Towards Efficient Rekeying for IEEE 802.16e WiMAX Networks

Jeremy Brown and Xiaojiang Du
Department of Computer Science
North Dakota State University
Fargo, ND 58108, USA
Email: {jeremy.brown | xiaojiang.du}@ndsu.edu

***Abstract*—In this paper, we study the rekeying issue in IEEE 802.16e WiMAX networks. The existing rekeying scheme - the Multicast and Broadcast Rekeying Algorithm (MBRA) unicasts new keys to each subscriber station (SS). This scheme does not scale well since it incurs large communication overheads when the number of SSs increase. In our work, first we propose a general tree-based rekeying scheme, which is more efficient than the MBRA. We then formulate an optimization problem to determine the optimal tree structure for a given number of SSs. Furthermore, we design a novel and efficient rekeying scheme for WiMAX networks. Our new rekeying scheme utilizes efficient primitives and application features of WiMAX networks. Both analysis and performance evaluation show that the new rekeying scheme can significantly reduce the communication overheads.**

I. Introduction

The IEEE 802.16 WiMAX network is rapidly gaining popularity among wireless service providers because of its open standard, extended coverage and high throughput. WiMAX networks enable the last mile wireless broadband access, and can deliver up to 70 Mbps or 30 miles. WiMAX stands for Worldwide Interoperability for Microwave Access. Several IEEE standards for WiMAX have been published, such as IEEE 802.16d (stationary WiMAX), 16e (mobile WiMAX), and 16j (mobile multi-hop relay-based network). As WiMAX technology evolves and becomes more popular, security is an increasingly important issue.

Wireless networks face serious security challenges, due to the inherent vulnerability of radio transmissions, in addition to other factors. Without careful design, wireless communications protocols will fall victim to a number of attacks that could cause various damages to the network.

The designers of IEEE 802.16 sought to incorporate security into the protocol, but in spite of that, serious security flaws remained [1]. The standards prior to revision 802.16e [2] suffered from a number of serious security problems. Early revisions required that the subscriber station (SS) authenticate itself to a base station (BS), but lacked mutual authentication: the BS's responses use publicly available information [1]. There are also serious issues with key distribution and management in the Privacy and Key Management protocol (PKMv1) [2]: an attacker may replay the original message or respond to the three messages that make up the authentication protocol [3]–[5]. These messages could result in Denial of Service (DoS) attacks by exhausting the BS's resources [6].

The IEEE 802.16e standard fixed many of these issues, but there are a number of other vulnerabilities that this revision does not address, include:

- DoS attacks on the BS could happen during the PKMv2 authentication because of the heavy public key computational load [7].
- The BS↔SS authentication process in PKMv2 is vulnerable to an interleaving attack. This results in unauthorized access to the network [1].
- Bandwidth request messages can be tampered with and forged, causing DoS attacks and other security problems. [7].
- Management messages are still passed in the clear, and this could be used to attack the network. Such a message would disrupt traffic between the BS and SS [3], [4], [7].
- Man in the Middle attacks are possible during SS basic capability negotiation because the standard does not make any attempt to secure the negotiation [8].
- Stateless Ranging Request messages are not encrypted or authenticated. This could be used for a DoS attack if an attacker tampers with any of these messages [8].
- The network descriptor message is still vulnerable to tampering and forgery, and attacking it could cause various issues, such as DoS [9].

Another flaw with the existing IEEE 802.16e protocol is that its Multicast and Broadcast Rekeying Algorithm (MBRA) does not scale well. Under the MBRA, the BS transmits the Group Key Encryption Key (GKEK) to each SS via a unicast message. The Group Transmission Encryption Key (GTEK) is subsequently transmitted via a multicast transmission, encrypted by the GKEK. The MBRA has a high communication overhead. The message overhead increases linearly with the number of SSs associated with a BS. The IEEE 802.16e specifies that the following messages are sent to set up the group key:

$$\mathrm{BS} \rightarrow \text{each SS} : \left\{\mathrm{GKEK}\right\}_{\{\mathrm{KEK}\}} \quad (1)$$

$$\mathrm{BS} \Rightarrow \text{all SS} : \left\{\mathrm{GTEK}\right\}_{\{\mathrm{GKEK}\}} \quad (2)$$

The BS sends message 1 to a SS via unicast when the SS connects to the BS for the first time. Message 1 includes the GKEK, which is protected by the shared individual key (KEK) between the SS and the BS. Once all SSs have the GKEK, the

BS can send the transmission key - GTEK to all nodes via a group broadcast message 2.

In this paper, we propose two efficient schemes for rekeying in WiMAX networks. In Section II, we describe a general tree-based rekeying scheme, and in Section V we present a new rekeying scheme for WiMAX networks.

II. Tree-based Rekeying Schemes

Huang et al. [4] proposed a method for improving the rekeying process by dividing the SSs into N subgroups, where $N = 2^k$, and k is the smallest power of 2 which can accommodate the desired number of SSs per BS. Each subgroup has a Sub-Group Key Encryption Key (SGKEK). Huang stated that k would be determined by the specific application to give the best performance. This method requires the BS to maintain $2^k - 1$ SGKEK keys for each subgroup. This scheme increases the number of keys transmitted when a member SS leaves a group, and it also requires more keys to be transmitted when a new SS joins a BS. A typical binary tree illustrates the scheme in [4].

In Fig. 1, the SSs are divided into a binary tree with four subgroups. Suppose a member in subgroup 1 left the BS, then the BS needs to update keys for all remaining members. First, the BS would unicast Message 3 to all remaining SSs in subgroup 1, which updates the following keys: SGKEK_{1234}, SGKEK_{12}, SGKEK_{1}, and GTEK. Note: Message 3 is encrypted by the individual key KEK shared between each SS and the BS.

$$\text{BS} \rightarrow \text{SS} : \{\text{SGKEK}_{1234}, \text{SGKEK}_{12}, \text{SGKEK}_{1}, \text{GTEK}\}_{\text{KEK}} \quad (3)$$

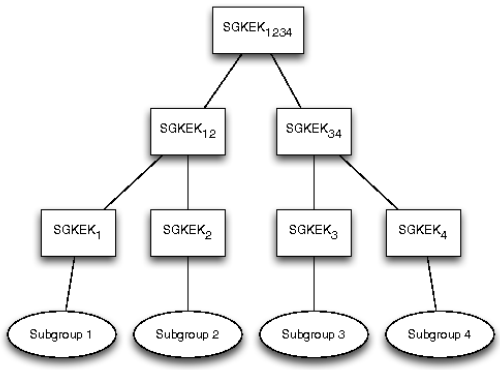

Fig. 1. A group segmented using a binary tree.

For the SSs in subgroups 3 and 4, the BS can update the two keys (SGKEK_{1234} and GTEK) using one broadcast, as shown in Message 4.

$$BS \Rightarrow \text{SS}_{\text{SG3}}, \text{SS}_{\text{SG4}} : \{\text{SGKEK}_{1234}, \text{GTEK}\}_{\text{SGKEK34}} \quad (4)$$

The BS updates keys for all SSs in subgroup 2 via one broadcast Message 5.

$$\text{BS} \Rightarrow \text{SS}_{\text{SG2}} : \{\text{SGKEK}_{1234}, \text{SGKEK}_{12}\}_{\text{SGKEK2}} \quad (5)$$

The above scheme is better than the MBRA because a subgroup is smaller, and thus requires fewer transmissions to accomplish the rekeying task. The number of transmissions will be discussed in III.

However, Huang et al. [4] only considered binary trees. The main problem with the binary tree structure is that the tree depth could become large as the number of SSs increases. In this paper, we propose to improve the rekeying scheme by using an n-ary ($n > 2$) tree. For given number of SSs, we formulate and solve an optimization problem that finds the optimal n which minimizes the total energy consumption of the rekeying process. In Table I, we list the notations used in this paper.

n	tree width
d	tree depth
k	total number of keys for the tree
N	total number of subscriber stations
s	number of subscriber stations per subgroup
g	number of subgroups
B	number of broadcast transmissions
U	number of unicast transmissions
Tx	total number of transmissions
Rx	total number of receptions
α	the ratio between transmission and reception energy consumption

TABLE I
Notations

An n-ary ($n > 2$) tree of the same depth would be able to accommodate more SSs than a binary tree, and therefore reduces the number of transmissions for rekeying. A fully-populated 3-ary tree is depicted in Fig. 2. The number of different keys for a binary tree is $2^k - 1$, where $k = \log_2 \lceil \frac{N}{s} \rceil$. In general, the number of keys required by an n-ary tree is given by equation 6.

$$k = \sum_{i=0,1,..}^{d} n^i \quad (6)$$

The tree depth d is given in equation 7, where g is the number of subgroups, given by $\lceil \frac{N}{s} \rceil$:

$$d = \lceil \log_n g \rceil = \left\lceil \log_n \left\lceil \frac{N}{s} \right\rceil \right\rceil \quad (7)$$

III. Overheads of the Rekeying Schemes

In this Section, we compare the overheads of the MBRA and the tree-based rekeying schemes.

A. Analysis of Storage Overhead

One overhead of using tree structures for key management is that the BS needs to store more group keys. However, this is a minor issue, since the BS is assumed to have sufficient storage space. Below, we use examples to show the storage overhead for group keys. In Fig. 3, we plot the maximum number of group keys needed for some tree structures, including binary,

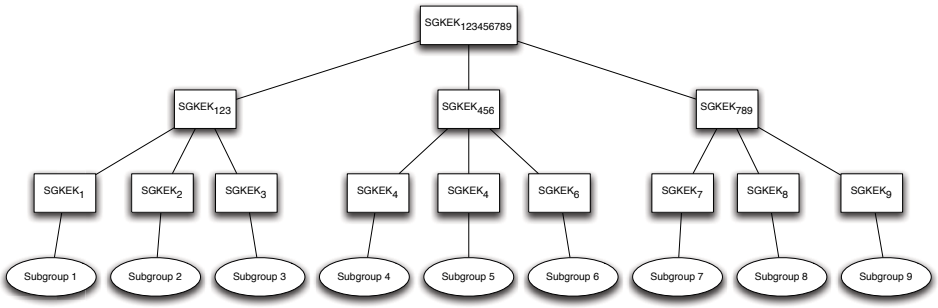

Fig. 2. A group segmented using a 3-ary tree

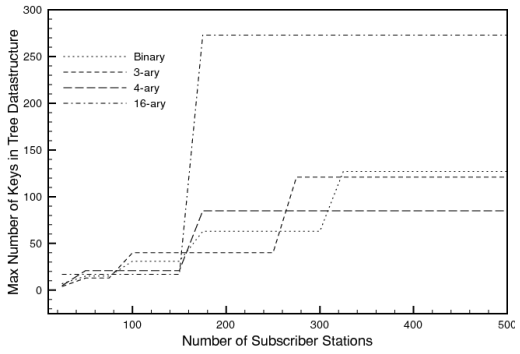

Fig. 3. The maximum number of keys using tree structures

3-ary, 4-ary, and 16-ary. Note the "Max Number" in the y-axis means when the tree is fully populated. In Fig. 3, the x-axis is the number of SSs, varying from 30 to 500.

Another comparison of key storage is shown in Table II, where the number of SSs is fixed at 500, and the tree depth, total number of group keys, and the maximum supported network size are computed for several tree structures. The number of SSs per subgroup - s is set to 10. Table II shows the non-linearity between the tree type n and the total number of group keys. The total number of group keys is calculated using equation 6. The *Max Network Size* column reflects the maximum number SSs supported by a tree at the depth indicated by the *Tree Depth* column. The exact number is M, given by Equation 8:

$$M = n^d \cdot s. \tag{8}$$

Tree Type	Tree Depth	Total Group Keys	Max Network Size
Binary	7	127	1280
3-ary	5	121	2430
4-ary	4	85	2560
5-ary	4	156	6250
6-ary	4	259	12960
7-ary	4	400	24010
8-ary	3	73	5120
9-ary	3	91	7290
16-ary	3	273	40960

TABLE II
COMPARISON OF TREE STRUCTURES

B. Preliminary Analysis of Communication Overhead

In a WiMAX network, there are four types of network events that require the transmission/update of group keys: (1) An SS joins the BS; (2) The GTEK expires; (3) The GKEK expires; (4) An SS leaves the BS.

When a SS joins a group, the BS unicasts the current group keys to it. There are no further improvements needed. Each GTEK and GKEK should be replaced before they expire such that strong security can be achieved. Because the GKEK and SGKEK are used infrequently, it is unlikely that they will expire at the same time as a GTEK. Shortly before a group key will expire, it can be securely replaced by using a broadcast, protected by the current GKEK, i.e.:

$$\text{BS} \Rightarrow \text{all SS} : \left\{\text{new group key}\right\}_{\text{GKEK}}$$

For case (4), when an SS leaves the BS, it is possible that the SS could gain access to the new key if a simple broadcast is used. To ensure the forward and backward security, the IEEE 802.16e specifies that the BS would have to unicast rekeying messages to each SS, using the MBRA.

The main problem with the MBRA is that the number of unicast transmissions increases linearly with the number of SSs associated with a BS. The tree-based rekeying scheme described in Section II could significantly reduce the communication overhead. Fig. 4 shows that the number of group keys transmitted under MBRA is much larger than those under tree-based rekeying schemes. The number of group keys transmitted under MBRA is a straight line with a slope of one. Fig. 5 provides a closer look at the communication overhead (the number of group keys transmitted) under different tree-based schemes. The results are based on equation 6 for the n-ary ($n > 2$) trees and $N = 2^k - 1$ for the binary tree. Fig. 5 shows that the communication overhead of the tree-based schemes does not increase much, when the number of SSs becomes large.

Fig. 5 shows that the n-ary tree-based schemes could significantly reduce the communication overhead of rekeying. However, it is not clear which n achieves the optimal results (e.g., consume the minimum total energy), for given number of SSs. The results in Table II also show that the rekeying overhead is a nonlinear function of n. In the next subsection, we formally analyze the overhead of rekeying, and we show how to obtain the optimal n for a given number of SSs.

C. Formal Analysis of Communication Overhead

In this subsection, we first calculate the number of transmissions and receptions under an n-ary tree-based rekeying

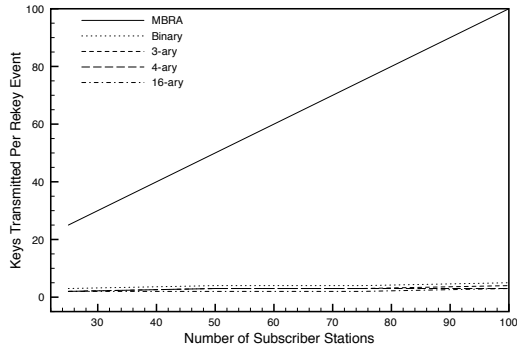

Fig. 4. Transmitted keys under MBRA and tree-based schemes

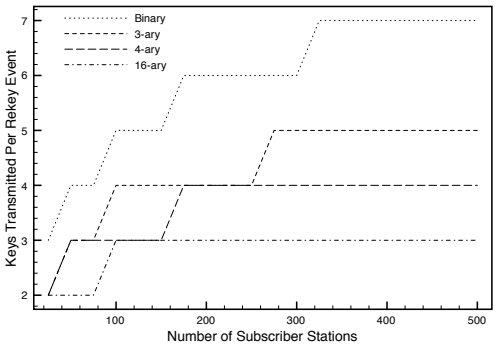

Fig. 5. Number of transmitted keys under different n-ary tree schemes.

scheme. Recall there are four events that cause rekeying. The overheads caused by the first three events under different rekeying schemes (including MBRA and n-ary tree-based schemes) are similar. The main difference of overhead is for event (4) (i.e., an SS leaves the BS).

Suppose an SS of subgroup 1 leaves the BS, the BS needs to broadcast new group keys to SSs in all other subgroups (i.e., subgroup 2 - g). If we look at the tree structure (e.g., Fig. 2) from top down, denote the root as level 0, one broadcast is required for each of the $n-1$ branch at level 1, i.e., one broadcast for subgroups 4-6, and one broadcast for subgroups 7-9. Similarly, at level two, one broadcast is required for each of the $n-1$ branch, i.e., one broadcast for subgroup 2 and one for subgroup 3. To sum up, $n-1$ broadcasts are required at each tree level, from level 1 to d. Hence, there are a total of $d*(n-1)$ broadcasts. In addition, the BS needs to unicast a rekeying message to each SS in subgroup 1. The maximum number of SSs in subgroup 1 is $s-1$ (after the SS leaves and before any new SS joins). Hence we have the total number of broadcasts B and unicasts U given in Equations 9 and 10, respectively.

$$B = d(n-1) \tag{9}$$

$$U = s-1 \tag{10}$$

Fig. 6 plots the total number of broadcasts under the schemes, for different number of SSs, varying from 20 to 500. Again, the number of transmissions under MBRA is a straight line with slope 1, and is much larger than the n-ary tree-based rekeying schemes.

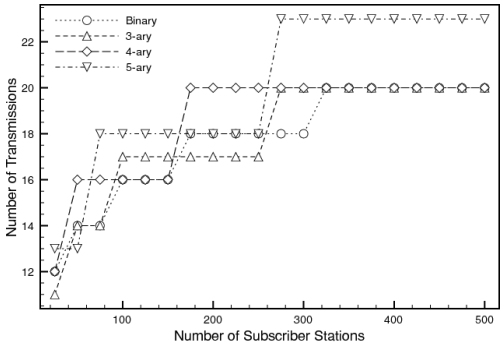

Fig. 6. Comparison of the total number of transmissions

IV. The Energy Consumption and Optimal Tree

A more important consideration of the overhead is the total energy expenditure of rekeying. Denote the energy of a transmission as e, and the energy of a reception as $\alpha * e$, where α is a factor between 0 and 1. Then the total energy consumption E of Tx transmissions and Rx receptions is given by:

$$E = e \cdot Tx + \alpha \cdot e \cdot Rx \tag{11}$$

Note that the lengths of the messages transmitted (and received) at various tree level are different. Using Fig. 2 as an example, suppose an SS leaves subgroup 1. The message broadcasted to subgroups 456 only includes two new group keys. On the other hand, the message broadcasted to subgroup 2 includes three new group keys. The energy consumption of transmission (and reception) can be approximately considered as a linear function of the message length. For simplicity, in the following we assume that the message length is proportional to the number of keys included.

Table III summarizes the number of transmissions and receptions, and the number of keys per message for different broadcasts and unicasts, for the 3-ary tree in Fig. 2.

	B to SG456, SG789	B to SG 2, SG 3	U to SG 1
Tx	$n-1$	$n-1$	$s-1$
Rx	$\left\lceil \frac{N}{n} \right\rceil (n-1)$	$\left\lceil \frac{N}{n^2} \right\rceil$(n-1)	$s-1$
Keys/Msg	1	2	$d+1$

TABLE III
THE NUMBER OF TRANSMISSIONS, RECEPTIONS AND KEYS

Next, we generalize the results in Table III to an n-ary tree. For an n-ary tree, suppose an SS leaves subgroup 1. During rekeying, the BS unicasts a message to each of the $s-1$ SSs in subgroup 1. This unicast message has $d+1$ new group keys, including one key per tree level, plus the global key, SG_{1-9}. Hence, considering the message length, the

total number of transmissions (and receptions) from unicasts is given in Equation 12.

$$U = (d+1)(s-1) \tag{12}$$

Now let's consider the number of broadcasts. At each tree level i from 1 to d, there are $n-1$ broadcasts, and each broadcast message includes i keys. Hence, considering the message length, the total number of broadcasts is

$$B = (n-1)\sum_{i=1}^{d} i \tag{13}$$

The total number of transmissions is $Tx = U + B$.

For each broadcast at tree level i, the total number of nodes that receive the broadcast is: the total number of SSs - N divided by n^i. The total number of receptions from broadcasts is the number of nodes multiplied by the number of broadcast messages to per node. Hence, considering the message length, the total number of receptions is

$$Rx = U + (n-1)\sum_{i=1}^{d} i \left\lceil \frac{N}{n^i} \right\rceil \tag{14}$$

If we plug in Equations 12,13, and 14 into Equation 11, we have the total energy consumption for rekeying. Our objective is to find the optimal n for given N, s and α. We can simplify Equation 11 by removing e, since e does not depend on n. Hence, we have:

$$E(n) = (n-1)\left(\sum_{i=1}^{\lceil \log_n \lceil \frac{N}{s} \rceil \rceil} \alpha i \left\lceil \frac{N}{n^i} \right\rceil + i\right) + (1+\alpha)(s-1)\left(\left\lceil \log_n \left\lceil \frac{N}{s} \right\rceil \right\rceil + 1\right) \tag{15}$$

As we can see, the total energy consumption $E(n)$ is a complicated, nonlinear function of n. We discuss how to obtain the optimal n in next subsection.

Next, we discuss how to determine the optimal tree structure. We want to find out the optimal n that minimizes the total energy consumption of rekeying (Equation 11), for given the following parameters: the total number of SSs - N; the number of SSs in each subgroup - s, and the energy ratio α. In the following discussion, without losing generality, assume that $s = 10$ and $\alpha = 0.5$.

Fig. 7 plots the total energy consumptions E for various values of N between 50 - 500, with an increase of 50; and for n between 1 to 10. Note that $n = 1$ is the MBRA. Table IV lists the optimal value of n, for the same values of N and n.

For given N, s, and α, the optimal n is the value that minimizes the total energy consumption in Equation 11. We propose the following approach to find the optimal n: Note that the tree width n should be no more than the total number of SSs - N, i.e., $n \leq N$. Hence, we can compute the total energy consumption $E(n)$ for every n between 2 and N, and the n with the smallest $E(n)$ is the optimal tree structure, i.e., the optimal value n (denote as n_{opt}) is:

$$n_{opt} = \arg\min E(n) \tag{16}$$

N	50	100	150	200	250
Optimal n	5	10	4, 5	5	5
N	300	350	400	450	500
Optimal n	6	6	10	9	10

TABLE IV
THE OPTIMAL VALUE OF n

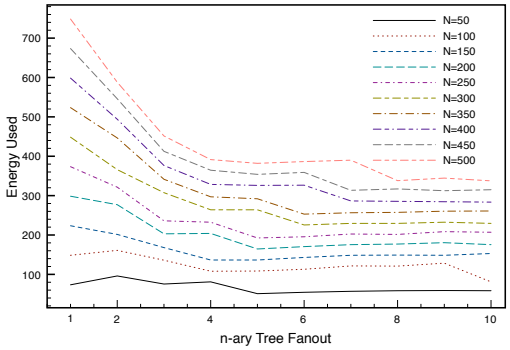

Fig. 7. Comparison of the total energy consumption

V. A New Rekeying Scheme

In this Section, we present a novel rekeying scheme that we designed for WiMAX networks. The IEEE 802.16e MBRA rekeying scheme is based on the assumption that an SS can authenticate itself to a BS. We propose a new rekeying scheme for 802.16e WiMAX networks, which is much more efficient that the MBRA. The new rekeying scheme is presented below:

1) When a new SS joins a BS, the BS transmits the current GKEK and GTEK to it, protected by the shared individual key (KEK) between the SS and the BS:
   $BS \rightarrow SS : \{GKEK, GTEK\}_{KEK}$
2) When a group key expires, instead of letting the BS transmit a new key, each SS generates a new group key by applying a one-way hash function f on the current group key, e.g.,:
   $\text{GTEK}_{new} = f(\text{GTEK}_{old})$
   $\text{GKEK}_{new} = f(\text{GKEK}_{old})$
3) When an SS leaves the group, the BS triggers a rekey process at every existing SS by broadcasting a random number r, protected by the old GKEK:
   $BS \Rightarrow$ all SS : $\{r\}_{GKEK}$
   Then each existing SS computes the new group key GTEK using the random number and the old key:
   $\text{GTEK}_{new} = f(\text{GTEK}_{old})$

Note that in step 2 there is no transmission required, which greatly reduces the communication overhead of rekeying, especially when the number of SSs is large. In step 3, if the leaving SS (denoted as L) is still within the transmission range of the BS, then L could decrypt the broadcast message and obtain the random number, and hence compute the new group

key. However, this is not a big security concern, since node L has been authenticated by the BS, and is considered as a legitimate node. Although it is possible that an authenticated node could launch attacks, this would be insider attacks and is a totally different story, which is out of the score of the rekeying scheme and this paper.

VI. Performance Comparison

In this Section, we compare the communication overheads of various rekeying schemes, including the IEEE 802.16e's MBRA, the binary-tree based scheme [4], the n-ary ($n \geq 3$) tree based scheme that we proposed, and the new rekeying scheme that we designed. MBRA is a very simple algorithm, basically the BS unicasts new keys to each SS individually. The tree-based rekeying schemes are more efficient than MBRA because they divide SSs into subgroups, and many transmissions are done via broadcasts rather than unicasts. These schemes take advantage of the wireless broadcast nature and require fewer transmissions than MBRA. Our new rekeying scheme utilizes a one-way hash function and eliminates many transmissions (in step 2). Furthermore, the new rekeying scheme utilizes the fact that a leaving SS has already been authenticated and can be trusted during the period when it is leaving the BS. With only one broadcast (in step 3), all group keys in every existing SS can be updated. This is much more efficient than the MBRA and the tree-based rekeying schemes.

Table V is a comparison of the number of transmissions for the four kinds of rekeying schemes, listed according to the four types of events, one event per row. Broadcasts and unicasts are listed separately in the table, because they cause different numbers of receptions. Each of the first three events (SS joins a BS, a GTEK expires, and a GKEK expires) requires the same number of transmissions for the MBRA, binary-tree based scheme, and n-ary tree based schemes. Note that the new scheme does not need any transmission for events 2 or 3. Table VI lists the number of transmissions (TX) and receptions (RX) for each event, under theses schemes.

Event	MBRA	Binary Tree Groups	n-ary Tree Groups	New Scheme
1	1 U per SS			
2	1 B			0 B/U
3	1 B			0 B/U
4	N U	$2^k - 1$ B $s - 1$ U	$d(n-1)$ B $s - 1$ U	1 B

TABLE V
Number of messages per event type. U - Unicast. B - Broadcast.

Event	MBRA	Binary Tree Groups	n-ary Tree Groups	New Scheme
1	1 TX; N RX			
2	1 TX; N RX			0 TX, 0 RX
3	1 TX; N RX			0 TX, 0 RX
4	N TX	$2^k - 1+$ $g - 1$ TX	$d(n-1)+$ $s - 1$ TX	1 TX
	N RX	$N(2^k - 1)+$ $g - 1$ RX	$d(n-1) \cdot N+$ $s - 1$ RX	N RX

TABLE VI
Number of transmissions and receptions per event type.

As we can see from the above comparisons, the new rekeying scheme is much more efficient than the MBRA and the tree-based schemes.

VII. Conclusions

In this paper, we studied the rekeying issue in WiMAX networks. The existing IEEE 802.16e rekeying scheme - the Multicast and Broadcast Rekeying Algorithm (MBRA) unicasts updated keys to each SS. However, the MBRA does not scale well and incurs large communication overheads as the number of SSs increase. First, we extended the binary-tree based scheme proposed by Huang et al., and we proposed general n-ary tree based rekeying schemes. Then we formulated an optimization problem for determining the optimal tree structure n based on the total energy consumption during rekeying. Furthermore, we presented a novel and efficient rekeying scheme for WiMAX networks. Our new rekeying scheme utilizes a one-way hash function and the existing trust for a leaving SS, and hence significantly reduces the communication overhead. The performance evaluation confirms the good performance of our rekeying scheme.

VIII. Acknowledgement

This research was supported in part by the US National Science Foundation (NSF) under grants CNS-0721907 and CNS-0709268, and the Army Research Office under grants W911NF-07-1-0250 and W911NF-08-1-0334.

References

[1] D. Johnston and J. Walker, "Overview of ieee 802.16 security," IEEE Security and Privacy, vol. 2, no. 3, pp. 40 48, 2004.

[2] IEEE Std 802.16e-2005. 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, IEEE, 2005.

[3] T. Shon and W. Choi, "An analysis of mobile WiMAX security: vulnerabilities and solutions," Lecture Notes in Computer Science, Springer-Verlag, no. 4658, pp. 8897, 2007.

[4] C. Huang and J. Chang, "Responding to security issues in wimax networks," IT Professional, vol. 10, no. 5, pp. 1521, 2008.

[5] S. Xu, M. Matthews, and C. Huang, "Security issues in privacy and key management protocols of ieee 802.16," in Proc. of the 44th Annual Southeast Regional Conference, pp. 113 118, 2006.

[6] E. Eren, "Wimax security architecture analysis and assessment," in Proc. of IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, pp. 673 677, 2007.

[7] L. Maccari, M. Paoli, and R. Fantacci, "Security analysis of ieee 802.16," in Proc. of the IEEE International Conference on Communications (ICC), pp. 1160 1165, 2007.

[8] T. Han, N. Zhang, K. Liu, B. Tang, and Y. Liu, "Analysis of mobile wimax security: Vulnerabilities and solutions," in Proc. of the 5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems (MASS), pp. 828 833, 2008.

[9] Y. Zhou and Y. Fang, "Security of ieee 802.16 in mesh mode," in Proc. of the IEEE Military Communications Conference (MILCOM), pp. 16, 2006.