Key Management for Pyramidal Security Model of Multicast Communication in Mobile Ad Hoc Networks

Bo Rong[1], Yi Qian[1], Rose Qingyang Hu[2], Sghaier Guizani[3], Michel Kadoch[3]
[1]Department of Electrical and Computer Engineering
University of Puerto Rico at Mayaguez
Mayaguez, PR 00681,USA
[2]Department of Electrical and Computer Engineering
Mississippi State University
Mississippi State, Mississippi 39762
[3]Department of Electrical Engineering
Ecole de technologie superieure
Montreal, Quebec, Canada H3C 1K3

Abstract — **Many applications in mobile ad hoc networks involve collaborative computing among a large number of nodes and are thus group-oriented in nature. Multicast is a very efficient way of supporting group-oriented applications, especially in mobile/wireless environments where bandwidth is scarce and equipment has limited power. For deploying group-oriented applications in an adversarial environment such as battlefield or disaster rescue cases, it is necessary to provide support for secure multicast communication. In this paper, pyramidal security model is proposed to safeguard a special multicast scenario of multi-security-level information broadcast in an information sharing domain of mobile ad hoc network. In order to give an efficient key management solution to the pyramidal security model, we propose an integrated tree key graph scheme. Performance comparison proves that this scheme possesses many advantages over its counterparts.**

I. Introduction

Mobile Ad Hoc Network (MANET) is an autonomous system of mobile nodes connected by wireless links. Each node operates not only as an end-system, but also as a router to forward packets. MANET does not require any fixed infrastructure such as base stations. Therefore, it is an attractive networking option for connecting mobile devices quickly and spontaneously. At the same time, the popularity of group computing grows rapidly nowadays. Example applications include audio/video conferencing as well as one-to-many data dissemination in critical situations such as battlefield or disaster rescue scenarios. Multicast is an efficient way of supporting group-oriented applications, especially in mobile/wireless environments where bandwidth is scarce and equipment has limited power. With the rapid growth of demand, the multicast technology in mobile ad hoc network has attracted a lot of attention recently [1, 2]. Since group-oriented applications in mobile ad hoc network often have to face hostile environment, security protection of the multicast communication has become a primary concern. As a special case of secure multicast, information broadcast to the users of different security levels in an information sharing domain is an indispensable function of battlefield and disaster rescue applications. To support this function, the pyramidal security model is proposed in this paper.

Unlike wired networks, the nodes in mobile ad hoc network are free to move randomly and can organize themselves in an arbitrary manner. As a result, group members are likely to join/leave the multicast session frequently, and an efficient key management scheme to deal with dynamic multicast is urgently required. The topic of key management for secure multicast was firstly discussed in [3, 4], which led to several solutions [5~11]. However, these previous studies only investigated the key management scheme for an independent multicast group, and no one has addressed the need of managing the frequent key changes for a set of multicast groups in a pyramidal security model. To fill this gap, the scheme of integrated tree key graph is proposed in this paper to provide key management for all multicast groups in a pyramidal security model.

The rest of the paper is organized as follows. At first, we introduce the pyramidal security model and the basic knowledge of key management for secure multicast in Section 2 and Section 3 respectively. Then, separated star/tree key graph is presented as a preliminary key management scheme for pyramidal security model in Section 4. Furthermore, to achieve a more efficient solution, the scheme of integrated tree key graph is proposed and investigated in Section 5. In the end, Section 6 summarizes the results.

II. The Pyramidal Security Model for Multicast Communication in Mobile Ad hoc Networks

Users working for the same mission in mobile ad hoc network form an information sharing domain. A mobile ad hoc network can contain one or several information sharing domains, and an information sharing domain may involve a part of users or all users in the mobile ad hoc network. We suppose that in an information sharing domain there are N security levels, and security level i ($i = 1, 2, \ldots, N$) has superior security power over any other security level j that satisfies $j < i$. All users of security level i build up security group G_S^i. Consequently, in term of security power, the relationship among security groups G_S^1, G_S^2, …, G_S^N, is subject to a pyramidal structure illustrated in Fig.1.

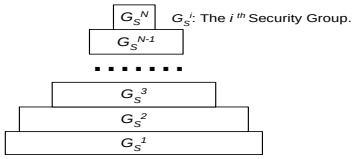

Fig.1. Pyramidal security model of multicast communication in mobile ad hoc network

Based on the definition of security group, N multicast groups (G_M^i, $i = 1, 2, ..., N$), are constructed to provide information broadcast among the users of different security levels in an information sharing domain. The following relationship holds between security groups and multicast groups:

$$G_M^i = G_S^i \cup G_S^{i+1} \cup ... \cup G_S^N \quad (1)$$

It indicates that the i^{th} multicast group G_M^i contains the users from security groups G_S^i, G_S^{i+1}, …, G_S^N. This design of multicast groups reflects the fact that the higher security groups have right to join and monitor the communication of lower security groups. For further explanation, equation (2) and Fig.2 demonstrate the relationship among different multicast groups.

$$G_M^1 \supset G_M^2 \supset ... \supset G_M^N \quad (2)$$

Assuming that there are x_i users in security group G_S^i ($i = 1, 2, \ldots, N$), then an N level pyramidal security model can be explicitly represented by $PSM_S(x_1, x_2, \ldots, x_N)$. In case of dynamic multicast, we suppose the number of G_S^i users is a random variable X_i, which is subject to Gaussian distribution with the mean of x_i and a small value of variance. Then the notation $PSM_S(x_1, x_2, \ldots, x_N)$ can still be used to represent a pyramidal security model in dynamic multicast environment. Moreover, it is easy to know in $PSM_S(x_1, x_2, \ldots, x_N)$ there are $y_i = (x_i + x_{i+1} + \ldots + x_N)$ users in G_M^i. In this respect, the pyramidal security model of $PSM_S(x_1, x_2, \ldots, x_N)$ can also be denoted by $PSM_M(y_1, y_2, \ldots, y_N)$.

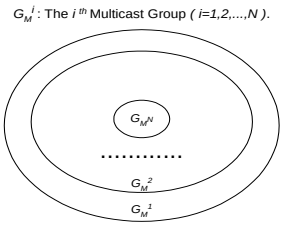

Fig.2. The relationship among different multicast groups in pyramidal security model

III. Key Management for Secure Multicast of an Independent Group

Previous study of security protection for multicast networks depicts an encryption procedure as the following [8]. As the prerequisite of a multicast session, a trusted server is set up to maintain the membership information and exercise group access control. When a client needs to join the group, the client and server mutually authenticate using an authentication protocol. Having been authenticated and accepted into the group, each member shares with the server a key, to be called the member's individual key. For group communication, the server distributes to each member a group key which is shared by all group members. In this paper, key means a symmetric key, unless explicitly stated otherwise.

The multicast communication in mobile ad hoc network is a typical dynamic scenario. To achieve security in such an environment, the group key has to be changed after every join and leave so that a former group member has no access to the current communication and a new member has no access to the previous communication. After a join request is accepted, the new group key can be sent via unicast to the new member (encrypted with its individual key) and via multicast to existing group members (encrypted with the previous group key). Thus, changing the group key for a join request does not cost too much. After a leave request is approved, however, the previous group key can no longer be used and the new group key must be encrypted for each remaining group member using its individual key. This means that changing the group key securely after a leave incurs cryptographic computation and communication costs proportional to the group size. Therefore, large groups whose members join and leave frequently pose a scalability problem. To solve this problem, Wallner et al. [3] and Wong et al. [4] independently proposed a scalable group key management scheme by constructing a logical key graph. Their method is briefly introduced as follows.

A. Basic Concept of Secure Group

A secure group can be formalized as a triple (*U, K, R*) where:

- *U* is a finite and nonempty set of users;
- *K* is a finite and nonempty set of keys;
- *R* is a binary relation between *U* and *K*, called the user-key relation of the secure group. User *u* has key *k* if and only if (*u*, *k*) is in *R*.

Each secure group has a trusted key server responsible for generating and securely distributing keys in *K* to users in the group. Specifically, the trusted server knows the user set *U* and the key set *K* and maintains the user-key relation *R*. Every user in *U* has a key in *K*, called its individual key, which is shared only with the trusted server and is used for pairwise confidential communication with the trusted server. There is a group key in *K*, shared by the trusted server and all users in *U*. The group key can be utilized by each user to send messages confidentially to other members of the group. Keys other than individual key and group key are named auxiliary keys.

B. Basic Concept of Key Graph

A key graph is a directed acyclic graph *G* with two types of nodes: *u*-nodes representing users and *k*-nodes representing keys. Given a key graph *G*, it specifies a secure group (*U, K, R*) according to the following rules.

1) There is a one-to-one correspondence between *U* and the set of *u*-nodes in *G*.
2) There is a one-to-one correspondence between *K* and the set of *k*-nodes in *G*.
3) (*u, k*) is in *R* if and only if *G* has a directed path from the *u*-node that corresponds to *u* to the *k*-node that corresponds to *k*.

As an example, a key graph of 5 *u*-nodes and 8 *k*-nodes is shown in Fig.3. Mainly two classes of key graphs, the star key graph and the tree key graph, are studied by previous research. Detailed performance analysis of both can be found in [4].

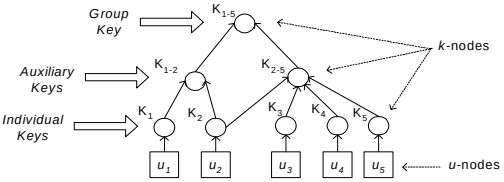

Fig.3. An example of key graph

Different from previous work on key management of independent multicast group, this paper highlights the key management of a set of multicast groups in pyramidal security model. The rest of this paper focuses on three key management schemes for pyramidal security model, i.e., separated star key graph, separated tree key graph, and integrated tree key graph.

IV. The Scheme of Separated Star/Tree Key Graph for Pyramidal Security Model

When designing a key management scheme for pyramidal security model, an easy idea is to consider different multicast groups separately. This method of key management is called separated key graph. Assuming that there are *N* multicast groups (G_M^i, i = 1, 2, ..., *N*) in pyramidal security model, *N* independent key graphs have to be constructed and maintained for G_M^1, G_M^2, … , G_M^N respectively. Separated key graph has an obvious advantage that all the previous research results on key management of independent multicast group can be utilized. In particular, if star/tree key graph is employed in the scheme of separated key graph, we call it separated star/tree key graph.

A. The Scheme of Separated Star Key Graph

For an independent multicast group, star key graph is defined as a special secure group (*U, K, R*) where each user in *U* have only two keys, its individual key and a group key shared by all group members. An example is given in Fig. 4 to demonstrate the star key graph of a multicast group of 9 users.

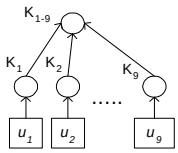

Fig.4. The star key graph of a multicast group of 9 users

If a star key graph is applied to an independent multicast group of *x* users, the number of necessary keys and the encryption/ decryption cost of a join/leave request are listed in Table 1 and Table 2 respectively. Here, the encryption/decryption cost means the amount of encryptions/decryptions conducted on server/user. Moreover, for each join/leave request, the user that requests the join/leave is called the requesting user, and the other users in the group are non-requesting users.

TABLE 1
NUMBER OF KEYS HELD BY THE SERVER AND BY EACH USER IN AN INDEPENDENT STAR KEY GRAPH

	Held by Server	Held by User
Number of Keys	x +1	2

TABLE 2
ENCRYPTION/DECRYPTION COST OF A JOIN/LEAVE REQUEST IN AN INDEPENDENT STAR KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	2	1	1
Leave	x-1	0	1

Since an N level pyramidal security model contains N multicast groups, the scheme of separated star key graph should establish a set of N independent star key graphs accordingly. For instance, Fig.5 shows that the scheme of separated star key graph has to construct three star key graphs for the pyramidal security model $PSM_S(18,6,3)$.

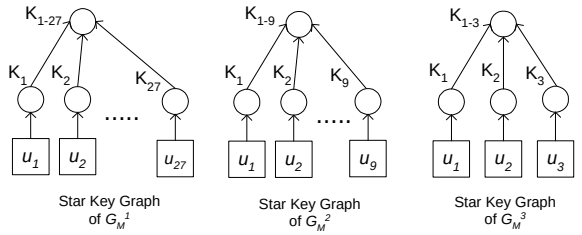

Fig.5. A set of three independent star key graphs constructed for $PSM_S(18,6,3)$

As to an N level pyramidal security model PSM_S (x_1, x_2,…, x_N), in the star key graph of G_M^i , the number of keys held by server is ($1+x_i+ x_{i+1}+\ldots+ x_N$). Therefore, the total number of keys held by the server in all star key graphs can be calculated by:

$$\sum_{i=1}^{N}(1+\sum_{j=i}^{N}x_j) = N + Nx_N + (N-1)x_{N-1} + \ldots + ix_i + \ldots + 1\cdot x_1 = N + \sum_{i=1}^{N} ix_i \quad (3)$$

In addition, in the scheme of separated star key graph, the number of keys held by a user of G_S^i is $2i$. Thus, the average number of keys held by a user in PSM_S (x_1, x_2, …, x_N) is $2\sum_{i=1}^{N} ix_i \Big/ \sum_{i=1}^{N} x_i$. Table 3 summarizes the above results.

TABLE 3
NUMBER OF KEYS HELD BY THE SERVER AND BY EACH USER IN THE SCHEME OF SEPARATED STAR KEY GRAPH

	Held by Server	Held by User
Number of Keys	$N+\sum_{i=1}^{N} ix_i$	$2\sum_{i=1}^{N} ix_i \Big/ \sum_{i=1}^{N} x_i$

If a user of G_S^i enters into or departs from the mobile ad hoc network, it incurs join/leave requests in the multicast groups of subset $\{G_M^j , j < i\}$. Based on this fact and the results in Table 2, the encryption/decryption cost of a join/leave event of a G_S^i user in the scheme of separated star key graph is shown in Table 4.

TABLE 4
THE COST OF A JOIN/LEAVE EVENT OF A G_S^i USER IN THE SCHEME OF SEPARATED STAR KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	$2i$	i	i
Leave	$\sum_{j=1}^{i}\sum_{k=j}^{N} x_k - i$	0	i

To attain the statistical performance of separated star key graph, more definitions need to be developed. For a coming/leaving user in $PSM_S(x_1, x_2,\ldots,x_N)$, we suppose it belongs to security group G_S^I, where I is a discrete random variable complying with the following distribution.

$$\mathrm{P}\{I=i\} = \begin{cases} x_i \Big/ \sum_{j=1}^{N} x_j, & if\ 1\le i\le N \\ 0, & otherwise \end{cases} \quad (4)$$

Then, the statistical performance of separated star key graph is shown in Table 5.

TABLE 5
THE AVERAGE COST OF A USER JOIN/LEAVE EVENT IN THE SCHEME OF SEPARATED STAR KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	2E(I)	E(I)	E(I)
Leave	$E(\sum_{j=1}^{I}\sum_{k=j}^{N} x_k) - E(I)$	0	E(I)

where,

$$E(I) = \sum_{i=1}^{N}(iP\{I=i\}) = \sum_{i=1}^{N}(ix_i \Big/ \sum_{j=1}^{N} x_j) = \sum_{i=1}^{N} ix_i \Big/ \sum_{j=1}^{N} x_j \quad (5)$$

$$E(\sum_{j=1}^{I}\sum_{k=j}^{N} x_k) = \sum_{i=1}^{N}(P\{I=i\}\sum_{j=1}^{i}\sum_{k=j}^{N} x_k) = \sum_{i=1}^{N}\left(\frac{x_i \sum_{j=1}^{i}\sum_{k=j}^{N} x_k}{\sum_{j=1}^{N} x_j}\right) = \sum_{i=1}^{N}(x_i \sum_{j=1}^{i}\sum_{k=j}^{N} x_k) \Big/ \sum_{j=1}^{N} x_j \quad (6)$$

B. The Scheme of Separated Tree Key Graph

In case of an independent multicast group, tree key graph is the special class of a secure group (U, K, R) whose key graph G is a single-root tree. A tree key graph is specified by two parameters.

- The height h of the tree is the length (in number of edges) of the longest directed path in the tree.
- The degree d of the tree is the maximum number of incoming edges of a node in the tree.

Since the leaf node of each path is a u-node, each user in U has at most h keys. An example is given in Fig.6 to demonstrate the tree key graph (d=3) of a multicast group of 9 users.

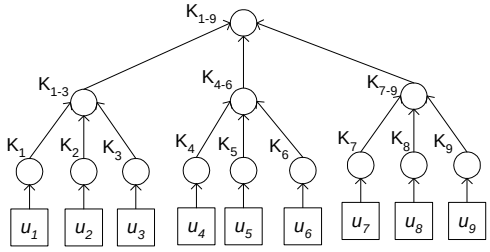

Fig.6. The tree key graph of a multicast group of 9 users

If a tree key graph is applied to an independent multicast group of x users, the total number of keys held by the server/user is listed in Table 6. For the sake of simplifying the analysis, this paper mainly concerns the full and balanced tree where the condition $x=d^{h-1}$ is satisfied.

TABLE 6
NUMBER OF KEYS HELD BY THE SERVER AND BY EACH USER IN AN INDEPENDENT TREE KEY GRAPH

	Held by Server	Held by User
Number of Keys	$(dx-1)/(d-1)$	$\log_d^x+1$

As for tree key graph, [4] proposed three approaches to construct and send rekey messages, namely User-Oriented Rekeying, Key-Oriented Rekeying, and Group-Oriented Rekeying. It also showed that the approach of Group-Oriented Rekeying has the best performance of all. Consequently, this paper addresses only Group-Oriented Rekeying when tree key graph is studied. The encryption/decryption cost of a join/leave request in tree key graph using Group-Oriented Rekeying is listed in Table 7.

TABLE 7
ENCRYPTION/DECRYPTION COST OF A JOIN/LEAVE REQUEST IN AN INDEPENDENT TREE KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	$2\log_d x$	$\log_d x$	$d/(d-1)$
Leave	$d \log_d x$	0	$d/(d-1)$

When employing the scheme of separated tree key graph to manage an N level pyramidal security model, a set of N independent tree key graphs would be established. For instance, Fig.7 shows that the scheme of separated tree key graph constructs three tree key graphs for the pyramidal security model $PSM_S(18,6,3)$.

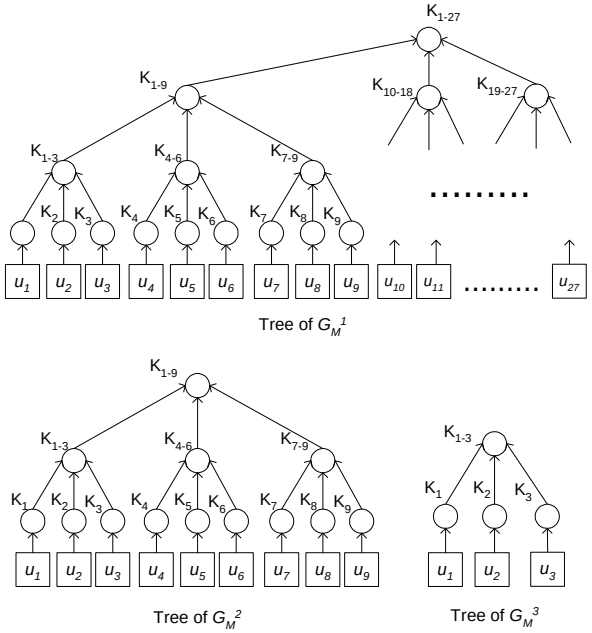

Fig.7. A set of three independent tree key graphs constructed for $PSM_S(18,6,3)$

For an N level pyramidal security model $PSM_S(x_1, x_2, \ldots, x_N)$, the number of keys held by server in the tree key graph of G_M^i is $(d(x_i + x_{i+1} + \ldots + x_N)-1)/(d-1)$. Therefore, the total number of keys held by the server in all tree key graphs can be calculated by:

$$\sum_{i=1}^{N}\left(\frac{d}{d-1}\sum_{j=i}^{N}x_j-\frac{1}{d-1}\right)=\frac{d}{d-1}(Nx_N+\ldots+ix_i+\ldots+1\cdot x_1)-\frac{N}{d-1} \quad (7)$$
$$=\frac{d}{d-1}\sum_{i=1}^{N}ix_i-\frac{N}{d-1}$$

Similarly, in the scheme of separated tree key graph the number of keys held by a user of G_S^i is:

$$\sum_{j=1}^{i}\left(\log_d \prod_{k=j}^{N} x_k+1\right)=\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k+i \quad (8)$$

Thus, the average number of keys held by a user in PSM_S $(x_1, x_2, \ldots, x_N)$ is:

$$\sum_{i=1}^{N}x_i\left(\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k+i\right)\Bigg/\sum_{i=1}^{N}x_i \quad (9)$$

Table 8 summarizes the above results.

TABLE 8
NUMBER OF KEYS HELD BY THE SERVER AND BY EACH USER IN THE SCHEME OF SEPARATED TREE KEY GRAPH

	Number of Keys
Held by Server	$\frac{d}{d-1}\sum_{i=1}^{N}ix_i-\frac{N}{d-1}$
Held by User	$\sum_{i=1}^{N}x_i\left(\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k+i\right)\Big/\sum_{i=1}^{N}x_i$

If a user of G_S^i enters into or departs from the mobile ad hoc network, it incurs join/leave requests in the multicast groups of subset $\{G_M^j, j < i\}$. Based on this fact and the results in Table 7, the encryption/decryption cost of a join/leave event of a G_S^i user in the scheme of separated tree key graph is shown in Table 9.

TABLE 9
THE COST OF A JOIN/LEAVE EVENT OF A G_S^i USER IN THE SCHEME OF SEPARATED TREE KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	$2\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k$	$\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k$	$id/(d-1)$
Leave	$d\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k$	0	$id/(d-1)$

To attain the average performance of separated tree key graph in case of $PSM_S(x_1, x_2, \ldots, x_N)$, the statistical model defined in equation (4) is used, and the results are listed in Table 10.

TABLE 10
THE AVERAGE COST OF A USER JOIN/LEAVE EVENT IN THE SCHEME OF SEPARATED TREE KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	$2E(\log_d \prod_{j=1}^{I}\prod_{k=j}^{N} x_k)$	$E(\log_d \prod_{j=1}^{I}\prod_{k=j}^{N} x_k)$	$E(I)d/(d-1)$
Leave	$dE(\log_d \prod_{j=1}^{I}\prod_{k=j}^{N} x_k)$	0	$E(I)d/(d-1)$

where,

$$E(\log_d \prod_{j=1}^{I}\prod_{k=j}^{N} x_k)=\sum_{i=1}^{N}(P\{I=i\}\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k)$$
$$=\sum_{i=1}^{N}(x_i\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k)\Big/\sum_{j=1}^{N}x_j) \quad (10)$$
$$=\sum_{i=1}^{N}x_i\log_d \prod_{j=1}^{i}\prod_{k=j}^{N} x_k\Big/\sum_{j=1}^{N}x_j$$

and, $E(I)$ can be computed by equation (5).

V. THE SCHEME OF INTEGRATED TREE KEY GRAPH FOR PYRAMIDAL SECURITY MODEL

A. Construction of Integrated Tree Key Graph

The shortcoming of the separated star/tree key graph is that it does not make use of the relationship among different multicast groups in pyramidal security model. For this reason, the scheme of separated star/tree key graph has to establish a set of independent tree key graphs, which lead to low efficiency in term of the number of keys held by the server/user and the cost of encryption/decryption. To overcome this shortage, the scheme of integrated tree key graph is proposed in this section.

The objective of integrated tree key graph is to use only one tree key graph to manage all the multicast groups in pyramidal security model. To achieve this goal, the integrated tree key graph should be established regarding the inherent relationship among different multicast groups of the pyramidal security model. The algorithm to construct such an integrated tree is demonstrated as follows.

The Algorithm of Constructing *d-ary* Integrated Tree Key Graph for *N* Level Pyramidal Security Model

1. Initially, construct a tree key graph of degree d for G_M^N
2. For i=N-1 to 1 {
   Use G_M^{i+1} as an immediate left sub-tree to construct a new tree key graph of degree d for G_M^i.
   }
3. Return the tree key graph of G_M^1 as the integrated tree key graph for pyramidal security model.

As an example, construction of the integrated tree key graph (d=3) for pyramidal security model PSM_S(18,6,3) is shown in Fig.8.

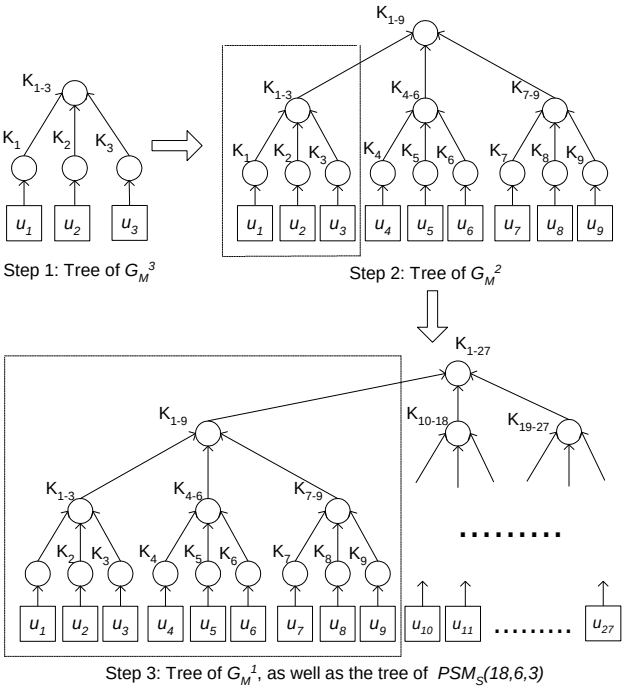

Fig.8. Construction of the integrated tree key graph (d = 3) for pyramidal security model PSM_S(18, 6, 3)

The integrated tree key graph has the similar shape as the independent tree of G_M^1 of separated tree key graph has. Nevertheless, there are at least two major differences between them. Firstly, the integrated tree contains group keys of G_M^1, G_M^2, …, G_M^N, while the independent tree of G_M^1 only contains the group key of G_M^1. This fact implies that the independent tree of G_M^1 has more auxiliary keys. Secondly, the integrated tree is a sorted tree, where the users of high security power are placed in the left. As a result, when a user of G_S^i comes, it must be inserted in the sub-tree of G_M^i, and this procedure costs more computing resource. On the contrary, in the independent tree of G_M^1, the users are placed in a random order. So, when a user of G_S^i comes, it can be inserted in any available position, and thus less computing resource is costed.

B. Performance Analysis and Comparison

Despite the differences, with respect to the number of keys held by server/user and the cost of encryption/decryption, integrated tree key graph has exactly the same performance as the independent tree of G_M^1 of separated tree key graph has. The detailed results are demonstrated in Table 11 and Table 12.

TABLE 11
NUMBER OF KEYS HELD BY THE SERVER AND BY EACH USER IN THE SCHEME OF INTEGRATED TREE KEY GRAPH

	Held by Server	Held by User
Number of Keys	$(d\sum_{i=1}^{N} x_i - 1)/(d-1)$	$\log_d^{\sum_{i=1}^{N} x_i} + 1$

TABLE 12
THE ENCRYPTION/DECRYPTION COST OF A USER JOIN/LEAVE EVENT IN THE SCHEME OF INTEGRATED TREE KEY GRAPH

	Server	Requesting User	Non-requesting User
Join	$2\log_d^{\sum_{i=1}^{N} x_i}$	$\log_d^{\sum_{i=1}^{N} x_i}$	*d/(d-1)*
Leave	$d\log_d^{\sum_{i=1}^{N} x_i}$	0	*d/(d-1)*

To carry out performance comparison between the integrated tree key graph and separated star/tree key graph, numerical results are given in this sub-section for the pyramidal security model of $PSM_M(4^N,4^{N-1},\ldots,4)$. Here, $PSM_M(4^N,4^{N-1},\ldots,4)$ means the pyramidal security model of $PSM_S(x_1, x_2,\ldots,x_N)$ where the condition $(x_i+ x_{i+1}+\ldots+ x_N)= 4^{N-i+1}$ $(i=1,2,\ldots,N)$ is met. Moreover, the degree d of any tree key graph in the simulation is set to be 4, since [4] pointed out that independent tree key graph has the best performance when d = 4.

Based on Table 3, 8, 11, Fig.9 and Fig.10 demonstrate the performance of three key management schemes in terms of the number of necessary keys. Fig.9 indicates that integrated tree key graph and separated star key graph have the similar number of server keys, which is much less than that separated tree key graph needs. In addition, as shown in Fig.10, integrated tree key graph owns more user keys than separated star key graph, and less user keys than separated tree key graph. Note that, as an important rule in the comparison of key numbers, the number of keys held by server is the major criterion and the number of keys held by user is only an auxiliary criterion.

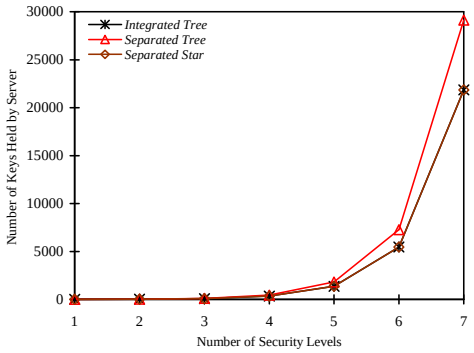

Fig.9. Number of keys held by server vs. number of security levels in $PSM_M(4^N,4^{N-1},\ldots,4)$

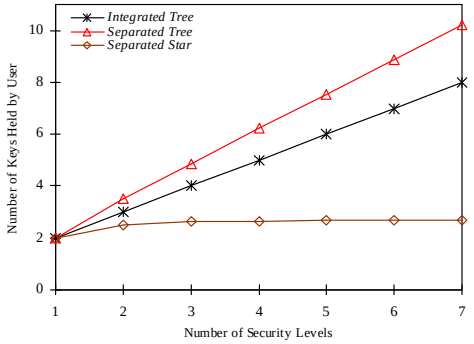

Fig.10. Number of keys held by user vs. number of security levels in $PSM_M(4^N,4^{N-1},\ldots,4)$

With the purpose of assessing the overall average encryption/ decryption cost, we suppose that a user is with equal probability to join or leave. Then, based on Table 5, 10, 12, Table 13 summarizes the results.

TABLE 13
AVERAGE ENCRYPTION/DECRYPTION COST PER USER EVENT IN THREE KEY MANAGEMENT SCHEMES

	Server	Requesting User	Non-requesting User
Separated Star	$\frac{1}{2}E(\sum_{j=1}^{I}\sum_{k=j}^{N} x_k) + \frac{1}{2}E(I)$	$\frac{1}{2}E(I)$	$E(I)$
Separated Tree	$\frac{d+2}{2}E(\log_d^{\prod_{j=1}^{I}\sum_{k=j}^{N} x_k})$	$\frac{1}{2}E(\log_d^{\prod_{j=1}^{I}\sum_{k=j}^{N} x_k})$	$\frac{dE(I)}{d-1}$
Integrated Tree	$\frac{d+2}{2}\log_d^{\sum_{i=1}^{N} x_i}$	$\frac{1}{2}\log_d^{\sum_{i=1}^{N} x_i}$	$\frac{d}{d-1}$

Using Table 13, Fig.11, Fig.12, and Fig.13 illustrate the average encryption/decryption cost per user event in three key management schemes. Fig.11 demonstrates that integrated tree key graph outweighs the other two schemes with respect to the average encryption cost per user event on server. Fig.12 and Fig. 13 show that integrated tree key graph has the intermediate performance regarding the average decryption cost per user event on a requesting user/non-requesting user. Similar as the comparison of key number, in terms of encryption/decryption cost, the average encryption cost per user event on server is the major criterion, and the average decryption cost per user event on a requesting user/non-requesting user is only an auxiliary criterion.

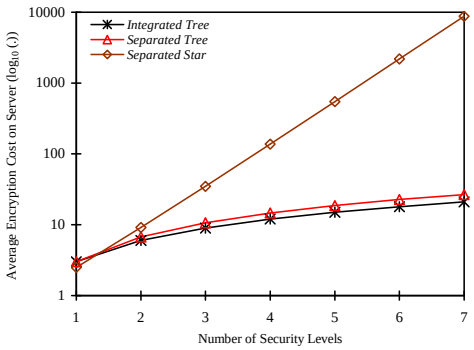

Fig.11. Average encryption cost per user event on server vs. number of security levels in $PSM_M(4^N,4^{N-1},\ldots,4)$

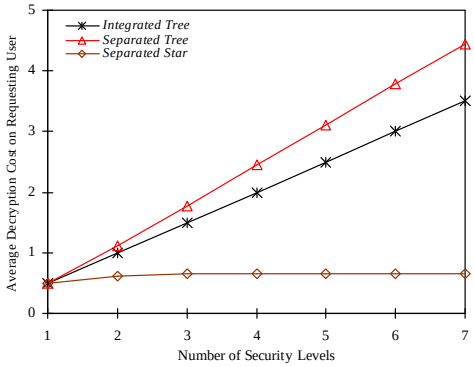

Fig.12. Average decryption cost per user event on a requesting user vs. number of security levels in $PSM_M(4^N,4^{N-1},\ldots,4)$

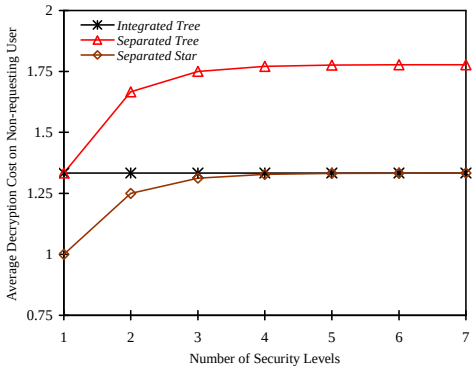

Fig.13. Average decryption cost per user event on a non-requesting user vs. number of security levels in $PSM_M(4^N,4^{N-1},\ldots,4)$

From the above discussion we can conclude that the scheme of integrated tree key graph has better performance than the scheme of separated star/tree key graph regarding the overall performance. Integrated tree key graph achieves this advantage because it makes a good use of the relationship among different multicast groups in the pyramidal security model.

VI. Conclusions

Pyramidal security model is presented in this paper to support multi-security-level information broadcast in an information sharing domain of mobile ad hoc network. Aiming at providing pyramidal security model with an efficient key management solution, three schemes, namely separated star key graph, separated tree key graph, and integrated tree key graph, are investigated. The performance comparison among these three key management schemes indicates that integrated tree key graph has the best overall performance in term of the number of keys and the encryption/decryption cost.

References

1. C. de Morais Cordeiro, H. Gossain, D. P. Agrawal, "Multicast over wireless mobile ad hoc networks: present and future directions," *IEEE Network*, Volume: 17, Issue: 1, Pages: 52 - 59, Jan.-Feb. 2003.
2. W. Liao, M.-Y. Jiang, "Family ACK tree (FAT): supporting reliable multicast in mobile ad hoc networks," *IEEE Transactions on Vehicular Technology*, Volume: 52, Issue: 6, Pages:1675-1685, Nov. 2003.
3. D. M. Wallner, E. J. Harder, and R. C. Agee, "Key management for multicast: Issues and architectures," *RFC 2627*, June 1999.
4. C. K. Wong, M. Gouda, S. S. Lam, "Security group communication using key graph," *IEEE/ACM Transactions on Networking*, Volume: 8, Issue: 1, Pages: 16 - 30, Feb. 2000.
5. M. Abdalla, Y. Shavitt, A. Wool, "Key management for restricted multicast using broadcast encryption," *IEEE/ACM Transactions on Networking*, Volume: 8, Issue: 4, Pages: 443 - 454, Aug. 2000.
6. R. Poovendran, J. S. Baras, "An information-theoretic approach for design and analysis of rooted-tree-based multicast key management schemes," *IEEE Transactions on Information Theory*, Volume: 47, Issue: 7, Pages: 2824 - 2834, Nov. 2001.
7. W. Trappe, Jie Song, R. Poovendran, K. J. R. Liu, "Key management and distribution for secure multimedia multicast," *IEEE Transactions on Multimedia*, Volume: 5, Issue: 4, Pages: 544 - 557, Dec. 2003.
8. K.-C. Chan, S. H. G. Chan, "Key management approaches to offer data confidentiality for secure multicast," *IEEE Network*, Volume: 17, Issue: 5, Pages: 30 - 39, Sept.-Oct. 2003.
9. Y. Sun, W. Trappe, K. J. R. Liu, "A scalable multicast key management scheme for heterogeneous wireless networks," *IEEE/ACM Transactions on Networking*, Volume: 12, Issue: 4, Pages: 653 - 666, Aug. 2004.
10. M. P. Howarth, S. Iyengar, Z. Sun, H. Cruickshank, "Dynamics of key management in secure satellite multicast," *IEEE Journal on Selected Area in Communications*, Volume: 22, Issue: 2, Pages: 308 - 319, Feb. 2004.
11. H. Lu, "A novel high-order tree for secure multicast key management," *IEEE Transactions on Computer*, Volume: 54, Issue: 2, Pages: 214 - 224, Feb. 2005.