ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Website: www.ijircce.com

Vol. 4, Issue 12, December 2016

A Survey on Group Key Agreement with Local Connectivity

Dhanashri Kulkarni, Prof. D.O.Shamkuwar

M.E Student, Dept of Computer Network, Flora Institute of Technology, Savitribai Phule Pune University, India

Dept of Computer Network, Flora Institute of Technology, Savitribai Phule Pune University, India

ABSTRACT: Because of the increased popularity of group oriented application and protocols, group communication occurs through tele and video conferencing. Security is the main way to provide privacy while communicating with each other. In this paper we study a problem of group key agreement in which user is familiar with his neighbour and he has no idea about locality of various customer while the connectivity graph is inconsistent. In this the problem is that there is no centralized initialization for users. For the social networks a group key agreement with these is no centralized initialization for users. For the social networks a group key agreement with these feature is suitable. We establish two coherent protocols with passive security. We replace the Diffie Hellman key exchange algorithm with another multicast key exchange algorithm. We get lower bounds on the round complexity protocol, which determines that our constructions are round systematic. Finally we establish an actively secure protocol from a passively secure protocol.

KEYWORDS: Group key agreement, Diffie-Hellman, lower bound, authentication, protocol.

I. Introduction

The mechanism of sharing secrete key between two or more parties in secure manner. There is one mechanism called as key agreement. That secrete key is also called as session key. Basically in the two-parties case Diffie-Hellman protocol is used. In this real world its not possible that 2 users can communicate directly with each other. But many of the protocols assumes a complete group of connectivity which means that two users can communicate directly.

There are many social networking sites like Facebook, Skype, Wechat and Google+ in which users is connected with his friends. For a group of users who wants to establish a secrete key it’s not required that any of two are friends from that group. But they are connected with each other through that n/w indirectly. But still we consider then as they are directly connected by regarding users as routers. Although it is dissimilar from a direct correction.

Firstly, the users which are connected indirectly with each other they don’t have the public information about each other. This public information is nothing but public key certificate. Second the customers which are indirectly connected they don’t know about the presence of each other. Third is that a message which is travels between two users who are indirectly connected, requires more time than that between users who are directly connected.

Now we study the group key agreement with an inconsistent connectivity group, in which each client is only aware for his neighbouring client and he has no idea about the region of different clients. Also he has no information about network topology. By using this setting a customer does not necessary to trust a customer who is not neighbour. In this manner, if one of users is initialized by using PKI, then that users need not to aware or remember public keys of user other than his neighbour.

II. Related work

Key pre-distribution scheme (KPS) (a.k.a. non-interactive conference distribution system) can be regarded as a non-interactive group key agreement. In this case, after the setup the shared key is fixed of given group. If a group is updated, then the group key changes to the shared key of the new group. The drawback of KPS is that the user key size is conditionally large [12], [36] in the total number of users (if the system is unconditionally secure). Second drawback is that the group key of a given group can not be changed even if it is leaked without any warning (e.g., cryptanalysis of ciphertexts bearing this key). The key size doubt may be affected if a estimate secure system is used, while the key leakage problem is not obvious. For the two-party case and the three-party case secure KPS is only known. KPS with a group size greater than 3 is still open.

The mechanism of broadcast encryption allows a sender to send a group key to a preferred set of users. This can be regarded as a group key agreement of one message that is sent by the sender. In a symmetric key based broadcast encryption the sender is a fixed authority. In this case, the user key size is combinatorially lower bounded . In addition, the user key secure for limited number of users. The key size problem can be relinquish in a public key broadcast encryption. But one still has to set the threshold for the number of bad users. Also the size of the ciphertext build upon the number of users and hence could be large (e.g., it p is O(n) in for n users). Further, users are initialized by a central authority which is not desired in our setting.

The pirate user can be trace out using the Traitor tracing which is a special broadcast en-cryption, where besides the usual broadcast capability: if a user helps build an illegal decryption device, he will be identified. This primitive inherits the drawbacks of a broadcast encryption.

A rekey scheme for a multicast can be regarded as a centralized dynamic broadcast encryption, where the authority always maintains the group as the set of dynamically changing privileged users, and updates the group key and some user keys whenever there is a membership change. Drawback of this mechanism has user key is provided by a centralized authority and has to be modernize upon a member leave. If a group key agreement approve this system, then the user key will be upgrade whenever the group changes. This is not desired. In addition, in a group key agreement setting, some times it possible that multiple groups wants to establish a group key at the same time. We can not adopt a rekeying scheme as a group key agreement because a rekeying scheme can not handle this case.

In all of KPS, broadcast encryption, traitor tracing and a rekey scheme, a user key is used by a single central authority and there is a rely on the keys of different users. Above first three mechanisms have a approach for the number of fraud. A centralized setup is not appropriate and it is also not possible to determine a corruption threshold in key agreement problem. Hence, For a group key agreement in setting they are not sensible candidates.

Throughly secure (interactive) key agreement has been considered in. Beimel and Chor showed from a domain of size at least jSj ,the user key in this setting must be taken, where S is the domain of the group key and is the maximum number of key agreements. If the user key is distributed uniformly (the typical case), it has an entropy of at least log jSj: In real applications, is usually large. Hence, this type of scheme does not provide an efficient solution even though it is unconditionally secure.

We now studied the computationally secure group key agreement in a compliant model. This started from the Diffie-Hellman protocol. In the following, we use the tuple (a; b; c) to represent a protocol that has a rounds, b elements of messages per user (the unit is a field element in Z_p for a large prime p) and computation cost c. Ingemarsson et al. designed a group key agreement for n users in a ring with an efficiency tuple (n 1; n 1; ne), where e stands for one exponentiation in Z_p. Burmester and Desmedt designed a more efficient protocol with an efficiency tuple (2; 2n; 4e), after ignoring the exponentiations with small exponents and identifying one division with an exponentiation. Their protocol estimate a complete connectivity graph. Steiner et al. proposed three protocols, where the most efficient one has an efficiency tuple (n + 1; 4; 5e). According to this protocol a complete connectivity graph and User n has a big computation cost of (n 1)**e** and a communication cost of n 1 messages. Wu et al. proposed a transport-like protocol through a novel aggregate-signature based broadcast from pairing. Their protocol has one round (or 2 rounds if the group setup is counted), 3 elements of message from the initiator and computation cost of two pairings and one division, when the setup of group public key is completed, while every group public key needs one round and each user needs a cost (1+n)e+1p+[3(n 1)+1]m, where p is a pairing and m is a multiplication. By Lv et al. using NTRU , the strategy of a transport protocol using an aggregate public-key is also executed, although we feel that it is hard to obtain a provable security as NTRU does not have. Both presume a complete connectivity graph.

Through the join and leave operations some of the group key agreements handle a group change (similar to the strategy of a rekey scheme in a multi-cast);

In an active mode we now observe the computationally secure group key agreement l. Tzeng and Tzeng proposed protocols in the random oracle model, where the construction which is intrested has an efficiency tuple (2; 3n; 2n + 1). Bresson et al. formalized a formal model for a group key agreement in the active model and made the protocol in secure with the help of signature based authentication. The Burmester-Desmedt protocol in the active model with a signature based authenticator is implemented by Katz and Yung. The transport-like 2-round protocol in the random oracle model is proposed by Boyd and Gonzalez´-Nieto, where one user needs to enumerate n public-key encryptions and each user has an outgoing message of length (n). In this paragraph presume a complete connectivity at all the works.

We are attentive in a protocol in the absence of a random oracle. From the discussion above, we can see that the compliant secure protocols that are really applicable to us are . They are not depend on a random oracle model and the if there are any long term secrets between users do not have dependency. We will compare them with our protocols. The only issue when we think as a solution in our setting is that the connectivity graph in them is either a ring or a complete graph and every user is aware about it. A user is only aware of his neighbours and has no information about others in our setting. For actively secure protocols viewed above, are interesting. But they only executed passively secure protocols. We will not compare them with us since we are mainly worried with the key agreement methodology (rather than how to acquire stronger security).

III. PROPOSED ALGORITHM

A. *STAGE ONE*:

- Each i 2 V takes a_i Z_q and sets $A_i = g^{ai}$:
- Each leaf user s in G (i.e., N_s = fig) sets A_{si} = 1 and sends (A_{si}; A_s) to i:
- [**Loop**] Each ` 2 V does the following.
- For i 2 N`, if user ` has received ($A_{j`}$; $A_j$) from each j 2 N`nfig and did not send ($A_{`i}$; A`) to i, then he computes $A_{`i}$ = $_{j2N`nfig}A_jA_{j`}$ and sends ($A_{`i}$; A`) to i.
- Each user ` continues step 2 until he has sent ($A_{`i}$; A`) to user i for each i 2 N`, in which case, he proceeds to Stage two. Let $_{`i} = g^{a`ai}$.

B. *STAGE TWO:*

- Each leaf user s (i.e., N_s = fig) computes $L_{si} = A^{a}_{is}{}^{s}$ and sends C_{si} = E $_{si}$ (L_{si}) to user i:
- [**Loop**] Each ` 2 V does the following.
- For i 2 N`, if user ` has received $C_{j`}$ from
- each j 2 N`nfig and did not send $C_{`i}$ to i, then he decrypts $L_{j`}$ = D $_{j`}$ ($C_{j`}$), defines
- $L_{`i}$ = ( $_{j2N`nfig}L_{j`}$) ( $_{j2N`}$ $A_{j`})^{a`}$ and sends
- $C_{`i}$ = E $_{`i}$ ($L_{`i}$) to user i.
- Each user ` continues step 2 until he has sent $C_{`i}$ to user i for each i 2 N`, in which case, he proceeds to Stage three.
-

C. *STAGE THREE:*

- *Upon C_s* for all s 2N; user decrypts L_s =D_s (C_s) (if not done before) and calculates group key S_k=$(LA)^a$ Q

IV. SYSTEM OVERVIEW

A. *SYSTEM ARCHITECTURE:*

To study the group key agreement with an arbitrary connectivity graph, where each user is only aware of his neighbours and has no information about the existence of other users. Further, user has no information about the network topology. Under this setting, a user does not need to trust a user who is not his neighbour. Thus, if one is initialized using PKI, then he need not trust or remember public-keys of users beyond his neighbours. Which means that user has to firstly user has to register his group. Then after the assignment of key user need to choose the destination node. When user select its destination node then by using that node user can send encrypted data. Then at the receiver end receiver needs to verify the key. When verification of key is done then receiver can receives that decrypted data. To enhance secure data communication between users, we propose, data encryption to convert data to cipher text and sent to receiver, where as receiver need to provide sender session key to retrieve the data.

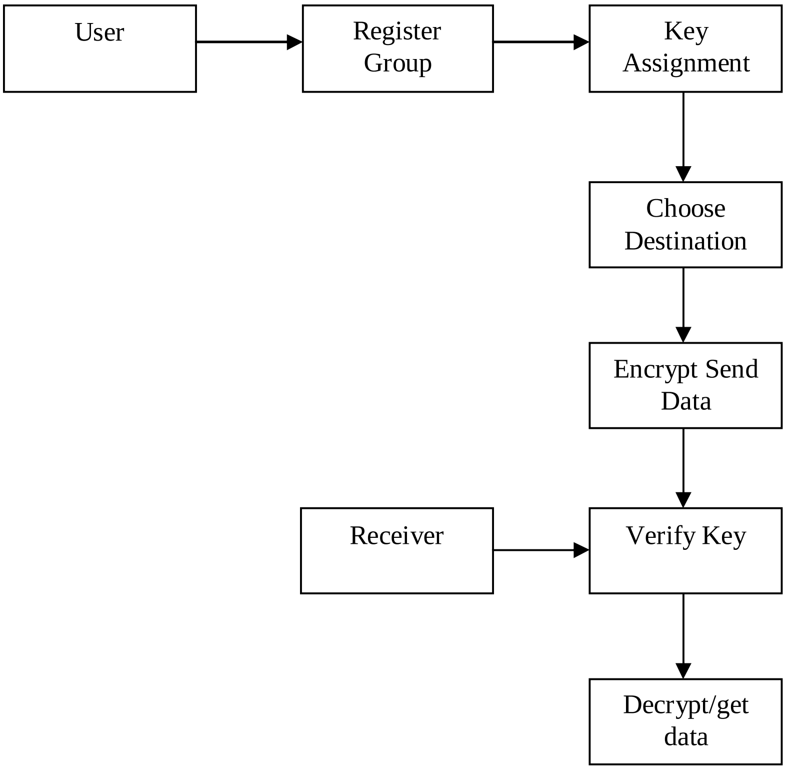

Fig. System Architecture of Group Key Agreement with Local Connectivity

Modules

- Node deployment
- Diffie-Hellman assumption
- Sender model
- Data security
- Receiver model

Node deployment

Nodes are deployed in wireless network, with unique identity. Nodes are registered with groups. Group nodes can communicate with other group member. Consider a spanning tree of G and n is the number of users in G. Further, d(G) is the maximum distance between nodes in G. For example, d(G) = 2 for a star-like G. Since the compared protocols are passively secure and assume a special (fixed) connectivity graph, passively secure protocols in the table assume an arbitrary but fixed connectivity graph, due to which the cost to find G is not included.

Diffie-hellman assumption

Let p; q be two large primes and qj(p-1). Let G be the subgroup of Z p of order q and g be a generator of G. The decisional Diffie-Hellman assumption is as follows. The decisional Diffie-Hellman assumption (DDH) holds if (g; gx; gy; gxy) and (g; gx; gy; gz) are indistinguishable when x; y; z ← Zq. Every nodes are assigned with Diffie Hellman key and the key is checked with the node for every time interval specified.

Sender model

User may be a sender, will create a unique session key for communicating with other group members. For security of node, node id and its key values are checked. The set of neighbours of i in G is denoted by Ni(G). The protocol allows users in V to agree on a shared key. Each user i in the protocol can only send messages to his neighbours Ni(G). Since user i has no knowledge about users other than Ui, we must facilitate him to determine Ni(G): Toward this, we assume that there is a basic description of G (denoted by basic(G)) such that with Ui and basic(G), user i can easily determine Ni(G): basic(G) is determined by the protocol initiator and it will appear in the first incoming message of any user.

Data security

Sender will encrypt the data with cryptographic module, the cipher text is generated and sent to receiver. User can send the request to his union neighbours and interact with them, who then continue the similar interaction with their own union neighbours, and so on. Finally, union members can obtain a group key.

Receiver model

User i will send a temporary DH public key to each of his neighbour j so that they can share a pair wise DH key, and he will also generate a secret as his share for the final group key. Specifically, user i takes a secret key and this is used to retrieve data from the sender node.

V. Conclusion

We studied a group key agreement problem, where a user is only aware of his neighbours while connectivity graph is arbitrary. In addition, users are initialized completely independent of each other. A group key agreement in this setting is very suitable for applications such as social networks. We constructed two passively secure protocols with contributiveness and proved lower bounds on a round complexity, demonstrating that our protocols are round efficient. Finally, we constructed a actively secure protocol from a passively secure one. In our work, we did not consider how to update the group key more efficiently than just running the protocol again, when user memberships are changing. We are not clear how to do this. One can either propose algorithms to our current protocols (as Dutta and Barua [22] did for [17]) or construct a completely new key agreement with these features. We leave it as an open question.

References

[1] R. Blom, “An Optimal Class of Symmetric Key Generation Sys-tems”, Proc. Advances in Cryptology-EUROCRYPT’84, vol. 209, pp. 335-338, 1984.

[2] C. Blundo, A. De Santis, A. Herzberg, S. Kutten, U. Vaccaro and M. Yung, “Perfectly Secure Key Distribution for Dynamic Conferences”, Inf. Comput., vol. 146, no. 1, pp. 1-23, 1998.

[3] C. Blundo, L. A. Mattos and D. R. Stinson, “Generalized Beimel-Chor Schemes for Broadcast Encryption and Interactive Key Distribution”, Theor. Comp. Sci., vol. 200, no. 1-2, pp. 313-334, 1998.

[4] C. Blundo and A. Cresti, “Space Requirements for Broadcast Encryption”, Proc. Advances in Cryptology - EUROCRYPT 1994, vol. 950, pp. 287-298, 1995.

[5] D. Boneh and M. K. Franklin, “An Efficient Public-key Traitor Tracing Scheme”, Proc. Advances in Cryptology (CRYPTO’99), vol. 1666, pp. 338-353, 1999.

[6] D. Boneh, C. Gentry and B. Waters, “Collusion Resistant Broad-cast Encryption with Short Ciphertexts and Private Keys”, Proc. Advances in Cryptology (CRYPTO’05), vol. 3621, pp. 258-275, 2005.

[7] D. Boneh, A. Sahai and B. Waters, “Fully Collusion Resistant Traitor Tracing with Short Ciphertexts and Private Keys”, Proc. 25th Int’l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT’06), vol. 4004, pp. 573-592, 2006.

DOI: 10.15680/IJIRCCE.2016. 0412123

[8] D. Boneh and M. Naor, “Traitor Tracing with Constant Size Ciphertext”, Proc. 15th ACM Conf. Computer and Comm. Security,
a. 501-510, 2008.

[9] D. Boneh and A. Silverberg, “Applications of Multilinear Forms to Cryptography”, Contemporary Mathematics, Vol. 324, American Mathematical Society, pp. 71-90, 2003.

[10] M. Burmester and Y. Desmedt, “A Secure and Efficient Con-ference Key Distribution System”, Proc. Advances in Cryptology-EUROCRYPT’94, vol. 950, pp. 275-286, 1994.

[11] R. Canetti, J. A. Garay, G. Itkis, D. Micciancio, M. Naor and B. Pinkas, “Multicast Security: a Taxonomy and Some Efficient Constructions”, Proc. IEEE INFOCOM 1999, vol. 2, pp. 708-716, 1999.

[12] W. Diffie and M. Hellman, “New Directions in Cryptography”,IEEE Trans. Information Theory, vol. 22, pp. 644-654, 1976.

[13] R. Dutta and R. Barua, “Provably Secure Constant Round Con-tributory Group Key Agreement in Dynamic Setting”, IEEE Trans. Information Theory, vol. 54, no. 5, pp. 2007-2025, 2008.

[14] A. Fiat and M. Naor, “Broadcast Encryption”, Proc. Advances in Cryptology (CRYPTO’93), vol. 773, pp. 480-491, 1994.

[15] Y. Kim, A. Perrig and G. Tsudik, “Tree-based Group Key Agree-ment”, ACM Trans. Inf. Syst. Secur., vol. 7, no. 1, pp. 60-96, 2004.

[16] K. Yongdae, P. Adrian and G. Tsudik, “Group Key Agreement Efficient in Communication”, IEEE Trans. Computers, vol. 53, no. 7, pp. 905-921, 2004.

[17] M. Steiner, G. Tsudik and M. Waidner, “Diffie-Hellman Key Distribution Extended to Group Communication”, Proc. 3rd ACM Conf. Computer and Comm. Security (CCS’96), pp. 31-37, 1996.

[18] C. K. Wong, M. Gouda and S. S. Lam, “Secure Group Communi-cation Using Key Graphs”, Proc. ACM SIGCOMM’98, pp. 68-79, 1998.

[19] Q. Wu, Y. Mu, W. Susilo, B. Qin and J. Domingo-Ferrer, “Asym-metric Group Key Agreement”, Proc. 28th Int’l Conf. Theory and Application of Cryptographic Techniques (EUROCRYPT’09), vol. 5479, pp. 153-170, 2009.

[20] T. Matsumoto and H. Imai, “On the Key Predistribution System: A Practical Solution to the Key Distribution Problem”, Proc. Advances in Cryptology (CRYPTO’87), vol. 239, pp. 185-193, 1987.

[21] X. Lv, H. Li and B. Wang, “Group Key Agreement for Secure Group Communication in Dynamic Peer Systems”, J. Parallel Distrib. Comput., vol. 72, no. 10, pp. 1195-1200, 2012.

[22] R. Safavi-Naini, S. Jiang, “Non-interactive Conference Key Distri-bution and Its Applications”, Proc. the 2008 ACM Symposium on Information, Computer and Communications Security (ASIACCS’08), pp. 271-282, 2008.

[23] R. Safavi-Naini and S. Jiang, “Unconditionally Secure Conference Key Distribution: Security Notions, Bounds and Constructions”,International Journal of Foundations of Computer Science, vol. 22, no. 6, pp. 1369-1393, 2011.