DYNAMIC AND FLEXIBLE ROBUST GROUP KEY AGREEMENT USING MOBILE ENVIRONMENT FOR SECURED COMMUNICATION

K. Banupriya[1] ,M.Lavanya[2]
[1,2] Assistant Professor, Department of Computer Science and Engineering,
Dr.APJ Abdul Kalam Centre for Research,
Adhi College of Engineering and Technology, Sankarapuram - 631 605, Tamilnadu, India.
Email: banupriya.cse@adhi.edu.in

Abstract :

A tough Group Key Agreement Protocol (GKAP) allows a set of players to establish a shared secret key, anyway of network or node failures. As our presented systems stated that broadcast encryption (BE) is necessary for secure data outsourcing over a group and Group Key Agreement Protocol (GKAP).Create a confidential channel among group members. Current constant-round GKAP protocols are either efficient or robust; assuming a reliable broadcast communication medium, the standard encryption-based Group Key Agreement protocol can be robust against arbitrary number of node faults, but the size of the messages transmit by every participant is proportional to the number of participant. In contrast, not a robust Group Key Agreement can be achieved with each player broadcasting just constant-sized messages and keys. But due to lack of key management and group member revocation is a still challenging issue. A novel 2-round group key agreement protocol is proposed. This tolerates up to T node failures which uses O (T) sized messages for various 'T'. The new protocol with logarithmic – sized messages and also reduced round complexity close to about 2 is proposed, that implies a fully-robust group key agreement. This also assumes random node faults. This protocol can be used to withstand malicious insiders at small constant factor which raises in bandwidth and computation. The proposed protocol is secure under the (standard) Decisional Square Diffie - Hellman assumption.

Key words - Group key agreement, fault-tolerance, algorithms, and security.

I. INTRODUCTION

The growth of cluster applications triggers the need for group-oriented security mechanisms over lacking confidence network channels. The applications include IP telephony, collaborative workspaces, secure conferences, as well as energetic coalitions common in law enforcement and disaster release scenarios. Standard security services required in such collection settings, e.g., confidentiality of group wide broadcasts can be very efficiently achieved if all cluster members share a group-wide secret key. The early design of contributory group key agreement protocol (GKAP) focuses on the efficiency of initial GKAP. Efficiency metrics include computation and round complexities. Although each metric is important in practice, the round complication can be more essential particularly in the distributed computing environment.

Several well known efficient two-round GKAP protocols are proposed in [4]. However, their performance degrades if faults occur during the protocol execution. Faults cause the normal protocol (without robustness) to be restarted from the scratch. To improve performance, current GKAP must be made robust. In this context, robustness refers to the ability to complete the protocol, despite player and/or communication faults. Robust GKAP is a serious concern in practice. Mobile nodes that communicate over a wireless medium can loose connectivity. Router failures, causing network partitioning (due to a mis configuration or congestion) as well as malicious attacks, also raises the failure probability. We list some motivating examples:

- Consider an emergent situation where some secure meeting for liberate missions and military negotiations must be held past to a special time. In that case, robust GKAP is prerequisite to minimize damage.
- Group communiqué (such as direct messaging and video- and audio-conferencing) operates on a real-time setting. Thus, robust GKAP is crucial to improve the overall Quality of Service.
- Security policies usually dictate that group keys must be refreshed periodically.[1] Thus, a GKA protocol needs to be re-run (perhaps often), and improving GKAP performance is essential.