Journal of Computer Science

Original Research Paper

Hybrid Broadcast Group Management Protocol for Secure, Scalable and Efficient Group Communication

[1]Sayee Kumar, M. and [2]T. Purusothaman

[1]*Muthayammal Engineering College, Rasipuram, Tamilnadu, India*
[2]*Government College of Technology, Coimbatore, Tamilnadu, India*

Article history
Received: 14-03-2014
Revised: 10-07-2014
Accepted: 01-09-2014

Corresponding Author:
Sayee Kumar, M.
Muthayammal Engineering College, Rasipuram, Tamilnadu, India
Email: sayee.academic@gmail.com

Abstract: In hasty growth of communication, security plays a central role in maintaining confidentiality of data in group communication. Keeping the data intended for the group in confidential manner is the most important security feature need to be sustained for the group communication. An efficient group key management mechanism named as Hybrid Broadcast Group Management Protocol (HBGMP) is devised based on the Reverse Function (RF) and Chinese Remainder Theorem (CRT). The distinctive security among the subgroups is ensured by the reverse function and the session ID of each subgroup is calculated by employing Chinese remainder theorem. By contraption, the Session ID using Chinese Remainder Theorem, with which a cohort of n users requires Sub Group Service Provider (SGSP) to do O (n/m) computation for communication and the storage cost are diminished by diverting the computing load of the Group Service Provider (GSP) into the SGSP. The significance of this protocol is the group member needs to store only two different values during the entire life span and also the rekey message is broadcasted which brings down the communication cost to O(1). The protocol is defined generally for any applications in hybrid architecture. The proposed architecture using CRT and Reverse function is scalable for hefty sized dynamically changing group.

Keywords: Chinese Remainder Theorem, Group Key Management Protocol, Group Service Provider, Hybrid Broadcast Group Management Protocol, Reverse Function, Sub Group Service Provider

Introduction

The vital problem in group oriented applications is to guarantee the confidentiality. The clandestineness of a broadcast communication session is normally ensured using a cryptographic mechanism. A common key known as group key is shared among the authorized group members in a broadcast group which is used to encrypt and decrypt the messages. In many applications, the group membership is dynamic, i.e., in broadcast session the new members are authorized to join while some existing may be evicted. The evicted members cannot access the information of future group transactions is called forward secrecy and the joining members cannot access the information of past group transactions is called backward secrecy. In order to ensure the forward and backward secrecy, the group key is to be updated and the new group is to be distributed only to the members of the group. Normally in the group communication, unicast or multicast methods are adopted to deliver the new session key which may impose communication overhead. It is a challenging task to distribute the group key in an efficient and scalable way.

A number of protocols have been proposed over a decade for group key management. Based on the architecture of the group communication a set of scalable, efficient and secured centralized group key management protocols (Sherman and McGrew, 2003). The protocols are on hierarchical structure with the cost

Science Publications

of O (log n) keys to be computed, communicated and stored by each members in the group and the overhead for group is large with restricted capacity.

In this study, hybrid architecture has been proposed which yields the advantages of both centralized and contributory group key management protocol using RF and CRT. All the protocols discussed make use of either multicast or unicast for rekeying which may increase the communication complexity. But the proposed protocol uses broadcast for rekeying which makes the communication overhead to a constant.

Related Work

Assorted researchers are paying attention towards the group key management for the past decade.

The group key management (Rafaeli *et al*., 2003) can be broadly classified into:

- Centralized key management
- Decentralized key management and distributed key management

In a centralized key management approach, only one entity called as group controller is responsible for the generation, distribution and the renewal of the group key. Mainly, this approach has single point of failure or “1 affects n” problem.

In decentralized key management, the large dynamic group is split into small subgroups. Different sub group controllers are used to alleviate the problem of single point of failure, but the processing time and communication requirements get increased proportionately in terms of the number of subgroups.

Distributed key management eradicates the existence of central group controller. The session key is generated in a fashion in which all members contribute their own share to compute group key. It is very important to ensure the integrity of the rekey messages.

Begum and Purusothaman (2011) has proposed a protocol which is based on Elliptic curve cryptography algorithm to form secure group key, even with smaller key size, it is capable of providing more security. This protocol can be used both in wired or wireless environments, but the attention should be made with the computation cost which is notable.

Manz *et al*. (2010) proposed a protocol in which costs can be analyzed, procedure and security can be improved and protocols can be implemented for wireless ad-hoc networks. In addition, it led two authors of this study to create a new protocol, DTEGK which increases the communication cost and also cannot be applied for hybrid architecture.

A decentralized protocol: Iolus has been proposed by (Mittra, 1997) in which it disintegrates the group control to each subgroup controller. The proposed protocol lacks and high in confidential overhead in the performance of the relied multicast data by each sub group controller and more over the computational overhead is high.

Steiner *et al*. (2000; Kim *et al*., 2004a; 2004b; 2004c) proposed distributed group key protocols based on group diffie hellman methods for small dynamic peer groups. Rodeh *et al*. (2002) proposed a distributed logical key hierarchy protocol using AVL trees. These protocols can be adopted only for small peer groups and also requires many rounds of communication to update a new group key.

A set of scalable hierarchical structure based group key protocols (Bresson *et al*., 2001; Sherman and McGrew, 2003) have been proposed. In the above protocols storage overhead for the key server is 2n keys and each users stores O (n) keys to perform decryption and for each time, rekeying overhead is O (log n) keys.

Chiou and Chen (1989) proposed a secure broadcasting protocol also based on CRT, used only for a single subgroup and cannot be adopted for hybrid architecture and also it requires O (n) encryptions for each broadcast while proposed protocol only needs 1 encryption.

Zheng *et al*. (2007) proposed a Hierarchical Scalable Group Key Management Based on Chinese Remainder Theorem, a protocol adopted only for single subgroups and not for hybrid architecture.

Vasanthi *et al*. (2014) proposed mathematical model that analyses the various power parameters for group rekeying and locates the finest values for the batch size and interval time using the M/M/1/K model queues.

The weight of server was condensed and also there was no rekeying when a member leaves the group (Saravanan and Purusothaman, 2012). The secret value of parting member was not added in the encryption and so the private value could not be obtained after decryption.

Materials and Methods

In this study, a hybrid broadcast group management protocol has been proposed that removes the above limitations of both centralized and contributory key management protocol. In particular, the protocol enables hybrid architecture with least possible communication and computational overheads and less number of keys. The protocol includes a secure, scalable and efficient hybrid rekeying protocol based on the reverse algorithm and CRT concept to handle (J) Join and (L) Leave scenario for a key tree.

Model Description

Hybrid Architecture

The fundamental reason behind the hybrid architecture (Fig. 1) is to avoid the bottle neck problem in centralized architectures is that when the key server is down (off-line). Here the sub group communications is processed using the reverse function and CRT. On the other hand, if the key server is on-line, then there is a centralized scheme for communication among sub group.

Reverse Function

The Broadcast Key (BK) generation is done using the reverse function to maintain the distinctiveness among the subgroups.

Self Inverse Function

If *x* is an identity then identity function on *x* is its own inverse Equation 1:

$$id_x^1 = id_x^{-1} \tag{1}$$

Chinese Remainder Theorem

The Chinese remainder theorem states that let Rp_1, Rp_2,…..,Rp_n be pair wise relatively prime positive integers and let a_1, a_2,…., a_n be the arbitrary integer. Then the system of linear congruence's in one variable is given by:

$$\begin{gathered} x \equiv a_1 \, mod \, Rp_1 \\ x \equiv a_2 \, mod \, Rp_2 \\ . \\ . \\ . \\ . \\ x \equiv a_n \, mod \, Rp_n \end{gathered}$$

Has a unique solution modulo Rp_1, Rp_2,…..,Rp_n and to compute the unique solution *X*:

$$X = \sum_{i=0}^{m} K_i \, M_i \, M_i{'} \, mod \, m$$

Where:

$$M = Rp_1, Rp_2, \; .....,Rp_n$$

$$M_i = \frac{M}{Rp_i}$$

$M_i{'}$ is the multiplicative inverse of M1 mod:

$$Rp_i \, ie., \, M_1 M_i{'} = 1 \, (mod \, Rp_i)$$

Group Initialization

In the initial stage the new group requires preparation. To begin with, the GSP will craft SGSP and assign a unique Group ID (G_{ID}). The number of Subgroups (SGSP) (2) will be decided by GSP with the number of users in the group which is done by Equation 2:

$$SGSPi = 2 * Log \left(Total \; No. \; of \, users \, in \, the \, whole \, group \right) \tag{2}$$

Now SGSP will be the soul responsible for the members of corresponding subgroups. The Broadcast Key (B_K), Broadcast Session ID (B_{SID}) will be computed and broadcasted to all members. Initially SGSP assigns unique *RPi*, *ai*, G_{ID} to each members with which the S_{ID} and G_K can be computed.

Member Join

After the initial group has been setup and when a user joins the group as above said protocol, RP_i, a_i, G_{ID} will be securely communicated to joining user and the rekeying will be computed in which B_{SID} (4) using Congruence system (CRT) and B_K Equation (3 to 5) using reverse function will be done and broadcasted to corresponding members.

$$\begin{gathered} Ki \; = S_{ID} + \; ai \\ B_{SID} \equiv \; K_1 \, mod \, Rp_1 \\ B_{SID} \equiv \; K_2 \, mod \, Rp_2 \\ BSID \equiv \; K_3 \, mod \, Rp_3 \\ . \\ . \\ . \\ . \end{gathered} \tag{3}$$

$$B_{SID} \equiv \; K_i \, mod \, Rp_i \tag{4}$$

And:

$$B_K = \; \left(G_K + \; G_{ID} \right) S_{ID} \tag{5}$$

Where:

$$G_K = \; Group \; Key$$

G_{ID} = Unique group *ID* generated by Group Service Provider (GSP).

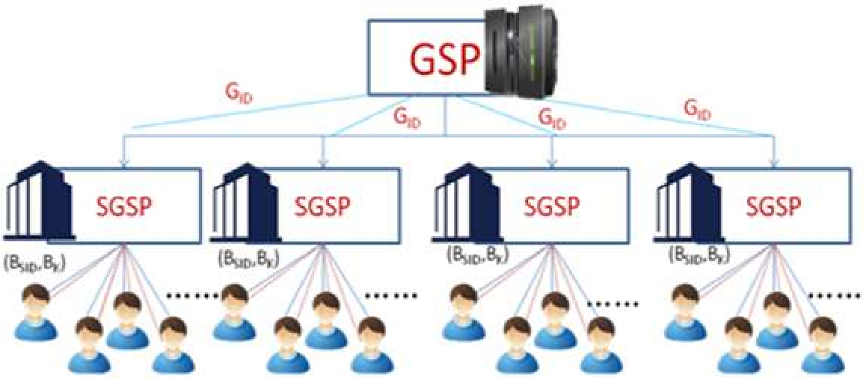

Fig. 1. Hybrid broadcast group management protocol architecture

The re-computation of S_{ID} (6), G_K (7) by the members is done as Equation 6:

$$V_i = B_{SID} \ mod \ RP_i$$

$$S_{ID} = V_i - a_i \tag{6}$$

Group key extraction by members Equation 7:

$$G_{val} = (B_{K1}) / S_{ID}$$

$$G_K = G_{val} \text{-} G_{ID} \tag{7}$$

Member Eviction

In the proposed protocol, the rekeying in member eviction is done similar to member join. After the member eviction, the corresponding member's RP_i, a_i will be black listed and it will be avoided in B_{SID} (8) calculation.

Let us consider member 3 has evicted from the group Equation 8:

$$\begin{gathered} B_{SID} \equiv K_1 \, mod \, Rp_1 \\ B_{SID} \equiv K_2 \, mod \, Rp_2 \\ B_{SID} \equiv K_3 \ mod \ Rp_3 \times evicted \ user \\ . \\ . \\ . \end{gathered}$$

$$B_{SID} \equiv K_i \, mod \, Rp_i \tag{8}$$

The re-computation of B_K and B_{SID} will be done and broadcasted to the corresponding groups.

Security Analyses

In the proposed protocol each current group members will keep their RP_i and a_i values secret, each SGCP will keep the same values of its member's secret.

Furthermore, the set of RP_i and a_i values are arbitrarily picked from an boundless large pool of pair wise relatively prime positive integers, hence knowing one number gains slight knowledge about the others except that the other members do not contain the same factors.

Forward Secrecy

The ultimate aspire of forward secrecy is about to prevent evicted members to enjoy accessing upcoming group communications. In proposed protocol, once each member evicted the RP_i and a_i values of member is discarded, no evicted member can obtain the S_{ID} and G_K values of the group users.

Backward Secrecy

Backward Secrecy for new users is to forbid the new members in accessing earlier group communications. In this protocol, each new group key is a capriciously picked value with no relation to any old group key, without that each new added member's ability of calculating the new group key will not gain knowledge about the previous group key.

Collusion Attack

Collusion attack is the set of evicted members joins together in prediction of the S_{ID} and G_k.

This is not possible with this protocol, since the RP_i and a_i values of member are discarded. For this protocol, even if the set of RP_i and a_i values are arbitrarily picked from an boundless large pool of pair wise relatively prime positive integers, then knowing one number gains slight knowledge about the others except that the other members do not contain the same factors. However, the pool of positive integers is made up of primes then collusion gains no added information.

Results

In the performance analyses of the proposed protocol, a comparative study has been made with some benchmark protocols. Most of the protocols adopt unicast or multicast for distribution of rekey values, but the proposed protocol will broadcast the rekey value which reduces the communication overhead drastically to O(1). Table 1 gives the comparative study of storage complexity where-number of users and m-Number of sub groups regarding storage complexity the proposed HBGMP (Fig. 2) reduces complexity better than the other methods.

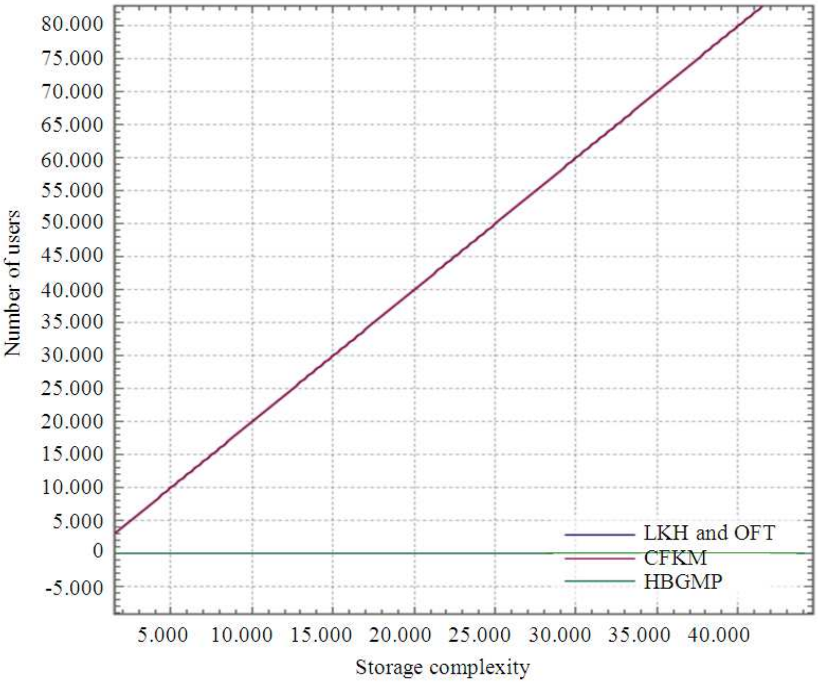

Fig. 2. Storage complexity

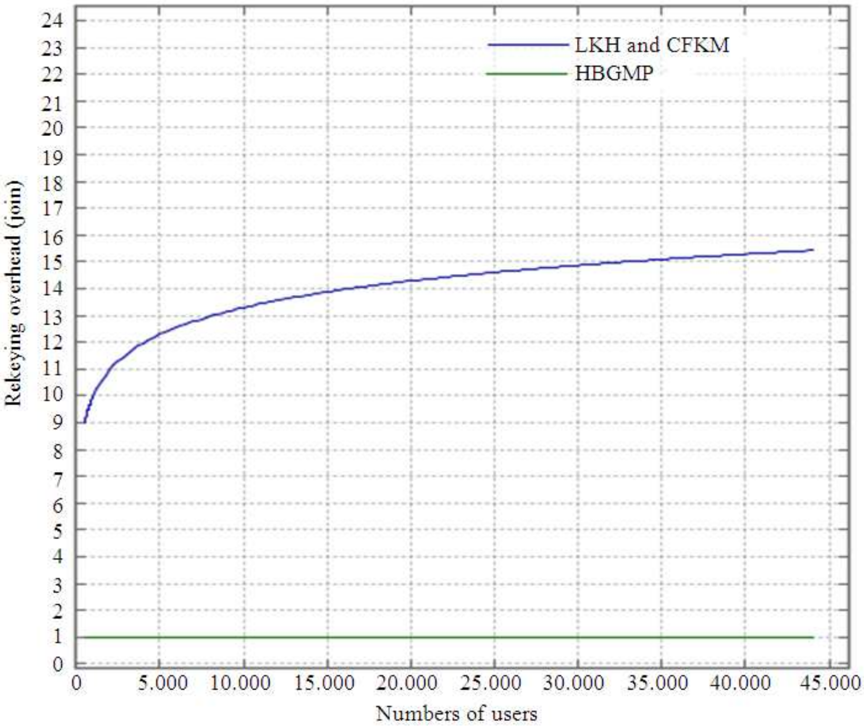

Fig. 3. Communication overhead (join)

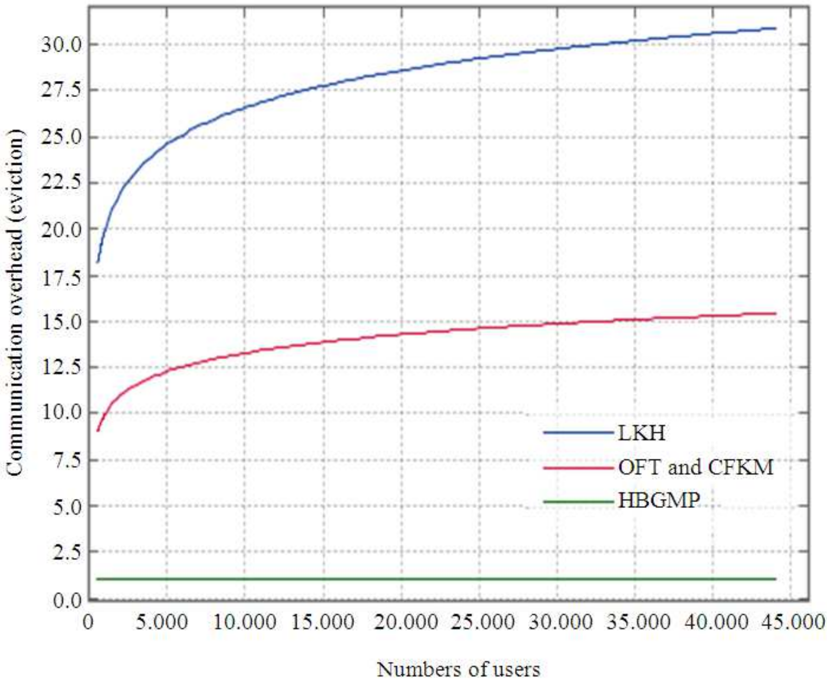

Fig. 4. Communication overhead (eviction)

Table 1. Storage complexity

	Server storage complexity
LKH	O(2n-1)
OFT	O(2n-1)
CFKM	O(2n+1)
HBGMP	log(n/m)

Table 2. Communication complexity in rekeying overhead

	Join	Leave
LKH	O(log(n)+1)	O(2 log(n))
OFT	O(log(n)+1)	
	O(log(n)+1)	
CFKM	O(log(n)+1)	
	O(log(n)+1)	
HBGMP	O(1)	O(1)

While considering the communication complexity, when users join the group into account (Fig. 3), the rekeying overhead is O(1) and also in the case of users evicted (Fig. 4) from the group the rekeying overhead is O(1). The comparative study is made with other methods for the rekeying overhead (Table 2).

Discussion

The proposed hybrid architecture with CRT and Reverse function offers scalability and also security for the group members. The key factors such as storage complexity and communication complexity have been come down to a notable value. This hybrid protocol is suitable for hefty sized and assorted secured group communications.

Conclusion

In this proposed method, a hybrid broadcast group management protocol based on CRT and reverse function is implemented which is very scalable for large size dynamically changing group. The scalability of the proposed protocol is relatively simple hybrid structure, which minimizes re-key message overhead and also requirements on regular group user computation overhead. In case of key storage space, the HBGMP makes suitable for a variety of secure large group communication. Even though the protocol is evaluated using simulation, the unit performance is based on modulo and power functions using broadcast method which is different from other protocol can cause real performance results deviated. In the future work, tuning of the protocol will be done to provide optimized performance in the real time process and comparison with other protocols. Limitations: The computational overhead may increase when the users in the group is maximum and the server has to generate more number of RP_i and a_i values, since the values should be unique.

Funding Information

The authors have no support or funding to report.

Author's Contributions

All authors equally contributed in this work.

Ethics

This article is original and contains unpublished material. The corresponding author confirms that all of the other authors have read and approved the manuscript and no ethical issues involved.

References

Begum, S.J. and D.T. Purusothaman, 2011. A new scalable and reliable cost effective key agreement protocol for secure group communication. J. Comput. Sci., 328-340. DOI: 10.3844/jcssp.2011.328.340

Bresson, E., O. Chevassut and D. Pointcheva, 2001. Provably authenticated group diffie hellman key exchange the dynamic case (extended). Proceedings of the 8th ACM Conference on Computer and Communication Gold Coast, Dec. 9-13, Berlin Heidelberg, Australia, pp: 290-309. DOI: 10.1007/3-540-45682-1_18

Chiou, G.H. and W.T. Chen, 1989. Secure broadcasting using the secure lock. IEEE Trans. Software Eng., 15: 929-934. DOI: 10.1109/32.31350

Kim, Y., A. Perrig and G. Tsudik, 2004a. Group key agreement efficient in communication. IEEE Trans. Comput., 53: 905-921, DOI: 10.1109/TC.2004.31

Kim, Y., A. Perrig and G. Tsudik, 2004b. Simple and fault-tolerant key agreement for dynamic collaborative groups. Proceedings of the 7th ACM Conference on Computer and Communications Security, Nov. 01-04, ACM, New York, pp: 235-24. DOI: 10.1145/352600.352638

Kim, Y., A. Perrig and G. Tsudik, 2004c. Tree-based group key agreement. ACM Trans. Inform. Syst. Secu., 7: 60-96. DOI: 10.1145/984334.984337

Manz, D., P. Oman and J.A. Foss, 2010. A framework for group key management protocol assessment independent of view synchrony. J. Comput. Sci., 6: 229-234. DOI: 10.3844/jcssp.2010.229.234

Mittra, S., 1997. Iolus: A framework for scalable secure multicasting. Proceedings of the ACM SIGCOMM 97th Conference on Applications, Technologies, Architectures and Protocols for Computer Communication, (PCC' 97), ACM New York, pp: 277-288. DOI: 10.1145/263109.263179

Rafaeli, S. and D. Hutchison, 2003. A survey of key management for secure group communication. Association Comput. Machin. Comput. Surveys, 35: 309-329. DOI: 10.1145/937503.937506

Rodeh, O., K.P. Birman and D. Dolev, 2002. Using AVL trees for fault Tolerant group key management. Int. J. Inform. Secu., 1: 84-84.

Saravanan, K. and T. Purusothaman, 2012. Efficient star topology based multicast key management algorithm. J. Comput. Sci., 8: 951-956. DOI: 10.3844/jcssp.2012.951.956

Sherman, A.T. and D.A. McGrew, 2003. Key establishment in large dynamic groups using one-way function trees. IEEE Trans. Software Eng., 29: 444-458. DOI: 10.1109/TSE.2003.1199073

Steiner, M., G. Tsudik and M. Waidner, 2000. Key agreement in dynamic peer groups. IEEE Trans. Parallel Distributed Syst., 11: 769-780. DOI: 10.1109/71.877936

Vasanthi, A and T. Purusothaman, 2014. Optimizing batch rekeying interval for secure group communication based on queuing model. J. Comput. Sci., 10: 325-329. DOI: 10.3844/jcssp.2014.325.329

Zheng, X., M.M. Manton, C.T. Huang, 2007. Hierarchical scalable group key management based on Chinese remainder theorem. Proceedings of the 6th Annual Security Conference, Apr. 11-12, Las Vegas, NV, pp: 22-3.