ISSN: 2278 – 1323
International Journal of Advanced Research in Computer Engineering & Technology
Volume 1, Issue 5, July 2012

SECURE MULTICAST COMMUNICATION USING GROUP BASED MULTICAST HIERARCHY

R. Varalakshmi[#1], Dr. V. Rhymend Uthariaraj[*2]

[#]*Teaching Research Associate – Department of Mathematics, Anna University, Chennai, India*

[*]*Professor and Director, Ramanujan Computing Centre Anna University, Chennai, India*

***Abstract*— Secure multicast communication is a significant requirement in emerging applications in adhoc environments like military or public emergency network applications. Membership dynamism is a major challenge in providing complete security in such networks. This paper proposes a efficient Group Based Multicast Hierarchy (GBMH) algorithm for secret multicast communication, in which source nodes used the Secure Ad hoc On Demand Distance Vector (SAODV) protocol is an extension of the AODV protocol. The Secure AODV scheme is based on the assumption that each node possesses certified public keys of all network nodes. The SAODV protocol collects its 1 hop neighbours to form group. This protocol sends acknowledgement for each transmission in order to reduce the retransmission. Membership dynamism was overcome by electing local controllers and with periodic updates of node join and leave information using multicast hierarchy. The performance is studied in terms of average end to end delay and fault tolerance in multicast transmission.**

***Keywords*— Secure Multicast Communication, Adhoc environments, Membership Dynamism, GBMH, SAODV**

I. INTRODUCTION

Many applications like pay-per–view, distribution of digital media etc., require secure multicast services in order to restrict group membership and enforce accountability of group members. A major issue associated with the deployment of secure multicast delivery services is the scalability of the key distribution scheme. This is particularly true with regard to the handling of group membership changes, such as membership departures and/or expulsions, which necessitate the distribution of a new session key to all the remaining group members. As the frequency of group membership change increases, it becomes necessary to reduce the cost of key distribution operations. A common method for secure multicast communications is to use a symmetric key called traffic encryption key(TEK), which is shared by all legitimate group members and used to encrypt the transmitted content. In order to prevent the joined members from reading previous content and the left member from reading the further content, TEK must be refreshed after the membership is changed. An easy way is to allow the key server to share a unique key encryption key(KEK) with every member. When the membership is changes, the key server uses the individual KEK of every member to encrypt the new TEK. This is an inefficient method because the cost of TEK updates grows linearly with the group size. Therefore, the key changing process becomes a critical problem in multicast key management. Efficient key management protocols should be taken into consideration for security requirements.

Security requirements:

1. Forward secrecy: In this case, users left the group should not have access to any future key. This ensures that a member cannot decrypt data after it leaves the group.
2. Backward secrecy: A new user who 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.
3. Non-group confidentiality: Here users that are never part of the group should not have access to any key that can decrypt any multicast data sent to the group.
4. Collusion freedom: Any set of fraudulent users should not be able to deduce the currently used key.

The process of updating the keys and distributing them to the group members is called rekeying operation. A critical problem with any rekey

technique is scalability [6]. The rekey process should be done after each membership change, and if the membership changes are frequent, key management will require a large number of key exchanges per unit time in order to maintain both forward and backward secrecies. More frequent membership dynamism causes node failure, link failure, power failure leads to time delay in multicast transmission. To overcome these problems, several approaches propose a multicast group clustering. [7, 8, 9]. Grouping is dividing the multicast group into several sub-groups. Local controller (LC) manages each subgroup, which is responsible for local key management within the cluster. Thus, after Join or Leave procedures, only members within the concerned cluster are affected by rekeying process, and the local dynamics of a cluster does not affect the other clusters of the group and hence it overcomes 1-affects-n [21,22,23,24] phenomenon. Moreover, few solutions for multicast clustering such as dynamic clustering did consider the issue of average end to end delay to achieve an efficient key distribution process, whereas delay in transmission constitutes main issue in ad hoc environments.

II. Related Work

In order to reduce the rekeying overhead, the key tree architecture has been widely used in multicast communications. First, a logical key hierarchy (LKH) tree approach [16,17] has been proposed to reduce the computational and transmitted cost from O(n) to O(log n) in the rekeying process, where n is the number of group members. Then several improvements are proposed. Lie et al. [18] proposed a periodic batch rekeying algorithm to solve synchronization and inefficiency problems. Sherman and McGrew [15] proposed a one-way function tree (OFT) to reduce the size of the rekeying message from 2(logd n) to only (logd n). However, the communication cost will be greater than (logd n) as soon as the key tree is out of balance. Recently, the approaches for keeping the tree architecture balance have been proposed. Goshi and ladner [18,19] solved the unbalance problem based on 2-3 trees and have the best performance with the degree-3 key trees. Lu[20] proposed the non-split balancing high-order (NSBHO) tree, which does not need to perform node splitting after the member joining but it has better than average rekeying performance than a B-tree.

Key management approaches can be classified into three classes: centralized, distributed or decentralized. Distributed key agreement protocols do not rely on a group leader which has an advantage over those with a group leader because, without a leader, all members are treated equally and if one or more members fail to complete the protocol, it will not affect the whole group. In the protocols with a group leader, a leader failure is fatal for creating the group key and the operation has to be restarted from scratch. The 1-affects-n phenomenon is not considered because in distributed protocols all the members are contributors in the creation of the group key and hence all of them should commit to the new key whenever a membership change occurs in the group. The decentralized approach divides the multicast group into subgroups or clusters, each sub-group is managed by a LC (Local Controller) responsible for security management of members and its subgroup. Two kinds of decentralized protocols are distinguished as static clustering and dynamic clustering. In Static clustering approach, the multicast group is initially divided into several subgroups. Each subgroup shares a local session key managed by

LC. Example: IOLUS [7] belongs to the categories, which are more scalable than centralized protocol. Dynamic clustering approach aims to solve the “1 affect n” phenomenon. This approach starts a multicast session with centralized key management and divides the group dynamically. Example: AKMP [8], SAKM [9] belong to this approach and are dedicated to wired networks. Enhanced BAAL [10] and OMCT [11, 12] propose dynamic clustering scheme for multicast key distribution in adhoc networks.

OMCT (Optimized Multicast Cluster Tree) is a dynamic clustering scheme for multicast key distribution dedicated to operate in ad hoc networks. This scheme optimizes energy consumption and

latency for key delivery. Its main idea is to elect the local controllers of the created clusters. OMCT needs the geographical location information of all group members in the construction of the key distribution tree.

Once the clusters are created within the multicast group, the new LC becomes responsible for the local key management and distribution to their local members, and also for the maintenance of the strongly correlated cluster property. The election of local controllers is done according to the localization and GPS (Global Positioning System) information of the group members, which does not reflect the true connectivity between nodes. Based on the literature reviewed, OMCT is the efficient dynamic clustering approach for secure multicast distribution in mobile adhoc networks. To enhance its efficiency, it is necessary to overcome the criteria, as OMCT needs geographical location information in the construction of key distribution tree by reflecting true connectivity between nodes. It does not acknowledge the transmission and results in delay in multicast transmission.

Several different protocols have been proposed for ad-hoc routing. The proposal of this paper is to present an efficient Group Based Multicast Hierarchy (GBMH) using Multicast version of SAODV for secure multicast key distribution.

III. SAODV Operation

The originator of the routing control packet appends its RSA signature and the last element of a hash chain to the routing packets. A packet transverse the network, intermediate nodes cryptographically authenticates the signature and the hash value. The intermediate nodes generate the kth element of the hash chain, with k being the number of transverse hops, and place it in packet.

The SAODV protocol gives two alternatives for ROUTE REQUEST and ROUTE REPLY messages. In the first case when a ROUTE REQUEST is sent, the sender creates a signature and appends it to packet. Intermediate nodes authenticate the signature before creating or updating the reverse route to the host. The reverse rout is stored only when the signature is verified. When the node reaches the destination, the node signs the ROUTE REPLY with its private key and sends it back. The intermediate nodes again verify the signature .The signature of the sender is again stored with the along with the route entry.

A. Features

i. Ownership of certified public keys enables intermediate enable intermediate nodes to authenticate all in-transit routing packets.

ii. The protocol operates mainly by using the new extension message with the SAODV protocol.

iii. The SAODV can be used to protect the route discovery mechanism of the AODV by providing security features like integrity, authentication and nonrepudiation.

SAODV have multicast connectivity between nodes. It sends acknowledgement for each transmission in order to reduce the retransmission. The LCs are elected easily with periodic updates of node join and leave information using multicast hierarchy. This overcomes the issues of end to end delay in multicast transmission and also tolerates the fault that occurs due to node failure. The GBMH algorithm is simulated with network simulator NS-allinone-2.33 and the performance is studied in terms of average end to end delay and fault tolerance in multicast transmission.

IV. EFFICIENT GBMH WITH SAODV

The proposed approach is to achieve secure multicast communication for adhoc networks. This approach uses Multicast version of SAODV routing protocol to maintain routing table periodically. It forms multicast hierarchy among the group members. Each node can determine their present physical location. It quickly adopts to the topology changes. It is used to discover alternate route for failure of existing route. It also sends acknowledgement for each transmission in order to reduce the retransmission. Thus the approach of GBMH using SAODV tends to have multicast connectivity between the nodes.

The approach of Efficient GBMH with SAODV is described in five phases with specific notations

Phase 1: **Authentication:** For each node, assign certificate key to verify its node identity. Each node has address, node address and certificate key. Certificate key and its IP address encrypt to form a public key. Thus, each node is authenticated based on broadcast request and reply.

Node Authentication and Access Control

mg $_{ik}$ → LC$_{ik}$: Join_Request, Pub_mg$_{ik}$

LC$_{ik}$ → mg$_{ik}$: Join_Request

mg$_{ik}$→ LC$_{ik}$: Join_Reply, Pub_ mg $_{ik}$,{GBID_}Pri_mg $_{ik}$

Phase 2: **Group Head Election**: Initially the list of Local Controllers(LCs) contains only the source Group Controller (GC). Then, GC collects all its 1 hop neighbours by SAODV routing protocol. Elect LCs which are group members and which have child group members (the LC belongs to the unicast path between the source and the child group members). Verify for each one if it a group member and if it has child group members then add the LC to the list of LCs. Thus, LCs are selected as group heads for its corresponding group members.

Phase 3: **Group Formation**: All the members reachable by this new LC will form a new group. If group members that exist and do not belong to the formed group then choose the nodes that have the maximum reachability to the other nodes in one hop from the remaining members. This reachability information is collected through the SAODV routing protocol. Thus, nodes are selected as local controllers for the remaining group members and forms new group.

Phase 4: **Secure Multicast Communication**: The source encrypts multicast data with the TEK, and then sends it to all the members of the group following the multicast hierarchy. The TEK distribution is achieved in parallel, according to the following steps. Initially, the entire group members receive from the source by unicast the session key KEKgsg-0(key encryption key of the group sub-group 0), encrypt with the respective public keys. Each local controller should join this group. The local controllers decrypt the message, extract the TEK, re-encrypt it with their respective group keys and send it to all their local members.

TEK Distribution

For all mg$_{ik}$,Gk → mg $_{ik}$:{TEK, Num_Seq, KEK_GSG $_{ik}$, IDG, IDGSG, Pub_G,(GBID_CG) Pri_G }Pub_ mg $_{ik}$

Phase 5: **Node mobility**: For frequent node mobility, a new member may join a group or an existing member may leave a group. To ensure secure multicast communication, both forward and backward secrecy has to be maintained.

Forward Secrecy: When a node leaves the multicast group, it cannot decrypt the future data. The leave operation is in two cases

I. When an ordinary node leaves, it gives less effect in multicast transmission. The leave operation of an ordinary node is specified as follows:

Leave Procedure

mg $_{ik}$: outgoing member leaving a group

For mg $_{ik}$: Local member,

mg $_{ik}$ < > mg $_{ik}$ _outgoing

LC $_{ik}$ → mg $_{ik}$:{IDLC, KEK_GSG $_{ik}$ }Pub_ mg $_{ik}$

II. When a local controller leaves, it leads to clusterization. It first sends the leave notification to the group controller and then all the members of the current LCs are merged with the other group based on the reachability information obtained by the SAODV routing protocol.

Leave Notification

LC $_{ik}$ → GLC:{ID_ LC $_{ik}$ }KEK_GLC

For all j< > i, GC k → LC $_{ik}$:{ID_GC, new_KEK_GLC} Pub_CL $_{jk}$

Merge

For all mg $_{ik}$, LC $_{ik}$: {ID_group, LL_LC $_{ik}$ }KEK_GSG$_{ik}$

Backward Secrecy: When a new node joins the multicast group, it cannot decrypt the past encrypted data. Each new node is authenticated based on broadcast request and reply.

Join Procedure

For old_ mg $_{ik}$: old member of group
LC $_{ik}$ → old_ mg$_{ik}$:{IDLC, KEK_GSG$_{ik}$ }old_KEK_ GSG $_{ik}$
LC $_{ik}$ → mg$_{ik}$:{IDLC, TEK, KEK_GSG$_{ik}$}Pub_ mg $_{ik}$

Thus the approach of an efficient Group Based Multicast Hierarchy (GBMH) using Multicast version SAODV is described in five phases in order to have secure multicast communication adhoc networks. This approaches the issues of end to end delay in multicast transmission and also tolerates the fault that occurs due to node failure.

V. PERFORMANCE ANALYSIS

The performance of secure multicast communication of the efficient GBMH for adhoc networks in terms of end to end delay and fault tolerance due to node failure is analyzed. This approach is simulated under Linux Fedora, using the network simulator NS2 version ns-allinone-2.33. The performance metrics are namely average end to end delay and fault tolerance of secure multicast communication.

End to End Delay: The average latency or end to end delay of keys transmission from the source to the receivers. This metrics allows evaluating the average latency to forward a key from a LC to its group members.

Fault Tolerance: This metrics allows evaluating the percentage of tolerance of fault that occurs due to node failure. Fig 1 and Fig 2 shows the simulation results of the comparison of OMCT with GBMH.

Fig. 1 Average end to end delays of multicast transmission

The average end to end delays of multicast transmission is efficient in proposed GBMH.

As number of nodes increases, it increases the fault-tolerance in key distribution. Indeed, this approach divides the multicast group with the effective connectivity between nodes. It allows fast reaction to topology changes. This is due to the fact that it sends acknowledgement for each transmission in order to reduce the retransmission. Hence it tolerates the fault that occurs due to node failure of multicast transmission in efficient GBMH compared to OMCT.

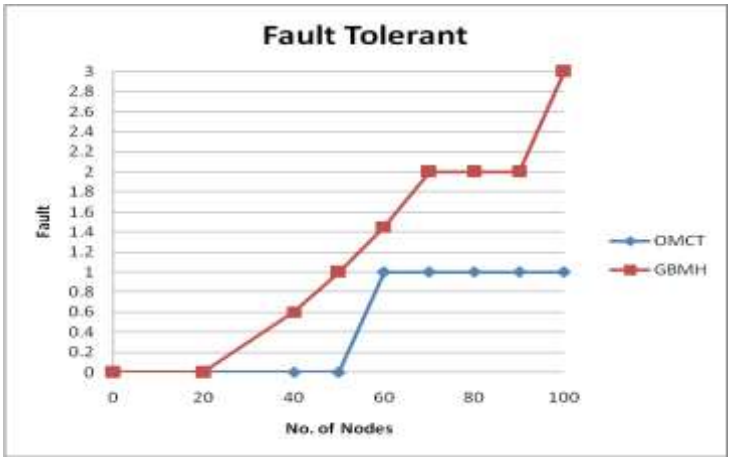

Fig. 2 Fault Tolerance in multicast communication

VI. CONCLUSION

Secure multicast communication in adhoc networks is challenging due to its inherent characteristics of infrastructure-less architecture with lack of central authority, limited resources such as bandwidth, time and power. Hence key management is the fundamental challenge in achieving secure communication using multicast key distribution in adhoc networks. This paper studies how to design key management schemes for such networks that will allow to identify nodes without the need of any kind of certification authority. In addition, it presents a method to reduce the delays in route establishment in cases where routing messages are signed and need to be verified. Finally, it applies all these to SAODV (an extension of the AODV routing protocol that protects the route discovery mechanism providing security features like integrity and authentication), and presents results from simulations that show how this method provides the same security with minimum impact in the network performance. Simulation results shows the demonstration of GBMH using SAODV have better system performance in terms of end to end delay and fault tolerance rate under varying network conditions. Therefore, providing a more complete solution to the problem of security in adhoc networks.

REFERENCES

[1] Chiang, T., Huang, Y.: Group keys and the multicast security in ad hoc networks. In: Proc. IEEE International Conference on Parallel Processing, October 2003, pp. 385–390. IEEE press, Los Alamitos (2003)
[2] T. Kaya, G. Lin, G. Noubir, and A. Yilmaz.:Secure multicast groups on ad hoc networks. In: Proc. 1st ACM workshop on security of ad hoc and sensor networks, ACM Press, pp 94-102.(2003).
[3] Lazos, L., Poovendram, R.: Energy-Aware Secure Multicast Communication in Ad Hoc Networks Using Geographical Location Information. In: Proc.IEEE International Conference on Acoustics Speech and Signal Processing, April 2003, pp. 201–204 (2003)
[4] Dondeti, L., Mukherjee, S., Samal, A.: Secure one-to many group communication sing dual encryption. In: IEEE sym. on Computers and Communications, July 1999, pp. 1–25 (1999)
[5] H. Harney and C. Muckenhirn. Group key management protocol (gkmp) specification. RFC2093, 1997.
[6] G. H. Chiou and W. T. Chen. Secure Broadcast using Secure Lock. IEEE Transactions on Software Engineering, August 1989.
[7] Mittra, S.: Iolus: A framework for scalable secure multicasting. In: SIGCOMM, pp. 277–288 (1997)
[8] Bettahar, H., Bouabdallah, A., Challal, Y.: An adaptive key management protocol for secure multicast. In: Proc. IEEE International Conference on Computer Communications and Networks, October 2002, pp. 190–195 (2002)
[9] Challal, Y., Bettahar, H., Bouabdallah, A.: SAKM: A Scalable and Adaptive Key Management Approach for Multicast Communications. ACM SIGCOMM Computer Communication Review, 55–70 (April 2004)
[10] Bouassida, M., Chrisment, I., Festor, O.: An Enhanced Hybrid Key Management Protocol for Secure Multicast in Ad Hoc Networks. In: NETWORKING 2004. LNCS, vol. 3042, pp. 725–742. Springer, Heidelberg (2004)
[11] Bouassida, M., Chrisment, I., Festor, O.: Efficient Clustering for Multicast Key Distribution in MANETs. In: NETWORKING 2005. LNCS, vol. 3462, pp. 138–153. Springer, Heidelberg (2005)
[12] Bouassida, M., Chrisment, I., Festor, O.: Group Key Management in Manets. International Journal of Network Security, 67–79 (January 2008)
[13] Zapata,M.G.,"Secure ad-hoc on-demand distance vector (SAODV) routing ," IETF MANET ,internetdraft (Work in progress),draft - guerrero-manet-saodv-00.txt,2001.- accessed 10/10/2006.
[14] A H A Rahman, Z A Zukarnain.: Performance Comparison of AODV, DSDV and I-DSDV routing protocols in Mobile Adhoc Networks. In.: European Journal of scientific Research, pp 566-576, 2009\
[15] Elgamal T., A public key cryptosystem and a signature scheme based on discrete logarithms, CRYPTO 84 on Advances in Cryptology Proceedings, 1984, 10-18.
[16] Debby M. Wallner, Eric J. Harder, Ryan C. Agee, "Key Management for Multicast: Issues and Architectures", Informational RFC, draft-Wallnerkey-arch-ootxt, July 1997.
[17] Chung Kei Wong, Mohamed Gouda, and Simon S Lam, "Secure Group Communication Using Key Graphs", Proceedings of ACMSIGCOMM, Vancouver, British Columbia, September 1998.
[18] Goshi, J. and Ladner, R.E (2003) Algorithms for Dynamic Multicast Key Distribution Trees. Procs. Twenty-second Annual Symp. Principles of Distributed Computing (PODC2003), New York, NY, USA, July, pp 243-251.
[19] Goshi, J. and Ladner, R.E. (2007) Algorithms for dynamic multicast key distribution. J. Exp. Algorithmics, 11, 1-37.
[20] Lu, H(2005) A novel high-orer tree for secure multicast key management. IEEE Trans. Comput., 54, 214-224.
[21] C.K.Wong, M. Gouda, S.S.Lam, 2000. "Secure Group Communications using key graphs", IEEE/ACM Transactions on networking, pp.16-30.
[22] Sandro Rafeli, David Hutchison, 2000." A survey of Key Management for Secure Group Communication", ACM Computing Surveys, Vol.35, No.3, pp.309-329.
[23] M. Manulis, "Security-Focused Survey on Group Key Exchange Protocols," Report 2006/395, Cryptology ePrint Archive, http:// eprint.iacr.org/, 2006.
[24] F. Zhu, A. Chan, and G. Noubir, "Optimal Tree Structure for Key Management of Simultaneous Join/Leave in Secure Multicast," Proc. Military Comm. Conf. (MILCOM), 2003