Secure Group Communication using Multicast Key Distribution Scheme in Ad hoc Network (SGCMKDS)

V. Palanisamy
Associate Professor and Head
Department of Computer Science and Engineering Alagappa University,
Karaikudi , Tamil nadu

P. Annadurai
Kanchi Mamunivar Center for Post Graduate Studies (Autonomous)
Government of Puducherry, Lawspet,
Puducherry

ABSTRACT

In Recent years, secure communications have become an important subject of research. The new service for wireless and wired networks is to provide confidentiality, authentication, authorization and data integrity. Security has always been a sensitive issue. In fact, this service becomes necessary to protect basic applications, especially E-commerce and bank transactions from a variety of attacks. An ad hoc network is a kind of wireless communication infrastructure that does not have base stations or routers. Each node acts as a router and is responsible for dynamically discovering other nodes it can directly communicate with. However, when a message without encryption is sent out through a general tunnel, it may be maliciously attacked. Securing group communication and group key establishment for ad hoc networks is covered in this paper. For a secure group communication in ad hoc networks, a group key is needed to be shared between group members to encrypt group messages. The main idea is to have group members actively participate to the security of the multicast group, therefore reducing the communication and computation load on the source. Since the group security is distributed among the group members, we propose a service right certificate, to verify that a node is authorized to join the group, and also a corresponding revocation mechanism.

Keywords

Secure Group communication, Multicast Key distribution, Rekeying and Key Update and, Key Tree

1. INTRODUCTION

An Ad hoc networks require no centralized administration or fixed network infrastructure such as base stations or access points, and can be quickly and inexpensively set up as needed. They can be used in scenarios in which no infrastructure exists, or in which the existing infrastructure does not meet application requirements for reasons such as security or cost [1]. Multicast is a communication service that provides data delivery from a source to a set of recipients, also known as multicast group. Multicast's major advantage over unicast is that it allows the sender to send each packet just once; the routers automatically forward the packet to each receiver that wants it, while minimizing the number of copies of the packet that traverses the network. Most multicast protocols require the creation and maintenance of a structure (such as a tree or a mesh) for distribution of information to the group members.

Multicast communication is an efficient means to support key applications of mobile ad hoc networks such as teleconferencing and message dissemination. The recent growth of the World Wide Web has sparked new research into using the Internet for novel types of group communication, like multiparty videoconferencing, multiplayer online gaming and real-time push-based information delivery systems such as stock quote services. These applications require multicast to minimize the volume of network traffic they generate. Multiparty communications have recently become the focus of new developments in the area of applications [2].

1.1 Goal

The goal of this paper is to secure group communication using multicast key distribution scheme for secure and efficient group key management in ad hoc network. Group communication is one of the most important services in a mobile ad-hoc network, in which data confidentiality and integrity is realized by encrypting data with group key. In order to meet the forward- secrecy membership and the backward secrecy polices, any change in the group membership will induce group rekeying. So how to update group-key securely and efficiently is a crucial problem in secure group communication.

1.2 Reading Roadmap

This paper starts with this section, which gives a brief introduction, and goal of this paper. **Section 2** presents the security issues in Multicast Routing. The improved model scheme (SGCMKD) is presented in **Section 3**. The Multicast Key distribution Scheme is presented in **Section 4. In Section 5**, we discuss the experimental results discussion. Finally, conclusions are given in **Section 6**.

2. SECURITY ISSUES IN MULTICAST ROUTING

The goal of multicast security is to ensure that the source of the multicast stream and the group of multicast recipients communicate securely. This can be achieved through the authentication of the message origin by the recipients and through confidentiality and integrity preventing disclosure and modification of the messages by any party other than the members of the multicast group. These services typically require the establishment of a security association between the source and the recipients of the multicast channel. The security association defines the set of cryptographic keys and algorithms used for each service. The establishment of a security association for a multicast channel is inherently more complex than with unicast. In the unicast, a security association is static in that the source, the recipient, and the dataflow do not vary during the association. In a dynamic multicast group, a session are ever-evolving entities as recipients can be added to or removed from the recipient group through join and leave operations, respectively. Here the group membership is dynamic, i.e., nodes leave and join the group continuously. Therefore, an efficient re-keying mechanism is mandatory to ensure a robust multicast system [3,4].

3. IMPROVED MODEL SCHEME (SGCMKDS)

3.1 Introduction

As in any distributed system, in ad hoc networks the security is based on the use of a proper key management system. As ad hoc networks significantly vary from each other in many aspects, an environment-specific and efficient key management system is needed. Key management plays an important role enforcing access control on the group key (and consequently on the group communication). It supports the establishment and maintenance of key relationships between valid parties according to a security policy being enforced on the group.

Secure group communication systems typically rely on a group key, a secret shared by all members of the group. Privacy is provided by encrypting all data with the group key. The key management system controls access to the group key, ensuring that only authenticated members receive the key. To facilitate this process, the key management system also manages a set of auxiliary keys, while are shared by some subset of group members, and individual keys, while are assigned one per group member. When group membership changes, it becomes necessary to change the group key and some of the auxiliary keys to provide continued privacy. This operation is known as rekeying. The problem is to perform rekeying in a scalable and secure fashion.

3.2 Overview.

The Secure Group Communication using Multicast Key Distribution Scheme (SGCMKDS) works by creating virtual group throughout the network. Each group has a group-head (GH) and the other nodes of the group are member nodes. With the help of the group-heads, the nodes authenticate each other and exchange their public key in a secure manner. The grouping technique employed here is a modified version of Basagin’s clustering algorithm [12] where the weight for group head selection is based on the degree of node (i.e. number of neighbors around the node) and node’s identification number. Apart from these parameters, the member nodes assess trust of the group head and the Key Management Scheme is adopted.

The nodes in the network are distributed. The nodes start to exchange neighborhood information, which is the prime activity at neighbor discovery phase and then form groups. Based on degree of the node (number of one hop neighbors), node’s Id and trust, a group -head for the group is selected. The group heads broadcast neighborhood messages, which are retransmitted to neighboring group by the member nodes. A global knowledge about neighboring groups in the network is obtained. Then trust between the group head is established. Nodes select the partners for secure communication from the global neighborhood (i.e. the nodes within a group are part of local neighborhood and groups are interlinked to form a larger network. Then the nodes in the network are said to be members of the global neighborhood) only. This ensures that communication exists only between connected nodes which will increase various performance factors. The group head are used as helper nodes for authenticating the partners and session keys are created by the partners and exchanged. The sequences of activity for successful group key management are:

1. Exchange neighborhood information to one- hop neighbors.
2. Partition the network into different group.
3. Perform Group head selection based on the degree of node, node’s id and trust on the group-head by the member nodes.
4. Trust establishment between group-heads
5. Key management activities.

Steps 1, 2 and 3 are activities that are to be performed during the group formation phase. Step 4 the trust establishment between the group-heads happens and in step 5 the nodes generate their public, private and self certified certificate and try to authenticate each other using the group-heads for secure communication.

3.3 Group creation process

The group formation technique is a modified version of Basagin’s cluster formation technique [12] where the weight for the selection of group head is the degree of the node, and node identification number, which is normally the node‘s IP address and trust of node which is included as the third parameter. For trusting the neighbor the method of trust is derived from the human behavior model. Humans mostly trust their neighbors if they cooperate for specific task or respond enthusiastically for all request. To trust members in different neighborhood, humans normally look for trustable person in that neighborhood. This response from trustable party is valued high and at times it can militate the true quality of that particular neighbor. Likewise the trust model is mapped to real world human society, as nodes need cooperation in packet forwarding and verifying trust of other nodes. The neighborhood in the real world is equivalent to the group in ad hoc network. The trust evaluation within the group is cheaper and more credible. The neighboring group head plays the role of the trustable person in other neighborhood. For implementation issues, the trust assessment parameters are limited to correctness in the information disclosed to the neighbors. This information is crosschecked with response from different parties. Nodes are trusted if only n responses arrive, where n represents half the number of the enquiry request sent. The group formation technique is built on top of a modified version of MAODV protocol.

Algorithm 1: Group Head selecting

```
Step 1:   The weight value of each node with hello message is
          broadcasted to adjacent nodes. The delivery range of
          each node is not more than 2-hop.
Step 2:   After Step 1, we collect all weight values of nodes and
          select the largest one to be the Group head (GH).
Step 3:   check GH trust of node
           3.1  if node high trustable then set Head Elect = High
                Send HeadElect message to neighbors Wait for
                Head message.
           3.2  Else if node high not trustable then delete node
                high from array Node List (NLn) store high in
                blacklist array.
Step 4: Other nodes will register to the selected GH and send all
          information to it.
```

3.4 During the head election process

Nodes assess the trust of other nodes during election process when a Cluster Head Elect message is received or chooses a Cluster

Head Elect or receives Head message. Under such events, nodes need to check the trust. The nodes retrieve the neighbor list for the node whose trust is to be evaluated. The common neighbors to both the nodes are left and nodes not in current node and present in neighbor list are the ones to be verified. So neighborhood status is enquired from those nodes. If at least half the responses are received then the nodes are trustable.

4. KEY MANAGEMENT SCHEME

4. 1 Overview of the Scheme

The system gets ready for key management phase after the successful formation of group, exchange of member list between the group heads and overall trust assessment has made. The nodes create self-certified certificates and store the certificate in their corresponding cluster heads. Algorithm 2 discusses the proposed key management scheme.

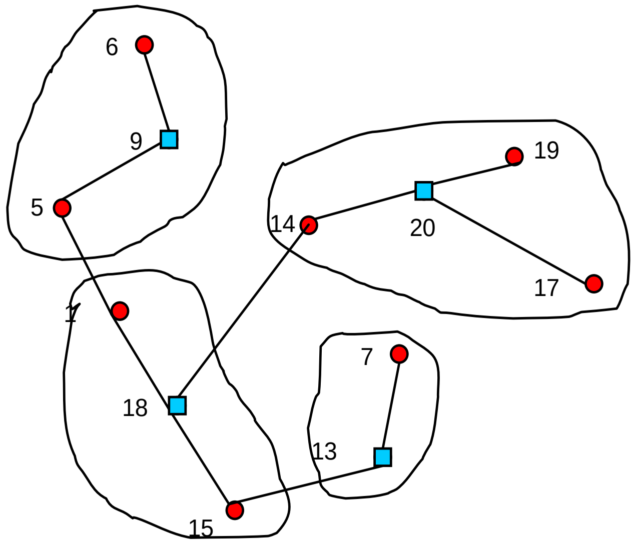


Fig. 1 Ad hoc network partitioned into different groups

In Fig. 1 Nodes 9, 13, 18 and 20 are group heads. Suppose node 14 selects node 7 as its destination node. Node 14 checks the blacklisted node list for nodes 7 and 13. if not listed then checks the trust with node 9 if group can be trusted, node 14 sends request for public key to node 7 and sends request for node's 7 certificate to 13. Node 13 obtains the public key of node 14 from 9 and encrypts the message for verifying the certificate using 14's public key. Node 14 decrypts and verifies the certificate. Likewise the same happens at 7's end. If both certificates are verified, nodes generate session key and exchange to communicate in secure manner. Based on the activities the key management scheme can be divided into three phases.

- Certificate Management
- Authentication with helper nodes
- Creation and Exchange of Session Key

Algorithm 2 Key Management scheme

```
Input: source node s, destination node d, source node's group head $s_{gh}$,    destination node's group head $d_{gh}$
Output: successful key exchange or failed attempt.
Parameters used: global neighbor list
1. check if d or $d_{gh}$ not in blacklisted node list.
2. request $s_{gh}$ regarding the trust of d and $d_{gh}$.
3. if (group head is trustable)
     Request public key from destination node d. The destination node checks the trust of source node group
     The group heads $s_{gh}$ and $d_{gh}$, exchange the public keys of s and d.
4. The source node s request destination node's certificate from $d_{gh}$
5. The $d_{gh}$ replies by sending the certificate and encrypts the verification message with the source nodes public key.
6. Tthe source node decrypts the message and verifies the certificate.
7. The same process takes place in the destination node's end also.
8. if the certificates are valid, then generate session key and exchange using Diffe-Helman key exchange protocol.
```

4.2 Multicast Key Distribution in ad hoc network

In a multicast group, there will be a trusted entity termed as the Group Head (GH). GH will be responsible for generating and updating the cryptographic keys. A sender in the group will use a key that is stored by the GH and is known to the entire group. This key is called Session Encryption Key (SEK). All the data the sender transmits is encrypted using the SEK and so the intended cluster members will then access the data by performing decryption using the SEK. However, every time a member leaves the group or joins the group then face a security threat. When a member leaves the group need to provide forward security in that need to change the SEK so that the departed member cannot access future group communications.

When a member joins the group, need to provide backward security in that need to change the SEK so that the new member can't access any of the past group communications. Hence updating the SEK is very essential when there is any change in group membership. It is also a healthy practice to do a periodic update of the SEK even when there is no membership change [6].

In order to distribute the updated SEK to the cluster members the GH would now need another set of keys. This set of keys is termed as Key Encryption Keys (KEK). Every cluster member will have a KEK that it shares only with the GH so that the GH could transmit the new SEK securely to it.

4.3 Rekeying /Key Update

Keying material must change each time the set of users in a multicast group changes. The group key must be revoked and redistributed to all the remaining nodes in a secure, reliable, and timely fashion whenever group membership changes, particularly when a node leaves the network. The keying material shared by the members of the multicast security association should be updated in order to fulfill the following conditions. i) When a user **JOINS** the group, he should not have access to past keying material. ii) When a user **LEAVES** the group, he should not have access to future keying material.

4.4 Vulnerability of conventional rekeying mechanism:

Consider the scenario, as shown in Fig.2, of a group consisting of a key server s and users u1…. u8. The server is responsible for initiating and maintaining the group in the presence of user joining and leaving. The keys are organized as a key tree, where the leaves are the users and the inner nodes are the keys. Moreover, each user holds the keys corresponding to the inner nodes on the path starting from the parent of the user and ending at the root. For example, in Fig. 3, user u1 holds keys K_1, K_{123}, and $K_{1\text{-}8}$ where $\mathbf{K_{1\text{-}8}}$ is the group key that can be used to encrypt the communications within the group [7].

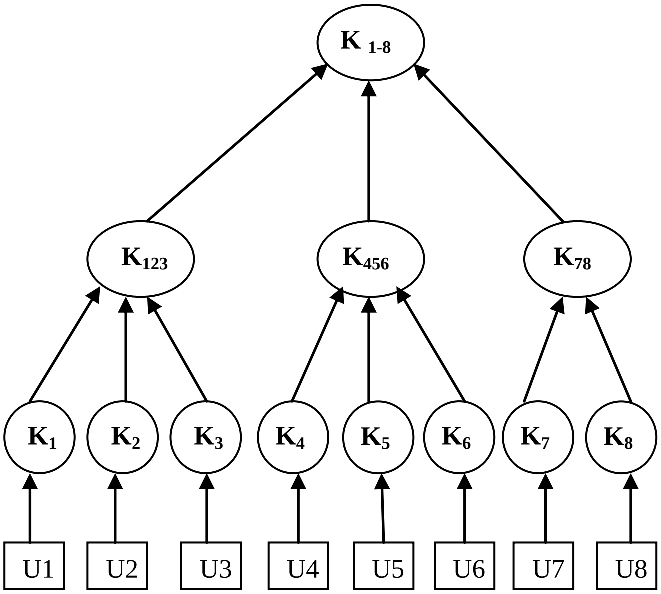


Fig. 2 Initial key trees

Key tree after u_9 joined: A dynamic group entertains the joining and leaving of some users. In order to maintain secure communications, each join or leave would require the key server to change some keys that also need to be securely distributed to certain users (via some rekeying messages). After granting a join request from user *u*9, server s shares a key k_9 with user u9. Besides, certain keys need to be changed and sent to the corresponding users. As shown in Fig. 4, in order to prevent u9 from accessing past communications, the key k_{78} and $k_{1–8}$ are changed to k_{789} and $k_{1–9}$, respectively. Moreover, k_{789} and $k_{1–9}$ need to be securely sent to user's u7, u8, and u9. One efficient way to achieve this is group-oriented rekeying strategy. Following is the notation used for group oriented rekeying strategy: x → {y1, …., yn} : $\{z\}_w$ denote that *x* sends the users y1, . . . , yn (via multicast or unicast) the encryption of plaintext *z* using key *w*, namely the ciphertext *$\{z\}_w$* Group oriented rekeying strategy for this group change:

$$s \rightarrow \{u1, \ldots, u8\} : \{k_{1–9}\}k_{1–8}\,,\ \{k_{789}\}k_{78}$$
$$s \rightarrow \{u9\} : \{k_{1–9}, k_{789}\}k_9$$

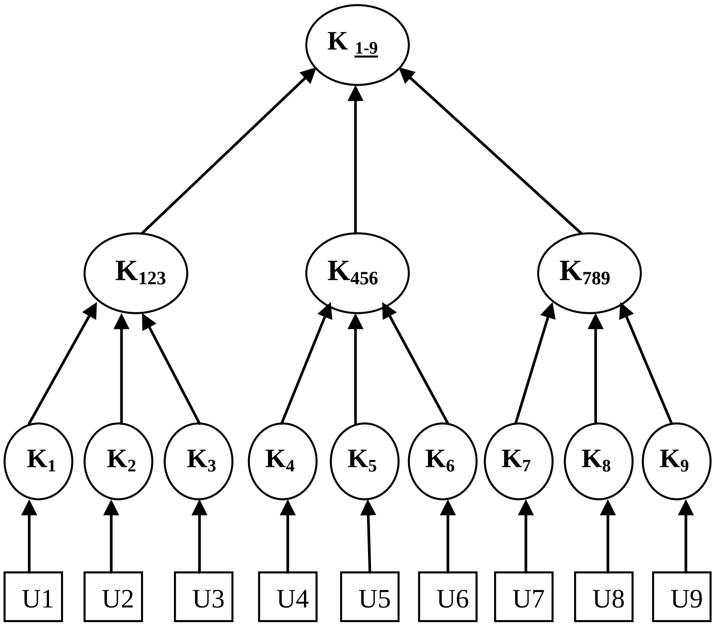
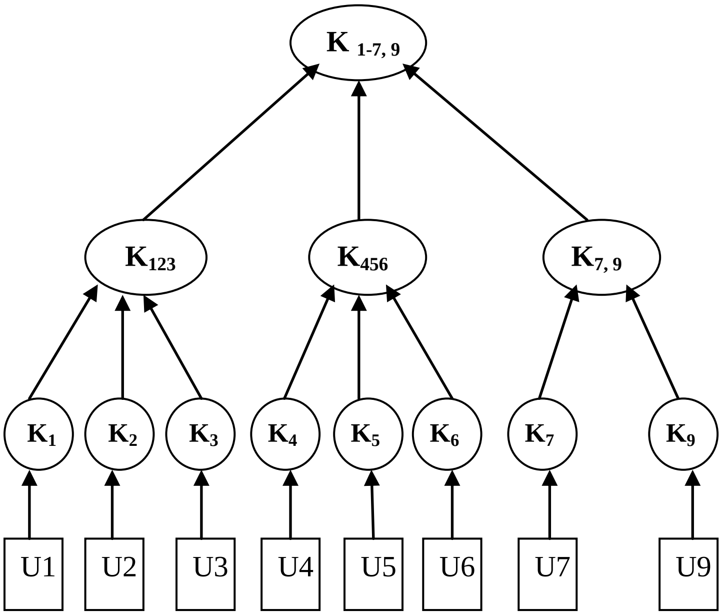


Fig.3 Key tree after U9 joined

The server s forwards to all users u1 to u8 the new encrypted key $k_{1–9}$ using the previous group key $k_{1–8}$. The server has to send the changed inner group key from k_{78} to k_{789} to users u7 and u8 encrypted using the key k_{78} which is the previous key held by the group prior to group change. The server has to send keys k_{789} and $k_{1–9}$ to the new user u9 using its shared key k_9 with server



$$s \rightarrow \{u9\} : \{k_{1–9}, k_{789}\}k_9$$

Fig. 4: Key tree after U8 left

Key tree after u_8 left: Now, suppose u8 leaves. To prevent u8 from accessing future communications. Server s needs to change the keys $k_{1−9}$ and k_{789} to new keys $k_{1–7,9}$ and $k_{7,9}$, respectively. Moreover, $k_{1–7,9}$ and $k_{7,9}$ need to be securely sent to users u7 and u9, and $k_{1–7,9}$ needs to be securely sent to users u1,…, u6. Group oriented rekeying strategy for this group change: s→{u1,… ,u7, u9} : $\{k_{1-7,9}\}k_{123}$, $\{ k_{1-7,9}\} k_{456}$, $\{ k_{1-7,9}\} k_{7,9}$, $\{ k_{7,9}\}k_7$, $\{ k_{7,9}\}k_9$.

4.5 Attack on Conventional Rekeying strategy

Suppose now an adversary compromises user u9. It is true that the adversary is always able to obtain the current group key $k_{1-7,9}$, no matter how the group rekeying scheme works. However, the adversary who has recorded the network traffic is also able to obtain the group key k_{1-9}, because it can decrypt the message incurred by the event that u9 joins the group.

As a consequence, the adversary can decrypt both the communications encrypted using group keys k_{1-9} and $k_{1-7,9}$. This is in sharp contrast to the desired property that the adversary can decrypt only the communications encrypted using group key $k_{1-7,9}$.

The above attack is not fundamentally related to the group-oriented rekeying strategy, or to the fact that u8 – the sibling of the newly joined user u9 – leaves the group or u9 is a recently joined node. For example, suppose group dynamics is incurred by some users belonging to {u4, u5, u6, u7, u8, u9}, then it is possible that k_{123} is always used to encrypt the new group keys so that u1, u2, and u3 can obtain them. As a consequence, u1, u2, and u3 are the “most valuable” users from the adversary’s perspective of view, and compromising any of them will enable the adversary to recover all the past and current group keys. It is clear that the past group keys, which were ever encrypted using any of the keys held by a user that is being corrupt, are exposed. Multicast routing should be able to route message to any number of new recipients in the multicast group as they join or leave [8,9].

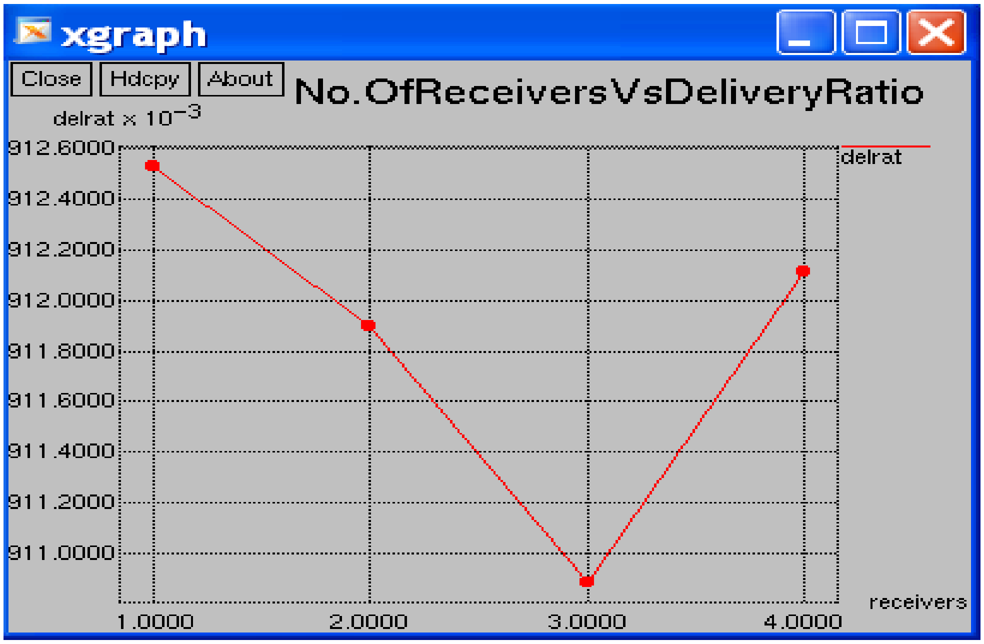


Fig. 5 Number of Receivers Vs Delivery Ratio

5. EXPERIMENTAL RESULTS AND DISCUSSION

5.1 Simulation Results

We run several simulations under Linux, using the network simulator NS2 version ns-allinone-2.26. The simulation environment is composed of:

- area: 500*500 meters.
- number of nodes 50 - 100.
- simulation duration: 1000s.
- physical/Mac layer: IEEE 802.11 at 2Mbps, 250 meters transmission range.
- mobility model: random waypoint model with no pause time, and mode

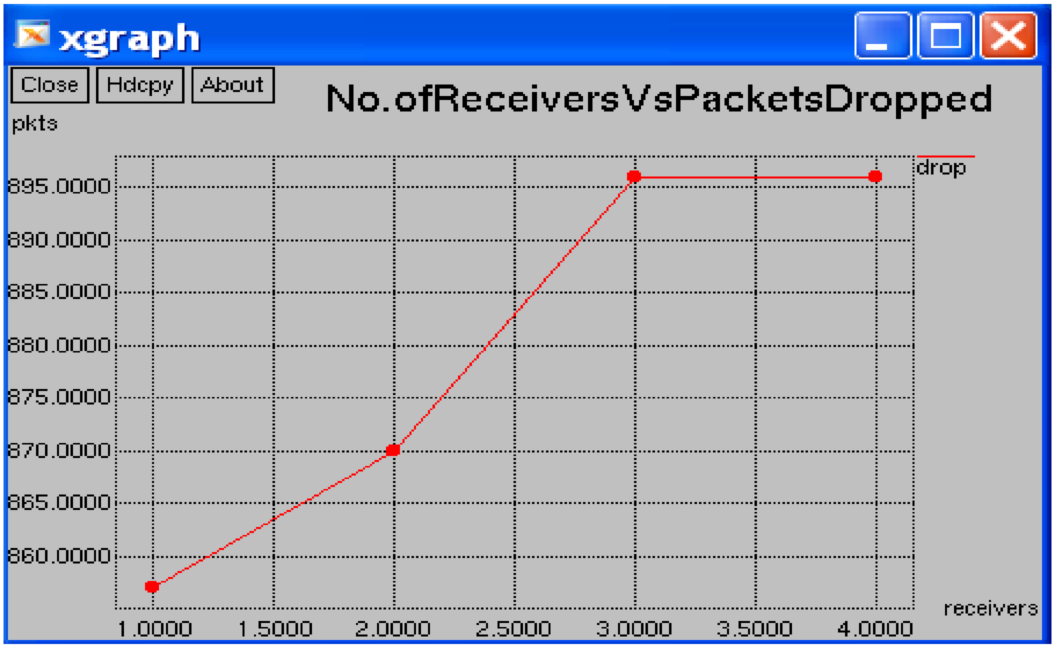


Fig. 6 Number of receivers Vs Packets Dropped

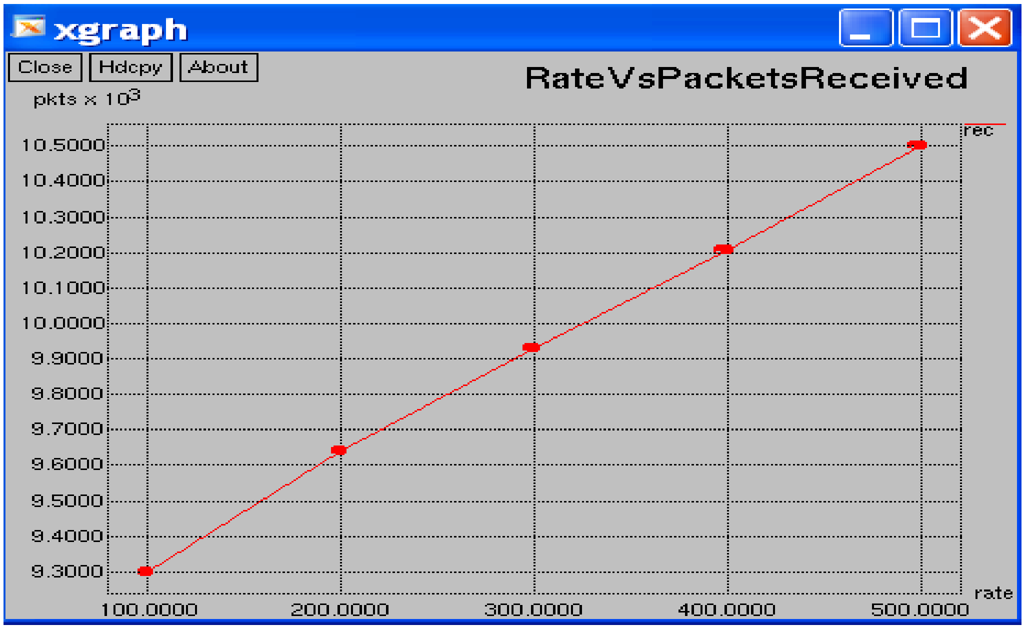


Fig. 7 Number of Receivers Vs Packets Received

- movement speed 0m/s, 1m/s and 10m/s.
- Using routing protocols are AODV and MAODV under NS2.26.

5.2 Number of Receivers Vs Delivery Ratio

The fig.5 shows that the delivery ratio is going down while the no of receivers are more in the network. The packet delivery ratio is the ratio of the number of packets actually delivered to the destination nodes.

5.3 . Number of receivers Vs Packets Dropped

The fig.6 shows that the packet dropped is high when the no of receivers are also high .In the simulation results the number of receivers rate is started from 1.0000, 1.5000, 2.0000, 2.5000, 3.0000 and the packet dropped is used in the unit of 860.0000, 865.0000 870.0000, 875.0000, 880.0000, 885.0000, 890,0000, 895.0000.

5.4 Number of Receivers Vs Packets Received

The Fig. 7 shows that Accuracy of multicast delivery is calculated as ratio of the number of multicast group members which actually receive the multicast packet, and the number of group members which were supposed to receive the packets.

6. CONCLUSION

Group key management in ad hoc network is challenging task. In this paper, presented and analyzed a secure group communication using multicast key distribution scheme in ad hoc networks. This SGCMKDS consists of key tree based group key distribution. Group communication is one of the most important services in an ad hoc network, in which data confidentiality and integrity is realized by encrypting data with group key. In order to meet the forward security membership and backward security polices, any change in the group membership will induce group rekeying. So how to update group-key securely and efficiently is a crucial problem in secure group communication.

6.1. Future Direction

In the cluster head selection process, the metric taken for the selection are degree of neighbor, id and trust. Apart for these parameters, power level, battery power, active mode power consumption, sleep mode power consumption, standby mode power consumption, coverage area and transmission mode could be taken as additional parameter to provide better cluster stability.

In future discuss the different types of nodes leaving a subgroup. Sometimes cluster head leaving in cluster. To select the new cluster head based on the above mentioned procedure and send the information of all the cluster member nodes under them to the new cluster head.

7. REFERENCES

[1]. Guangming Hu, Xiaohui Kuang, and Zhenghu Gong, “A Cluster-Based Group Rekeying Algorithm in Mobile Ad Hoc Networks”, Springer-Verlag Berlin Heidelberg 2005 ICCNMC 2005, LNCS 3619, pp. 344 – 353, 2005.

[2]. T. Kaya, G. Lin, G. Noubir, A. Yilmaz , “Secure Multicast Groups on Ad Hoc Networks” Proceedings of the 1st ACM Workshop Security of Ad Hoc and Sensor Networks Fairfax, Virginia © 2003 ACM.

[3]. Tzu-Chiang Chiang and Yueh-Min Huang “ Group Keys and the Multicast Security in Ad Hoc Networks “ Proceedings of the 2003 International Conference on Parallel Processing Workshops (ICPPW’03) 1530-2016/03 2003 IEEE.

[4]. Justin Goshi and Richard E. Ladner , “Algorithms for Dynamic Multicast Key Distribution” ACM Journal of Experimental Algorithmics, Vol. 11, Article No. 1.4, 2006, Pages 1–37.

[5]. Sandro Rafaeli and David Hutchison, “A Survey of Key Management for Secure Group Communication” , Lancaster University, ACM Computing Surveys, Vol. 35, No. 3, September 2003, pp. 309–329.

[6]. Nen-Chung Wang , Shian-Zhang Fang , “A hierarchical key management scheme for secure group communications in mobile ad hoc networks”, Science Direct, The Journal of Systems and Software 2007.

[7]. Bing Wua,_, Jie Wua, Eduardo B. Fernandez, Mohammad Ilyas, Spyros Magliveras, “Secure and efficient key management in mobile ad hoc networks”, Journal of Network and Computer Applications, Science Direct, www. Elsevier.com, July 2005.

[8]. G. Noubir, F. Zhu, A. H. Chan, “Key Management for Simultaneous Join/Leave in Secure Multicast’ Northeastem University, USA, ISlT 2002, Lausanne, Switzerland, June 30 -July 5,2002, IEEE.

[9]. Ozkan M. Erdem, “ EDKM: Efficient Distributed Key Management for Mobile Ad Hoc Networks. Oregon State University, 0-7803-8623-W04/ IEEE.

[10]. Ling Luo, Rei Safavi-Naini, Joonsang Baek and Willy Susilo, “Self-organised Group Key Management for Ad Hoc Networks”, *ASIACCS* ’06 March 21 - 24, 2006, Taipei, Taiwan. Copyright 2006 ACM.

[11]. Jun Li , Guohua Cui, Xiaoqing Fu, Zhiyuan Liu, Li Su, “A Secure Group Key Management Scheme in Mobile Ad Hoc Networks”, Huazhong University of Science and Technology Wuhan , China, 2005 IEEE.

.