

## A Group Key Management Architecture for Mobile Ad-hoc Wireless Networks\*

KYUNG-HYUNE RHEE, YOUNG-HO PARK AND GENE TSUDIK<sup>+</sup>

*Division of Electronic, Computer and Telecommunication Engineering*

*Pukyong National University*

*Busan, 608-737, Korea*

<sup>+</sup>*Information and Computer Science*

*University of California at Irvine*

*Irvine, CA 92697, U.S.A.*

In recent years, mobile ad-hoc networks have received a great deal of attention in both academia and industry because they provide anytime-anywhere networking services. As wireless networks are rapidly deployed in the future, secure 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 the Implicitly Certified Public Keys method, which reduces the overhead of the certificate validation checking process and improves computational efficiency. The architecture uses a two-layered key management approach, where a 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.

**Keywords:** key exchange, key agreement, group key management, implicit certificate, secure ad-hoc network

### 1. INTRODUCTION

Mobile ad-hoc networks (MANETs) offer convenient infrastructure-free communication over a shared wireless medium. MANETs are also regarded as ideal technology for creating instant communication networks 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 largescale. 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 the commercial and military sectors in heightening the security concerns since the very nature of these networks makes them more vulnerable (than wired networks) to certain attacks, such as passive eavesdropping and denial of service.

---

Received April 7, 2003; revised July 25, 2003 & January 9, 2004; accepted April 1, 2004.

Communicated by Shiuhyng Shieh.

\* The early version of this paper was presented at the 3<sup>rd</sup> International Workshop on Information Security Application (WISA 2002), Jeju, Korea, Aug. 28-30, 2002.

Secure group communication requires scalable and efficient group membership management with appropriate access control measures to protect data and to 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<sup>1</sup>.

Several proposals for group key management have been made recently in the literature. They range from key distribution schemes for large-scale single-sender multicast [2, 8] to contributory key agreement schemes for small any-to-any peer groups [9, 18]. 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 the cell members. In addition, each MBN node functions as a peer member of its control group.

The whole group is clustered into several cell and control groups in our model, and each cell group autonomously manages its cell group key so that each cell group key is independent of the other cell group keys. 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 in 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 needs to employ decentralized key management since MBN nodes have significant computational and communication power. Furthermore, decentralization helps us avoid 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 2 discusses security threats and summarizes previous work. Section 3 presents the proposed architecture including the actual group key management protocols. Section 4 provides an analysis and discusses the features of the proposed architecture. The paper is concluded in section 5.

## 2. SECURITY THREATS AND RELATED WORK

We will begin by discussing the security threats faced by MANETs and then address the necessary requirements for security services. In the process, we will also review the relevant previous works.

---

<sup>1</sup> Informally, backward secrecy is attained if it is computationally difficult for a member to discover the group keys(s) that were used before it joined the group. On the other hand, forward secrecy is attained if it is computationally difficult for a member to discover group key(s) that were used after it left the group.

## 2.1 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 not only to consider malicious attacks from outside the network, but also to 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 nature of MANETs (i.e., frequent changes in both topology and membership) make key management particularly challenging, even more than that in other wired and wireless networks. It is not surprising, therefore, that many traditional key management approaches are not well-suited to this environment. In popular network authentication architectures, two entities authenticate each other's globally trusted certificates authority (CA). While this model works well in wired networks, it fails in large ad-hoc wireless environments for several reasons [11].

## 2.2 Related Work on 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 critical goal is to provide the so-called *key independence* property [17], where knowledge of all (but one) group keys cannot be used to efficiently derive the one "missing" group key.

In addition, group key management must be scalable, i.e., its protocols should be efficient in terms of resource usage and should be able to minimize the effects of a membership change. Much research on group key management has been conducted in the last decade. Prior work can be roughly partitioned into centralized approaches, in which a key center is responsible for creating and distributing keys, and collaborative key agreement approaches, in which all members contribute to group key agreement with no key center. (The reader may refer to Table 1.)

Many key-tree schemes, such as those described in [2, 8, 9, 13], have been proposed for the purpose of minimizing the 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.

**Table 1. Comparing group key management types.**

	Centralized		Collaborative
Key management type	Key distribution by key center		Key agreement by member's contribution
Computation costs	Key Center	Member	Large(similar complexity)
	large	small	
Features	Single point of failure of key center		Multiple communication rounds
Examples	Key graph [13], OFT [2]		GDH [18], TGDH [9], STR [8]

Table 2 summarizes the communication and computation costs [1] of the peer group key agreement protocols, GDH [18], STR [8] and TGDH [9], and OFT [2].

**Table 2. Performance analysis of group key management protocols.**

OFT		Member	Controller
	Number of keys	$2\log_2 n$	$2n - 1$
Computation costs	$O(\log n)$	$O(\log n)$	
Msgs sent on join/leave	$\log_2 n$		

		Communication		Computation
		Rounds	Message	Exponentiations
GDH	Join	4	$n + 3$	$n + 3$
	Leave	1	1	$n - 1$
	Merge	$m + 3$	$n + 2m + 1$	$n + 2m + 1$
	Partition	1	1	$n - p$
STR	Join	2	3	4
	Leave	1	1	$(3n/2) + 2$
	Merge	2	3	$3m + 1$
	Partition	1	1	$(3n/2) + 2$
TGDH	Join	2	3	$3h/2$
	Leave	1	1	$3h/2$
	Merge	2	3	$3h/2$
	Partition	$h$	$2h$	$3h$

$n$  – the number of members in the group  
 $h$  – the height of the key tree  
 $m$  – the number of merging groups  
 $p$  – the number of members partitioned from a group of  $n$  members

### 3. GROUP KEY MANAGEMENT IN UAV-MBN NETWORKS

In this section, we will present our key management model.

### 3.1 UAV-MBN Networks

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 studies, such as [3-5], suggest the use of more heterogeneous, hierarchical MANETs, namely, the UAV-MBN networks. The authors [10] proposed authentication services via certificates for security in UAV-MBN networks, including certificate issuing, renewal, and revocation, and storage/retrieval of certificates and CRLs. While their security services differ from group key management services, we use the same networking model and some assumptions for our group key management structure.

In an UAV-MBN network, there are three node levels: UAV, MBN, and ground MANETs. Nodes at each level have different communication and computation abilities, described as follows (see also Fig. 1):

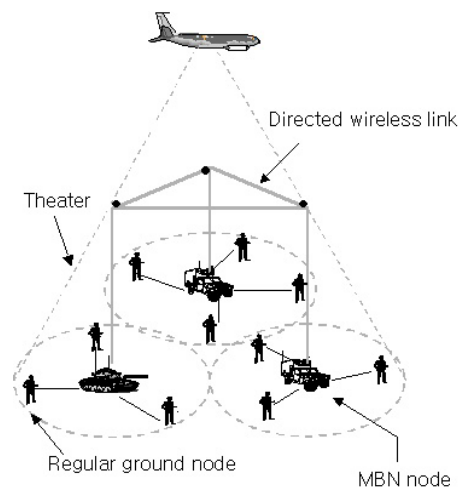


Fig. 1. Hierarchical MANET with MBNs and UAV.

#### (1) Ground MANET

This 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 channels.

#### (2) Ground mobile backbone network (MBN)

MBN nodes are special units, such as tanks and personnel carriers. They have more extensive facilities than regular ground nodes. In particular, they have more communication and computational power. MBN nodes can establish direct wireless links for communication amongst themselves. Regular ground nodes and MBN nodes form a super-MANET with a 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 a shared beam to its MBN nodes to maintain line-of-sight connectivity for one area of operations below.

We also make 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.

### 3.2 Group Key Management Architecture

The main feature of our structure is a layered approach to group key management. In the lower layer are the cell groups composed of ground MANET nodes within the same wireless broadcast range, and in the upper layer are control groups 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 have sufficiently power and can establish point-to-point direct wireless links among themselves.

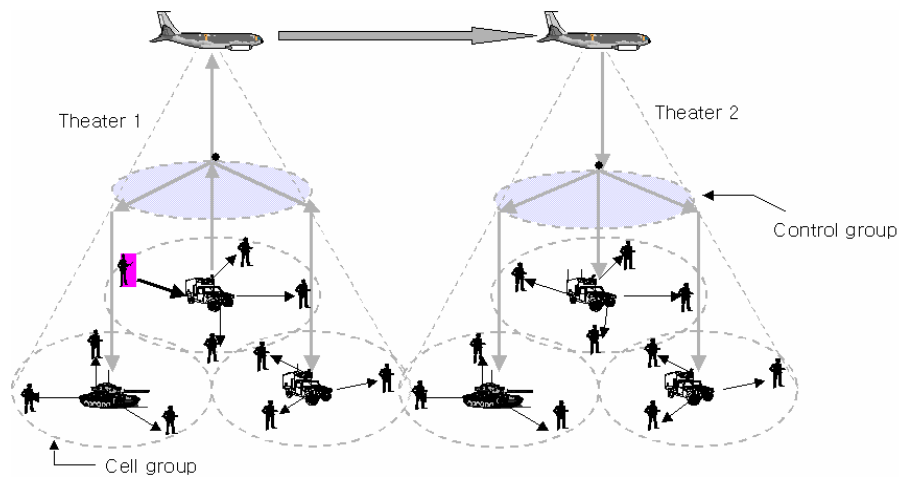


Fig. 2. Group communication and key management model.

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 relaying data from within its cell group to other cell groups, if necessary. For this purpose, the relayed 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 the MBN nodes that are members of the control group.

The main reason for choosing centralized key management in cell groups is limited communication and computation ability of the ground nodes. (It is a well-known fact that contributory key agreement is more resource-intensive [15].) In contrast, the control group uses contributory key agreement since MBN nodes are equipped with much more powerful computational and communication abilities. Also, contributory key agreement is better suited to coping with the single-point-of-failure problem.

We adopt the One-way Function Tree (OFT) [2] and Tree-based Group Diffie-Hellman (TGDH) [9] schemes for cell and control group key management, respectively. In order to conserve space, we will not describe them here in detail; we refer the reader to [2] and [9]. However, we do not restrict key management in each group to only OFT or TGDH. The group key manager can choose an appropriate group key management scheme that he wants for his group according to his communication and computational environment without regard to what group key management schemes are being used in other groups.

In our model, we divide the whole group into several cell groups and a control group, and each cell group is managed by its cell group controller independently of the other cell groups. That is, the dynamics of one cell group do not affect the other cell groups, for the reason that each controller may use an appropriate group key management scheme that the group controller wants for its cell group according to its communication environment without regard to what group key management schemes are used in other cell groups.

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. Usually, this requires a public key infrastructure (PKI) for checking the validity of the certificates. However, since regular ground nodes are resource-constrained, the exchange and verification of public key certificates represents a heavy burden. Therefore, we use the *Implicitly Certified Public Keys* (ICPK) method by modifying Günther's key exchange scheme [6], for the purpose of authenticating key agreement between ground and MBN nodes within a theater. The established key is assigned to the appropriate leaf node on the OFT key tree adopted in the cell group key management scheme.

ICPK was first proposed by Günther as a variation of the ElGamal signature scheme. For our purpose, we modify ICPK as suggested in Zheng's SDSS proposal [14]. This modification is motivated by the need to achieve the best possible computational efficiency. Indeed, we are able to reduce the computational cost as compared to that of Günther's original method.

### 3.3 Group Initialization

We assume that all nodes are properly set up before the ad-hoc network is formed. Each node obtains its ICPK from a top level group controller, such as a headquarters in our scenario, according to Procedure-1 in an off-line manner (see below). The group controller provides public key  $P_i$  and secret key  $S_i$  for each ground and MBN node. Each MBN node distributes its local group key to all the ground nodes in its cell group. All the cell group keys are distributed through pairwise secure channels.

Procedure-1 : ICPK Generation (executed by the group controller)

1. Generate a random large prime  $p$  such that  $p - 1 = kq$  for some large prime  $q$ .
2. Generate  $\alpha$  such that  $Z_p^* = \langle \alpha \rangle$  and  $\text{ord}(\alpha) = q$ .
3. Generate random  $x \in {}_R Z_q$ .
4. Compute  $y = \alpha^x \bmod p$ .
5. Publish:  $(p, q, \alpha, y)$ .
6. For each node  $M_i$  (MBN node as well as ground node):
  - (a) generate random key  $K_i \in {}_R Z_q^*$ ;
  - (b) calculate  $K_i^{-1}$ , where  $K_i^{-1} \cdot K_i \equiv 1 \pmod{q}$ ;
  - (c) public key  $P_i = \alpha^{K_i} \bmod p$ ;
  - (d) private key  $S_i = K_i^{-1} \cdot (H(P_i \parallel ID_i) + x) \bmod q$ ,  
where  $H$  is a secure hash function and  $ID_i$  is the identity of the node;
  - (e) provide  $M_i$  with:  $\{S_i, P_i\}$ .

### 3.4 Adding/Removing a 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 an MBN node  $M_i$ , who is the manager of the cell group which  $G_u$  wants to join. Key agreement between  $M_i$  and  $G_u$  is carried out according to Protocol-1 shown below.

Protocol-1 : ICPK exchange (executed by  $M_i$  and  $G_u$ )

1.  $M_i$  and  $G_u$  choose session random numbers  $r_i \in {}_R Z_p^*$  and  $r_u \in {}_R Z_p^*$ , respectively.
2.  $M_i \Rightarrow G_u : ID_{M_i}, P_{M_i}, T_{cur}, h_i = H(ID_{M_i} \parallel P_{M_i} \parallel T_{cur})$ .
3.  $G_u \Rightarrow M_i : ID_{G_u}, P_{G_u}, (P_{M_i})^{r_u}, T_{cur}, h_u = H(ID_{G_u} \parallel P_{G_u} \parallel (P_{M_i})^{r_u} \parallel T_{cur} \parallel h_i)$ .
4.  $M_i$  : calculate key  $K_{ui}$ ;  
 $M_i \Rightarrow G_u : (P_{G_u})^{r_i} \bmod p, E_{K_{ui}}(ID_{M_i} \parallel ID_{G_u})$ .
5.  $G_u$  : calculate key  $K_{ui}$ ;  
 $G_u \Rightarrow M_i : E_{K_{ui}}(ID_{M_i} \parallel ID_{G_u})$ .

$G_u$  and  $M_i$  exchange their respective public keys  $P_{G_u}$  and  $P_{M_i}$ , and then compute a common secret key  $K_{ui}$  according to Procedure-2. It should be noted that public key  $P_{M_i}$  and identification string  $ID_{M_i}$  of  $M_i$  may be contained in a beacon message which is periodically sent by an MBN node to make known the existence of the MBN node in a cell. In step 2,  $M_i$  adds a description of the valid time duration  $T_{cur}$  for going on protocol to the message, and  $G_u$  also adds the hashed  $T_{cur}$  and  $h_i$  to the returned message as an acknowledgement so that both parties can check the appropriate date of transfer to the receiver. In addition to the date integrity check, when each party computes common key  $K_{ui}$ ,  $G_u$  and  $M_i$  exchange a key confirmation message by encrypting concatenated identity strings of both parties using the key in order to validate the consistency of the established key.

Procedure-2 : Pairwise key,  $K_{ui}$ , calculation

$$- K_{ui} = \alpha^{(K_{G_u} \cdot S_{G_u} \cdot r_i + K_{M_i} \cdot S_i \cdot r_u)} \equiv \alpha^{K_{G_u} \cdot S_{G_u} \cdot r_i} \cdot \alpha^{K_{M_i} \cdot S_{M_i} \cdot r_u} \pmod{p}$$

From the  $G_u$ 's viewpoint

$$- \alpha^{K_u \cdot S_u \cdot r_i} = ((P_u)^{r_i})^{S_u} \pmod{p}$$

$$- \alpha^{K_{M_i} \cdot S_{M_i} \cdot r_u} = \alpha^{(H(P_{M_i} \| ID_{M_i}) + x) \cdot r_u} \equiv (\alpha^{H(P_{M_i} \| ID_{M_i})} \cdot y)^{r_u} \pmod{p}$$

Once key  $K_{ui}$  is established, the MBN node  $M_i$ , the cell group controller, carries out the cell group key updating procedure and distributes the new cell group keys to its cell group members according to the OFT key update protocol for adding a new member. At this time, only  $M_i$ 's cell group keys are updated, and other cell groups' keys are not affected.

We should stress that the authorization procedure between  $G_u$  from group controller is performed once before  $G_u$  first joins a theater. (The procedure is not needed whenever  $G_u$  moves from one cell to another as long as  $G_u$  is not compromised and is not being attacked in the theater).

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

### 3.5 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  $M_j$ , it must be able to maintain 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 (in the key agreement protocol with  $M_j$ ) by means of the cell group key of the departing cell. The details are provided in Protocol-2.

We assume that  $G_u$  migrates to the new cell managed by  $M_j$ . When  $G_u$  enters the cell of  $M_j$ ,  $G_u$  sends the current time  $T_{roam}$  and previous cell controller name "from  $M_i$ " to  $M_j$  in step 2; then,  $M_j$  contacts  $M_i$  to tell it that  $G_u$  is roaming. After  $M_i$  checks  $T_{roam}$ , it sends a hashing value for the appropriate previous keys,  $K_{ui}$  and  $CK_i$ , which share roaming node  $G_u$ . At this time,  $M_i$  must check the lifetimes of the current session keys and  $T_{roam}$  to guarantee key consistency with  $G_u$ .

To perform key agreement with  $G_u$ ,  $M_j$  computes  $h_j$  by using its cell group key  $CK_j$ .  $G_u$  is authenticated implicitly if it possesses the valid key  $K_{ui}$  and  $CK_i$  used in the previous cell group to compute  $h_u$ , and uses it subsequently to compute  $K_{ju}$ . If  $G_u$  does not know the valid  $K_{ui}$  and  $CK_i$  of the departing cell group, it can not compute the key  $K_{ju}$ , and this protocol ends in failure.

<p>Protocol-2 : roaming of <math>G_u</math> from <math>M_i</math> to <math>M_j</math></p> <ol style="list-style-type: none"> <li>1. <math>G_u</math>: chooses random <math>r'_i \in_R Z_p^*</math>;  <math display="block">v_u = \alpha^{r'_i} \pmod p.</math></li> <li>2. <math>G_u \Rightarrow M_j : v_u, T_{roam}, from\_M_i.</math></li> <li>3. <math>M_j \Rightarrow M_i : roam\_G_u, T_{roam}.</math></li> <li>4. <math>M_i \Rightarrow M_j : h_i = H(K_{ui} \parallel CK_i).</math></li> <li>5. <math>M_j : h_j = H(CK_j), v_j = \alpha^{h_j} \pmod p;</math>  compute key <math>K_{ju} = v_u^{h_i \cdot h_j} \equiv \alpha^{r'_i \cdot h_i \cdot h_j} \pmod p.</math></li> <li>6. <math>M_j \Rightarrow G_u : v_j.</math></li> <li>7. <math>G_u</math>: compute key <math>K_{ju} = v_j^{r'_i \cdot h_u} \equiv \alpha^{r'_i \cdot h_i \cdot h_j} \pmod p,</math>  where <math>h_u = H(K_{ui} \parallel CK_i) = h_i.</math></li> </ol>
---

#### 4. FEATURES

In our key management architecture, cell-level membership changes in one cell group 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 propose TGDH as the key management method for the control group and OFT for cell group key management. As alluded to previously, 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 the MBN nodes which possess superior computing and communication power. Also, through decentralization, we avoid the single-point-of-failure problem. Moreover, by dividing the whole group into several cell groups, our architecture provides a scalable solution. When one cell group's keys are changed as a ground node's membership changes, other cell groups are not affected.

Table 3 shows the costs of key computation and key storage for the proposed architecture. In a cell group, we can use MD5 or SHA-1 as the underlying hash 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. (Keep in mind that TGDH is a computationally efficient scheme among group key agreement protocols [1] and incurs, at worst, a  $O(\log n)$  cost.) Specifically, cell group controllers are fewer in number and more stable than regular ground nodes, so cell group key agreement is both faster and is performed less frequent. However, since the shared cell group key may become vulnerable if it changes very infrequently, a security policy should impose additional refreshing operations that are triggered, for example, by a maximum elapsed time between successive key changes or a maximum volume of data exchanged.

The main security properties of group key agreement are forward secrecy and backward secrecy. In our architecture, the security of a group key depends on both OFT and TGDH. We should note that their security in OFT and TGDH was demonstrated in [2] and [9], respectively.

**Table 3. Storage and computational costs.**

	Number of stored keys	Re-keying computation
Regular ground node	$\log n_{G_i} \cdot  K_{OFT} $	$O(\log n_{G_i})_{OFT}$
Cell group controller (MBN node)	$2n_{G_i} \cdot  K_{OFT}  + \log n_c \cdot  K_{TGDH} $	$O(\log n_{G_i})_{OFT} + O(\log n_c)_{TGDH}$
$n_G$ : the number of ground nodes in a cell group $G$ . $n_c$ : the number cell group controllers i.e., MBN nodes.		

Our flavor of ICPK is based on Zheng's SDSS proposal [14], and its security depends on the difficulty of solving the discrete logarithm problem [12] and the security of the underlying hash function. If an adversary wants to obtain an OFT group key, he must first compute  $K_{ui}$  shared between a member ground node,  $G_u$ , and a cell group manager, MBN node  $M_i$ . In order to obtain this key, the adversary must know one of the secret values of the two parties,  $S_j = K_j^{-1} \cdot (H(P_j || ID_j) + x) \bmod q$ , and a random value  $r_j$  of another party. However, the secret value  $S_j$  is provided securely by the group controller before joining the ad-hoc network; therefore, the adversary is not able to compute  $S_j$  without knowing the UAV's secret value  $x$ . Moreover, it is computationally hard to compute  $r_u$  by eavesdropping on exchanged messages, which is, in turn, based on the difficulty of the Diffie-Hellman and the discrete logarithm problems.

As for using ICPKs for authentication and key agreement, we intend to allow the possible replacement of a PKI representative of an explicit certificate with an implicit certificate by using an identification value during the public key generation process for the sake of secure mobile ad-hoc networking. Authenticity is verified at the time when the key is used for encryption or key exchange. In our model, if a node is compromised and the group controller detects the compromised node then the controller will notify all the remaining nodes of the compromised the node, and the ICPK of the compromised node will no longer be used for key agreement.

## 5. CONCLUSIONS

In this paper, we have proposed a group key management architecture for UAV-MBN mobile ad-hoc network. In this setting, group key management needs to be especially efficient and scalable since the constant mobility of ground nodes increases the rate of change of 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 it to other cell groups.

In order to minimize the computational and communication costs for normal resource limited ground nodes, the centralized OFT group key management scheme is used for cell groups, whereas to avoid the single-point-of-failure problem, we adopt the TGDH group key agreement scheme for control groups. In addition, we have proposed using ICPK for the purpose of achieving authenticated key agreement between ground and MBN nodes within a theater. This is done to avoid the need to manage (i.e., exchange and verification) public key certificates.

## ACKNOWLEDGEMENT

This work was supported by the Korean Research Foundation under Grant (KRF-2001-013-E00064). Moreover, the authors would like to express their gratitude to the anonymous reviewers of their valuable comments.

## REFERENCES

1. Y. Amir, Y. Kim, C. Nita-Rotaru, and G. Tsudik, "On the performance of group key agreement protocols," in *Proceedings of the 22nd IEEE International Conference on Distributed Computing Systems (ICDCS 2002)*, 2002, pp. 463-464.
2. D. Balenson, D. McGrew, and A. Sherman, "Key management for large dynamic groups: one-way function trees and amortized initialization," IETF Internet Draft: draft-balensongroupkeymgmt-oft-00.txt, 1999.
3. D. Gu, H. Ly, X. Hong, M. Gerla, G. Pei, and Y. Lee, "C-ICAMA: a centralized intelligent channel assigned multiple access for multi-layer ad-hoc wireless networks with UAVs," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2000)*, 2000, pp. 879-884.
4. D. Gu, G. Pei, H. Ly, M. Gerla, and X. Hong, "Hierarchical routing for multi-layer ad-hoc wireless networks with UAVs," in *Proceedings of IEEE Milcom 2000*, 2000, pp. 310-314.
5. D. Gu, G. Pei, H. Ly, M. Gerla, B. Zhang, and X. Hong, "UAV-aided intelligent routing for ad-hoc wireless network in single-area theater," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC 2000)*, 2000, pp. 1220-1225.
6. C. Günther, "An identity-based key exchange protocol," *Advances in Cryptology, EUROCRYPT '89*, LNCS 434, 1989, pp. 29-37.
7. J. Kong, P. Zerfos, H. Luo, S. Lu, and L. Zhang, "Providing robust and ubiquitous security support for mobile ad-hoc networks," in *Proceedings of the 9th International Conference on Network Protocols (ICNP '01)*, 2001, pp. 251-260.
8. Y. Kim, A. Perrig, and G. Tsudik, "Communication-efficient group key agreement," in *Proceedings of IFIP-SEC 2001*, 2001, pp. 229-244.
9. Y. Kim, A. Perrig, and G. Tsudik, "Simple and fault-tolerant key agreement for dynamic collaborative groups," in *Proceedings of 7th ACM Conference on Computer and Communications Security*, 2000, pp. 235-244.
10. J. Kong, H. Luo, K. Xu, D. Gu, M. Gerla, and S. Lu, "Adaptive security for multi-layer ad-hoc networks" *Wireless Communications and Mobile Computing, Special Issue on Mobile Ad Hoc Networking*, Vol. 2, 2002, pp. 533-547.
11. H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang, "Self-securing ad hoc wireless networks," in *Proceedings of IEEE 7th Symposium on Computers and Communications (ISCC '02)*, 2002, pp. 567-574.
12. A. Menezes, P. van Oorschot, and S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
13. C. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," *ACM SIGCOMM '98*, 1998, pp. 68-79.

14. Y. Zheng, "Shortened digital signatures, signcryption and compact and unforgeable key agreement schemes," *Submission to IEEE P1363a: Standard Specifications for Public-Key Cryptography*, 1998.
15. Y. Amir, Y. Kim, C. Nita-Rotaru, J. Schultz, J. Stanton, and G. Tsudik, "Exploring robustness in group key agreement," in *Proceedings of the 21st IEEE International Conference on Distributed Computing Systems (ICDCS)*, 2001, pp. 399-408.
16. M. Bellare, R. Canetti, and H. Krawczyk, "Keying hash functions for message authentication," *Advances in Cryptology – CRYPTO '96*, LNCS 1109, Springer-Verlag, 1996, pp. 1-15.
17. M. Steiner, G. Tsudik, and M. Waidner, "CLIQUES: a new approach to group key agreement," in *Proceedings of International Conference on Distributed Computing Systems*, 1998, pp. 380-387.
18. M. Steiner, G. Tsudik, and M. Waidner, "Key agreement in dynamic peer groups," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 11, 2000, pp. 769-780.



**Kyung Hyune Rhee** (Division of Electronics, Computer and Telecommunications Engineering, Pukyong National University). He received his M.S. and Ph.D. degrees from Korea Advanced Institute of Science and Technology (KAIST), Daejeon Korea in 1985 and 1992, respectively. He worked as a senior researcher in Electronic and Telecommunications Research Institute (ETRI), Daejeon 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** (Division of Electronics, Computer and Telecommunications Engineering, Pukyong National University). He 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 ID-based cryptosystem.



**Gene Tsudik** (Information and Computer Science, University of California, Irvine, U.S.A.) He received the Ph.D. in Computer Science from the University of Southern California (USC) 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). Over the years, his research interests included: internetwork routing, firewalls, authentication, mobile network security, secure e-commerce, anonymity, secure group communication, digital signatures, key management, ad hoc network routing and, more recently, database privacy, and secure storage. Some of his notable research contributions include: interdomain policy routing (IDPR), IBM Network Security Program (KryptoKnight), IBM Internet Keyed Payment (iKP) protocols, Peer Group Key Management (CLIQUES), and Mediated Cryptographic Services (SUCSES). He is currently serving as Associate Dean of research and graduate studies in the School of Information and Computer Science at the University of California, Irvine.