A Broadcast-Encryption-Based Key Management Scheme for Dynamic Multicast Communications Work-in-Progress

Chin-Chen Chang
Department of Information Engineering and Computer Science
Feng Chia University
100 Wenhwa Rd., Seatwen, Taichung 40724, Taiwan, R.O.C.
886-4-24517250-3790
ccc@cs.ccu.edu.tw

Yu-Wei Su
Department of Computer Science and Information Engineering
National Chung Cheng University
168 University Rd., Min-Hsiung, Chia-Yi 621, Taiwan, R.O.C.
886-4-24517250-3790
syw@itri.org.tw

Iuon-Chang Lin
Department of Management Information Systems
National Chung Hsing University
250 Kuo-Kuang Rd., Taichung, 402, Taiwan, R.O.C.
886-4-22840864-609
iclin@nchu.edu.tw

ABSTRACT
Broadcast encryption is an efficient and secure method to allow a user to be able to communicate securely with multiple receivers in an insecure broadcast environment. Previously, scholars usually applied tree structures to design multicast key management schemes, in which the distributed key was used to encrypt the transmitted message. However, in order to achieve forward secrecy and backward secrecy, previous tree-structure-based approaches had to conduct the rekeying process when a member joined or left a multicast group. However, the rekeying process usually requires that other members change their cryptographic keys. Consequently, this proved to be inconvenient and impractical. In this paper, we propose an efficient multicast key management scheme to solve this problem. Compared with previous tree-structure-based multicast key management schemes, the benefits of our scheme include eliminating the rekeying process and reducing the required key storage for each member.

Keywords
Multicast key management, broadcast encryption

1. INTRODUCTION

1.1 Broadcast Encryption
Broadcast is an efficient method to deliver messages to a number of receivers simultaneously. No matter who sends a message, all members could receive all transmitted data in a broadcast environment. This property makes the transmitted data unconcealed and public to all users. How to broadcast securely was first presented in [1] and formally defined in [2]. Members in a broadcast environment are classified into legal multicast groups or illegal multicast groups. And groups are dynamically organized according to the authorized receivers in different sessions. Berkovits applies secret sharing to hide the secure messages [1]; so only the receivers in legal multicast groups possess the essential parameters to retrieve secure messages. Broadcast encryption is another strategy to protect the transmitted data in an insecure broadcast environment. The sender uses a cryptographic key to encrypt the confidential data according to the multicast group, and the receivers in the multicast group can decrypt the received data by using the common shared cryptographic key, but illegal receivers cannot retrieve the original data. However, how to distribute the common shared cryptographic key is a key problem in multicast communications. An intuitive approach is that the sender encrypts the shared cryptographic key by using every receiver's individual key. But this approach increases the sender's computation overheads and makes the network bandwidth requirements up especially when the number of receivers is great.

1.2 Tree-Structure-Based Key Management Using Broadcast Encryption
In order to overcome the problems of high computation overheads and large network bandwidth requirements, there are several tree-based key management schemes [2, 3, 4, 5, 6, 7, 8, 9, 10] using broadcast encryption which have been proposed. In a tree-structure-based key management scheme for broadcast, each receiver is treated as a leaf node in the tree structure, and each node in the tree structure is assigned a unique key. Every receiver holds the keys along its ancestor path. Before the sender transmits a secret message to the authorized receivers, the sender has to employ the associated key to encrypt the secret message. Because every receiver holds the assigned key of the root of its subtree, it can successfully decrypt the encrypted message. However, most of the tree-structure-based key management schemes encounter rekeying problems. While a new receiver joins the broadcast group, some keying information has to be assigned to the new receiver. The new receiver can not obtain the content of the previous communications; so all keys along the new receiver's ancestor path should be updated, and other relative receivers will be informed. Similarly, whenever a member leaves the broadcast group, all keys along the leaver's ancestor cannot be used in future communications. Therefore, the relative receivers need to update their keys as well. The key storage and rekeying communications become the major problems in such key management schemes for multicast communications.

1.3 Goals
Our goal is to design a novel broadcast-encryption-based key management scheme for dynamic multicast communications. The scheme aims to eliminate the rekeying process and reduces the key storage for each member, and it must satisfy the following requirements. 1. The members not belong to the multicast group

hardly retrieve the encrypted messages. In other words, they should not obtain the session key. 2. The members in a multicast group can decrypt the encrypted messages by the derived from session key. 3. Every member cannot obtain other member's secrets. The multicast group is dynamic in different sessions.

2. Fundamental Idea

Assume that the number of the broadcast group members is U, and a set denotes the broadcast group, where $U = \{u_1, u_2, ...u_n\}$. The sender prearranges the seeds s_{u_i} to the receiver u_i, for $i = 1,2,...,n$. The sender also announces a well-known one way hash function $H(*)$. The sender announces a symmetric encryption algorithm $E(*)$ as the base technique for encrypting messages. $E_K(M)$ denotes that a message M is encrypted with a session key K. U_m denotes a multicast group, where $U_m \in U$. The sender arbitrarily selects a prime p_s, and a random number X, then broadcasts $\{B, X, E_K(M)\}$, where $B = \left(p_s \prod_{u_i \in U_m} H(s_{u_i} \| X) \right) + K$. For a receiver $u_x \in U_m$, u_x learns K by $K = B \bmod H(s_{u_x} \| X)$ and retrieves M by decrypting $E_K(M)$ with K. Note that U_m is a dynamic multicast group. As a result, members of U_m are different in distinct sessions, and the sender uses different session key to encrypt messages. The session key K must satisfy $K < \min\{H(s_{u_i} \| X) \| u_i \in U_m\}$.

3. Costs Evaluation

Table 1. Costs Comparisons between the Tree-Structure-Based Schemes and Our Approach with Tree-Structure-Based Scheme

	Tree-Based Scheme	Our Approach
(1)	$(\log_d + 1)k$	s
(2)	$\left(\frac{dn-1}{d-1}\right)k$	nk
(3)	$\log_d n$	0
(4)	$k \sim \left(\frac{n}{2}\right)k$	ml

(1): The Storage Costs of Each Member
(2): The Storage Costs of the Center
(3): The Communication Costs for Members Join/Leave
(4): The Extra Communication Costs for Deriving Session Keys

k : the length of the session key
s : the length of secret s_{u_i}
n : the number of all members
d : the degree of the tree structure
m : the size of the legal subgroup
l : the output length of the adopted hash function

The scalable problem is overcome, and the required key storage for each member is reduced and independent of the number of members. Besides, the measurement results show that our scheme can be implemented with low computation costs. Summarizing this section, we achieved two major goals. 1. Eliminating the rekeying process. 2. Reducing the required key storage for each member.

4. Conclusions

Previously, tree-structure-based schemes are wildly employed to broadcast securely. Compared with tree-structure-based methods, our approach has more benefits to overcome the scalable problem and to reduce each member's required key storage. When members join or leave, other members are unaffected. Because our scheme is scalable, it is especially suitable to be applied to the applications, in which the frequency of members joining or leaving is very high..

5. REFERENCES

[1] S. Berkovits. *How to Broadcast a Secret.* Proceedings of Advances in Cryptology---EUROCRYPT 1991, 535-541.

[2] A. Fiat and M. Naor. *Boradcast Encryption.* Proceedings of Advances in Cryptology---CRYPTO 1992, 480-491.

[3] R. Canetti and J. Garay and G. Itkis and D. Micciancio and M. Naor and B. Pinkas. *Multicast Security: A Taxonomy and Some Efficient Constructions.* Proceedings of IEEE Infocom 1999, 708-716.

[4] R. Canetti and T. Malkin and K. Nissim. *Efficient Communication-Storage Tradeoffs for Multicast Encryption.* Proceedings of Advances in Cryptology---EUROCRTPT 1999, 459-474.

[5] H. Lu. *A Novel High-Order Tree for Secure Multicast Key Management.* IEEE Transactions on Computers, Vol. 54, No. 2, 2005, 214-224.

[6] S. Mittra. *Iolus: A Framework for Scalable Secure Multicasting.* Proceedings of ACM SIGCOMM 1997, 277-288.

[7] D. Naor and M. Naor and J. B. Lotspiech. *Revocation and Tracing Schemes for Stateless Receivers.* Proceedings of Advances in Cryptology---CRYPTO 2001, 41-62.

[8] R. Nojima and Y. Kaji. *Using Trapdoor Permutations in a Complete Subtree Method for Broadcast Encryptio.* IEICE Transaction on Fundamentals, Vol. E88-A, No. 2, 2005, 4 568-574.

[9] D. M. Wallner and E. J. Harder and R. C. Agee. *Key Management for Multicast: Issues and Architectures.* RFC, No. 2627, 1999.

[10] C. K. Wong and M. Gouda and S. S. Lam. *Secure Group Communication Using Key Graphs.* IEEE/ACM Transactions on Networking, Vol. 8, No. 1, 2000, 16-30.