VOL. 3, NO. 3, March 2012 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

The Improvement of Key Management Based On Logical Key Hierarchy by Implementing Diffie Hellman Algorithm

[1] Nur Alyani Jusoh@Mohd, [2] Kamaruzzaman Seman, [3] Norita Md Nawawi,[4] M. Norazizi Sham Mohd Sayuti

Fakulti Sains Dan Teknologi, Universiti Sains Islam Malaysia, 71800 Nilai, MALAYSIA

Email: [1] nuralyani86@yahoo.com, [2] drkzaman@usim.edu.my, [3] norita@usim.edu.my, [4] azizi@usim.edu.my

ABSTRACT

Several studies have been conducted on how to manage key management in secure and effective key management model in multicast based on Logical Key Hierarchy (LKH). The improvement model based on LKH are discusses in this paper together with Diffie Hellman (DH) algorithm as method for key distribution and result based on computation cost and performances also discussed through this paper. Simulation has been done to test performance of the model. It has been observed that performance process for improvement LKH gives better results in terms of speed, time and security in dynamics communications.

Keywords: *Diffie Hellman (DH), key management, multicast, Logical Key Hierarchy (LKH)*

1. INTRODUCTION

The future growth of internet nowadays has ledto new communication network that emerge into several services where multicast communication is one of these kinds. Almost all types of group applications such as interactive TV, Teleconference and many more are actually based on this service are need to reduce the server's overhead and bandwidth usage by enabling one source to send a single copy of message to multiple receivers in the group. The main problem occurring in this communication lies in making sure that only legitimate user enters in the group communication. Common technique used in secure multicast is to keep a group key that is acknowledgedby all users in the multicast group, but is kept secret to unauthorized user outside from the group. Each time a user wants to join or leave the group, the group key has to be refreshed or can be called as rekeying process[1].The users in the group should be able to generate new group key efficiently, guaranteeing forward and backward secrecy simultaneously. In situation when dynamic user joins and leaves in multicast, the group key needs to be refreshed frequently to make it more secure and updated[2].This basic solution in providing secure and efficient multicast communication. In this paper, we focus on how to manage keys in effective and faster ways to all users but at the same time secure and low communication overhead will also be discussed in details.

2. PREVIOUS WORK

Key management in multicast communication has been studied and a lot of models has been proposed to overcome problem in key management and performance in multicast communication. The simplest way is the group key or also known as Key Server (KS) that can distribute key to user by doing secret unicast connection for each member in the group[3].However this solution is the worst ever in multicast communication because all node in key structure have a dependency on the number of members in the multicast group*(n)*.So in this problem it would become order $n(O(n))$ and is not effective dynamic communication. Wallner et al [4] and Wong et al. [5] introduced a scalable key management scheme by constructing a logical tree of Key Encryption Key (KEK)'s which is called Logical Key Hierarchy (LKH) for a given group.Figure 1 below represents their model

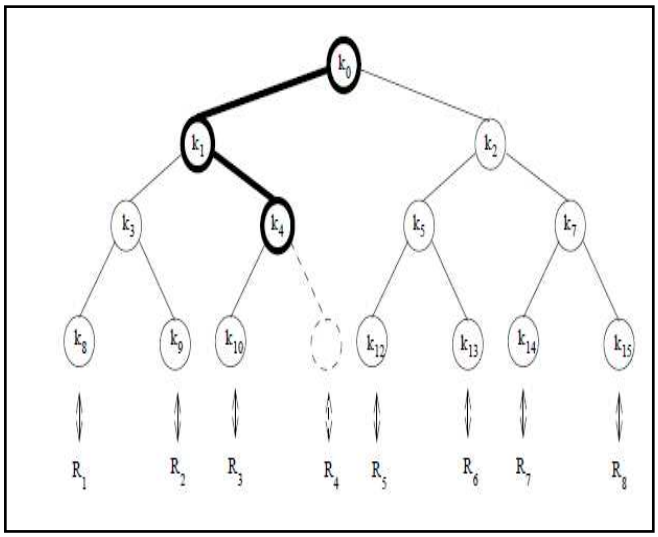

Figure 1: A common logical key hierarchy for a group of eight members (*Source: D. M. Wallner, E.G.H., and R. C. Agee, 1998*)

Since a user shares the root key and all the intermediate KEK's with other users, all the keys contributed by the member except the one at the leaf node has to be rekeyed when the member is deleted. In this model, the complexity is $O(\log n)$ for key update communication. Figure 1 above, if member R_4 wants to leave, k_4, $k1$ and k_0 need to update with k_4', k_1' and k_0' respectively. Then, key sever will broadcasts E_{k10} (k_4'), E_{k13} (k_1'), $E_{k4'}$ (k_1'), $E_{k1'}$ (k_0') and $E_{k2'}$ (k_0'). However, the problem lies when group key is needed to update and it will involve all users that share root path and become inefficient. Some modifications have been done in LKH

VOL. 3, NO. 3, March 2012 ISSN 2079-8407
Journal of Emerging Trends in Computing and Information Sciences

http://www.cisjournal.org

which improve message complexity by a factor of two with small computation like in [6].

3. PROPOSED WORK

In this paper, improvement model that will be proposed is actually enhancement from Logical Key Hierarchy (LKH). A large group of user is divided into several smaller subgroups. Each subgroup will be independently managed by the subgroup controller (SGC) which is one of them will be selected to be a leader and performs as SGC. Figure 1 below shows model that has been introduced.

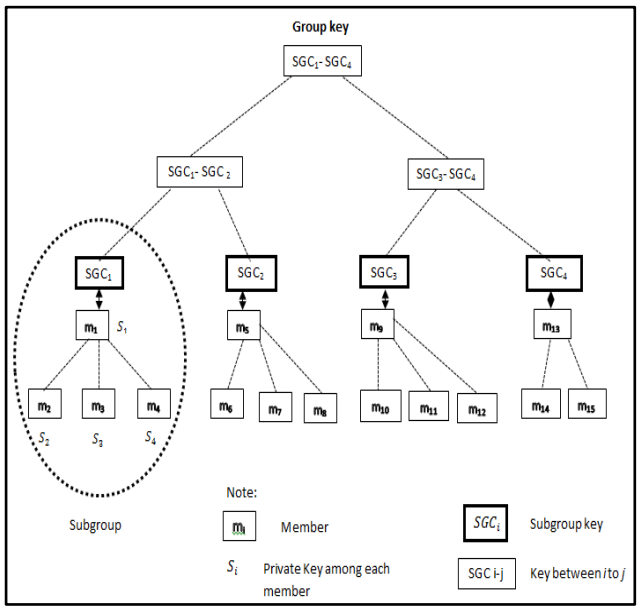

Figure 2: Proposed Model based on LKH

Authentication protocol for users in the group can be obtained by encryption. In this proposed model Diffie Hellman (DH) algorithm for key agreement and El Gamal cryptography as encryption method have been used in order to improve level security in this model.

In this proposed model, DH has been used because it will include in key agreement for shared secret between two entities that later can be used as secret key for both of them [7].

4. ALGORITHM

In our proposed model involves three processes: group generation, user joining and user leaving.

A. Key Generation

For key generation process, first part is to generate subgroup key for each group. This process was done by calculating $SGC_i = g^{si} \bmod n$ where s_i is secret key of user, n is prime and g as base generator. Selected leader in the subgroup will use $SGC_i = g^{si} \bmod n$ subgroup key and distribute to others member in the group. Once users received the SGC key, they will encrypt that key using DH algorithm with their own secret key to retrieve. In order to communicate between group key (GK) and SGC, leader in SGC once again select another secret key $SGC_i' = g^{si'} \bmod n$. It later sends SGC key to user in particular group by encrypting them with their share common key of individual user. Then after done all of this operation, leader of the group can compute their own group key via DH algorithm.

B. User Joining

In order to communicate with other user in secure way, the subgroup key has to be renewed to prevent the new user obtain any information about past communication details. The joining process is shown in Figure 2 and explained as follows:

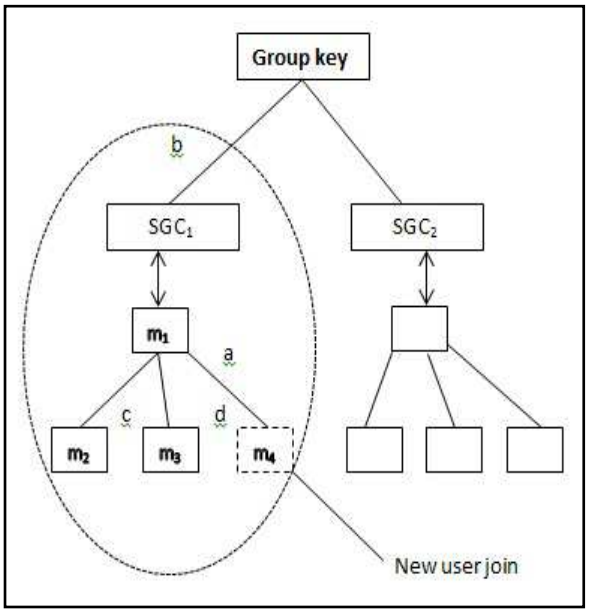

Figure 3: New Users Join

a) When a user wants to join the multicast group, he will send request to the subgroup controllerSGC_i.

b) The SGC_i will evaluate the request. If the user is allowed to join subgroup, the SGC_i informs group controller (GK) that the joining request is authorized and become one member in multicast

c) The SGC_i selects new private and computes its new subgroup key for example in this case m_4 is new user so its private key S_4 and computes $K_4 = g^{s}{}_4 \bmod n$. The new member will DH exchange with their leader in the subgroup to obtain common key

d) Now leader will calculate new subgroup controller key SGC_i and notifies to others members via multicast in subgroup to renew the SGC_i key by encrypting new SGC_i key with old group keyis depicted inFigure 3 below.

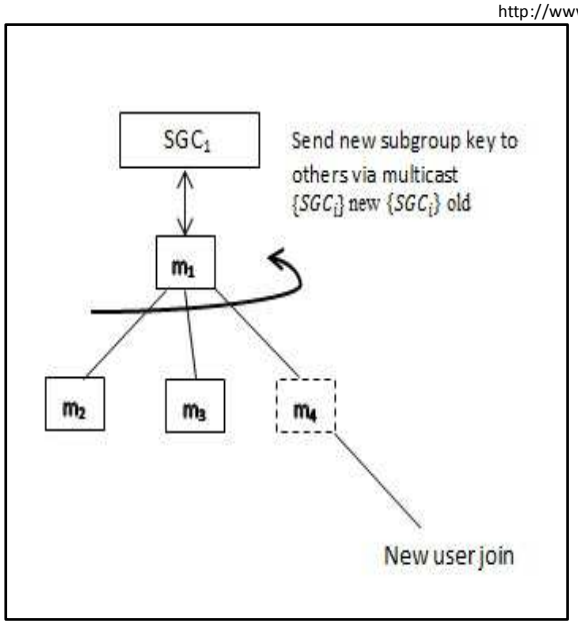

Figure 4: New Subgroup Key Generated

C. User Leaving

When a user wants to depart from the secure multicast group, it should be made sure that leaving user cannot obtain any information about communication whether in past or future. The leaving process is shown in Figure 4 below

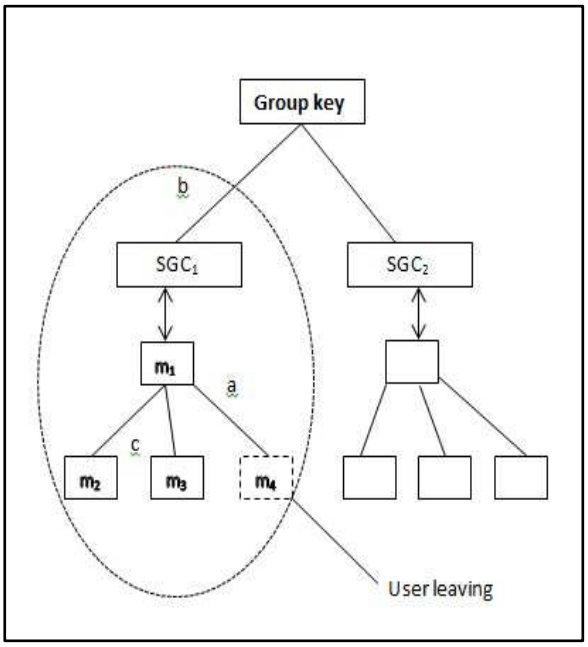

Figure 5: Leaving User

a) If a user wants to leave the group, he will send request to his subgroup key controller SGC_i

b) After SGC_i receives the leaving request, it will inform the group controller the departure of the user.

c) The leader will randomly rekey and select new private key S_i' and compute $K_i = g^{s}_{i}$ 'mod n as new subgroup key. The process to distribute new subgroup key is the same as joining process where subgroup key controller SGC_i will notify remaining members in subgroup to renew the subgroup key with their own subgroup keys.

5. RESULT ANDANALYSIS

The performance of the proposed scheme can be evaluated in terms of the complexity of communication, storage, and computation. From a group controller's standpoint, a storage requirement for key management in the conventional tree-based schemes [4-6, 8] is *O(n),* which means that the group manager should maintain all the node keys (*2n - 1* keys) on the tree. In this proposed model, computation cost based on this three parameters:

a) N which is number of user in the group
b) SGC_i which is total of leader of subgroup in the group
c) n_s which is total of users in subgroup

Since in this model based on DH algorithm, summary of computation cost conclude in the Table 1 below to differentiate performance and cost for different models.

Table 1: Comparison between normal tree based structure and proposed model

Process	Trivia	Tree Based Structure	Improvement of LKH Structure
Number of keys stored in Group Controller	n	$\frac{D}{D-1}$n	*s*
Number of keys stored in user	n	L	3
Number of messages per Join	1	2	s
Number of messages per Leave	n	n	s

n = Number of Users

L = Level of Tree

D = Degree of Tree

s = number of subgroups

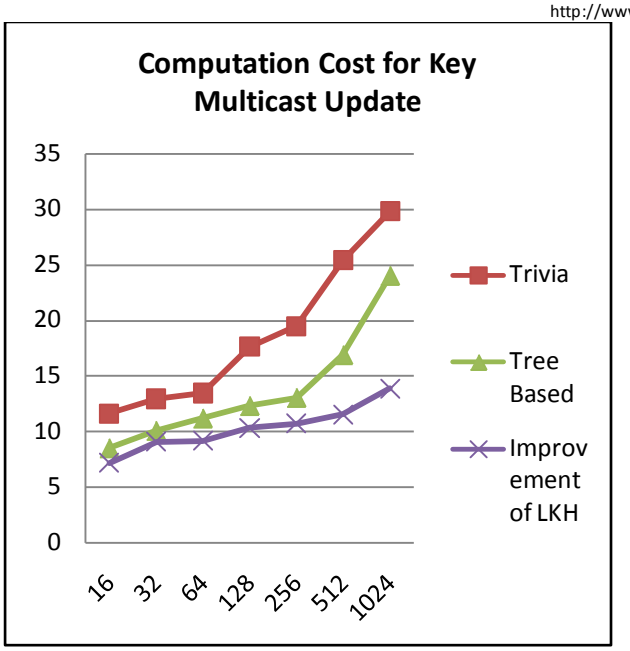

Figure 6: Computation Cost for Multicast Key Update for Different Model

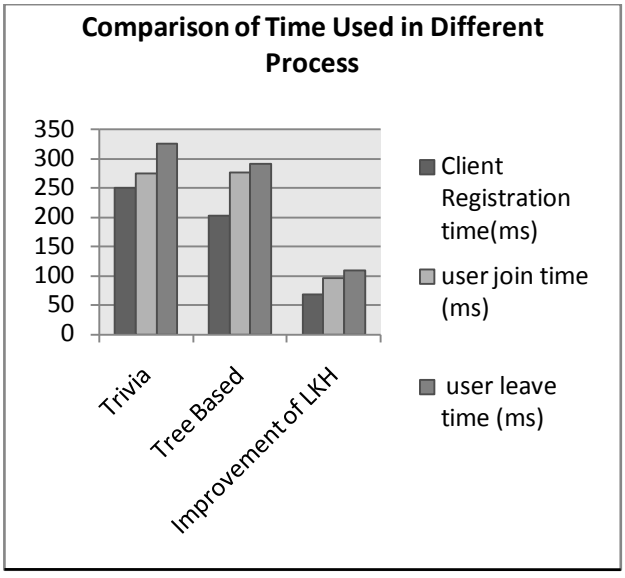

Figure 7: Comparison of Time Used in Different Process

Based on proposed model discussed above, simulation had been done to test performance of the model. The testing that had been done also applies El Gamal as encryption for a message with different number of key bits and size. Overall about 10000 group of user based random number generation had been tested and performance result is represented in Figure 5 and 6 above. Referring from two graphs above, performance process for improvement LKH give better achievement in terms of speed, time and security in dynamic communications.

6. CONCLUSION

Key management in multicast communication is an interesting application of cryptography and an exciting field with many directions to take it. In this paper we discussed about improvement model of key management in multicast based LKH structure and we had implemented Diffie Hellman concept in order to make scalable key management where it can reduce computation of $O(\log n)$ into subgroups. Performance from proposed model we can saw it through result that has been discussed above.

7. ACKNOWLEDGMENT

The author would like to thank to her university, University Sains Islam Malaysia (USIM), Faculty of Science and Technology Universiti Sains Islam Malaysia, USIM grant and also for all who contributed to this study

REFERENCES

1. D.Hutchison, S.R.a., *A Survey of Key Management for secure Group Communication.* ACM Computing Surveys (CSUR), 2003: p. pp. 309-329.
2. R. Srinivasan, V.V., R. Rajaraman, S. Kanagaraj, R. Chidambaram Kalimuthu, and R. Dharmaraj, *Secure Group Key Management Scheme for Multicast Networks.* International Journal of Network Security, 2010. **Vol.11**(No.1): p. pp.33-38.
3. Muckenhirn, H.H.a.C., *Group Key Management Protocol Architecture.* IETF International RFC 1997.
4. D. M. Wallner, E.G.H., and R. C. Agee, *Key Management for Multicast: Issues and Architecture.* IETF International RFC, 1998.
5. Wong, M.G., and S. S. Lam, *Secure Group Communication Using Key Graphs.* Journal IEEE/ACM Transactions on Networking, 2000. **Vol. 8**(No.1): p. pp. 16 - 30.
6. R. Canetti, G., G. Itkis, D. Micciancio, M. Naor, and B. Pinkas, *Multicast Security: A Taxonomy and Efficient Authentication.* Proceeding of IEEE Infocom, 1999. **Vol.2**: p. pp.708-716.
7. Scheneir, B., *Applied Cryptography.* 2nd Edition ed. 1996: John Wiley and Sons, New York.
8. D. A. McGrew, a.A.T.S., *Key Establishment for Large Dynamic Groups: One-way Function Trees and Amortized Initialization.* IETF International RFC, 1999.

Nur Alyani Jusoh @ Mohd received the B.S degrees in Computer Science from Universiti Sains Islam Malaysia in 2009 respectively. Currently, she is doing her master in computer science focusing on Network and Cryptography

Kamaruzzaman Seman is a professor at Universiti Sains Islam Malaysia. He is also one of the senior members IEEE USA since 1998.He obtained his Bachelor from Univertiti Teknologi Malaysia, Malaysia in 1985. Then in 1986, he got his M.Sc in Telematics from Essex University.

In1994, he obtained his PhD in Broadband ISDN from Strathclyde University UK. He experienced as being a chairman of Sub-Working Group 3 (Information Superhighway) of MNIC (now known as MNIIF),chairman of Next Generation Network (NGN), Telekom R&D S/B Task Force. (June 2003 – July 2005) and many more.

Norita Md Norwawi is an Associate Professor at Universiti Sains Islam Malaysia. She obtained her Bachelor in Computer Science in 1987 from the University of New South Wales, Australia. She received her Masters degree in Computer Science from National University of Malaysia in 1994. In 2004, she obtained her PhD specializing in Temporal Data Mining and Multiagent System from University Utara Malaysia. As an academician, her research interests include artificial intelligence, multi-agent system, temporal data mining, text mining, knowledge mining, information security and digital Islamic application and content. Her works have been published in international conferences, journals and won awards on research and innovation competition in national and international level.

M. Norazizi Sham Mohd Sayuti is a lecturer at Universiti Sains Islam Malaysia (USIM). He obtained his Bachelor of Engineering (Electronics Engineering) from the Shibaura Institute of Technology, Japan. Norazizi has a Master of Science in Computer Science (Software Engineering) from the Universiti of Teknologi Malaysia. He worked as a Software Engineer and has many experiences in developing cryptography applications. His research interests including Systems-on-Chips Design Space Explorations, Network-On-Chips, Real-Time Embedded Systems Design, Asymmetric and Symmetric Key Management, Data Encryption and Decryption and Safety-Critical applications.