AN ENERGY EFFICIENT KEY MANAGEMENT AND AUTHENTICATION TECHNIQUE FOR MULTICASTING IN AD HOC NETWORKS

PAVITHIRA LOGANATHAN[1] AND DR.T.PURUSOTHAMAN[2]

[1]Assistant Professor, Department of Computer Science

CMS College of Science and Commerce, Coimbatore, India

[2]Associate Professor, Department of Computer Science and Engineering,

Government College of Technology, Coimbatore, India.

E-mail :[1] pavithira1080@gmail.com

ABSTRACT

In Mobile Ad hoc Networks (MANETs) multicasting, when an attacker inserts spurious packets into the network any time, its neighbors can drop these packets when proper authentication is performed. Hence, efficient authentication with key management is required for multicasting in MANET. Also multicast key management in ad hoc networks involves energy expenditure in key distribution and rekeying. In this paper, we propose a trust authority based key management and authentication technique for multicasting in ad hoc networks. Initially we construct an energy efficient topology aware key tree which mainly aims to reduce the re-keying load by pre-processing the joining members during the idle re-keying interval. Key management is processed based upon Diffie-Hellman key pair and RSA secret public key pair. A trust authority establishes public key certificates for each group member by signing the public key with its secret key. From the simulation results we show that this key management guarantees key authentication, enhances fault-tolerance and protects the tree from impersonation attacks.

Keywords: *Mobile Ad hoc Networks (MANET), Key Management, Authentication Technique*.

1. INTRODUCTION

1.1 Mobile Ad hoc Networks (MANET)

A mobile ad hoc network is a self-organizing system of mobile nodes that communicate with each other through wireless links without infrastructure or centralized administration such as base stations or access points. Nodes in MANET can either work as hosts or routers to forward packets for each other in a multi-hop fashion. The application of MANETs includes military battlefield, emergency rescue, vehicular communications and mining operations in which there is no infrastructure existence [1]. A communication session is attained by single-hop transmission if the recipient is in the transmission range of the source node, or by relaying via intermediate nodes. Hence MANETs are also termed as multi-hop packet radio networks. But the transmission range of each low-power node is limited to each other’s closeness level, and out of range nodes are routed via intermediate nodes [2].

1.2 Multicasting in MANET

Multicasting is the transmission of packets to group of hosts which is identified by a single destination address. This is proposed for group-oriented computing, where the membership of a host group is dynamic which means that the hosts can either join or leave groups at any time. The limit is not assigned for location or number of members in a host group. A host can be a member of one or more group at particular time duration and host need not be a member of a group to forward the packets to members in the group [2]. The merit of multicast is that it allows the desired applications to service many users without overloading a network and resources in the server. [3]

1.3 Security Issues in Multicast

Security is needed to transmit data via insecure network. The multicast approach is vulnerable than unicast since the transmission takes place through multiple network channel. The challenging issue arises in multicast for its dynamic character. The users' activity of leaving and joining the groups makes the issue more difficult in large scale systems. The following attacks can be launched on multicasting: Resource Consumption Attack, Rushing Attack, Blackhole Attack, Grayhole attack, Wormhole attack and Selfish Nodes [7][8]9]. So there is a need to provide forward secrecy and backward secrecy.

Forward secrecy: Whenever a member leaves the group, the member should not hear the further conversation in that group which is termed as forward secrecy.

Backward secrecy: Whenever a new member joins the group, the member should not be able to access the previous conversations in that group which is termed as backward secrecy.

Real time as well non-real time applications are contained in multicasting. In case the issues are taken into account, it could result in severe bottleneck specifically real time application like VoIP systems. Hence it is necessary that a security scheme in a multicast environment must be secure and efficient for minimizing bottlenecks [3] .

1.4 Schemes for Secure Multicasting

- Centralized scheme: The group key management is carried out using Group controller (GC) and there are fewer burdens for the users of the group [5].
- Distributed scheme: The group key management is performed by user and hence there are more burdens over the users. [3]

The deployment of MANETs in various environments can result in varying requirement and constraints for nodes operating in such environments. MANETs for key distribution is classified as follows.

- Over-layered oriented: Sufficient trusted-entities, special nodes or hubs exist, accessible from all nodes in the network.
- Flat oriented: Few, if any trusted special hubs or nodes exist, that may not be accessible to all nodes at all times, high-level of self-organization is thus expected.-
- Military oriented: Heterogeneous environment and nodes, assumed to be a combination of the above frameworks. [6]

1.5 Key Management

Creating, distributing and updating the keys constituting a basic block for secure group communication applications, is termed as key management. The main aim of key management is secure distribution of keying material. [4] The security services mainly focus on encryption using Traffic Encryption Keys (TEKs) and re-encryption using Key Encryption Key (KEKs). Each member holds a key to encrypt and decrypt the multicast data. In order to meet the above requirements, the key has to be updated and distributed to all group members whenever a member joins and leaves a group.

1.6 Problems and Proposed Solution

Re-keying is the process by which keys are updated and distributed to the group members. In order to ensure that a new member cannot decrypt the stored multicast data and to prevent a member leaving from eavesdropping future multicast data, re-keying is necessary in secure multicast communication [4].

The nodes have limited battery capacity in ad hoc wireless networks. The energy spent by each node for data transmission is valuable and must be utilized for network management operations of key distribution. Hence incorporating energy as design topology, key trees have to be deigned.

Key management results in energy expenditure while updating and distributing the keys. For reducing the energy expenditure of the key distribution, energy efficient approaches need to be considered [5].

In this paper, we propose an energy efficient key management and authentication technique based on a trust authority, for multicasting in ad hoc networks. We construct an energy efficient topology aware key tree which mainly aims to reduce the re-keying load by pre-processing the joining members during the idle re-keying interval. Key management is processed based on Diffie-Hellman key pair and RSA secret public key pair. In the authentication technique, a trust authority establishes public key certificates for each group member by signing the public key with its secret key.

2. RELATED WORK

Shaobin Cai et al [14] have proposed group-based key management scheme (GBKM). Depending upon the relativities of missions and numbers of shared keys among the missions, the key pool is divided into some sub-pool using GBKM. The nodes of same missions help in the communication in ad hoc sensor networks. The probability of the sharing between the communicating nodes and a key can be improved and probability of decrypting a shared key is reduced by the group-based key management.

Bo Rong et al [15] have proposed a Key Management for Pyramidal Security Model. For a mobile ad hoc network, a special multicast scenario of multi-security-level information broadcast can be protected. An integrated tree key graph scheme is proposed for an efficient key management solution to the pyramidal security model.

Feng He et al [16] have proposed a novel secure routing protocol S-MAODV based on MAODV. Trusted computing technology, secure node authentication and security indicator bit-set mechanism combine to form the S-MAODV. A trusted third party (TTP) is not required for the SMAODV since it is an anonymous protocol.

Mahalingam Ramkumar et al [17] have proposed a novel key management scheme, RPS- Random Preloaded Subset key distribution. RPS in particular is an n-secure r-conference key pre distribution scheme. The session keys are obtained from the shared keys using the symmetric crypto primitives for one-way functions. This happens to be a computational complexity of RPS.

D.Suganya Devi et al [18] have proposed a new efficient cluster based multicast tree (CBMT) algorithm for secure multicast Communication. The multicast version of destination sequenced distance vector (MDSDV) routing protocol is used by the source nodes to collect one hop neighbors. When these neighbors form clusters, a local controllers of the created clusters is elected from each node having child node. The defects caused by node failure can be endured.

Yun Zhou et al [13] have proposed a novel authentication scheme MABS. It is absolutely durable to packet losses since the correlation among packets can be dealt efficiently using the DoS attack. Correlation among packets can be eliminated by MABS-B so that perfect resilience can be provided to the packet loss. Latency, computation and communication overhead can be efficiently achieved using batch signature. Multiple packets can be authenticated simultaneously.

3. ENERGY EFFICIENT TOPOLOGY AWARE KEY MANAGEMENT

3.1 Grouping the Nodes

In the homogenous medium, the spending of energy can be reduced by key distribution by assigning common keys to members physically close. The transmission power for communication between nodes is a monotonically increasing function of the distance. Under the assumption that routing is optimally selected to minimize the total transmission power, nodes that are physically close have overlapping routing paths and will have common links in the path from the group head GH towards them. Hence they should also share common keys in order to receive the same key updates and reduce the energy expenditure of the key distribution.

We assume that the members are grouped according to their physical distance. Figure 1 and 3 illustrate the ad hoc network and corresponding routing tree with the minimum total transmission power, deployed in homogenous medium.

In Figure 1, node q is grouped with r, and node e is grouped with node i resulting in the physical distance based key tree. Similarly node u is grouped with node v and node f is grouped with node l in Figure 3.

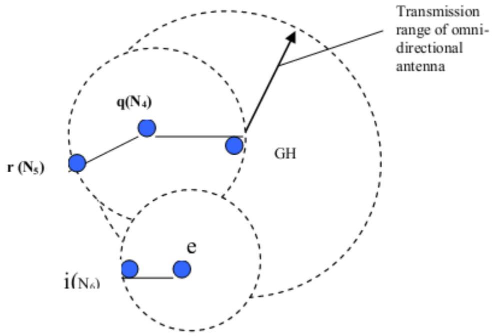

Figure 1: Physical Topology

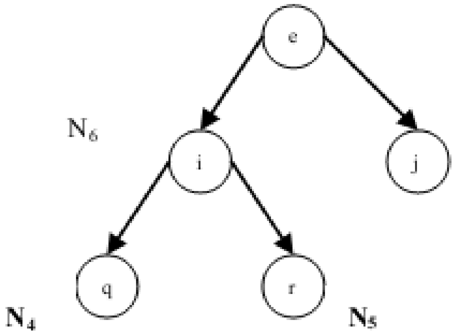

Figure 2: Key Tree Diagram for Topology in Figure. 1

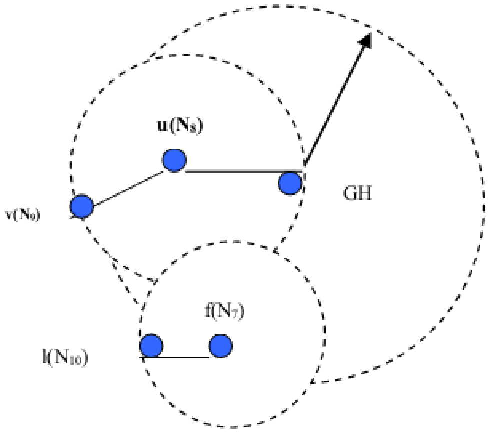

Figure 3: Physical Topology

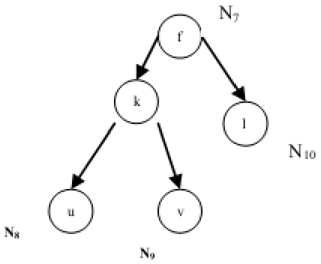

Figure 4: Key Tree Diagram for Topology In Figure 3

3.2 Temporary Key Tree Construction

The several proposed approaches in the past involve performing all re-keying steps at the beginning of every re-keying interval. This will result in high processing load during the update instance and thereby delays the start of the source group communication. Hence we propose an effective algorithm termed as T**emporary key tree construction algorithm**. This algorithm mainly aims to reduce the re-keying load by pre-processing the joining members during the idle re-keying interval.

The algorithm is as follows:

```
If TKT =empty, then
            Create a new TKT with only one latest
            member
Else
            Find the insertion node and add the latest
            member to TKT;
             If node= initiator, then          /*Elect the
            rightmost member under the sub-tree
            rooted the sibling of the joining node to be
            the initiator */
            Initiator starts re-key process and nodes
            are renewed.
            End if
End if
   If leaving node =empty, then
            add TKT to either the shallowest node
            (which need not be the leaf node) of KT
  Else
             Add TKT to the highest leave position of
            the key tree KT and remove
            remaining L_N-1leaving leaf nodes and
            promote their siblings;
            If node=initiator, then /*elect the
            rightmost members of the sub- tree rooted
            at the sibling nodes of the departed leaf */
             Initiator starts re-key process and nodes
            are renewed.
             End if
End if
            [TKT = Temporary Key Tree
            KT= Key Tree]
```

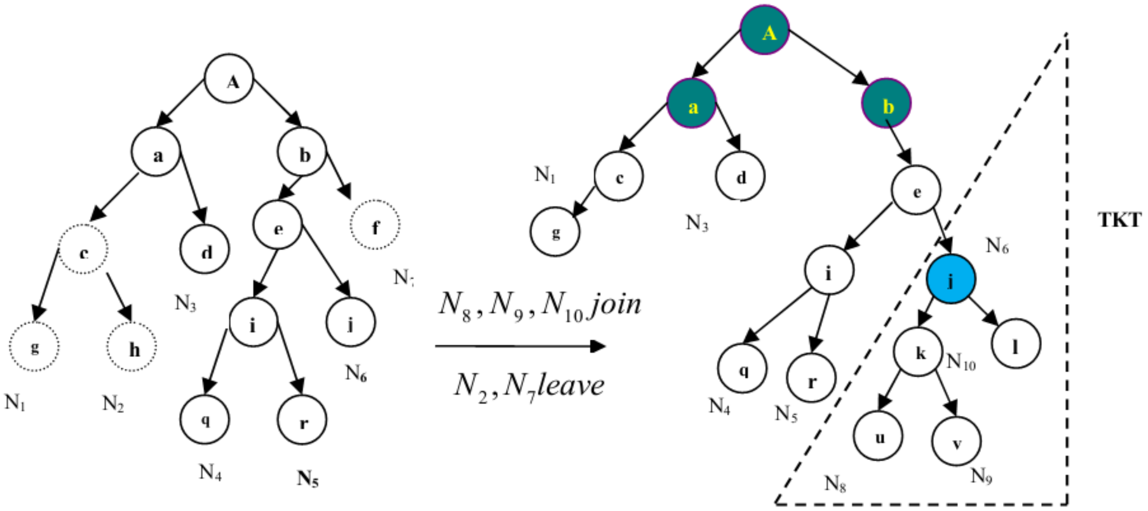

Figure 5: Temporary Key Tree Construction

The temporary key tree algorithm significantly reduces both computation and communication costs where there exist highly frequent membership events.

The re-keying process taking place in the Figure 5 is described as follows.

Here the members, N_8, $N_{9,}$ and N_{10} wishes to join the communication group while N_2 and N_7 wish to leaves the group.

i) The three new members N_8, N_9, and N_{10}, first form a TKT, N_{10}, in this case is elected to be the initiator.

ii) The tree TKT is added at the highest departed position, which is at node 6. Also the blinded key B_{key} of the root node TKT which is B_{key6} is broadcasted by N_{10}

iii) The initiators N_1, N_6, N_{10} are elected. N_1 renews the secret key k_2 and broadcast the blinded key B_{Key2}

iv) Finally all members can compute the group key. [19]

4. TRUST AUTHORITY BASED KEY MANAGEMENT

In this section, trust authority based key management is processed. Diffie-hellman key pair and RSA secret-public key pair is the two key pairs used in each group member. This Diffie-hellman key pair {Sk_{wi}, Pk_{wi}} generates the group key and RSA secret-public key pair, *{A_i,B_i}* generates source authentication. The RSA key pair is not present in the non-leaf nodes L.

- The existing group members are given public key certificates using an offline trust authority (TA).
- Each group member *Wi* are given a public key certification using a RSA secret-public key pair *{Sk, Pk}*, by signing *Wi's* public key with its secret key Sk.
- The set of nodes in union of all clusters which contains one or more leaf nodes of subtree which is rooted at W_i sibling node is known as the co-ordinator set CS(Wi).
- Co-ordinator set obtains W_i's public key certificate $<W_i,Pk_{wi},B_i>_{Sk}$.

The shamir's threshold sharing scheme is used by W_j in order to distribute secret share Sk_j of secret key S_k to each group member. The members of the co-ordinator sets create partial public key certificates using this threshold sharing. The original certificates of W_i are verified by the node W_j in $CS(W_i)$ and re-encrypts it with Sk_j in order to create partial certificates. The group member's public key certificate can be offered by any k members in the co-ordiantor set of a given group member by group signing of certificates.

4.1 Certificate Generation & Distribution

Initially TA randomly selects a (t-1) degree polynomial $f(x) = Sk + e_1.z + ... +e_{t-1}. Z^{t-1}$
Such that the shared secret is f(0) = Sk. Each group member obtains a secret share SS_{Wi} = (f(Wi)mod w). For any k group members {W_1,W_2,…..W_k} lagrange's interpolation yields

$$Sk = \sum_{i=1}^{k}(SS_{Wi}\cdot G_{Wi}(0)) = \sum_{i=1}^{r} Ski(\mod w)$$

The certificate C for any node is served by the node's *co-ordinator set*, with each member in that co-ordinator set providing a partial certificate C^{Ski}. With any r partial certificates, the requesting member can compute the valid certificate as

$$C^{Sk1}.C^{Sk2} . . . . C^{Skr} = C^{((\sum_{i=1}^{r} Ski)} =C^{Sk}$$

Thus, these r members can work like a trusted authority, and jointly offer the certificate.
The co-ordinator sets are created as per figure 6.

Group members are allotted unique member ID initially. They are arranged in ascending order along with the leaf node of the key tree. Group is spitted into k-member clusters in order to define the co-ordinator sets. Secret key shares are distributed to all the group members and the stored public key certificates in the trusted authority are distributed to the appropriate co-ordinator sets. The new members joining the group are initialized when the TA works offline. When the function is fully distributed to appropriate co-ordinator sets, TA is not required for providing key authentication service to the renewed RSA keys.

4.2 Secret share Updating

The Diffie-hellman keys are updated during a session or in prior to a session as it is selected as a subsidizer for each group member. The sender's RSA signature guarantees the source authentication of the updated blinded keys. The compromised secret shares can be nullified by the proactive secret share update algorithm which updates the system secret shares periodically.

If two nodes n1 and n2 is on a co-path, n1 is capable of determining the blinded key of another node n2. These two nodes can be considered as a set of siblings of each node in the key path of n1. There is no necessity to send the updated blinded key to the entire group. Only a small subset of a group is given the updated keys. Group keys are generated using Diffie-Hellman key exchange in a key tree where only the leaf nodes of the subtree rooted at *ni*'s sibling are required by the blinded key. Efficiency and key authentication can be improved by the *ni's* co-ordinator set. The blinded keys are sent to the co-ordinator sets alone and it has to respond to pubic key requests, and to the threshold cryptographic scheme which provides key certificates.

4.3 Joining Process

When any node wishes to join the communication group, it sends a signed join request to the group. When the other group members receive this request, they determine the insertion node in the tree. They also select a subsidizer which initiates the joining process. Each group member adjusts the clusters in its key tree by adding the new node to the smallest cluster adjacent to the insertion point, or to the cluster on its right one in case of a tie.

We consider the tree constructed in figure 6.

- When the nodes n_9, and n_{10} wish to join the communication group, they send a signed join request to the group.
- They join at the insertion point n8 to form the key tree.
- The node n8 is also selected as the subsidizer which computes the new group key.
- The blinded keys f and b are updated along the paths to their co-ordinator sets *f{n4,n5,n6,n12}* and *b{n1,n2,n3,n11}*. The messages are signed by n8 along with its certificate.
- The co-ordinator sets request the subsidizer n_8's certificate to verify the updated blinded keys received.
- Each member W*j* of the n_8's co-ordinator set creates a new partial share *SS'j* of the secret key *Sk*, and forwards it to n_9 and n_{10}, which combine them to obtain their new secret share *Skn*+1.
- The nodes n_9 and n_{10} also send their signed public key certificate to the members of their co-ordinator set C*S*n_8, and get the blinded keys needed for generating the group key.

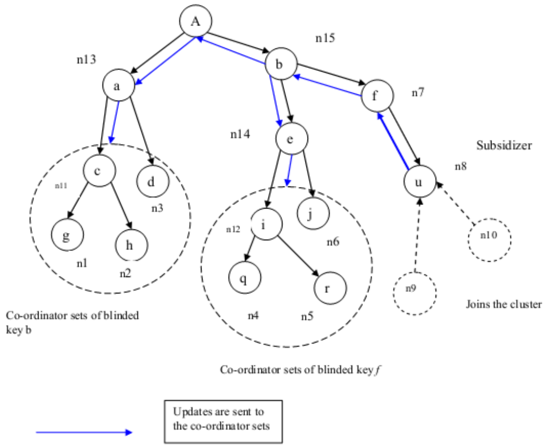
A
n15
n13
b
a
f
n7
n14
e
Subsidizer
n8
c
d
u
n11
n3
j
i
n12
g
h
n6
n10
nl
n2
q
r
n9
Co-ordinator sets of blinded key b
n4
n5
Joins the cluster
Co-ordinator sets of blinded key *f*
Updates are sent to the co-ordinator sets

Figure 6: Joining Process

4.4 Leaving Process

- In figure 7, when the node n_2 wishes to leave the group, it sends a leave message by sending a leave request.
- When the other group members receive the request, they independently determine the subsidizer, to be the right-most leaf node of the subtree rooted at the leaving member's sibling node. In this case node n_3 is selected as the subsidizer.
- The size of the cluster that formerly contained the node n_2 is decreased by one, and combines with an adjacent cluster if the size becomes lesser than k.
- When n2 leaves the cluster, the subsidizer n_3 picks a new secret key and computes the new group key.
- The updated blinded keys of nodes n13 on its key path are sent to their corresponding co-ordinator sets e*{n8,n9,n10} and f{ n4,n5,n6,n12}*. These messages are signed by the subsidizer.
- The members in these co-ordinator sets request certificate from TA to verify the updated blinded keys they received.

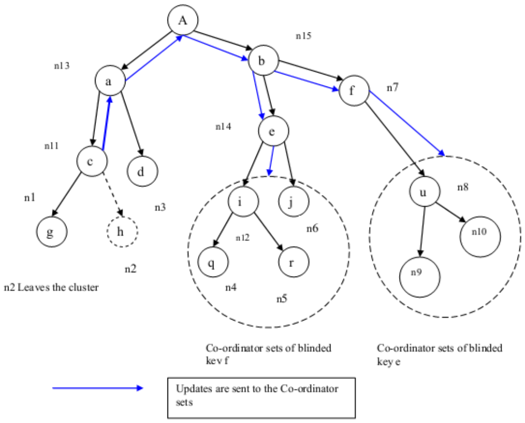
A
a
b
c
d
e
f
g
h
i
j
q
r
u
n1
n2
n3
n4
n5
n6
n7
n8
n9
n10
n11
n12
n13
n14
n15
n2 Leaves the cluster
Co-ordinator sets of blinded kev f
Co-ordinator sets of blinded key e
Updates are sent to the Co-ordinator sets

Figure 7: Leaving Process

5. SIMULATION RESULTS

5.1. Simulation Model and Parameters

We use NS2 [21] to simulate our proposed protocol. In our simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. We use the distributed coordination function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. For multicasting, we have used multicast AODV (MAODV) [12] routing protocol.

In our simulation, 50 mobile nodes move in a 1000 meter x 1000 meter region for 50 seconds simulation time. We assume each node moves independently with the same average speed. All nodes have the same transmission range of 250 meters. In our simulation, the minimal speed is 5 m/s. The simulated traffic is Constant Bit Rate (CBR).

Our simulation settings and parameters are summarized in table 1.

TABLE1: SIMULATION PARAMETERS

No. of Nodes	50
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	50 sec
Traffic Source	CBR
Rate	250Kb
Mobility Model	Random Way Point
Receivers	10,20,…50
Transmit Power	0.660 w
Receiving Power	0.395 w
Idle Power	0.335 w
Initial Energy	3.3 J

5.2. Performance Metrics

We compare our Energy Efficient Key Management and Authentication Technique (EEKMAT) with the traditional GKMP [5]. We evaluate mainly the performance according to the following metrics.

Average Packet Delivery Ratio: It is the ratio of the No. of packets received successfully and the total no. of packets sent.

Average Energy Consumption: The average energy consumed by the nodes in receiving and sending the packets are measured.

Overhead: It is the control overhead (in terms of packets) occurred in keying and rekeying operations

Packet Drop: It is the average number of packets dropped at each receiver

5.3 Results

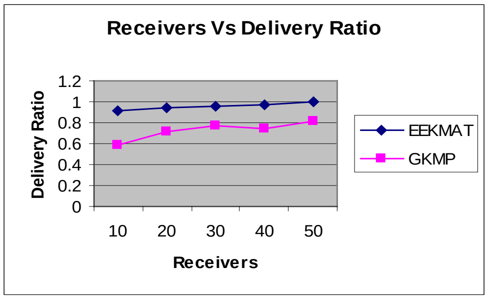

Figure 8: Delivery Ratio Vs Receivers

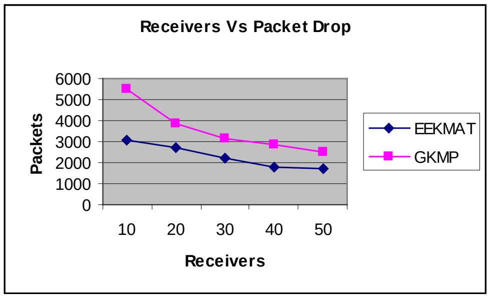

Figure 9: Packet Drop Vs Receivers

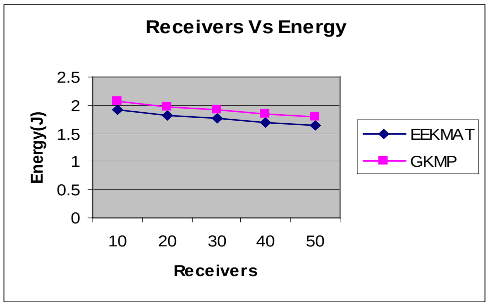

Figure 10: Energy Vs Receivers

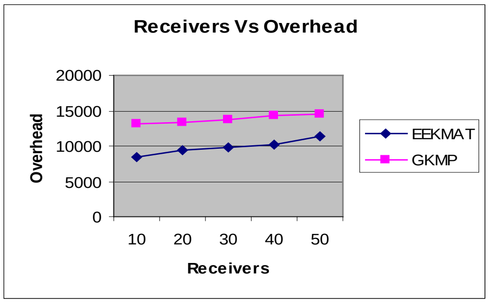

Figure 11: Overhead Vs Receivers

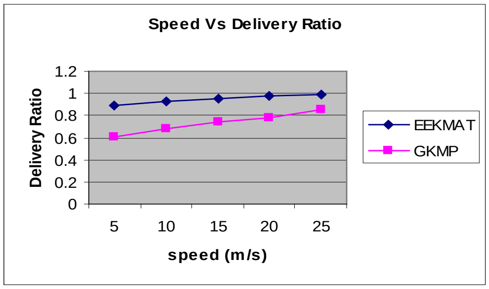

Figure 12: Packet Delivery Ratio Vs Speed

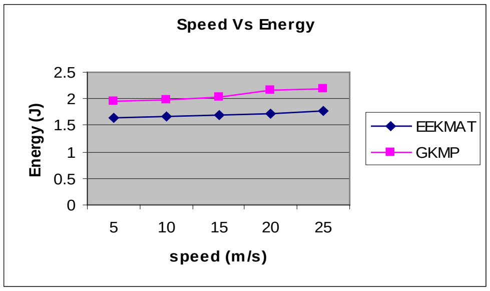

Figure 13: Energy Consumption Vs Speed

We have tested the scheme by increasing the joining receivers from 10 to 50 and measured the above parameters. Because of its energy efficient key tree construction, the average energy consumption in EEKMAT is significantly less than GKMP. Figure 7 shows this. Since the frequent rekeying operations are reduced in EEKMAT, the control overhead becomes less. So we can see from Figure 8, EEKMAT has less overhead than GKMP. Naturally when energy and overhead are less, the average packet delivery ratio is expected to improve. As we can see from Figure 6, EEKMAT has packet delivery ratio more than GKMP.

Then we vary the speed of the mobile nodes from 5m/s to 25m/s having 10 receivers. Figure 9 and 10 show the packet delivery ratio and energy consumption of both the schemes. As we can see from the figures, the delivery ratio is more and energy consumption is less for EEKMAT when compared to GKMP.

6. CONCLUSION

In this paper, we have proposed a trust authority based key management and authentication technique for multicasting in ad hoc networks. Initially we construct an energy efficient topology aware key tree which mainly aims to reduce the re-keying load by pre-processing the joining members during the idle re-keying interval. The temporary key tree algorithm significantly reduces both computation and communication costs where there exist highly frequent membership events. Key management is processed based upon Diffie-hellman key pair and RSA secret public key pair. This scheme distributes each updated public key to a co-ordinator set so that the performance can be improved. The trust authority uses an RSA secret public key pair and establishes public key certificates for each group member by signing the public key with its secret key. It adopts the proactive secret share update algorithm to periodically update the system secret shares to invalidate compromised secret shares. The nodes joining and leaving the cluster is updated by the subsidizer node. They send valid public key certificate to its co-ordinator set and obtains the public key required for the group key. Thus from our simulation results we show that this key management guarantees key authentication, enhances fault-tolerance and protects the tree from impersonation attacks.

REFERENCES

[1] V.Palanisamy and P.Annadurai, "Impact of Rushing attack on Multicast in Mobile Ad Hoc Network", *International Journal of Computer Science and Information Security (IJCSIS)*, Vol. 4, pp. 1 & 2, 2009.

[2] Luo Junhai, Xue Liu and Ye Danxia, "Research on multicast routing protocols for mobile ad-hoc networks", *Computer Networks* 52 ,pp 988–997, 2008.

[3] R. Srinivasan, V. Vaidehi, R. Rajaraman, S. Kanagaraj, R. Chidambaram Kalimuthu, and R. Dharmaraj, "Secure Group Key Management Scheme for Multicast Networks", *International Journal of Network Security*, Vol.10, No.3, PP.205–209, May 2010.

[4] D. SuganyaDevi, and G. Padmavathi, "A Reliable Secure Multicast Key Distribution Scheme for Mobile Adhoc Networks", *World Academy of Science, Engineering and Technology*, 2009

[5] Mohamed Salah Bouassida, and Mohamed Bouali "On the Performance of Group Key Management Protocols in MANETs" *Joint Conference on Security in Network Architectures and Information Systems (SAR-SSI'07)*, Annecy : France (2007)

[6] Maria Striki and John S. Baras "Key Distribution Protocols for Secure Multicast Communication Survivable in MANETs" 2003 *IEEE Military Communications Conference (MILCOM*)

[7] N. Shanthi, Dr. Lganesan And Dr. K.Ramar "Study Of Different Attacks On Multicast Mobile Ad Hoc Network" *Journal of Theoretical and Applied Information Technology* © 2005 - 2009 *JATIT*.

[8] R. Kalaidasan, Mrs. V.Hemamalini, and Anoop K Babu "SORB: Secure On Demand Resilient to Byzantine Multicast Routing in Multihop Wireless Networks" 2010.

[9] Hoang Lan Nguyen, and Uyen Trang Nguyen "A study of different types of attacks on multicast in mobile ad hoc networks" 2006 *Elsevier*.

[10] Sencun Zhu, Shouhuai Xu, Sanjeev Setia1, and Sushil Jajodia "LHAP: A Lightweight Hop-by-Hop Authentication Protocol for Ad-Hoc Networks" *Distributed Computing Systems Workshops, Proceedings. 23rd International Conference on* 2003.

[11] D.SuganyaDevi, and Dr.G.Padmavathi "Secure Multicast Key Distribution for Mobile Adhoc Networks" *(IJCSIS) International Journal of Computer Science and Information Security*, Vol. 7, No. 2, 2010.

[12] Yacine Challal, Hatem Bettahar, and Abdelmadjid Bouabdallah "SAKM: A Scalable and Adaptive Key Management Approach for Multicast Communications" *ACM SIGCOMM Computer Communications Review* Volume 34, Number 2: April 2004.

[13] Yun Zhou, Xiaoyan Zhu, and Yuguang Fang "MABS: Multicast Authentication Based on Batch Signature" *IEEE Transactions On Mobile Computing*, Vol. 9, No. 7, July 2010.

[14] Shaobin Cai, Wenbin Yao, NianminYao, Yong Li, and Guochang Gu "Group-Based Key Management for Multicast of Ad Hoc Sensor Network" *ICCS 2007, Part III, LNCS 4489, pp. 50–57, 2007. © Springer-Verlag Berlin Heidelberg* 2007.

[15] Bo Rong, Yi Qian1, Rose Qingyang Hu, Sghaier Guizani, and Michel Kadoch "Key Management for Pyramidal Security Model of Multicast Communication in Mobile Ad Hoc Networks" *Global Telecommunications Conference, 2006. GLOBECOM '06. IEEE.*

[16] Feng He, Kuan Hao, and Hao Ma "S-MAODV:A Trust Key Computing Based Secure Multicast Ad-hoc On Demand Vector Routing Protocol" 2010 *IEEE.*

[17] Mahalingam Ramkumar, Nasir Memon, and Rahul Simha "Pre-Loaded Key Based Multicast and Broadcast Authentication in Mobile Ad-Hoc Networks" *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE.*

[18] D.Suganya Devi and Dr. G.Padmavathi "Efficient Cluster Based Multicast Tree for Secure Multicast Communication for Mobile Ad Hoc Networks" *International Journal of Engineering Science and Technology* Vol. 2(5), 2010, 1304-1310.

[19] Loukas Lazos and Radha Poovendran "Power Proximity Based Key Management for Secure Multicast in Ad Hoc Networks" *Journal Wireless Networks archive* Volume 13 Issue 1, January 2007.

[20] Li Zhou and Chinya V. Ravishankar "Efficient, Authenticated, and Fault-Tolerant Key Agreement for Dynamic Peer Groups" *IFIP International Federation for Information Processing* 2004.