DISTRIBUTIVE COLLABORATIVE KEY AGREEMENT PROTOCOL FOR DYNAMIC PEER GROUPS

Chilla Sruthi[1], Yarlagadda Sravani[2], Gampa Ravi Kumar[3], Laskey Babla[4] , N Praveena[5]

[1,2,3,4] IV/IV B Tech, [5]Assistant Professor, Department of Information Technology, 4th YEAR, VRSEC, Vijayawada, India

---***---

Abstract - *There are several distributive collaborative key agreement protocols for dynamic peer groups. This problem has several important features that makes it different from existing group communication which are listed down as follows. Firstly, they are distributed in nature where there is no centralized key server, and they are collaborative in nature which means that each and every person in the group contributes their own part in generating the group key and finally dynamic in nature where existing members can leave the group and new members may join. We need a protocol that allows the members of the group to communicate in a risk freeway so that they can share private, confidential and vital information over the internet. Cryptography has become an important tool for protecting the important information. So, we use Diffie-Hellman key exchange to generate the group key to solve the problem. This is one of the most west known asymmetric algorithms but is restricted for two users. We use the concept of an extended Diffie-Hellman algorithm known as*

Tree-Based Diffie-Hellman algorithm to generate the group Key.

Key Words: Distributive, Collaborative, Dynamic, Cryptography, Diffie – Hellman, group key

1. INTRODUCTION

These days there is a vital growth in the need of group communication, but there is also a growing problem of communication privacy which requires some sort of protection. For such security there should be a common group key. The group members agree on the common key for secure communication. The nature of the group is dynamic where a member can join or leave the group at any time. The key is also distributed and contributory within the peers of the group.

If there is no group key for communication and if the communications are being done using broadcasts or some open mechanisms then there is a lot of risk coming forth. There will be attackers ready at all the time to loot the information. If the information is just a common greeting or information which is not of much importance then there is no need to worry but if the information contains sensitive data like country defense secrets, financial mechanisms, etc. then we cannot send the information without any security. If doesn't take care of the security many kinds of attacks will take place such as, denial of service, spoofing, etc. The basic approaches for group key management include centralized, distributed and collaborative.

A *centralized group key management* is the simplest way to generate and distribute keys among the group as it involves only single entity. Though it is very simple it involves some disadvantages, in this concept the central key server should always be available else the system may stop functioning due to the unresponsiveness from the server. Another problem is the collapse of the central key server, it maybe unrecognizable and leads to inactiveness of the group. Major problem arises when the channel used to distribute keys is attacked, the attacker can now know the keys and may perform any actions on the messages being sent leading insecure communication.

The *distributed group key management* is the best way when the channel of communication is unreliable. This method uses dynamic approach to select a key server at a particular time. While changing the server the data structures and other information need to be recreated or updated resulting in high computations.

Collaborative group key management is a type where each group member contributes in generating the group key. Every member takes equal share and the final result is a common group key. This the best suitable concept for dynamic peer groups which avoids single point of trust and failure. Unlike other most group key distribution protocols, these offer strong key management security properties like key independence and perfect forward secrecy.

2. LITERATURE SURVEY

The Diffie-Hellman algorithm was first published in the year 1976 by Whitefield Diffie and Martin Hellman. This is the first scheme where two users agree on a common key for their communication. This algorithm uses arithmetic modules as the basis of its calculation. The key is very difficult to be acquired by the hacker. It gives most secure communication as it is very difficult to solve the discrete logarithm and the shared key is never itself transmitted over the channel. Resembling the 2 sides of the coins, DH algorithm has its own drawbacks. The main goal of the algorithm is to generate common key for communication which doesn't involve encryption and decryption by default. The algorithm uses expensive exponential operations and lacks authentication.

Group Diffie-Hellman is an extension of 2-party Diffie-Hellman key exchange algorithm and is proposed by Steiner and Tsaddik. This protocol also doesn't involve sharing of key over the channel. Here a set of partial keys are sent. A group controller is entrusted to build and distribute the key set. He sends a token between the members of the group to collect the contributions by new members. Unlike the centralized theme, the group controller doesn't have any special security privileges. The protocol works as follows: When a merge event occurs (new member join), the current controller refreshes its own contribution to the group key and generates a new token and passes it to one of the new members. When the members leave the group, the controller removes their corresponding partial keys from the set of partial keys.

Sherman (2003) introduced a new method to compute group key by using a special one-way function to compute a tree of keys called One-way function tree (OFT) algorithm, which is an improvement over the binary tree and reduces the size of rekeying message from ($2\log_2 n$) to ($\log_2 n$). The algorithm works as follows: a new member is always joined at a leaf node closest to the root to maintain the balance of the key tree. When a member evicts from the group his sibling is made as the parent position. This approach is good in reducing the rekeying broadcasts and has the problem of collision.

Tree based Group Diffie-Hellman (TGDH) is proposed by Kim in the year 2001. This uses the concept of Group Diffie-Hellman protocol and uses tree structure to arrange the keys. Every member only holds the keys along their path, which distributes the rekeying workload to all the members. The main advantages of this concept are: there is no Group Controller, each member does equal work and the message size b is constant.

3. PROPOSED SYSTEM

The proposed system involves a collaborative key agreement where all the users in the group become a part of key group. Moreover rekeying is done after join or leave of every user. This also remains efficient even when join or leave events are very frequent with less computational and communication cost. Along with this, rekeying gives two utmost advantages like Backward Secrecy where a new user who got added to the group just then is unable to decrypt the information prior to his introduction and Forward Secrecy where a user who got deleted from the group is unable to his deletion. As the information of key does not depend on centralized key server, it is free from single point of failure.

3.1 DIFFIE HELLMAN KEY EXCHANGE:

Asymmetric Encryption of the data requires the transfer of a key from one user to the other without anyone interrupting this key. Instead of transferring the key, The Diffie-Hellman algorithm makes it possible for the user to generate the keys rather than transferring them using some computation which at the end results in keys that are equal on either side. This algorithm is developed by Whitfield Diffie and Martin Hellman in 1976. This algorithm is designed to generate the key but does not encrypt the information. The generated key and the data are given to another algorithm like AES, DES to encrypt the data. This is the most popular and first published key generation algorithm that is used for secure key exchange mechanism.

The main purpose if this algorithm is it ensures that no user apart from A and B can learn any information from the agreed value.

The general algorithm of Diffie-Hellman key exchange is as follows:

Step I.
Select a prime value p, and a random number generator a.

Step II.
Generate 2 random numbers with the help of random() function
a=Math.random()
b=Math.random()

Step III.
Generate 2 keys using the mod operation
a1=g.modpow(a,p)
b1=g.modpow(b,p)

a sends a1 to b
b sends b1 to a

Step IV
Key is generated on either sides using

KeyACalculates = g.modpow(a1,p)
KeyBCalculates = g.modpow(b1,p)

Finally KeyACalculates= KeyBCalculates

But as this is restricted to 2 users, this algorithm makes us of Tree based Diffie Hellman algorithm that is specified down below:

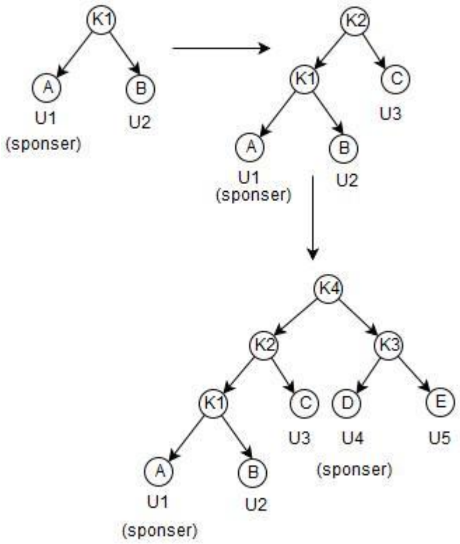


Fig 1 : Tree Based Diffie-Hellman Algorithm

Each user is placed in the leaf node and the key which each of them is using to communicate is placed in an intermediate node.

Every user will be having two keys:

1) Private key K_u
2) Public key (generated using private key), PK_u

$PK_u = \alpha^{Ku}$ modp

Where,
α is generator
p is prime number
Secret Key, k = $(PK_u)_{Ku}$ modp

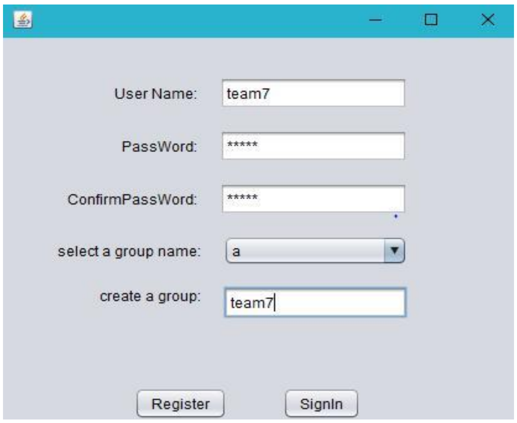
The algorithm goes as follows:

1. If there are no members, then create a new group with that member.
2. If there is a group, then find the insertion point and add new member to it.
3. Consider the left most person in the group as sponsor
4. If any member wishes to leave, then delete him from the group

(It is then the duty of sponsor to update the group key and broadcast the key)

4. RESULTS

A login page is provided where we can enter our credentials:



User Name: team7

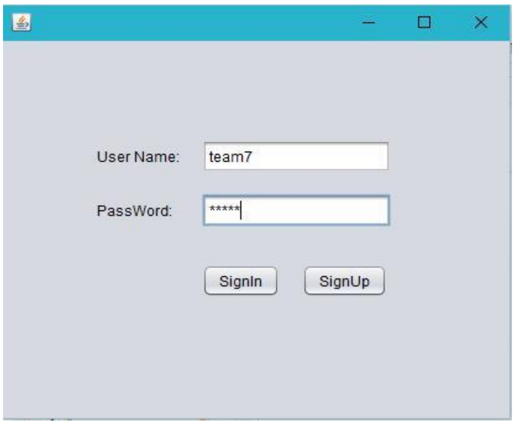
PassWord: *****

ConfirmPassWord: *****

select a group name: a

create a group: team7

Fig 2 : Sign up Page

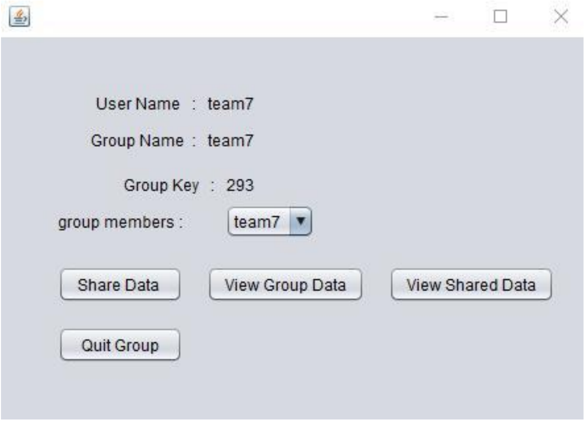


User Name: team7

PassWord: *****

Fig 3 : Login page

After logging into the page, home screen is displayed which looks like this:



User Name : team7

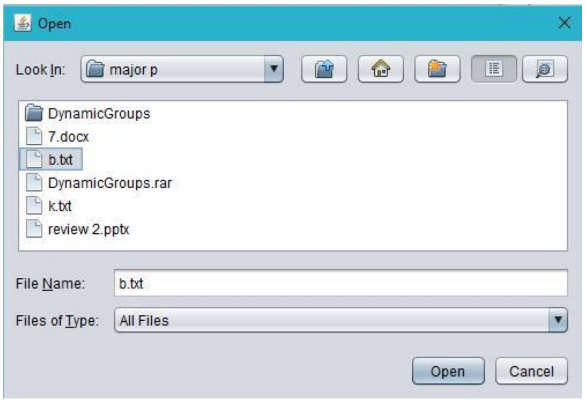
Group Name : team7

Group Key : 293

group members : team7

Fig 4 : Home screen

If one wants to share data, they will click on share data:


Look In: major p

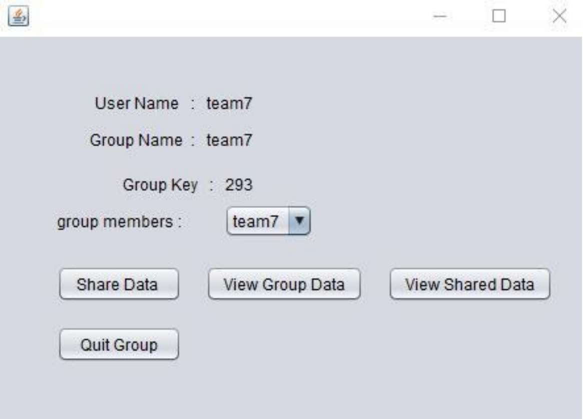
DynamicGroups
7.docx
b.txt
DynamicGroups.rar
k.txt
review 2.pptx

File Name: b.txt

Files of Type: All Files

Fig 5 : GUI to share data

So as to view the members of group, click on view group data:


User Name : team7

Group Name : team7

Group Key : 293

group members : team7

Fig 6: Displaying group members

5. CONCLUSIONS AND FUTURE WORK

The Diffie Hellman Key exchange exploits mathematical properties and produces a common result between two or more parties who are wishing to exchange information without any of them particularly providing any type of information. They agree on 2 variables and the resulting secret key is identical without exchange. But, it is possible to intervene by masquerading or brute force but the first common concern is authentication. With all of these done properly, DH provides a powerful security component.

In future fault tolerance can be implemented and further efficient authentication protocols can also be implemented

REFERENCES

[1] W. Diffie and M. Hellman. "New directions in cryptography.IEEE Transactions on Information Theory", IT-22(6):644–654, 1976.

[2] Y. Kim, A. Perrig, and G. Tsudik. "Simple and fault-tolerant key agreement for dynamic collaborative groups". Proc. Of 7th ACM Conference on Computer and Communications Security,pages 235–244, November 2000.

[3] Y. Kim, A. Perrig, and G. Tsudik. "Communication-efficientgroupkeyagreement. Information Systems Security", Proceedingsof the 17thInternational Information Security ConferenceIFIP SEC'01, November 2001.

[4] P. P. C. Lee, J. C. S. Lui, and D. K. Y. Yau." Distributed collaborativekey agreement protocols for dynamic peer groups".Technical report cs-tr-2002-04, Dept of Computer Scienceand Engineering, Chinese University of Hong Kong,. Also as CS TR-02-013, Purdue University, WestLafayette, IN.May2002

[5] X. S. Li, Y. R. Yang, M. G. Gouda, and S. S. Lam. "Batchrekeying for secure group communications". Proceedings ofTenth International World Wide Web Conference (WWW10),May 2001.

[6] S. Setia, S. Koussih, and S. Jajodia. Kronos: "A scalable groupre-keying approach for secure multicast". Proc. of IEEE Symposiumon Security and Privacy 2000, May 2000.

[7] W. Stallings. "Cryptography and Network Security": Principlesand Practice. Prentice Hall, 2nd edition, 1999.