Comparison of Scalable Key Distribution Schemes for Secure Group Communication

Lakshminath R. Dondeti
University of Nebraska-Lincoln

Sarit Mukherjee
Panasonic Technologies

Ashok Samal
University of Nebraska-Lincoln, asamal1@unl.edu

COMPARISON OF SCALABLE KEY DISTRIBUTION SCHEMES FOR SECURE GROUP COMMUNICATION

Lakshminath R. Dondeti
University of Nebraska-Lincoln
115 Ferguson Hall, Lincoln, NE 68588-0115
ldondeti@cse.unl.edu

Sarit Mukherjee
Panasonic Technologies Inc.,
2 Research Way, Princeton, NJ 08540
sarit@research.panasonic.com

Ashok Samal
University of Nebraska-Lincoln
115 Ferguson Hall, Lincoln, NE 68588-0115
samal@cse.unl.edu

Abstract

Scalable secure key distribution is the most important feature of a scalable secure group communication protocol. Most of the existing scalable secure group communication protocols are based on a hierarchical key distribution tree. These schemes can be classified as hierarchical node based schemes and hierarchical key based schemes. In this paper, we compare recently proposed hierarchical key distribution schemes through simulation using real-life multicast group membership traces. Our simulations show that hierarchical node based approaches better distribute encryption cost among the entities of a multicast group. However, hierarchical node based schemes "trust" internal nodes of a key distribution tree. We show that the dual encryption protocol recently proposed by us overcomes the aforementioned limitation of hierarchical node based schemes, with a marginal performance penalty.

1. Introduction

Multicasting is a scalable way of transmitting data to a group of hosts. Several multicast applications, including data streaming applications, collaborative applications may require secure data transmission [5]. Members of a multicast session must not be able to access the multicast data transmitted before their membership has begun or after their membership has expired. Thus, in dynamic multicast groups, where members join and leave during the multicast session, the secret keys need to be updated each time the membership changes. Scalability in this context implies that the overhead involved in key updates, data transmission and encryption must be independent of the size of the multicast group. The other requirement of scalability is that the addition or removal of a host from the group must not affect all the members of the group. This requirement is called "1 affects n" scalability problem [6].

Several protocols have been proposed to support scalable secure multicasting [2, 4, 6, 8, 9, 10]. Most of these protocols distribute encryption keys via a distribution tree. We can classify the tree-based approaches into two groups. The first class uses a hierarchy of keys [4, 8, 9, 10] while the second uses a hierarchy of nodes [2, 6] to achieve scalability. The hierarchical key based schemes suffer from the 1 affects n scalability problem. Some hierarchical node based schemes [6] entrust internal nodes of the key-distribution tree with the distribution of the encryption keys. But they offer no mechanism to hide secure multicast data from the internal nodes. We recently proposed a dual encryption protocol (DEP) [2] which provides the capability to deny access of secure multicast data to third party entities.

In this paper, we compare the en(de)cryption cost at the sender, members and internal nodes (where applicable) of the key distribution tree in the hierarchical approaches, through simulation. In particular, we compare the protocols' performance as the multicast group sizes increase. We use real-life multicast traces [1] of a few multicast sessions in the MBone to simulate real world behavior. Our simulations show that hierarchical node based schemes incur less encryption cost than hierarchical key based schemes. The node based schemes also better distribute the cost among the entities of a key distribution tree and their performance benefits increase with group size. We show that DEP incurs only a marginal increase in encryption cost while eliminating the need to trust third parties.

The rest of this paper is organized as follows. In Section 2 we provide a classification of scalable secure multicast protocols used in our comparison study. We characterize the workload and describe it in Section 3. Section 4 describes the simulation results in detail. The final section summarizes our conclusions.

0-7803-5796-5/99/$10.00 © 1999 IEEE

2. Classification of secure multicast protocols

Most of the previous work in the area of secure multicasting has been in key distribution. In a majority of the proposed scalable key distribution schemes [2, 6, 8, 9, 10], the members of the multicast group are part of a tree-like hierarchical structure. We classify these protocols into hierarchical key based schemes and hierarchical node based schemes. In hierarchical key based schemes, the sender or a group manager distributes a set of key encrypting keys (KEK) to each member, based on the member's location in the tree. The sender uses the KEKs to securely send new KEKs and the session key to members. Hierarchical node based schemes employ internal nodes of the tree as subgroup managers (SGM), which assist in key distribution. These internal nodes may not be members of the multicast group. SGMs also forward data encryption keys (DEK) received from the sender, to their subgroup members. While hierarchical node based schemes [6] support distributed group management, they expose secret keys to the internal nodes, which may be third party entities.

For our comparison study, we choose one hierarchical key based scheme, the Centralized Tree-Based Key Management scheme (CTKM) [8, 9, 10], and one hierarchical node based scheme, Iolus [6]. Since Iolus "trusts" internal nodes of a key distribution tree, we include DEP [2], which does not trust the internal nodes, in our comparison. CTKM has been proposed independently with minor variations by several research groups. For our study we use the protocol as described by Wong et. al [10]. We compare the characteristics of the aforementioned protocols in Table 1.

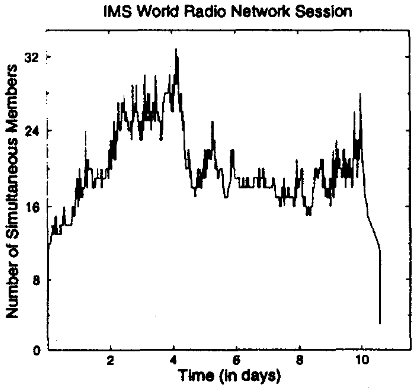

Figure 1. Group size in IMS session

3. Workload characterization

In order to simulate real world behavior, we use real-life multicast group membership traces collected by Almeroth *et. al* [1] as our workload. In Figures 1 and 2 we plot the multicast group size in real-life traces collected from various multicast sessions[1].

The sessions differ in inter-arrival rates of members, membership durations and popularity of sessions as indicated by the number of simultaneous members in the sessions. Notice that the popularity of a session plays an important role in the multicast key distribution as it governs the size of the distribution tree. In the rest of the discussion, we use activity in a session and popularity of a session interchangeably.

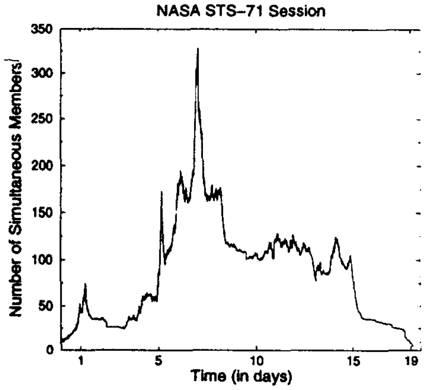

Figure 2. Group size in STS-71 session

In the IMS World Radio Network Session the number of simultaneous members is more than 15 but less than 40 (refer to Figure 1). The session lasted more than a week [1]. The NASA shuttle mission session STS-71 broadcasting the docking of the Space Shuttle and Space Station Mir over the MBone [1] is a more popular session. For most of the time the number of simultaneous members is more than one hundred, and more than three hundred at its peak (shown in Figure 2). This session lasted more than two weeks.

4. Performance comparison using simulation

In this section we compare the scalable key distribution approaches through simulation. We develop a simulation model in C using the CSIM [7] simulation package. In the model we simulate the join and leave operations following each of the three key distribution schemes and DEK distribution in case of DEP and Iolus.

Recall that scalable encryption cost is one of the most important requirements of a key distribution scheme. Therefore, we compute the per session encryption cost at the sender and constitute the comparative study. In case of CTKM, we build a virtual key distribution tree and use it as a reference to determine the number of encryptions necessary at the sender during each join or leave. To ensure a

[1] For a complete version of our study with additional workloads and a more detailed analysis of the performance of the protocols refer to [3].

Table 1. Comparison of scalable secure multicast protocols

	Iolus	CTKM	DEP
No. of keys in the multicast group	$n+l+1$	$\frac{dn-1}{d-1}$	$n+l+1+c$
≈	$O(n)$	$O(n)$	$O(n)$
No. of keys managed by the sender	2	$\frac{dn-1}{d-1}$	$c+2$
No. of keys at a member	3	$O(\log_d n)$	4
No. of keys at an SGM	4	—	5
Public key/ Secret key	Both	Secret	Both
Scalable Encryption Cost	Yes	Yes	Yes
1 affects n scalability	Yes	No	Yes
No. of messages at join	$O(1)$	$O(\log_d n)$	$O(1)$
No. of messages at leave	$O(\bar{l})$	$O(d\log_d n)$	$O(\bar{l})$
Total key encryptions during data transmission	$O(l)$	$O(1)$	$O(l+c)$
No. of key encryptions at the sender	$O(1)$	$O(1)$	$O(c)$
Intermediate nodes	Trusted	Not trusted	Not trusted

n: number of members l: number of subgroups d: degree
c: size of the sender's subgroup $\bar{l}$: average size of a subgroup

fair comparison, we assume that the SGMs in DEP and Iolus are third party entities. Thus, in case of DEP all SGMs are participant SGMs. The sender and the SGMs in DEP and Iolus incur local subgroup management costs. Additionally, the senders in DEP and Iolus incur DEK encryption costs, the SGMs incur DEK translation costs and the members incur DEK decryption costs. The sender in DEP also incurs a unit cost per join in the session, due to KEK distribution. Correspondingly each member incurs a one-time unit decryption cost. Finally, each member of a subgroup incurs a unit decryption cost each time a join or leave occurs in the subgroup.

In each simulation the number of encryptions and/or decryptions performed by the sender the members and the SGMs were observed. We plot the encryption cost versus the degree of the key distribution tree for each of the above three schemes. We analyze those plots in the following.

Each scheme was simulated using the workload described earlier. The performance metrics shown are the join, leave and the total encryption costs per session at the sender. In the following we present and discuss the results for each of these metrics.

4.1. Encryption cost at the sender

The per session encryption overhead at the sender for the protocols is plotted in Figure 3. The per session join cost depicts the cost due to joins at the sender's subgroup and the KEK distribution cost. In the figure the rows represent the workload used for the simulation, and the columns show the number of encryptions during join, leave and the total, respectively.

Observe from Figure 3 (column (i)) that the join cost at the sender in CTKM decreases very sharply, and then shows more smooth decrease. Join cost at the sender in CTKM is approximately equal to $2(\log_d n)$, where d is the degree of the key distribution tree and n is the group size [10]. This expression decreases with increasing degree which explains the CTKM curves in Figures 3.(i). Figure 3 (column (i)) also indicates that the per session cost at the sender due to joins increases with degree in DEP and Iolus. The per session join cost at the sender is dependent on the size of the sender's subgroup, which increases with degree. It also increases with frequency of joins at the sender. The join cost in DEP also includes the KEK distribution cost which is one encryption per member in the session. In our simulations, the join cost per session at the senders in DEP and Iolus was much lower than the cost at the senders in CTKM. More importantly, the gap between the curves corresponding to DEP (Iolus) and CTKM widens as the the number of simultaneous members increases. This shows that DEP (Iolus) can scale very well to multicast sessions with large group sizes.

Figure 3.(ii) shows that leave cost per session increases with degree. At low degree, for both workloads leave cost in CTKM is higher than that in DEP and Iolus. In the IMS session, for higher values of degree, CTKM performs better than DEP and Iolus. However, DEP and Iolus perform better than CTKM in STS-71, the more active session. The per leave encryption cost at the sender in CTKM is approximately $d(\log_d n)$ whereas in DEP and Iolus it is proportional to size of the top level subgroup. That explains the increase in cost as the degree increases. Also in CTKM the sender is responsible for the key changes during all leaves, while in DEP and Iolus, the sender changes keys only when its local subgroup members leave.

Figure 3.(iii) shows the total cost per session. The total cost at the sender during a session in CTKM schemes is

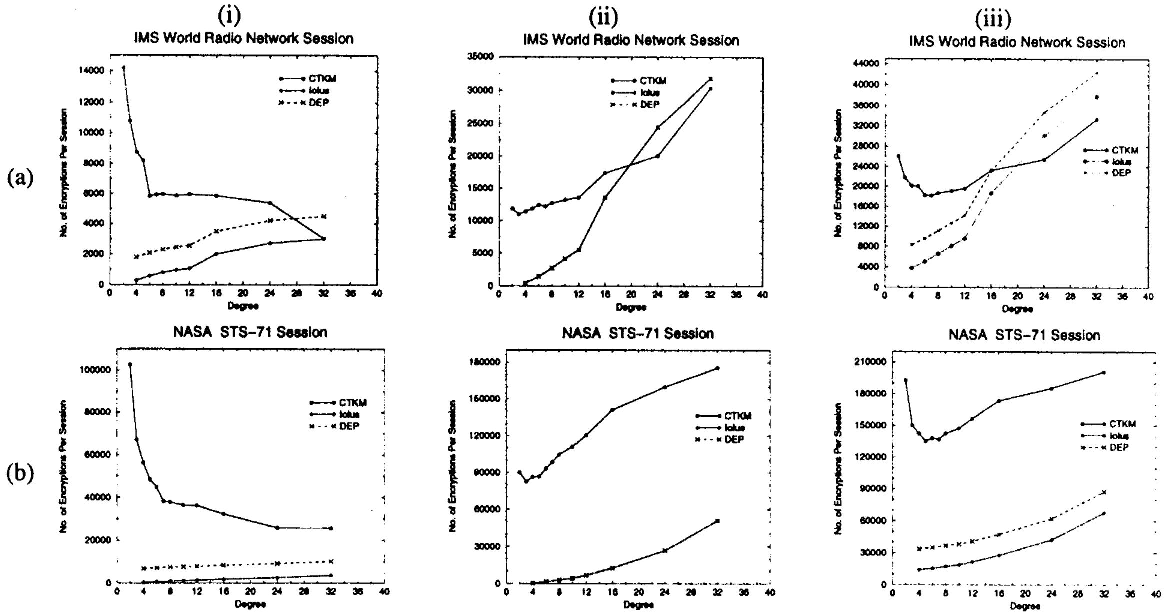

Figure 3. Encryption cost at the sender during the session

(i) Number of encryptions at the sender due to joins vs. degree
(ii) Number of encryptions at the sender due to leaves vs. degree
(iii) Total number of encryptions at the sender vs. degree

the sum of the cost during joins and leaves. The senders in DEP and Iolus also incur encryption overhead due to DEK distribution in addition to the overhead corresponding to the management of the top level subgroup. For a fair comparison, the DEK in DEP (Iolus) must be changed each time a join or leave occurs. Thus, we change the DEK approximately as many times as there are joins and leaves, in our simulations[2]. Finally, in simulating DEP, we use a single KEK for the whole group.

Figure 3.(iii) indicates that the total cost at the sender in CTKM increases with degree after an initial dip. This dip indicates that the optimal degree of key distribution tree for CTKM is 4, 5 or 6. The cost in DEP and Iolus increases with degree. Note that DEK distribution cost is independent of degree. The increase in cost with degree is due to the increase in subgroup size at the sender with degree. In the IMS session (small group size), the cost in DEP and Iolus was lower than the cost in CTKM for low values of degree. At higher values of degree, CTKM performed better than DEP and Iolus. In the STS-71 session (bigger group size) the cost per session is significantly lower in DEP and Iolus than that in CTKM for all values of degree. Also, with increasing group sizes, the gap between the cost curves corresponding to CTKM and DEP (Iolus) increases, showing that DEP (Iolus) scales better.

4.2. Distribution of en(de)cryption cost

We compute the distribution of total en(de)cryption cost at the sender, members and the SGMs in all three schemes and plot them as percentages in histograms shown in Figure 4. The rows correspond to the workloads while the columns correspond to CTKM, Iolus and DEP, respectively. These graphs show that the sender in CTKM incurs a larger percentage of cost compared to the sender in DEP (Iolus). This is because DEP distributes the encryption cost between the sender and the SGMs whereas CTKM burdens the sender with all the encryption cost.

4.3. Summary

We conclude this section with a summary of our observations in comparing the three scalable secure multicasting protocols.

- Hierarchical node based protocols incur less encryption cost compared to hierarchical key based protocols. They also distribute the cost "evenly" among the entities of a multicast group.
- Hierarchical node based schemes keep the per member cost at the sender independent of the multicast group size, whereas cost in hierarchical key based schemes

[2] Note that this corresponds to the worst case in DEP (Iolus). Realistically, the sender changes the DEK at a frequency dictated by the security requirements and performance constraints.

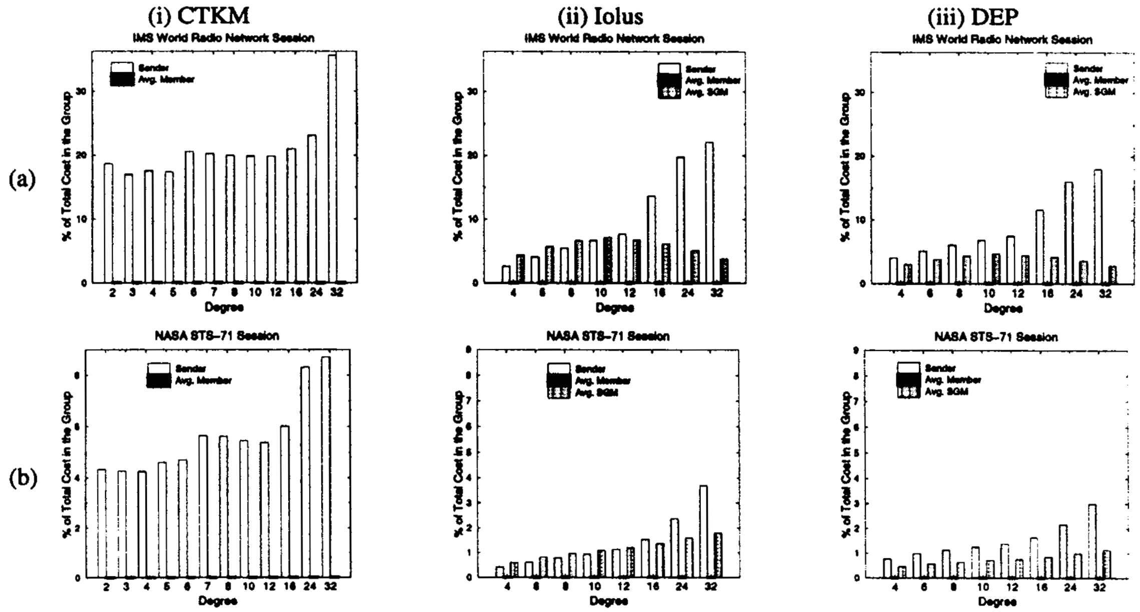

Figure 4. Distribution of en(de)cryption cost

increases with group size. Consequently, Iolus and DEP scale much better than CTKM as the number of simultaneous members in a multicast session increases.

- Unlike in Iolus, DEP can do away with the so called "trusted" third parties (e.g., participant SGMs) incurring marginally more aggregate cost than Iolus.

5. Conclusion

We compared encryption/decryption cost incurred in various secure group communication protocols using real-life group membership data. We conclude that hierarchical node based approaches fare better than hierarchical key based approaches. The performance advantage of hierarchical node based approaches increases with the multicast group size. While most hierarchical node based approaches automatically give access of secure multicast data to third party hosts which assist in subgroup management, DEP avoids that drawback using dual encryption. Although DEP incurs marginal increase in cost due to dual encryption, it is more secure than other hierarchical node based approaches, while still delivering better performance than hierarchical key based approaches.

Acknowledgment

We thank Dr. Kevin Almeroth for giving us access to the MBone multicast group membership data he collected.

References

[1] K. Almeroth and M. Ammar. Characterization of MBone Session Dynamics: Developing and Applying a Measurement Tool. Technical Report GIT-CC-95-22, Georgia Institute of Technology, June 1995.

[2] L. R. Dondeti, S. Mukherjee, and A. Samal. A Dual Encryption Protocol for Scalable Secure Multicasting. In *Fourth IEEE Symposium on Computers and Communications*, pages 2–8, Red Sea, Egypt, July 1999.

[3] L. R. Dondeti, S. Mukherjee, and A. Samal. Secure Group Communication Using Dual Encryption: its Security and Performance Analysis. Technical Report UNL-CSE-1999-001, University of Nebraska-Lincoln, February 1999. Submitted for Publication.

[4] D. A. McGrew and A. T. Sherman. Key establishment in large dynamic groups using one-way function trees. *Submitted to IEEE Transactions on Software Engineering*, May 1998.

[5] C. K. Miller. *Multicast Networking and Applications*. Addison Wesley Longman, Inc., September 1998.

[6] S. Mittra. Iolus: A Framework for Scalable Secure Multicasting. In *Proc. ACM SIGCOMM*, pages 277–288, Cannes, France, September 1997.

[7] H. Schwetman. Introduction to process-oriented simulation and CSIM. In *Proceedings of the 1990 Winter Simulation Conference*, 1990.

[8] M. Waldvogel, G. Caronni, D. Sun, N. Weiler, and B. P. ner. The VersaKey Framework: Versatile Group Key Management. *JSAC Special Issue on Service Enabling Platforms For Networked Mu ltimedia Systems*, August 1999.

[9] D. Wallner, E. Harder, and R. Agee. Key Management for Multicast: Issues and Architecture. IETF Draft, July 1997.

[10] C. K. Wong, M. Gouda, and S. S. Lam. Secure group communications using key graphs. In *Proc. ACM SIGCOMM*, August 1998.