IJCSI International Journal of Computer Science Issues, Vol. 7, Issue 3, No 10, May 2010
ISSN (Online): 1694-0784
ISSN (Print): 1694-0814


# Policy based Decentralized Group key Security for Mobile Ad-hoc Networks

**Mrs. Sugandha Singh[1], Dr. Navin Rajpal[2], Dr. Ashok Kale Sharma[3] and Mrs. Ritu Pahwa[4]**

[1] Associate Professor, C.S.E. Department, Jodhpur Institute of Engineering and Technology, JIET, Jodhpur, Rajasthan, 342001, India

[2] Professor, I.T. Deptt., University School of Information Technology, USIT, Delhi, India

[3] Professor & Head, C.S.E. Departt., YMCA College of Engineering Faridabad, India

[4] Lecturer, E.C.E. Deptt., Vaish College of Engineering Rohtak, 124001 India

**Abstract**

The unique characteristics and constraints of MANET have made the traditional approach to security inadequate. With this view in mind decentralized group key management is taken into consideration. A novel structure of the node is proposed and each entity holds a secret share $SS_i$ of each node in cluster is controlled by its cluster head, the policy enforcer decides for the working of intelligent agent, which is assigned to do the management, which allows two or more parties to derive shared key as a function of information associated with the protocol and so no party can predetermine the resulting value. Group membership certificate is used for group authentication and by the use threshold key scheme secret data is transferred. The $SS_i$ of each node is calculated by use of Polynomial interpolation and cluster head key by modular arithmetic, and information is carried by the policy based agents named intelligent agents.

***Keywords:*** *Mobile Adhoc Network, Group key management, Decentralized group key, Verifiable secret sharing.*

## 1. Introduction

Network contamination describes the situation where a network is polluted with unsolicited commercial, political and/or malicious software. It also degrades the utility of belonging of the network in that it imposes negative effects to the systems, networks and the users and today's networks are mobile adhoc networks, where every node act as a router also and can route the traffic to other nodes. They are highly dynamic in nature and susceptible to failures. Such networks pose stringent requirements for security and reliability. Unlike typical internet application, most applications of MANET involve one-to-many and many-to-many communication patterns. Group communication is typical mode in MANET and high level security is required in it.

The new requirements in security are arisen in MANET because of its characteristics [3], and the secure group key management meets some challenges as below:

1) It is hard to securely distribute and update the group key because there is lack of fixed infrastructure.
2) The solutions based on fixed topology structure of nodes and unpractical because all the locations of the nodes might change at any moment.
3) Multi-hop relaying and hidden attack make group re-keying hard to achieve satisfied performance.
4) The lack of an online CA or trusted third party adds the difficulty to deploy security mechanism.
5) Mobile devices tend to have limited power consumption and computation capabilities, which make it more vulnerable to denial of service attacks and incapable to execute computation heavy algorithms like public key algorithms.
6) There is more probability for trusted node being compromised and then being used by adversary to launch attacks on networks. It is difficult to distinguish between stale routing information and faked routing information. So it is necessary to deal with attacker inside the network.

There are five main security services for MANET's: **authentication, confidentiality, integrity, non-repudiation & availability**. In order to address these needs, a policy based network management system that has provided the capability to express network requirements at

a high level and have them automatically realized in the network by configuration agents. This approach provides the network administrator with the capability to specify high-level policies that:

- Specify long-term, network-wide configuration objectives, e.g. all private communications must be encrypted.
- Provide an automated feedback loop so that information reported by monitoring agents can be used to automatically trigger correction of network problems based on policies.

Once policies such as those described above are defined, they are automatically enforced by the policy enforcer. These capabilities can provide military personnel with very powerful tools to configure and control their network, and reconfigure their network, in response to network conditions [4]. In general Group Key management security in ad-hoc networks is divided into three main classes:

1. **Centralized group key management protocols:** A single entity called the key distribution center (KDC) is employed for controlling the whole group.
2. **Decentralized group key management protocols:** The management of the large group is divided among subgroup managers, trying to minimize the problem of concentrating the work in a single place.
3. **Distributed group key management protocols:** There is no explicit KDC, and all the members participate in the generation of the group key and each member contributes to a portion of the key.

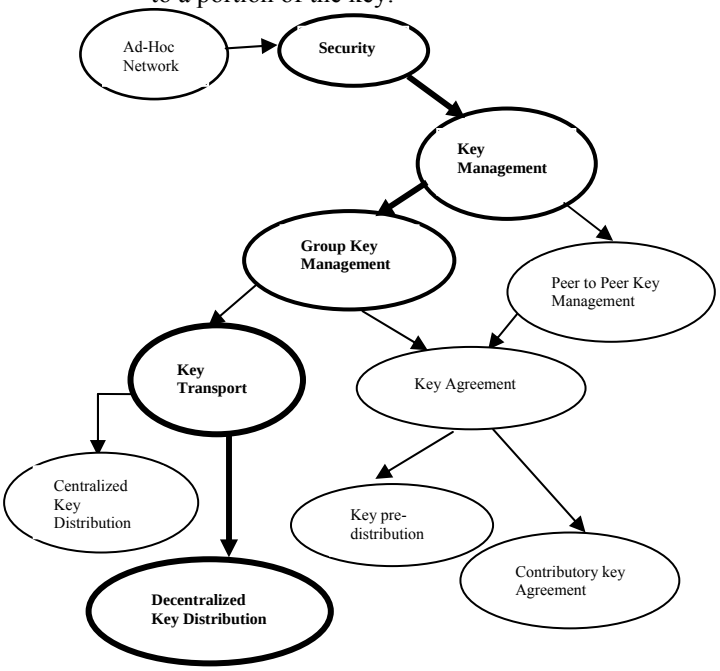


Fig.1 Summary of Area of work

This paper concentrates on decentralized group key management protocol. As it is assumed that the adversary can attack any node at any time, but not every node every time, so an efficient group key distribution or group key agreement is made that addresses the special needs posed by mobile ad-hoc networks and the system provides the capability to express networking requirements at a high level. They are then automatically released in the network by agents. Network management functionality is released by policy agents that are organized in a hierarchy to provide both scalability and autonomy. Survivability is achieved by enabling any component to take over the role of another component in case of failure.

The rest of the paper is organized as follows: In Section 2, a review of the related work is given. In section 3 the proposed policies on Group key security is explained. Finally the discussion on some performance issues is shown in Section 4. Section 5 considers the further work and finally acknowledgments for helping hands.

## 2. Related Work

Decentralized Group key Management Protocols: In the decentralized subgroup approach, the large group is split into small subgroups, minimizing the problem of concentrating the work on a single place. In this approach, more entities are allowed to fail before the whole group is affected. Following attributes are used to evaluate the efficiency of decentralized frameworks: key independence, decentralized controller, Local re-key [5], keys vs. data and Re-key per membership protocol is based on password authenticated multi party Diffie-Hellman Key exchange. Protocol described in [6] does not give any idea about the structure of Ad-hoc network and described in a vague way and the structure of the final session key is not the same as explained in the protocol. Scalable multicast key distribution [7], Kronos [8], Intra-Domain Group key management [9], Hydra [10] are some of the popular protocols that follow the decentralized architecture. [10] gives the contributory group key agreement gives the group key protocol: $K = H(N_1,N_2,\ldots\ldots,N_n)$ where $H(\ )$ is a one way collision hash function and $N_i$ is the secret key share of group member $P_r$. Each group member $P_i$ chooses a secret $X_i$ and computes: $Z_i = g^{xi}$ and each group member broadcast $Z_i$ to all other group members. Each

group member computes and broadcast: $X_i = (Z_{i+1}/Z_{i-1})^x_i$ [11]. Consider the security mechanism in from the system architecture view. It depicts the five layer security architecture for MANET's as: Layer 5 $SL_5$, End to End security layer. Layer 4 $SL_4$ Network Security Layer. Layer 3 $SL_3$, Routing security Layer. Layer 2 $SL_2$, Communication security layer. Layer 1 $SL_1$, Trust Infrastructure Layer. [12] has implemented key management service and described the use of RSA key generation technique to create a threshold certificate authority. The creation of this scalable key management solution does not rely on prior infrastructure for its inception. Public key Infrastructure (PKI) is the most scalable form of key management. Several different PKI techniques exist: [13], [14], and [15]. Aura [16] proposes the use of a group oriented Public key infrastructure for large group formation. The leader of the group acts as a certificate authority (CA), which issues group membership certificates. Zhou [17] suggests the use of threshold cryptography to create a distributed threshold certificate authority.

## 3. Proposed Work

A dynamically adjustable multi-tier hierarchy is used which enhances the scalability of the management system by expanding or shrinking of number of tiers in hierarchy depending on the network conditions. The basic idea is to form the clusters and implement the threshold scheme (K, i) for the management of cryptographic keys, which are used for the security of the data while movement. **Threshold scheme** are ideally suited to the applications in which a group of mutually suspicious individuals or must say here the nodes with conflicting interests cooperate. Different types of agents are used and detailed according to their property to do specific jobs. **Intelligent agents/Policy agents** are the agents who are to behave intelligently. This *policy agent* is responsible for enforcing the policies of the policy domain. The policy agent of an atomic policy domain is referred to as ***Local Policy Agent (LPA).*** The policy agent of top level policy domain is referred to as ***Global Policy Agent (GPA).*** Intermediate Policy agents are called ***Domain Policy Agents (DPA).***

### Assumptions

The GPA, DPA and LPA of all the networks have the same basic structure and consist of same code base. They have similar functionality but different scope. Different agents can be installed within the GPA, DPA's or LPA's to enable different functionality. **Policy Enforcer** is the entity responsible for enforcing policies. It monitors events and evaluates conditions to decide which agent should be instructed to perform its management action. They can receive events published by other system components via an event bus. Each agent implements a **standard interface** that enables the policy enforcer to communicate with agents. **Policy distributor** is used to receive policy updates from the remote node and to send these updates to the LPA's on its node.
The network considered is not very highly volatile.

### 3.2 Working of System

Ideally it is considered that the cooperation is based on mutual consent, but practically the veto power this mechanism gives to each member or each node can paralyze the activities of the cluster. By properly choosing the ***k*** and ***x*** parameters we can give sufficiently large majority of authority to take some action while giving any sufficiently large minority the power to block it. The proposed new structure of each node is shown below. This includes the node identification number (n), current location of the node $(X_n,Y_n)$ which is calculated with the GPS which is on each and every node. It also helps in predicting the direction on movement of the node. Cluster Head Identification number (CH) is used to specify the cluster leader and different number of clusters in one network. The threshold scheme considered is the dynamic threshold scheme because let's say threshold minimum number of votes is say ***t*** then proactive secret sharing doesn't help as resultant size of the group is less than t i.e. ***(n-t)*** <= ***t*** so in such a case it is necessary to reduce ***t***. Similar condition is if larger member of the group leaves. Similar condition arises when at group inception time first few members join. In such special cases the group needs some special admission rules. In dynamic threshold the minimum number of votes is a fraction of the number of current group member. As it is decentralized Scheme one network (N) is divided in different clusters, which carries the information of all the members present in the cluster and when a member node becomes mobile it informs the cluster head (CH) about its migration and on traveling to a new region boundary it will send

request packet to the current cluster head for membership.

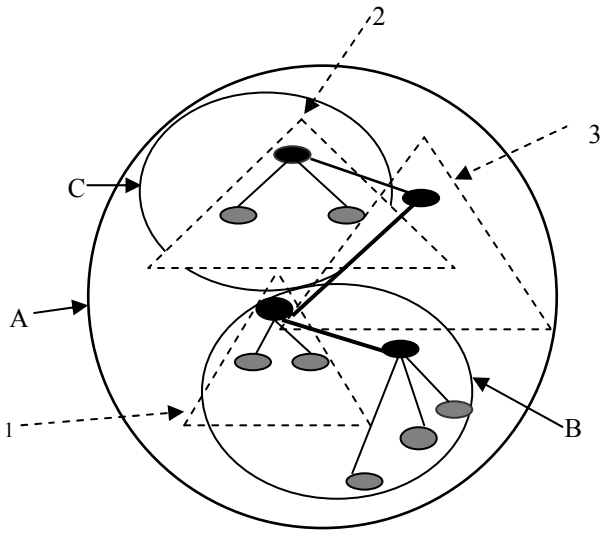


Fig. 2 Multi Tier Management Hierarchy

Thus our cluster heads are vested with the responsibility of keeping neighborhood integrity record with periodic refreshment. For this shift over we assume or impose the trust authority charged with particular's date membership authority, so it just keeps track of membership admission not on security of data for that we impose the key decentralization with threshold scheme. For initial cluster formation clusters and their leaders are specified by a pre-defined network plan (i.e. manually). After the initial formation of the clusters, it is necessary to maintain the management hierarchy for distributing policies and collecting management information. This is achieved by using the following method. The leader of the cluster named Cluster Head (CH) periodically sends heartbeat messages to its entire cluster associates, which is acknowledged as well by each cluster associate. If a cluster associate misses a number of consecutive heartbeat messages from its leader, **it assumes that the leader no longer exists** and tries to join the cluster of an ancestor of its current parent. If it fails to do so, it assumes the role of the GPA, i.e. the root of the management hierarchy. A cluster leader may decide to split its current cluster into two clusters, based on network conditions e.g. if the cluster become geographically dispersed and so the management performance is adversely impacted (which is detected using the GPS system of each node). The cluster leader then appoints one of its associates as the leader for a subset of its associates. If an associate detects the existence of another GPA, it notifies its own GPA, who then initiates a negotiation session with the other GPA. The other GPA signals the session initiator its decision to remain GPA or to become a child of the session initiator. Once the session initiator acknowledges this decision, the two domain merge and one of the GPA's step down to become a child of the other. The LPA carries the secret key named SK of the node and store it in the ACL (Access Control List) and the PK is carried with the GPA and stored in its ACL.

**(i)** ***Join Request by new Member:*** A newly joining member must acquire the secret share of the group secret SK for itself. This enables it to give access in future and voting procedure in order to admit other new members. A new member named, $M_{new}$, initiates the protocol by sending **JOIN_REQ** message to the group. Then after signed by $\mathbf{M_{new}}$ & contained among other values, $M_{new}$'s public key certificate $\mathbf{(PKC_{new})}$ and the target cluster name. The sending of message to cluster is application dependent and out of the scope of this paper. The newly joined or joining member should be able to receive to partial secret shares from each of the ***t*** members only; those who elect to admit this new node to be in their group secrecy of node and above to ***t*** are maintained secret from $M_{new}$

**(ii)** ***Voting:*** After receiving JOIN_REQ, a group member out of ***t*** first extracts the sender's $PKC_{new}$ & verifies the signature (Verification explained in sec.) and then moved for GMC.

**(iii)** ***Group Membership Certificate:*** Who will issue the $GMC_{new}$ for $M_{new}$ depends on security policies applied on the agents. Once enough votes are collected $M_{new}$ verifies the individual votes & computes its own $GMC_{new}$. Once $M_{new}$ become legitimate member it needs to obtain its own secret share $\mathbf{SS_{new}}$, which enables to participate in future admission protocols.

**(iv)** ***Secret Data Sharing:*** After the management the security of the data is required for which the threshold scheme is used ***(k, x).*** The basic secret sharing is to divide a secret s in to pieces or

ISSN (Online): 1694-0784
ISSN (Print): 1694-0814

shares which are distributed among x users a trusted dealer is chosen [1] and it chooses the large prime ***p*** & select the polynomial ***f(z)*** over to $Z_p$ of degree ***t-1***. To distribute share among The new node structure is used with the detail of in which cluster it is and the main network for which it is working for or the network basically which is sending the message. The structure of the node is as shown in Fig. 3

| Node Identification No. (n) |
|---|
| Current Location ($X_n$ , $Y_n$) |
| Cluster Identification No. (CH) |
| Network Identification (N) |
| Threshold Scheme (K, i) |

Fig. 3 Structure of the Node

The dealer computes each user share $SS_i$ such that $SS_i$= f(i)mod (q) and securely transfer $SS_i$ to $M_i$. Then any group of t members can recover the secret. This solves the issue specified in Eq. 1 by making use of polynomial interpolation and Modular arithmetic which gives the verification that the specified node is a valid node member of t. [1]

$$f(z) = \sum_{i=1}^{t} SS_i \, l_i(z) \bmod (p)$$

$$\text{Where } l_i(z) = \prod_{\substack{j=1 \\ i \neq j}}^{t} (z-j) / (i-j) \qquad (1)$$

Since f (0) = S the secret share is expressed as

$$S = f(0) = \sum_{i=1}^{t} ss_i \, l_i(0) \bmod (p) \qquad (2)$$

So secret share will be recovered only is minimum required keys of node are combined.

## 4. Performance *of Proposed work*

The network used for the simulation of 500 nodes in 250m X 250m simulation area. The movement of nodes is kept random so **nodes are allowed to move not more than 30m/sec.**

Performance is calculated for the below specified criteria's

1. Secret verification share has fixed time slot/ period. E.g. Mobile Intelligent agents hop around the network for delivering messages in this current flexible and decentralized framework any autonomous node can send message to any other node at any instant within the network by just issuing a mobile agent. The Intelligent agent then carries the message to the corresponding cluster head. The cluster head then becomes responsible for delivering the message to proper destination. Analog to the real life, these agents actually play the role of messengers and the cluster heads play the role of post offices in the adhoc wireless scenario. The cooperating agent scheme has been explicitly designed to reduce the agent traffic in the network. The unnecessary redundant node visits made by the agents moving for a common destination has been avoided by sharing and merging with other agents. These agents together with the cluster heads take the responsibility of providing communication services and improvement of overall traffic coordination in the network.
2. Number of Messages in the system i.e. the total number of mobile agents traffic issued by the node in network.

### *Verifiable Secret Sharing*

The t members receive their share $ss_i$ and each member $M_i$ verifies $ss_i$ by

$$G^{ss}{}_i = \prod_{j=0}^{t-1} (w_j)^{ij} \bmod (p)$$

Where $w_i$ is the witness and

$$w_i = g^{ai} \bmod (p)$$

The TD publishes this $w_i$-s in ACL of GPA.

### Acknowledgments

Author gives heartfelt thanks to both the guides Dr. Navin Rajpal and Dr A.K. Sharma for devoting time and patience for this work to get some conclusion and to the Director Er. Navneet Agarwal, JIET for his cooperation while completing this work.

### References

[1] A. Shamir. How to share a secret. Commun. ACM, 22(11), 1979

[2] Sugandha Singh, Dr. Navin Rajpal, Dr. A.K. Sharma. "Mobile Agent Based Message Communication in Large Ad hoc Networks through Co-operative Routing using Inter-Agent Negotiation at Rendezvous Points". In 4th international conference of challenges and developments in IT, at Punjab College of Technical education, May 2008.

[3] F. Stajano, R. Anderson. "The resurrecting duckling: Security issues for adhoc wireless networks," The 7th Int'l workshop on security protocols, LNCS 1796, Berlin: Springer, 2000. pp. 172-194

[4] Rony H. Rahman and Lutfar Rahman "A New Group key Management Protocol for Wireless Ad-Hoc Networks", International Journal of computer and Information Science and Engineering, Springer 2008. pp. 74-79

[5] B. Carlsson and A. Jacobsson, "Security Consistency in information Ecosystems: Structure of risk environment on the Internet", Journal of Information system security 2(1), p 2-26, 2006.

[6] N. Asokan and P. Ginzboorg, "Key agreement in ad-hoc networks" In Elsevier Journal of Computer Communications. Computer Commun. 23 (2000) 1627-1637.

[7] A. Ballardie, "Scalable Multicast Key Distribution". RFC 1949, 1996.

[8] S. Setia, S. Koussih, S. Jajodia and E. Harder. "Kronos: A scalable group re-keying approach for secure multicast". IEEE Symposium on security and Privacy, May 2000.

[9] B. DeCleene, L. Dondeti, S. Griffin, T. Hardjono, D. Kiwior, J. Kurose, D. Towsley, S. Vasudevan, and C. Zhang. " Secure Group communications for wireless networks". MILCOM, June 2001.

[10] Johann van der Merwe, Professor D. Dawoud and Mr. S. McDonald. " Military Mobile Ad hoc Network Security: Group Key Management". In Journal of ARMSCOR university of Kwazulu-Natal.

[11] Shuyao Yu, Youkun Zhang, Chuck Song and Kai Chen. "A security architecture for Mobile AdHoc Network" ACM Workshop on Wireless Security (Wise 2003), San Diego, CA, September 19, 2003

[12] B. Lehane, L. Doyle and D.O' Mahony. "Shared RSA Key generation in Mobile Adhoc networks" European Office of Aerospace Research and Development, 2003.

[13] http://world.std.com/~cme/html.charters/spki.html

[14] http://www.pgpi.org

[15] http://www.ietf.org/html.charters/pkix-charter.html

[16] T. Aura, S. Maki. "Towards survivable security architecture for ad-hoc networks" Security protocols 9th international workshop. Cambridge, UK, April '01, LNCS 2467, p 63-73 2002

[17] L. Zhou, Z. J. Haas. "Securing Ad-Hoc Networks" IEEE Networks, 13(6): 24-30, 1999.

[18] S. Rafaeli, and D. Hutchison. "Hydra: a decentralized group key management". 11th IEEE International WETICE: Enterprise security Workshop, June 2002.