Reducing Overhead on Double Rekeying in Secure Group Communication

J.Dhanalakshmi ,
Asst Professor,Dept of CSE,
J.J.College of Engineering &Technology,Trichy

Viji Vinod,
Professor& HOD of Computer Applications
Dr.M.G.R Educational and Research Institute
University, Chennai

Abstract

Nowadays networks require flexible dynamic group communication with the internet. When we develop these systems on multicast communication framework, various types of security threat occurs .As a result construction of secure group communication that protects users from intrusion and eavesdropping are very important.

In this paper, we propose an efficient key distribution method for a secure group communication over multicast communication framework. In this method, we use IP multicast mechanism to shortest rekeying time to minimize adverse effect on communication. In addition, we introduce proxy mechanism for replies from group members to the group manager to reduce traffic generated by rekeying.

We define a new type of batching technique for rekeying in which new key is generated for both leaving and joining member. The rekeying assumption waits for 30 sec so that number time's key generation will be reduced.

Keywords: IP, Secure communication, Re-key

1. Introduction

With recent improvements in high-speed broadband technology, many new multi-point multi-user applications on the Internet, such as distributed simulations, multi-user games, conferencing and contents distribution, have emerged. They would be realized on IP multicast communication framework IP multicast technology consists of a group management protocol (IGMP Internet Group Management Protocol) and multicast routing protocols. It reduces transmission overhead, requirements for network bandwidth and the latency observed by receivers. However, current limiting factor in the wide deployment of IP multicast for commercial purpose is its lack of security.

To support security in multicast on the existing network, we introduce a secure group communications in which a unique key, called session key is shared between group members. This paper is concerned with the dynamic secure communications (DSGs). In particular, we discuss the cases when a new member joins or a member leaves the group. In the security requirements for DSGs, re-keying is still considered as open research issues. Recently, several key management methods geared for DSGs were proposed. Furthermore, on distributing and updating the session keys, the problems of communications traffic and transfer delay are very important issues that must be solved. That is, when a key update occurs and a member sends a message encrypted by a new session key, it is not guaranteed that the others members have already received the new session key due to network delays. Consequently, a members have has not yet received the new session key will not be able to decrypt the message and the data must be retransmitted after the members receive the key. In addition, key distribution needs reliability and this causes the concentration of acknowledgements from members.

In this paper, we define a distributed system model for key distribution and propose an efficient key updating protocol. In this model, we introduce the key management server (KMS) to distribute session keys and the authentication server (AS) to authenticate members. In addition, we introduce a subnet management server (SGM) that has proxy function on the same subnet to prevent concentration of data from KMS. We also discuss an efficient key distribution protocol using IP multicast. To show that the key updating time is shortened and traffic of key distribution is reduced, we will evaluate the proposed method on a multicast network model.

In this paper, we define a new method of rekeying in which server will generate a new group key for both two leaving or two joining or one leaving and one joining members at a time. In this way, the number of key generation will be reduced to half when compared to single rekeying method. If the first user waits for long time without a paired member, the server will generate a new key after a particular time. It will be similar to Batch rekeying in which new key will be generated after particular time. We combine the method of both single and batch rekeying in the single system. It is also very efficient method for key distribution because it reduces the number of key generation when compared to other method. Hence, the distribution time and traffic are reduced.

In the rest of the paper, section 2 briefly gives the some related work. In section 3, we describe a concept of existing method of secure multicast communication. In section 4, we propose an efficient key distribution method of dynamic secure group communication. Section 5, discusses an evaluation of the proposed method. We will conclude the paper in section6.

2. Related Work

In [17], the group controller maintains logical hierarchy of keys that are share by different subsets of users. To revoke multiple users, the group controller aggregates all the necessary key updates to be performed and processes them in the single step. However, the group controller interrupt the group communication until all the necessary updates are performed, and then, distributes new group key to restore group

communication. This interruption to group communication is undesirable for real-time multimedia applications.

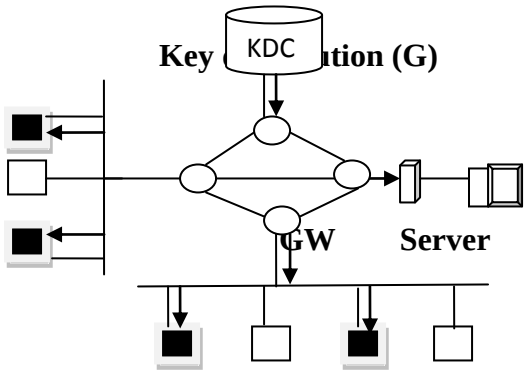


Fig.1 Secure Group communication System model

In [18], to handle multiple group membership changes, the group controller performs periodic re-keying, i.e., instead of re-keying whenever group membership changes, the group controller performs rekeying only at the end of selected time intervals. However, the revoked users can access group communication until the group is rekeyed. This can either cause monetary loss to the service provider or compromise confidentiality of other users.

In [22], the group controller maintains a logical hierarchy above schemes; the logical key tree structure tends of keys similar to the solutions in [17]. To revoke multiple users, the group controller distributes the new group key by using keys that are not known to the revoked users. However, this solution achieves the good re-keying cost only if the size of the revoked user either very small or very large. In the above scheme, logical tree tend to become unbalanced after some membership and result in tree which has large height (O (N)). As the height of the tree determines the re-keying cost, several approaches have been proposed.

3. Existing Re-Keying Methods For Dynamic Secure Communications

We consider secure group communications for the members of a group sharing a session key. In particular, re-keying is a key point to realize DSGs and a member of works are in progress. In those methods, some centralized re-keying methods are reported.

A system model and a re-keying sequence in centralized methods are shown in Fig.1. Where, the function is assigned to KDC, which generates session keys and distributes them to group members and data encryption gateway (GW).

For the safety of DSGs, session key should be updated every time a member join or leave the group, since all group members share the unique session key. For example, re-keying sequence is

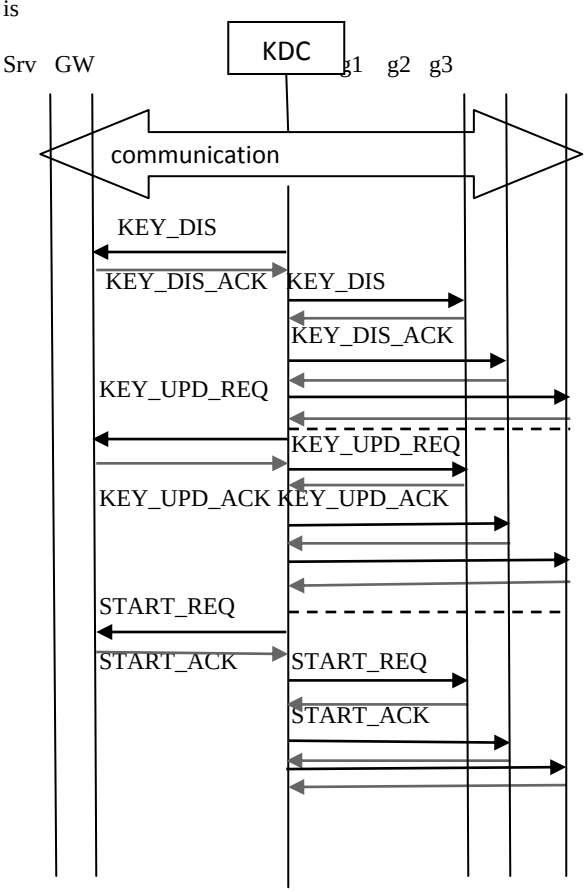


Fig 2.Re-Keying sequence

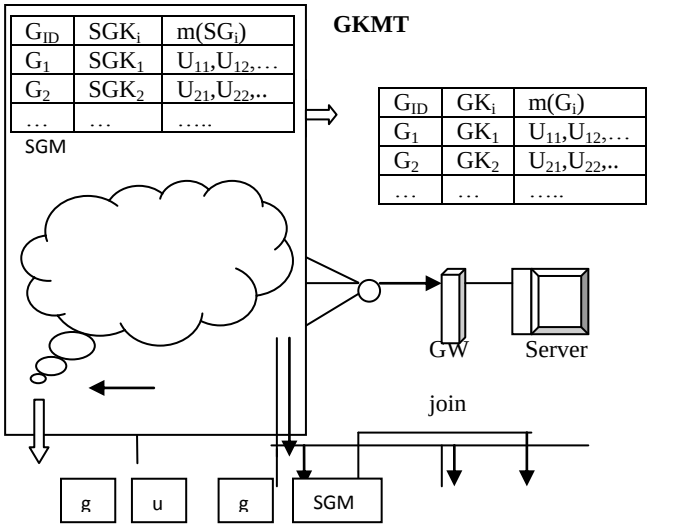
shown in Fig.1 (b) where member g4 request to leave. This sequence consists of two parts: a distribution sequence of a new session key and an update sequence. In distribution sequence, a new
session key, which encrypted by the old session key, is distributed to each member except member g4. Update sequence is started after KDC receive acknowledgement from member's g1 and g3. For consistency, KDC sends key update request (KEY_UPD_REQ) to each member (g1). Encrypted by its master key. Each member sends acknowledgement (KEY_UPD_ACK) to KDC. After receiving this message KDC sends START_REQ to restart the data communication and wait for acknowledgement form all the members.

This method tries to improve system reliability by confirming an acknowledgement of each distribution data. This leads, however to the longer duration time, when, in particular, it is applied large groups. In addition, network traffic is increased and the acknowledgements are concerned at KDC.

4. Efficient Re-Keying Protocol

4.1 Distributed System Model for Dynamic Secure Group Communications

In order to solve the above problems, we propose a distributed system model for DSGs as shown in



g : router : host u : group G host :non-member host

GKMT : Group Key Management Table

SGMT : Subgroup Management Table

Fig.3 Distributed system model for DSGs.

Fig. 3. In this system model, we introduce an SGM, which have proxy function on the same subnet to prevent concentration of data from KMS, and a group and AS. Each gathers acknowledge messages of key distribution on the subnet and sends an acknowledge message to KMS. Each entity composing the system model is defined as follows:

Definition 1: KMS generates session keys , encrypts them with shared common keys between KMS and SGMs and distributes them to SGMs. KMS has a group key management table (GKMT), in which session keys and access control lists are included.

Definition 2: AS authenticates senders or receivers of multicast data, and issues certifications for them. AS has a user management table (UMT), in which a certification and access duration time of each member are included.

Definition 3: SGM manages user hosts on the same subnet. SGM generates a sub-group session key (SGK), encrypts it with the members public key and distributes it to each member. SGM has a sub-group management table (SGMT), in which SGK and member list are included.

The notations used throughout the paper are shown in Table.1.

Table.1. Notations

u_i	host i
PK_i	public key of u_i
SK_i	Private key of u_i
CK_i	session key of each communication group i
SGK_k	subnet session key for encryption
k[M]	encrypted message with key k
U_{ID}	user identifier
G_{ID}	group identifier
AT_i	access duration time of ui
$m(SG_k)$	sub group member under management of SGM_k
$m(G_i)$	group member of group i
x->y:w	X sends "w" to y

4.2. Assumptions

We give some assumptions based on design issues of the proposed re-keying protocol.

Assumption 1: KMS and AS are positioned at physically safe place in server provider.

Assumption 2: Each member trusts SGM.

Assumption 3: Public keys, PK_{SGM} for SGM and PK_0 for members, are initially set up as secret information for authentication.

Assumption 4: Common key, CK_1 between KMS and SGM_1, is initially setup as secret information for authentication.

Assumption 5: Group member keep their session keys secret.

4.3. Proposed re-keying Protocol

4.3.1. Leaving and joining of members

When a member wishes to leave the group during secure group communication, the session key must be updated. The proposed re-keying protocol consists of distribution sequence of a new session key and key updating sequence. In the case of leaving or joining member g_k, re-keying sequence is shown in Fig.3 (a).

Step 1 Distribution sequence

1.g_k->SGM_k: $PK_{SGM}[LV\_REQ,U_{ID},G_{ID})]$

A group member g_k sends a request for leaving to its own SGM. This message includes user ID and group ID information encrypted by public key of SGM.

2. SGM_k->KMS :$CK_k[LV\_REQ,U_{ID},G_{ID}]$

SGM_k transfers a leave request, which is encrypted by common key shared between SGM and KMS to KMS.

3. KMS->SGM :CK_k[WAIT]

KMS send the WAIT message to SGM encrypted with common key if there is no other member sent the request to leave/join.

4. SGM_k->SGM_k->m($SG`_k$) : $SGM`_k$[WAIT];

SGM send WAIT to g_k if it receives WAIT from KMS.

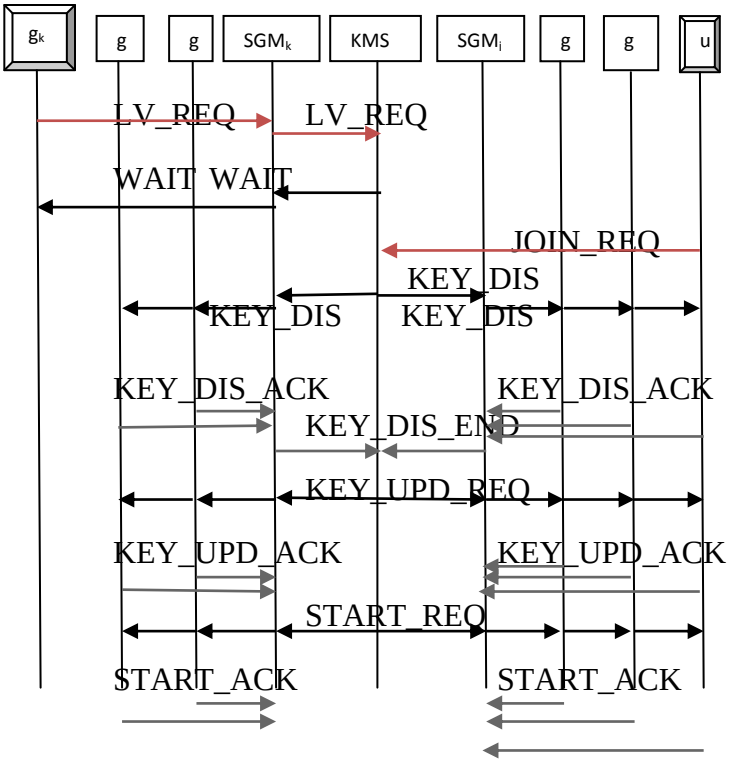


Fig.4.Leaving and Joining sequence

5. g_h->SGM_h: PK_{SGM}[LV_REQ,U_{ID},G_{ID})] ; g_h->SGM_h: PK_{SGM} [JOIN_REQ,U_{ID},G_{ID})]

Other member send the leave request or new member send the join request by encrypted with public key of SGM.

6. KMS->SGM :CK_k[KEY _DIS,$GK`_1$,G_{ID}]

KMS distribute the new session key encrypted with common key shared between SGM and KMS.

7.SGM->m($SG`_k$) : $SGM`_k$[KEY_DIS,$GK`_i$,G_{ID}]

Otherwise it removes g_k from table and distribute new session key to all remaining members.

8. g_k->SGM_k : PK_{SGM}[KEY_DIS_ACK,U_{ID}]

After receiving the new session key, each member sends an acknowledgement message to its own SGM.

9. SGM_k->KMS :CK_k[KEY_DIS_END,G_{ID}]

After collecting all acknowledgements, SGM informs the ending of key distribution for KMS.

Step 2: Update sequence

1.KMS->m(G_i): $GK`_i$[KEY_UPD_REQ,G_{ID}]

KMS broadcasts a key update request to the new group members. This means that a member who does not have a new session key cannot access the message.

2. g_k->SGM_i: PK_{SGM}[KEY_UPD_ACK,U_{ID}]

After receiving the update request message, each member sends acknowledge to its own SGM.

3. SGM->g_i :PK_i[KEY_UPD,G_{ID}]

If a member doesn`t send an acknowledgement back, SGM re-sends a key-update message to the member

4.SGM_i-> KMS: CK_i[KEY_UPD_ACK,G_{ID}]

After collecting all acknowledge messages, SGM informs key update acknowledgement messages to KMS.

5.KMS-> m(G_i) : $GK`_i$[START_REQ,G_{ID}]

KMS broadcast a request for starting of data transfer to the member of group G_i.

6.g_i->SGM_i : Pk_{SGM}[START_ACK,U_{ID}]

After receiving the request, each member sends an acknowledgement to its own SGM.

5. Evaluations

In this section, we define a multicast communications network model and evaluate the proposed method by computation using the model.

5.1. Network model

We introduce a multicast communication network model shown in Fig 4. This model consists of multicast routers, group management centre (GMSs), which consists of hosts and KMS. In this model, we define the LAN including GMC as level 0 routers, the routers connected to the level i+1 routers excluding the level i-1 routers as level i+1 routers and the LANs including level j routers as level j LANs, respectively.

To evaluate the time for distributers and characteristics of communication traffic during key distribution by consumption, we use the distribution function as an index of distribution of group members in the network.

Distribution function X(p) is defined as follows

$$X(p) = \sum_{j=1}^{k} (j.hj(p)) \quad (1)$$

Where, k denotes the maximum level of the network model. In this paper, we assume that k is 5 and take account of 16 types of distribution, in which the values of distribution function are from 3700 to 7000 when the number of hosts is 1500.

5.2. Distribution time

From Fig 1(b), key distribution time of the conventional method is estimated as

$$D_U(p) = 4.S(p) + 2e.S(p) - k.D_k, \quad (2)$$

$$S(p) = D_k.\sum_{j=1}^{k-1} (H - \sum_{j}^{i=1} hi(p)) \quad (3)$$

Where, e denotes the proportion of the number of packets that does not arrive their destinations to all packets.

On the other hand, key distribution time of proposed method is estimated as:

$$D_M(p) = (3k-1)D_k + 2e.S(p) \quad (4)$$

Fig.5 shows the ratio of D_U to D_M. From the figure, we can see that distribution time of the proposed method is about one to ten percent of the conventional method.

5.3. Traffic

Next, we consider the network traffic at the group management servers.

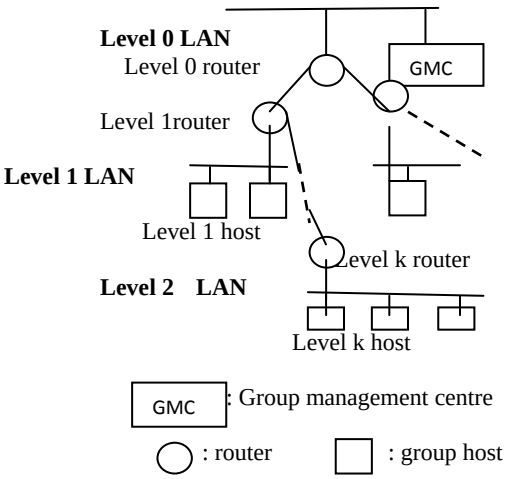


Fig.5. Network model for evaluations.

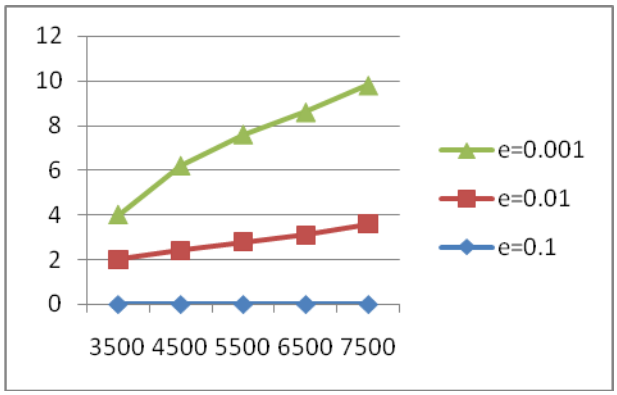


Fig.6 D_U/D_M ratio for host distribution

Table 2 Number of packets sent to/transmitted from the group management centre.

error rate	T_U	T_M		
		R=15	R=50	R=150
e=0.001	3006	9	14	19
e=0.01	3060	36	41	46
e=0.1	3600	306	311	316

From Fig. 2, the packets transmitted in key distribution sequence in the conventional method is estimated as

$$T_U=2H+2eH=2(1+e)\ H \quad (5)$$

On the other hand, the packets transmitted in key distribution sequence in the conventional method is estimated as

$$T_M=R+2eH \quad (6)$$

Where, R is the number routers. By assuming that e<<1 and 1/H<<1, the proportion of T_U to T_M is:

$$T_M/T_U= (R+2eH)/ (2H+2eH) =R/2H \quad (7)$$

So that we can understand number of packets sent to or transmitted are reduced very well.

6. Conclusions

In this paper, we defined a distributed system model for key distribution applicable to dynamic group communications and proposed an efficient key updating protocol. In this model, we introduced a subnet management server as a proxy on the same subnet to prevent concentration of data. We also used new method of re-keying which will reduce the number of new key generation by half. We also evaluated the proposed method on a multicast network model.

References

[1] S. Berkovits, "How to broadcast a secret," in *Adv. Cryptol.—Eurocrypt'91 (Lecture Notes in Computer Science)*. Berlin, Germany:Springer-Verlag, 1991, vol. 547, pp. 536–541.

[2] D. Boneh and A. Silverberg, "Applications of Multilinear Forms to Cryptography," in *Contemporary Mathematics*. Philadelphia, PA: Amer. Math. Soc., 2002, vol. 324, pp. 71–90.

[3] A. Fiat and M. Naor, "Broadcast encryption," in *Adv. Cryptol.Crypto'93 (Lecture Notes in Computer Science)*. Berlin, Germany:Springer-Verlag, 1993, vol. 773, pp. 480–491.

[4] M. T. Goodrich, J. Z. Sun, and R. Tamassia, "Efficient tree-based revocation in groups of lowstate devices," in *Adv. Cryptol.—Crypto'04(Lecture Notes in Computer Science)*. Berlin, Germany: SpringerVerlag, 2004, vol. 3152, pp. 511–527.

[5] D. Halevi and A. Shamir, "The LSD broadcast encryption scheme," in *Adv. Crytol.—Crypto'02 (Lecture Notes in Computer Science)*. Berlin, Germany: Springer-Verlag, 2002, vol. 2442, pp. 47–60.

[6] N.-S. Jho, J. H. Cheon, M.-H. Kim, and E. S. Yoo, Broadcast Encrytion 2005 [Online]. Available: http://eprint.iacr.org /2005/073

[7] N.-S. Jho, J. Y. Hwang, J. H. Cheon, M.-H.Kim, D. H. Lee, and E. S. Yoo, "One-way chain based broadcast encryption schemes," in *Adv. Cryptogr.—Eurocrypt'05 (Lecture Notes in Computer Science)*. Berlin, Germany: Springer-Verlag, 2005, vol. 3494, pp. 559–574.

[8] M. Burmester and Y. Desmedt, "A Secure and Efficient Conference Key Distribution System," Proc. Advances in Cryptology EUROCRYPT '94, pp. 275-286, 1994.

[9] M. Steiner, G. Tsudik, and M. Waidner, "Diffie-Hellman Key Distribution Extended to Group Communication," Proc. Third ACM Conf. Computer and Comm. Security (CCS '96), pp. 31-37, 1996.

[10] J. Katz and M. Yung, "Scalable Protocols for Authenticated Group Key Exchange," Proc. Advances in Cryptology—CRYPTO '03,pp. 110-125, 2003.

[11] Y. Kim, A. Perrig, and G. Tsudik, "Tree-Based Group Key Agreement," ACM Trans. Information and System Security, vol. 7, no. 1, pp. 60-96, Feb. 2004.

[12] M. Manulis, "Security-Focused Survey on Group Key Exchange Protocols," Report 2006/395, Cryptology ePrint Archive, http://eprint.iacr.org/, 2006.

[13] I. Chang, R. Engel, D. Kandlur, D. Pendarakis, and D. Saha, "Key Management for Secure Internet Multicast Using Boolean Function Minimization Techniques," Proc. IEEE INFOCOMM '99, vol. 2, pp. 689-698, Mar. 1999.

[14] D. Bruschi and E. Rosti, "Secure Multicast in Wireless Networks of Mobile Hosts: Protocols and Issues," Mobile Networks and Applications, vol. 7, no. 6, pp. 503-511, 2002.

[15] S. Zhu, S. Setia, S. Xu, and S. Jajodia, "Gkmpan: An Efficient Group Rekeying Scheme for Secure Multicast in Ad-Hoc Networks," Proc. IEEE Mobiquitos '04, pp. 42-51, 2004.

[16] Y. Sun, W. Trappe, and K.J.R. Liu, "A Scalable Multicast Key Management Scheme for Heterogeneous Wireless Networks," IEEE/ACM Trans. Networking, vol. 12, no. 4, pp. 653-666, Aug.2004.

[17] X.S. Li, Y.R. Yang, M. Gouda, and S.S. Lam, "Batch Updates of Key Trees," Proc. 10th Int'l World Wide Web Conf. (WWW10), May 2001.

[18] S. Setia, S. Koushish, and S. Jajodia, "Kronos: A Scalable Group Re-Keying Approach for Secure Multicast," Proc. IEEE Symp. Security and Privacy, pp. 215-228, 2000.

[19] W.H.D. Ng, M. Howarth, Z. Sun, and H. Cruickshank, "Dynamic Balanced Key Tree Management for Secure Multicast Communications," IEEE Trans. Computers, vol. 56, no. 5, pp. 577-589, May 2007.

[20] 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.

[21] M.H. Heydari, L. Morales, and I.H. Sudborough, "Efficient Algorithms for Batch Re-Keying Operations in Secure Multicast," Proc. 39th Ann. Hawaii Int'l Conf. System Sciences, vol. 9, 2006.

[22] H. Kurnio, S. Rei, and H. Wang, "Efficient Revocation Schemes for Secure Multicast," Proc. Int'l Conf. Information Security and Cryptology '01, pp. 160-177, Dec. 2001.

[23] M. Luby and J. Staddon, "Combinatorial Bounds for Broadcast Encryption," Proc. Advances in Cryptology—EUROCRYPT '98, pp. 512-526, 1998.

[24] A. Fiat and M. Naor, "Broadcast Encryption," Proc. Advances in Cryptology—CRYPTO '93, pp. 480-491, 1994.

[25] R. Poovendran and J.S. Baras, "An Information-Theoretic Approach for Design and Analysis of Rooted-Tree-Based Multicast Key Management Schemes," IEEE Trans. Information Theory, vol. 47, no. 7, pp. 2824-2834, Nov. 2001.

[26] C. Blundo and A. Cresti, "Space Requirements for Broadcast Encryption," Proc. Advances in Cryptology—EUROCRYPT, pp. 287-298, 1994.

[27] S. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-Secure Key Distribution for Dynamic Conferences," Proc. Advances in Cryptology—CRYPTO '92, pp. 471-486, 1992.

[28] J.H. Cheon, N. Jho, M. Kim, and E. Yoo, "Skipping, Cascade, and Combined Chain Schemes for Broadcast Encryption," IEEE Trans. Information Theory, vol. 54, no. 11, pp. 5155-5171, Nov. 2008.

[29] Y.-H. Chu, S.G. Rao, S. Seshan, and H. Zhang, "A Case for End System Multicast," IEEE J. Selected Areas in Comm., vol. 20, no. 8, pp. 1456-1471, Oct. 2002.

[30] B. Zhang, S. Jamin, and L. Zhang, "Host Multicast: A Framework for Delivering Multicast to End Users," Proc. IEEE INFOCOM, Mar. 2000.