SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks. 2009; **2**:392–400
Published online 30 July 2009 in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/sec.124

Efficient rekeying algorithms for WiMAX networks

Jeremy Brown[1], Xiaojiang Du[*,†2] and Mohsen Guizani[3]

[1]*Department of Computer Science, North Dakota State University, Fargo, ND 58105, U.S.A.*
[2]*Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, U.S.A.*
[3]*Department of Computer Science, Western Michigan University, Kalamazoo, MI 49008, U.S.A.*

Summary

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 also formulate an optimization problem to determine the optimal tree structure for given number of SSs. Furthermore, we present a novel and efficient rekeying scheme for WiMAX networks. Our new rekeying scheme utilizes efficient security schemes and the WiMAX network application feature. Both analysis and performance evaluation show that our rekeying scheme can significantly reduce the communication overheads. Copyright © 2009 John Wiley & Sons, Ltd.

KEY WORDS: WiMAX; IEEE 802.16e; security; key management

1. Introduction

Recently, 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 World-wide 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 involves and becomes increasingly popular, security becomes an important issue.

Wireless networks face serious security problems, simply because of the lack of privacy inherent to radio transmissions. Without careful design, communications protocols will fall victim to a number of attacks that will seriously compromise the network.

The designers of 802.16 sought to incorporate security into the protocol from the outset, but in spite of that, serious security flaws remained [1]. The standard used a different threat model as it was revised from a line of site protocol in its original revisions to one that could support mobility in its 2005 revision: 802.16e [2].

The versions of 802.16 prior to revision 802.16e suffered from a number of serious security problems. Early revisions required that the subscriber stations (SSs) authenticate itself to a base station (BS), but lacked

[*] Correspondence to: Xiaojiang Du, Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, U.S.A.
[†] E-mail: dxj@ieee.org

mutual authentication because the BS's responses used publicly available information, allowing attackers to impersonate a BS [1]. There are also serious issues with key distribution and management in PKMv1 [1] because an attacker may replay either the original message or responses to the three messages that make up the authentication protocol [3–5]. These messages could result in denial of service (DoS) by exhausting the BS's resources [6].

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

- DoS attacks on the BS could happen during the PKMv2 authentication because of the heavy public key computational load. An attacker could simply flood the BS with messages, and the BS could use up its computational resources evaluating signatures and decrypting messages [7].
- The BS ↔ SS authentication process in PKMv2 is vulnerable to an interleaving attack. In this attack, the attacker impersonates a valid SS, exchanges the first two messages of the PKMv2 sequence with a valid BS, and then it replays these to the original, valid SS to gain the final PKMv2 message. The attacker then uses the final message from the original SS to complete the original PKMv2 sequence with the BS. This results in unauthorized access to the network [4].
- Bandwidth request messages can be tampered with and forged, causing DoS attacks and other security problems. This is possible because 802.16 features centralized resource allocation. An attack on the resource allocation messages could potentially starve legitimate traffic of resources [7].
- Management messages are still passed in the clear, and this could be used to attack the network. One such attack is possible when an attacker sends bandwidth request and *sleep* control messages to a BS using a valid SS's identifiers. 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. This phase passes vital capability information to the BS [8].
- Stateless Ranging Request messages are not encrypted or authenticated. It includes such information as time synchronization, power adjustment, ranging status, etc. 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. This message enables a SS to build a list of BSs and neighboring SSs, and attacking it could cause various issues, such as DoS. [9]

Many of the above attacks could be defended by simply securing and/or authenticating management messages.

Another flaw with the existing 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:

$$\text{BS} \rightarrow \text{each SS} : \{\text{GKEK}\}_{\{\text{KEK}\}} \tag{1}$$

$$\text{BS} \Rightarrow \text{all SS} : \{\text{GTEK}\}_{\{\text{GKEK}\}} \tag{2}$$

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 algorithms for rekeying in WiMAX networks. In Section 2, we discuss segmenting the group into subgroups, and then in Section 4 we discuss an improved version of the MBRA.

2. Tree-based Rekeying Schemes

Huang *et al.* [4] propose a method of improving the rekeying process by dividing the SSs into N subgroups, where $N = 2^k$, and k is the smallest power of two which will accommodate the desired number of SSs per BS. Each subgroup has a Sub-Group Key Encryption Key (SGKEK). Huang states 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

DOI: 10.1002/sec

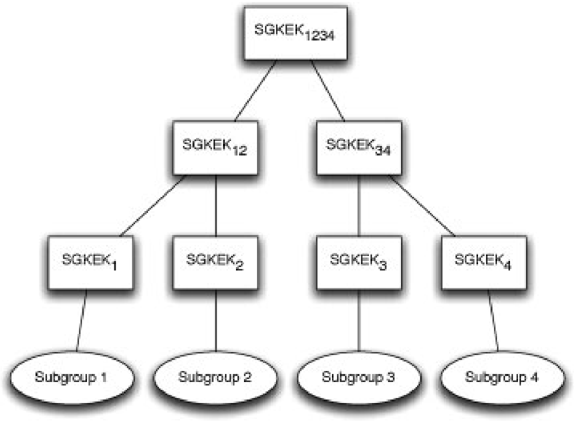

Fig. 1. A group segmented using a binary tree.

transmitted when a new SS joins a BS. We use Figure 1 to illustrate the scheme in Reference [4].

In Figure 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.

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

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.

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

Table I. Notations.

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

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

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

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 Section 3.

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 a 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 that will be used in this paper.

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 Figure 2.

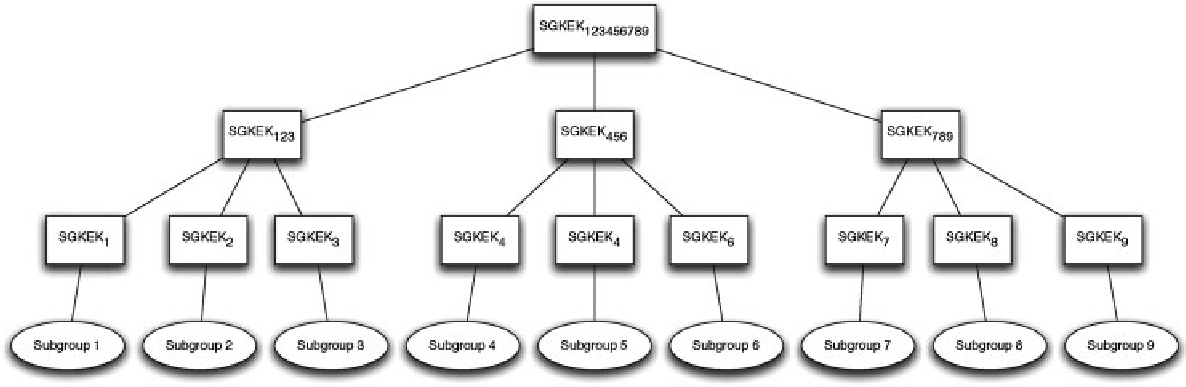

Fig. 2. A group segmented using a 3-ary tree.

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).

$$k = \sum_{i=0,1,\ldots}^{d} n^i \tag{6}$$

where g is the number of groups, and

$$g = \left\lceil \frac{N}{s} \right\rceil \tag{7}$$

The tree depth d is given in Equation (8):

$$d = \lceil \log_n g \rceil = \left\lceil \log_n \left\lceil \frac{N}{s} \right\rceil \right\rceil \tag{8}$$

3. Overheads of the Rekeying Schemes

3.1. Analysis of Storage Overhead

The 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 some examples to show the storage overhead for group keys. In Figure 3, we plot the maximum number of group keys needed for some tree structures, including binary, 3-ary, 4-ary, and 16-ary. Note the maximum means when the tree is fully populated. In Figure 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 nonlinearity between the tree type n and the total number of group keys. The total number of group keys is calculated using Equation (6).

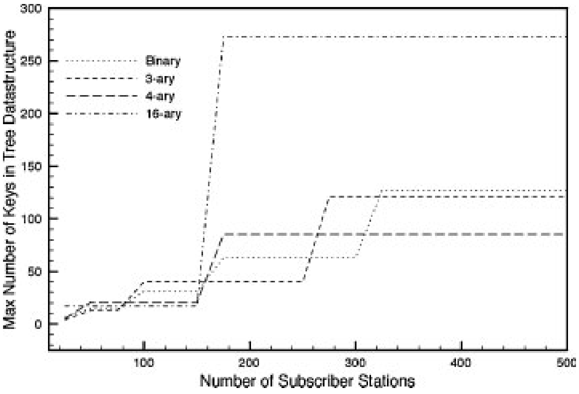

Fig. 3. The maximum number of keys using tree structures.

Table II. Comparison of tree structures.

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

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, as determined by Equation (9).

$$M = n^d \cdot s \tag{9}$$

3.2. 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. If a group key is expiring, it can be securely replaced using a broadcast, protected by the current GKEK:

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

For Case 4, when an SS leaves the BS, there is the chance that it could gain access to the new key if a

DOI: 10.1002/sec

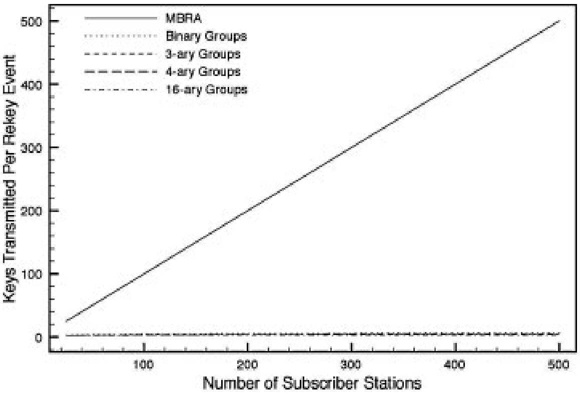

Fig. 4. The number of group keys under different schemes.

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 2 could significantly reduce the communication overhead. Figures 4 and 5 plot the number of group keys (which indicates the communication overhead) under different rekeying schemes. The results are based on Equation (6) for the n-ary ($n > 2$) trees and $N = 2^k - 1$ for the binary trees. As we can see from Figure 4, the number of group keys (and hence the number of transmissions) under MBRA increases with a slope of 1, while the number of group keys under the n-ary tree-based rekeying schemes does not increase much. Figure 5 gives a closer look at the variations of the number of group keys under the n-ary tree-based schemes, which shows a slight increase with the number of SSs.

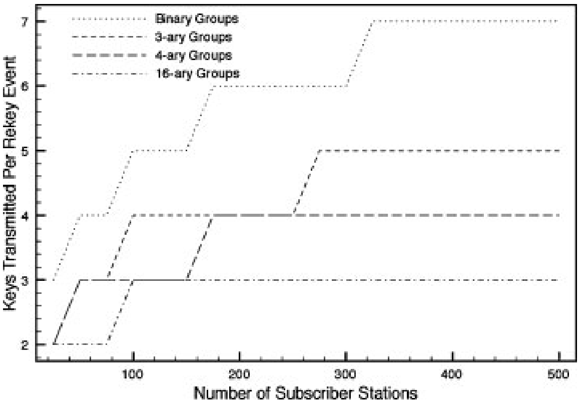

Fig. 5. A closer look at the overhead.

Figures 4 and 5 show 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 (i.e., N). The results in Table II also show that the rekeying overhead is a nonlinear function of n. In the next section, we formally analyze the overhead of rekeying, and we show how to obtain the optimal n for given number of SSs.

3.3. Formal Analysis of Communication Overhead

In this section, we first calculate the number of transmissions and receptions under a n-ary tree-based rekeying 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 2g). If we look at the tree structure (e.g., Figure 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 2, 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 (10) and (11), respectively.

$$B = d(n - 1) \tag{10}$$

$$U = s - 1 \tag{11}$$

Figures 6 and 7 plots the total number of transmissions (the sum of B and U) for different number of SSs, varying from 20 to 500. Figure 6 shows that the MBRA requires much more transmissions that n-ary tree-based rekeying schemes. Figure 7 is a closer look at the total number of transmissions for several n-ary tree-based schemes. Also, Figure 7 only shows the broadcasts.

DOI: 10.1002/sec

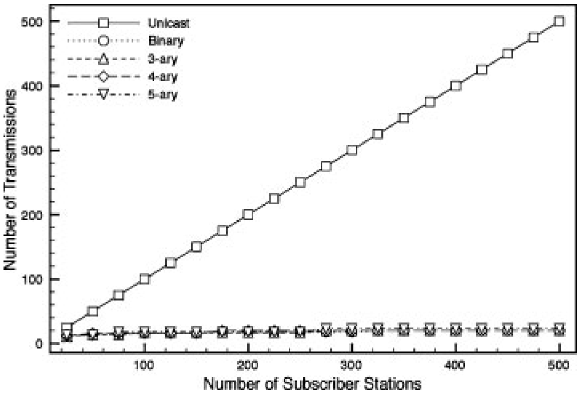

Fig. 6. Comparison of the total number of transmissions.

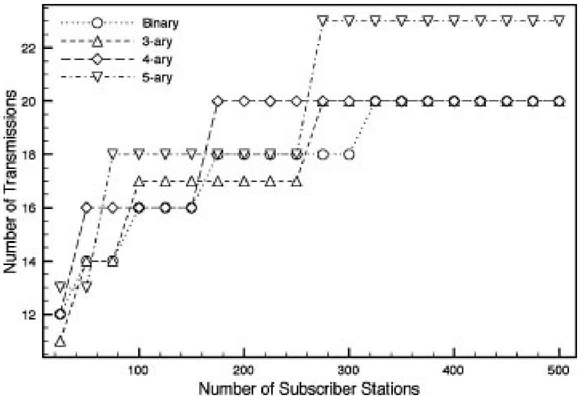

Fig. 7. Comparison of the total number of broadcasts.

3.4. Analysis of Energy Consumption

A more important consideration 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

$$E = e \cdot Tx + \alpha \cdot e \cdot Rx \tag{12}$$

Note that the lengths of the messages transmitted (and received) at various tree level are different. Using Figure 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. The number of transmissions, receptions and keys.

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

Table III summarizes the number of transmission and reception, and the number of keys per message for different broadcasts and unicasts, for the 3-ary tree in Figure 2.

Next, we will generalize the results from Table III to a n-ary tree. For a 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 (13).

$$U = (d+1)(s-1) \tag{13}$$

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

$$B = (n-1)\sum_{i=1}^{d} i \tag{14}$$

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

$$Rx = U + (n-1)\sum_{i=1}^{d} i \left\lceil \frac{N}{n^i} \right\rceil \tag{15}$$

Now we have the total energy consumption, including both transmissions and receptions, given in Equation (12):

$$E = Tx \cdot e + Rx \cdot \alpha \cdot e \tag{16}$$

DOI: 10.1002/sec

Our objective is to find the optimal n for given N, s and α. We can simplify Equation (12) by removing e, since e does not depend on n. Hence, we have

$$E(n) = (n-1)\left(\sum_{i=1}^{\left\lceil \log_n \left\lceil \frac{N}{s} \right\rceil \right\rceil} \alpha i \left\lceil \frac{N}{n^i} \right\rceil + i\right) + (1+\alpha)(s-1)\left(\left\lceil \log_n \left\lceil \frac{N}{s} \right\rceil \right\rceil + 1\right) \tag{17}$$

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 section.

3.5. Obtaining the Optimal *n*

We want to find out the optimal n that minimizes the total energy consumption of rekeying (i.e., Equation (12)), 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 $\alpha = 0.5$.

We compute the total energy consumption E for different values of N and n. The results are listed in Table IV, where n varies from 1 to 10, and N ranges from 100 to 500. Note that $n = 1$ is MBRA. The bold-face number in each column is the minimum total energy consumption for that N. For example, when $N = 200$, the minimum total energy consumption is 164.5, and it is the 5-ary tree that achieves the result.

Figure 8 plots the energy consumptions for various sizes of N between 50 and 500, with an increase of 50; and for n between 1 to 10. Note that $n = 1$ is the MBRA. Table V lists the optimal value of n, for the same values of N and n.

Table IV. The total energy consumption. Bold values indicate minimal consumption.

	100	200	300	400	500
1	148.5	298.5	448.5	598.5	748.5
2	161	277	366	494	588
3	136	203	307.5	376.5	451.5
4	108	204	264	328.5	391.5
5	108.5	**164.5**	264	326	382
6	113	170.5	**225.5**	326.5	386.5
7	121.5	175.5	229.5	286.5	390
8	121	177	229.5	285.5	338
9	128.5	180.5	232.5	284.5	344.5
10	**81**	175.5	229.5	**282.75**	**337.5**

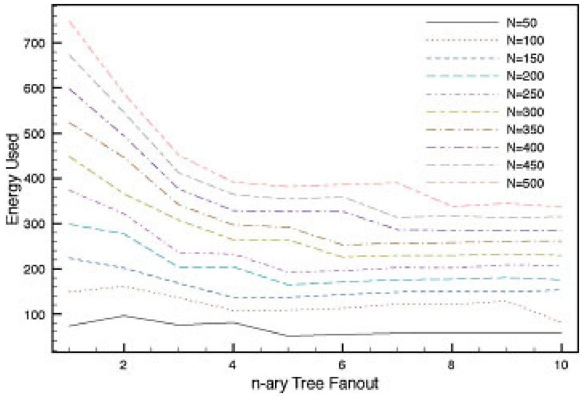

Fig. 8. Comparison of the total energy consumption.

Table V. The optimal value of *n*.

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

For given N, s, and α, we can obtain the optimal n that minimizes the total energy consumption in Equation (12) *via* the following approach: 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

$$n_{\text{opt}} = \arg\min E(n) \tag{18}$$

4. A New Rekeying Scheme

In this section, we present a novel rekeying scheme that we designed for WiMAX wireless networks. We describe the scheme in section 4.1, and present the performance comparison of our scheme and other schemes in section 4.2.

4.1. The Rekeying Scheme

The 802.16e MBRA rekeying scheme is based on the assumption that an SS can authenticate itself to a BS. Since all SSs are authenticated to the BS as part of the PKMv2 handshake protocol, forward and backward security of the group keys is not important. We propose a new rekeying scheme for 802.16e WiMAX networks, which is much more efficient that the MBRA.

DOI: 10.1002/sec

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:

$$\text{BS} \rightarrow \text{SS} : \{\text{GKEK, GTEK}\}_{\text{KEK}}$$

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.,

$$\text{GTEK}_{\text{new}} = f(\text{GTEK}_{\text{old}})$$
$$\text{GKEK}_{\text{new}} = f(\text{GKEK}_{\text{old}})$$

3. When a SS leaves the group, the BS triggers a rekey event at every existing SS by broadcasting a random number r, protected by the old GKEK:

$$\text{BS} \Rightarrow \text{all SS: } r_{\text{GKEK}}$$

Then each existing SS computes the new group key GTEK by using the random number and the old group key:

$$\text{GTEK}_{\text{new}} = f(\text{GTEK}_{\text{old}})$$

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 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.

4.2. Performance Comparison

In this section, we compare the communication overheads of various rekeying schemes, including 802.16e's MBRA, the binary-tree based scheme [4], the n-ary ($n \geq 3$) tree-based scheme that we propose, 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 VI. Messages under MBRA.

Initial keying	$\text{BS} \rightarrow \text{SS} : \{\text{GKEK}\}_{\text{KEK}}$
Rekey at key expiry	$\text{BS} \Rightarrow \text{all SS} : \{\text{GTEK}\}_{\text{GKEK}}$
Rekey at SS departure	$\text{BS} \rightarrow \text{SS} : \{\text{GKEK}\}_{\text{KEK}}$

Table VII. Messages under tree-based rekeying schemes.

Initial keying	$\text{BS} \rightarrow \text{SS} : \{\text{SGKEK}_1, \text{SGKEK}_2, \ldots\}_{\text{KEK}}$
Rekey at key expiry	$\text{BS} \Rightarrow \text{all SS} : \{\text{GTEK}\}_{\text{SGKEK}_{\text{top-level}}}$
Rekey at SS departure	$\text{BS} \rightarrow \text{SS} : \{\text{SGKEK}_1, \text{SGKEK}_2, \ldots\}_{\text{KEK}}$ $\text{BS} \Rightarrow \text{SGroup} :$ $\{\text{SGKEK}_1, \text{SGKEK}_2, \ldots\}_{\text{SGKEK}_i}$

Table VIII. Messages under the new rekeying scheme.

Initial key	$\text{BS} \rightarrow \text{SS: } \{\text{GKEK, GTEK}\}_{\text{KEK}}$
Rekey at key expiry	No message needed
Rekey at SS departure	$\text{BS} \Rightarrow \text{all SS:}\{\text{random number}\}_{\text{GKEK}}$

The detail for the message transmissions under MBRA, the tree-based schemes, and our new rekeying scheme are given in Tables VI–VIII, respectively.

Under the tree-based rekeying schemes, the number of keys transmitted in the SS-departure event depends on which subgroups are impacted by the given departure.

Table IX 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,

DOI: 10.1002/sec

Table IX. Number of messages per event type.

	MBRA	Binary tree groups	*n*-ary tree groups	New scheme
SS joins a BS	1 unicast per SS			
GTEK expires	1 broadcast			0 transmissions
GKEK expires	1 broadcast			0 transmissions
SS leaves the BS	*N* unicasts	$2^k - 1$ broadcasts $s - 1$ unicasts	$d(n - 1)$ broadcasts $s - 1$ unicasts	1 broadcast

Table X. Number of transmissions and receptions per event type.

	MBRA	Binary tree groups	*n*-ary tree groups	New cast MBRA
SS joins a BS	1 *Tx*; *N Rx*			
GTEK expires	1 *Tx*; *N Rx*			0 *Tx*, 0 *Rx*
GKEK expires	1 *Tx*; *N Rx*			0 *Tx*, 0 *Rx*
SS leaves the BS	*N Tx* *N Rx*	$2^k - 1 + g - 1$ *Tx* $N(2^k - 1) + g - 1$ *Rx*	$d(n - 1) + s - 1$ *Tx* $d(n - 1) \cdot N + s - 1$ *Rx*	1 *Tx* *N Rx*

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 X lists the number of transmissions (*Tx*) and receptions (*Rx*) for each event, under theses schemes.

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

5. Conclusions

In this paper, we studied the rekeying issue in WiMAX networks. The existing IEEE 802.16e rekeying scheme—the 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.

Acknowledgement

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

References

1. Johnston D, Walker J. Overview of IEEE 802.16 security. *IEEE Security & Privacy* 2004; **2**(3): 40–48.
2. IEEE. *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 Std 802.16e-2005*. IEEE, 2005.
3. Shon T, Choi W. An Analysis of Mobile WiMAX Security: Vulnerabilities and Solutions. *Lecture Notes in Computer Science*, 4658, Springer-Verlag, 2007; 88–97.
4. Huang C, Chang J. Responding to security issues in wimax networks. *IT Professional* 2008; **10**(5): 15–21.
5. Xu S, Matthews M, Huang C. Security issues in privacy and key management protocols of IEEE 802.16. *Proceedings of the 44th annual Southeast regional conference*, 2006; 113–118.
6. Eren E. Wimax security architecture—analysis and assessment. *IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, 2007; 673–677.
7. Maccari L, Paoli M, Fantacci R. Security analysis of IEEE 802.16. *IEEE International Conference on Communications*, 2007; 1160–1165.
8. Han T, Zhang N, Liu K, Tang B, Liu Y. Analysis of mobile wimax security: vulnerabilities and solutions. *5th IEEE International Conference on Mobile Ad Hoc and Sensor Systems*, 2008; 828–833.
9. Zhou Y, Fang Y. Security of ieee 802.16 in mesh mode. *IEEE Military Communications Conference*, 2006; 1–6.

DOI: 10.1002/sec