Enhanced Group Key Distribution Protocol for Intra Group and Inter Group Communication Using Access Control Polynomial

M. Ragunathan[1], T. Kathirvalavakumar[2(✉)], and Rajendra Prasath[3]

[1] Department of Information Technology, V.H.N. Senthikumara Nadar College, Virudhunagar, Tamil Nadu 626001, India

[2] Research Centre in Computer Science, V.H.N. Senthikumara Nadar College, Virudhunagar, Tamil Nadu 626001, India
kathirvalavakumar@yahoo.com

[3] Department of Computer Science and Engineering, Indian Institute of Information Technology, Sri City, Chittoor, Andhra Pradesh, India
rajendra.prasath@iiits.in

Abstract. In today's Internet world, group communications have become very crucial for several applications. It is essential to maintain confidentiality during communication hence it is very important to efficiently and securely distribute the common keys to the group members and target group members for encrypting and decrypting the message. This paper proposes an access control polynomial based on Chinese remainder theorem (CRT) for group key distribution (ACPGKD). Also proposes an authentication protocol for dynamic members to join or leave the group using the polynomial to keep backward and forward secrecy in inter-group and intra-group communications. It has been shown that the proposed work is secure and computationally efficient.

Keywords: Group key distribution · Rekeying · Secure group communication · Polynomial based key communication · Chinese remainder theorem

1 Introduction

Group communication plays a vital role in pay-per-use like Pay-television, online classes, OTT, online game, video conferencing and video broadcasting. A group can be a fixed one or as a varying one. In a dynamic multicast communication, members can join or leave the group at any time. In a group, all group members require a common group key namely intra-group key to communicate with each other. Outside group members called as target group members need an inter-group key when they want to communicate with one of the group members. To achieve secure group communication, the group key (GK) must be shared only to the group members. The group key management system is divided into three categories namely centralized, decentralized and distributed. In centralized group

R. Chbeir et al. (Eds.): MIKE 2021, LNAI 13119, pp. 225–232, 2022.
https://doi.org/10.1007/978-3-031-21517-9_23

key management group controllers (GC) are responsible for generating and distributing the common group key among the authorized group members. In a decentralized group key management scheme, larger groups are divided into sub groups, and common group key is generated and distributed by corresponding sub group controllers. In a distributed group key management scheme group members have to cooperate with each other and contribute equally to generate the group key.

The group key management schemes must fulfill the basic security requirements namely forward and backward secrecy in the group. Group controller is responsible to prevent newly joining members from having access the previously communicated data to provide backward secrecy. To provide forward secrecy, the group controller is responsible to prevent the leaving members from further accessing future communications. Whenever group members leave the group or new members enter into the group, group controller has to update and distribute the group key, which is called as rekeying.

Vinod kumar et al. [9] have proposed centralized group key distribution protocol (CGKD) based on RSA public key cryptosystem and implemented on key star and cluster tree structure. Key star structure mainly focuses on the key update phase to minimize the computation and storage load of the key server. Clustered tree based architecture has achieved scalability and minimizes the computation complexity of the key update phase whenever the members want to join or leave the group in batches. Huo Guo et al. [3] have proposed self-healing group key distribution protocol in wireless sensor networks for secure IoT communications. They have used access control self-healing group key distribution (AP-SGKD). It satisfies basic security properties with optimal storage requirement and optimal session key recovery time. The proposed protocol holds mt-wise forward secrecy, wise backward secrecy and mt revocation capability. Additionally the storage requirement of the protocol is constant and this protocol is suitable for the Zigbee network.

Alphonese et al. [1] have proposed scalable and secure group key agreement for wireless ad-hoc networks by extending the RSA scheme. It uses linear time to generate group key agreement computations, and create private keys. The partial and group signatures are dynamic. The security of framework completely depends on the RSA scheme to achieve shared key authentication and user authentication. On receipt of a group message, every member of the group can calculate their group key by using their partial signature and private key. Each member requires internal memory to store their private key, even though they belong to many number of groups. Partial signatures of the groups and public keys of the members are stored in the memory which is accessed by members and non-members of the group. Sirui et al. [7] have proposed a secure communication system in self-organizing networks by lightweight group key generation. It is based on the difference of quantization results at one device from different channels (DORCE). It uses adaptive quantizer to generate pairwise keys. The users share the group key via the difference between pairwise keys. The users of the self organizing network can flexibly join and exit, compared to chain topology and star topology.

Yanji Piao et al. [10] have proposed polynomial based key management for secure intra group and inter group communication. The proposed group key management scheme uses two kinds of polynomials. One to derive the intra group key and the other

to create the inter group key. In this proposed scheme, group members and group controllers can share the intra-group key without any encryption/decryption. It reduces the number of rekeying messages during group changes. The proposed scheme drastically reduces the amount of broadcast traffic in the inter group communication. Shaukat Ali et al. [6] have proposed a scalable group key management protocol. The significance of the protocol is reducing the number of rounds in the key generation process irrespective of the group size. Also each user needs only two transmissions for the entire process of the group key generation, and the communication complexity is distributed among the group users. This protocol does not require any secure channel or trusted server for the key management. Velumadhava et al. [8] have proposed hierarchical group key management for secure data sharing in a cloud based environment. This protocol reduces the complexity whenever a member leaves the group. This system uses inverse values to update the key during leave operation. The group members can calculate the key by themselves using the inverse value.

In the literatures most of the authors generate inter and intra group keys separately for group communication. Group controller is responsible for securely sending the group key. When members of the group are dynamic there exist computational overhead to the group controller. This paper proposes a protocol for group key distribution for communicating within a group and between groups with a new access control polynomial proposed with the base of Chinese remainder theorem. In the proposed method group controller is not sending group key but user finds the key from the polynomial given by the group controller. The proposed method needs minimum processing time. Section 2 has proposed group key distribution method. Section 3 evaluates the proposed approach in term of factorization attack, time attack, forward secrecy and backward secrecy. Section 4 discusses the experimental results. Section 5 concludes the work.

2 Access Control Polynomial Based Group Key Distribution Method

Proposed an Access control polynomial based group key distribution (ACPGKD) protocol and a rekey generation procedure for the dynamic access control in a secure inter and intra group communications. Private and public keys are generated using the Chinese remainder theorem [4].

2.1 Chinese Remainder Theorem

Let m1, m2,….mn are positive integers and are relatively prime in pairs. For any given integers b1, b2,…bn, following system of congruence equations have a unique solution.

$$\begin{aligned} GK &\equiv b1 \bmod m1, \\ GK &\equiv b2 \bmod m2, \\ &.. \\ GK &\equiv bn \bmod mn, \end{aligned}$$

It can be represented as

$$GK \equiv bi \bmod mi \quad \text{for } i = 1, 2, \ldots..n, \tag{1}$$

where GK is a group key, and can be computed from

$$GK = \sum_{i=0}^{n} bi\ Mi\ Yi (mod\ M) \tag{2}$$

where $M = \prod_{i=1}^{n} mi$, Mi = M/mi and MiYi ≡ 1 (mod mi) (since MiYi are multiplicative inverse).

2.2 Key Initialization Phase

Let P = {u1, u2,…..un} denote the set of group members. The Group Controller (GC) generates n positive integers m1, m2,….mn which are relatively prime in pairs (i. e, gcd(mi, mj) = 1 for i ≠ j and 1 ≤ i, j ≤ n) and are considered as the public keys for the group members. Any given integers b1, b2,….bn, which satisfies the congruence Eq. (1) are private keys for the group P. As per Chinese remainder theorem, the group key GK is calculated from Eq. (2).

GC assigns the generated keys mi to all the group members and broadcast all the mis to outside group members called as target group members. The private keys bi is sent to the corresponding member ui,, i = 1,2,..,n in a secured channel and members have to keep it as secret.

GC generates the public group key (PGK) by

$$PGK = \prod_{i=1}^{n} mi \tag{3}$$

The target group members have to use the PGK for communicating with the group members.

The GC generates an Access Control Polynomial (ACP) using (4) which uses public keys of all members.

$$ACP = (x - m1)\,(x - m2)\,\ldots..(x - mn) + PGK \tag{4}$$

The group controller broadcasts the ACP to the group members and the target group members. This ACP can be used by a member or target group member for inter and intra group communications**.**

2.3 Key Recovery Phase

The PGK is used for broadcasting any message to the group members by the target group members. The public group key (PGK) is computed after all the group members and target group members received the ACP. The public key of any one of the member is to be substituted in x of ACP to find the PGK.

When the group members want to communicate with themselves they need exchange key (EK) of everyone in the group. Everyone in the group can find PGK from ACP by substituting their public key instead of x. Everyone can find their exchange key by (5).

$$EKi = bi * Mi * Yi \tag{5}$$

Mi = PGK/mi.

Let Yi be the Multiplicative inverse of Mi where Mi*Yi ≡ 1 (mod mi).

bi is the private key for the group member.

All group members have to share their exchange key with all other members in their group. Now every group member has an exchange key of everyone in the group. Each group member can find the group key (GK) by Eq. (6)

$$GK = \sum_{i=0}^{n} EKi (\text{mod } PGK), \text{ where } i = 1, 2, ..., n \tag{6}$$

GK is a group key generated by every group member. Using the GK, all the group members can communicate with others in the group.

2.4 Key Update Phase

Whenever new members want to join or existing members want to leave the group G, the GC needs to regenerate and distribute new ACP to all the present group members and target group members.

Member Leave Phase

Whenever a member ui wants to leave the group, the GC has to delete their corresponding mi and bi from the active list and has to update the value of public group key(PGK) using Eq. (7).

$$\text{New PGK} = \text{current PGK}/mi \tag{7}$$

The GC has to update the group key by

$$\text{New GK} \equiv bi \text{ mod } mi, i = 1, 2, ...n - 1 \tag{8}$$

Now GC regenerates a new polynomial

$$\text{New ACP} = (x - m1)(x - m2) .....(x - mn - 1) + \text{new PGK} \tag{9}$$

All current group members can derive new PGK but relieved members cannot derive new PGK, so leaving members cannot access future communications (Forward Secrecy).

Member Join Phase

Whenever a member ui wants to join the group, the GC updates mi, bi values in the active list and update the value of public group key by

$$\text{New PGK} = \text{current PGK} * mi \tag{10}$$

The GC updates the group key by

$$\text{New GK} \equiv bi \text{ mod } mi, \ i = 1, 2, ..., n + 1 \tag{11}$$

GC regenerates a new polynomial

$$\text{New ACP} = (x - m1)(x - m2) \ldots (x - mn)(x - mn + 1) + \text{new PGK} \tag{12}$$

The new group member ui can derive new PGK but cannot derive old PGK. So, new members cannot derive the previous Group keys (Backward secrecy).

3 Evaluating the Proposed Approach

In this section, security strength, safety, the number of rekeying messages, storage and communication overhead of the proposed ACPGKD protocol are computed and prove that the protocol is secure against the factorization attack, Timing attack and fulfill the backward and forward secrecy requirements.

3.1 Efficiency

In the existing works of the literatures, inter and intra group keys are distinct and are send separately to the members of the group. The GC encrypts an intra-group key and sends it to the group members. GC encrypts the inter group key and sends it to target group members. But in this proposed work the group controller does not need to encrypt the key and also uses single polynomial ACP to the group members and target group members for inter and intra group communication. Normally computing the polynomial is easier than performing encryption and decryption.

3.2 Factorization Attack

In order to derive GK, the group controller sends a polynomial without any encryption. However it is not easy to guess the inter group key GK from the polynomial and it is very hard to do polynomial factorization because there is actually an O(nlog n) solution to the polynomial expansion [4, 5] (n represents degree of the polynomial) and the problem for polynomial factorization is NP-hard [2].

3.3 Group Key Attack

Suppose any intruder enters into a group, the intruder has to compute the multiplicative inverse pair Mi and Yi in such a way that Mi*Yi $\equiv$ 1 (mod mi). The intruder has to know their bi but it can be shared only by GC to their authorized members, so their EK cannot be computed by them. To view the communication messages between intra and inter group members, the intruder should know the EK of others but it is not possible as the intruder is not in the authorized active list to get them from others. Without knowing EKs the GK cannot be computed, so the intruder cannot communicate with others in the group.

3.4 Forward and Backward Secrecy

Let u1, u2,…..,un are group members and m1, m2,….,mn are the private keys generated by GC for the group members. Every group member ui received their key mi from the GC. PGK $= \prod_{i=1}^{n} m_i$ is a public group key known to the GC.

ACP $= (x - m1)\ (x - m2)\ \ldots..(x - mn) + PGK$, is given to each member by the GC for sending PGK securely. PGK is used to encrypt or decrypt the message within a group. If a member ui wants to know the PGK, their private key mi is to be subsitituted in the ACP. Now in the ACP, only PGK is there as the first term of ACP becomes vanish.

When a member uj leaves the group, GC creates new ACP without the term (x-mj) but with new PGK as $\prod_{i=1}^{n-1} m_i$ and is sent to all the group members. If the left member uj apply their private key mj in the ACP they hold, first term of ACP becomes vanish and get the value for ACP, which is nothing but old PGK that is $\prod_{i=1}^{n} m_i$ and is not eqivalent to new PGK. So this PGK can not be used to decrypt any message of the current group members. Hence forward secrecy is maintained.

If a new member uk join the group, GC creates private key mk corresponding to the new member and is sent to the member and creates new ACP with new PGK as $\prod_{i=1}^{n+1} m_i$ and is sent to all the members currently in the group. New member uk gets PGK from the new ACP by substituting their private key mk in the ACP received from the GC. As this PGK is differed from previous PGK, new member uk could not decrypt the previous messages passed inside the group as the previous messages were created with previous PGK. Hence backward secrecy is maintained.

3.5 Re-keying Overhead

In the proposed ACPGKD scheme, when a member joins the group, the group controller needs to unicast a private key to the new member in a secure channel and multicast a new polynomial ACP to the current members in the group and target group members. When a member leaves from the group, the group controller needs to change the polynomial ACP and multicast it to the group members and target group members.

4 Experimental Results

This section shows the experimental results of the proposed ACPGKD protocol. Experiments are performed on a system with 3.3-GHz Intel Core i3–3220 processor, 4-GB RAM and the OS Windows 10. JAVA programming language with Java Runtime Engine (JRE) 1.6 is used to evaluate the performance of the Group Controller and Group Members. Here the assumption is, the group is with 3 members. The Processing time is in nano seconds (ns) for Group key generation, key recovery, Rekeying, and average communication cost. The processing times are compared with the CGKD [9] and are shown in Table 1. In the figures x-axis represents number of systems and y-axis represents consumed processing time in ns. It is observed from the table that the proposed work is better than CGKD.

Table 1. Processing time

#of systems	Generation time(ns)		Recovery time(ns)		Rekeying time(ns)		Avg. computing time(ns)	
	ACPGKD	CGKD	ACPGKD	CGKD	ACPGKD	CGKD	ACPGKD	CGKD
1	133216	1034562	543456	103465	104320	1534254	573322	1201157
2	162145	1503142	702987	1503124	150298	2203780	789373	1736682
3	195136	1905482	925364	2105741	192547	3005698	1015326	2338974

5 Conclusion

The protocol for distributed group key management scheme is proposed. ACPGKD for membership authentication and rekeying for the dynamic group are proposed. The protocol provides both inter group and intra group key distribution with single polynomial. The proposed protocol has reduced the computational complexity of group controller and group members. The protocol secures against Factorization attack and guarantees the backward secrecy and forward secrecy in the group. Experimental results show that the proposed ACPGKD needs less computation time, less recovery time and lesser rekey complexity than the CGKD protocol. The same protocol may be enhanced in future to implement secure group communication in the IoT environment.

References

1. Alphonse, P.J.A., Venkatramana Reddy, Y.: Scalable and secure group key agreement for wireless ad-hoc networks by extending RSA scheme. Concurrency Comput. Pract. Experience **31**(14), e4969 (2018). https://doi.org/10.1002/cpe.4969
2. Gao, S., Hoeiji, M.V., Kaltofen, E., Shoup, V.: The Computational complexity of polynomial factorization. American institute of Mathematic, **364,** 1–5, Palo Alto, California (2006)
3. Guo, H., Zheng, Y., Li, X., Li, Z., Xia, C.: Self healing group key distribution protocol in wireless sensor networks for secure IoT communications. Future Gener. Comput. Syst. **89**, 713–721 (2018)
4. Sipser, M.: Introduction to the Theory of Computation, 2nd edn. Thomson course Technology, Boston (2006)
5. Roche, D.S., Space- and time- efficient polynomial multiplication. In: ACM International Symposium and Algebraic Computation, pp. 28–31. ACM, Seoul Republic of Korea (2009)
6. Ali, S., Islam, A.R.N., Farman, H., Jan, B., Khan, M.: A Scalable group key management protocol. Sustain. Cities Soc. **39**, 37–42 (2018)
7. Peng, S., Han, B., Wu, C., Wang, B.: A secure communication system in self-organizing networks via lightweight group key generation. IEEE Open J. Comput. Soc. **1**(1), 182–192 (2020)
8. Velumadhava Rao, R., Selvamani, K., Kanimozhi, S., Kannan, A.: Hierarchical group key management for secure data sharing in a cloud based environment. Concurrency Computat Pract Exper. **31**, e4866 (2019). https://doi.org/10.1002/cpe.4866
9. Kumar, V., Kumar, R., Pandey, S.K.: A computationally efficient centralized group key distribution protocol for secure multicast communications based upon RSA public key. J. King Saud Univ. – Comput. Inf. Sci. **32**(9), 1081–1094 (2018)
10. Piao, Y., Kim, J., Tariq, U., Hong, M.: Polynomial based key management for secure intra group and inter group communication. Comput. Math. Appl. **65**(9), 1300–1309 (2013)