PLEASE, 'P'-LEAder SElection for Multicast Group Communication

Mary Vennila S[1], Srinivasan S[2], Rangarajan T.C[3], Sankaranarayanan V[4] and Rhymend Uthariaraj V[5]

1 Senior Lecturer, Department of Computer Science, Presidency College, Chennai, India.
2 3 Student, Department of Information Technology, Anna University, Chennai, India.
4 Professor (Retd), Anna University, Chennai, India.
5 Professor, Department of Information Technology, Anna University, Chennai, India.

Summary: The problem for secure multicast group communication has been dealt with excessively, via both centralized and distributed approaches. LeaSel is one such particular secure multicast model for key management. This model preserves the forward and backward confidentiality and solves the 1 affects N problem. This paper proposes PLEASE ('p'-Leader Selection), a model based on this LeaSel multicast model. It introduces the concept of 'p' leaders for load sharing, increased robustness and added security**.**

***Keywords*:** LeaSel, Secure Multicasting, PLEASE,

Group communication

1. Introduction

Multicast is an internetwork service for group communication, using the multicast address. Though, it thus reduces sender transmission overhead, the problem of scalability arises when multicast data need to be securely transmitted [19].The data can be secured by encrypting it with the group key, shared among all the members of the group. But, whenever the group members join or leave during the course of a multicast session, group re-keying must be done, to preserve the forward and backward confidentiality. When there are frequent member changes, this also gives rise to scalability problem.

The available approaches can be grouped into three main classes viz. centralized, distributed subgroup and distributed approaches. Though the centralized approach [6, 8, 9, 11, 12, 13] achieves reliable and synchronized key distribution, the central server is a crucial single point of failure. The distributed subgroup approach [7, 14, 15, 18, 19] scales well for large groups but each sub-group controller is vulnerable to attack. The distributed approach [16, 17] is scalable but trusts all members of the group. For non-trusted members, this approach fails.

LeaSel is a scalable, secured distributed subgroup model [1, 2, 3, 4 5].The member who ranks first among the members will be designated as a leader and will be authorized to perform key generation and distribution. The deputy controller alone knows the leader and it is hidden from all other members of the subgroup, including the leader itself. Here, the authors propose *PLEASE*, incorporating the concept of 'p' leaders. This model, instead of electing a single leader, selects 'p' leaders of top remarks. This ensures a greater security and increased availability.

2. Secure Multicast Problems

Secure multicast communication involves issues like forward and backward confidentiality, as dealt in [1]. In a multicast group, whenever members join and leave during the course of a session, then the encryption key should be updated for every join and leave operation to prevent the former group member accessing the future communications (forward confidentiality) and a new member accessing the past communications (backward confidentiality). Moreover, when a member joins or leaves the group, it affects all other members of the group. This is referred as "1 affects n" scalability problem [19].Thus the essential components for secure multicast are group membership control, secure key distribution and secure data transfer [19].

3. Leasel Overview

LeaSel introduces two trusted entities called deputy controller (DC), one per subgroup and the controller (CR) to manage and control groups and subgroups. The deputy controllers manage a subgroup each, and the controller manages all deputy controllers. The controller participates in the creation of a multicast group session, but does not take any role during the key

management of that session. When a member joins the group, the controller performs authentication and after approval the deputy controller prepares a rank list for all its members.

In LeaSel, the components of secure multicast are performed by deputy controllers and leaders. LeaSel has been proven mathematically and has been verified through implementation [1, 2, 3, 4, 5]. The identification of leader is a critical issue in Leasel. Though the identity of the leader is hidden from group members, a proper traffic analysis can reveal the identity of the leader. Also, the leader single-handedly manages the key distribution process, thus further burdening it.

4. Please Multicast Model

The architecture, as per 'PLEASE' is presented in figure 1. This is an adopted version of LeaSel architecture, already proposed and proved for both wired and wireless environment [1, 2, 3, 4, 5].

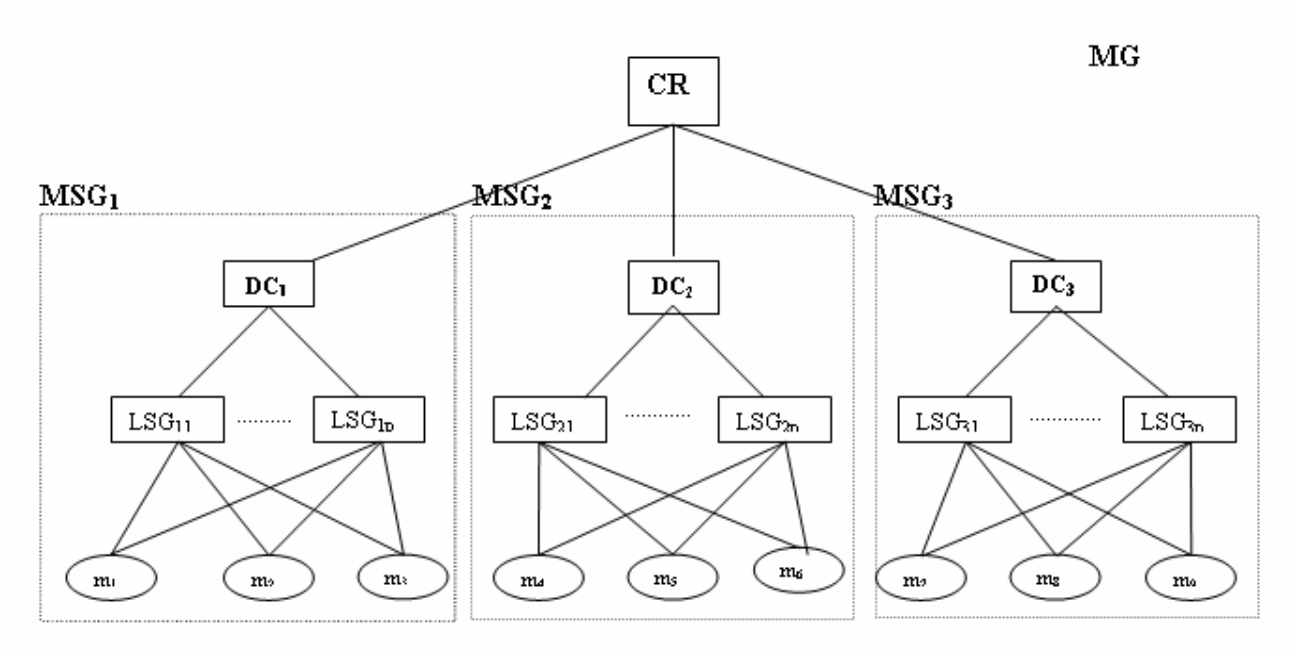

Figure 1. PLEASE Multicast model

The Leader selection is where PLEASE differs from LeaSel. Instead of a single leader, the DC selects a set of 'p' leaders. At a given time, only one of them acts as the leader and the leader is alternated for every transaction. Thus, the 'p'-Leaders share the Key Management work-load among them. Moreover, attacking this sub group becomes more difficult, as it involves attacking all the 'p' leaders, instead of one. Thus, the group key generation and distribution is not performed by any dedicated controller but instead by the 'p' leaders of the group and it is completely hidden from the group members [1, 2, 3, 4, 5]. Thus the model achieves high scalability with secure key generation and distribution.

5. Please Operational Overview

A detailed description of the operation of the PLEASE model is presented in this section.

Creation of group: The Controller distributes the individual member key k to all the members of the group in advance. Then the controller prepares the group access control list (GACL) and the subgroup access control list (SACL).The SACL is distributed it to the deputy controllers. The access control list contains the time duration and session for which the group member is authorized to receive the multicast data. The controller generates group key GK and shares it with deputy controller.

Single member ENTER protocol: When a new member joins a multicast session, the DC verifies its SACL and if approved, it updates the rank list and informs the one of the 'p'-leaders about this. The rank list is prepared by the deputy controller based on the membership permission for different sessions, remarks of deputy controller, capability to handle key generation and distribution etc. Since ranking the new member as the highest may produce a change in 'p'-leaders' set, PLEASE always rank the new member as rank 2. The selected 'p'-leader acts as leader and changes the subgroup key SK to SK`

to ensure forward and backward confidentiality and informs it to the current sub-group members and the new member. It generates the new Sub-group key SK`. Then, the 'p'-leader multicasts KEYUPDATE_ENTER message containing the newly generated subgroup key SK`, which is generated and encrypted with SK to the current members. The members obtain the new Shared key SK` from this message. The Leader then communicates SK` to the new member via separate secure channel. For single member ENTER,

Table 1: Single member Enter

No of Encryptions for key distributions	**2**
No of members who receive the Re-key-messages	$\mathbf{N_{SG}}$

KEYUPDATE_ENTER by LEADER :

HDR	$\{ SK` \}_{SK}$

Multiple members ENTER: This is similar to Single member ENTER, but more than one member join the sub-group. For multiple members ENTER ('y' number of members),

Table 2: Multiple Members Enter

No of Encryptions for key distributions	**y + 1**
No of members who receive the Re-key-messages	$\mathbf{N_{SG} + y}$

Single member EXIT: When a member wishes to leave the group, it sends EXIT request to the deputy controller. The deputy controller selects one of the 'p'-leaders as leader and forwards EXIT request to him .In scenarios where deputy controller wants to expel a member, it sends EXPEL message to the leader. In both the cases SK' needs to be changed, to preserve backward confidentiality. The leader computes the new SK` and sends it to each member encrypted with that member's individual key k_i. For single member EXIT,

Table 3: Single Member Exit

No of Encryptions for key distributions	$\mathbf{N_{SG} - 1}$
No of members who receive the Re-key-messages	$\mathbf{N_{SG} - 1}$

KEYUPDATE_EXIT LEADER:

HDR	$\{SK_{L2}`\}_{k1}$	$\{SK_{L2}`\}_{k2}$	….

Multiple members EXIT: This is similar to Single member EXIT, but more than one member leave the sub-group. For multiple members EXIT ('x' number of members),

Table 4: Multiple Members Exit

No of Encryptions for key distributions	$\mathbf{N_{SG} - x}$
No of members who receive the Re-key-messages	$\mathbf{N_{SG} - x}$

Message Transmission: The sender multicasts the message to the subgroup encrypted with the group key. The deputy controller receives this multicast message and directs it to any one of the 'p'-leaders, which decrypts them, and then re-multicasts to all the subgroup members encrypted with Subgroup key SK. Every authorized member decrypts the multicast message using the Subgroup key SK.

6. Mathematical model for PLEASE

In this analysis, four approaches for Key Management namely Centralized, Distributed, LeaSel and the PLEASE Model are compared, to prove the increased security of the proposed model.

Let N be the total number of members in the multicast group MG and C be the number of subgroups. Let n be the number of malicious attackers. Let there be at least one attacker in each subgroup. Every malicious attacker possesses 'breaking software' to attack the group.

Definition:

a) Fully armed malicious attacker: At least one software is capable of breaking the system.
b) Partially armed malicious attacker: No software is capable of breaking the system.

Let T_a and $T_{a'}$ be total number of fully armed malicious attackers and partially armed malicious attackers in the system, respectively. Let *a* and a' be the number of fully armed malicious attackers and partially armed malicious attackers in each subgroup, respectively. In all the following cases, it is assumed that i) Every malicious attacker are not fully armed with breaking software. ii) Every a has equal number of breaking software denoted by S_a and every a' has equal number of breaking software denoted by $S_{a'}$. Assume that n is equally distributed but a and a' are different in each subgroup such that a + a'= n.

If the number of attempts to successfully break the service is given by NA, then ,

Case 1:

Assumption: N is equally distributed

Centralized:

$$1 \le NA \le (S_{a'} * T_{a'}) + [T_a * (S_a - 1)] + 1 \quad \text{....Eq 1}$$

Distributed:

$$C \le NA \le (S_a \cdot * \sum_{i=1}^{C} a_i') + (S_a - 1) \sum_{i=1}^{C} a_i + C \quad \text{.....Eq 2}$$

LeaSel:

$$C \le NA \le (S_a \cdot * N/C) \sum_{i=1}^{C} a_i' + ((S_a - 1) * N/C) \sum_{i=1}^{C} a_i + (N/C - 1) \sum_{i=1}^{C} a_i + C \quad \text{.....Eq 3}$$

This is the complexity involved in finding the leader for a session.

PLEASE:

$$C \le NA \le (S_a \cdot * N/C) \sum_{i=1}^{C} a_i' p_i + ((S_a - 1) * N/C) \sum_{i=1}^{C} a_i p_i + (N/C - 1) \sum_{i=1}^{C} a_i p_i + C \quad \text{.....Eq 4}$$

This is the complexity involved in finding the leader for a single multicast message. Therefore to get control of the entire session, the hacker must make p_i consecutive successful attempts.

Case 2:

Assumption:

1) N are unequally distributed among subgroups. Let N_1, N_2…, N_C are number of members in subgroups MSG_1 , MSG_2 ,…. , MSG_C respectively.

Centralized:

$$1 \le NA \le (S_a \cdot * T_a \cdot) + [T_a * (S_a - 1)] + 1 \quad \text{....Eq 5}$$

Distributed:

$$C \le NA \le (S_a \cdot * \sum_{i=1}^{C} a_i') + (S_a - 1) \sum_{i=1}^{C} a_i + C \quad \text{....Eq 6}$$

LeaSel:

$$C \le NA \le (S_a \cdot * \sum_{i=1}^{C} a_i' N_i + (S_a - 1) \sum_{i=1}^{C} a_i N_i + (\sum_{i=1}^{C} a_i N_i - \sum_{i=1}^{C} a_i) + C \quad \text{.....Eq 7}$$

PLEASE:

$$C \le NA \le (S_a \cdot * \sum_{i=1}^{C} a_i' N_i p_i + (S_a - 1) \sum_{i=1}^{C} a_i N_i p_i + (\sum_{i=1}^{C} a_i N_i p_i - \sum_{i=1}^{C} a_i p_i) + C \quad \text{....Eq 8}$$

The equations prove that in PLEASE model, the malicious attacker takes even larger number of attempts to successfully break the service compared to LeaSel and other approaches,. Hence this model is difficult to break.

The careful examination of these mathematical equations reveals that in PLEASE, for both case 1 and case 2, the upper boundary depends on the total number of members in the group N(from Eq.4 and Eq.8). Thus for a group with large N, the computational complexity to successfully break the multicast service is O(pN), or O(N) itself.

7. Implementation

The proposed PLEASE model was implemented using ns2.26 to check for its performance. Simulations were done considering a group of 2000 nodes and the results were obtained. In the simulation, arbitrary hackers were introduced into the model and the complexity to compromise the system was found out. This reflects the robustness of the PLEASE model. These results were compared with the LeaSel model and a comparative performance graph was drawn.

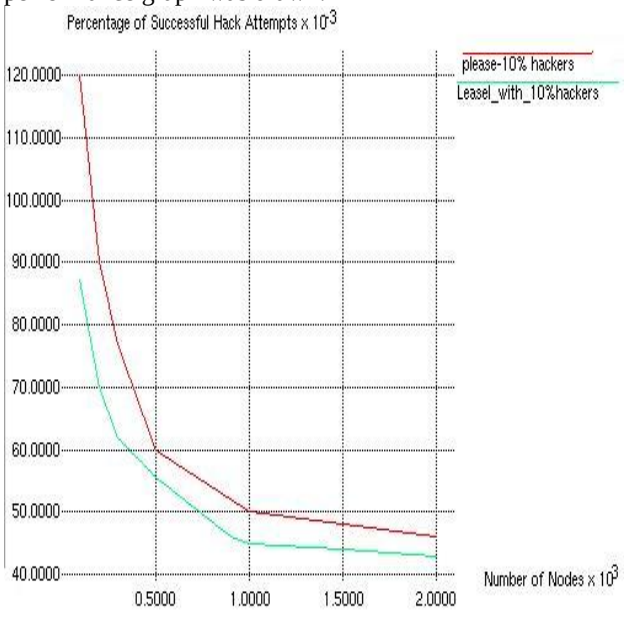

Figure 2.Security Improvement Graph for PLEASE

Also, the throughput of the system was calculated for the same number of nodes and was compared with the throughput of the LeaSel model.

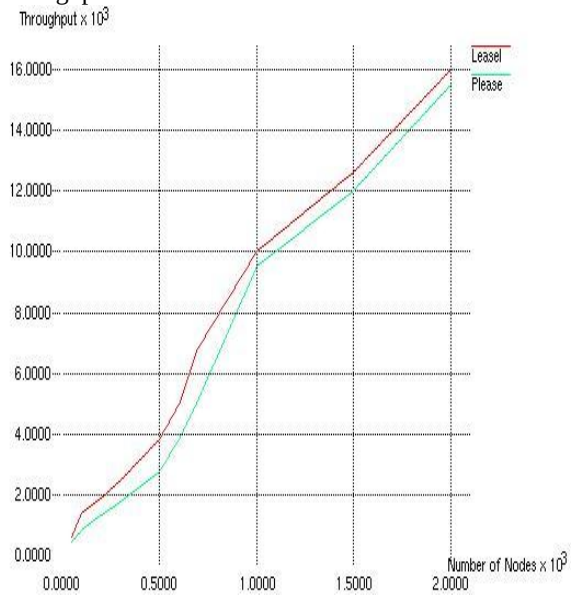

Figure 3 Throughput Graph for PLEASE

8. Advantages

(I)The revelation of the leader by observing the traffic flow (passive attack) becomes difficult as the leader among 'p'-leaders change for every message. (II)In case of the leader compromise, the new leader can immediately be chosen from the remaining 'p'-leaders. So, the complex procedures involved in recalculating the trust values and electing a new leader are and so there is a drastic decrease in time. (III) In order to compromise the sub group, an intruder has to compromise all the 'p'-leaders instead of a single leader, as with LeaSel. Thus, the complexity increased by a factor of 'p'.

9. PLEASE with Peer Leaders

Here, authors have proposed a Peer Leader Selection model as a variation of PLEASE. Peer Leader Selection model is a 'p'-leader model with p=2. But, as against 'p' leader, where only one of 'p' leaders is active at any time , here both the leaders take part in key generation and distribution process. Each peer leader generates one half of the key. At the members' end, they get both the halves from the peer leaders and form their new sub group key by combining them. In this case, any intruder outside the group needs to compromise both the peer leaders at the same time, in order to take control of the key generation and distribution function. But, this increased security comes with increased communication overheads. As both peer leaders are involved in key generation and distribution process, the number of communication messages becomes twice. Peer Leaders model will stand the best in applications which need a higher degree of security treatment and can bear these overheads.

Table 5: Peer Leaders Model

Message Types and number of communication messages involved	No of Encryptions for key distributions	No of Re-key messages
Single Member ENTER	4	$2N_{SG}$
Multiple members(y) ENTER	2(y + 1)	$2(N_{SG} + y)$
Single Member EXIT	$2(N_{SG} - 1)$	$2(N_{SG} - 1)$
Multiple Members(y) EXIT	$2(N_{SG} - x)$	$2(N_{SG} - x)$

10. Conclusion

The model was designed, simulated, tested and analyzed in terms of complexity, overheads and throughput, for all the multicast events in the wired environment. The results obtained were positive and satisfactory. This is an encouraging stride forward and future work will be aimed at optimizing the performance of the model in terms of computational complexity and extending the model to wireless and adhoc environment

11.References

[1] R. Elijah Blessing and V. Rhymend Uthariaraj, "LEASEL: AN EFFICIENT KEY MANAGEMENT MODEL FOR SCALABLE SECURE MULTICAST SYTEM" in Proceedings of ICORD 2002, DEC 2002, India.

[2] R.Elijah Blessing, "DESIGN AND ANALYSIS OF SECURE MULTICAST MODELS FOR WIRED AND MOBILE NETWORKS", PhD thesis submitted at Anna University, 2004.

[3] R.Elijah Blessing, V.Rhymend Uthariaraj, "EVALUATION AND ANALYSIS OF COMPUTATIONAL COMPLEXITY FOR SECURE MULTICAST MODELS", Springer Verlag 2003,

Vol.2668, Lecture Notes in Computer Science, pp 684-694.
[4] R. Elijah B1essing, V. Rhymend Uthariaraj, "FAULT TOLERANT ANALYSIS OF SECURE MULTICAST MODELS" **",** in Proceedings of IEEE International Conference ICICS-PCM 2003, Dec. 2003, Singapore
[5] R. Elijah Blessing, V. Rhymend Uthariaraj, "SECURE AND EFFICIENT SCALABLE MULTICAST MODEL FOR ONLINE NETWORK GAMES ", in Proceedings of 2nd International Conference on Application and Development of Computer Games, pp.480-491.
[6]C.Wong, M.Gouda, S.S.Lam, "Secure group communication using key graphs," IEEE/ACM Transaction on Networking, vol. 8, no.1, Feb 2000, pp.16-30.
[7]T.Ballardie, "SCALABLE MULTICAST KEY DISTRIBUTION," RFC 1949, May 1996.
[8].H.Harney and C.Muckenhirn, "GROUP KEY MANAGEMENT PROTOCOL (GKMP) ARCHITECTURE," RFC 2094, July 1997.
[9]D.M.Wallner, E.J.Harder and R.C.Agee, "KEY MANAGEMENT FOR MULTICAST: ISSUES AND ARCHITECTURES," RFC 2627, July 1997.
[10]T. Ballardie and J.Crowcroft, "MULTICAST-SPECIFIC SECURITY THREATS AND COUNTER-MEASURES," in Proc. Symposium on Network and Distributed system security, San Diego, California, February 1995, pp.2-16.
[11]T.Dunigan and C.Cao, "Group key management," Experimental, July 1997.
[12].R. Poovendran, S. Ahmed, S. Corson, and J. Baras, "A SCALABLE EXTENSION OF GROUP KEY MANAGEMENT PROTOCOL," Technical Report TR 98-14, Institute for Systems Research, 1998.
[13]D. Balenson, D. McGrew, and A. Sherman, "KEY MANAGEMENT FOR LARGE DYNAMIC GROUPS: ONE-WAY FUNCTION TREES AND AMORTIZED INITIALIZATION," IETF Internet draft, August 2000.
[14] S. Setia, S. Koussih, and S. Jajodia, "KRONOS: A SCALABLE GROUP RE-KEYING APPROACH FOR SECURE MULTICAST," In 2000 IEEE Symposium on Security and Privacy, California, May 2000, pp 215-228.
[15]B. Briscoe, "MARKS: ZERO SIDE EFFECT MULTICAST KEY MANAGEMENT USING ARBITRARILY REVEALED KEY SEQUENCES," In First International Workshop on Networked Group Communication, November 1999.
[16]M. Steiner, G. Tsudik, and M. Waidner, "DIFFIE-HELLMAN KEY DISTRIBUTION EXTENDED TO GROUP COMMUNICATION," In Proceedings of 3rd ACM Conference on Computer and Communications Security, New Delhi, March 1996.
[17]M. Steiner, G. Tsudik, and M. Waidner, "KEY AGREEMENT IN DYNAMIC PEER GROUPS. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS," vol. 11, no. 8, August 2000, pp. 769-780.
[18]T. Hardjono, B. Cain, and N. Doraswamy, "A FRAMEWORK FOR GROUP KEY MANAGEMENT FOR MULTICAST SECURITY," IETF Internet draft, August 2000.
[19] S.Mittra, "IOLUS: A FRAMEWORK FOR SCALABLE SECURE MULTICASTING," Proc. of ACM SIGCOMM '97, pp.277-288.

Mary Vennila .S received the M.Sc (Computer Science) from Bharathidasan University and M.Phil (Computer Science) from Mother Therasa University. She is now working as a Senior Lecturer in the Department of Computer Science in Presidency College India. Her research area Includes Network Security, Adhoc Networks, and Grid Technology

Srinivasan .S is pursuing the final year Bachelor of technology degree in Information technology, at Madras Institute of Technology, Anna University, Chennai, India. He has also developed the Telemetry Module for Enertec Satellite device as a part of ISRO MicSat project at MIT. His research interests include Grid Technology, Network Security and Distributed Computing

Rangarajan T.C is now a final year student aspiring for Bachelor of technology degree in Information technology, at Madras Institute of Technology, Anna University, Chennai, India. His research interests include Grid Technology, Network Security and Distributed Computing.

Dr. V Sankaranarayanan received the PhD from Indian Institute of Technology, Chennai, India. He had worked as a Professor at Anna University, Chennai, India and the last position he held was Director, Tamil Virtual University, Chennai, India. He has completed many commendable projects.His research area includes Computer Networks, OOP and Optimization.

Dr V Rhymend Uthariaraj received the M.E (Computer Science and Engineering) and PhD, from Anna University, Chennai, India. He is now working as a Professor and Head, Department of Information Technology Anna University, Chennai, India. His research interests include Network Security, Pervasive computing and Optimization.

.