An Architecture for Key Management in Hierarchical Mobile Ad-hoc Networks

Kyung Hyune Rhee, Young Ho Park, and Gene Tsudik

Abstract: **In recent years, mobile ad-hoc networks have received a great deal of attention in both academia and industry to provide anytime-anywhere networking services. As wireless networks are rapidly deployed, the security of wireless environment will be mandatory. In this paper, we describe a group key management architecture and key agreement protocols for secure communication in mobile ad-hoc wireless networks (MANETs) overseen by unmanned aerial vehicles (UAVs). We use implicitly certified public keys method, which alleviates the certificate overhead and improves computational efficiency. The architecture uses a two-layered key management approach where the group of nodes is divided into: 1) Cell groups consisting of ground nodes and 2) control groups consisting of cell group managers. The chief benefit of this approach is that the effects of a membership change are restricted to the single cell group.**

Index Terms: **Group key management, implicit certificate, key management, secure mobile ad-hoc network.**

I. INTRODUCTION

Mobile ad-hoc networks (MANETs) offer convenient infrastructure-free communication over the shared wireless medium. MANETs are also regarded as an ideal technology for creating an instant communication network for civilian and military applications. In recent years, MANETs have received a great deal of attention in both academia and industry. This emerging technology aims to provide "anytime-anywhere" networking services on a potentially large-scale. MANET users (nodes) expect to communicate securely and seamlessly among themselves as well as with the rest of the global Internet. The growing deployment of MANETs in both commercial and military sectors heightens the security concerns, since the very nature of these networks makes them more vulnerable (than wired networs) to certain attacks, such as passive eavesdropping and denial of service.

Ad-hoc networks are created on demand without supporting from fixed infrastructure such as central servers and the organization of this network is based on groups of nodes. Secure group communication requires scalable and efficient group membership management with appropriate access control measures to protect data and cope with potential compromises. To this end, a secret key for data encryption must be distributed securely and efficiently to current members. Each time a membership change occurs, the group key must be changed to ensure backward and forward secrecy[1].

There have been several proposals on group key management in the recent literature. They range from key distribution schemes for large-scale single-sender multicast [1], [2] to contributory key agreement schemes for small any-to-any peer groups [3], [4]. Although most of them focus on wired networks, extensions to wireless networks (and MANETs) should be explored as such networks are becoming more commonplace.

Consequently, in this paper, we propose a group key management architecture for MANETs overseen by unmanned aerial vehicles (UAVs). In doing so, we exploit existing group key management algorithms. In addition, our design is equally applicable in several other scenarios. We divide a so-called operations theater managed by a single UAV into a control group and cell groups. The former is composed of mobile backbone nodes (MBNs) and the latter is the set of regular ground nodes clustered in cells; each cell is managed by a single MBN node. An MBN node manages its group by generating, updating and distributing the group key shared among all cell members. In addition, each MBN node functions as a peer member of its control group.

Key management within a cell group is carried out by the cell group manager (an MBN node) in a centralized fashion. The responsibility for key management of the control group is distributed among the cell group managers (all MBN nodes). We argue that, a centralized scheme is appropriate for cell group key management since most regular ground nodes are equipped with limited communication and computation devices. However, a control group can afford to employ a decentralized key management since MBN nodes have significant computing and communication power. Furthermore, decentralization helps in avoiding a single point of failure. It also provides a more scalable and efficient key management service in a MANET setting.

The rest of this paper is organized as follows. Section II discusses security threats and summarizes previous work. Section III presents the proposed architecture including the actual group key management protocols. Section IV provides an analysis and discusses the features of the proposed architecture. The paper concludes with the future work in Section V.

II. SECURITY THREATS AND RELATED WORK

We start by discussing the security threats faced by MANETs and then address the requirements necessary for security services. In the process, we also summarize relevant previous

Manuscript received March 22, 2003; approved for publication by Wha Sook Jeon, Division III Editor, April 26, 2004.

K. H. Rhee and Y. H. Park are with Division of Electronic, Computer and Telecommunication Engineering, Pukyong National University, Busan, Korea, email: khrhee@pknu.ac.kr, pyhoya@mail1.pknu.ac.kr.

G. Tsudik is with Information and Computer Science University of California at Irvine, Irvine, USA, email: gts@ics.uci.edu.

[1] Informally, backward secrecy is attained if it is computationally difficult for a member to discover group key(s) used before it joined the group. Whereas, forward secrecy is attained if it is computationally difficult for a member to discover group key(s) used after it left the group.

Table 1. Comparing group key management types.

	Centralized	
Type	Key distribution by the key center	
Computations	Center	Member
	Large	Small
Features	Single point of failure of key center	
Examples	Key graph [7], OFT [3]	
	Collaborative	
Type	Key agreement by member's contribution	
Computations	Large (similar complexity)	
Features	Multiple communication rounds	
Examples	GDH [17], TGDH [8], STR [9]	

work.

A. Security Threats and Services

The wireless communication medium renders a MANET more susceptible (than a wired network) to certain attacks ranging from passive eavesdropping to active impersonation, message replay and message distortion. Mobile nodes in a hostile environment, such as a battlefield, with relatively poor physical protection have a greater probability of being compromised. Therefore, it is necessary to consider not only malicious attacks from outside the network, but also take into account potential attacks launched from within the network by compromised nodes. The latter are attacks on the basic network mechanisms such as routing. Although such attacks are often ignored in the design of secure systems, we feel it is necessary to address them explicitly in MANETs.

Key management is a basic issue in secure communication and is certainly not limited to MANETs. However, the highly dynamic natures of MANETs (i.e., frequent changes in both topology and membership) make key management particularly challenging, moreover that in other wired and wireless networks. It is not surprising, therefore, that many traditional key management approaches are not well-suited for this environment. In popular network authentication architectures, two entities authenticate each other via certificates issued by a trusted certification authority (CA). While this model works well in wired networks, it fails in large ad-hoc wireless environments for several reasons resulted from lacking of infrastructure [5].

B. Requirements for Group Key Management

First and foremost, group key management must be performed securely with relevant keying material delivered via secure channels. Group key management must be resistant to a wide range of attacks by both outsiders and rogue members.

Group key management must also handle adjustments to group secrets usually triggered by either timeouts or membership changes in the underlying group communication system. In doing so, it must provide forward secrecy with respect to former members and backward secrecy with respect to newly admitted members. A stronger goal is to provide so-called key independence property [6] which states that knowledge of all (but one) group keys cannot be used to efficiently derive the one "missing" group key.

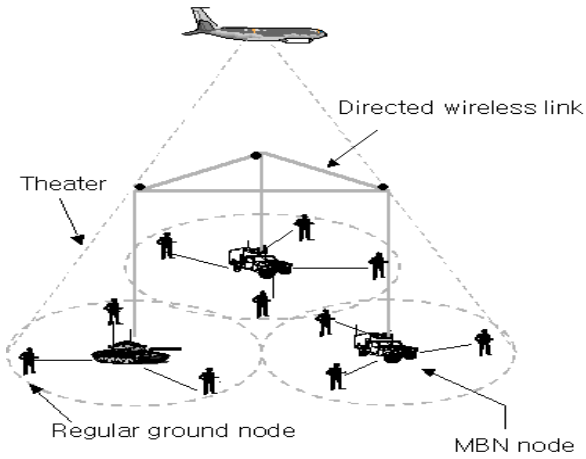

Fig. 1. Hierarchical MANET with MBNs and UAV.

In addition, group key management must be scalable, i.e., its protocols should be efficient in resource usage and should be able to minimize the effects of a membership change.

There has been a lot of research on group key management in the last decade. Prior work can be roughly partitioned into: Centralized approaches where a key center is responsible for creating and distributing the keys, and collaborative key agreement approaches that all members contribute group key agreement with no key center (refer to Table 1).

Many key-tree schemes, such as [3], [7]–[9] have been proposed for the purpose of minimizing communication and computation complexity of group re-keying. Most key-tree schemes are used in the context of centralized key management and reduce the cost of re-keying from $O(n)$ to $O(\log n)$ (where n is the group size). The exceptions are the two schemes proposed in [8] and [9] where key-trees are used for collaborative group key agreement. In these schemes, whenever a membership change occurs, the group collectively re-computes the new key.

III. GROUP KEY MANAGEMENT IN UAV-MBN NETWORK

A. UAV-MBN Network

Homogeneous MANETs are ad-hoc wireless networks where all nodes have the same transmission capabilities while using the same frequency and channel access scheme. In such MANETs, the bandwidth available to each node rapidly decreases as the network size grows. Recent literature [10]–[12] suggest using more heterogeneous, hierarchical MANETs, namely, the UAV-MBN networks. In a UAV-MBN network, there are three node levels: UAV, MBN, and ground MANETs. Nodes at each level have different communication and computation abilities, as follows (see also Fig. 1):

1. **Ground MANET**: It includes both regular ground nodes and MBN nodes. Regular ground nodes are typically soldiers/agents equipped with communication and computation limited devices. They communicate through bandwidth-constrained short-range broadcast wireless channel.
2. **Ground mobile backbone network (MBN)**: MBN nodes are special units such as tanks and personnel carriers. They

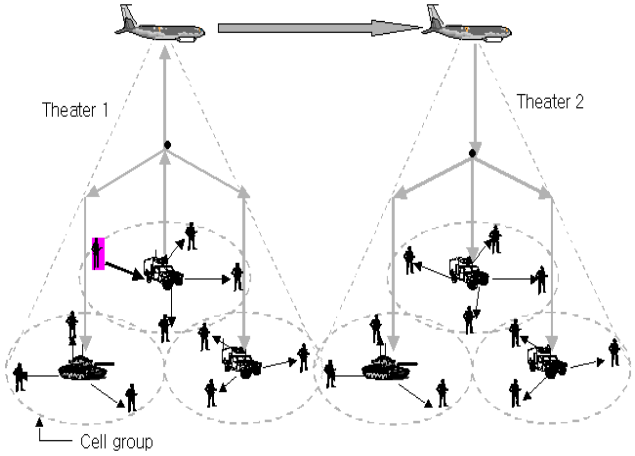

Fig. 2. Group communication model in UAV-MBN.

have more extensive facilities than regular ground nodes. In particular, they have more communication and computation power. MBN nodes can establish direct wireless links for communication amongst themselves. Regular ground nodes and MBN nodes form a super-MANET with clustered hierarchy where MBN nodes act as cluster-heads.

3. **Unmanned aerial vehicles (UAVs)**: Each UAV leads a single-area theater. With the help of phased-array antennas, a UAV can provide the shared beam to its MBN nodes to maintain line-of-sight connectivity for one area of operations below.

We also consider the following assumptions: Each node has a unique ID and some one-hop neighborhood discovery mechanisms. Communication between one-hop neighboring nodes is considered more reliable compared with multi-hop communication.

B. Group Key Management Structure and Notations

A central feature of our structure is a two-layered approach for group key management. At the lower layer, the cell is groups composed of ground MANET nodes, and at the upper layer control groups are composed of MBN nodes. Each MBN node acts as a cell group manager and controls key management for ground nodes within its cell group. As mentioned earlier, MBN nodes can establish point-to-point direct wireless links among themselves.

Nodes within the same cell group share a cell group key which is generated and distributed by the cell group manager and used for traffic encryption. MBN nodes share a control group key. Each MBN node is responsible for transferring data from within its cell group to other cell groups, if necessary. For this purpose, the transferred data is re-encrypted with control group key after being decrypted with the cell group key and delivered to other MBN nodes.

For reasons of efficiency and scalability, group key management within a cell is performed by the cell group manager (an MBN node) in a centralized fashion. At the same time, key management within the control group is done in a contributory fashion by all MBN nodes that are members of the control group.

The main reason for choosing centralized key management in cell groups is due to limited communication and computation ability of ground nodes. (It is well-known that contributory key agreement is more resource-intensive [1].) In contrast, the control group uses contributory key agreement since MBN nodes are equipped with much more powerful computation and communication facilities. Also, contributory key agreement is better suited for coping with the single-point-of-failure problem.

Table 2. Notations.

Notation	Description
M_i	a MBN node
G_i	a ground node
P_i, S_i	public/private key pair of an entiry i
T_{cur}	current time
$H()$	cryptographic one-way hash function
K_{ij}	shared secret key between entity i and j
CK_i	cell group key of MBN node M_i
$E_K()$	symmetric encryption using key K

We adapt One-way Function Tree (OFT) [3] and Tree-based Group Diffie-Hellman (TGDH) [8] schemes for cell group and control group key management, respectively. The one-way function of OFT can be based on a cryptographic hash function such as MD5 or SHA-1 requiring relatively small computation, and TGDH extends the traditional Diffie-Hellman key agreement protocol to group key agreement and it is more computationally efficient protocol among contributory group key agreement protocols [3]. In order to conserve space, we do not describe them in detail (we refer to [3] and [8]). However, we do not put restriction on each group key management to only OFT or TGDH. Group managers can adopt other appropriate schemes to the group according to their communication environments.

In most group key management schemes (including OFT and TGDH protocols) message authentication and initial secure channel establishment between a member and a group manager, when a member joins a group, is achieved by exchanging long-term credentials, i.e., public key certificates. As usual, this prompts the need for a public key infrastructure (PKI). In traditional PKI method, the public key of an entity is explicitly certified by exchanging public key certificates issued by a CA and checking the validity of certificate that may involve signature signed by the CA, and then a shared secret is established between two parties. In our scheme, however, the public key of each entity (ground or MBN node) is derived from its identifier string by the authority and the public key may be implicitly certified if both parties can successfully compute a valid common key using the identities of the other parties. For this purpose, we added key confirmation message in key agreement protocol, so that both parties are assured that they established a particular session key. Indeed, regular ground nodes are resource-constrained, the exchange and verification of public key certificates represent a heavy burden. Therefore, we use the Implicitly Certified Public Keys (ICPK) method, for the purpose of authenticated key agreement between ground and MBN nodes within a theater.

ICPK was first proposed by Günther in [13] as a variation of the ElGamal signature scheme and it is a kind of ID-based public key cryptograhpic scheme [14]. Assuming the trusted authority,

the public key of each entity is derived from its identifier string by the authority and the public key may be implicitly certified if both parties can successfully compute a valid common key after exchanging their public key. For the purpose of this work, we modify ICPK as suggested in Zheng's SDSS proposal [15]. This modification is motivated by the need to achieve the best possible computation efficiency. Indeed, we are able to reduce the computation cost as compared to that of Gunther's original method.

In the Table 2, we described notations used in our protocol.

C. Group Initialization

Each node obtains its ICPK from the trusted group authority, who is the top level group manager such as a headquarters in our scenario, according to Protocol-1 through off-line. Group authority provides public key, and secret key for node, MBN nodes as well as ground nodes. Group authority is only involved in ICPK generation and distribution to a prospective node to participate in ad-hoc network. Simultaneously, each MBN node distributes its local group key to all ground nodes to be located in its cell group. All cell group keys are distributed through pairwise secure channels. We assume that group authority is trusted and secure as a CA is, and all nodes are properly set up before constituting the ad-hoc networks.

Protocol-1: ICPK Generation (by GA)
1. chooses efficiently large prime p and q such that $q|p-1$
2. generate α such that $Z_p^* =< \alpha >$ and $ord(\alpha) = q$
3. generate random $x \in_R Z_q$
4. compute $y = \alpha^x \bmod p$
5. publish: (p, q, α, y)
6. for each node N_i (MBNs as well as ground nodes):
   (a) generate random key $k_{N_i} \in_R Z_q^*$
   (b) calculate $k_{N_i}^{-1}$, where $k_{N_i}^{-1} \cdot k_{N_i} \equiv 1 (\bmod q)$
   (c) $P_{N_i} = \alpha^{k_{N_i}} \bmod p$
   (d) $S_{N_i} = k' \cdot (H(ID_{N_i}||P_{N_i}) + x) \bmod q$
   ; H is a secure hash function and ID_{N_i} is the identity of the node
   (e) provide N_i with: $\{ S_{N_i}, P_{N_i} \}$

D. Adding/Removing Ground Node

In order to join a theater, a ground node G_u possessing ICPK takes part in a key agreement protocol by exchanging ICPKs with the MBN node M_i who is the manager of the cell group which G_u joins. The key agreement between M_i and G_u is carried out according to Protocol-2.

Protocol-2: Key Agreement(between M_i and G_u)
1. M_i and G_u choose respectively random $r_i \in Z_q^q$ and $r_u \in Z_q^q$
2. $M_i \Rightarrow G_u : ID_{M_i}, P_{M_i}, T_{cur}, h_i$
   ; where, $h_i = H(ID_{M_i}||P_{M_i}||T_{cur})$
3. $G_u \Rightarrow M_i : ID_{G_u}, P_{G_u}, (P_{M_i})^{r_u} \bmod p, h_u$
   ;where, $h_u = H(ID_{G_u}||P_{G_u}||(P_{M_i})^{r_u} \bmod p||T_{cur}||h_i)$
4. M_i computes key $K_{G_u M_i}$ ((1) and (2))
5. $M_i \Rightarrow G_u : (P_{G_u})^{r_i} \bmod p\ , E_{K_{G_u M_i}}(ID_{M_i}||ID_{M_i})$
6. G_u computes key $K_{G_u M_i}$
7. $G_u \Rightarrow M_i : E_{K_{G_u M_i}}(ID_{M_i}||ID_{M_i})$

M_i and G_u exchange their respective public keys P_{M_i} and P_{G_u}, and then compute a common secret key $K_{G_u M_i}$ according to step 4 and step 6 in Protocol-2, respectively. We note that, if P_{M_i} and ID_{M_i} of M_i were contained in a beacon message which periodically sent by an MBN node to notify the existence of the MBN node in a cell, the procedure would become more efficient. G_u and M_i each need three modular exponentiation and one multiplication to obtain a common secret key.

M_i knows P_{M_i}, S_{M_i}, P_{G_u}, and $(P_{M_i})^{r_u}$, and it can computes $\alpha^{K_{M_i} \cdot S_{M_i} \cdot r_u} = ((P_{M_i})^{r_u})^{S_{M_i}}$ and $\alpha^{K_{G_u} \cdot S_{G_u} \cdot r_i} = \alpha^{(H(P_{G_u}||ID_{G_u})+x) r_i} = (\alpha^{H(P_{G_u}||ID_{G_u})} \cdot y)^{r_i}$, then multiplies these two values to compute the key in step 4.

$$K_{G_u M_i} = \alpha^{(K_{G_u} \cdot S_{G_u} \cdot r_i + K_{M_i} S_{M_i} r_u)} \quad (1)$$

$$= (\alpha^{K_{G_u} \cdot S_{G_u} \cdot r_i}) \cdot (\alpha^{K_{M_i} \cdot S_{M_i} \cdot r_u}). \quad (2)$$

Similarly, G_u knows P_{G_u}, S_{G_u}, P_{M_i}, and $(P_{M_u})^{r_i}$, and it can compute $\alpha^{K_{G_u} S_{G_u} r_i} = ((P_{G_u})^{r_i})^{S_{G_u}}$ and $\alpha^{K_{M_i} S_{M_i} r_u} = \alpha^{(H(P_{M_i}||ID_{M_i})+x) r_u} = (\alpha^{H(P_{M_i}||ID_{M_i})} \cdot y)^{r_u}$, then multiplies these two values in step 6.

In step 2, M_i adds the description of valid time duration T_{cur} for going on protocol to message and G_u also adds the hash value h_i to the returned message as an acknowledgment so that both parties check the appropriate date from transfer to receiver.

In addition to date integrity check, when each party computes common key $K_{G_u M_i}$, G_u, and M_i exchange key confirmation messages by encrypting concatenated identity string of both parties with the key to validate the consistency of the established key.

Because the public key of each entity is derived from identifier string by group authority and the above key agreement protocol uses ID of the other party to compute common key, if the key confirmation message is properly decrypted and both parties are convinced that they established a same session key, the public key of the other party is certified simultaneously.

Once the key $K_{G_u M_i}$ is established, the MBN node M_i, cell group manager, performs cell group key updating procedure and distributes the new cell group keys to its cell group members according to OFT key update protocol for joining of new member. At this time, only M_i's cell group keys are updated but other cell group's keys are not affected.

We stress that the authorization procedure between G_u and group authority is performed once, before G_u first joins into a theater. (It is not needed whenever G_u moves from one cell to another.)

When a ground node leaves a cell group or an MBN node detects a ground node compromised, the MBN node removes the ground node from its cell group and performs cell group key updating and then securely distributes the new cell group key to remaining nodes except the compromised node according to OFT key updating protocol for eliminating the node.

E. Inter-Cell Migration

Recall that ground nodes are assumed to move freely between cells. When a ground node G_u moves from a cell controlled by M_i to another cell controlled by M_j's, it must be able to maintain its ongoing communication sessions without interruption. To do so, G_u and M_j need to quickly establish a pairwise secret key and M_j needs to provide its cell group key to G_u.

Although G_u moves another cell group, its membership in the theater remains unchanged. Therefore, explicit authentication of G_u is not required; instead, G_u is indirectly authenticated (during the agreement protocol with M_j) via the cell group key of the departed cell. The details are illustrated in Protocol-3.

We assumed that G_u migrates to the new cell managed by M_j. When G_u enters the new cell of M_j, G_u sends the current roaming time T_{cur} and previous cell controller's name, "from_M_i", to M_j in step 2, then M_j contacts M_i telling the roaming of G_u ("roam_G_u") in step 3. After M_i checks the T_{cur}, it sends the hashed value for the right previous keys, $K_{G_uM_i}$ and CK_i, shared with the roaming node G_u. At this time, M_i must check the lifetime of current session keys and T_{cur} to guarantee the key consistency with G_u.

Protocol-3: Migration of G_u from M_i to M_j.

1. G_u :
   (a) chooses random $r'_u \in Z_q^*$
   (b) computes $v_u = \alpha^{r'_u} \bmod p$
2. $G_u \Rightarrow M_j : v_u$, from_$M_i$, T_{cur}
3. $M_j \Rightarrow M_i$: roam_G_u, T_{cur}
4. M_i : computes $h_i = H(K_{G_uM_i}||(CK_j)$
5. $M_i \Rightarrow M_j : h_i$
6. M_j : computes the followings:
   (a) $h_j = H(CK_j||Tcur)$
   (b) $K_{G_uM_j} = v_u^{h_ih_j} \equiv \alpha^{r'_uh_ih_j} (\mathrm{mod} p)$
   (c) $v_j = \alpha^{h_j}$
7. $M_j \Rightarrow G_u : v_j$
8. G_u : computes key $K_{G_uM_j} = v_j^{r'_uh_u} \equiv \alpha^{r'_uh_ih_j} (\mathrm{mod} p)$
   ;where $h_u = H(K_{G_uM_i}||CK_i) = h_i$ is satisfied, if G_u was legitimate node in the previous cell of M_i

G_u is authenticated implicitly if it possesses the valid key $K_{G_uM_i}$ and CK_i used in the previous cell group to compute h_u and uses it subsequently to compute $K_{G_uM_j}$. If G_u does not know the valid $K_{G_uM_i}$ and CK_i of the just departed cell group, it cannot compute the key $K_{G_uM_j}$ and this protocol ends in failure.

IV. FEATURES

A. Security

The main security properties of group key management, forward secrecy and backward secrecy, depends on underlying group key management protocols, OFT and TGDH in our architecture. We note that their security is demonstrated in [1] and [3], respectively. However, the secrecy of the distributed cell group keys depends on the shared secret key between cell group manager and ground node for distributing group keys securely, therefore, in this section, we sketch the security of the initial key agreement between MBN node and ground node in Protocol-2.

Our flavor of ICPK is based on Zheng's SDSS proposal [17] and its security depends on the difficulty of solving the discrete logarithm problem [6] and the security of the underlying hash function. On the assumption that the difficulty of these problem, our key agreement protocol described in Protocol-2 has the following security properties.

Implicit key authentication: This property assures an entity that only the intended other entities can compute a particular key.

If an adversary wants to obtain a cell group key, he must be first able to compute the key $K_{G_uM_i}$ shared between a member ground node, G_u, and a cell group manager, MBN node M_i, in Protocol-2. However, without knowing one of private keys of both parities, S_{G_u} or S_{M_i}, a passive adversary cannot compute the shared key by eavesdropping the Protocol-2, assuming the secret key of group authority is not revealed. Furthermore, both entities are assured that the other entity actually has possession of the shared key by exchanging key confirmation message, and then public keys of both entities are instantly certified each other.

Known session key security: An adversary who has learned some previous session key must be not allowed to deduce of future session key. Our key agreement protocol has this property because each run of the protocol produces a different session key by using a session random value, therefore knowledge of past session keys does not allow deduction of future session keys.

Forward secrecy: If the long-term private key of an entity is compromised, the secrecy of previous session keys must be not affected. Because each node chooses random value in the key agreement protocol, compromising of a private key of any node, such as S_{G_u} and S_{M_i}, at some point in the future does not lead to the compromise of communications in the past, without knowing the random value. However, if an adversary can compromise any node and obtain his private key, it is possible that the adversary can start new key agreement protocol by impersonating the compromised node. To prevent this impersonating attack, membership revocation mechanism is required.

In our model, if a node is compromised and group manager detects the compromised node then the controller will be required to notify all remaining nodes of compromising the node so, the ICPK of the compromised node will not be used for key agreement any more. We leave a practical revocation scheme as our future work.

B. Scalability

In the presented key management architecture, cell-level membership changes do not affect any other cell groups. In general, each cell and control (MBN) group is free to choose its own group key management method. We adapted TGDH as the key management method for the control group and OFT for cell group key management. As alluded to before, centralized schemes (such as OFT) are appropriate for cell group key management since most regular ground nodes are equipped with limited communication and computation facilities. In a control group, the burden of group re-keying is distributed among all MBN nodes that possess superior computing and communication power. Also, decentralization avoids the single-point-of-failure problem. Supposing an MBN node member of control group is compromised by some kind of attack, other survived nodes can reconstitute control group while removing the compromised node, and TGDH provides group key management protocol for supporting this situation.

Moreover, by dividing the whole group into several cell groups, our architecture provides scalable solution. If one cell group's keys are changed as a ground node's membership changes, only the cell group where the membership process is triggered updates its cell group key while other cell groups are not affected. Furthermore, it is possible that each cell group manager selects his cell group key management scheme to whatever the manager desires regardless of what other cell groups select.

Table 3 reflects the cost of key computation and key stor-

Table 3. Storage and computation costs.

	# of stored keys	Comp. of rekeying
Regular node	$\log N_G \cdot \lvert K_{OFT}\rvert$	$O(\log N_G)_{OFT}$
MBN node	$2N_G \cdot \lvert K_{OFT}\rvert$ + $\log N_M \cdot \lvert K_{TGDH}\rvert$	$O(\log N_G)_{OFT}$ + $O(\log N_M)_{TGDH}$

N_G	the number of ground nodes in a cell
N_M	the number of manager nodes (MBNs)
$\lvert K_{OFT}\rvert$	the size of the OFT key
$\lvert K_{TGDH}\rvert$	the size of the TGDH key

age for the proposed architecture. In a cell group, we can use cryptographic hash function, such as MD5 or SHA-1 requiring relatively small computation, as the underlying one-way function in OFT. Although TGDH involves modular exponentiations for key computation, since the number of MBN nodes is relatively small and they are equipped with more CPU power, the cost of TGDH is likely to be relatively low. (Keeping in mind that TGDH is computationally efficient scheme among group key agreement protocols and incurs, at worst, a $O(\log n)$ cost.) Specifically, cell group managers are fewer in number and more stable than that of regular ground nodes, cell group key management processing is both faster and less frequent. However, since the shared cell group key can be vulnerable if it changes very infrequently, a security policy should impose additional refreshing operations, triggered, for example, by maximum elapsed time between successive key changes or maximum volume of data exchanged.

V. CONCLUSIONS

In this paper, we proposed a group key management architecture for UAV-MBN mobile ad-hoc network. In this setting, group key management must be especially efficient and scalable since the constant mobility of ground nodes increases the rate of change for the topology and the membership of the group. In our architecture, a theater is divided into a control group and cell groups. The impact of a membership change is contained to a single cell group and does not propagate outside, to other cell groups.

In order to minimize computation and communication costs for regular resource limited ground nodes, the centralized OFT group key management scheme is employed for cell groups, whereas, to avoid the single-point-of-failure problem, we adopted the TGDH group key agreement scheme for control groups. In the same vein, we proposed using ICPK for the purpose of authenticated key agreement between ground and MBN nodes within a theater. This was done to avoid managing (i.e., exchange and verification) public key certificates.

In recent years, ID-based cryptography (IBC) has been an area of very active research since a practical ID-based encryption scheme was proposed by Boneh and Franklin [4]. When we wrote this article, we did not consider the IBC for the purpose of implicit certification. We are considering the employment of IBC for secure ad-hoc network for future work.

ACKNOWLEDGMENT

This work was supported by the Korean Research Foundation Grant (KRF-2001-013-E00064). The authors wish to thank anonymous reviewers for their helpful comments and invaluable suggestions for revision of this paper.

REFERENCES

[1] Y. Amir *et al.*, "Exploring robustness in group key agreement," in *Proc. ICDCS'2001*, Apr. 2001, pp. 399–408.
[2] Y. Amir *et al.*, "On the performance of group key agreement protocols," in *Proc. IEEE ICDCS'2002*, July 2002.
[3] D. Balenson, D. McGrew, and A. Sherman, "Key management for large dynamic groups: One-way function trees and amortized initialization," in *IETF Internet Draft: Draft-balensongroupkeymgmt-oft-00.txt*, Feb. 1999.
[4] D. Boneh and M. Franklin, "Identity based encryption from the weil pairing," in *Proc. Advances in Cryptology Crypto2001*, Lecture Notes In Computer Science 2139, Springer-Verlag, 2001, pp. 213–229.
[5] H. Luo *et al.*, "Self-securing ad hoc wireless networks," in *Proc. IEEE ISCC'2002*, 2002, pp. 567–574.
[6] M. Steiner, G. Tsudik, and M. Waidner, "Cliques: A new approach to group key agreement," in *Proc. IEEE ICDCS'98*, 1998, pp. 380–387.
[7] C. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," in *Proc. ACM SIGCOMM'98*, Sept. 1998, pp. 68–79.
[8] Y. Kim, A. Perrig, and G. Tsudik, "Simple and fault-tolerant key agreement for dynamic collaborative groups," in *Proc. ACM Conference on Computer and Communications Security 2000*, Nov. 2000, pp. 235–244.
[9] Y. Kim, A. Perrig, and G. Tsudik, "A communication-efficient group key agreement," in *Proc. IFIP-SEC'2001*, June 2001, pp. 229–244.
[10] D. Gu *et al.*, "C-ICAMA: A centralized intelligent channel assigned multiple access for multi-layer ad-hoc wireless networks with UAVs," in *Proc. IEEE WCNC'2000*, 2000, pp. 879–884.
[11] D. Gu *et al.*, "Hierarchical routing for multi-layer ad-hoc wireless networks with UAVs," in *Proc. IEEE MILCOM'2000*, 2000, pp. 310–314.
[12] D. Gu *et al.*, "UAV-aided intelligent routing for ad-hoc wireless network in single-area theater," in *Proc. IEEE WCNC'2000*, 2000, pp. 1220–1225.
[13] C. Gunther, "An identity-based key exchange protocol," in *Proc. Advances in Cryptology EUROCRYPT'89*, Lecture Notes in Computer Science 434, 1989, pp. 29–37.
[14] A. Shamir. "Identity-based cryptosystems and signature schemes," in *Proc. CRYPTO'84*, Lecture Notes in Computer Science 196, Springer-Verlag, 1984, pp. 47–53.
[15] Y. Zheng, "Shortened digital signatures, signcryption and compact and unforgeable key agreement schemes," Submission to *IEEE* P1363a: Standard Specifications for Public-Key Cryptography, Aug. 1998.
[16] A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
[17] M. Steiner, G. Tsudik, and M.Waidner, "Key agreement in dynamic peer groups," *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, Issue 8, pp. 769–780, Aug. 2000.

Kyung Hyune Rhee was received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Daejon Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Daejon Korea from 1985 to 1993. He also worked as a visiting scholar in the University of Adelaide, the University of Tokyo, and the University of California, Irvine. He has served as a Chairman of Division of Information and Communication Technology, Colombo Plan Staff College for Technician Education in Manila, the Philippines. He is currently a professor in the Division of Electronic, Computer and Telecommunication Engineering of Pukyong National University, Busan Korea. His research interests center on key management and its applications, mobile communication security and security evaluation of cryptographic algorithms.

Young Ho Park was received his B.S. and M.S. degrees in Department of Computer Science from Pukyong National University, Busan Korea in 2000 and 2002, respectively. He is currently a Ph.D. candidate in graduate school of Information Security, Pukyong National University. His interests are related with information security and network security; ad-hoc network security, secure peer-to-peer network, key management, and identity-based cryptosystem.

Gene Tsudik was received the Ph.D. in computer science from the University of Southern California in 1991; his dissertation focused on access control in internetworks. He is a professor in the Computer Science Department at the University of California, Irvine. Before joining the University of California, Irvine in 2000, he was a project leader at IBM Research, Zurich Laboratory (1991–1996) and the USC Information Science Institute (1996–2000).