The Secured Key Issuing for Message Transmission in Group Network Using Elliptic Curve Cryptography

Lokesh. A, Prapulla.C, Aruna M G, Anusha, Divya P

* Department, Institute Name
** Department, Institute Name, if any

Abstract- The main constraint in message transmission is security. The key must be shared among users so that we have a secure transmission in group based message transmission. In this paper we discuss an important security problem that happens in mobile ad hoc network for key agreement in dynamic group. For a communication to be secure, a group key must be shared by all the members of the group. The key must be updated when any member of the group leaves existing group or become a new member of the existing group. In this paper, we establish a secured transmission. The main idea here is to spilt the existing network into smaller sub-network and each sub-network maintains its own keys. These keys should be updated when a existing node leave the network or when an new node joins the existing system. The nodes that have the high stability in the sub-network will be made as the Inner gateway member and among Inner gateway member the one that has high stability will be made as Outer gateway member.

Index Terms- adhoc network, secured group transmission, elliptic curve cryptography, KASP.

I. INTRODUCTION

A mobile ad-hoc network is a self-configuring infrastructureless network of mobile devices connected by wireless each device in MANET is free to move independently in any direction, and will therefore change its link to other devices frequently. Each must forward traffic unrelated to its own use, and therefore be a router. A wireless ad hoc network is decentralized type of wireless network. The network is ad hoc because it does not rely on pre-existing infrastructure such as routers. An ad hoc network is made up of multiple nodes connected by links.

The primary challenge in building a MANET is equipping each device to continuously maintain the information required to properly route traffic. Different protocols are then evaluated based on measure such as packets drop rate, the overhead introduced by the routing protocol, end-to-end packets delay etc. An efficient group key agreement protocol has been introduced he nodes as selected based on stability and power. Here the main idea is to split a large network into several subnetwork, each maintaining its subgroup keys to manage the subgroup and managing many subgroups using Elliptic Curve Diffie-Hellman (ECDH) key agreement algorithm. In this paper, we develop two protocols namely, Subgroup Key Generation(SKG) for subnetwork and Group KeyGeneration(GKG) for a outer network based on ECDH. When the membership changes these subgroup keys n the groupkeys should be changed. The messages and the key updates will b limited within sub network and outer networkdue to the introduction of group based approach. This computed load is being distributed to many mobile adhoc network nodes. In terms of efficiency and security it been proved both practically and theoretically that new protocol KASP performs well and efficient for key establishment problem.

In ad hoc network that is large and highly mobile cannot use a single key for entire network because the rekeying process will become costly. The Inner gateway member that is been selected will maintain a Inner key and the Outer gateway member maintains a Outer key. When a group member leaves the existing network or joins a existing network the group key agreement protocol needs to represents the security issue due to mobility of nodes.

II. EXISTING SYSTEM

Security is the main constraint in message transmission using mobile ad hoc network.

Limitations of existing system:

- Ad-hoc networks are not generally having a trusted third party.
- Keys are generated for singal user which is not secured in case of group communication.

III. PROPOSED SYSTEM

In this paper an important security issue has been authorities an mobile adhoc network for establishing dynamic group key agreement. The group members must share the group key among themselves for secure communication.

Advantages of proposed system:

- Keys are generated for the group user.
- Communication of messages takes place through a trusted third party in mobile adhoc network called gateway member.

IV. DESIGN

4.1 Design Constraints

The project aims to develop a system which allows group communication in network, by splitting the group into subgroups and generating keys for group and subgroup using ECDH. These keys will be applied for the message which needs to be transmitted over the network. The algorithms being implemented

have a range of efficiency and complexity. The code is intended to be written in Java language, supporting JCreator and JDK environment.

4.2.Interfaces

Interface refers to a point of interaction between components and is applicable at the level of hardware and software.

4.2.1User Interface

The users interact with the machine using the user interface in the system.The hardware and software forms the components of the user interface . various system consists of this user interface and provide a means of input that allow manipulation of the system by the users, and the effects of users manipulation can be indicated by the output.

The two proposed algorithms i.e. registration protocol and signature generation protocol to achieve high performance and scalability with respect to the security being provided. Application can be accessed from the cloud storage service over all nodes, and then exchanges the data via a network.

4.3 High Level Design

Software Development is generally a stepwise process. Before the process of implementing the software at hand it involves the process of software design. A software design is a description of the structure of the software to be implemented, the data which is part of the system, the interfaces between the components, sometimes, the algorithms used. Designers do not arrive at a finished design immediately but develop the design iteratively through a number of different versions. The design process involves adding formality and detail as the design is developed, with constant backtracking to correct earlier designs.

In many software development projects, software design is an ad hoc process. Starting from the set of requirements, usually in natural language, an informal design is prepared. Coding commences and the design stage is modified as the system is implemented. When the implementation stage is complete, the design has usually changed so much from the initial specification that the original design document becomes an incorrect and incomplete description of the system. There are several advantages of the design phase. Some of them are listed below:

- The design phase helps to understand the user requirements and helps to map the user requirements into implementation phase.
- The iterations in the design phase helps in incorporating as many user requirements as possible in the final software being developed.
- The design phase reduces the cost involved in the development of the software as many changes would be made to the software in the implementation if the design is not clear.

The design process is iterative and requires consideration of various design alternatives at every stage. The objective of the design stage is to produce the overall design of the software.

The design stage involves two sub-stages namely:

- High-Level Design
- Detailed-Level Design

In the proposed application of high-level design the technical architecture will study the functional and non functional design of the overall solution architecture, which can handle those needs. High Level Design means precisely that. A discussion of high level design gives us an over view of how the top level components and how a few things should work which will comprised to give the proposed system.

A complete implementation details must not be given and at sometimes very little information on the implementation is enough. At some point we should not even give details of database like relation and object sometimes providing details of platforms and programming language gives a detail idea of implementation. In this chapter we give an overview of the design of the system and how it is organized and the flow of data through the system. By reading this document the user should have an overall understanding of the problem and its solution. We have also discussed about the problems encountered during the design of the system and justified the use of the design. The Data Flow Diagrams (DFD), given in the later sections of the chapter, shows the flow of data through the system.

4.4 Design Considerations

The design process is iterative and requires consideration of various design alternatives at every stage. The design process is constrained by the assumptions made prior to the development of the system. It involves deciding on the type of approach used for the development of each portion of the system selecting the same rationale. Thus this section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution.

4.4.1 Assumptions and Dependencies

Several assumptions regarding the hardware required and the working environment of the system influence design decisions. The assumptions have been made after considerable consultation with the end user and are more or less reasonable.

- The system will be implemented on the Windows operating system, using Java. The system should be modified so as to enable its use on the Linux operating system as well.
- The system runs on the Windows Operating System which is Windows XP or above.

4.4.2General Constraints

There might be some global limitations or constraints that have a significant impact on the design of the system's software or an associated impact. Such constraints may be imposed on the following issues related to our project which are as follows:

- There exists a maximum limit on the number of nodes that can be deployed in the network so as to prevent degradation in performance of the proposed algorithm while it is being executed on the system.
- As the number of nodes increases, the complexity of showing different cases for the algorithm becomes difficult. Therefore, we limit ourselves to a fewer number of nodes for demonstration purpose.

V. ARCHITECTURAL STRATEGIES

The system is developed using the Top-Down approach. In this method, the system divides the files based on the number of nodes. Each node downloads the files and then requests the missing files from the neighboring nodes.

- The 'JAVA' programming language has been used for development of the application.
- The WINDOWS XP or later versions of windows XP operating system has been used as the platform for development.
- The processes communicate through Sockets.

5.1 System Architecture

This section provides a high-level overview of how the functionality and the responsibilities of the system were partitioned and then assigned to subsystems or the components or the modules appropriately. The main purpose here is to gain a general understanding of how and why the system was decomposed, and how the individual parts work together to provide the desired functionality. The system architecture is as shown in figure 1.

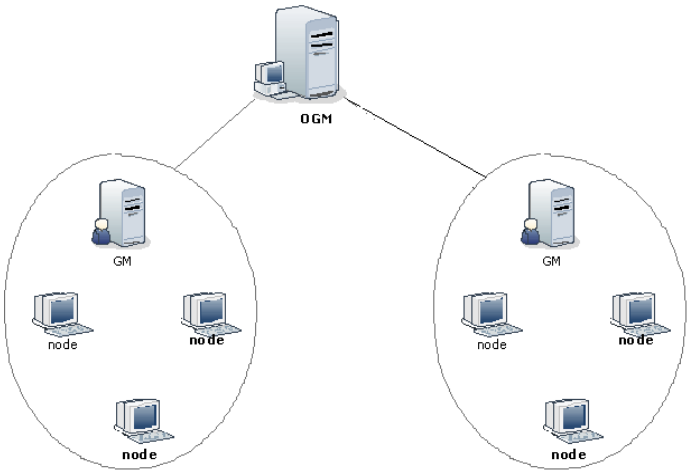


Figure 1: System Architecture

Diffie Hellman Algorithm:

Diffie–Hellman key exchange (D–H)[nb 1] is a specific method of exchanging cryptographic keys. It is one of the earliest practical examples of key exchange implemented within the field of cryptography. The Diffie–Hellman key exchange method allows two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure communications channel. This key can then be used to encrypt subsequent communications using a symmetric key cipher.

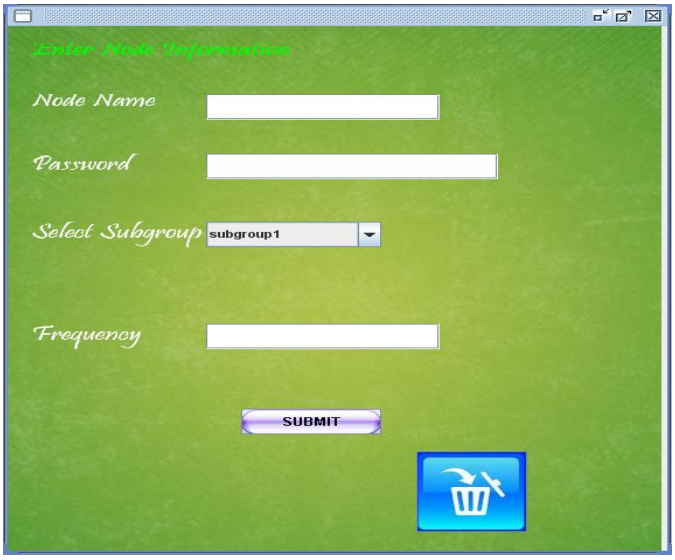
Elliptic Curve Cryptography using Diffie Hellman Algorithm

Elliptic curve Diffie–Hellman (ECDH) is an anonymous key agreement protocol that allows two parties, each having an elliptic curve public-private key pair, to establish a shared secret over an insecure channel.[1][2][3] This shared secret may be directly used as a key, or better yet, to derive another key which can then be used to encrypt subsequent communications using a symmetric key cipher. It is a variant of the Diffie–Hellman protocol using elliptic curve cryptography.

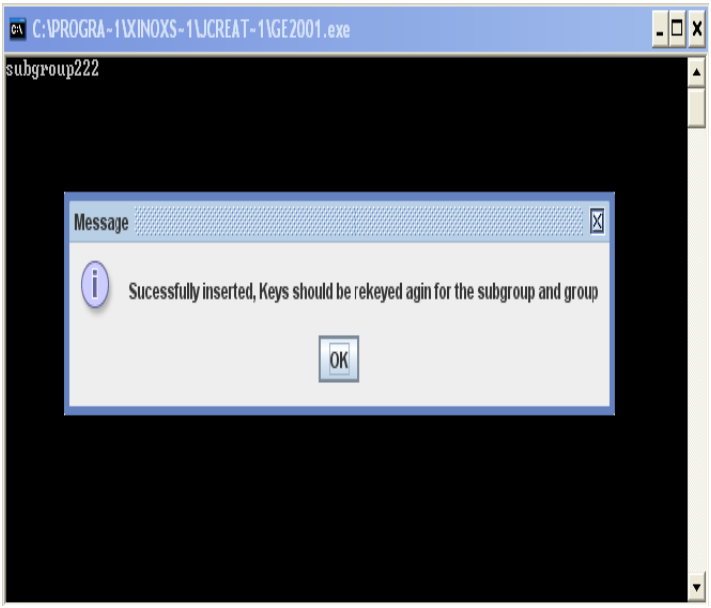
VI. RESULTS

Setup mechanism

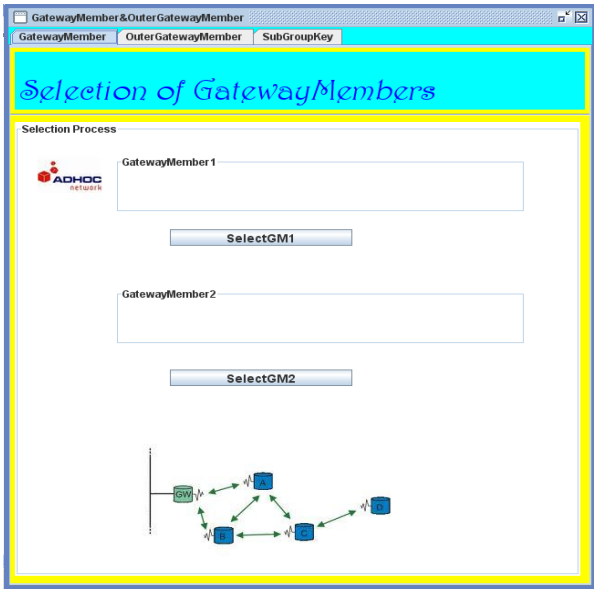
2.1 Open Jcreator and run java programs



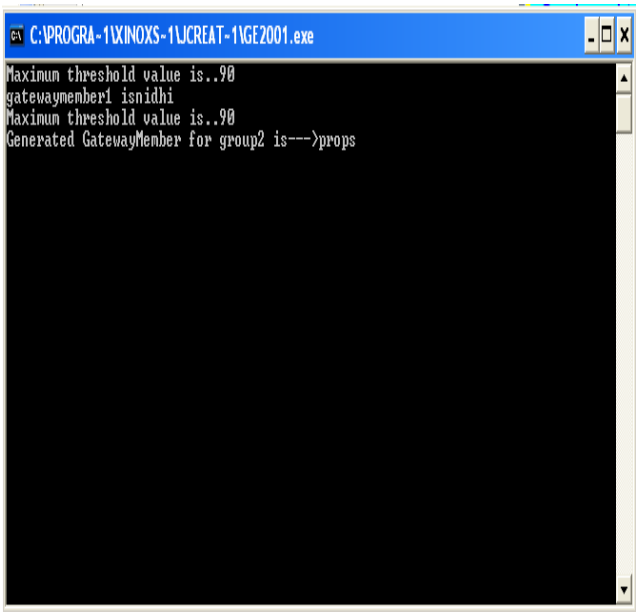
2.2 Enter the node details to add or delete a node



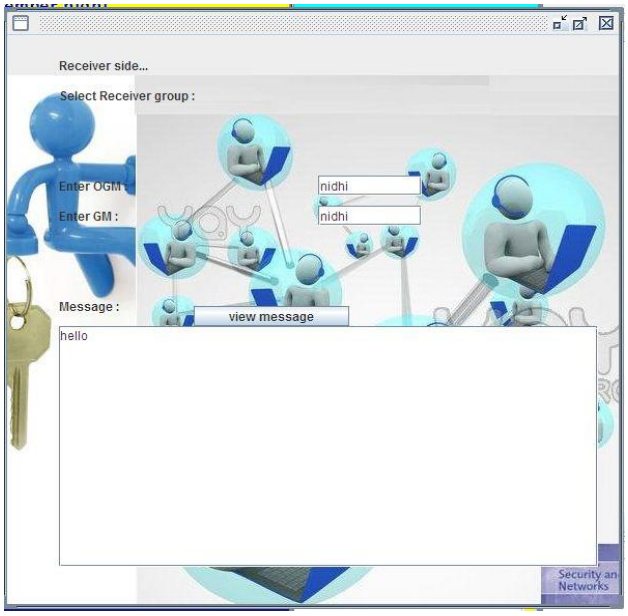
2.3After successfully inserting the node a message appears



2.4 Select the GatewayMembers



2.5 Gateway Members for two groups are selected



2.6 Decrypted message is received by the reciever

VII. CONCLUSION

Mechanisms that enable individual peers of unstructured P2P content sharing networks to register longstanding queries and receive notification when new matching items appear can significantly improve their utility and effectiveness. While the pub-sub paradigm can provide this capability, implementing pub-sub systems on unstructured overlays is often a very complex endeavor. The continuous query paradigm studied in this paper is similar to pub-sub, but it provides best effort notification service. We presented the design and evaluation of a lightweight system, called CoQUOS, which supports continuous queries in unstructured P2P networks.

The CoQUOS system incorporates severalnovel features such as cluster resilient random walk for query propagation and dynamic probability scheme for query registration, and a lazy replication technique for countering network churn.

VIII. FUTURE ENHANCEMENT

7.1 3GPP:

- The 3rd Generation Partnership Project(3GPP) is developing an architechture for next generation mobile communication system ie. System Architecture Evolution.
- SAE/LTE architecture, Extensible Authentication Protocol-Authentication and Key Agreement (EAP-AKA) is used to provide secure 3G-WLAN interworking.
- Disclosure of user identity, man-in-the-middle attack, Sequence Number (SQN) synchronization, and additional bandwidth consumption. Are the vulnerabilities in EAP-AKA protocol
- The analyzes threats and attacks in 3G and a new authentication and key agreement protocol is been proposed based on EAP-AKA.

- To overcome the vulnerabilities present in the EAP-AKA a combined protocol has been proposed with symmetric key cryptosystem and elliptic curve Diffie Hellman.

7.2 FGPA:

- FGPA involves the hardware implementation of elliptic curve cryptography (ECC).
- Experimental results demonstrate that the FPGA implementation can speedup the point multiplication by 31.6 times compared to a software based implementation.
- The main contribution of FPGA based design is the resources sharing and parallel processing optimization.

REFERENCES

[1] Gnutella P2P Network. www.gnutella.com.

[2] Kazaa P2P Network. www.kazaa.com.

[3] TIB/Rendezvous. White paper, 1999.

[4] S. Androutsellis-Theotokis and D. Spinellis. A Survey of Peerto-Peer Content Distribution Technologies. ACM Comput. Surv., 2004.

[5] B. Arai, G. Das, D. Gunopulos, and V. Kalogeraki. Approximating Aggregation Queries in Peer-to-Peer Networks. In Proceedings of the 22nd International Conference on Data Engineering (ICDE), 2006.

[6] R.Baldoni, C.Marchetti, A.Virgillito, and R.Vitenberg. Contentbased Publish-Subscribe over Structured Overlay Networks. In Proceedings of ICDCS, 2005.

[7] G.Banavar, T.Chandra, B.Mukherjee, J.Nagarajarao, R.E.Strom, and D.C.Sturman. An Efficient Multicast Protocol for Content- Based Publish-Subscribe Systems. In Proceedings of ICDCS 1999, 1999.

[8] T. Bu and D. F. Towsley. On Distinguishing between Internet Power Law Topology Generators. In INFOCOM, 2002.

[9] A. Carzaniga, D. S. Rosenblum, and A. L. Wolf. Design and evaluation of a wide-area event noti_cation service. ACM Transactions on Computer Systems, 19(3):332.383, 2001.

[10] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker. Making Gnutella-like P2P Systems Scalable. In Proceedings of ACM SIGCOMM 2003, 2003.

[11] J. Chen, L. Ramaswamy, and A. Meka. Message Diffusion in Unstructured Overlay Networks. In Proceedings of NCA, 2007.

[12] P. Chirita, S. Idreos, M. Koubarakis, and W. Nejdl. Publish/ Subscribe for RDF-based P2P Networks. In Proceedings of the 1st European Semantic Web Symposium, May 2004.

AUTHORS

First Author – Lokesh A is with Dept. of CS&E Asst.Prof, M S Engineering College, Bengaluru-560064,Karnataka. (email: lokeshyadav.ka@gmail.com)

Second Author – Prapulla C is with Dept. of CS&E,M.Tech II Sem M S Engineering College, Bengaluru-560064,Karnataka. (email: prapulla1990@gmail.com)

Third Author – Anusha M S is with Dept. of CS&E, Asst.Prof,MSEC, Karnataka. (email: anushams.ms@gmail.com)

Fourth Author – Divya P is with Dept. of CS&E, Asst.Prof,MSEC, Karnataka.(email:divya.sda@gmail.com)

Fifth Author – Aruna M G Asst.Professor, M S Engineering College, Bengaluru-560064,Karnataka (email:aruna_mg@yahoo.com)