ISSN: 2250–3676

[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-6, 1608 – 1613

AN IMPROVED SHAMIR SECRET SHARING FOR SECURE COMMUNICATIONS

Aruna Maturu [1], K.Trinadh Ravi Kumar [2]

[1]*Associate Professor, Dept.of Computer Science & Engineering, Swarnandhra Engineering College, Narsapuram, W.G Dist., AP.*

[2]*Associate Professor, Dept.of Computer Science in S.V.K.P & Dr.K.S.Raju Arts & Science College, Penugonda, W.G Dist, A.P.*

Abstract

A key exchange protocol enables parties to share a common key for encrypting a large amount of data. Authentication is an essential requirement prior to the key exchange process in order to prevent man-in-the-middle attack. It is important to understand the capabilities and performance of the existing key exchange protocols before employing the protocols in our applications. In this paper, we propose an authenticated key transfer protocol based on secret sharing scheme that KGC can broadcast group key information to all group members at once and only authorized group members can recover the group key but unauthorized users can not recover the group key. In this proposed mechanism shows More security levels In case of confidentiality and authentication than the traditional group key protocols can be analysed in detail.

Index Terms- *Group key transfer protocol, session key, secret sharing, confidentiality, authentication*

----------*****----------

1. INTRODUCTION

Multicast key management, which is much different from unicast key management , is one of the most attractive area of cryptography. For an unicast application, the Diffie-Hellman key exchange protocol can be employed to establish a KEK (Key Encryption Key) between two entities. Then use this KEK to dispatch or update a session key. In contrast, the situation is much more complicated for a multicast application. A multicast application must dynamically handle multi-entities. For example, in a dynamic multicast group, the membership is changeable all the time due to frequently users' addition and eviction. Therefore, the key materials will probably be revealed if no security policies are adopted. For instance, if the key is not updated after the membership change, a new comer is able to read the contents before his coming, or a evictor is capable of reading the content after his leaving. In this case, multicast key management scheme should provide forward secrecy and backward secrecy for security reasons in some special applications, e.g. Pay-Per-View. In the past two decades, researchers have proposed many multicast key management schemes [5], [6], [7]. These schemes can be categorized into three different types: centralized, decentralized and distributed. A centralized group key management scheme involves a Key Server (KS) to generate and distribute shared key to all group members via a secure channel. A decentralized key management divides the whole group into smaller subgroups. Each subgroup is controlled by a single or several KS. A Distributed scheme allows each member to take part in a group key generation collaboratively. Each of the three schemes has its own advantages and disadvantages. Centralized scheme is the simplest one but has the risk of single-point-failure. Decentralized scheme adds some communication complexity between two members within different subgroups. Distributed scheme is somehow more complex than the other two, but it doesn't involve KS. This feature is very useful in the case of no one can play the role of KS, e.g. a sensor Ad-hoc network application.

Secret sharing has been used to design group key distribution protocols. There are two different approaches using secret sharing: one assumes a trusted offline server active only at initialization [4], [14], [25], [3] and the other assumes an online trusted server, called the key generation center, always active. The first type of approach is also called the key pre-

distribution scheme. In a key pre-distribution scheme, a trusted authority generates and distributes secret pieces of information to all users offline. At the beginning of a conference, users belonging to a privileged subset can compute individually a secret key common to this subset. A family of forbidden subsets of users must have no information about the value of the secret. The main disadvantage of this approach is to require every user to store a large size of secrets. The second type of approach requires an online server to be active [20] and this approach is similar to the model used in the IEEE 802.11i standard [17] that employs an online server to select a group key and transport it to each group member. However, the difference between this approach and the IEEE 802.11i is that, instead of encrypting the group temporal key (GTK) using the key encryption key (KEK) from the authentication server to each mobile client separately as specified in the IEEE 8-2.11i, the trusted KGC broadcasts group key information to all group members at once. In 1989, Laih et al. [20] proposed the first algorithm based on this approach using any(t,n) secret sharing scheme to distribute a group key to a group consisting of (t-1) members. Later, there are some papers [2], [21], [25] following the same concept to propose ways to distribute group messages to multiple users. In this paper, we propose a solution based on this approach and provide confidentiality and authentication for distributing group keys. Furthermore, we classify attacks into insider and outsider attacks separately, and analyse our protocol under these attacks in detail.

We list following unique features of our proposed group key transfer protocol using secret sharing scheme.

- Each user needs to register at KGC to subscribe the group key transfer service and to establish a secret with KGC. Thus, a secure channel is needed initially to share this secret with each user. Later, KGC can transport the group key and interact with all group members in a broadcast channel.
- The confidentiality of group key distribution is information theoretically secure; that is, the security of this transfer of group key to each group member does not depend on any computational assumption.
- The authentication of the group key is achieved by broadcasting a single authentication message to all group members.

2. LITERATURE SURVEY

Definition 1 (Factoring Problem). Let us choose two large safe primes p and q (i.e., primes such that p'=(p-1)/2 and q'=(q-1)/2 are also primes) and compute n = pq. n is made publicly known. Factoring problem is defined to compute factors p and q such that n =pq.

Definition 2 (Factoring Assumption). It is computationally intractable to solve the Factoring Problem.
Secret sharing schemes were introduced by both Blakley [1] and Shamir [26] independently in 1979 as a solution for safeguarding cryptographic keys and have been studied extensively in the literatures. In a secret sharing scheme, a secret s is divided into n shares and shared among n shareholders in such a way that, with any t or more than t shares, it is able to reconstruct this secret; but, with fewer than t shares, it cannot reconstruct the secret. Such a scheme is called a(t,p) secret sharing, denoted as ðt; nÞ-SS. Shamir's ðt; nÞ-SS. In Shamir's (t,p)-SS [26] based on Lagrange interpolating polynomial, there are n shareholders U(U1; . . . ; Un) and a mutually trusted dealer D. The scheme consists of two algorithms:

1. Share generation algorithm: dealer D does the following:.

- dealer D first picks a polynomial f(x) of degree (t-1) randomly $f(x)= a_0+a_1 +\ldots.+a_{t-1}x^{t-1}$ in which the secret $s=a_0=f(0)$ and all coefficients $a_0; a_1; \ldots ; a_{t-1}$ are in a finite field $F_p = GF(p)$ with p elements.
- D computes all shares: $s_i = f(i)(\text{mod } p)$ for i=1 . . . ; n.
- Then, D outputs a list of n shares $s_1. s_2, \ldots ;,s_n$ and distributes each share s_i to corresponding shareholder Pi privately.

2. Secret reconstruction algorithm: this algorithm takes any t shares (si...sit)as input, it can reconstruct the secret s as

$$s = f(0) = \sum_{i \in A} s_i \beta_i$$
$$= \sum_{i \in A} s_i \left(\prod_{j \in A - \{i\}} \frac{x_j}{x_j - x_i} \right) \quad (\text{mod } p),$$

where $A = \{i_1, \ldots, i_t\} \subseteq \{1, 2, \ldots, n\}$, β_i for $i \in A$ are Lagrange coefficients.

Lagrange coefficients. We note that the above scheme satisfies the basic security requirements of secret sharing scheme as follows: 1) with knowledge of any t or more than t shares, it can reconstruct the secret s easily; and 2) with knowledge of fewer than (t-1) shares, it cannot reconstruct the secret s. Shamir's scheme is information theoretically secure since the

scheme satisfies these two requirements without making any computational assumption. For more information on this scheme, readers can refer to the original paper [26]. In Shamir's (t,n)-SS, the secret of each shareholder is just the y-coordinate of f(x) and the x-coordinate is made publicly known. However, in our proposed group key transfer protocol, for security reason, we need to keep both x-coordinate and y-coordinate as each user's secret. Furthermore, in Shamir's (t,n)-SS, the modulus p used for all computations is a prime number. In our proposed protocol, to prevent insider attack as we will explain this later, the modulus n used for computations is a composite integer. We should point out that finding a modular inverse is needed in secret reconstruction process. We can use Euclid's extended algorithm [30] to compute modular inverse without factoring the composite modulus n.

3. PROPOSED PROTOCOL

3.1. Prototype

Group key transfer protocol relies on one trusted entity, KGC, to choose the key, which is then transported to each member involved. Each user is required to register at KGC for subscribing the key distribution service. The KGC keeps tracking all registered users and removing any unsubscribed users. During registration, KGC shares a secret with each user. In most key transfer protocol, KGC encrypts the randomly selected group key under the secret shared with each user during registration and sends the ciphertext to each group member separately. An authenticated message checksum is attached with the ciphertext to provide group key authenticity. In this approach, the confidentiality of group key is ensured using any encryption algorithm which is computationally secure. Our protocol uses secret sharing scheme to replace the encryption algorithm. A broadcast message is sent to all group members at once. The confidentiality of group key is information theoretically secure. In addition, the authentication of broadcasting message can be provided as a group authentication. This feature provides efficiency of our proposed protocol.

Our authenticated group key transfer protocol consists of three processes: initialization of KGC, user registration, and group key generation and distribution. The detailed description is as follows:

Initialization of KGC. The KGC randomly chooses two safe primes p and q (i.e., primes such that p1=(p-1)/2 and q1= (q-1)/2 are also primes) and compute n = pq. n is made publicly known.

User Registration. Each user is required to register at KGC for subscribing the key distribution service. The KGC keeps tracking all registered users and removing any unsubscribed users. During registration, KGC shares a secret, (xi,yi), with each user Ui, where xi, yi E Zn

Group key generation and distribution. Upon receiving a group key generation request from any user, KGC needs to randomly selects a group key and access all shared secrets withgroup members. KGC needs to distribute this group key to all group members in a secure and authenticated manner. All communication between KGC and group members are in a broadcast channel. For example, we assume that a group consists of t members, {U1; U2; . . . ; Ut}, and shared secrets are (xi,yi), for i={1; . . . t}. The key generation and distribution process contains five steps.

Step 1. The initiator sends a key generation request to KGC with a list of group members as {U1; U2; . . . ; Ut}.

Step 2. KGC broadcasts the list of all participating members, {U1; U2; . . . ; Ut}, as a response.

Step 3. Each participating group member needs to send a random challenge, Ri E Z+n, to KGC.

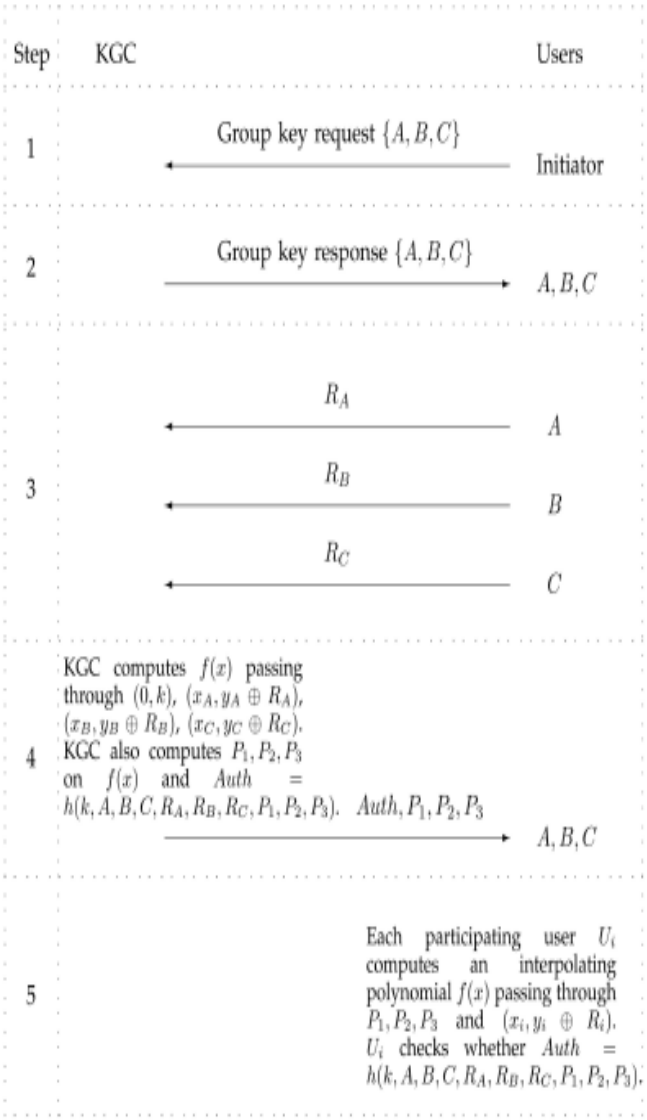
Step 4. KGC randomly selects a group key, k, and generates an interpolated polynomial fðxÞ with degree to pass through (t+1) points, (0,k) and (xi,yi XOR Ri), for i = 1; . . . ; t. KGC also computes t additional points, Pi, for i =1; . . . ; t, on f(x) and Auth =h(k;U1; . . . ; Ut; R2; . . .;Rt; P1; . . . ; Pt), where h is a one-way hash function. All computations on f(x) are over Z+n. KGC broadcasts {Auth; Pi, for i = 1; . . . ; t, to all group members. All computations are performed in Z * n.

Step 5. For each group member, Ui, knowing the shared secret, (xi,yi XOR Ri),, and t additional public points, Pi, for i = 1; . . . ; t, on f(x), is able to compute the polynomial f(x) and recover the group key k =f(0). Then, Ui computes h(k, U1; . . . ; Ut;R1; . . .; Rt; P1; . . . ; Pt) and checks whether this hash value is identical to Auth. If these two values are identical, Ui authenticates the group key is sent from KGC.

In Fig. 1, we illustrate this group key transfer protocol for a group containing three members, A, B, and C..

In our protocol, during registration, KGC shares a secret, (xi,yi), with each user Ui. Adding/removing any user does not need to update any existing shared secret. However, for distributing a secret group key involving t group members, KGC needs to broadcast a message containing (t+1) elements

to all group members. At the same time, each group member needs to compute a t-degree interpolating polynomial f(x) to decrypt the secret group key. Thus, our proposed protocol is only suitable for distributing secret group key to a group with a small group size. If a group containing a large group size, such as applications in pay-per-view system, centralized group key distribution protocols, such as EBS protocol [13], can be used to reduce the length of broadcast message and computational load of each group member



Group key transfer protocol

4. SECURITY ANALYSIS

ATTACKS

4.1.Attacks

Adversaries can be categorized into two types. The first type of adversaries are outsiders of a particular group. The outside attacker can try to recover the secret group key belonging to a group that the outsider is unauthorized to know. This attack is related to the confidentiality of group key. In our proposed protocol, anyone can send a request to KGC for requesting a group key service. The outside attacker may also impersonate a group user to request a group key service. In security analysis, we will show that the outside attacker gains nothing from this attack since the attacker cannot recover the group key. The second type of adversaries are insiders of a group who are authorized to know the secret group key; but inside attacker attempts to recover other member's secret shared with KGC. Since any insider of a group is able to recover the same group key, we need to prevent inside attacker knowing other member's secret shared with KGC.

4.2 Outsider attack

Assume that an attacker who impersonates a group member for requesting a group key service, then the attacker can neither obtain the group key nor share a group key with any group member Although any attacker can impersonate a group member to issue a service request to KGC without being detected and KGC will respond by sending group key information accordingly; however, the group key can only be recovered by any group member who shares a secret with KGC. This security feature is information theoretically secure. If the attacker tries to reuse a compromised group key by replaying previously recorded key information from KGC, this attack cannot succeed in sharing this compromised group key with any group member since the group key is a function of each member's random challenge and the secret shared between group member and KGC. A compromised group key cannot be reused if each member selects a random challenge for every conference.

4.3. Insider attack

Assume that the protocol runs successfully v times and the applied factoring instances are intractable, then the secret (xi,yi) of each group member shared with KGC remains unknown to all other group members (and outsiders). For a group key service request, KGC generates a the degree polynomial f(x) passing through (t+1) points,(0,k) and (xi, yi XOR Ri) for i=(1; . . . ; t). For each authorized group member, with knowledge of the secret shared with KGC and t public

information, he/she knows (t+1) points on f(x). Thus, any authorized group member is able to reconstruct the polynomial f(x). However, the secret (xi,yi) of each group member shared with KGC remains unknown to outsiders.

CONCLUSION:

In this paper we proposed a novel mechanism for group key transfer protocol based on secret sharing with the help of trusted KGC and preshare a secret with KGC and it broadcasts all the information slimantaniously.It provides the security measures like confidentiality and authentication. We provide group key authentication. Security analysis for possible attacks is included.

REFERENCES

[1] G.R. Blakley, "Safeguarding Cryptographic Keys," Proc. Am. Federation of Information Processing Soc. (AFIPS '79) Nat'l Computer Conf., vol. 48, pp. 313-317, 1979.
[2] S. Berkovits, "How to Broadcast a Secret," Proc. Eurocrypt '91 Workshop Advances in Cryptology, pp. 536-541, 1991.
[3] R. Blom, "An Optimal Class of Symmetric Key Generation Systems," Proc. Eurocrypt '84 Workshop Advances in Cryptology, pp. 335-338, 1984.
[4] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly Secure Key Distribution for Dynamic Conferences," Information and Computation, vol. 146, no. 1, pp. 1-23, Oct. 1998.
[5] C. Boyd, "On Key Agreement and Conference Key Agreement," Proc. Second Australasian Conf. Information Security and Privacy (ACISP '97), pp. 294-302, 1997.
[6] E. Bresson, O. Chevassut, D. Pointcheval, and J.-J. Quisquater, "Provably Authenticated Group Diffie-Hellman Key Exchange," Proc. ACM Conf. Computer and Comm. Security (CCS '01), pp. 255-264, 2001.
[7] E. Bresson, O. Chevassut, and D. Pointcheval, "Provably-Secure Authenticated Group Diffie-Hellman Key Exchange," ACM Trans. Information and System Security, vol. 10, no. 3, pp. 255-264, Aug. 2007.
[8] J.M. Bohli, "A Framework for Robust Group Key Agreement," Proc. Int'l Conf. Computational Science and Applications (ICCSA '06), pp. 355-364, 2006.
[9] M. Burmester and Y.G. Desmedt, "A Secure and Efficient Conference Key Distribution System," Proc. Eurocrypt '94 Workshop Advances in Cryptology, pp. 275-286, 1994.
[10] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, "Multicast Security: A Taxonomy and Some Efficient Constructions," Proc. IEEE INFOCOM '99, vol. 2, pp. 708-716, 1999.
[11] J.C. Cheng and C.S. Laih, "Conference Key Agreement Protocol with Non Interactive Fault-Tolerance Over Broadcast Network," Int'l J. Information Security, vol. 8, no. 1, pp. 37-48, 2009.
[12] W. Diffie and M.E. Hellman, "New Directions in Cryptography," IEEE Trans. Information Theory, vol. IT-22, no. 6, pp. 644-654, Nov. 1976.
[13] M. Eltoweissy, M.H. Heydari, L. Morales, and I.H. Sudborough, "Combinatorial Optimization of Group Key Management," J. Network and Systems Management, vol. 12, no. 1, pp. 33-50, 2004.
[14] A. Fiat and M. Naor, "Broadcast Encryption," Proc. 13th Ann. Int'lCryptology Conf. Advances in Cryptology (Crypto '93), pp. 480-491, 1994.
[15] H. Harney, C. Muckenhirn, and T. Rivers, "Group Key Management Protocol (GKMP) Architecture," RFC 2094, July 1997.
[16] K.H. Huang, Y.F. Chung, H.H. Lee, F. Lai, and T.S. Chen, "A Conference Key Agreement Protocol with Fault-Tolerant Capability," Computer Standards and Interfaces, vol. 31, pp. 401-405, Jan. 2009.
[17] IEEE 802.11i-2004: Amendment 6: Medium Access Control (MAC) Security Enhancements, 2004.
[18] I. Ingemarsson, D.T. Tang, and C.K. Wong, "A Conference Key Distribution System," IEEE Trans. Information Theory, vol. IT-28, no. 5, pp. 714-720, Sept. 1982.
[19] J. Katz and M. Yung, "Scalable Protocols for Authenticated Group Key Exchange," J. Cryptology, vol. 20, pp. 85-113, 2007.
[20] C. Laih, J. Lee, and L. Harn, "A New Threshold Scheme and Its Application in Designing the Conference Key Distribution Cryptosystem," Information Processing Letters, vol. 32, pp. 95-99, 1989.
[21] C.H. Li and J. Pieprzyk, "Conference Key Agreement from Secret Sharing," Proc. Fourth Australasian Conf. Information Security and Privacy (ACISP '99), pp. 64-76, 1999.
[22] A. Perrig, D. Song, and J.D. Tygar, "Elk, A New Protocol for Efficient Large- Group Key Distribution," Proc. IEEE Symp. Security and Privacy, pp. 247-262, 2001.
[23] M.O. Rabin, "Digitized Signatures and Public-Key Functions As Intractable As Factorization," Technical Report LCS/TR-212, MIT Laboratory for Computer Science, 1979.

[24] R.L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public Key Cryptosystems," Comm. ACM, vol. 21, pp. 120-
126, 1978.
[25] G. Saze, "Generation of Key Predistribution Schemes Using Secret SharingSchemes," Discrete Applied Math., vol. 128, pp. 239-249, 2003.
[26] A. Shamir, "How to Share a Secret," Comm. ACM, vol. 22, no. 11, pp. 612-613, 1979.

BIOGRAPHIES

ARUNA MATURU, working as Associate Professor in Dept.of Computer Science & Engineering, Swarnandhra Engineering College, Narsapuram, West Godavari District, A.P. Total years of experience: 8 years Educational Qualifications: Completed Post Graduation M.Tech (CSE) in Sri Vasavi Engineering college, Tadepalligudem, affiliated to JNTUK-Kakinada and B-Level, Equivalent to M.C.A. from DOEACC Society, An Autonomous Scientific Society of Dept. of Information Technology, Ministry of Communications and Information Technology, Govt. of India. Interested areas: Computer Networks, Network security & Data mining.

K.TRINADH RAVI KUMAR, Associate Professor, Dept.of Computer Science in S.V.K.P & Dr.K.S.Raju Arts & Science College, Penugonda, West Godavari District, A.P,
Total Years of Experience: 12 years Educational Qualification: Completed Post Graduation M.Tech (CS&E) in Aditya Institute of Technology and Management, Tekkali, affiliated to JNTUK, Kakinada and M.Sc (Computer Science) in Sri Y.N.College, Narasapuram, Affiliated to Andhra University, Visakhapatnam.Interested Areas: Computer Networks, Network security & Data mining.