Std 802.16-2004," ed: IEEE Press, 2006.

[4] "IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems," ed: IEEE Press, 2004.

[5] B. Aboba, L. J. Blunk, J. R. Vollbrecht, J. Carlson, and H. Levkowetz, "Extensible Authentication Protocol," RFC 3748, 2004.

[6] R. Laboratories, PKCS #1: RSA Cryptography Standard, 2002.

[7] S. Ahson and M. Ilyas, WiMAX: Standards and Security. CRC Press, Inc. Boca Raton, FL, USA, 2008.

[8] D. Johnston and J. Walker, "Overview of IEEE 802.16 Security," IEEE Security and Privacy, vol. 2, pp. 40-48, 2004.

[9] A. Deininger, S. Kiyomoto, J. Kurihara, and T. Tanaka, "Security Vulnerabilities and Solutions in Mobile WiMAX " IJCSNS International Journal of Computer Science and Network Security vol. 7, pp. 7-15, 2007.

[10] T. Shon, B. Koo, J. H. Park, and H. Chang, "Novel Approaches to Enhance Mobile WiMAX Security," EURASIP Journal on Wireless Communications and Networking, Article ID 926275, 2010.

[11] "P802.16m/D6, IEEE Standard for Local and Metropolitan Area Networks - Part 16: Air Interface for Broadband Wireless Access Systems - Advanced Air Interface," May 2010

[12] J. Hur, H. Shim, P. Kim, H. Yoon, and N.-O. Song, "Security Considerations for Handover Schemes in Mobile WiMAX Networks," in IEEE Wireless Communications and Networking Conference (WCNC), 2008, pp. 2531-2536.

[13] S. Rafaeli and D. Hutchison, "A survey of key management for secure group communication " ACM Computing Surveys, vol. 35, pp. 309-329, 2003.

[14] M. Baugher, R. Canetti, L. R. Dondeti, and F. Lindholm, "Multicast Security (MSEC) Group Key Management Architecture," in RFC 4046, 2005

[15] T. Hardjono and L. R. Dondeti, Multicast And Group Security. USA, 2003.

[16] Y. Challal and H. Seba, "Group Key Management Protocols: A Novel Taxonomy," International Journal of Information Technology, vol. 2, pp. 105-118, 2005.

[17] S. Gharout, A. Bouabdallah, M. Kellil, and Y. Challal, "Key Management With Host Mobility in Dynamic Groups," in International conference on Security of information and networks New York, USA, 2010, pp. 186-193.

[18] S. Xu, C.-T. Huang, and M. M. Matthews, "Secure Multicast in WiMAX," Journal of Networks, vol. 3, pp. 48-57, 2008.

[19] G. Kambourakis, E. Konstantinou, and S. Gritzalis, "Revisiting WiMAX MBS security," Computers and Mathematics with Applications, vol. 60, pp. 217-223, 2010.

[20] H. Li, G. Fan, J. Qiu, and X. Lin, "GKDA: A Group-Based Key Distribution Algorithm for WiMAX MBS Security," Advances in Multimedia Information Processing, LNCS, Springer Verlag, vol. 4261, pp. 310-318, 2006

[21] C. T. Huang and J. M. Chang, "Responding to Security Issues in WiMAX Networks," IEEE IT Professional vol. 10, pp. 15-21, 2008.

[22] C.-T. Huang, M. Matthews, M. Ginley, X. Zheng, C. Chen, and J. M. Chang, "Efficient and Secure Multicast in WirelessMAN:A Cross-layer Design," Journal of Communications Software and Systems, vol. 3, pp. 199-206, 2007.

[23] S. Chakraborty, S. Majumder, F. A. Barbhuiya, and S. Nandi, "A Scalable Rekeying Scheme for Secure Multicast in IEEE 802.16 Network," Communications in Computer and Information Science, Springer, vol. 132, pp. 471-481, 2011.

[24] J. Brown, X. Du, and M. Guizani, "Efficient rekeying algorithms for WiMAX networks," Security and Communication Networks, vol. 2, pp. 392–400, 2009.

[25] M. M. G. Sadeghi, B. M. Ali, M. Ma, J. A. Manan, N. K. Noordin, and S. Khatun, "Scalable Rekeying Algorithm in IEEE 802.16e," in 17th Asia-Pacific Conference on Communications (APCC), Sabah, Malaysia, 2011, pp. 726-730.

[26] J. A. Store, An Introduction to Data Structures and Algorithms. Waltham, USA: Birkhauser, Springer, 2001.