IJCSI International Journal of Computer Science Issues, Vol. 8, Issue 2, March 2011
ISSN (Online): 1694-0814
www.IJCSI.org

Certificateless Efficient Group Key Management Scheme in Mobile Adhoc Networks

Sanjeev Kumar Rana[1], Manpreet Singh[2]

[1]Assoc. Professor, Computer Engineering Department, M. M. Engineering College, M. M. University, Ambala, India-133207.

[2]Professor, Computer Engineering Department, M. M. Engineering College, M. M. University, Ambala, India-133207.

Abstract

Securing mobile ad hoc networks (MANETs) is a crucial task for their good deployments. One fundamental aspect of providing confidentiality and authentication is key management. While Public Key Infrastructure (PKI) based solutions has provided these properties historically, MANETs are resource constrained and benefit from symmetric key encryption. In this paper, we proposed a certificateless efficient group key management scheme (CE-GKMS) in MANETs for group confidentiality which uses identity based cryptography for secure multicast group communication. The scheme does not need PKI in which mobile nodes needs large storage to carry certificates and to perform public key cryptography based large computation. The scheme introduced a new idea of hiding the public keys and making them visible only to the trusted nodes which not only make it difficult for cryptanalyst to crack the private information but also permit to keep small value of encryption and decryption component causes asymmetric cryptography operation faster. For scalability and dynamic reconfigurability, we divide the network into groups. Leaders in these groups securely communicate with each other to agree on group key in response to membership change and member mobility-induced events. The performance results prove the effectiveness of our proposed key management scheme CE-GKMS.

Keywords: *Network Security, MANETs, Group Confidentiality, Key Management.*

1. Introduction

MANET is a network consisting of collection of nodes capable of communicating with one another without the assistance of network infrastructure. In a MANET, each mobile node acts as a router. The main advantage of MANET is that it can operate in isolation or in coordination with wired infrastructure. If a message is sent out through a general tunnel without encryption, it may suffer malicious attacks [1], [2], [3].

Each node, which acts like a mobile router, has full control over the data that pass through it. Some of these are malicious nodes, which enter the network during establishment phase while others may originate indigenously by compromising an existing benevolent node. These malicious nodes can carry out both passive and active attacks against the network. In passive attacks, a malicious node only eavesdrops upon packet contents without disrupting the network operation, while active attacks can fabricate, modify or drop a packets [4] [5]. Because of these attacks, security is necessary to guard against attacks.

Cryptography is an important and powerful tool for security services, namely authentication, confidentiality, integrity and non-repudiation. Key management is a basic part of any secure communication. Key management deals with key generation, storage, distribution, updating, and revocation and certificate services, in accordance with security policies. Absence of secure key management makes a network vulnerable to attack [6]. Key management schemes usually focus on improving security and optimizing the key storage [7], [8]. The limited resources and mobility of nodes are bottleneck of MANET security. An effective key management system can solve this problem. In mobile ad hoc network, a group can hasten message delivery and prevent bandwidth waste effectively. Group confidentiality is one of the issues in group key management used in assuring secure multicast group communication where limited broadcast is used. Group confidentiality requires that only valid group users could decrypt the multicast data even if the data is broadcast to the entire network.

In this paper, we proposed a certificateless efficient group key management scheme (CE-GKMS) which does not require online certification authority for secure multicast group communication. Rest of the paper is organized as follows: Section II summarizes some of the previous works that have been proposed for key management in MANETs and the advantages as well disadvantages of such works. Section III discuss about the proposed scheme and working of proposed new scheme is discussed for secure group communication in MANET in section IV.

The effectiveness of proposed scheme is described in Section V. Then this paper is concluded in Section VI.

2. Related Work

Many of the security solution have been proposed for MANET. Some of the research papers focus on either secure routing transmission or key management in MANET are described below:

A distribution of symmetric key generation system (SKGS) based on key pre-distribution scheme is given in [9]. In SKGS, a central server is responsible for the creation and distribution of nodal key chains. Drawback of this scheme is nodes can derive future key from the key chain which they receive from the main server and decrypt future traffic, hence lack of backward secrecy. Another problem in the SKGS scheme is single point of failure of the central server.

In reference [10], a secure key management scheme is proposed based on *(t, n)* thresholds cryptography has been presented. Where *n* is the total number of nodes in network and *t* is the number of nodes required to generate certificate. The system can tolerate *t-1* compromised servers, however, this scheme does not describe how a node can contact *t* servers securely when server are scattered in a large area and minimum *t* number of servers nodes have to present on the ground every time, otherwise a new node cannot join network until *t* servers nodes available. Communication to *t* nodes increases the congestion in network.

Reference [11] proposes a threshold cryptography based scheme suited for MANET to provide robustness and defense against single point of failure in the central server. But main drawback of threshold cryptography is the difficulty in applying the distributive function. Other drawbacks are same as described in [10].

Bing Wua els [12] propose a secure and efficient key management (SEKM) framework for MANETs. They build a public key infrastructure (PKI) by applying a secret sharing scheme and using an underlying multi-cast server groups. Problem with scheme is mobile nodes needs large storage to carry certificates and to perform public key cryptography based large computation.

S. Capkun et al [13] suggested a method based on the users issuing certificates to each other based on personal acquaintance. These certificates are used to bind a public key and node identity. Every node should collect and maintain an up-to-date certificate repository. Certificate conflict is just another example of a potential problem in this scheme.

In this paper, we proposed a CE-GKMS which uses identity based cryptography for secure group communication in MANET. This scheme does not need any online certification authorities.

3. The Proposed CE-GKMS Scheme

Our proposed scheme CE-GKMS addresses the issue of group key management used in assuring group communication confidentiality. Group confidentiality requires that only valid users could decrypt the multicast data even if the data is broadcast to the entire network. The confidentiality requirements can be translated into following key distribution rules:

Non-group confidentiality: Nodes that were never part of the group should not have access to any key that can decrypt any multicast data sent to the group.

Forward secrecy: Nodes which left the group should not have access to any future key. This ensures that a member cannot decrypt data after it leaves the group.

Backward secrecy: A new node that joins the session should not have access to any old key. This ensures that a member cannot decrypt data sent before it joins the group.

In order to meet the above requirements, we propose a new scheme CE-GKMS to dynamically manage rekeying process in multicast groups. The scheme introduced a new idea by hiding the public keys and making them visible only to the trusted nodes and thus difficult for the cryptanalyst to crack the private information.

3.1 Optimizing RSA

Our proposed scheme CE-GKMS uses RSA for asymmetric cryptosystem and assumes that each node is able to generate its own public and private keys pair using RSA. This section presents RSA, how to speed its cryptography operations.

RSA is based on the equation:

$$e * d = 1 \bmod \phi(N)$$

$$where, \phi(N) = ((p-1)*(q-1))$$

Where, e is the public key exponent, d is the private key exponent: N is the modulus of RSA (i.e. $N = p * q$, where p and q are two large prime numbers). Large value of p and q in RSA makes its secure and less vulnerable to various attacks. But it causes encryption and decryption operation slower. The performance of RSA for encryption and decryption can be improved as described below:

Speeding up the RSA encryption and verifying

For RSA encryption and verification process, it is suggested to use small public key exponent *e*. but this makes it possible from an attacker to recover the plaintext from the cipher text if $m<N^{1/e}$, where *m* is the message [13]. As our proposed scheme disclose the public key of a node to trusted node only, it is possible to keep small value of encryption exponent *e*. Since, public key exponent is

secured here; RSA is safe from Private Key cryptanalysis attacks.

Speeding up the RSA decryption and signing

The performance of RSA for decryption and signing of a message can be improved using Chinese Remainder Theorem (CRT). Time required computing the decryption of a long message (m) in RSA, i.e. $m = C^d \bmod N$, can be improved using CRT [13] as following:

$$m = (m1, m2)$$

$$m1 = (C_p^d \bmod (p-1))$$

$$m2 = (C_q^d \bmod (q-1)/2)$$

This is called the modular representation of message m.

Using small secret exponent d and CRT allows rewriting the equation: $e.*d = 1 \bmod((p-1)*(q-1))$ having complexity $\theta(\log d \log^2 N)$ [14] using two forms:

$$e*d = 1 \bmod(p-1)$$

$$e*d = 1 \bmod(q-1)$$

The proposed scheme uses the idea of hiding the public keys and making them visible only to the trusted nodes which make it difficult for cryptanalyst to crack the private information. Also, with the use of small public key exponent e and small secret CRT based private key exponent d in RSA, asymmetric cryptography operations become faster.

3.2 System Model

The scheme we propose for key management in ad hoc networks, assumes the existence of a clustering protocol which split the network into groups that are stable enough as shown in Figure1.

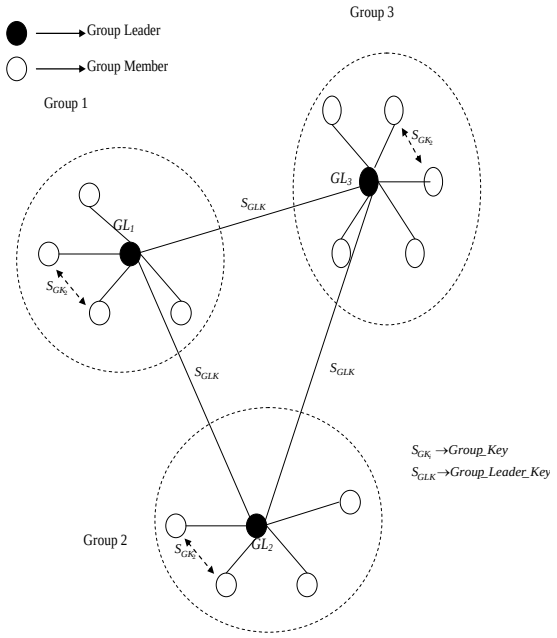


Figure1: System Model for MANET with number of groups using Clustering protocol

The users in the group are divided into two types: group leader and ordinary members. The group leader is responsible for group management, membership maintenance and group key creation, distribution and updating. We assume the distance between a group leader and its group member is considered not more than one hop and every node receive a message broadcasted from other node in the group. Each node in the group is equipped with unique and unforgeable identity i.e. finger print. A node within a group can take the role of a group leader to perform Group Key Generation Distribution Algorithm (GKGDA) for group key management. If there are multiple initiators, then the node with the smallest identity value will win as the leader and will execute GKGDA to completion to generate a group key, say S_{GK}. Group leader will disseminate this key to all of its group members. Once a leader is generated in all the groups, all group leaders will execute GKGDA to agree on a secret group leader key. If there are multiple leaders initiating the execution of GKGDA, the leader with the smallest identity value will win as the coordinator to execute GKGDA to completion to generate secret group leader key, say S_{GLK}. Group Coordinator (GC) will disseminate this key to all group leaders in the network. This key S_{GLK} will be used for intergroup communication. Group Coordinator maintains the list of all group leaders. In MANET, any node can join or leave the network at any time. If there is any change in the member list of either group leader or group coordinator, re-keying process take place for group key or group leader key respectively in order to ensure both forward secrecy and backward secrecy. The proposed scheme CE-GKMS uses a two level of hierarchy with GC

at the top and group members as a leaf node as shown in Figure2.

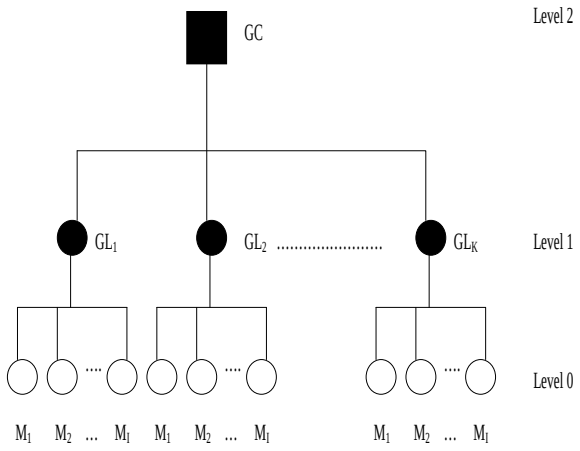


Figure2: Two-Level environment in CE-GKMS

The aim of constructing the grouped based architecture is not only to reduce the quantity of keys but also to reduce the time required for rekeying process when group membership change. The proposed scheme CE-GKMS uses some notation described in Table1.

Table1. Notation used in the Proposed Key Management Scheme

Notation	Description
G_i	Group having index *i*
GL_i	Group leader from group *i*
GL_{ID}	Identity of group leader
X_{ID}	Identity of node X
$(e_{GL_i,N_x}, d_{GL_i,N_x}, n_{GL_i,N_x})$	Public Key Cryptosystem (PKC) between Node X and its group leader
$E_k(m)$	Message *m* is encrypted using key *k*
$D_k(m)$	Message *m* is decrypted using key *k*
(e_i, d_i, n_i)	Public key cryptosystem (PKC) of node *i*
$X \rightarrow Y:(\alpha)$	Node *X* is sending message α to node *Y*
S_{GKi}	Secret symmetric group key used within group *i*
S_{GLK}	Secret symmetric group leader key used between group leaders

4. Working of CE-GKMS

When Group Leader receives a beacon signal from a node say X, Group Leader checks if he has already exchanged the PKC with node X earlier i.e. in case if communicated with node X earlier. If yes, and validity of PKC is not expired, group leader perform group key generation and distribution phase for new group key generation. If group leader not communicated with node X or the validity of PKC is expired, group leader execute public key distribution phase to exchange new PKC with node X before the execution of group key generation and distribution phase.

4.1 Public Key Distribution Module

Group Leader and node X generate a secret session key which is required for public key transferring between them. Here, we are motivated with the scheme proposed in [16] for the secret session key as follows:

Step1. Node X sends a following message to Group Leader GL.

$$X \rightarrow GL:(\alpha, b, s)$$

$$\alpha = b^{GL_{ID} * X_S} \bmod s$$

Where GL_{ID} is the identity of Group Leader and X_S is the secret used by node X. b is a primitive root mod s.

Step2: Group Leader GL sends the value β to node X.

$$\beta = b^{X_{ID} * GL_S} \bmod s$$

Where, X_{ID} is the identity of node X and GL_S is the secret used by Group Leader.

Step3: Both nodes calculate the same secret session key K_S at their end as:

$$K_S = \alpha * \beta$$

$$K_S = ((b^{GL_{ID} * X_S} \bmod s) * (b^{X_{ID} * GL_S} \bmod s))$$

$$K_S = b^{(X_{ID} * GL_{ID} * X_S * GL_S)} \bmod s$$

$$e_{GL_i,N_1}, e_{GL_i,N_2}, \ldots e_{GL_i,N_m}$$

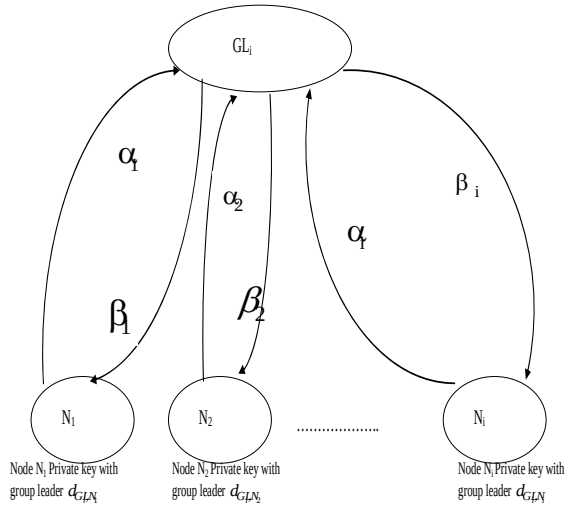


FIGURE 3: Session secret key and public key generation and distribution between group leader and its group members

Both nodes will generate respective public key cryptosystem using RSA i.e. public key using short exponent *e* and private key using small secret CRT exponents *d*. The newly created secret session key K_S is used to exchange the public key of group leader and node X with each other in a secure fashion. The public key of node X is disclosed to only trusted member, i.e. group leader, and hides from other members of the same group. Key K_S is used just once, after transferring the public

key, both nodes remove K_S key from their database. The proposed scheme assumes that there is no man in the middle attack. Similarly, all members of Group exchange secret session key thus Public key with Group Leader shown in Figure3. Group leader uses different PKC's with each members of group.

4.2 Group Key Generation and Distribution Module

Whenever a node comes within the radius of any Group, first it will send a hello message in the group along with its identity. Group Leader executes GKGDA for following steps:
Exchange public key with the new node (if required).
Regenerate the new group key using public key of each group members.
Distribute the new group key to each member of group in secure fashion ensuring forward and backward secrecy.
The detail of GKGDA is described next: Let N_1, N_2, N_3, ….., N_m are different ordinary nodes in a group. After receiving the public key of each member of group, Group Leader initiate the group key calculation as follows:

$$S_{GK_i} = ((b)^{((\sum_{i=1}^{m} e_{GL,N_i}) + GL_{Secret})} \bmod s) * R_s$$

Where, b is the primitive root of s and s is a prime number. e_{GL_i,N_1}, e_{GL_i,N_2},... e_{GL_i,N_m} are the public key of each Group members N_1, N_2, N_3, ...N_m respectively. Nonce R_s is a secret random number generated every time while rekeying. GL_{Secret} is the secret key of Group Leader. Group Leader distribute group key S_{GK_i} among group members with respective public key using RSA as follows:

$$\forall k : All\_members(k) = E_{e_{GL_i,N_k}}(S_{GK_i})$$

$$E_{e_{GL_i,N_k}}(S_{GK_i}) = (S_{GK_i})^{d_{GL_i,N_k}} \bmod n_k$$

All nodes decrypt message using respective private component as follows:

$$\forall k : All\_member(k) =: D_{d_{GL_i,N_k}}(E_{e_{GL_i,N_k}}(S_{GK_i}))$$

$$D_{d_{GL_i,N_k}}(E_{e_{GL_i,N_k}}(S_{GK_i})) = (E_{e_{GL_i,N_k}}(S_{GK_i}))^{d_{GL_i,N_k}} \bmod n_k$$

The complete working of GKGDA of the proposed key management scheme is explained in Algorithm1.

Algorithm 1: Group Key Generation and Distribution algorithm (GKGDA) executed by Group Leader of group *i*, say GL_o in reception of beacon signal from a group member, say node X.

Group Leader GL_o receives a beacon signal from node X which includes the identity of node X.
Begin
If (Group Leader communicated with node X earlier)
Then
{ Group Leader already exchanged the public key component (PKC) with node X
If (Validity of PKC is not expired)
{
Goto *Group Key Generation and Distribution module*;
}
}
If ((Group leader not communicated with Node X earlier) or (the validity of PKC is expired))
Then
{
*** Public Key Distribution module ***
a. Group Leader GL_i initiate process to establish secret session key with node X by sending message

$$GL \rightarrow X : (\alpha, b, s)$$

$$where, \alpha = b^{Y_{GL} * GL_s} \bmod s$$

b. In response, node X sends a message back to Group Leader

$$X \rightarrow GL : \beta$$

$$\beta = b^{GL_{Sec} * X_S} \bmod s$$

c. Both Group Leader and node X compute same secret session key K_S at their end.

$$K_S = \alpha * \beta$$

$$K_S = ((b^{X_{Sec} * GL_s} \bmod s) * (b^{GL_{Sec} * X_S} \bmod s))$$

$$K_S = b^{(X_{Sec} * GL_{Sec} * X_S * GL_s)} \bmod s$$

d. Both Group Leader head and node X generate their public private key pairs (e, d, n) using optimized RSA. Newly established secret session key K_S is used just once, to exchange the public key of node X and Group Leader each other in secure fashion.
}

*** Group key Generation and Distribution module ***
1) Group Leader OF Group *i* uses the public key of all group members to calculate the group key S_{GK}.

$$S_{GK_i} = ((b)^{((\sum_{i=1}^{m} e_{GL,N_i}) + GL_{Secret})} \bmod s) * R_s$$

2) Group Leader of group *i* uses RSA algorithm for the distribution of group key S_{GK} among all the members of Group using their public key component (e, n).

$$\forall k : All\_member(k) = E_{e_{GL_i,N_k}}(S_{GK})$$

$$E_{e_{GL_i,N_k}}(S_{GK}) = (S_{GK})^{d_{GL_i,N_k}} \bmod n_k$$

3) All nodes decrypt the message using private key component of PKC and receive group key S_{GK} in secure fashion.

$$\forall k : All\_member(k) =: D_{d_{GL_i,N_k}}(E_{e_{GL_i,N_k}}(S_{GK}))$$

$$D_{d_{GL_i,N_k}}(E_{e_{GL_i,N_k}}(S_{GK})) = (E_{e_{GL_i,N_k}}(S_{GK}))^{d_{GL_i,N_k}} \bmod n_k$$

End

4.3 Network Dynamics

New Node joins
When a new member joins the group, it sends a join request to the Group Leader. Group Leader must ensure that new node is not able to receive or interpret the previous information that was exchanged prior to its joining the network. In our scheme, the Group Leader executes GKGDA algorithm to regenerate the new group key. It then, broadcast new group key to the old existing members by encrypting it using old group key. Group Leader unicasts new group key to new joining node encrypted with its public key using RSA.
Group Leader regenerate the new group key by including the public key of new node in key generation process as follows:

$$S_{new\_GK_i} = ((b)^{((\sum_{k=old\_members+new\_member} e_{GL_i,N_k}) + GL_{new\_Secret})} \bmod s) * R_s$$

This new group key S_{new_GK} encrypted with the old group key (previous symmetric key) and multicast to all the existing group members as follows:

$$\forall k : Existing\_members(k) : ((E_{S_{GK_i}}(S_{New\_GK_i}))$$

Group Leader uses new node, say N_r , public key for encryption and unicast it using RSA.

$$GL_i \rightarrow New\_node(r) : E_{e_{GL_i,N_r}}(S_{New\_GK_i})$$

$$where, E_{e_{GL_i,N_r}}(S_{New\_GK_i}) : (S_{New\_GK_i})^{e_{GK,N_r}} \bmod n_{GK_i,N_i}$$

As new node have no information about the previous group key, new node cannot interpret the previous messages which ensures backward secrecy.

Existing member leaves

Whenever an existing node leaves the group, Group Leader again regenerate the new group key ensure that leaving node should not receive the later information exchanged after it leaves the group. Group leader calculate new group key using public key of all the existing group members except leaving node, say N_q.

$$S_{new\_GK_i} = ((b)^{((\sum_{p=1,i\neq q}^{m} e_{GL_i,N_p}) + GL_{new\_Secret})} \bmod s) * R_s$$

Group Leader unicast $S_{new\_GK_i}$ to each group member, using respective public key, in a secure fashion.

$$\forall k : Existing\_members(k) = E_{e_{GL_I,n_K}}(S_{New\_GK_i})$$

$$E_{e_{GL_I,n_K}}(S_{New\_GK_i}) = (S_{New\_GK_I})^{e_{GL_I,n_K}} \bmod n_{GL_I,n_K}$$

As new node have no information about the new group key, leaving node cannot interpret all further information which ensures forward secrecy. There is no need to generate new group leader key because no change of group leader in any group.

Group Leader Leaves

When a group leader leaves, both new group key and new group leader key will be generated. There are two situations when group leader leaves the group:

Group leader leaves group suddenly without prior information

Group leader inform before leaving the group

Group key should be changed if group leader leaves the group suddenly without prior information i.e. battery go down. A new group leader is elected by group members to replace the leaving group leader. Group member with smallest identity will be selected as new group leader. If group leader inform prior leaving the group, it first elect new group leader with next smallest identity. The old group leader transfer all secret and group membership information to new elect group leader encrypted with its public key using RSA, and then deletes this information from its storage. Because we assume that a node is trusted only if it is a group leader, after it can be compromised. Therefore, it is necessary to delete confidential information, so that any information about key generation could not be revealed to outsider node. If it is not deleted, new group leader exchange new PKC with each group member and finally regenerate new group key to ensure forward secrecy.

In either case, when group leader leaves, first new group leader regenerate new group key by executing GKGDA and that is used for group member communications. As the group coordinator membership list updated, group coordinator regenerate new group leader key for secure communication between the group leaders.

4.4 Communication Protocol

Intra-group Communication

The members within the group G_i communicate using the group key. Suppose, member N_j want to send a message to member N_k in the same group. It encrypts the message with the group key S_{GK_i} as shown in figure.

$$N_j \rightarrow N_k : E_{S_{GK_i}}(m)$$

$$Where node\ N_k\ decrypt\ message with\ group key\ S_{GK_i}$$

$$N_k \rightarrow D_{S_{GK_i}}(E_{S_{GK_i}}(m))$$

Inter-Group Communication

The members in one group communicate with member of other group through their respective group leaders. Suppose member N_m in group 1 wants to communicate with member N_s in group 2, it first sends the message encrypted with group key of group 1 to its group leader. Then group leader decrypt the message and encrypt it again using group leader key before sending to the parent group leader of N_s. The group leader of group 2 decrypts the message using group leader key. Now, group leader of group 2 forward this message by encrypting its group key. Finally, N_s will decrypt the message using its group key. All the steps are described in Figure4.

Step1: $N_m \rightarrow GL_1 : E_{S_{GK_1}}(m)$

Step2: $GL_1 \rightarrow GL_2 : E_{GLK}(D_{S_{GK_1}}(E_{S_{GK_1}}(m)))$

Step3:

$$GL_2 \rightarrow N_s : E_{S_{GK_2}}(z)$$

$$where,$$

$$z = D_{GLK}(E_{GLK}(D_{S_{GK_1}}(E_{S_{GK_1}}(m))))$$

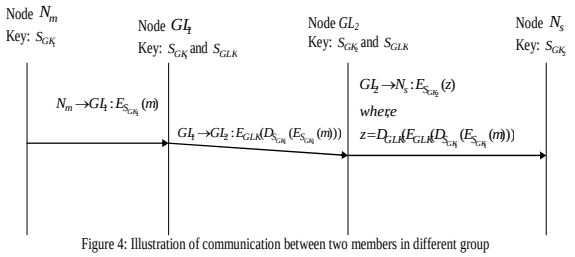


Figure 4: Illustration of communication between two members in different group

5. Performance Analysis

5.1 Security Analysis

As there is a lack of centralized authority for authentication, members of mobile ad hoc network are considered to be the part of security issue. We assume that all of the incoming members of the MANET carry an unforgeable unique identity.

Forward Secrecy
When an existing node leaves a group, leaving node should not receive any information from the network. There are two cases:
Case 1: when group member leaves
A new group key is regenerated by the group leader every time when a group member leaves. This new group key will be sent to each existing member using public key of individual group member in a secure fashion. Thus, leaving member cannot have this new group key and also not receive later information.
Case 2: when group leader leaves
Whenever a group leader leaves, the members in the group communicate wit each other and elect the node with smallest identity as the group leader. This group leader execute GKGDA to regenerate the new group key and sends it to all the group members encrypted using respective public key in a secure fashion. As there is a change in membership of group leader list, Group coordinator execute GKGDA to regenerate new group leader key and sends it to all group leaders encrypted using respective public key in secure fashion.
Case 3: when group coordinator leaves
When a group coordinator leaves, first group leader elect new group coordinator with smallest identity. This elected group coordinator execute GKGDA to regenerate new group leader key and sends it to all group leaders encrypted using respective public key in secure fashion.
This ensures that the forward secrecy is maintained and the group key is communicated in a secure fashion.

Backward secrecy
When a new member joins a group, we ensure that new member is not able to access the previous information that was exchanged prior to its joining the network. In our scheme, group leader regenerate the group key and broadcast to all the existing members encrypted using old group key and unicast it to new node encrypting using its public key. Therefore, new member can encrypt and decrypt later information but cannot decrypt the previously exchanged information.
Node Compromised
When a group member is compromised by an adversary, there is no threat to the security as only the group key is compromised which can be regenerated by the group leader and distributed to the other members in a secure fashion. This is similar to when group member leaves the group. When a group leader is compromised, security mechanism in only a part of the network is affected and can be rectified with in a short span of time. Whereas, in centralized key management scheme, if centralized key distribution center itself is compromised, the security of the entire network fails. This indicates that our approach is robust against attacks.

5.2 Cost Analysis

We compute communication cost of our proposed scheme under different network organizations. We also compare the communication cost of rekeying for various schemes. For the computation, we use some notation described below:

N	Network Size
M	Group Size
G	Number of Groups
GC	Group Coordinator
GL	Group Leader
GM	Group Member

Member joins
When a new member joins, first group leader exchange PKC with new group member and then new group key is calculated and broadcast to all the old members encrypted with previous group key old group key and unicast to new member using RSA. Let, the average number of members in a group is M. The scheme requires two messages for PKC, one broadcast message to all existing member and one unicast message to new joining member. The group key of other groups need not be changed. The number of messages required for rekeying is shown in Table2.
Members Leaves
When a node leaves, there are two cases
The group member leaves
The group leader leaves
The group Coordinator leaves
When group member leaves
When group member leaves, the new group key is regenerated and unicast to each old member encrypted

with respective public key using RSA i.e. M-1 unicast messages

When Group Leader Leaves

When group leader leaves, two actions performed.

First, a new group leader is elected and exchanges PKC with all the existing group members then group key is regenerated and unicast to each members of the group encrypted with respective public key using RSA i.e. M-1 unicast messages. Second, group coordinator exchange public key with the new group leader and then regenerate the new group leader key and unicast it to all the group leaders encrypted with respective public key using RSA i.e. G-1 unicast messages.

When group Coordinator Leaves

Group coordinator leaves the group event is same as group leader leaves because in both cases there is a change in group leader list and group coordinator list. As group coordinator also worked as group leader, first new group leader is elected in the group of coordinator which regenerate the new group key and disseminate among the group members encrypted with respective public key using RSA i.e. M-1 unicast messages. Second, new group coordinator is elected which regenerate new group leader key and disseminate it to all the group leaders encrypted with respective public key using RSA i.e. G-1 unicast messages.

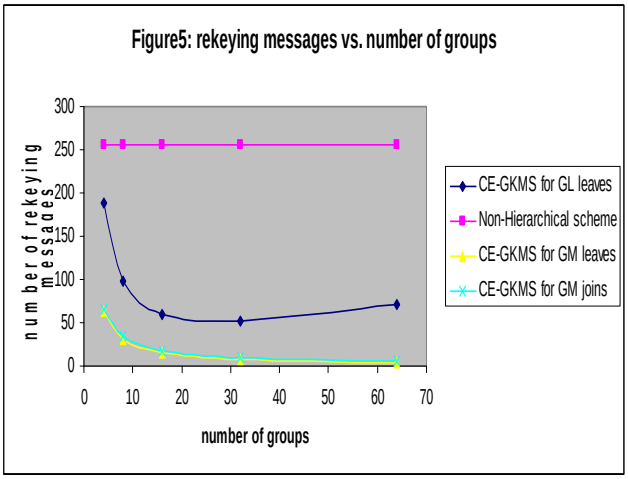
Table2: No. of Rekeying Messages for different network sizes

Network Organization	Number of nodes that receive rekeying messages in our scheme B=Broadcast and U=Unicast		Non-Hierarchical scheme
	GM leaves or GM joins	GL or GC leaves	For all cases of GM & GL
N=256 M=64 G=4	Join= 2 + 63(B)+1(U)=66 Leave=63U	124+62+2+3=189U	255U
N=256 M=32 G=8	Join= 2 + 31(B)+1(U)=34 Leave=31U	60+30+2+7=99U	255U
N=256 M=16 G=16	Join= 2 + 15(B)+1(U)=18 Leave=15U	28+14+2+15=59U	255U
N=256 M=8 G=32	Join= 2 + 7(B)+1(U)=9 Leave=7U	12+6+2+31=51U	255U
N=256 M=4 G=64	Join= 2 + 3(B)+1(U)=6 Leave=3U	4+2+2+63=71U	255U

The improvement of the proposed scheme CE-GKMS over non-hierarchical for number of rekeying messages required for maintaining group confidentiality is also shown in figure5. The number pf rekeying messages reduced as the number of groups increases. The purpose of grouping is to reduce the number of group members at level 0 but at the same time need to check on the size of group leaders at level 1. When number of group leader increases at level 1, threshold point arrives which start increasing the number of rekeying messages. When a group member leaves or joins, only group key is required at level 0 because there is no change in the member list of group leader at level 1. The number of rekeying messages is more when a group member joins than a group member leaves because group leader first exchange PKC with the new coming group member.

6. Conclusion and Future works

Security schemes are an urgent need for multicasting to ensure a secured deployment for confidential group communication. Key management schemes play a key role in the whole secure multicast groups. This paper introduces a certificate less group key management scheme for MANET which overcomes the major challenges of dynamic keying scheme.



The scheme does not need PKI which required large computation capability and memory storage for each node in the network. Our scheme is efficient because each group member requires performing one encryption to guarantee the confidentiality of communications. The proposed scheme not only reduced the number of rekeying messages with improvement over non-hierarchical schemes but also makes it efficient using fast asymmetric cryptography operation permitting small PKC. Thus, the proposed scheme is efficient and secure for large and dynamic multicast groups.

References

[1] Mishra. A, Nadkarni, K, and Patcha, A," Intrusion Detection in wireless ad hoc networks", IEEE Journal Personal Communication on wireless communication, 11(1), 48-60, 2004

[2] Patwardhan, A., Parker, joshi, A., Iorga, M. & karygiannis, T. , " Routing and intrusion detection in ad hoc networks" in proceeding of the 3rd IEEE international conference on pervasive computing and communication, pp. 191-199, Kauai Island, USA, 2005

[3] Schmoyer, T. R. Lim, Y. X. & Owen, H. L.," wireless intrusion detection and response: a classic study of man in the middle attack", proceedings of the 2004 international conference on wireless communications and networking, pp8830888, Atlanta, USA, 2004.

[4] B. Wu, J. Chen, J. Wu, M. Cardei, "A survey of attacks and countermeasures in mobile ad hoc networks", Wireless Network Security, volume: 30, issue: 3, pages: 103-135, 2007

[5] K. S. Win, "Analysis of detecting wormhole attack in wireless networks", WORLD ACADEMY OF SCIENCE, ENGINEERING AND TECHNOLOGY, Volume 36, Dec 2008

[6] N. Kettaf, H. Abouaissa, P. Lorenz, "An Efficient Heterogeneous key management approach for secure multicast communication in ad hoc networks", Springer telecommunication System, vol-37, pp 29-36, February 2008

[7] Chang. C. C. and Chung, "An efficient session key generation protocol", proceeding of the 2003 IEEE international conference on communication technology, pp. 203-207, Beijing, China

[8] Rafeali S. and Hutchison D., "A survey of key management for secure group communication" ACM Computing Surveys, 309-329, 2003

[9] R. Blom, "Optimal class of symmetric key generation systems:, proceeding of EUROCRYPT 84 workshop on Advances in Cryptology: theory and application of cryptographic techniques, pp 335-338, December 1985, France

[10] L. Zhou and Z. J. Hass, " Securing Ad hoc networks", IEEE Network, vol 13, no-6, pp24-30, 1999.

[11]H. Nam Nguyen, H. Morino, "A key management scheme for mobile ad hoc network based on Threshold Cryptography for providing fast Authentication and low signaling load", EUC Workshop 2005, LNCS 3823, pp. 905-915, 2005

[12]Bing Wu, Jie Wu, Eduardo B. Fernandez, Spyros Magliveras, "Secure and Efficient key management in mobile adhoc networks", proceeding of the 19th IEEE International Parallel and Distributed Processing Symposium (IPDPS'05) IEEE 2005.

[13]S. Capkun, L. Buttyan and J-P Hubaux, "Self organized public key management for mobile adhoc networks", IEEE Transaction on Mobile Computing,2(1), January-March 2003.

[14].Menezes, P. van Oorschot, and S. Vanstone, "Handbook of Applied Cryptography", CRC Press, p:102-110,610-612, 1996

[15]Daniel Bleichenbacher, Alexander, "May: New Attacks on RSA with Small Secret CRT-Exponents", Public Key Cryptography, pages 1-13, 2006

[16]Anil Kapil and S. Rana, "Identity-Based Key Management in MANETs using Public Key Cryptography", International Journal of Security (IJS), Vol. (3), Issue (1), pp 1-8. Jan 2008

First Author Sanjeev Kumar is working as an Assoc. Professor Department of Computer Engineering, M.M Engineering College, M.M. University, Mullana, Ambala. He obtained his BTech (Computers Engineering) from Kurukshetra University, Kurukshetra and MTECH (IT) from University School of Information Technology, GGSIP University Delhi. He has about 10 publications in various International journals/Conferences to his credit. His current research interest includes operating system, Wireless communications which include mobile Ad hoc network and sensor based networks, Network Security etc.

Second Author Dr. Manpreet Singh is working as Professor. & Head of computer science and Engineering department at MMEC, M. M. University Mullana, Ambala, India. He obtained his Ph.D. (Computer Science) from Kurukshetra University. He has number of publications in International journals/Conferences to his credit. His current research interest includes Grid Computing, Wireless communications, MANETs etc.