A HIERARCHICAL GROUP KEY MANAGEMENT WITH HOST MOBILITY PROTOCOL IN WIRELESS MOBILE ENVIRONMENT

BABAK DAGHIGHI

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2016

A HIERARCHICAL GROUP KEY MANAGEMENT WITH HOST MOBILITY PROTOCOL IN WIRELESS MOBILE ENVIRONMENT

BABAK DAGHIGHI

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
KUALA LUMPUR

2016

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: BABAK DAGHIGHI (I.C/Passport No: …….)

Registration/Matric No: ……

Name of Degree: Doctor of Philosophy

Title of Project Paper/Research Report/Dissertation/Thesis ("this Work"):

A HIERARCHICAL GROUP KEY MANAGEMENT WITH HOST MOBILITY PROTOCOL IN WIRELESS MOBILE ENVIRONMENT.

Field of Study: Computer Science (Network security)

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University of Malaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation:

ABSTRACT

Group communication has been increasingly used as an efficient communication means for facilitating emerging applications that require packet delivery from one or many sources to multiple recipients. Due to insecure communication channels, group key management schemes have recently received special attention as a fundamental building block to preserve secrecy of group communication. Developing group key management in highly dynamic environments particularly in wireless mobile networks due to their inherent complexities faces additional challenges. On one hand, the constraint of wireless devices in terms of resources scarcity, and on the other, the mobility of group members complicates design of a group key management scheme.

While a multitude effort has been given to establish a secure group communication in wired network environments, few attempts have been made to extend the group key management scheme to wireless mobile environments to explicitly address the member mobility issue. The aim of this research is to propose a group key management scheme that addresses the mobility issues.

This work involves four main parts. First, an introduction is given to group communication to recognize its capabilities as an efficient type of communication. Then the research activities pertaining to group key management schemes are explored to distinguish various design approaches along with the advantages and disadvantages related to each approach particularly when they are developed for wireless mobile environments.

Second, the investigation is continued by scrutinizing the existing solutions that take consideration host mobility issue, and identifying the remaining weaknesses pertaining to each scheme. Then, the focus goes toward the primary constraints and challenges

involved in adapting and developing a group key management scheme for wireless mobile environments.

Third, a group key management scheme is proposed and designed for establishing secure group communication suitable for infrastructure-based wireless mobile environments. This scheme identifies an overall architecture, which includes the main components and their roles, trust and keying relationships, as well as detailed functional requirements. The main protocols required within the scheme are then described in detail.

Finally, a simulation analysis is conducted to assess HIMOB with regard to the security requirement, and the performance requirements. The impact of group size variation and mobility rate variation are studied on the average of rekeying messages generated per each event and *1-affects-n* phenomenon. The results obtained from the simulation experiments show HIMOB surpasses the other existing solutions with minimizing the number of rekeying messages sent and the number members affected on each event. The security requirements studies also show the backward and forward secrecy is preserved in HIMOB even though the members move between areas. The research work is concluded by outlining the future research direction.

ABSTRAK

komunikasi Kumpulan telah semakin digunakan sebagai komunikasi yang cekap bermakna untuk memudahkan aplikasi baru muncul yang memerlukan penghantaran paket dari satu atau banyak sumber kepada berbilang penerima. Oleh kerana saluran komunikasi yang tidak selamat, kumpulan skim pengurusan utama baru-baru ini mendapat perhatian khas sebagai blok bangunan asas untuk memelihara kerahsiaan komunikasi kumpulan. Membangunkan kumpulan pengurusan utama dalam persekitaran yang sangat dinamik khususnya dalam rangkaian mudah alih tanpa wayar kerana kerumitan yang wujud mereka menghadapi cabaran tambahan. Dalam satu tangan, kekangan peranti wayarles dari segi sumber kekurangan, dan di pihak yang lain, pergerakan ahli-ahli kumpulan merumitkan reka bentuk skim kumpulan pengurusan utama. Walaupun usaha pelbagai telah diberikan kepada mewujudkan komunikasi kumpulan yang selamat dalam persekitaran rangkaian berwayar, beberapa percubaan telah dibuat untuk memperluaskan skim pengurusan utama kumpulan dengan persekitaran mudah alih tanpa wayar secara jelas menangani isu mobiliti anggota. Tujuan kajian ini adalah untuk mencadangkan satu skim pengurusan utama kumpulan yang menangani isu-isu mobiliti. Kerja-kerja ini melibatkan empat bahagian utama. Pertama, pengenalan diberikan kepada komunikasi kumpulan mengiktiraf keupayaannya sebagai jenis yang cekap komunikasi. Kemudian aktiviti penyelidikan yang berkaitan dengan skim pengurusan utama kumpulan yang diterokai untuk membezakan pelbagai reka bentuk pendekatan bersama-sama dengan kebaikan dan keburukan yang berkaitan dengan setiap pendekatan terutamanya apabila mereka dibangunkan untuk persekitaran mudah alih tanpa wayar. Kedua, siasatan itu diteruskan dengan meneliti penyelesaian yang sedia ada yang mengambil kira isu mobiliti tuan rumah, dan mengenal pasti kelemahan yang tinggal berkaitan dengan setiap skim. Kemudian, tumpuan pergi ke arah kekangan utama dan cabaran yang terlibat dalam menyesuaikan diri dan membangunkan satu skim

pengurusan utama Kumpulan untuk persekitaran mudah alih tanpa wayar. Ketiga, skim pengurusan utama Kumpulan adalah dicadangkan dan direka untuk mewujudkan komunikasi kumpulan selamat sesuai untuk persekitaran mudah alih tanpa wayar berasaskan infrastruktur. Skim ini mengenal pasti seni bina pada keseluruhannya, termasuk komponen utama dan peranan mereka, kepercayaan dan hubungan menaip, serta keperluan fungsian terperinci. Protokol utama yang diperlukan dalam skim ini kemudian diterangkan secara terperinci. Akhir sekali, analisis simulasi dijalankan untuk menilai HIMOB dengan mengambil kira keperluan keselamatan dan keperluan prestasi. Kesan perubahan saiz kumpulan dan variasi kadar mobiliti dikaji pada purata mesej rekeying dijana bagi setiap peristiwa dan fenomena *1-affects-n*. Keputusan yang diperolehi dari eksperimen simulasi menunjukkan HIMOB melampaui penyelesaian yang lain yang sedia ada dengan mengurangkan bilangan mesej rekeying dihantar dan ahli bilangan terjejas pada setiap acara. Keperluan keselamatan kajian juga menunjukkan kerahsiaan ke belakang dan ke hadapan dipelihara dalam HIMOB walaupun ahli-ahli bergerak antara kawasan. Kerja-kerja penyelidikan selesai dengan menggariskan arah penyelidikan masa depan.

ACKNOWLEDGEMENTS

Completing a thesis takes perseverance, time, and patience and it would not have been possible without the help of a number of individuals. First and foremost, I would like to record my genuine gratitude to my supervisor, Professor Dr. Miss Laiha Mat Kiah for her excellent supervision, advice, guidance, and support from the early stage of this research. I was not sure I could handle an undertaking of such magnitude, but was able to, thanks to her consistent efforts and true desire to keep me on track. I would also like to thank Dr. Said Gharout of the University of Paris for his constructive comments.

My deepest and most heart-felt thanks go to my most beloved family and adored family in-law, for their endless love and support. They have instilled in me a love for knowledge and a strong work ethic which has enabled me to accomplish anything I set my mind to. They have always supported me in whatever I do and wherever it is possible.

To those who were directly and indirectly involved in this research, particularly, Salman Iqbal, Vahid Maleki Raee, Muhamad Habib Ur Rahman, Ibrahim Targio, Hamid Tahaei, Ahmad Firdaus Zainal Abidin, Mazrullhisham Yusuf Mohd Zain, Rita Afriani Mohd Yusu, and Nor Shuhadah Yahiya, your decorousness, continuous tangible, moral support, and willing to help me shall not be forgotten; please accept my utmost appreciation to all of you.

Above all, I would like to thank God for his grace in granting me to complete my studies.

To my wife Sara,

and my son Arash,

for their unconditional loves, devotions, and supports.

Their sacrificial cares and encouragements made possible for me to complete this work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AP	:	Access Point
BS	:	Base Stations
CDF	:	Cumulative Distribution Function
CDMA	:	Code division multiple access
DH	:	Diffie-Hellman
GC	:	Group Controller
GCKS	:	Group Controller and Key Server
GSM	:	Global System for Mobile communication
IETF	:	Internet Engineering Task Force
KD	:	Key Distributor
KDF	:	key derivation function
KEK	:	Key Encryption Keys
LKH	:	Logical Key Hierarchy
MANET	:	Mobile Ad-hoc Networks
MBone	:	Multicast Backbone
PRF	:	Pseudo Random Function
SA	:	Security Association
SH	:	Supervisor Hosts
SMuG	:	Secure Multicast Research Group
UMTS	:	Universal Mobile Telecommunication System
WLAN	:	Wireless Local Area Network
WPAN	:	Wireless Personal Area Networks
WSN	:	Wireless Sensor network

CHAPTER 1: INTRODUCTION

There has been a rapid proliferation of wireless communication and portable computing devices due to substantial technological improvement in terms of communication infrastructure, performance, and computing power. In addition, there has been concurrently the phenomenal advances in Internet technology (Sathiaseelan & Crowcroft, 2012) during the last few years. It is also forecasted the global mobile data traffic will have tremendous growth by 2020 (Cisco Visual Networking Index, 2016a), which may provide inspiration and motivation for the development of new group based applications and services such as multimedia conferencing, interactive group games, video on demand, IP-TV, and broadcasting stock quotes, and social group networks (Chang, Chen, & Zhou, 2009; Holzer & Ondrus, 2011; Y. Shin, Choi, Koo, & Choi, 2013). Group based applications provide an efficient communication by delivering a single copy of data to the network elements such as routers and switches making copy as necessary for the receivers, which result in better utilization of network resources such as bandwidth and buffer space.

Ensuring the security of group based applications is no trivial matter since most of group based applications take place over insecure network (Judge & Ammar, 2003; Sakarindr & Ansari, 2007) and moreover, members can openly and anonymously join the group (Martin & Haberman, 2008). Depending on the application need, basic security services such as confidentiality, data integrity and entity authentication need to be in place to ensure backward and forward secrecy, as well as the integrity of group members and group operations. These services, particularly the backward and forward secrecy can be established by sharing a common key (known as the traffic encryption key, denoted by TEK), which is then used to encrypt all traffic of a specific group. As a result, only members of the group can decrypt the received messages. According to the Internet

engineering Task Force (IETF) (MSEC, 2011), managing a group key is one of the fundamental challenges in designing a secure and reliable group communication.

In order to achieve data confidentiality in a secure group communication, only members of the group must be able to read data in spite of the data may be broadcasted into the whole networks (Baugher, Canetti, Dondeti, & Lindholm, 2005; Kim, Perrig, & Tsudik, 2004a, 2004b). A group key management scheme is a fundamental building block for provision of secure group communication. Its role is to generate, update and distribute keying materials to the legitimate group members in order to maintain backward secrecy (preventing newly joining member from having access to previous information) and forward secrecy (Preventing leaving member from having access to next information) (Kim et al., 2004a, 2004b). In the last few years, several key management schemes have been proposed in order to address the secrecy issue in group communication (Yacine Challal & Seba, 2005; Judge & Ammar, 2003; Mapoka, 2013; Rafaeli & Hutchison, 2003; Sakarindr & Ansari, 2007, 2010). The main objective of these solutions is to provide protocols which address some challenging issues in group key management in terms of scalability, efficiency, and performance. These existing group key management schemes can be organized into three main classes; centralized approach, decentralized approach, and distributed (or known as contributory) approach.

In a centralized approach, a single entity (i.e. a group controller, GC) is responsible to generate keys, and securely distribute them to all other group members. In other words, it is a main reference for security information for all group members. Logical Key Hierarchy (LKH) is one of the famous schemes in this category that was proposed by several research groups nearly at the same time (Chung Kei, Gouda, & Lam, 2000), (Wallner, Harder, & Agee, 1999). Several group key management protocols based on centralized approach have been presented to improve performance and security (Baugher

et al., 2005; Je, Lee, Park, & Seo, 2010; Lin, Huang, Lai, & Lee, 2009; Ng, Howarth, Sun, & Cruickshank, 2007; Yan & Liu, 2007). Although this approach has the advantage of simplicity, and scale group key management to the group with large size, dependencies on a single key server leave a single point of failure.

In distributed approach, group key management has no explicit key distributor center and all members contribute to manage the traffic key. This scheme uniformly distributes the workload for key management to all the group members and eliminates the need for central entity. The advantage of this approach is to alleviate the problem of a single point of failure and trust found in the centralized approach. Some distributed group key management schemes have been presented in (Amir, Nita-Rotaru, Stanton, & Tsudik, 2005; Kim et al., 2004a, 2004b; Lv, Li, & Wang, 2012; Magliveras, Wandi, & Xukai, 2008; Mortazavi, Pour, & Kato, 2011; Michael Steiner, Tsudik, & Waidner, 1996; Zheng, Manz, & Alves-Foss, 2007). Nevertheless, the processing time (in terms of sequential exponentiations, message signatures, and verification) and communication requirement (in terms of messages sent) increase by growing the number of members in the group.

In the decentralized approach, a large group is split into some small subgroups so that they make some hierarchical levels. In each level one or more entities are responsible to manage the other entities in its level. These entities are dependent to their up level entity so that form a hierarchal key management while governing independently group members in their jurisdictions. The group key management protocols based on this class have been well documented in the following work (Yacine Challal, Bettahar, & Bouabdallah, 2004; Y. Challal, Gharout, Bouabdallah, & Bettahar, 2008; Cho, Chen, & Wang, 2008; Thomas Hardjono, Cain, & Monga, 2000; Hur & Yoon, 2009; J. H. Li, Bhattacharjee, Yu, & Levy, 2008; Mat Kiah & Martin, 2007; Nemaney Pour, Kumekawa, Kato, & Itoh, 2007). This approach is observed to provide a trade-off between both centralized and distributed

approaches. Furthermore, it leads to establishing a scalable and secure group communication.

While numerous efforts were made to establish a secure group communication in wired network environments, several attempts have made to extend the group key management scheme to wireless mobile environments, where group members can easily change their point of attachment to the network while still remaining in the group communication. Providing a mechanism to tackle the member mobility issue is critical if secure group communication is to be deployed in wireless mobile environments.

1.1 Problem statement

The challenging problem in designing a group key management scheme is to minimize the rekeying process performed each time there is any group membership changes in order to achieve efficiency and scalability (Chung Kei et al., 2000; Mittra, 1997). The impact of this rekeying process can be critical particularly in a group with high membership dynamics (where frequent join and leave events occur) since network bandwidth and buffer space is considerably consumed with the distributed rekeying messages. Logical key hierarchy (LKH) scheme efficiently reduce the rekeying messages overhead (Chung Kei et al., 2000). To limit the impact of rekeying process on the group members (referred to as *1-affects-n* phenomenon) as one of the scalability requirements, some proposed schemes organized the entire group members into several hierarchical subgroups or areas, which cause the rekeying process to be restricted only to a subgroup in which an event occurs (Yacine Challal et al., 2004; Saïd Gharout, Challal, & Bouabdallah, 2008; Heba K, 2004; Mat Kiah & Martin, 2007). Nevertheless, the problem becomes more difficult and complex when group key management schemes are developed for wireless mobile environments (Daghighi, Mat Kiah, Shamshirband, & Rehman, 2015).

The proliferation of mobile computing and communication devices ranging from cell phone, laptops, handheld digital devices, personal digital assistants, or wearable computers have stimulated the explosive growth of the wireless and cellular networks market (Gartner, 2015). While the number of global mobile devices and connections reached to 7.9 billion by year 2015, it is forecasted such devices and connections will grow to 11.6 billion by 2020 (Cisco Visual Networking Index, 2016a). Besides that, by 2020, wired devices will account for 22 percent of Internet traffic growth, whereas Wi-Fi and mobile devices will account for 78 percent of Internet traffic as shown in Figure 1.1 (Cisco Visual Networking Index, 2016b).

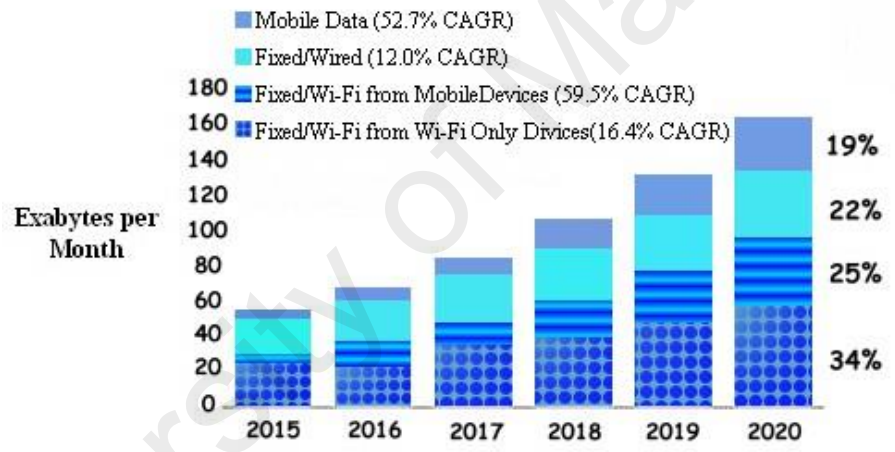

Figure 1.1: Growth of Internet traffic in relation with wired, Wi-Fi, and mobile devices.

Although infrastructure based networks offer a more reliable way for mobile devices to access network services, the deployment of such infrastructure takes time or potentially needs high cost. To avoid unnecessary delay and cost for dynamic environments where people need to be temporarily interconnected in areas without pre-existing communication infrastructure (e.g., battlefield, extensive disaster recovery operations), or to convince the users demands for mobile, or ubiquitous access to services regardless where they are, infrastructure-less networks are considered as a suitable solution to

provide the functionalities usually provided by the infrastructure-based networks. Various kinds of infrastructure-less networks from the 802.11 networks in home, ad hoc networks, mobile ad hoc networks to CDMA (Code division multiple access) and GSM (Global System for Mobile communication) networks in cell phone make wireless as a vast and complex topic.

Examining literature with an eye toward extending group communication into the wireless mobile environment imposed more complexity in designing group key management schemes. The wireless constraints can be classified into two categories: 1) resource scarcity of wireless devices, and 2) characteristics of wireless networks. The primary constraints of wireless devices are typically less computing power, low storage capacity, and limited battery power compared to desktop computers (Bouassida, Chrisment, & Festor, 2008; Goldsmith & Wicker, 2002; A. K. Gupta, 2008). Such restrictions prevent wireless devices from performing complicated security computation of cryptography algorithms like public keys. Hence, using time consuming and complicated computational algorithmic techniques in designing group key management causes wireless networks to avoid adopting such protocols to prevent draining their resources.

On the other hand, the characteristics of wireless networks, including the mobility of members, narrow bandwidth, and the transmission error rate imposes more burden in designing the group key management. Wireless networks are generally implemented using radio communication that omits the needs of wire for connection. Therefore, wireless devices are able to move from one area of networks to another. Member mobility (Romdhani, Kellil, Hong-Yon, Bouabdallah, & Bettahar, 2004; Schmidt, Waehlisch, & Fairhurst, 2010) as a unique property of wireless networks poses a new challenge for securing group communication; how to deliver the keys to mobile members while moving

from one area to another while still remaining in the session. Indeed, member mobility complicates the group key management in the mobile environments (Koodli, 2009) where the member can move inside the group since the dynamic member location (mobility) must be managed along with the dynamic group membership (join and leave).

In mobile environments, when a member changes his position from one area to another, its access to the data traffic is indeed transferred to the new access point. Therefore, the complexity of key management increases in terms of communication and computation cost as the member is not known in the new area even though a fast hand off mechanism exists. In other words, the moving member is treated as a leaving member from the group in the departing area and subsequently as a new member joining in that group in the new area in a naïve solution. In this case, the keying materials must be updated in both old and new areas. Since the wireless networks may not be fully connected because of signal interference, obstruction, and limited bandwidth, huge amount of updating messages may interrupt the group communication.

To the best of our knowledge, few attempts have been carried out to address the mobility issue in wireless mobile environments. Nevertheless, they suffer from lack of an explicit protocol for member mobility issue, security flaws in terms of breaching forward secrecy and backward secrecy, many signaling messages, and high cost of communication overhead. Bruschi et al. presented the earliest research which identified the critical factors in implementing a secure group communication in mobile environments, but it did not address the mobility issue in secure group communication explicitly (Bruschi & Rosti, 2002). The Immediate Rekey (IR) protocol (Zhang, DeCleene, Kurose, & Towsley, 2002) and WSMM protocol requires to repeatedly update the keying materials if a member rapidly visits different areas which impose a cost of communication overhead in the both old area and new area. Meanwhile FEDRP (Zhang

et al., 2002) showed the high rekey rate in situations with high mobility rate. KMGM (Said Gharout, 2010, 2012) and M-Iolus (Kamat, Parimi, & Agrawal, 2003) exhibited minimum rekeying cost on member mobility, however, they suffered from a backward secrecy violation when a moving member enters into a new area. The KTMM has imposed the expense of lack of backward secrecy in the visited area since the moving member likely gain security services information before joining the group. The key management framework (Mat Kiah & Martin, 2007) proposed by Mat Kiah et al. has to burden many signaling messages and suffer from the violation of forward secrecy when a moving member leaves the group. The group key manager in TMKM (Sun, Trappe, & Liu, 2004) was considered as the main reference for security parameters of the entire group, which may result in a single point of failure at the managerial level particularly in a group with a large size. As such, the poor use of binary trees causes the mobility events make the tree unbalanced and consequently increase the costs of key management.

As a result, the existing group key management schemes present some problems and consequent challenges in terms of security flaws and performance reduction. Group key management in wireless mobile environments requires to protect the safety of the keying materials not only when members join or leave the group but also when they move between areas of a network. Moreover, group key management needs to achieve operational efficiency by reducing the communication and computation overheads in order to overcome the constraints of both the wireless networks and mobile devices.

The objective of this research is to propose a **hi**erarchical group key management scheme with host **mob**ility protocol in wireless mobile environments (called herein HIMOB), where the group is organized into hierarchical areas, and all areas use a common *TEK*. The host mobility protocol facilitates the group members' movements between areas while remaining in a session of group communication by managing keying

materials and minimizing rekeying process. While HIMOB reduces the communication overhead in terms of number of rekeying messages as well as the number of members affected by rekeying process, the security requirements in terms of backward and forward secrecy are also achieved during all events such as join, move, and leave.

1.2 Research objectives

This research focuses on the issue of user mobility in secure group communication. The objectives that to be met in this research are as follows:

- To identify the constraints and challenges in designing a secure group communication in a wireless mobile environment.
- To design and develop a key management to secure group communication taking into consideration mobility issue in wireless mobile environment.
- To test and evaluate HIMOB in terms of security and performance.

1.3 Research Contributions

This research proposed a group key management scheme suitable for wireless mobile environment. This work makes the following contributions to knowledge of secure group communication in wireless environment.

i. Various approaches used in the design of group key management schemes namely, centralized, decentralized, and distributed (contributory) are explored to identify the underlying common concepts and mechanism associated with each approach. A comparison against identified criteria further highlights the advantages and disadvantages related to each approach particularly when they are deployed in wireless mobile environments.

ii. A comprehensive review on the existing solutions that take consideration host mobility issue is given, and the advantages and the remaining weaknesses pertaining to each scheme is investigated.

iii. Critical challenges and principle constraints corresponding to the design of a secure group communication are identified when a group key management scheme is extended from the wired networks to the wireless mobile networks.

iv. A specific group key management scheme is designed to address the mobility issue in wireless mobile environments where group members freely move between areas while still maintaining session continuity.

v. The HIMOB along with several schemes such as KMGM (Said Gharout, Bouabdallah, Challal, & Achemlal, 2012), GKMW (Mat Kiah & Martin, 2007), FEDRP (DeCleene et al., 2001)and LKH++ (Pietro, Mancini, & Jajodia, 2002)are developed in a simulation environment. Some analyses are conducted in terms of security, communication cost, and *1-affects-n* behavior.

1.4 Thesis structure

The rest of thesis is organized as follows:

Chapter 2 - Literature review: the existing efforts that have been devoted to establish secure group communication in wireless mobile environments are covered in this chapter.

Chapter 3 - research methodology: the research methodology adopted in this research is outlined in this chapter.

Chapter 4 - group key management scheme design: the design of the proposed group key management scheme is covered in this chapter.

Chapter 5 - analysis and result: the simulation results as well as associated evaluation and analysis are presented in this.

Chapter 6 - conclusion and future works: the achievement of this work is concluded and future work direction is outlined in this chapter.

CHAPTER 2: LITERATURE REVIEW

This chapter gives an introduction to group based application and covers significant efforts that have been devoted to establish secure group communication.

Section 2.1 introduces the definition of group based application. Different design approaches for group key management scheme are presented in Section 2.2. The underlying common concepts and mechanisms of each approach in terms of centralized, distributed, and decentralized are further highlighted respectively in Subsection 2.2.1, Subsection 2.2.2, and Subsection 2.2.3. The useful criteria for examining and comparing various group key management design approaches are covered under Subsection 2.2.4, and then, a comparison between these approaches is given in Subsection 2.2.5. Finally, the existing group key management schemes which consider host mobility issue are scrutinized, and then compared against a number of identified criteria in Section 2.4.

2.1 Group based application

Group based applications are able to exploit group communication (or more precisely multicasting communication) as an internetwork function and deliver a single stream of information to a group of destinations (also called recipients, hosts, or members) that want to receive it (Chockler, Keidar, & Vitenberg, 2001; Cisco Systems, 2012; Paul, 2012). A (or many) source(s) sends a single copy of a data packet and the network intermediate devices duplicates the packet as required at the network elements such that each destination receives a copy of the packet (Cisco Systems, 2001; Savola, 2008). This type of communication makes the most efficient use of network resources by attenuating processing overhead associated with replication at the source and the bandwidth overheads due to sending duplicated packet on the same link.

Figure 2.1 depicts an example of multicast communication including a set of network elements such as multicast enabled routers and wireless access points as well as a set of multicast recipients. A single source sends data to the network, and the routers copy the data only at the network segments where currently contain some members of the group. Data transmission from the source to the recipients is indicated with dotted arrows in Figure 2.1. A set of members who wishes to receive data traffic is {*m*1, *m*2, *m*3, *m*4, *m*5, and *m*6}. The router of each network segment which has a recipient duplicates the data traffic and send a single copy to its recipients.

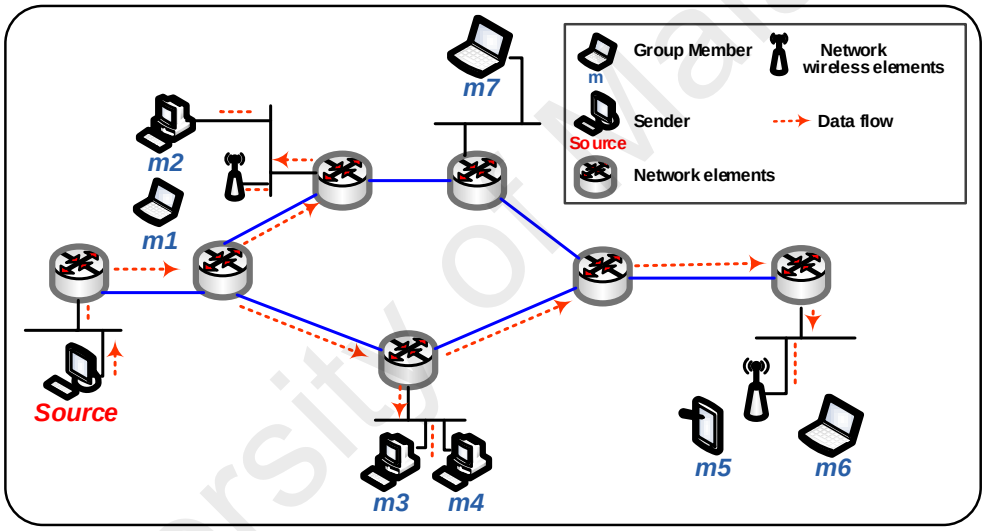

Figure 2.1: An example of group communication.

Group communication may either be deployed with IP multicast architecture or application layer multicast (L. Li, Jun-Hong, Gerla, & Maggiorini, 2005). In the IP Multicast architecture, the network routers are responsible for duplicating data to deliver it to the intended receivers (Quinn & Almeroth, 2001; Ratnasamy, Ermolinskiy, & Shenker, 2006; Savola, 2008). In contrary, application layer multicast is simply the implementation of multicasting as an application service instead of network service. In such multicast service, the end hosts must replicate data instead of routers, which causes possibly immediate deployment of group communications over the Wide Area Network

(Hosseini, Ahmed, Shirmohammadi, & Georganas, 2007; Yeo, Lee, & Er, 2004). However, multiple copies of the same packet are required to be sent on the same link.

Since the terms group communication, and multicast communication carry the same meaning (Banerjee & Bhattacharjee, 2002; El-Sayed, Roca, & Mathy, 2003), the group communication is used as the representative of both terms throughout this thesis. Likewise, the terms members, hosts, and recipients reflect the same meaning and are used interchangeably in the thesis.

2.2 Group key management approach

The design of a group key management is a vital component of any security architecture for group communication. The role of entities and the processes involved in managing all aspects of cryptography keying materials is specified with a group key management scheme. Depending on who is the designated entity for governing the keying materials, the group key management is distinguished by three approaches as illustrated in Figure 2.2; centralized, decentralized and distributed. In the following section, each approach is presented and then, the underlying common concepts and mechanism are further highlighted in order to identify the advantages and drawbacks of each category.

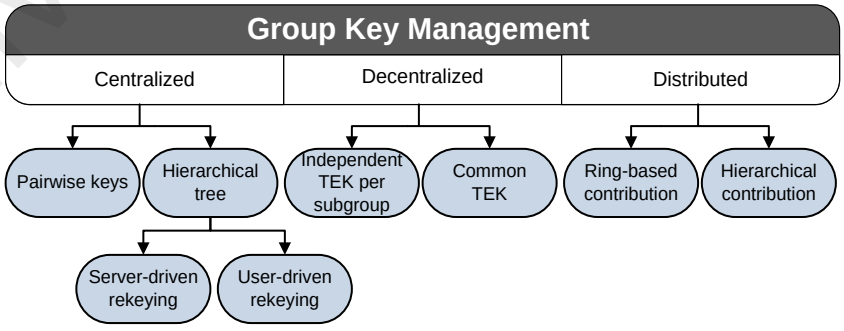

Figure 2.2: Taxonomy of group key management protocols.

2.2.1 Centralized group key management

A single entity referred to as group key manager (*GKM*) is responsible for generating, distributing and updating the traffic encryption key (*TEK*) whenever it is required (Baugher et al., 2005). This approach is further classified into two categories as illustrated in Figure 2.2 depending on the technique used to disseminate the *TEK*. A summary of each category is provided in the following sections.

2.2.1.1 Pairwise keys

To manage the keying materials, the *GKM* shares an individual secret key with each member of the group. This key is used to set up a secure channel between each member and the GKM in order to securely deliver the new *TEK* whenever any changes occur in group membership (H. Harney & C. Muckenhirn, 1997; H. Harney & C. Muckenhirn, 1997). In this scheme, when a member joins the group, it is prevented from having access to the previous group information (i.e. achieving backward secrecy) by updating the keying materials and delivering to all group members with one multicast message. However, to prevent the leaving member from having access to future information (i.e. preserving forward secrecy), *O(n)* rekeying messages are required, where *n* is the number of group members. This is due to the fact that the rekeying message can be encrypted with the old *TEK* during join event, whereas the rekeying messages must be encrypted with each member individual key in leave event as the old *TEK* is compromised with the leaving member. While maintaining the backward secrecy (i.e. a new joining member must be prevented from having access to previous security information of the group) requires only one multicast message, the forward secrecy (i.e. a leaving member must be prevented from having access to future security information) is assured with *O(n)* rekeying unicast messages, where *n* is the number of group members. Thus, this solution is not suitable for large and dynamic groups.

2.2.1.2 Logical Key Hierarchy

The Logical Key Hierarchy (*LKH*) approach is one of the famous schemes in this category that was proposed by several research groups nearly at the same time (Chung Kei et al., 2000; Wallner et al., 1999). A key server is responsible for maintaining a logical key tree. The key tree consists of key nodes and user nodes. The key corresponding to the root of the tree is considered to be the traffic encryption key *TEK*. The leaves of the tree are the individual keys associated with each member of a group. The intermediate keys referred to as key encryption keys (*KEK*) are used by the key server to securely deliver the *TEK* to group members. Figure 2.3 shows a binary hierarchy of keys built for a group with seven members {*m*1 … *m*7}. In such schemes, each member must hold the keys on the path from the leaf to the root of the tree. For example, Member *m*1 owns {*KEK*1, *KEK*12, *KEK*1234, *TEK*}.

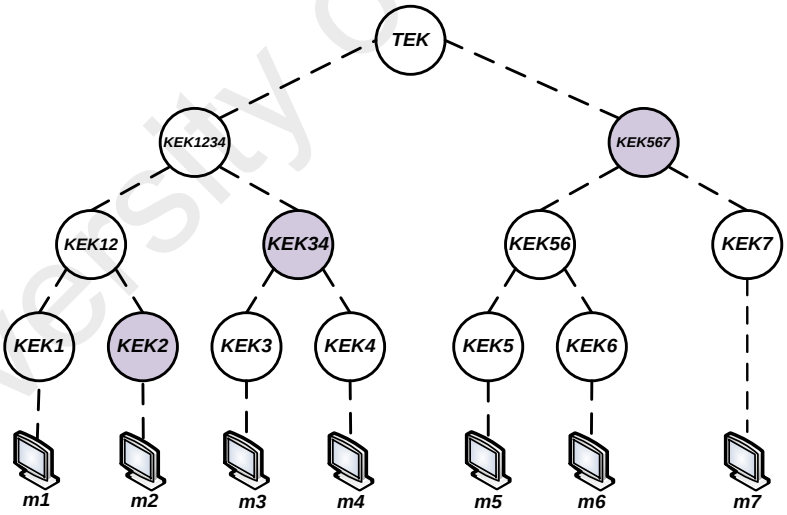

Figure 2.3: Logical Key Hierarchy.

Using *KEK*s reduced the required number of update messages, specifically when a member leaves the group. As a result, this method can scale to the large group size since the number of messages for updating keying materials is significantly reduced on any

changes in group membership. Nevertheless, dependency on a single key server shows a single point of failure and creates a performance bottleneck.

As illustrated in Figure 2.2, this approach can be classified as being either a server driven rekeying, or user driven rekeying. The detail of each of these categories is provided in the following sections.

(a) ***Server driven rekeying***

In this sub category, when a member joins or leaves the group, the GKM or key server is responsible for updating the keying material including the *TEK* and auxiliary keys (*KEK*s), and delivering them to the remaining members in the group (Chung Kei et al., 2000; Wallner et al., 1999) . For example, in Figure 2.3, when *m7* joins the group, the GKM generates the set of keys {*KEK*7, *KEK*567, and new *TEK*} and delivers them to *m7*. The GKM then, sends the new *TEK* by a multicast message encrypted under old *TEK* to the group members. In this method the communication overhead for maintaining backward secrecy is $\log_2 n + 1$, the required number of update messages to assure forward secrecy is $2\log_2 n$, where *n* is the number of group members. Further research in this area has been conducted by (Angamuthu & Ramalingam, 2012; Z.-Z. Chen, Feng, Li, & Yao, 2008; Desmond Ng, Cruickshank, & Sun, 2006; Heydari, Morales, & Sudborough, 2006; Jun, Yu, Fanyuan, Dawu, & Yingcai, 2006; Ng et al., 2007; Park, Je, Park, & Seo, 2014).

(b) ***User driven rekeying***

OFT (Sherman & McGrew, 2003), ELK (Perrig, Canetti, Song, & Tygar, 2001), SKD (Lin et al., 2009), and (Rossi, Pierre, & Krishnan, 2010; Yi-Ruei, Tygar, & Wen-Guey, 2011) transfer the calculation of the *KEK* to members rather than attributing by key server. In such schemes, each member is able to calculate the entire required ancestor *KEK*s (all the *KEK*s through the path from the leaf to the root of the tree) using a key derivation

function based on pseudo random functions. The advantage of this mechanism is to reduce the number of rekey messages from $2\log_2 n$ to only $\log_2 n$.

2.2.2 Distributed approach

There is no explicit key or central entity in this approach, and all members contribute to computing the keying materials. This approach eliminates the need for a central entity while providing uniform distribution of the work load for key management. Therefore, member failure will not affect the whole group, since all members are treated equally. In this approach the *TEK* is established by extending an asymmetric cryptography algorithm such as the Diffie-Hellman key exchange protocol to a group of participants. All members can form either a hierarchical tree or a ring to generate the traffic encryption key (TEK).

2.2.2.1 Ring based

The group members form a virtual ring of their contributions to generate the *TEK*. For this purpose, the Diffie-Hellman (DH) key exchange algorithm, which is regularly used for two parties to agree on a common key, is extended to a group that may have *n* members. The group agrees on a pair of primes (*p* and α) and calculates the intermediary values in a distributed fashion. The first member computes the first value and passes it to the next member. The last member is eventually able to generate the *TEK* and the cardinal value that is sent to other members to extract the *TEK*. GDH (Michael Steiner, Waidner, & Tsudik, 1998) was the earliest scheme in this category, and subsequent studies (Burmester & Desmedt, 2005; X. Guo & Zhang, 2010; C. Li & Xu, 2013; Wu, Mu, Susilo, Qin, & Domingo-Ferrer, 2009) have attempted to advance this scheme. The advantage of such approach is all members are treated equally and if any member fails to complete the setup, it will not affect the whole group. However, key computation must be calculated serially in multiple rounds and requires strict synchronization.

2.2.2.2 Hierarchical

The group uses two party *DH* key exchange scheme and forms a logical hierarchy tree to agree on the *TEK*. Using the logical key hierarchy results in reducing the number of keys held by group members. For example, TGDH (Kim et al., 2004b) is a well-known scheme in this category where the secret key of each parent node in the tree is derived from the secret key of one of its two children and the blind key of the other child by using the *DH* key exchange protocol. Several schemes in this category have been presented in the (Amir et al., 2005; X. Chen, Ma, & Yang, 2007; Konstantinou, 2011; Lv et al., 2012; Magliveras et al., 2008; Zheng et al., 2007). The amount of interaction between members of a group to compute the *TEK* is either independent of the number of members or minimized as low as $\log_2 n$. Nevertheless, the group members must be synchronized to iteratively compute parental keys from their two children's keys, because any delay causes interruption in the key agreement. Moreover, relying on a leader during the setup time still leaves a single point of failure.

2.2.3 Decentralized group key management

In this approach, a large group is split into smaller subgroups and placed in hierarchical levels. Each level, which could consist of one or more entities, is responsible for the key management in its constituent levels while maintaining some dependencies on the upper level entity. This approach is distinguished with two categories including the common *TEK* per all subgroups, and the independent *TEK* per each subgroup (as in Figure 2.2).

2.2.3.1 Common TEK

In common *TEK* approach such as (Thomas Hardjono et al., 2000), one entity is responsible for generating and distributing the *TEK* to members of a group through the subgroup managers. To ensure perfect backward and forward secrecy, all group members

commit to the new *TEK* on any changes in the group membership. Therefore, this approach suffers from *1-affect-n* phenomenon since all group members are affected by any changes in the group membership. This approach has been studied by (Heba K, 2004; J. H. Li et al., 2008; Mat Kiah & Martin, 2007; Nemaney Pour et al., 2007).

2.2.3.2 *TEK* per each subgroup

In order to alleviate the *1-affect-n* phenomenon, another approach which was employed by (Y. Challal et al., 2008; Cho et al., 2008; Mehdizadeh, Hashim, & Othman, 2014; Mittra, 1997; Piao, Kim, Tariq, & Hong, 2013; Youngjoo Shin & Hur, 2012) used the independent *TEK* for each subgroup. As a result, when a membership change occurs in a subgroup, it affects only the members residing in that specific subgroup. However, this approach has the drawbacks of affecting data path, since the data passing from one subgroup to another must be translated at the edge of each subgroup. Figure 2.4 shows a group divided into five independent subgroups in which each subgroup has its own *TEK*.

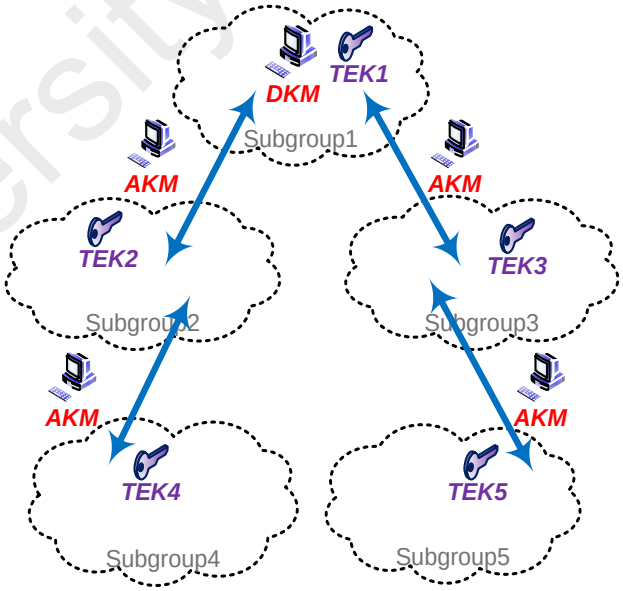

Figure 2.4: An example of decentralized approach with *TEK* per subgroup.

2.2.4 Group key management requirements

This section presents specific requirements for group key management; security, efficiency and scalability. The criteria from this requirement will also be used in further discussion and analysis in subsequent chapters. Each requirement is listed and explained as follows.

2.2.4.1 Security requirements

Backward secrecy: A user who wishes to join the group should be prevented from having access to the previous keying materials. The backward secrecy is preserved by updating the keying materials in a group whenever any join event occurs. As a result, the new member is prevented from decrypting previous traffic in the group, which he may have recorded. This requirement ensures that the former encrypted data remains secret.

Forward secrecy: A user who leaves a group should not have access to any future keying materials. In order to achieve the forward secrecy, the keying materials in a group should be updated when a leave event occurs. Thus, the leaving member is prevented from decrypting and having access future traffic in the group. This security requirement makes sure that the future encrypted data remains secret even if the key is compromised.

Key independence: the entire keying materials should be absolutely independent from each other. Thus, the disclosure of one key should not comprise other keys employed in secure group communication.

Resistant to collusion: any set of fraudulent users should not be able to collude and deduce the current traffic encryption key.

2.2.4.2 Efficiency requirements

Communication overhead: updating keying materials regarding any changes in group membership should not induce a high number of messages, especially for dynamic groups.

Computation overhead: the generation and distribution of traffic cryptography keys should not induce an intensive computation processing at the control manager and user levels.

Storage overhead: the number of keys that need to be managed and securely stored by all communicating entities should be kept to a minimum.

2.2.4.3 Quality of service requirement

***1-affects-n* phenomenon:** a single change in group membership must not result in a rekeying process that affects the entire group members to update the traffic encryption key (*TEK).*

Service availability: the operation of group communication should not be cease due to the failure of a single entity in the key management architecture.

Scalability: the solution should be capable to scale the scope of key management to large and widely distributed groups.

Reliability: the delivery of keying materials must be reliable so that all members of a group are guaranteed to receive the keys in a timely fashion.

2.2.5 Group key management approach summary

A comparison of three different design approaches to group key management is presented in Table 2.1. While the centralized approach scales group key management to the group with large size, dependencies on a single key server leave a single point of

failure. Even though the distributed approach alleviates the weakness exhibited in the centralized approach, the processing time and communication requirement increase because of the growing number of members in the group. The decentralized approach establishes a scalable secure group communication and allows more entities to fail before the whole group is affected. This approach can collaborate with other group key management scheme to build an integrated efficient solution.

Extension of aforementioned approaches to wireless networks introduce more challenges that prevent them from performing efficiently. Bandwidth limitation, resource constraints and widely dispersed mobile devices can affect the performance of group key management schemes.

Centralized approach encounters the lack of scalability in wireless networks where offer group members flexibility by allowing them to receive group communication ubiquitously. When the number of widely dispersed group members increase, the required messages to update the keying materials significantly increases. Therefore, a single key server can be swarmed with multitude requests from group members which trigger failure of entire group key management operation. Meanwhile, in highly dynamic environments, member location is required to be tracked by assisting a third party like base station in cellular networks. Thus, the movement of a members leads to enforcing to update keying materials throughout the wireless domain.

Decentralized approach divides a group into several subgroups which may belong to same or different networks. Since the members move across the wireless networks and need to access to the content of group communication ubiquitously, an authentication mechanism is required to be integrated with group key management in order to verify members before they could obtain the necessary keying materials used in target subgroup. Another security concern in this approach is the existence of third party entities that can

compromise the security properties of the protocols. Thus, the key management scheme must consider the level of trust impart to these entities.

Table 2.1: Comparison of secure group key management approaches.

Evaluation properties	Centralized Approach	Decentralized Approach	Distributed Approach
Key management	A key server	A number of key servers	Contributory
Dependency on a central entity	Yes	No	No
Easy to manage	Yes	Yes	No
Key types	Asymmetric / symmetric	Asymmetric / symmetric	Asymmetric
Trust among group entities	Between each member and the key server	Between each member and its subgroup key server	Between all members
Structure of keys	Pairwise, or Logical key hierarchy	Pairwise, or Logical key hierarchy	Ring based, or Hierarchical cooperation
Join communication cost	Logarithm of the group size	Minimize to one message for join event.	Varying
Leave communication cost	Logarithm of the group size	Number of members in a subgroup where leaving occurs.	Varying
Computation cost: Server	High	Moderate	…
Computation cost: User	Low	Low	High
Bottleneck	Yes	No	No
Service availability	The key server failure ceases the service	Yes	Yes
Scalability supported	Yes	Yes	No
Scalability in group size	Moderate	High	Low

Distributed approach offers fault tolerance feature, although this feature can sacrifice the efficiency of the schemes at the cost of high communication and computation overhead. These kind of schemes tend to apply widely asymmetric cryptography key algorithms in particular Diffie-Hellman key exchange protocol to compute the TEK,

which is computationally expensive due to the process of multiple exponentiations. The use of asymmetric cryptography algorithms induces a significant computation cost due to the process of multiple exponentiation as well as a long time to reach the common TEK by all contributing members. Therefore, when the group size increases, the wireless device resources are drained faster.

Prior to scrutinize the existing group key management schemes proposed for wireless mobile environments, the characteristics and constraints of such environments as well as the challenges that must be considered in designing a group key management scheme for such environment are explored in next section.

2.3 Design challenges in wireless mobile environments

Wireless mobile networks provide effective and efficient data delivery services where the deployment of infrastructures is impossible. In the following sections, the wireless mobile environments are investigated in terms of the characteristic of single-hop operation mode and multi-hop operation mode. The different features of such environments are illustrated to highlight the challenges that must be considered in the design of a group key management scheme.

2.3.1 Types of wireless environments

The popularity of wireless communication can be seen everywhere in the form of wireless Local Area Network (WLAN) (Crow, Widjaja, Kim, & Sakai, 1997), Wireless Personal Area Networks (WPAN) (Fisher, 2007), Cellular networks such as GSM / UMTS (Taferner & Bonek, 2013), Ad-Hoc networks (Mohapatra, 2005) and Wireless Sensor network (WSN) (Yick, Mukherjee, & Ghosal, 2008). Wireless networks can be organized into the infrastructure based and infrastructure less which respectively provide single-hop operation mode and multi-hop operation mode for mobile nodes to access the system (Cavalcanti, Agrawal, Cordeiro, Bin, & Kumar, 2005; Pahlavan, 2011).

2.3.1.1 Infrastructure-based

WLAN, and Cellular networks such GMS / UMTS are classified as infrastructure based platform. In such networks, the entities referred to as base station (BS) or access point (AP) are attached to the wired networks by a fixed infrastructure. These entities are responsible for establishing point to point links between wireless nodes or networks at two distant locations (Hiertz et al., 2010). Therefore, a source node can reach its destination node directly in a scenario comprised of a pair of communication entities. Figure 2.5 depicts an example of an infrastructure-based wireless network.

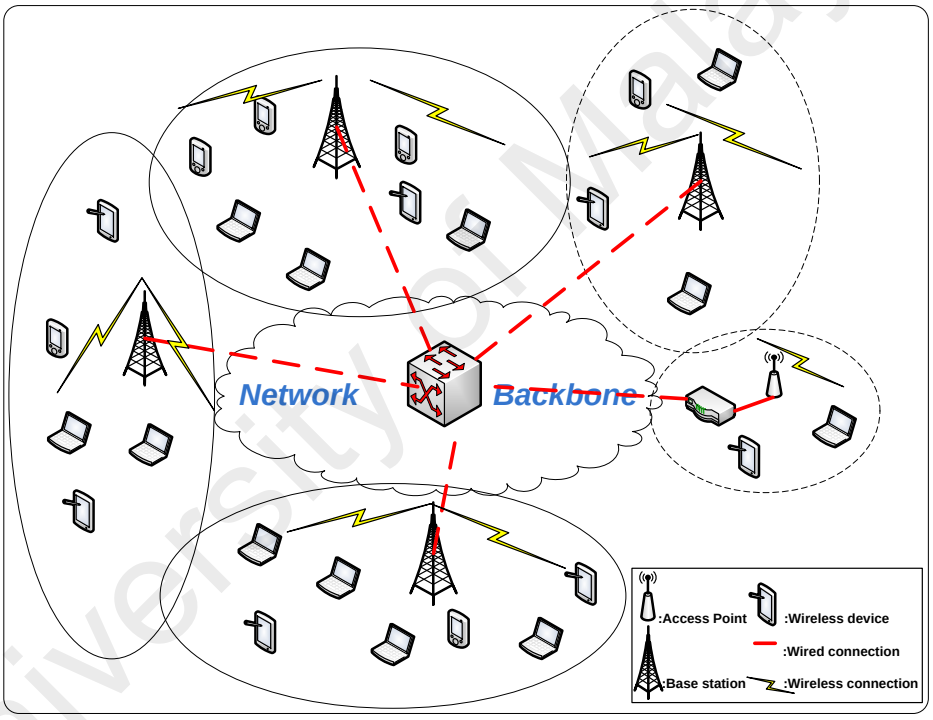

Figure 2.5: Infrastructure-based wireless network.

2.3.1.2 Infrastructure-less

Infrastructure-less platform such as Ad-Hoc networks, WPAN, and MANET is another paradigm of mobile communications that is multi-hop environment where a collection of wireless nodes communicates among themselves without the help of any infrastructure such as a base station (Benyamina, Hafid, & Gendreau, 2012; Chlamtac, Conti, & Liu, 2003; Yick et al., 2008). Every node can play the role of intermediary

station that relays packets of other nodes towards their destinations that otherwise cannot be reached using a single-hop transmission. Therefore, in a pair of communication entities a source can reach its destination node through only two or more single-hop communication links. Figure 2.6 illustrates an infrastructure-less network.

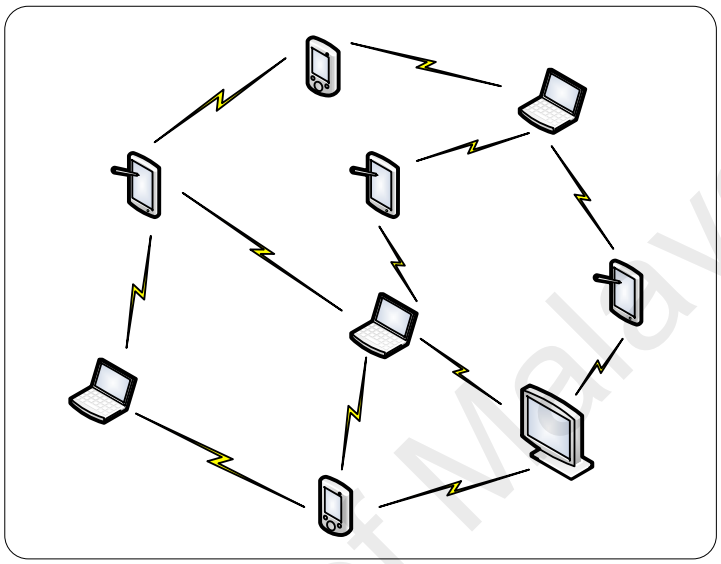

Figure 2.6: Infrastructure-less wireless network.

2.3.2 Wireless mobile networks characteristics

In comparison to wired networks, wireless networks introduce some interesting features which complicates establishing secure group communications. Using air for traveling data make such networks more insecure and susceptible to numerous attacks than wired networks. Several principle constraints and challenges induced by the wireless mobile environments that were sparsely discussed by (Bouassida et al., 2008; Goldsmith & Wicker, 2002; A. K. Gupta, 2008; Pierre, 2001) are summarized as follows:

2.3.2.1 Non-fixed and wireless network connectivity

Wireless networks use radio frequency signals to exchange data between two or more physical devices instead of relying on pre-existing infrastructure such as wire. This property enables wireless networks to be easily deployed in different environments in

spite of lack of infrastructure. However, in comparison to fixed wired networks, the wireless mobile networks suffer from some different security vulnerabilities:

Security attacks: precisely connection to a wireless link is so easy, thus wireless communication is more susceptible to attacks such as eavesdropping and monitoring of data traffic. Therefore, the security of such communications is likely to be compromised much more easily than that of wired communication. This could cause further security breaches if certain measures are not in place. For example, messages or group data can easily be read or tampered by an adversary who is eavesdropping on the group data traffic if the data were not encrypted.

Trust within foreign networks: the absence of wire facilitates the members with allowing to move around. Location changes may require mobile members to occasionally communicate via foreign networks where cannot be always trusted. This affects the amount of trust to impart on governing entities within foreign networks. On the other hand, the visiting mobile hosts may gather information about the local security services for the networks they visit, which result in security threats, so the foreign networks require to consider the amount of trust they want to place on the visiting mobile members.

Network disconnections: network failure is a greater concern in mobile environments than in traditional networks since transmission rates in wireless networks have high tendencies for changing over time in comparison with wired networks. The frequent changes in transmission rate result in unnecessary disconnection, jitter, and delay in communication between collaborating entities. This issue can be more aggravated when a member changes its physical location because the member needs to disconnect from the old location and re-connect with respect to the new location.

Nature of wireless devices: While the performance of mobile devices particularly mini PDAs or smart phones is improving rapidly, they still suffer from some limitations as compare to the desktops regarding limited data storage capacity, having shorter battery life, as well as having slower processing speed, which hinder intensive processing. With such constraints, the cryptographic computation must be remained as low as possible, and the number of keys that a mobile device needs to store must be minimized.

Absence of infrastructure: in compare to the other kind of wireless networks, mobile Ad-hoc networks has its own specific characteristics, which is the absence of infrastructure. This characteristic eliminates any possibility to establish a central entity for managing the access of members to the network and defining the security services and policies for the network, as well as it may be responsible for distributing the keying materials. The lack of central network manager leads to applying the traditional authentication and key distribution models are hardly applicable in such networks.

2.3.2.2 Host mobility

Mobile members are able to freely move within wireless mobile networks. However, this issue exhibits new problems as follows:

Handoff operations: when a user changes his/her physical location, some kind of handoff operation is required to handle his/her movement. This can affect network connection (disconnection, and poor connectivity) while the user is moving. Network connection has to adapt to this behavior of user mobility by re-connecting user with respect to new location without interruption to the on-going service used by the user.

Management of keying materials: host mobility implies new issues pertaining to the management of cryptographic keys, which include who is responsible for governing the moves and who must keep track of keying materials.

Heterogeneous network: In contrast to most stationary computers, which stay connected to a single network, mobile computers encounter more heterogeneous network connections in several ways. First, as they leave the range of one network transceiver and switch to another, they may also need to change transmission speeds and protocols. Second, in some situations a mobile computer may have access to several network connections at once, for example, where adjacent cells overlap or where it can be plugged in for concurrent wired access. Moreover, mobile devices may need to switch interfaces, for example, when going between indoors and outdoors.

2.3.3 Design challenges to group key management

The natures of wireless mobile environments and the characteristics of the group based applications as well as the required level of security remain more complexity in developing a group key management scheme in wireless mobile environments. The security issues and critical factors concerning with designing a group key management scheme in wireless mobile environment are discussed in this section.

2.3.3.1 Supplementary key management protocol for handling host mobility

In wireless mobile environments, group members are not only allowed to join and leave a group communication for reason discussed in Section 2.3, but also able to move between the areas of a network while remaining in the session. Thus, in addition to the protocols for achievement the requirement specified in Section 2.3.1, another protocol is needed to govern mobility of members between areas.

In a naïve solution, a move event may be treated as a leave in the old area where the moving member has moved from, and followed by a join in the new area where the moving member enters. This solution result in updating keying materials in both areas in turn for provision of forward and backward secrecy, which leads to increasing rekeying messages overhead and the number of affected members. However, the provision of

forward secrecy is not necessary in the old area as the member is remaining in the session while changing point of attachment to the network.

Moving members are able to accumulate the local security services information of areas which have previously visited. It is imperative to ensure that the area is protected against malicious activities carried out by such group members who are moving one area to another to collect security information corresponding to each visited area. Another security breach may arise in the visited areas is that the keying materials may be compromised by the moving members after they leave the group if the keying materials are still valid in the visited areas. Therefore, in order to achieve forward secrecy, not only the keying materials of the area where the moving member leaves the group must be updated, but also the visited areas that their corresponding keying materials are still valid and carried by the leaving member need to perform rekeying process.

2.3.3.2 Performance Requirements

The performance requirement can be divided into several categories in terms of communication, computation, storage, energy and bandwidth requirements.

Computation and storage requirements: mobile nodes are usually equipped with limited power and storage, which restrict their abilities to do rigorous computation and to store large amount of data. As a result, the number of keys that each member needs to store throughout a group communication as well as all computation related to security operations to be done by mobile devices must be kept as low as possible.

Communication, bandwidth, and energy requirements: the bandwidth available in wireless environments are limited compared to wired networks, which likely lead to the frequent cut off and high latency. The bandwidth is a critical aspect, as it increases, power consumption consequently grows up which shorten the battery life of wireless devices.

Hence, energy restrictions of wireless devices will limit the effectiveness of data throughput to and from the device even if wireless networks connections deliver stable high bandwidth. Therefore, the number of messages exchanged between the key managers and receivers must be minimized to overcome these limitations.

Next section discusses the group key management schemes particularly designed for wireless environments, which tried to take address the aforementioned challenges.

2.4 Related work

Previous proposals such as those discussed in the literature are mostly designed for wired environments. Few efforts have been carried out to extent the group key management protocols to mobile environments, however most do not address host mobility issues (Mat Kiah & Martin, 2007). Indeed, in wireless environments participating members in a group communication are able to move from one subnet or area to another one as depicted in Figure 2.7.

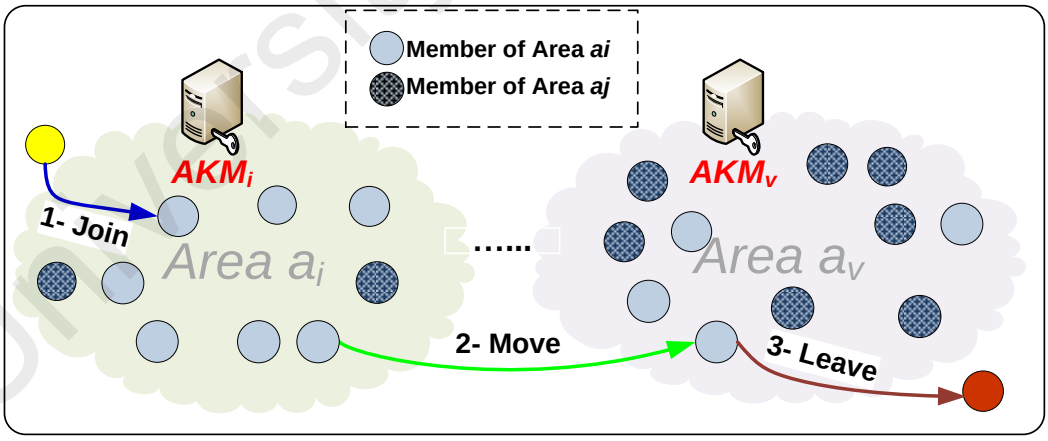

Figure 2.7: Decentralized approach with the additional possibility of transferring between areas.

This mobility can be considered as a leave in the old area and subsequently a join in the new area in traditional solutions, which result in updating keying materials twice. As

discussed in Section 2.5, the scarcity of resources in wireless environments indicates that this naïve solution is not efficient since it needs to send more controlling messages to manage mobile entities, which induces further interruptions in group communications.

Little research has been conducted that explicitly addresses the mobility issue of members from one area to another one while remaining in the session. This section gives an overview of several existing key management as shown in Figure 2.8 for host mobility in secure group communication in details.

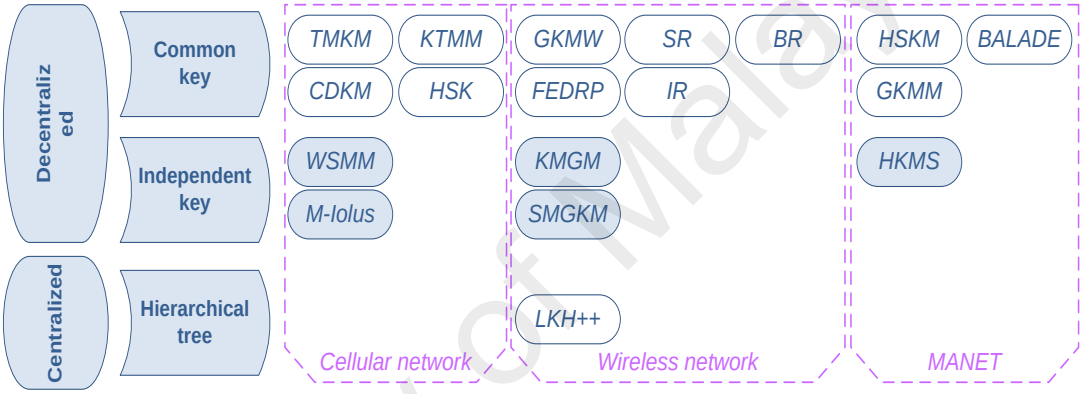

Figure 2.8: Classification of group key management with host mobility.

2.4.1 KMGM (Said Gharout et al., 2012)

KMGM is a key management protocol for secure group communication in mobile environments. KMGM adopted *ASGK* (a decentralized approach with the independent *TEK*) (Y. Challal et al., 2008) as its main group key management scheme. To manage mobility of users, each *AKM* maintains two lists, a list of current members residing inside the area called *ListM* and list of the old members who have already moved to the other areas called *ListO*.

When a member moves from the residing area to another one in the same cluster, it sends a move request to the manager of the visiting area. The AKM_v verifies that the visiting member is valid member and really comes from the previous area. If the

verification succeeds, the new area key manager AKM_v adds the visitor to the current member list $ListM_v$ and the previous AKM_i adds the departure to the old member list $ListO_i$. The *TEK* and *KEK* of the new area are delivered to the visitor since these keys are different from the departed area. Figure 2.9 illustrates the message flow of this protocol.

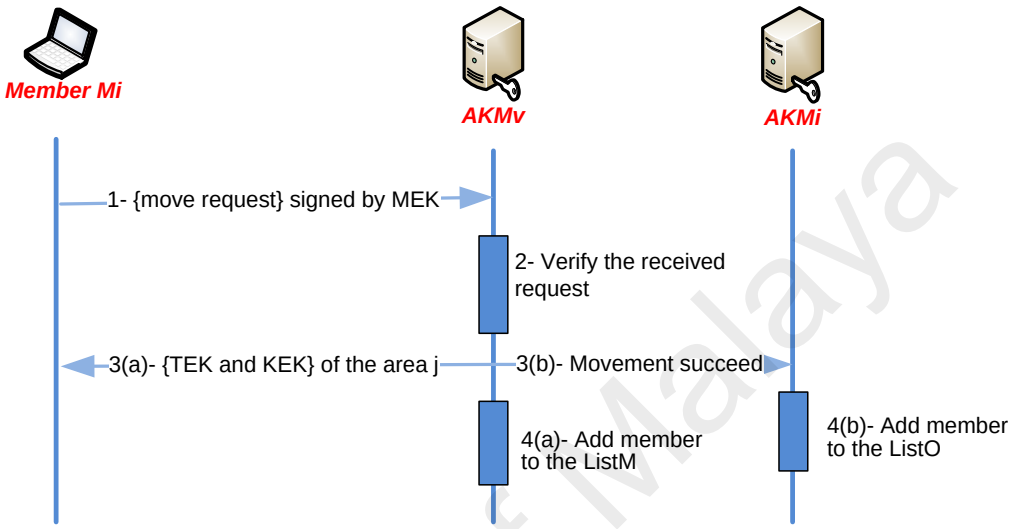

Figure 2.9: KMGM host mobility protocol message flow.

During the leaving event, all keys including *KEK*s and *TEK*s of the areas or clusters which have already been visited by an existing member must be refreshed. When a member leaves an area, other areas are informed about the member departure. The new *TEK* and *KEK* are delivered to the remaining members of the area which leaving event occurred protected by each member individual key. The *AKM* empties the old member list *ListO*. In other areas a_t where the leaving member is in $ListO_t$, the new *KEK* and *TEK* must be sent by secure unicast message. In the remaining areas, the new *TEK* is sent by a multicast message protected under *KEK* to the members residing within the area.

Pros: the scheme introduces null rekeying for intra move between areas in a cluster. The simulation of SGC including five subgroups with inter-move variation showed that the communication overhead and *1-affects-n* overhead using KMGM is lower than the other similar schemes.

Cons: the backward secrecy is breached as the mobile member may access to the security information of visited area which is valid prior the mobile member joined the group.

2.4.2 SMGKM (Abd-Alhameed, Mapoka, & Shepherd, 2014)

SMGKM is a slot based multiple key management scheme, which support the mobility of single and multiple members across a homogeneous or heterogeneous wireless network while participating in multiple group services. A two tier decentralized approach is adopted where the first level is the domain level comprised of the core wired part and the second level is the wireless part consisting of multiple clusters. The *DKM* is responsible for initial key management and authentication procedure at domain level, and the *AKM*s as area key managers are responsible for generating and distribution the TEK in each cluster independently. In order to maintain host mobility and to track mobility, each *AKM* manage a mobility list called session key distribution list (*SKDL*).

$SKDL_i$ is a list of mobile members with their corresponding session key SK_{Mi} derived by the DKM in cell *i*. when a member wishes to move from its original area *i* to new area *v*, it simultaneously informs both *AKM*s in both areas encrypted under its private session key specific to the AKM_i (called $SK_{Mx\text{-}AKMi}$). The moving member generates the private session key associated to the *AKM* using a key derivation function such as *SHA1* using parameters such as *AKM* identity and member's authentication key. The move request of the moving member is verified by the AKM_i using member's private key. AKM_i waits for a period T_{up} to accommodate incoming request.

After elapsing T_{up}, the AKM_i transmit security information of the moving member to AKM_v via a fast link using context transfer protocol (CXTP) which reduces service disruption during movement. Moving member on arrival at the new area *v* sends a move request encrypted with specific private key related to the AKM_v. If the verification of new

arrival against the security information received from AKM_i succeeds, the AKM_v updates the *TEK* of corresponding area and send it to the new moving member and residing member in area *v*. AKM_i also removes the security information of moving member from its $SKDL_i$.

Pros: the scheme shows resource economy in terms of communication complexity and storage overheads. It also provides key management for multiple group communications.

Cons: Although a mobile member must revoke the holding traffic keys related to the old area upon entering the new area, this criterion cannot prevent a malicious member from accumulating the keys of the visited areas, thus the forward secrecy is breached when the member leaves the group. The new *AKM* is not able to authenticate the new arrival if the old *AKM* failed before sending the moving member's security information. The signaling messages for managing mobility event is still slightly high.

2.4.3 WSMM (Jong-Hyuk & Kyoon-Ha, 2006)

The Wireless Subgroup in Mobile Multicast, WSMM, manages separate wired and wireless area. The WSMM organizes the group members into a number of subgroups that are managed by base stations in each wireless cell. The base stations are responsible for generating and managing the keying materials in their wireless cells. Two types of keys are used for data transmission, a wired group key and a wireless subgroup key. The wired key is the root of the logical key hierarchy, where the base stations are located at the leaves of the tree. The wireless key in each subgroup is generated by its base station, and shared between the members of that specific subgroup. Since each subgroup has its own wireless key, the data transmission between cells must be translated at the edge of each cell by its base station.

WSMM considers host mobility as a leave from the old area follow as a join at the new area. Therefore, when a mobile host moves to the new area, if the member authentication succeeds a new individual key is generated and delivered by the base station of the new area to the arrival. Subsequently base station in the old area generates a new subgroup key and disseminates it to the remaining members in its area.

Pros: average delay time for member joining and member leaving process showed better result for this scheme.

Cons: data multicasting latency is still high. The scheme has to incur significant communication overhead for movement process, which is considered as a leave in original area and follow with a join process in new area. The group key manager can become a bottleneck since it may be overwhelmed with the membership requests, thus this also become a performance hurdle when multiple membership occur. The trustworthy issue rises since the BSs cannot be trusted by the content provider to open the data content.

2.4.4 M-IOLUS (Kamat et al., 2003)

In Micro-grouped IOLUS (M-IOLUS), each subgroup manager (called *GSA*) dynamically forms its jurisdiction into a number of micro groups in order to reduce the communication overhead of updating keying materials for any change. A micro key is shared between all members belonging to the same micro group, and the key is used to protect all controlling messages transmitted.

In the case of user movement between micro groups, the moving member informs its subgroup manager regarding its intended move direction from the old micro group to the new micro group. Since the transfer occurs in the same subgroup, the subgroup manager does not need to change the subgroup key. The *GSA* makes a note of the timestamp that the move happened and delivers the micro key of the new micro group to the mobile host.

In this scheme, each *GSA* maintains a table of mobile members who have already moved in a subgroup. When a mobile member transfers from its subgroup to a new subgroup, it sends a move message to the *GSA* of the new subgroup. The new subgroup manager authenticates the moving member by requesting from the old subgroup manager. Once the member is authenticated, the new *GSA* if already connected to the multicast group, sends the group key and the micro group key to the new arrival and then updates the mobile member table. If a leave process occurs, all *GSA*s from the subgroups previously visited by the departed member update their subgroup keying materials.

Pros: the communication overhead is kept as low as possible with a null rekeying cost on member mobility.

Cons: the scheme shows backward secrecy violation in visited areas. The authors did not discuss some important aspects such as security associations between the previous GSA and new GSA. Multiple authentication request received by the GSA incur a performance hurdle. Moreover, if the previous key manager fails, members moving will not be authenticated by the new GSA which result in service disruption. It also increases the number of decryption and encryption at the edge of micro-subgroup.

2.4.5 TMKM (Sun et al., 2004)

This scheme matches the key management tree to the network topology thereby the delivery of keying materials is localized and consequently the communication costs are reduced. The cellular network is comprised of mobile users, Base Stations (*BS*), and Supervisor Hosts (*SH*). The Topology Matching Key Management (TMKM) tree is comprised of three key management sub-trees. The User sub-tree manages the hosts within an area controlled by each base station. The *BS* subtree is used by the supervisor host to govern the hierarchy of keying materials between the *BS*s and the *SH*. Eventually the group manager establishes a *SH* subtree to govern the *SH*s.

In order to keep track of the mobile host and the key updating process, each cell maintains a *Wait to Be Removed* (WTBR) list that contains information about the users who possess a set of valid keys and has already left the cell. The group manager maintains these WTBRs lists.

When a user moves from one area to another area, the following steps take place:

i. The departed user is removed from the user subtree of the old cell, and added its information to the WBTR list of the old cell.
ii. If user has previously visited the new cell, it is located on the branch of the subtree that it previously belonged to. The user's information is eliminated from the WTBR list of the new cell. Otherwise, the user is located at the most recently updated branch of the user subtree.
iii. If the time that the user joined is later than the time of the last key updated due to any departure from the new cell, the user's key subset is updated by the user join procedure described by (Waldvogel, Caronni, Dan, Weiler, & Plattner, 1999). Otherwise, the keys do not need to be updated. The purpose of this procedure is to prevent a mobile member from having access to information before it joins the group in the new cell.

When a member leaves the session, all subset of keys processed by the member and still valid must be updated. All cells that their WTBR lists have the departure information are required to update the leaving member's key subset in accordance with the leave procedure developed by (Waldvogel et al., 1999). The members that have key set similar with departure member must be removed from the WTBR.

Pros: low communication and computation overheads.

Cons: using a single KDC leaves a single point of failure for key manager. The deployment of the KDC in actual environments may show poor performance since it has to incur significant storage and computation overheads. The scheme is network topology dependent, which its actual deployment remains a great challenge. The scheme deems not to be suitable for highly dynamic membership, since the leaving latency increases due to many subgroup keys need to be updated when a member who already roamed many wireless regions leaves the group.

2.4.6 KTMM (Jong-Hyuk & Kyoon-Ha, 2006)

The Key Tree in Mobile Multicast, KTMM, matches the key management tree to the mobile IP network topology. In this scheme, the logical key hierarchy has a fixed degree at the intermediate key node level and a varying degree at the user level. The lowest level of the key tree corresponds to the connection between the base stations *BS* and the mobile members in the wireless cell. The lowest intermediate key in the key tree is associated with each *BS*. In other words, this lowest key node is the subgroup key, which is shared between the base station and the mobile members in each wireless cell.

When a mobile member moves to a new area a_j, the base station BS_j of new area authenticates the arrival member. If the verification of the member is successful, the subgroup key, which is the intermediate key in the key tree already assigned to the new base station, is sent to the mobile member. Meanwhile, the group key manager modifies the key tree.

Pros: reducing communication overhead with matching the key management tree to the mobile multicast environments for localizing the delivery of the rekeying messages. The data transmission and handoff processing showed better performance.

Cons: solely authentication of mobile member in the visited area cannot guarantee the backward secrecy, since the mobile host may access to the security information in visited area valid before he joined the group.

2.4.7 CDKM (Min-Ho, Young-Hoon, & Seung-Woo, 2010)

In Cell-based Decentralized Key Management (CDKM), the group key manager is responsible to handle a join event or a leave event (Min-Ho et al., 2010). The user mobility and keeping track of the user is managed by the Base Station in each cell. For this purpose, each BS handles a key tree, called Independent Subgroup Tree (IST), for its associated users within the cell, and thus does not need to inform the group key manager about the mobility of a member.

CDKM divides an entire group into multiple cell-based subgroups. In order to manage mobility of group users, a user is labeled either with the state of Present in the Cell (*PIC*) or Absent in the Cell (*AIC*) in each cell. When a member moves from its current cell to a new cell, the current cell BS labels the moving member as AIC in the subgroup key tree. In the new cell, the *BS* looks the transferred member up the subgroup key tree. If the member has already visited the new cell, the BS changes the moving member state from *AIC* to *PIC* in its *IST*. Otherwise, the moving member is located at a leaf of subgroup key tree and set its status as *PIC*.

Pros: the member's mobility key management overhead distribute to the BSs, which lead to reducing the communication overhead.

Cons: High dynamic membership imposes extra communication overhead. The subgroup key tree will be unbalanced as it has to keep the location of absence member which causes significant overheads. The BSs cannot be trusted by the content provider to open the data content, which result in the trustworthy issue.

2.4.8 HSK (S. K. S. Gupta & Cherukuri, 2003)

An adaptive scheme was suggested that could provide an efficient and secure multicast in wireless LAN. Users can access the service based on their location. Three schemes were presented; the single session key (SSK) where the base station shares a common key with all the members of its cell, the different session key (DSK) where each one of the members within the cell shares a unique session key with the base station, and finally a hybrid scheme that is a combination of the two schemes. In the hybrid scheme, the base station shares one session key with members who are relatively less mobile and a separate session key with the remaining members.

After a new member joins a cell, the new session key can be delivered by either a multicast communication encrypted using the old session key or unicast messages to all members of the group protected under unique key of each member. When a leave process occurs in group membership since the old session is known by the leaving member, the new session key is unicasted to the remaining members of the group.

In the hybrid scheme, each member is classified as stable or unstable based on the time they have been part of the group. The base station shares the same session key with all members who have been identified as stable members. While the BS shares a separate session key with each unstable member. If a member node remains in the cell more than a predefined time interval, the node is labeled as a stable node. A node is considered stable until it leaves the cell.

Pros: HSK showed the lowest communication overhead in simulation of group communication over all cellular networks with high mobility in compare to the other solutions.

Cons: in order to determine whether a member must be classified as stable or none-stable, a strict time synchronization is required. Moreover, the host mobility, which imposes more overheads, is not addressed by the scheme.

2.4.9 GKMW (Mat Kiah & Martin, 2007)

Mat Kiah et al. (Mat Kiah & Martin, 2007) employed a decentralized approach with a common *TEK* and proposed a group key management scheme that facilitates host mobility in wireless mobile environments using a list as part of the protocol. In this protocol the group key managers including *DKM* and *AKM*s, maintain a list of mobile members who have already moved out to other areas. This list referred to as *MobList*, and used to keep track of mobile users as well as avoid frequent rekeying in an area which may cause disruptions in group communication, keep the *ID*s of the moving member, the multicast group joined by the member, the area that a member is moving from, and the *ID* of the visited area. The transfer of a member of a group from one area to another with backward secrecy is completed as follows:

iv. The mobile member who wishes to move into another area *v* sends a *move_notify* message to the both key managers in the old area (AKM_i) and the visited area (AKM_v) as shown in Figure 2.10.

v. The old AKM_i receives the message and informs the *DKM*. On receiving the message from the old AKM_i, *DKM* checks and sends the *move_notify* message to the new *AKM* in the form of *move_token*.

vi. The new AKM_v checks the received message from the moving member and the *DKM*. If the received message is valid, the new *AKM* looks it up on its $MobList_v$. If the member is not in the *Moblist*, AKM_v has to generate a new area key KEK and deliver to the new arrival and other residing members in its area, since the arrival is a new member. Otherwise, if the arrival member is

already on the $MobList_v$, AKM_v checks whether there have been any updates to its area key since its last visit to the area. If there have been no updates, AKM_v sends a *move_welcome* message to the arrival along with current area key. Otherwise the new area key is sent to the member.

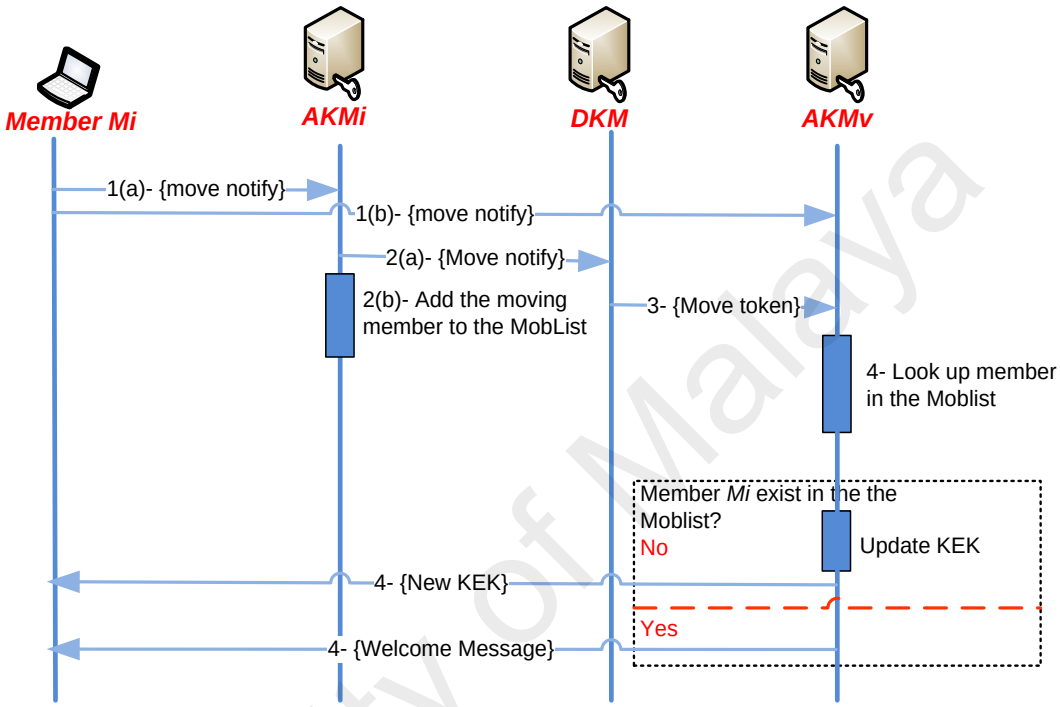

Figure 2.10: GKMW host mobility protocol message flow.

Pros: Using a list as a feature of move protocol allows for efficient processing of members who are returning to recently visited areas during host mobility. The backward secrecy is also maintained by updating the keying materials when a member moves to an area for first time.

Cons: the authors did not discuss the communication overhead of the scheme and only validated the security of the scheme. The number of keys and signaling messages used to manage host mobility incur communication cost on the scheme. The leaving member can hold the valid area keys associated with areas he already visited even after he leaves the session, which lead to breaching forward secrecy in the visited areas.

2.4.10 BR, IR, and FEDRP

Multiple inter area key management protocols namely static rekey (SR) protocol, baseline rekey (BR) protocol, Immediate rekey (IR) protocol, and First Entry Delayed Rekey + Periodic (FEDRP) protocol based on the decentralized approach using a common TEK were proposed to preserve the secrecy of group communications by focusing on distributing, updating, and revoking key encryption keys (*KEK*) as members move within the hierarchy (DeCleene et al., 2001; Zhang et al., 2002).

2.4.10.1 Static rekey (SR)

In this protocol, when a member joins a group via a specific area, it would remain a member in that area. When the member leaves the group, it informs the area manager by sending a signaling message.

Pros: Mobility affects neither the original area nor the new area, which result in the least communication overhead.

Cons: the security requirement is not met since the backward and forward secrecy are breached when member moves between areas.

2.4.10.2 Baseline Rekey (BR)

This is a straightforward approach treats the mobility of a member as a leave from the old area followed by an entry into a new area. The mobile member notifies its local AKM_i about its intents. Then, the AKM_i halts the local transmission and updates the area key within the area for remaining members. Once the area key update is completed, a new traffic key *TEK* can be distributed to existing members. In the new area, the new member informs the new *AKM* of its intent to join. Data transmission is halted one more time while the new area key is distributed to the existing members inside the new area a_j and to the new member, then the new traffic key is delivered to the all members of the group.

Pros: mobile members carry only one valid area key while they are remaining in the session

Cons: the shortcoming of this scheme is that data transmission is unnecessarily interrupted twice during movement among areas. The simulation result showed high communication overhead in both scenarios, low and high membership dynamics, which linearly increase with growing the mobility rate since it cannot distinguish join or leave to/from the group from movement cross the areas.

2.4.10.3 Immediate Rekey (IR)

When a member wishes to move from its current area to a new area, it sends a signal message to the area managers in the old area and the new area. The manager of the old area generates a new area key *KEK* and distributes it via unicast communication to the residue members in its area using their public keys. Meanwhile, the key manager of the new area generates a new area key *KEK* and sends it via a multicast transmission to all members residing within its area and by a unicast message to the newly joined member. Each member holds only a *KEK* of the area in which it currently resides.

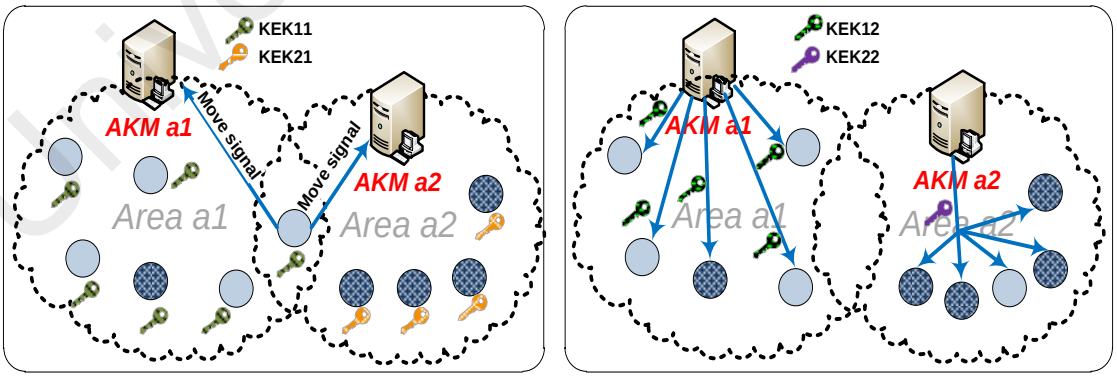

Figure 2.11: diagram of managing a moving member in the immediate rekey protocol.

As shown in Figure 2.11, after transferring the moving member to the new area a_2 the area key manager of area a_1 updates KEK_{11} to KEK_{12} and delivers the new area key to the remaining members by unicast message. In addition, the area key manager of area a_2 refreshes the old KEK_{21} to the new KEK_{22} and distributes it to all members residing in the area by a multicast message.

Pros: the mobile member cannot accumulate valid area keys associated to the visited areas, and therefore the area key is not compromised. The data transmission is not interrupted because of member movement.

Cons: frequent and fast visits of the areas by moving members incur burden communication overhead since the area keys must be changed repeatedly. The communication overhead is still high.

2.4.10.4 First Entry Delayed Rekey + Periodic (FEDRP)

Members moving between areas are able to accumulate multiple area keys and reuse these keys when they return to previously visited area. When a member moves from its current area to a new area, it informs the key manager in the old area and the new area by sending a signaling message. The key manger in the old area adds the departing member to a list called Extra Key Owner List (*EKOL*) and it does not refresh the area key *KEK* in its area. Upon arriving in the new area, the area key manager AKM_v checks the *EKOL* list to verify whether the entering member has previously visited this area. If the member is not on the list, the keying materials are updated by the *AKM* in the new area. Otherwise member can reuse the previous area key. This list is reset whenever any changes occur in the area. Figure 2.12 shows the message flow of FEDRP protocol.

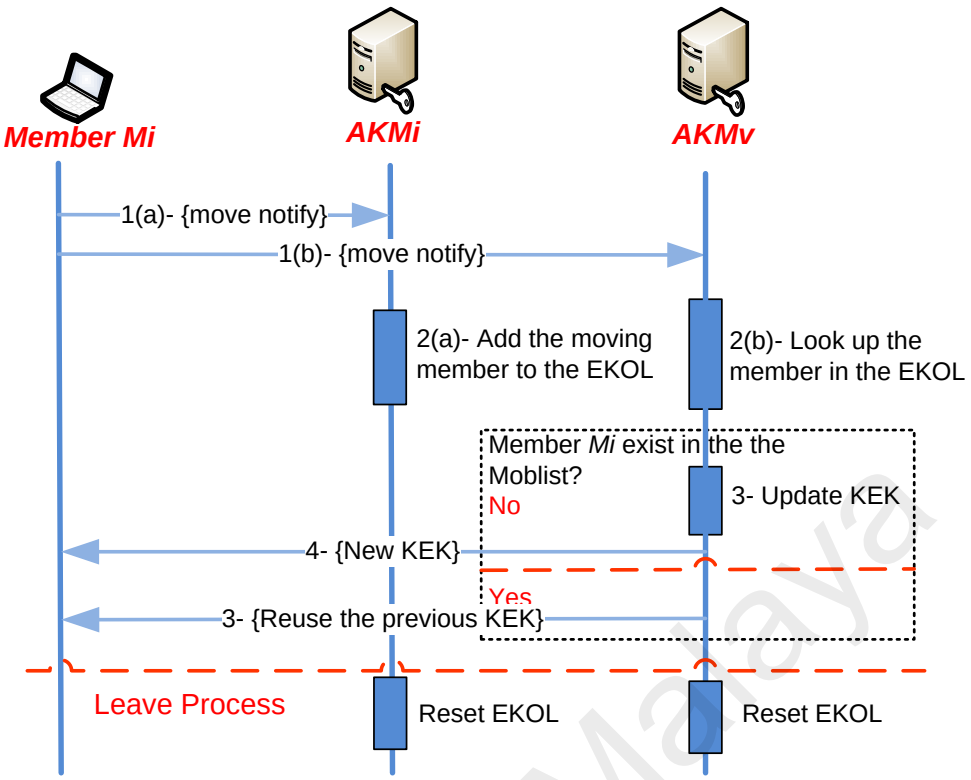

Figure 2.12: FEDRP message flow.

Since the member can accumulate the *KEK*s of different areas during his movement, a maximum amount of time is given to each *KEK* held by a member outside the area. The timer runs and counts up until it reaches a threshold value, at which point a new *KEK* is generated and distributed via unicast transmission to all members within that specific area.

Pros: Hierarchical structure reduces the overheads and supports the highly dynamic membership.

Cons: the scheme may suffer from colluding attack since the area key have been repeatedly reused for often moving members. The authentication of user is not addressed in the new area.

2.4.11 SHKM (Cao, Liao, & Wang, 2006)

SHKM is a decentralized scheme that uses the independent *TEK* approach where the subgroups are organized into a hierarchical structure with different priorities. The priorities of cluster heads are higher than the priorities of the local users and defined by the level of positions where they join. Moreover, users belonging to a higher priority subgroup have the right and are capable of deriving the key of lower priority subgroups but the opposite operation is not allowed. The forwarding entities in the multicast communication undertake the responsibility of subgroup management. Since the forwarding entities are hierarchical, the predecessor entities can deduce the traffic keys of the successor entities. In this scheme, the traffic key of each subgroup is randomly generated within the subgroup and reported to the trusted *CA* which then computes a parameter for any two predecessor and successor subgroups.

To manage user mobility, the FEDRP protocol is adopted by this scheme. When a member moves from the current subgroup to the new subgroup, the old manager does not immediately perform a rekeying procedure. If the entering member has previously visited the new subgroup, it receives the new subgroup traffic key through a unicast message. Otherwise, a new *TEK* is generated by the subgroup manager and reported to the *CA*. The parameters according to the obtained information are re-computed by the *CA*. Each manager maintains a table of members that hold the valid traffic key while residing outside the subgroup. The table is reset when a member holding a valid *TEK* leaves the group or the timer expires.

Pros: the communication overhead can be significantly reduced since each subgroup can deduce the *TEK*.

Cons: the solution is infeasible for large group members. Rely on the trusted third party authority for computing keying materials remains a single point of failure problem.

2.4.12 HKMS (N. C. Wang & Fang, 2007)

A hierarchical key management scheme (HKMS) for secure group communication in MANET has been proposed. This scheme manages members efficiently and reduces the amount of rekeying. A two layers' structure organizes the key management scheme. The subgroup in level 1 is composed of all the nodes in the subgroup. The subgroup found in level 2 is organized based on the location information of the node in Subgroup 1.

In each subgroup L1, the node with the most weight value is selected as the Level 1 cluster head (Dhurandher & Singh, 2005). Subsequently, the node with the most weight value in each Level 2 will be the Level 2 cluster head. Upon L1 head receives all the information for the nodes, it generates the L1-subgroup key using RSA. Next, the L1 head delivers this key to all the nodes in the subgroup. Since subgroup L1 is divided into a number of L2 subgroups, the head of each L2 subgroup generates a L2GK key and distributes it to the nodes residing inside its subgroup.

When a new member wishes to join a subgroup, it sends a request message to the neighboring nodes. The neighboring nodes inform the L2 head. Subsequently, the L2 head sends a reply message to the new member. The GKL2 is updated and distributed among the members of the subgroup. When an ordinary node leaves the subgroup, it informs the L2 subgroup head. The L2 head sends a reply message to the leaving node and regenerates a new L2GK key. When the heads of a subgroup leave the subgroup, new selection for choosing new head occur.

Pros: the hierarchical structure of the scheme reduce the cost of rekeying procedure. No prior knowledge and member serialization are required. The scheme requires to store 1 key at member level and keys associated with the number of L2 heads at L1 head. Double encryption of data transmission avoids the disclosure of group content by intermediate entities in the path flow.

Cons: the lack of a specific protocol for handling host mobility result in the communication overhead in movement event. The overhead of the scheme is still rather high as it uses unicast messages encrypted with public key of each member to update the keying materials upon any changes in group membership.

2.4.13 GKMM (Hernandez Serrano, Pegueroles, & Soriano, 2005)

A decentralized approach was adopted to establish secure group communication in MANET (Hernandez Serrano et al., 2005). Due to the lack of fixed infrastructure in MANET, which acts as an area key manager in wired networks, a weight based scheme was developed in order to select an area manager for each area. The weight parameters for each station include mobility, battery power level, and geographical position. The scheme is comprised of two phases; *AKM*s selection, and the generation and distribution of the session key. The stations organize a cluster, which is managed by a station called the "clusterhead". The maximum hop between ordinary stations and the clusterhead is equal one.

The *DKM* is selected via a leader election algorithm that elects an *AKM* with greatest weight. Instead of choosing a *DKM*, a key agreement such as the one developed by (Michael Steiner et al., 1998) is used by the *AKM*s in order to generate a common traffic key. Group events including joining and leaving in each area is managed by the *LKH* (see Section 2) within each area, while node mobility is handled by protocols such as static rekeying (SR), delayed rekeying (DR), or immediate rekeying (IR).

Pros: Creating a virtual fixed backbone allows to make extend the existing group key management scheme to MANET. The algorithm adapts to the large ad-hoc secure group communication with a slightly mobile set of nodes.

Cons: The scheme handle secure group communication with low mobility scenario, so that it will not run efficiently for highly mobile groups of members. The clustering process and the election of a leader in each cluster imply a high computation overhead. The clusterheads can cause bottleneck and consequently compromised since all the key management and the inter-cluster are restricted to them.

2.4.14 BALADE (Bouassida et al., 2008)

BALADE is a decentralized scheme with the common *TEK*, which decomposes a group dynamically into a number of clusters. Each cluster is managed by a local manager. A common key is shared between the cluster manager and the members residing in the cluster. The source has two roles, as it acts as the group manager who is responsible for generating the *TEK*, and it acts as the sender of the encrypted multicast flow to the members. A session key *KEK* is shared between the source and the cluster managers, which is used for distributing the *TEK* securely to the cluster managers. The *TEK* is delivered to the members of each cluster by each cluster manager and it is protected under the cluster key. The *TEK* is updated at each data semantic depending on the application. For example, a source multicasting a song renews the *TEK* after every song.

If a member moves from its subgroup or cluster to the new subgroup or cluster, it is re-authenticated before it can join the multicast group. Each member possesses a re-authentication ticket, which is a password encrypted with the *TEK*, that confirms its identity to the manager of the new cluster in order to be verified and joined the group.

Pros: using optimized group communication cluster tree algorithm result in bandwidth and energy efficiency. Moreover, encrypting data traffic with a common *TEK* eliminates the overhead induced by encryption and decryption operation on the group communication.

Cons: the perfect forward and backward secrecy were not achieved as the updating keying materials triggered at each data semantic unit. The authors did not discuss how the cluster leaders are selected. Since the source of group communication is the global controller and responsible for managing the *TEK,* it leaves the vulnerability of single point of failure. The scheme incurred computation overhead at the key manager level. The performance efficiency of the scheme is still rather low and the reliability of the keys distribution process still remains challenges in deploying in ad hoc environments. The construction of key distribution tree based on the geographical location information requires reliable and trustworthy connectivity between nodes which is a hinder in actual environments.

2.4.15 LKH++ (Pietro et al., 2002)

LKH++ improves the performance of the LKH by exploiting the properties of both one way hash functions and information of users which have already been shared in the LKH model (Di Pietro, Mancini, Yee Wei, Etalle, & Havinga, 2003). The set of information shared in the LKH is used to generate new keys locally without the need for communication between users and the key server. Moreover, users can autonomously compute the keys from a certain point upward along the path to the root of LKH by applying a one-way function. The proposed scheme is suitable for wireless mobile environments since each member stores a number of keys that are equal to the logarithm of the number of members in the group. Additionally, this scheme reduces the number of multicast messages sent from the center, and employs symmetric cryptography key which results in reducing the length of the encryption key and the computational efforts required for encrypting and decrypting message by user devices, which saves battery power.

Pros: the modified logical key hierarchy tree scheme with applying one-way hash function showed efficiency in communication and computation overheads of joining and leaving process. The scheme is free of collusion attack.

Cons: the scheme did not discuss how to address host mobility issue in wireless mobile environments.

2.4.16 Discussion

In this section, the protocols in Section 2.4 are further discussed and compared against the following criteria in Table 2.2.

Design approach: a centralized or decentralized approach used by each proposal to design the group key management scheme in their solution. Most of the centralized schemes discussed in this study used the tree based key management. By contrary, the decentralized approach can be categorized as either a common *TEK* approach or an independent *TEK* per subgroup or area approach. In Table 2.2, the centralized approach is represented by *C* and the decentralized approach is represented by *D*.

Data transformation: required by the decentralized approach with independent *TEK* per subgroup or area. In such schemes, the data must be translated (decrypted and re-encrypted) at the edge of each area when it passes from one area to another one. The advantage of using such schemes is that they mitigate the *1-affect-n* phenomena since any changes in a subgroup will affect only the members residing in that specific subgroup.

Host mobility protocol: illustrates schemes that explicitly proposed a protocol for managing the movement of group members between areas. Some of the schemes, even though they extend group key management to the wireless mobile environment, do not propose any mechanism to address host mobility issue.

Security: concerned with host mobility protocols to achieve forward and backward secrecy. As discussed in Section 2.2.4, the rekeying process is needed to ensure that a new member who joins a group is denied access to the messages prior joining the group (i.e. backward secrecy). And the rekeying process is also needed to make sure that the previous messages cannot be accessed by a former member (i.e. forward secrecy). The provision of backward and forward secrecy should be extended during member's mobility (i.e. move event) as moving members may breach these security requirements. When a moving member transfers from one area to another, it may access to the information and keying materials prior to its membership. The rekeying process is performed in the visited area to ensure that the backward secrecy is maintained during move event. In contrary, the provision of forward secrecy in the old area is not necessary since the moving member is still in the same session.

On the other hand, a moving member is able to accumulate information about the local security services for each area it visits. When the moving member leaves the group, he may still possess valid keys associated with the areas, where it has previously visited. Thus, the rekeying process must be performed not only in the area where the leave event occurs, but also in the areas where the leaving member has already visited, and having valid keying materials of the area (s). This rekeying process is necessary to be performed throughout the group during leave event in order to ensure the forward secrecy is achieved.

Strict rekeying management: schemes that update immediately the keying materials in response to any changes in group membership while other schemes adopt the concept of delaying key updates to defeat *out-of-sync* problem between keys and data by updating keying materials at the end of a period of time.

Using a list for host mobility: maintained by each area manager to keep track the moving member who still have valid keys from the areas they visited.

Number of messages: the total messages generated by the key managers in order to achieve forward and backward secrecy respectively in the old area and the new area as well as deliver the keying materials to the mobile member. The total number of members residing in area *i* and the total number of areas in a domain are indicated by n_i and $|A|$, respectively in Table 2.2. The *u* and *m* are used to represent unicast and multicast messages, respectively.

Except for LKH++ (Pietro et al., 2002), and KTMM (Jong-Hyuk & Kyoon-Ha, 2006) that follow the centralized approach by using the logical key hierarchy scheme, other key management schemes exploit the decentralized approach by decomposing group communication into several subgroups or area. As shown in Table 2.2, some decentralized schemes translate data when it is transferred from one subgroup to another because different *TEK* is used in each subgroup.

BALADE (Bouassida et al., 2008), and M-Iolus (Kamat et al., 2003) use batch rekey techniques so that the keying materials are updated at a certain time to improve efficiency and to prevent the ping-pong effect that might appear when a user at the boundary of two subgroups frequently cross over from one subgroup to another one. Despite improving efficiency, batch rekeying can increase the time it takes to access the data since a new member must wait longer to join the group communication. Additionally, batch rekeying can breach perfect forward secrecy as the departed member will still have access to the group communication after leaving, at least until the TEK is updated. In other words, perfect forward and backward secrecy is not achievable, and must be avoided when dealing with critical applications.

Table 2.2: Comparison of explored group key management schemes with host mobility protocol.

Scheme	Approach Design	Data transformation	Strict rekeying management	Host mobility protocol	Using list for host mobility management	Security		No. of messages for host mobility
						Forward secrecy in the old area	Backward secrecy in the new area	
KMGM (Said Gharout et al., 2012)	D	Yes	Yes	Yes	Yes	Yes	No	≤ 1
GKMW (Mat Kiah & Martin, 2007)	D	No	Yes	Yes	Yes	No	Yes	≤ 4
HKMS (N. C. Wang & Fang, 2007)	D	Yes	Yes	No	No	Yes	Yes	$n_i u + 1m$
TMKM (Sun et al., 2004)	D	No	Yes	Yes	Yes	Yes	Yes	$2Im + (I + 1)u$
CDKM (Min-Ho et al., 2010)	D	No	Yes	Yes	No	Yes	Yes	$\leq \log n_i + 1$
HSK (S. K. S. Gupta & Cherukuri, 2003)	D	Yes	Yes	No	No	Yes	Yes	…
BR (DeCleene et al., 2001)	D	No	Yes	No	No	Yes	Yes	$(n_i + 1)u + (\|A\| - 1)m$
IR (DeCleene et al., 2001)	D	No	Yes	Yes	No	Yes	Yes	$n_i u + 1m$
FEDRP (DeCleene et al., 2001)	D	No	Yes	Yes	Yes	Yes	Yes	≤ 2
GKMM (Hernandez Serrano et al., 2005)	D	No	Yes	Yes	No	Yes	Yes	$\leq (n_i + 1)u + (\|A\| + 1)m$
LKH++ (Pietro et al., 2002)	C	No	Yes	No	No	Yes	Yes	…
BALADE(Bouassida et al., 2008)	D	No	No	Yes	No	N/A	N/A	N/A
KTMM (Jong-Hyuk & Kyoon-Ha, 2006)	C	No	Yes	Yes	No	Yes	No	$< \log n_i + 1$
WSMM (Jong-Hyuk & Kyoon-Ha, 2006)	D	Yes	Yes	No	No	Yes	Yes	$n_i + 2$
M-IOLUS (Kamat et al., 2003)	D	Yes	No	Yes	Yes	Yes	No	1
SHKM (Cao et al., 2006)	D	Yes	Yes	Yes	Yes	Yes	Yes	≤ 2

M-Iolus (Kamat et al., 2003) and KMGM (Said Gharout et al., 2012) assume that the moving member has already authenticated their identity with the manager of the area where the member joined the group and thus remains a valid and authentic member of the session even though they changed areas. Base on this assumption, user mobility is treated

with the less rekeying process. The expense of such treatment is a backward secrecy violation since the moving members might be able to have access to the communication that occurred before they joined the group in visited area. On the other hand, GKMW (Mat Kiah & Martin, 2007) protocol suffers from a breach of forward secrecy as the keying materials from each area visited by a leaving member are not updated when the mobile member leaves the group communication.

Table 2.2 shows that the key managers in KMGM (Said Gharout et al., 2012), GKMW (Mat Kiah & Martin, 2007), TMKM (Sun et al., 2004), FEDRP (DeCleene et al., 2001), M-Iolus (Kamat et al., 2003) , and SHKM (Cao et al., 2006) employ a secure list to handle the mobility of members of highly dynamic groups. This can be useful for avoiding frequent key updates in the areas affected by the moving member and to facilitate tracking keying materials during movement. In doing so, the key managers update or empty this list whenever an existing mobile member who has previously visited their area leaves the group. Maintaining a list can become complex in very big and dynamic groups where frequent movements result in increasing the size of the list.

Although LKH++, and HSK (S. K. S. Gupta & Cherukuri, 2003) were primarily designed for the wireless mobile environment, they did not propose any explicit protocol for mobility (i.e. move event). While these solutions provided some mechanisms for managing keying materials during join or leave event, they did not address the management of keying materials during move event. Therefore, member mobility imposes further overheads since movement is treated as a leave in the old area and consecutively as a join in the new area. KMGM, M-Iolus, and BALADE showed the least rekeying cost when transferring a mobile member from one area to another one in compared to other protocols. However, other disadvantages are shared by these schemes since backward secrecy can be breached in the visited areas. Moreover, BALADE does

not indicate how it manages the keying materials in visited areas in order to deliver them to the moving user. Therefore, there exists an additional implicit communication cost arising as a result of handling mobility rekey.

2.5 Chapter summary

This chapter has looked at the different approaches in terms of centralized, decentralized, and distributed used in designing group key management schemes. The concepts of each approach have been defined and several existing schemes associated with each approach have been explored. Several requirements which are necessary to be considered during the design of a group key management have been presented and explained, then used to compare the different design approaches. Furthermore, the wireless mobile networks in terms of infrastructure based and infrastructure less have been illustrated, and then their characteristics have been investigated in order to highlight the design challenges of group key management scheme in such environments. Afterward, several group key management designed for wireless mobile environments have been scrutinized and critically analyzed against a number of identified criteria to identify the weaknesses that need to be addressed. The perspective gained from this chapter will be used to influence the proposed scheme in this work.

CHAPTER 3: RESEARCH METHODOLOGY

Research methodology is a set of procedure, schemes, and algorithms used to carry out a research. It includes theoretical procedure, experimental studies, numerical schemes, and/or statistical approaches. Thus, a researcher requires to design a methodology for the research problem.

This chapter defines the details of processes which are used and incorporated for carrying out this research in order to achieve an efficient group key management scheme taking into account the mobility of hosts in wireless mobile environment. The processes involved in the research methodology include the study, design, implementation, and analysis activities. Figure 3.1 illustrates flow of research methodology in this work.

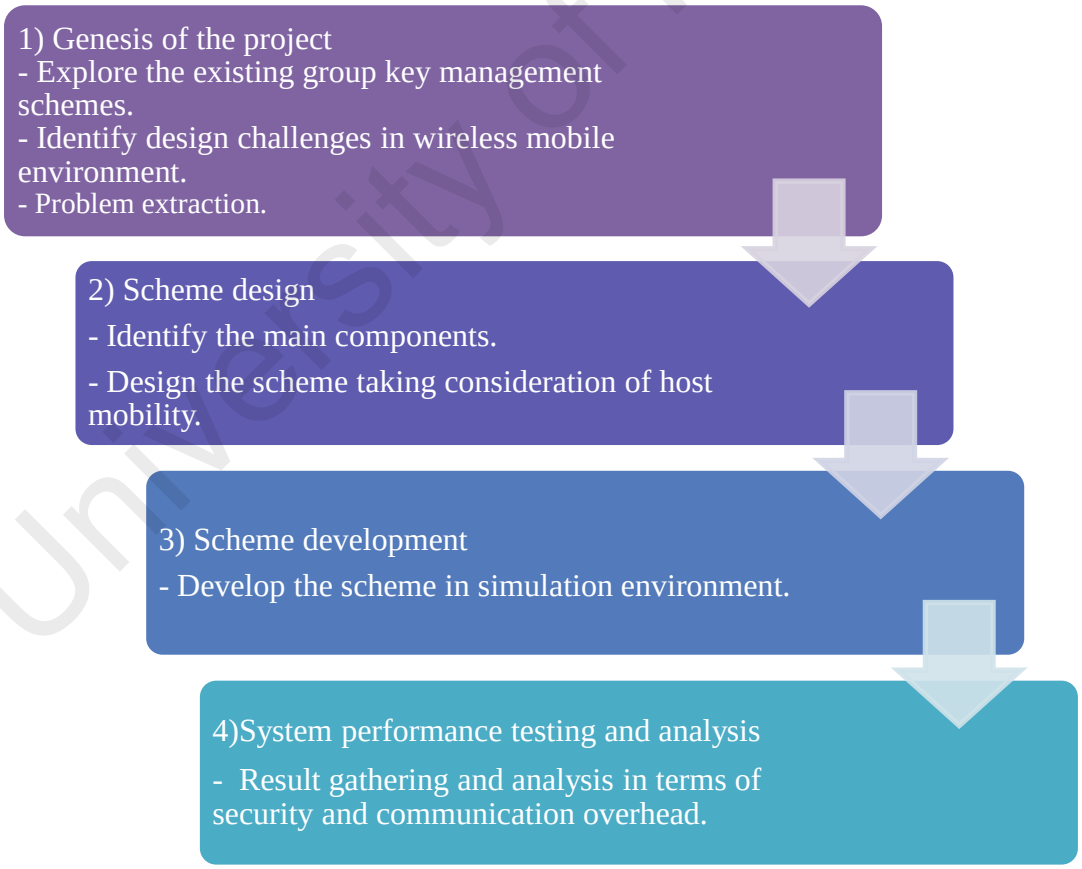

Figure 3.1: Research methodology flow.

3.1 Literature review and problem extraction

Preparing the review of literature and exploring work background in the area of secure group communication is a preliminary step before attempting to design a group key management scheme. This step leads to sharpening or reformulating the problem of deploying secure group communications in wireless mobile environments. Background exploration makes a way to acquire proper theoretical and practical knowledge for investigating the member mobility issue in secure group communication, to understand what external knowledge factors have not been examined, and how the existing solution can be exploited in this work.

The focus of this research study is on the design of a group key management scheme taking into consideration of group members move for wireless mobile environments.

- Initially, the existing key management schemes designed for secure group communication are studied and organized under three categories namely, (1) centralized approach, (2) distributed approach and (3) decentralized approach.
- Afterward, the nature of wireless networks is investigated in terms of one hop and multiple hops. The limitations of wireless mobile devices are explored to highlight the primary constraints and critical factors which must be considered in designing a secure group communication in wireless mobile environment.
- The selected group key management schemes designed for wireless mobile environments are critically analyzed to identify the weaknesses that need to be addressed.

3.2 Group key management scheme design

This stage creates the foundation of the entire research study, and describes a new system including different modules and subsystems. System design identifies functions and operation in details, including business rules, process diagrams and other

documentation. It specifies the various approaches being employed in solving the research problem, as well as source and information related to the problem.

In this phase, design of group key management scheme is carried out by keeping in view the resource limitation of mobile devices and characteristics of wireless networks so that achieve backward and forward secrecy in all events with minimum communication cost and least resource utilization on mobile device (as in Chapter 4). The specifications of the scheme such as various features and operation definitions are described in details (see Section 4.3). Some set of interaction sequence diagrams are used to model the behavior and interaction of components involved in the group key management scheme (see Section 4.4).

Different events in the group key management scheme including join event, leave event, and mobility event as well as updating keying materials operation are modeled by interaction diagrams (see Section 4.5). These diagrams show the flow of messages between main entities involved in a secure group communication while attempting to achieve backward and forward secrecy. The entire design and data model are documented so that it could be used in implementation phase.

3.3 Prototype implementation

In field of computer science, simulation can be used as a means to evaluate a system, a protocol or an algorithm. Simulation provides designing a model of an actual or theoretical system, executing the model (an experiment) on a digital computer instead of using real environment, as well as statistically analyzing the execution. This property of simulation program result in relative simplicity usage and wide applicability.

Secure **G**roup **C**ommunication **S**imulator called hereafter SGCSim is developed to simulate HIMOB and analyze its performance. The implementation of the simulation

software SGCSim is carried out using C# as a readily available programming language in order to gain a total control over developing phase of HIMOB and its related protocols. However, the model construction takes considerable time during the implementation. The main entities including key managers such as *DKM* and *AKM*s and group members along with key management protocols such as new member joining protocol, member mobility protocol, and existing member leaving protocol are developed in SGCSim for the proposed scheme. The members can send their request to the corresponding area key managers in order to join or leave the session and move between areas. The key managers receive the request of members and manage the keying materials accordingly. OpenSSL (Young & Hudson, 2015) is used as a cryptography library for generating and managing keying materials.

OPNET modeler (OPNET Technologies Inc, 2014), NS2 (Bajaj et al., 1999), and OmNetpp (Varga & Hornig, 2008) were examined for use as well, but were finally excluded because they provided more features and options than were necessary in this simulation, which cause the scenario model being too complex to model.

Although NS2 is well known as a free license and simply open source simulation software, it needs recompilation whenever a change done in the user code. OmNetpp suffer from lack of a great variety of protocols, and poor analysis and performance management. Opnet as a commercial license and proprietary software is limited in terms of customizability. More information in terms of performance comparison between the aforementioned simulators have been provided in (Weingartner, vom Lehn, & Wehrle, 2009). Table 3.1 gives a specification comparison between SGCSim and the other explored simulators such as Opnet, NS2, and OmNetpp.

Network simulators NS2, OmNetpp, and Opnet use C++ as the efficient programming language to model the behavior of the simulation nodes. NS2 also use oTcl scripts to

specify the network topology and control the schedule of events. In NS2, if any changes are made by the user in the simulation scenario, the code must be recompiled. The mixture of compilation and interpretation make NS2 difficult to analyze and to understand the code. NS2 covers almost variants of TCP such as wired networking, multicast protocols, and several ad-hoc routing protocols. Nevertheless, modeling a real system in this simulator is too complex.

Table 3.1: Simulators comparison based on properties.

Name	Language	Available module	Scalability	Number of nodes	GUI	Ease of use
NS2	C++ and OTCL	Wired, Wireless, Ad-hoc and WSN	Limited	Up to 3000	Limited	Hard
OmNetpp	C++	Wired, Wireless, Ad-hoc and WSN	Moderate	Not limited	Yes	Moderate
OPNET	C and C++	Wired, Wireless, Ad-hoc and WSN	Large	Up to 300	Yes	Hard
SGCSim	C#	Wireless, group communication, Key management	Moderate	Not limited	Yes	Easy

OmNetpp provides a powerful graphical user interface environment, which facilitates users for having easier tracing and debugging in the simulator. It supports Internet protocols, wired networks, wireless ad-hoc networks, and sensor networks. However, it suffers from fairly incomplete support for a great variety of protocols in wired and wireless networks. Similarly, OPNET offers a powerful graphical editor interface, which enable users to build all kinds of networks topology and entities from the application layer to the physical layer. However, the graphical user interface operation is very complex. OPNET can efficiently handle the simulation of a complex network with a big number of devices and traffic flow.

A primary aim of the SGCSim design is to ensure it works modular, and has forward thinking design. A network simulation may consist of different plug-ins which allow to run and test any number of current or future protocols. SGCSim separates different layers of the network such as the protocol layer, the network layer, and the scenario layer, which result in the layers being changed or modified independently without affecting the others. The relevant components of wireless networks are simulated for the purpose of this experiment. However, unnecessary components or modules not required for testing the proposed key management scheme such as a full TCP/IP stack, wireless radio constraints, and unnecessary network communication are eliminated. The focus of SGCSim is to deploy a secure group communication while developing the key management protocols for handling different events such as join, leave and move.

3.3.1 Secure group communication simulator (SGCSim)

The primary version of SGCSim consisted of hundred lines of code in one file, which creates a topology and display it in a visual format as shown in Figure 3.2. However, it has gradually grown up so that it deploys HIMOB, and other proposals such as FEDRP (DeCleene et al., 2001), KMGM (Said Gharout et al., 2012), GKMW(Mat Kiah & Martin, 2007), and LKH++ (Pietro et al., 2002) over a wireless network topology. The network topology is comprised of a X and Y grid. The values of grid are determined by the user at run time. This grid is then divided into number of areas *A*, determined by the user at run time, at random locations throughout with equal probability. Each area coverage zone is also identified by the area radius value at run time. The area radius can be identified based on the radio coverage of each entity in the selected network environment. The user is also able to capture the case where the wireless network is homogenous or heterogeneous.

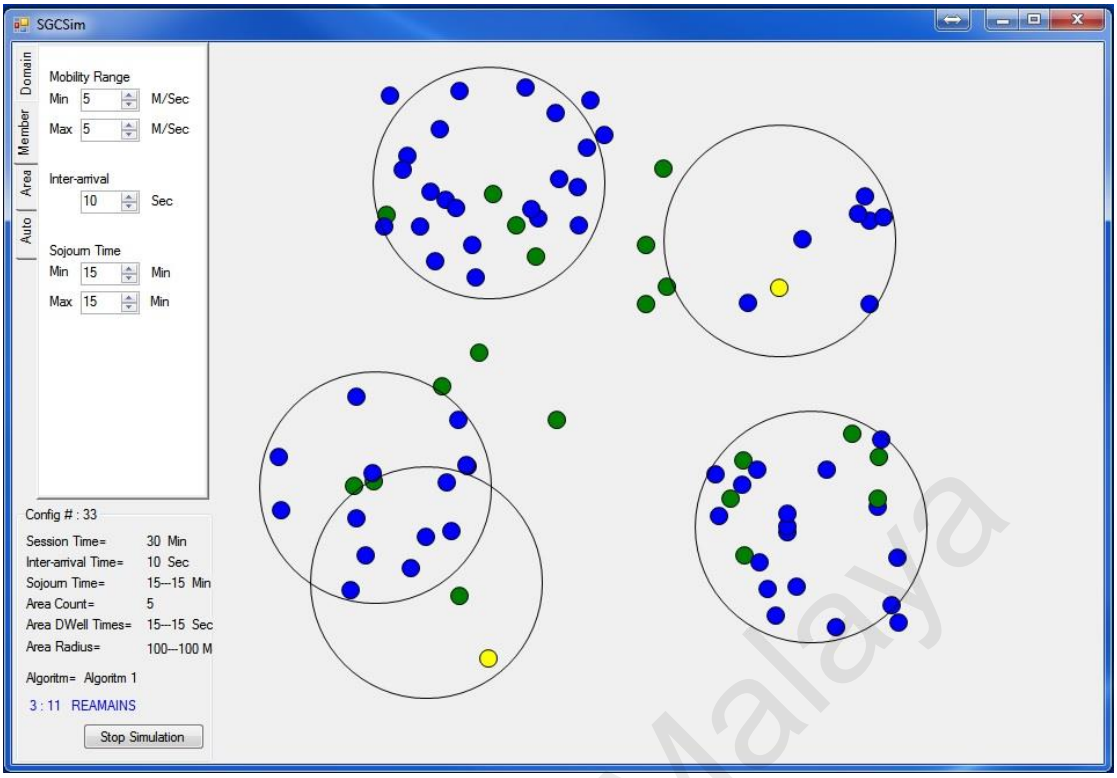

Figure 3.2: Secure Group Communication Simulation (SGCSim)

Each area is randomly selected with a uniform distribution to be populated with group members. Members arrive to each area according to the inter arrival time with rate λ (Almeroth & Ammar, 1996, 1997), determined by the user at run time. In other words, when the specified inter arrival time value expires, a member is automatically added to the group and also located at a random area. The inter arrival time is the time T_i between two successive members M_i joining the group as shown in Figure 3.3. Therefore, the group population is depending on the inter-arrival time. Changing the time value for inter-arrival result in varying the population of members in the group. If the small amount is considered for the inter arrival time value, the group will have a big size with a great number of members at the end of simulator execution and vice versa. This property of the simulation enables a user to evaluate the impact of group size variation on scalability of a group key management scheme.

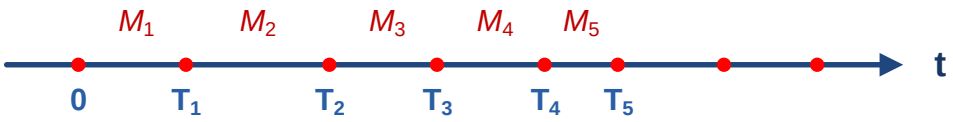

Figure 3.3: The inter arrival and arrival time.

Members are able to freely move between areas according to the random way point model (Hyytiä & Virtamo, 2007) in this simulation. Future work, of course, could entail further mobility models (Aschenbruck, Gerhards-Padilla, & Martini, 2008) but the focus here was successfully implementing a mobility model in this simulation. In mobility event, each member or node choose a random destination (i.e. area) with the same probability. The member moves toward the destination with the velocity rate determined at the simulation run. The velocity rate can be either a constant value or varied between a ranges of values. The speed of mobile node can be distinguished by foot speed between 1-10 km/h, urban vehicle speed between 20-80 km/h, and highway speed between 50-120 km/h.

Finally, the user can select the session duration, the average session sojourn time (also called membership duration herein), and the average dwell time in each area at run time. The session duration reflects the duration of a group communication session, which is proportional with the simulation run time. The average sojourn time (i.e. membership duration) is equal to the period of time that a member is allowed to remain in a session. The period of time that a member remains in a given area is the area dwell time. When the dwell time expires, the member transits to another area.

The topology of the simulation is developed using standard C# library. Random generator class of C# was used to ensure network topologies randomly created. The random generator seed is an input parameter that feeds up the random generator class in order to generate a random result each time a simulation is run. The random values

produced by the random generator class is used with different modules of the network simulation. First, a random value is employed to determine the location of an area on the grid. As a result of random placement of the areas, some overlapping occurs between some areas as it may happen in real world. The location of each node on each area is also determined by the result of random generator class. The node is place in the identified dimension x, y, if the spot was not already occupied by a previous node. New nodes are only place within coverage range of existing areas. Finally, the destination area in the move cases are chosen randomly and assigned to each member to ensure random distribution of nodes across the areas created in a topology in a given experiment.

3.4 Results and analysis

The proposed scheme is evaluated on the basis of rekeying messages overhead as well as *1-affects-n* phenomenon while performing in different group sizes. Two parameters control population size of a group in the simulation namely, 1) the average of inter arrival into the group and 2) the average membership duration (i.e. session sojourn time). Variation on inter arrival time value and session sojourn time (i.e. membership duration) lead to increasing the size of a group.

The mobility rate is another interested factor used to evaluate its impact on the schemes in terms of number of rekeying messages generated for updating keying materials as well as the number of affected members in the session. The mobility rate varies by changing the value of the area dwell time. Decreasing the area dwell time results in increasing the mobility rate and establishing more dynamic environment for experiment.

The rekeying messages overhead and *1-affects-n* phenomenon are evaluated as follows:

Rekeying messages overhead: the total messages distributed to update the keying materials whenever any changes occur in the session in order to preserve backward and forward secrecy. There is no discrimination between unicast and multicast messages. The average number of rekeying messages per events is calculated as follows:

$$Average\ rekying\ messages = \frac{total\ number\ of\ rekeying\ messages}{total\ number\ of\ events} \quad (3.1)$$

1-affects-n phenomenon: the total number of members are affected by every event in the session. These group members need to replace their old keying materials with the new one via receiving controlling messages from key managers. The average number of affected members is calculated as follows:

$$Average\ number\ of\ affected\ member = \frac{Total\ number\ of\ affected\ member}{total\ number\ of\ events} \quad (3.2)$$

The impact of security in terms of backward secrecy and forward secrecy are analyzed when every event such as join, move, and leave occur in the secure group communication. The provision of entity authentication is discussed for the move event when a moving member changes its location.

3.5 Chapter summary

This chapter has presented the adopted research methods for carrying out this research study. This chapter has provided the detail information of the tools and techniques used in developing a secure group communication in wireless mobile environments. Furthermore, the comparison parameters for evaluation of HIMOB has identified in the chapter. Next chapter looks at the constraints and design challenges of a secure group communication in wireless mobile environments.

CHAPTER 4: DESIGN OF GROUP KEY MANAGEMENT SCHEME FOR WIRELESS MOBILE ENVIRONMENTS

The design of the group key management scheme taking into consideration member mobility and the corresponding protocols is specified in this chapter.

The chapter begins with an introduction to the notation used in the protocol designs in Section 4.1. The scope of the proposal is presented in Section 4.2, which represent the boundary aspects of the work. In Section 4.3, the reference model is explored to identify the main properties and design of the scheme. The main architecture of the proposal and the main functionalities of the protocol designs are described respectively in Section 4.4 and Section 4.5. Finally, a scenario is used to demonstrate various operations of the solution in detail in Section 4.6.

4.1 Notation

The nomenclature used for describing the proposed protocols is summarized in Table 4.1 and Table 4.2. Table 4.1 illustrates the notations used for describing entities. Table 4.2 presents the nomenclatures for operatives necessary for group key management operations.

Table 4.1: Summary of the notations used for describing the proposed protocols.

Symbol	Signification	Role
DKM	Domain Key Manager	This entity is responsible for key management throughout the domain.
AKM	Area Key Manager	This entity is responsible for key management in an area. All these entities are under control of domain key manager.
AKM_i	Area Key Manager *i*	Area key manager that governs keying materials in area *i*.
a_i	Area *i*	A set of group members using the same auxiliary key.

Symbol	Signification	Role
M	Member	Member of group communication who can be either a sender or receiver of the group.
M_i	Member in area *i*	A set of group members that are residing in area *i*.
ID_{M_i}	Identity of Member *i*	Used by the AKM_i to generate M_iEK and to track M_i.
ID_{A_i}	Identity of area *i*	Used by M_i and the *DKM* to respectively send requests and controlling messages.
ID_G	Identity of the group communication *G*	Used by the *DKM* and the AKM_i to identify the particular group M_i joined.
n	Number of group members	The total number of group members that are in the session.
n_{a_i}	Number of member in area *i*	The total number of members that stay in area *i*.
$MemL_i$	List of current members	This contains the list of current members residing in area a_i.
$AMOL_i$	List of mobile members	This contains the list of mobile members which previously left the area *i* and moved to other areas.

Table 4.2: Summary of operatives.

Symbol	Signification	Role
$\{m\}_k$	Message encryption	Message (or data) *m* is encrypted with a symmetric key *k*.
||	Concatenation operator	Concatenates different fields of a message.
$Text$	A message field	It is a field of a message that may contain optional information.
$a \rightarrow b$	Unicast transmission	Delivering a message from entity *a* to entity *b* using unicast communication.
$a => x$	Multicast transmission	Disseminating a message from entity *a* to a group of members *x* using multicast communication.

4.2 Scope of proposal

The scope of the proposed group key management scheme specification needs to be noted as follows:

- **Infrastructure based environments.** Wireless networks are divided into two categories (1) infrastructure-based and (2) infrastructure-less as discussed in Chapter 4. HIMOB in this research relies on infrastructure-based environments with a basic underlying cellular architecture (Crow et al., 1997; Rappaport, 2001; Taferner & Bonek, 2013). It is not intended in this phase to extend its usage to infrastructure-less environments such as wireless Ad-Hoc networks (Chlamtac et al., 2003), wireless sensor networks (Yick et al., 2008), or vehicular ad hoc networks (Al-Sultan, Al-Doori, Al-Bayatti, & Zedan, 2014). Future work, of course, could entail further refinements of this scheme to apply to infrastructure-less environments.
- **Reliable transport of rekey message.** Keying materials are typically sent via multicast messages for efficiency when any changes occur in the group membership. It is the responsibility of key manager entities to ensure that all members have received the current information security and keying materials. Therefore, the group key managers need to use a reliable transport mechanism to distribute rekey messages. Reliable multicasting such as (Floyd, Jacobson, Liu, McCanne, & Zhang, 1997; Kobayashi, Nakayama, Ansari, & Kato, 2009; Srinivas & Lu, 2009; C. Wang, Li, Han, & Ma, 2009) are assumed to be in place in order to provide some level of reliability added to the key delivery mechanism.
- **Group key management policy.** The proposal focuses solely on the entity and functions relating to the generation and management of cryptographic keys for purpose of providing a secure group communication. It is not the aim of the

proposal to specify the details of entity and functions used to create and manage security policies specific to a group communication as well as rules needed to govern the behavior of entities, group initialization, membership changes and emergency situations.

- **Application types and requirements.** Group applications can be organized as one-to-many or many-to-many, depending on the number of sources sending data traffic to many receivers in the group communication. This research is primarily concerned with key management aspects, and is not concerned with the data communication. Thus, the type of group application in place does not influence the scheme design. In other words, the scheme is independent from the type of application in place and does not impose any restriction on it.
- **Key management aspects.** Different aspects of key management consist of key generation, key distributions, key updates, key storage, and the cryptographic keys disposal. The key management aspects of HIMOB are concerned with key generation, key updates and key distributions. However other aspects of key management have their own significance, they are not considered in detail. That is because they can be handled by generic techniques that are not particular to the group communications.

4.3 Reference model for group key management

Group key management architecture can be divided into centralized or decentralized approach. A reference model proposed by SMuG research group (Baugher et al., 2005) is depicted in Figure 4.1 to better understand functional elements, and interfaces of a group key management scheme. The reference scheme incorporates the main entities and functions relating to secure group communication, and depicts the interrelations among them.

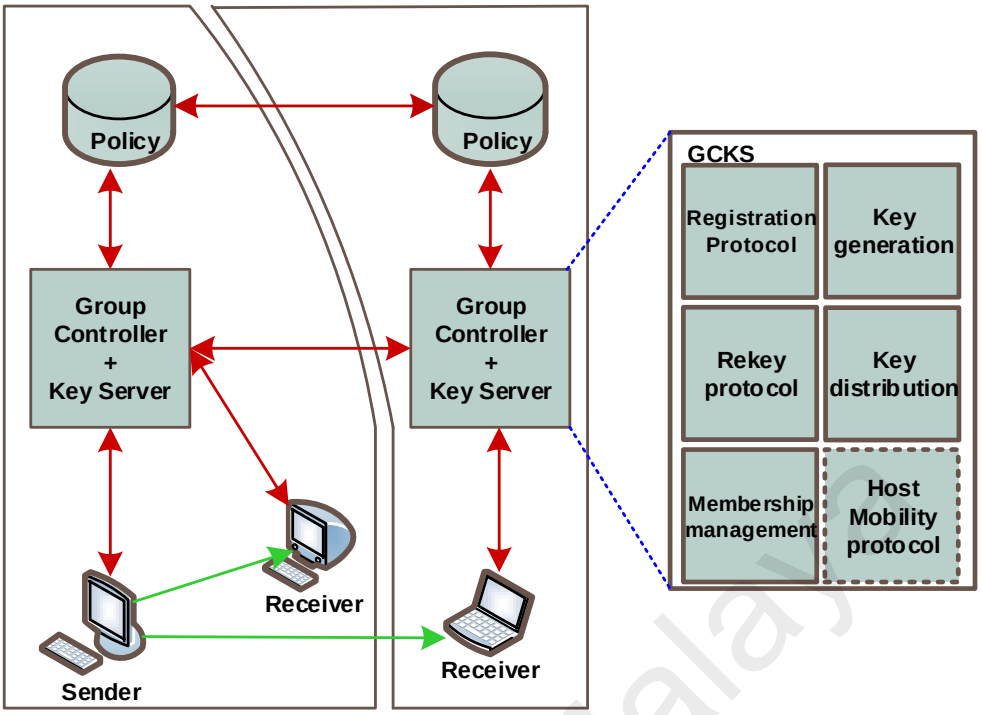

Figure 4.1: Reference model of group key management architecture.

The reference model consists of singular boxes that imply a corresponding entity implementing a given function. Each box can be implemented as one or more physical entities dependent on the particular solution in real environments. The box labeled "key server" is referred to as the function of key manager, that can be a server or a number of decentralized servers distributed in a domain.

4.3.1 Main components of reference model

The reference model diagram in Figure 4.1 contains boxes and arrows. The boxes are the functional entities that manage secure group communication. The arrows between the boxes represent interfaces, which consist of standard protocols to support the group communication between the main entities. The summary of the functional entities is as follows:

i. Group Controller and Key Server (*GCKS*): are indeed some servers, which are responsible for governing all group processes. In particular, they manage the

issuance of cryptographic keys used by a group communication. The key server also referred to as key distributor (KD) as it delivers the keying materials to the group members. The GCKS entity is allowed to interact with the other GCKS in regards to the key management to achieve scalability in decentralized architecture.

ii. Sender and receiver: are lower level entities which engage in the multicast communication. Each group communication involves at least one sender of data and one (or more) receiver(s) of data. Both sender and receiver need to interact with the GCKS regarding key management in order to obtain keying materials in accordance with key management policies, new keys upon updating keying materials, and other managing messages related to the keying materials, and security parameters.

4.3.2 Main protocols

In this section, required protocols for the provision of a secure communication among a group of wireless members are investigated. The term protocols describe the set of procedures, message exchanges, and message payloads that manage the behavior of the entities involved in a secure group such as key managers and group members.

4.3.2.1 Protocol for creating a new group

This protocol manages the creation of new multicast groups. This protocol is run by request of first member who is desired to establish a new multicast group. This protocol carries out the initial registration of new members to multicast group. The initial distribution of new cryptographic keys is done during the execution of this protocol.

4.3.2.2 Protocol for joining a group

This protocol governs new joins of group members to multicast communication. It also contains distribution of new cryptographic key to new members in order to enable them

to communicate with multicast group. In this protocol, new members must be prevented from accessing to previous data traffic or old group keys. So, all cryptographic keys are associated with that particular multicast group need to be rekeyed. All the members of that specific group would have new keys after conducting rekey.

4.3.2.3 Protocol for leaving from a group

This protocol manages existing member leaving from group communication. It unregisters the membership of leaving member from the multicast group. Departure members can be two types, voluntary and involuntary. Like the protocol for joining in a group communication, it is necessary that a leaving member is prevented from having access to future data traffic or new cryptographic keys within a multicast group. The remaining members in a multicast group after leaving occurrence need to be rekeyed in order to receive new cryptographic keys.

4.3.2.4 Protocol for rekeying within a group

This protocol governs the rekeying occurrence that appears due to group membership changes, periodic rekeying, expiration of cryptographic keys and compromised keys. In wireless mobile environment, the member movement from one area to another one result in rekeying. Each of these processes causes all group members gain new cryptographic keys which are required for secure group communication.

4.4 Main architecture

This scheme adopts a two tier hierarchical approach with the common traffic key for the group communication similar to (Thomas Hardjono et al., 2000; Mat Kiah & Martin, 2007) as shown in the Figure 4.2.

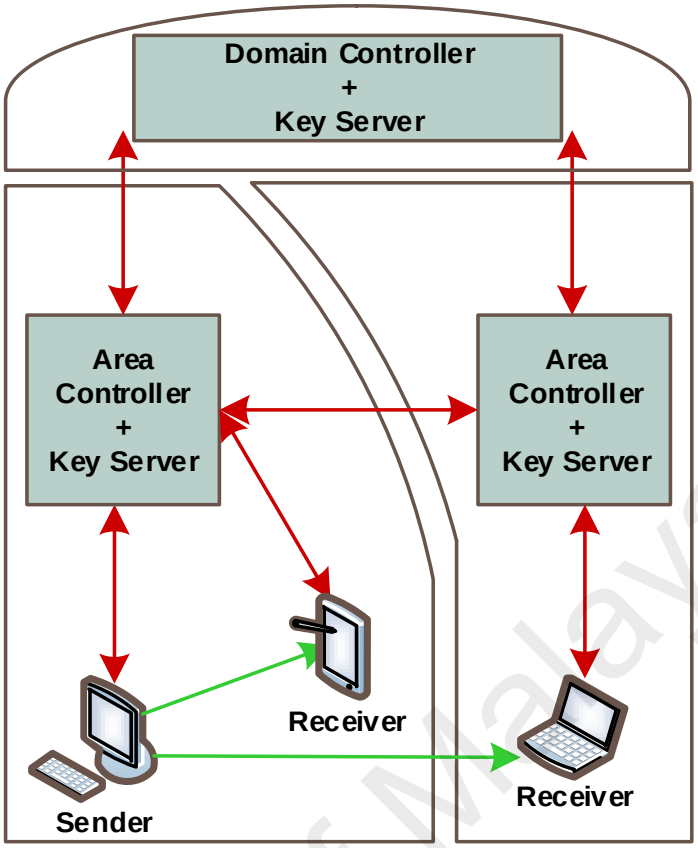

Figure 4.2: The architecture of HIMOB.

The first level is the domain level, which consists of the domain key manager (*DKM*) for initial key management and authentication procedure. The second level is the area level that is managed by area key managers (*AKM*) independently. The areas are indeed made by dividing the domain into a number of administratively scoped regions. Each area contains a set of members subscribed to diverse group communications. The members are allowed to freely move between distributed areas. The area can be aligned with the network topology, such that regions can be defined to be the size of subnets, autonomous systems, or larger. The aim of placing members in areas is to achieve flexible and efficient management, particularly when changes occur in the membership of a group due to join, move or leave. Therefore, the rekeying process is localized within the area, which alleviates the *1-affects-n* scalability problem.

The *1-affects-n* phenomenon occurs when an action of a member affects the entire group. A change in group membership requires to replace the old keying materials with the new one, and the whole group members may need to receive the new keying materials in order to maintain the group communication continuity. If the whole group members have to involve in rekeying process, it can become a scalability hurdle as some members may not receive the new keying materials on each event due to the network disruptions or network bandwidth limitation.

The *DKM* communicates to the *AKM*s either through secure one to one communications (i.e. unicast communications), or through the secure multicast communication. The administratively scoped multicast communications used for keying management are independent of other group communications for data, and exists even when there are no members belonged to any group communication in the domain. The keying materials in an area that has a member of group communication is managed by the *AKM* of that area. The *AKM* delivers the parameters for the group to host members residing in its area either through a secure one to one channel (i.e. unicast communication), or through a secure multicast channel.

The data traffic is encrypted with the domain wide cryptographic key, denoted by traffic key encryption *TEK*. The unique *TEK* is generated by the *DKM* and assigned for the group communication having a member in the domain (see Section 4.4.6.5). Each area has an associated auxiliary key called as area encryption key (*AEK*) to encrypt other keying materials or the group traffic key within the jurisdiction of the *AKM* (see Section 4.4.6.4).

4.4.1 Domain and Area(s)

The objective of the notion of domain and area(s) is to achieve a flexible and efficient key management for group communications. These notions are used to provide a means

to have hierarchical administratively scoped environments for group communications to take place.

Two hierarchy levels of regions corresponding to network entities and functions pertaining to group communication were introduced by (T. Hardjono & Cain, 1998), namely one "Inter-region" and one or more "Intra-region". The Intra-region may contain one or more group communication-capable entities as well as senders or receivers. Inter-region is the network backbone and covers the topology between all intra-regions. The term "domain" can be defined logically or physically as (pre-defined key management region whose scope is determined on a per case basis) a single Intra-region of a network which is administrated and controlled by a trusted entity operating under one system (Thomas Hardjono et al., 2000). Regions can be size of network subnets, autonomous systems, or larger, for instances the global system for mobile communication (GSM) operator's network (Friedhelm, 2002), or the Internet infrastructure as a collection of autonomous systems (AS), some being stub ASs and some transit ASs, connected to each other via Internet Service Providers (ISP). A trusted entity such as domain key manager (*DKM*) is responsible for managing the domain (see Section 4.4.2.1).

The domain is further divided into one or more small administratively scope areas (Meyer, 1998), each of which is managed by a trusted entity called area key manager (*AKM*) (see Section 4.4.2.2). *AKM*(s) closely contributes with the *DKM* in order to achieve an efficient key management. The host members of a group communication are located across these areas. Placing host members in areas leads to achievement of flexible and efficient key management particularly in the case of group membership changes due to join, leave, and move event.

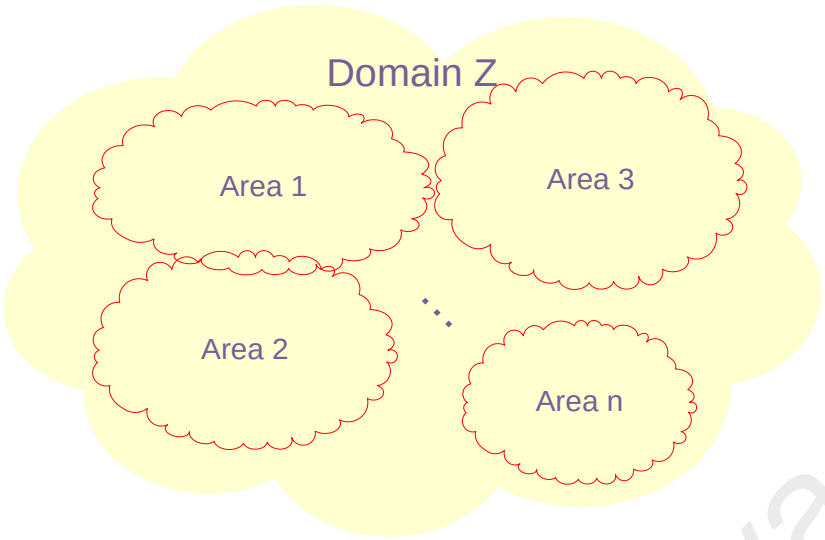

Figure 4.3: An example of Domain and Areas notion.

Figure 4.3 depicts the basic division along with a domain and one or more areas. Domain *Z* is divided into small manageable areas labeled *area 1* to *area n* which can physically or logically have overlapping each one to another. This division can be aligned with the network topology, so that the members are grouped based on their location in the network and formed several areas.

All corresponding entities across the domain are able to communicate with each other as the domain is controlled by one *DKM*. Although each area is unique and has its own security information, it should not preclude the mobile host who moves from its old area to the new visited area to obtain the security parameters associated with the new area. The term old area is referred to the area where the member is currently residing. The term visited area refer to area in a domain, where the member moves into during mobility event throughout the lifetime of its group membership.

4.4.2 Main entities and its functionalities

This section presents the main entities developed in the architecture. The main controlling entities in both domain and areas are defined as follows:

4.4.2.1 Domain Key Manager (*DKM*)

The *DKM* is defined as the key manager and group controller in a domain. There is only one *DKM* in a domain, and it is the main reference for security parameters for other key managers at the domain level. The main responsibility of the *DKM* is generating, distributing, updating, and storing keying material that may be required.

The *DKM* generates a new *TEK* whenever changes occur in the membership of a group in terms of joins or leaves. The new *TEK* is delivered to the host members by collaboration with other existing key managers at the domain level. The *DKM* is not involved in governing host mobility across the domain as this responsibility is delegated to the distributed area key managers in the domain.

The responsibilities of *DKM* are summarized as follows:

- main key manager of a Domain,
- collaborating with other key managers (i.e. area key managers) to provide a secure and efficient key management service,
- generation of the new *TEK* when a new member joins the group or an existing member leaves the group.
- updating and distributing keying materials when the rekeying process takes place during the lifetime of a group communication,
- managing group membership, security policies, and rekeying processes.

4.4.2.2 Area Key Manager (*AKM*)

An area key manager (*AKM*) is defined to exist at the area level which its main responsibility is to manage keying material across its corresponding area. Each area has only one *AKM*, which operates under authority of the *DKM*. An *AKM* that has a member of a group communication in its area associates it with one control group. All traffic

within the area is encrypted so that only the *AKM* associated with that area and the intended receivers can decrypt the traffic. The rekeying events occurred in an area are managed with the associated *AKM* to that area. The member moves events that may occur in the domain is handled by the *AKM* of the visited area without involving the *DKM*.

The main roles of an *AKM* having a member of a group communication in its corresponding area is outlined as follows:

- main key manager at area level,
- contributing with the *DKM* to provide secure and efficient key management service for group members at the area level,
- generation and distribution of cryptographic keys to all group members residing in an area,
- managing the rekeying events at the area level, operating under governance of the DKM,
- managing member move event that take place in its area.

4.4.2.3 Group member (*M*)

The entity who wishes to participate in a group communication is considered as group members. The member of a group can be either source(s) who sends a single copy of data to the group or receiver(s) who wishes to receive the data. Each member is located within an area at a given time.

4.4.3 Placement of entities

The core wired part of the network comprises the first tier which is the domain level consisting of Domain Key Manager for initial key management and authentication procedure. The wireless part of the network makes up the second tier that is considered as the area level consisting of multiple divisions, each of which is managed by an *AKM*

independently. Each area contains a set of members subscribed to the group communication. The group members are able to dynamically move across widely distributed areas.

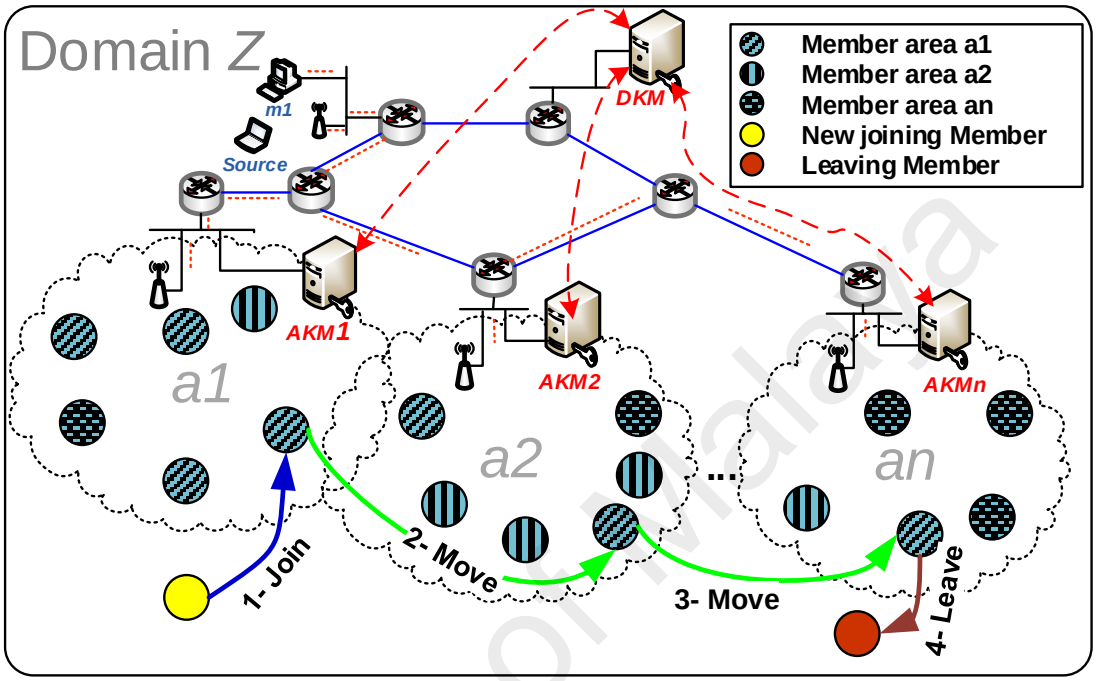

Figure 4.4: Placement of entities in domain *Z*.

Figure 4.4 illustrates the placement of entities of secure group communication in domain *Z*. From the illustration, the entity labeled *DKM* is the main key manager of domain *Z*, and the entities labeled AKM_i are the key manager of each area *i*. Each area consists of a number of group members who are allowed to perform handoff and change their locations.

In order to send controlling messages such as notification on rekeying that has occurred, some control channels represented by the dotted arrow lines in Figure 4.5, are established from *DKM* to *AKM* and members as well as from *AKM* to members. The double arrow lines illustrate the key exchange and the secure association management between peer entities such as *DKM* to *AKM* or *AKM* to members. The single arrow line

from senders to receivers shows the data channel for group traffic, which may be established after receiving security parameters by members in a specific multicast group.

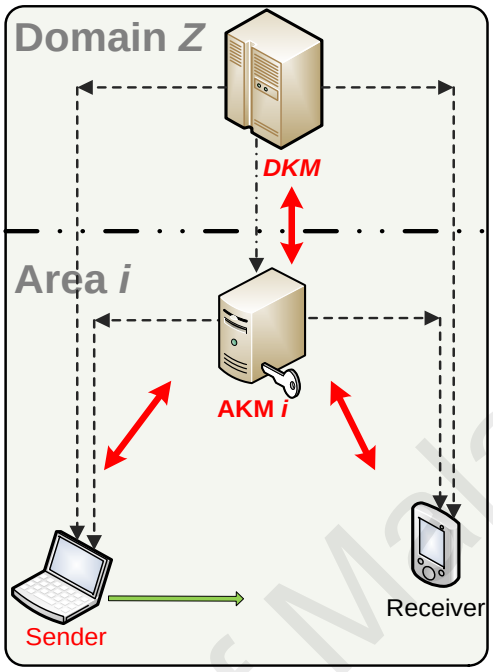

Figure 4.5: An example of main entities placement.

4.4.4 List management

The list management is an important concept, which is used in the design of this group key management scheme. The list management is maintained with the key managers in order to manage mobile members who have previously transited to another area as well as members who are residing within the area. More details of the lists regarding what their importance are, and how they are managed, particularly within members' moves are described as follows:

(a) ***KMOL***

An important concept used as part of the mobility protocol design is a managing list referred to as Key encryption key Mobile Owner List (KMOL). Each area key manager in a domain securely maintains its own *KMOL* and stores information of group members that already move from its managing area to another. Each time a member transits to a

new area, the following information associated with the mobile member is logged in the KMOL:

- the *ID* of the moving member,
- the *ID* of the group communication subscribed by the moving member,
- the *ID* of the visited area that the member is moving to.

The area key managers exploit the information logged in the *KMOL* to efficiently manage the mobile host members who may frequently move between areas while still remaining in the group session. They are able to keep track of moving members with the use of this list, which result in preventing from preforming extra rekeying process every time a member moves back into the area that it has recently visited. As the rekeying process needs to be performed whenever a member moves into an area, which may lead to disruption in a group communication, an area key manager uses this list to keep track of members who recently visited the area to prevent from performing frequent rekeying process. Furthermore, this list provides a means for each *AKM* to monitor the highly dynamic members who may accumulate the keying materials when they move between areas.

When a member enters an area, the *AKM* of the visited area can determine whether the member is a returning member who is just moving back into the area or is a new visiting member by looking up its *KMOL*. In a case that the member is moving back into the area, the area key manager skips to perform the rekeying process. Therefore, the scalability problem *1-affects-n* phenomenon is mitigated as the residing members in the visited area are not affected with the moving back members.

(b) ***MemL***

The current member lists referred to as *MemL* are maintained by key managers (*DKM*, and *AKM*s) contained the information of current members residing in the domain or the

area. Each time a member joins or leaves the group, or moves between areas, the following information associated with the member are captured into *MemL*.

- The *ID* of the member,
- The *ID* of the group communication that the member has subscribed,
- The *ID* of the area that the member is moving from,
- The *ID* of the area from which the members leaves the group.

The information in *MemL* is used by key managers in a domain in order to handle refreshing locally the keying materials within the area upon any changes occur in the group membership. The domain key manager uses this list as future reference to identify each member joined from what area and what was the reason to leave the session.

4.4.5 Trust relationships

The group key managers are known as the main key distributors in this design, thus the trust relationship issues often turn around them. There may exist the issue of trust for a given *AKM* or a given group member to the domain key manager as the primary security reference point. Then there is the issue of area level trust, the members located in a given area trust the *AKM* of that specific area more compared to one that is located in another area. Therefore, trust relationship among the entities is an important factor. In this scheme, all key managers (*DKM*, and *AKM*s) in a domain are assumed to be trustworthy and reliable. All group members can trust to these key managers to acquire secure group key management services. Two levels of trust relationships are defined as follows:

- Domain level. All *AKM*s trust *DKM* as the primary reference point for security parameters at the domain level for various group communications operating in that domain.

- Area level. All members residing in an area trust the *AKM* as the main key distributor at area level. The level of trust relationships given to the key manager of the visited area is equal to the trust relationship level given to the key manager of the area where the member joined the group.

4.4.6 Arrangement of keys in the domain

Most mobile devices exhibit special restrictions in terms of communication bandwidth and computation processing power. Thus, the symmetric cryptography approach benefits the scheme offering computationally faster and less complex techniques as well as minimizing the exchange of messages required to initiate the keying materials (Lenstra & Verheul, 2001) (Buchmann, 2013). As a result, the scheme employs the symmetric cryptography approach because of the exhibited characteristics.

There are five types of cryptography keys in this scheme such as *Domain-Area* Encryption key, *Domain Encryption Key*, *Member Encryption Key*, *Area Encryption Key* and *Traffic Encryption Key*, which are used to encrypt data traffic of group communication as well as to securely deliver the keying materials to the members. The details of all aforementioned cryptography keys as one of the fundamental components of HIMOB are described as follows.

4.4.6.1 Domain-Area Encryption Key (DAK_i)

The Domain-Area Encryption key denoted by DAK_i is a unique key shared between the *DKM* and a specific *AKM*. More precisely, the symmetric key DAK_i corresponds to *DKM* and the area key manager AKM_i of area i. Every *AKM* in the domain invokes the *DKM* to establish this key prior to commencement of any group communication in the domain. The *DKM* generates and sends each key to corresponding *AKM* by an appropriate secure means such as a secure association like SSL (Freier, Karlton, & Kocher, 1996) or

TLS (Dierks & Rescorla, 2008). The function of this key is limited only to unicast communication to transfer encrypted messages between the *DKM* and a particular *AKM*.

4.4.6.2 Domain Encryption Key (*DEK*)

Domain Encryption Key denoted by *DEK* is a symmetric cryptography key generated by the domain key manager *DKM* and shared between all key managers in a domain. More precisely, *DEK* is a unique symmetric key shared between *DKM* and all *AKM*s in a domain. The establishment of this key is carried out prior commencement of each group communication in the domain. The *DKM* delivers the domain encryption key to all *AKM*s via secure channels.

The function of *DEK* is to provide a means for setting up a secure multicast transmission among all *AKM*s in order to distribute the new traffic key generated upon any changes in group membership, or to disseminate the controlling messages such as notification of establishment of a new group communication, process of updating keying materials, as well as host mobility announcement. Since the membership of key managers is static, the policy associated with the domain encryption key is defined that the key is fixed and valid until the policy is changed.

4.4.6.3 Member Encryption Key (MEK_i)

A unique key shared between the *AKM* of an area and each group member residing in that particular area is called member encryption key. More precisely, MEK_i is a symmetric cryptography key shared between an area key manager and a group member M_i. The member receives MEK_i during the first contact to become a member of a group communication. The member encryption key is generated by the *AKM* of the particular area, where member registers with a particular group communication.

The member encryption key is derived with each *AKM* using a pseudo random function without involving the *DKM*. The *AKM* delivers the new MEK_i to member M_i using a secure means such as secure association like SSL. The characteristic of the member encryption key is that all area key managers are able to independently generate the same MEK_i for each member without the collaboration of the other *AKM*s or the *DKM*.

The member encryption key is only used for secure unicast communications that take place between the *AKM* of an area and a member of that particular area. Since group membership can be dynamic, the lifetime of MEK_i is proportional with the membership lifetime of the corresponding member. Therefore, the member encryption key remains valid until the member has departed a particular group communication.

4.4.6.4 Area Encryption Key (*AEK*)

Area encryption key is a unique key shared between an area key manager and the group members residing in that specific area. More precisely, AEK_i corresponds to the symmetric key shared between AKM_i and group members residing in area *i*. Therefore, each area has its own area key AEK_i that is different from the other area key AEK_j. This key is generated by an *AKM* of a particular area. A group member does the establishment of an area encryption key after it joins a group communication.

The aim of having an area encryption key is to provide efficient and scalable rekeying process since all members of a group within an area are managed under a key with minimum communication overhead. The function of area encryption key is restricted only to secure multicast communication between an *AKM* and its associated members. The area encryption key is assumed to be valid until any changes happen in the group membership in the corresponding area as long as there is any member in that area. Once the area encryption key expires or needs to be refreshed, a new key must be generated and disseminated to the affected area.

4.4.6.5 Traffic Encryption Keys (*TEK*)

Traffic Encryption key referred to as *TEK* is a unique key used by all members of a particular group communication such as senders and receivers to encrypt and decrypt the data traffic in a domain. The *DKM* is responsible for generating *TEK* and distributing it to all *AKM*s in the domain, which then in turn disseminate this key to all members of the group residing in their area. The Traffic key is established when the first member invokes to join a group communication.

The traffic encryption key can be protected using several options to securely deliver to the group members. At the domain level, if the *DKM* uses unicast transmission, the *TEK* is sent to every *AKM* independently protected under the domain-area encryption key associated with that specific area manager. To achieve more efficiency, the *DKM* can use multicast communication and sends a single message to all *AKM*s protected under domain encryption key *DEK*.

At the area level, each *AKM* employs either unicast or multicast depending on the event occurs in its area. In case of using unicast the message is protected under each member encryption key MEK_i, whereas the message is encrypted by *AEK* when multicast communication is used.

The main functionality of traffic key is for protecting the real data in communication. The traffic key is valid until any variation occurs in the membership of the group. Whenever the traffic key expires or need to be updated, a new traffic key must be generated and distributed to all group members in a domain.

4.4.6.6 Summary of Keys

The different aspects of cryptography keys described in Section 4.4.6 are summarized in Table 4.3 in terms of the type of key, the responsible entity for the generation of that particular key, the entities which hold the key and finally the function of each key.

Table 4.3: Summary of keys needed in secure group communication.

Type of Key	***Generated by***	***Shared between***	***Function***
DAK$_i$	*DKM*	*DKM* & *AKM$_i$*	• A common unique key shared between *DKM* and a specific *AKM$_i$*. • Protect unicast communication between *DKM* and *AKM$_i$*.
DEK	*DKM*	*DKM* & *AKM*s	• A unique share key shared between *DKM* and all *AKM*s in a domain. • Encrypt the keying materials distributed via the multicast communication to all *AKM*s.
MEK$_i$	*AKM*	*AKM* & Member *i*	• A symmetric cryptography key shared between *AKM* and member *i*. • Protect unicast messages sent from *AKM* to member *i*. • *AKM$_j$* authenticates the visiting member *i* in area *j*.
AEK$_i$	*AKM*	*AKM$_i$* & Members	• A symmetric key shared between *AKM$_i$* and all members residing in area *i*. • Protect keying materials distributed via multicast communication to all members within area *i*.
TEK	*DKM*	*DKM* & *AKM* &Members	• A unique common key shared between all members of a group. • Protect data traffic in group communication.

4.4.7 Mobility key management

In this scheme, some system security parameters initially set up by the trusted *DKM* is securely delegated to the *AKM*s for individual key establishment of each group member. A unique cryptographic key *DEK* is shared by *DKM* between all *AKM*s. This key is used for secure communication among the area key managers. Moreover, this key is one of the

chosen security parameters that enable each *AKM* to derive an individual key of each member without involving the *DKM* and other *AKM*s.

The *AKM* uses a key derivation function like PRF-HMAC-SHA-256 (Frankel & Kelly, 2007) to generate MEK_i of a newly joining member. While PRF-HMAC-SHA-256 provides secure pseudo random functions suitable for generating keying materials, its goal is to ensure the packets are authentic and not modified in transit. To generate the MEK_i, each *AKM* uses Formula 4.1 as follows:

$$MEK_i = PRF\text{-}HMAC\text{-}SHA\text{-}256(DEK \,||\, ID_{M_i} \,||\, text) \qquad (4.1)$$

The *text* contains other security parameters corresponding to the member. All *AKM*s require to use the same PRF-HMAC-SHA-256 in order to achieve a coordination throughout the domain for deriving the same member encryption key in all areas. Using the same PRF enables the *AKM*s to generate a unique MEK_i specific for member M_i. Therefore, the *AKM* is able to proceed with the authentication of the visiting member and area key dissemination without involving the *DKM*. This verification mechanism enables all the *AKM*s to verify the *MEK* presented by a mobile member. For instance, in Figure 4.6, when member M_i moves from area *i* to area *v*, it sends a *Move Notify* message signed with MEK_i to AKM_v. AKM_v calculate a new MEK_i using Formula 1. If the new MEK_i^* is equal to MEK_i presented by the member M_i, the member is authorized to access the information of the new area.

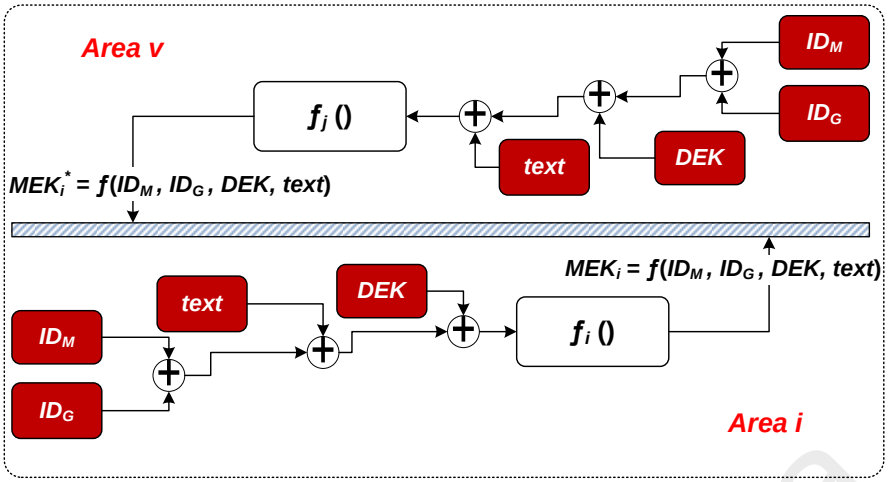

Figure 4.6. Mobility key management procedure.

The advantages of using this mechanism are as follows:

1) The bottleneck on the *DKM* is mitigated for managing mobility of dynamic members,
2) The resource constrained mobile devices do not undergo the heavy computing process during authentication in the visited area, and
3) The management of mobile members is distributed between all *AKM*s, which result in saving enormous bandwidth utilization during rekeying process.
4) The scalability problem *1-affects-n* is alleviated.

4.5 Protocol functionalities

In this section, the functionalities of the identified protocols used in this architecture are described. These protocols are used for the provision of backward and forward secrecy within a group communication.

4.5.1 New member joining protocol

This protocol governs the join of a new host member to a group communication with providing backward secrecy. In other words, the new member who wishes to join the group is prevented from having access to the traffic flow before its admission to the group.

Taking consideration backward secrecy, the main functional requirements of the protocol are as follows. The new member must add to the group communication. A set of keys including the new traffic key *TEK* and the new area encryption key *AEK* must be delivered to the newly joining member. Moreover, a rekeying process of the new traffic key in the domain and a rekeying process of an area encryption key *AEK* where the join occurs must be initiated throughout respectively the domain and the area in order to achieve backward secrecy.

In order to join the group session, the new member sends its request to its area key manager AKM_i. If the verification of member request succeeds, the AKM_i informs the *DKM* and concurrently generates a new member encryption key and a new area encryption key. The *AKM* sends the keying materials to newly joining member. Then, it invokes the rekeying process to replace the old area encryption key for the remaining group members in its area to achieve backward secrecy at area level. The *DKM* generates a new *TEK* and then initiates rekeying process throughout the domain to guarantee backward secrecy in the domain.

Figure 4.7 depicts the flow diagram of the join protocol for member M_i in area *i*. Take note the AEK_i, MEK_i, and n_i used in Figure 4.6 respectively denote area encryption key of area *i*, member encryption key associated with member M_i, and the number of members residing in area *i*.

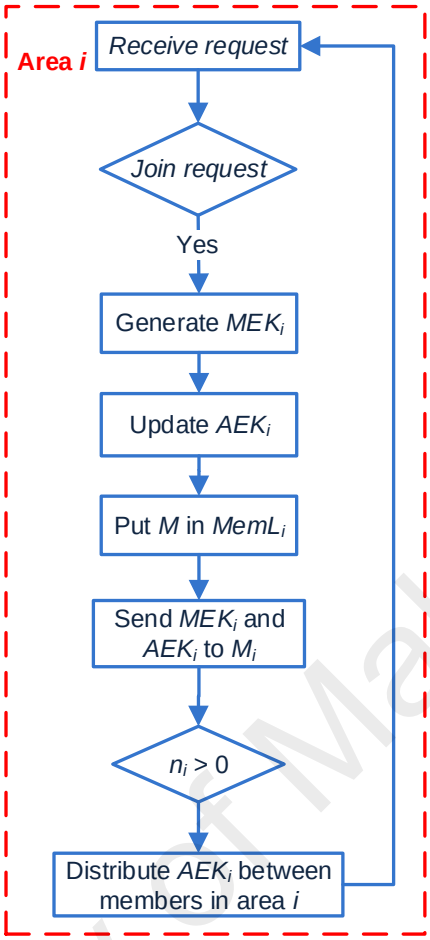

Figure 4.7: join protocol flow when a new member joins the group from area *i*.

The details of the join protocol are described as follows:

i. Member M_i who is willing to join a group communication sends a join request message to the area key manager AKM_i of area *i* encrypted with its private key as shown in Expression 4.2.

$$M_i \rightarrow AKM_i : \{ID_{M_i} || ID_{A_i} || ID_G || text\}K_{Mi} \quad (4.2)$$

ii. On receipt of the join request, AKM_i verifies the member's request. If the member is authorized to join the group session, AKM_i informs the *DKM* by sending a join request message encrypted with DAK_i as shown in expression 4.3. AKM_i simultaneously generates MEK_i for the member M_i and sends it to

the new joining member protecting under member's public key as demonstrated in expression 4.4. Moreover, AKM_i generates the new area encryption key AEK_i to maintain backward secrecy at its area.

$$AKM_i \rightarrow DKM : \{ID_{A_i} \,||\, ID_{M_i} ||\, ID_G \,||\, text\}DAK_i \qquad (4.3)$$

$$AKM_i \rightarrow M_i : \{ID_{A_i} \,||\, ID_{M_i} ||\, ID_G \,||\, MEK_i \,||\, text\}PK_M \qquad (4.4)$$

iii. On receipt, the *DKM* checks the message by decrypting it with secret Domain-area encryption key DAK_i which is shared by AKM_i. With this assumption that the host *M* is granted permission to join the group, the *DKM* generates a *join grant* message containing the new traffic key *TEK*.

iv. The *DKM* sends the *join grant* message as well as the *ready to rekey* message to all *AKM*s including AKM_i in the domain in order to replace the old traffic key with the new traffic key in a particular group session. These messages encrypted with domain encryption key *DEK* and send with a multicast communication as shown as follows:

$$DKM \Rightarrow AKM : \{ID_A \,||\, ID_{M_i} ||\, ID_G \,||\, new\_TEK \,||\, text\}DEK \qquad (4.5)$$

v. When AKM_i receives the new *TEK*, it delivers the new *TEK* and the new AEK_i by a unicast message encrypted with MEK_i as depicted in expression 4.6.

$$AKM_i \rightarrow M_i : \{ID_{A_i} \,||\, ID_G \,||\, new\_TEK \,||\, new\_AEK_i \,||\, text\}MEK_i \qquad (4.6)$$

It also distributes the new keying materials to the other group members M_i^* residing in area *i* using a multicast message encrypted with the old AEK_i as shown in expression 4.7.

$$AKM_i => M_i^* : \{ID_{A_i} \| ID_{M_i^*} \| ID_G \| new\_TEK \| new\_AEK_i \| text \} old\_AEK_i \quad (4.7)$$

vi. Every AKM_t distributes the new traffic key in area t, if there is a member of that specific group residing in its area, using a secure multicast communication protected under area key AEK_t as follows:

$$AKM_t => M_t : \{ID_{A_t} \| ID_G \| new\_TEK \| text \} AEK_t \quad (4.8)$$

Figure 4.8 illustrates the sequence message flow of the join protocol when member M_i joins the group via area i.

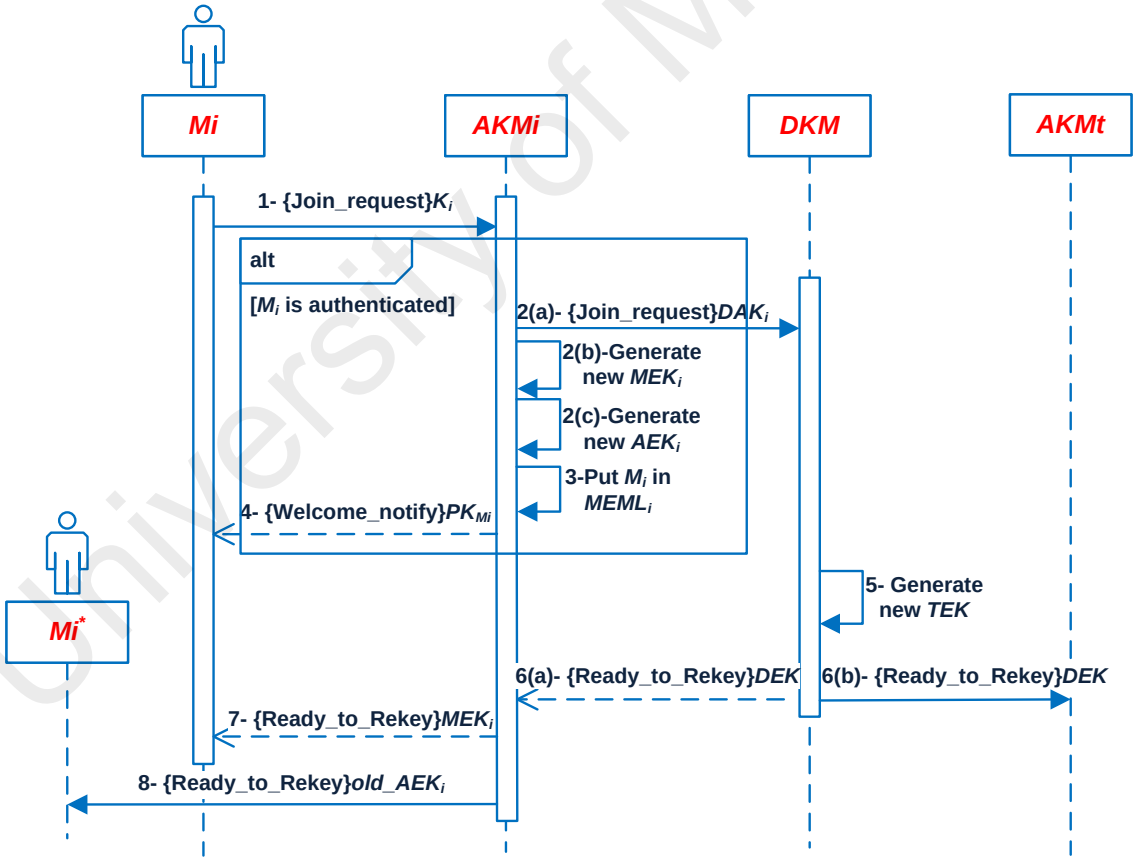

Figure 4.8: New member joining protocol sequence diagram.

Figure 4.9 describes the implementation procedure corresponding to the new member joining protocol in the form of pseudo code.

```
Procedure newMemberJoin(memID, areaID)
Begin
Join_Req_Message = Encrypty (joinReqMsg, memID, areaID)
SentToDKM (join_Req_Message);
If (M[memID] verification is successful) then
{
    MEK[memID]= Generate new_MEK;
    AEK = Generate new_AEK;
    MemL[areaID].add (M[memID]);
    welcome_Notify_Message = Encrypt (AEK, MEK) with PK
    SentToMem (welcome_Notify_Message)
}
Else
    SendToMem(rejectRequest)
Endif
Receive (Ready_To_Rekey);
TEK = Decrypt (Ready_To_Rekey);
Ready_to_Rekey  = Encrypt (TEK) with AEK;
MultiSend (Ready_to_Rekey);
n[areaID] = n[areaID]++;
End
```

Figure 4.9. New member joining protocol pseudo code.

4.5.2 Member mobility protocol

The mobility is the most and unique features of dynamic wireless mobile environments, which facilitates the members not only join or leave the group communication, but also move between areas while remaining in the session. Mobility refers to the process by which a moving member changes its network attachment point or its area.

A moving member may access to the local security information of the visited area that had been probably valid before the time that the visiting member joined the group. In addition to the old keying materials associated with the origin area where a moving member joined, the moving member may accumulate the security information for each area it visits. The access of a moving member to the security information is necessary to be controlled in the visited area to avoid breach of backward secrecy. This protocol will thus consider backward secrecy for member mobility by refreshing the area encryption key in the visited area when a group member moves in.

Group members may frequently move between the areas while still remaining in the session in wireless mobile environments. To maintain the backward secrecy, each time a member moves into an area, the rekeying process may need to be performed in order to refresh the old area encryption key. In dynamic environments, frequent rekeying may lead to disruption of group communication, thus it is necessary to keep track of mobility for preventing frequent rekeying processes whenever a moving member returns back to an area that it has previously visited.

To facilitate this, each area key manager in a domain needs to securely maintain a list which stores the information of group members that moves from its managing area to another. This list is referred to as a Key encryption key Mobile Owner List (KMOL). When a member enters an area, the *AKM* of the visited area can determine by looking up its *KMOL* whether the member is a returning member who is just moving back to the area or is a new visiting member. In case the member appears in the list, the area encryption key may not need to be updated. If such a member is not on the list, depend on the join time of the moving member and the latest update time of the area encryption key, the rekeying process for the area encryption key need to be carried out. As long as the time which the member has joined the group is after the latest time, which the area encryption key of the visited area was updated, the area encryption key need to be refreshed. On the other hand, forward secrecy requirement is pointless at the old area where the mobile member moved from because the member is still in the session.

When a member moves into a new area, the security information such as cryptographic keys needs to be exchanged between the moving member and the *AKM* of the visited area via a secure channel. Both entities require to establish a common cryptography key. The *AKM* of the visited area is able to generate the member encryption key of each moving

member without involvement of the *DKM* and the *AKM* of the old area. The member encryption key is used for secure transfer between a moving member and the new *AKM*.

This protocol governs the establishment of a member encryption key corresponding to a moving member and the *AKM* of a visited area. The generation of a member encryption key is conducted by the *AKM* of a visited area using a key derivation function. All *AKM*s in a domain employ the same key derivation function, which causes every *AKM* to generate similar member encryption key for a moving member that visits their areas. An *AKM* uses the system security parameters initially set up by the *DKM* and the security information of a group member as the input parameters of the key derivation function. The protocol also provides authentication mechanism for every *AKM* to verify the authenticity of the moving member.

Throughout this protocol, the local or old area and new or visited area terminology are used when member transfer between areas. The aforementioned terms are distinguished as follows:

- The area where the moving member is moving from is referred to as the old or local area.
- The area where the moving member is moving into is called as the new or visited area.

Figure 4.10 shows the flow diagram of the member moving protocol when a member M_i moves from area *i* to area *v*. The details of the protocol in terms of the provision of backward secrecy in the visited area, the verification of the moving member, and finally the delivery of an area encryption key of a visited area to the moving member is described as follows:

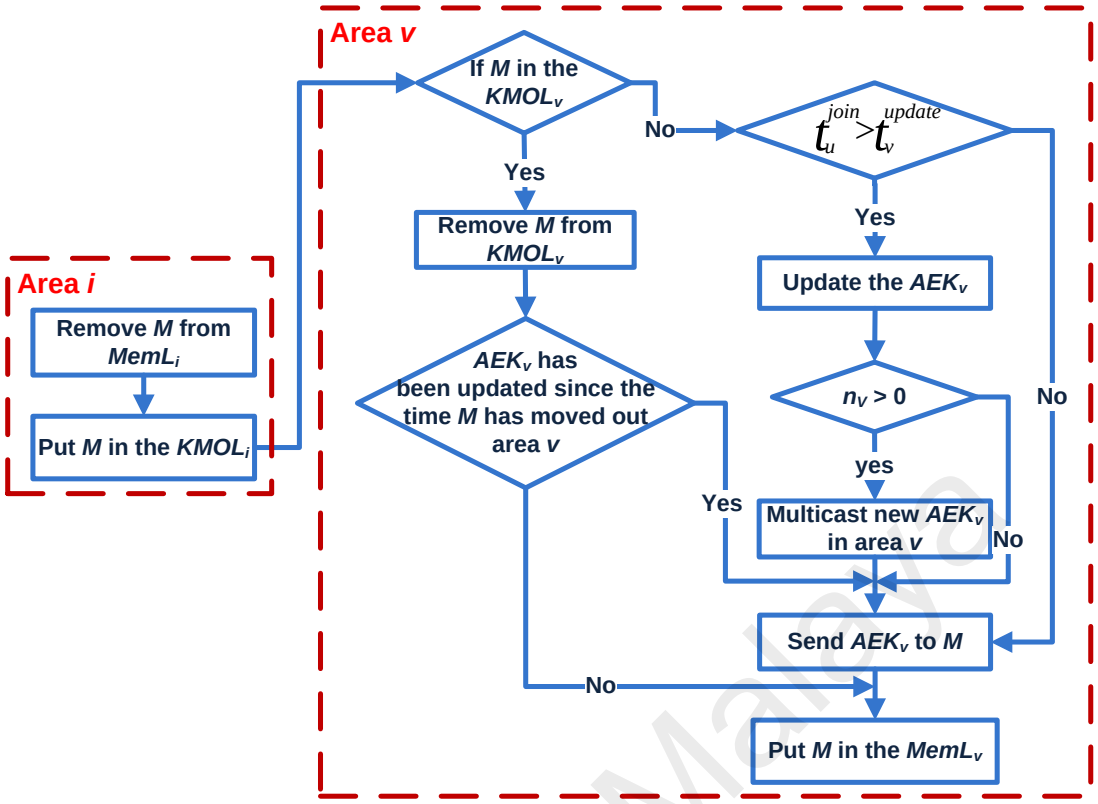

Figure 4.10: Member mobility protocol when a member M_i moves from area *i* to area *v*.

i. The group member M_i who wishes to move into area *v* informs its local area key manager AKM_i by sending a *move notify* messages containing the *ID* of the area that he is moving into. The *move notify* message is protected and signed with MEK_i as shown in expression 4.9.

$$M_i \rightarrow AKM_i : \{ID_{A_i} \,||\, ID_{A_v} \,||\, ID_{M_i} || \, ID_G \,||text\,\} MEK_i \qquad (4.9)$$

The member also sends a *move notify* message to the target area key manager AKM_v protected and signed with MEK_i as follows.

$$M_i \rightarrow AKM_v : \{ID_{A_i} \,||\, ID_{A_v} \,||\, ID_{M_i} || \, ID_G \,||text\,\} \, MEK_i \qquad (4.10)$$

ii. On receipt, AKM_i (area key manager of the old area *i*) checks the message by decrypting it with MEK_i and informs the *DKM* about the member movement

by passing the message protected under DAK_i. The expression 4.11 shows the encrypted message sent to *DKM*.

$$AKM_i \rightarrow DKM : \{ID_{A_i} \| ID_{A_v} \| ID_{M_i} \| ID_G \| \text{text}\} DAK_i \quad (4.11)$$

AKM_i does not require to carry out the rekeying process for the area encryption key AEK_i since the moving member still maintains session continuity while changing point of attachment to the network.

iii. Upon receiving the message from M_i, the target area key manager AKM_v requires to do the following:

   a. It checks the message and authenticates the moving member M_i in order to grant the local keying materials to the visiting member. To verify the moving member M_i, AKM_v needs to generate the member encryption key MEK_i.

   b. AKM_v (area key manager of visited area v) is able to derive independently the member encryption key MEK_i without involving the *DKM* and AKM_i. AKM_v uses a key derivation function such as PRF-HMAC-SHA-256 along with security information associated with the moving member M_i and security parameters shared previously by the *DKM* (see Section 4.4.7). After successful derivation of MEK_i, the AKM_v verifies whether M_i is valid member or not. If the member verification succeeds, AKM_v looks M_i's identity up in its $KMOL_v$, and does following:

   c. If M_i is not in the *KMOL* that means this is the first time M_i that visits the area v and thus AKM_v requires to compare the time that member M_i

has joined the group with the time that the latest area encryption key rekeying performed.

If the time that member has joined the group is after the last update of area encryption key, AKM_v needs to perform the key update process to refresh the area encryption key AEK_v in order to maintain backward secrecy. AKM_v generates a new area encryption key and sends it to M_i encrypted with MEK_i as described in expression 4.12. It also distributes the new AEK_v between all members of the group M_v residing in area v preferably by a multicast message as shown in expression 4.13.

$$AKM_v \rightarrow M_i : \{ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel new\_AEK_v \parallel text\}MEK_i \qquad (4.12)$$

It also distributes the new AEK_v between all members of the group M_v residing in area v preferably by a multicast message as shown in expression 4.13.

$$AKM_v => M_v : \{ID_{A_v} \parallel ID_G \parallel new\_AEK_v \parallel text\}\ old\_AEK_v \qquad (4.13)$$

d. If M_i already visited area j and appears in the $KMOL_v$, there is no need to rekey the area encryption key. AKM_v only requires to check whether the area encryption key AEK_v has been updated since the last visit paid to area j by the member M_i.

   If there is a new area encryption key, AKM_v sends the new AEK_v encrypted with MEK_i as shown in expression 4.14. Otherwise, it sends a *move welcome* message to M_i to inform it the previous area encryption key is still valid and can be used.

$$AKM_v \rightarrow M_i : \{ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel AEK_v \parallel text\}MEK_i \quad (4.14)$$

e. AKM_v needs to update the information of its $MemL_v$ by adding the information concerning with the new arrival.

iv. AKM_v notifies AKM_i and the *DKM* about accomplishing the movement of M_i from area *i* to area *v*.

v. AKM_i needs to remove the member information from $MemL_i$ and update its $KMOL_i$ in order to keep track of member M_i's mobility along with the number of area encryption keys that have been accumulated by the moving member.

Figure 4.11 depicts the sequence message flow of the member moving protocol when a member M_i moves from area *i* to area *v*.

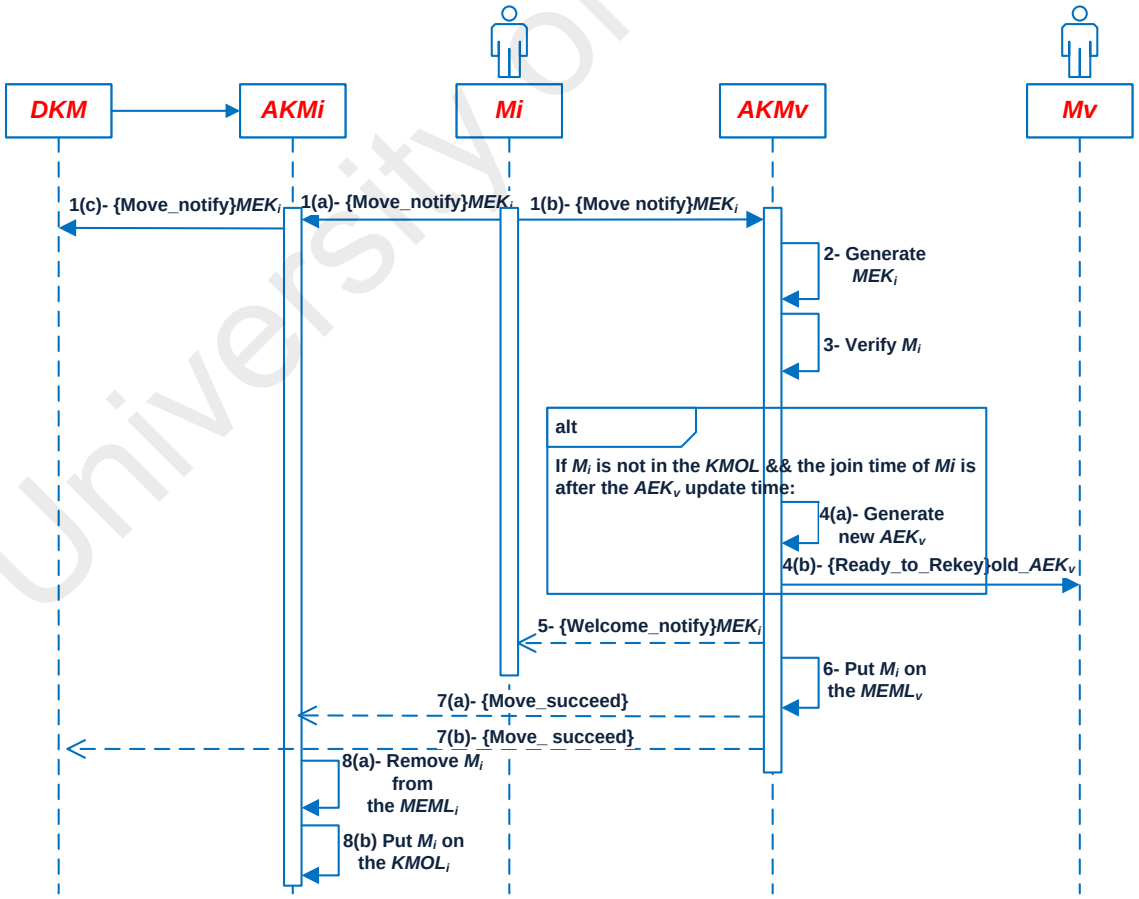

Figure 4.11: Member mobility protocol message flow diagram.

The pseudo code of the implementation corresponding to the member mobility protocol is illustrated in Figure 4.12.

```
Procedure memberMove(message, memID, areaMovFrm, areaMovTo)
Begin
    MEK* = Generate MEK of memID;
    If (MEK[memID] = MEK*)
    {
        Search KMOL;
        If (M[memID] is not on KMOL[areaMovTo] & M[memID].JoinTime > AEK.updateTime)
        {
          old_AEK = AEK[areaMovTo];
          AEK[areaMovTo] = Generate new_AEK;
          Ready_To_Rekey= Encrypt (AEK[areaMovTo]) with old_AEK
          MultiSend (Ready_To_Rekey);
        }
        Else
          KMOL[areaMovTo]= Remove (M[memID]);
        Endif
        Welcome_Notify_Message= Encrypt (AEK[Moveto]) with MEK[memID];
        SendToMem (memID , welcome_Notify_Message)
        MemL[areaMovTo].add (M[memID]);
    }
    Else
        Send (reject_Move_Notify);
    Endif
    SendToDKM (MoveSecuceed);
    SendToAKM[areaMovFrm] (MoveSecuceed);
End
```

Figure 4.12: Member mobility protocol pseudo code.

In summary, this protocol manages the moves of member M_i from an area managed by AKM_i to another area in a domain managed by AKM_v while still remaining in the session. The management of the move is delegated to the *AKM* that is responsible for governing the destination area. Therefore, the burden of mobility management transfers from the *DKM* to the *AKM*s, which result in omitting unnecessary delays and possible bottlenecks at the *DKM*.

4.5.3 Existing member leaving protocol

This protocol governs the leave of an existing group member from the group with provision of forward secrecy. In other words, the departure is prevented from having access future group communication. Since the member leaving affect the current keying materials, the forward secrecy requires the protocol to replace the current keying materials with a new set of keys.

The functional requirements of the protocol taking consideration forward secrecy are as follows. The leaving member must be eliminated from the group communication. The process of rekeying the *AEK* must be initiated in the area where the leave occurs or area that have been visited by the leaving members and their area encryption key *AEK* are still valid in order to achieve forward secrecy at area level. Furthermore, the rekeying process of the traffic key *TEK* in the domain must be initiated to update the old traffic key.

When a member leaves the group session, it informs its area key manager AKM_i. The AKM_i needs to inform the *DKM* as well as to replace the current area encryption key. In order to maintain forward secrecy at the area level, AKM_i generates new keys and invokes the rekeying process to update the keying materials for the remaining group members in the area.

The *DKM* removes the information of departure from the group session by updating the list of current group members *MemL*. The *DKM* generates new *TEK* and invokes the rekeying process to update the old *TEK* for the residue members of the group throughout the domain. It also sends information of the departing member to all the *AKM*s. Since the leaving member may hold valid keying materials associated with every visited area, the *AKM*s of the visited area needs to initiate the rekeying process to replace their area encryption key and update their *KMOL*.

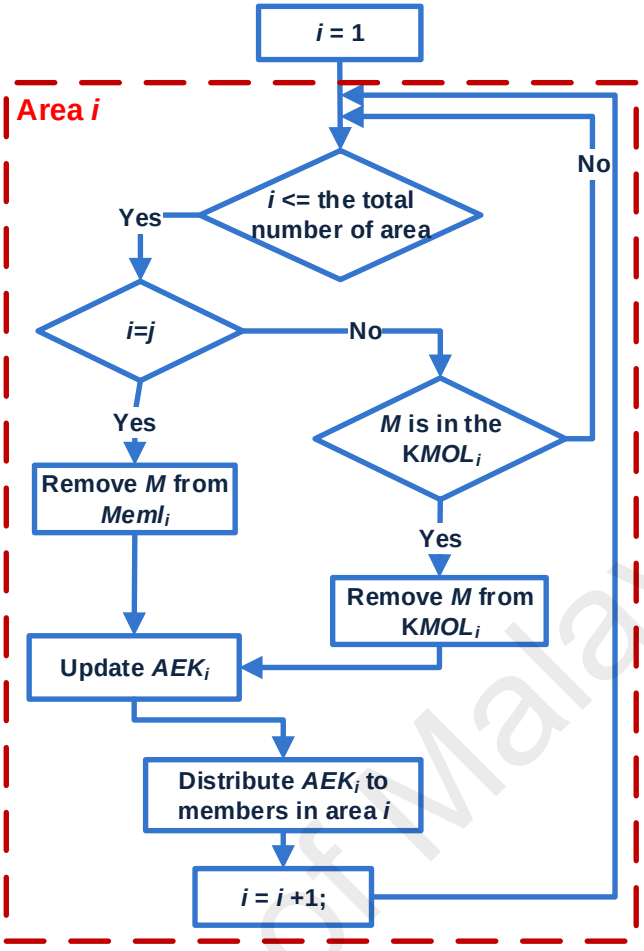

Figure 4.13: Leave protocol when member *M* leaves the group from area *i*.

Figure 4.13 depicts the flow diagram of leave protocol when the member M_i residing in area *i* leaves a group communication taking consideration to secure access to future data traffic. Thus, the traffic key as well as area encryption keys held by leaving member needs to be replaced with new keys in this event. The different steps of this protocol describe in details as follows:

i. A member M_i wishes to leave a group communication informs its area key manager AKM_i by sending a *leave notify* message protected under the member encryption key MEK_i as follows:

$$M_i \rightarrow AKM_i : \{ID_{A_i} \parallel ID_{M_i} \parallel ID_G \parallel text\}\ MEK_i \qquad (4.15)$$

ii. Upon receiving the *leave notify* message by AKM_i, it checks the message with decrypting it by using secret shared key MEK_i, and then passes the leave notify message to the *DKM* encrypted with domain-area encryption key DAK_i as shown in expression 4.16

$$AKM_i \rightarrow DKM : \{ID_{A_i} \parallel ID_{M_i} \parallel ID_G \parallel text\}DAK_i \quad (4.16)$$

In addition, AKM_i generates new area encryption key AEK_i and sends a *ready to rekey* message containing the new area encryption key to remaining members in the group (excepting leaving member) in area *i*. The new area encryption key AEK_i is sent by a unicast message to each member M_i^* protected with the member encryption key MEK_i^* as follows:

$$AKM_i \rightarrow M_i^* : \{ID_{A_i} \parallel ID_{M_i^*} \parallel ID_G \parallel new\_AEK_i \parallel text\}\ MEK_i^* \quad (4.17)$$

iii. On receipt, *DKM* decrypts the message using the Domain-area encryption key DAK_i and updates the list of the group communication members *MemL* by eliminating the information of the leaving member M_i. In addition, the *DKM* initiates a *ready to rekey* message containing the new traffic key *TEK* along with the *ID* of the member M_i, and then sends it to all *AKM*s in the domain as shown in expression 4.18.

$$DKM \Rightarrow AKM : \{ID_A \parallel ID_{M_i} \parallel ID_G \parallel new\_TEK \parallel text\}DEK \quad (4.18)$$

iv. Upon receiving the new *TEK*, AKM_i sends the new *TEK* to residue members M_i^* inside area *i* encrypted with new_AEK_i as described in expression 4.19. AKM_i removes M_i from $MemL_i$ as well.

$$AKM_i \Rightarrow M_i^* : \{ID_{A_i} \| ID_{M_i^*} \| ID_G \| new\_TEK \| text \} MEK_i^* \quad (4.19)$$

v. Since the leaving member might have visited the other areas v in the domain and knows all the AEK_vs associated with the visited areas, all AKMs of areas v need to perform the rekeying process. Thereby, all the AEKs that are still valid by the time that the member is leaving the group must be refreshed in each area, where M_i has previously visited.

In these areas ($v \neq i$) where the leaving member M_i is on the $KMOL_v$, and AEK_v is still valid, AKM_v ($v \neq i$) requires to update the area encryption key AEK_v and sends it along with the new TEK to the members M_v residing in area v encrypted with the secret key MEK_v associated with each member as shown in expression 4.20. AKM_vs also remove the information about M_i from $KMOL_v$.

$$AKM_v \rightarrow M_v : \{ID_{A_v} \| ID_{M_v} \| ID_G \| new\_AEK_v \| new\_TEK \| text \} MEK_v \quad (4.20)$$

vi. To distribute the new TEK in the other areas v^* in the domain, $AKM_v{}^*$ sends a multicast message containing the new TEK protected under $AEK_v{}^*$ to all members residing in area v^* as follows:

$$AKM_v^* \Rightarrow M_v^* : \{ID_{A_v^*} \| ID_{M_v^*} \| ID_G \| new\_TEK \| text \} AEK_v^* \quad (4.21)$$

Figure 4.14 shows the message flow diagram of this protocol when a member M_i in area i leaves the group. The pseudo code of implementation procedure corresponding to the member leaving protocol is depicted in Figure 4.15.

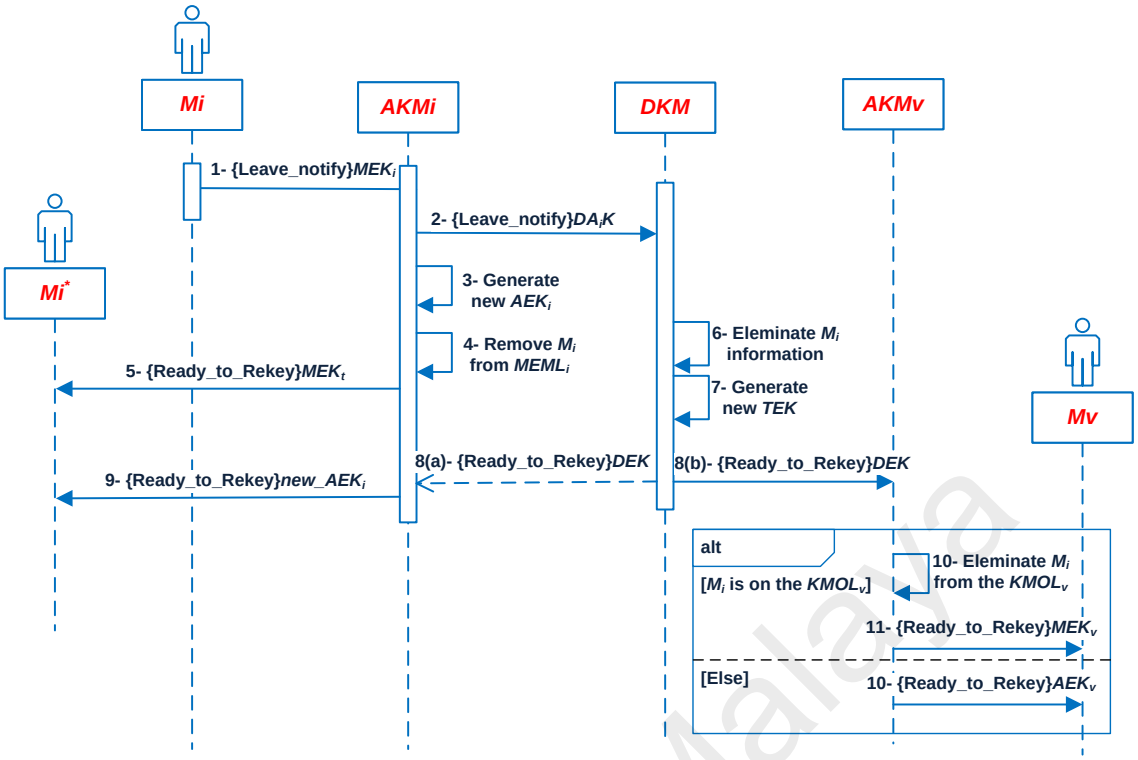

Figure 4.14: Existing member leaving protocol message flow diagram.

The above case was about a member who likes to leave the group communication voluntary. In case of involuntary leave, a member may be expelled from the group communication by the *DKM*. For this purpose, an *eject notify* message included the *ID* of the member is generated by the *DKM* and sent to that AKM_i which the mentioned member is residing in. AKM_i sends an *eject notify* message to the member M_i^*. The *DKM* also initiates rekeying protocol to replace old traffic key throughout the domain.

```
Procedure memberLeaving(memID)
Begin
    a = 0;
    While (a < number of areas) do
    {
    a++;
    m = number of members in area a;
    If (M[memID] is in MemL[a])
    {
        SentToDKM (Leave_Notify_Message)
        Meml.AKM[a].remove (M[memID]);
        AEK[a] = Generate new_AEK;
        m = m-1;
        n[areaID]=m;
        For (z=0, z<m; z++)
            Ready_To_Rekey = Encrypt (AEK[a]) with MEK[z];
            SentToMem (z, Ready_To_Rekey);
            Endfor
    Else If ( M[memID] is on KMOL[a])
    {
        KMOL[a].remove (M[memID]);
        AEK[a] = generate new_AEK;
        For (z=0, z<m; z++)
            Ready_To_Rekey = Encrypt (AEK[a]) with MEK[z];
            SentToMem(z, Ready_To_Rekey);
        Endfor
    Endif
    }
    If (m > 0)
    {
        ReceiveFrmDKM (message);
        TEK = decrypt (message);
        Ready_To_Rekey = Encrypt (TEK) with AEK[a];
        MultiSend (Ready_To_Rekey);
    Endif
    }
    a++;
    } Endwhile
End
```

Figure 4.15: Existing member leaving protocol pseudo code.

4.5.4 Rekeying traffic encryption key protocol

This protocol conducts the rekeying of a traffic key that needs when any changes occur in the group membership. As mentioned in joining protocol (see Section 4.5.1) and leaving protocol (see Section 4.5.3), rekeying protocol is necessary to provide backward secrecy when a member joins and forward secrecy when a member leaves a group communication. This protocol delivers the new traffic key to all members in a domain

(excluding member leaving). Note that the rekeying traffic key is always immediately carried out upon any membership changes occurred in a group communication in this design.

The sequence messages diagram of the rekeying traffic key protocol is depicted in Figure 4.16. The steps of protocol are described in details as follows:

i. *DKM* generates new traffic key *TEK*. It then distributes a ready to rekey message containing the new traffic key, and group communication *ID* to all *AKM*s in the domain either by:
   - Secure unicast channel, which protects each message under Domain-area encryption key DAK_i.
   - Secure multicast communication, which protects each message under Domain encryption key *DEK*.

ii. Upon each *AKM* receives the message, the message is decrypted with the keys shared between the *AKM* and the *DKM* to obtain the new traffic key *TEK*. The *AKM* then sends a ready to rekey message to all members of a multicast group in its area via either:
   - Secure unicast channel, which encrypts each message with member encryption key MEK_i.
   - Secure multicast communication, which encrypts each message with the area encryption key *AEK*.

When members receive the ready to rekey message, they decrypt it with the secret key shared by their area key manager *AKM* and obtain the new keying materials.

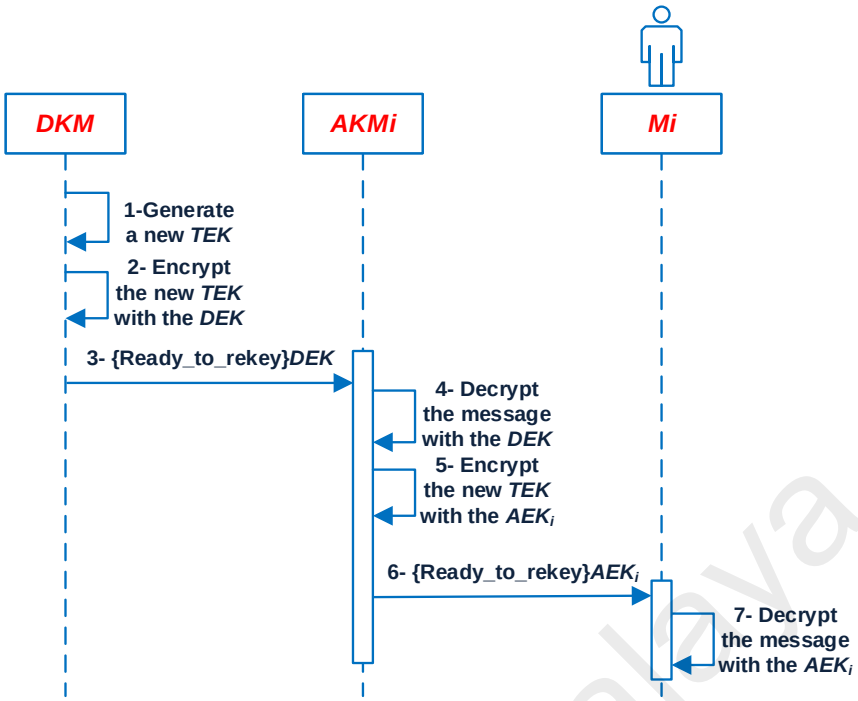

Figure 4.16: Rekeying *TEK* message flow.

4.5.5 Rekeying area encryption key protocol

This protocol conducts the rekeying process of an area encryption key that needs to occur due to the join protocol, leave protocol and mobility protocol in order to maintain backward secrecy and forward secrecy in an area. This protocol delivers the new area encryption key to all residing member in an area.

The steps required rekeying protocol in an area are shown in Figure 4.17. The description of this protocol comes as follows:

i. *AKM* generates new Area encryption key *AEK*. It sends the ready to rekey along with new *AEK* to all members of a group communication in its area. Distribution of area encryption key can be done either via

- Secure multicast communication, which protects the message under old *AEK*. Since the old area encryption key is not exposed in join or

move event, the old area can be used to protect multicast rekeying message in the join protocol or the mobility protocol.

- Secure unicast channel, which protects each message under secret key shared between area key manager and each member. In case existing member leaving protocol, the new area encryption key must be sent separately protected under the MEK_i of each member to all group members. Due to the old area encryption key *AEK* is compromised with the leaving member, this key cannot be used to protect rekeying messages in the leave event.

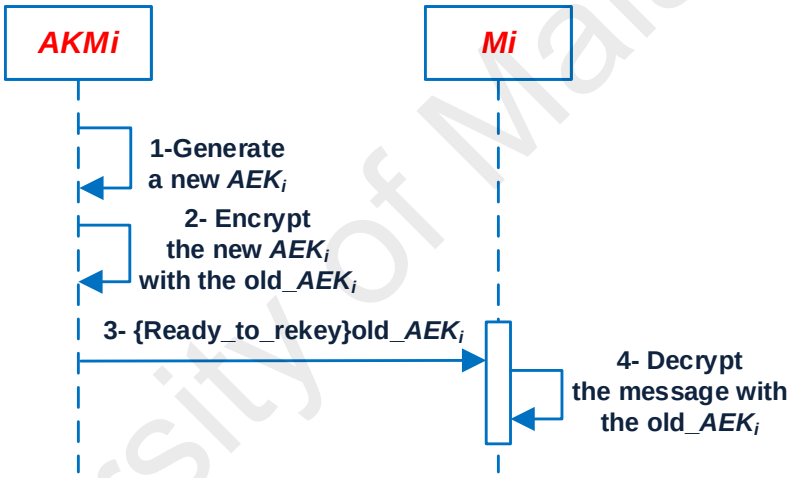

Figure 4.17: Rekeying AEK_i message flow diagram.

After receiving message by members, they encrypt message with the secret key shared by area key manager and obtain new area encryption key *AEK*.

4.6 Scenario example

This scenario depicts an example of a secure group communication in order to describe further the protocols required for managing the join, move, and leave events. Figure 4.18 shows a domain of group communication containing *n* areas in a domain. The *n* implies the number of existing areas in the domain, which is here equal to 3. It is assumed member *M*1, *M*3, *M*6, and *M*9 join the group through the area *A*1. The members *M*3, *M*11, *M*12,

and *M*13 as well as the members *M*2, *M*4, *M5*, *M7*, and *M*8 enter the group session respectively through area 2 and area *n*.

As shown in Figure 4.18, the area encryption key of each area is identified with A_iEK_j. The first subscript *i* illustrates the area identity, and the second subscript *j* refers to the last update that has occurred in the area *i*. For example, A_1EK_3 shows the area encryption key in area *A*1 which has been updated for the third time as a result of the changes in the area group membership. The *TEK* depicts the traffic key, which is common between all group members, and thus there is no need to be distinct with any subscript. The MEK_t is an individual key associated with the member *t*, for example MEK_1 is the member encryption key of *M*1. All *AKM*s can generate this key independent from the *DKM*.

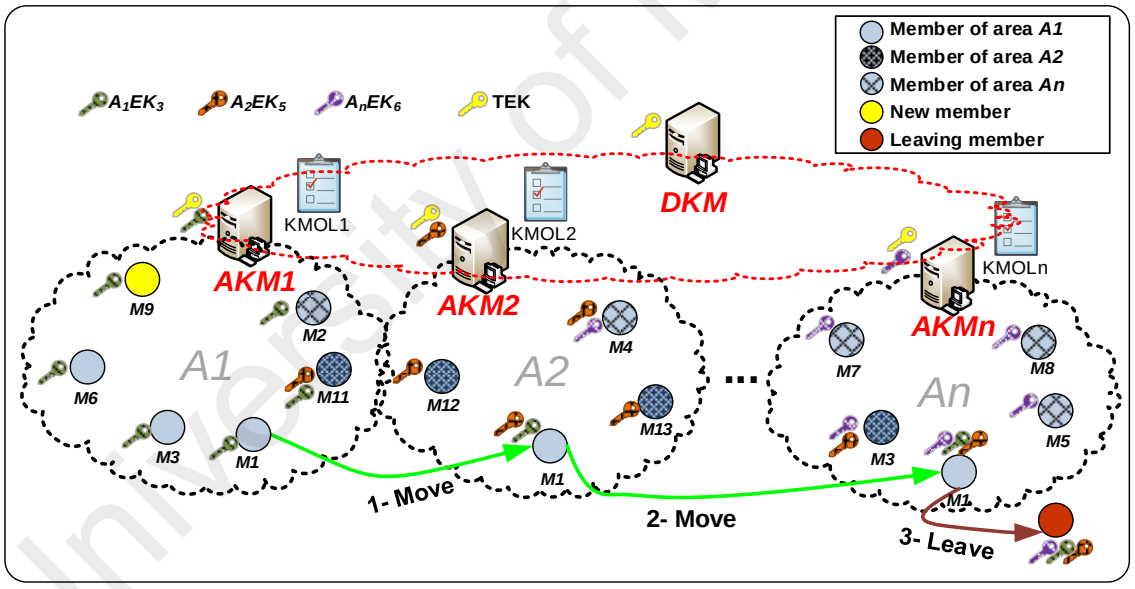

Figure 4.18: Example of secure group communication, including 3 areas and 13 members.

The domain governing entity is a key server labeled by the *DKM*, which is responsible for governing cryptographic keying materials throughout the domain. Similarly, every area is managed by area key manager labeled with *AKM1, AKM2, and AKMn* as shown

in Figure 4.18. The group managers such as the *DKM* and all *AKM*s are assumed to be static, and predetermined. They also trust in each other.

This scenario is comprised of three parts; 1) joining a new member, 2) a member movement, and 3) leaving an existing group member. It is assumed the group communication is already established and the keying materials are generated and distributed. The scenario starts with joining *M*9 in the group as a new member via the area *A*1. To study member mobility, the member *M*1 changes its location in area *A*1 and moves into area *A*2, and subsequently to area *An*. Finally, member *M*1 leaves the group while still carrying the valid area encryption key of *A*2. It is to investigate the treatment of *AKM*1 and *AKM*2 with leaving member *M*1 who previously visited their areas. All parts are described in details as follows:

- The member *M9* joins the group communication

  The member *M9* sends the join request containing the desired group communication ID to *AKM1*. The area key manager of *A1* sends the join request to the *DKM* as shown in Figure 4.19.

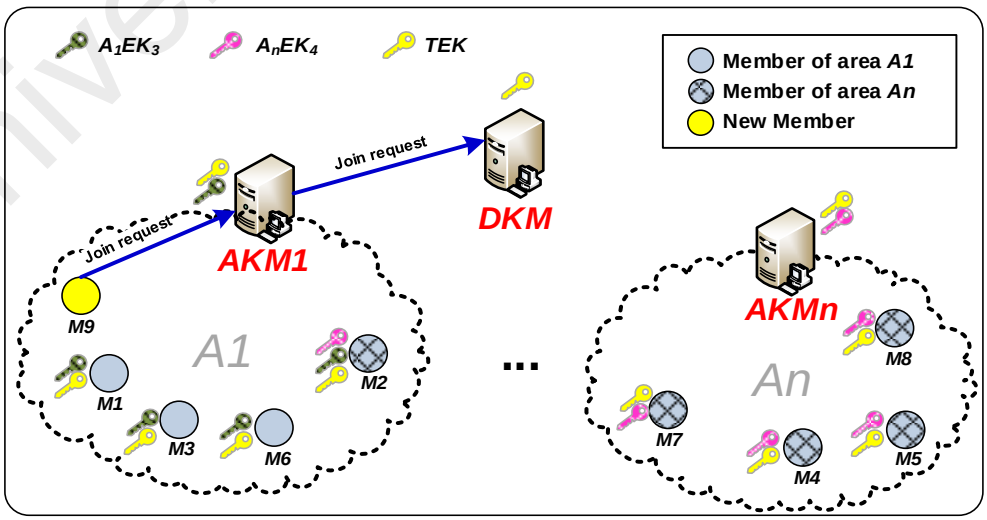

Figure 4.19: Sending the join request from the new member *M*9 to *AKM*1 and subsequently the *DKM*.

AKM1 generates MEK_9 as well as the new area encryption key A_1EK_4, and delivers these new keys to the member *M*9.

The *DKM* generates a new *TEK* and sends it along with ready to rekey message to *AKM*1, *AKM*2,… and *AKMn* as shown in Figure 4.20. *AKM*1 sends the new area encryption key A_1EK_4 and the new *TEK* to the member *M*9 with a unicast message.

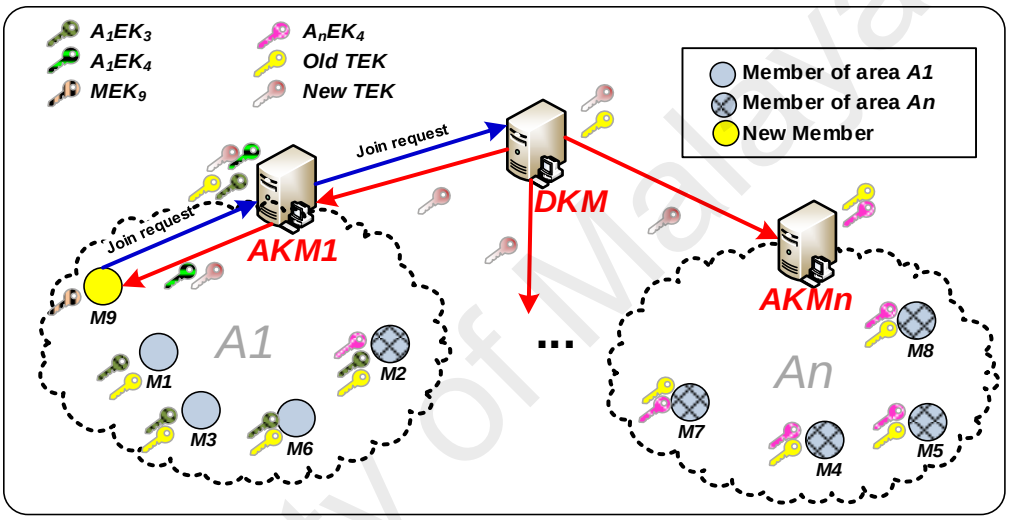

Figure 4.20: The *DKM* generates the new *TEK* and runs the rekeying process.

Upon receiving the new *TEK*, all *AKM*s carry out the rekeying process in their area to replace the old *TEK* with the new one as depicted in Figure 4.21. *AKM*1 also requires to send the new area encryption key A_1EK_4 along with the new *TEK* to all residing members *M*1, *M*2, *M*3, *M*6 in order to update A_1EK_3 by a multicast message protected under A_1EK_3.

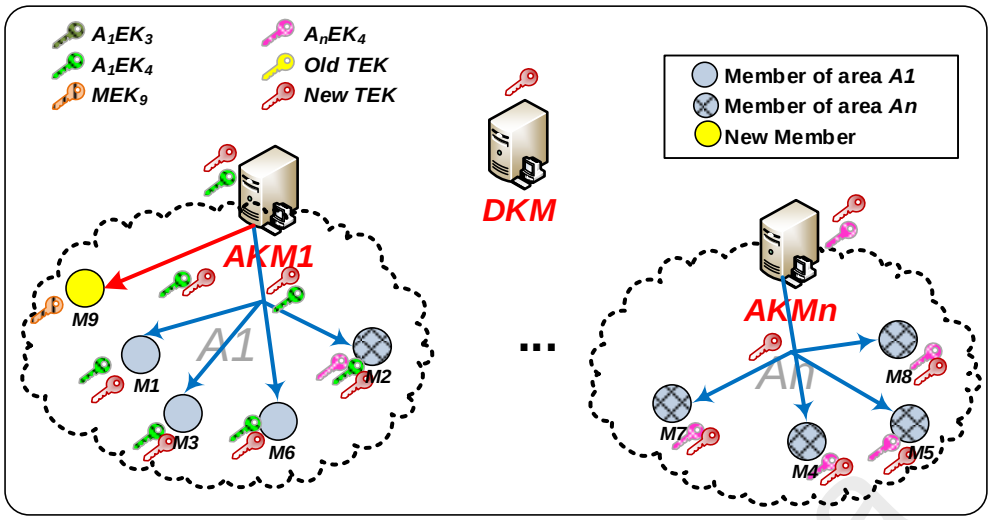

Figure 4.21: Rekeying process is conducted in area *A*1 to deliver new A_1EK_4 and the new TEK. *AKM*n multicast the new *TEK* in area *A*n.

- Member *M*9 moves to area 2

  Figure 4.22, and Figure 4.23 show an example that the member *M*1 move from area *A*1 to area *A*2. The member *M*1 is assumed to pay first visit to area *A*2. Therefore, the *AKM*2 does not have any information history about *M*1 in its *KMOL*2.

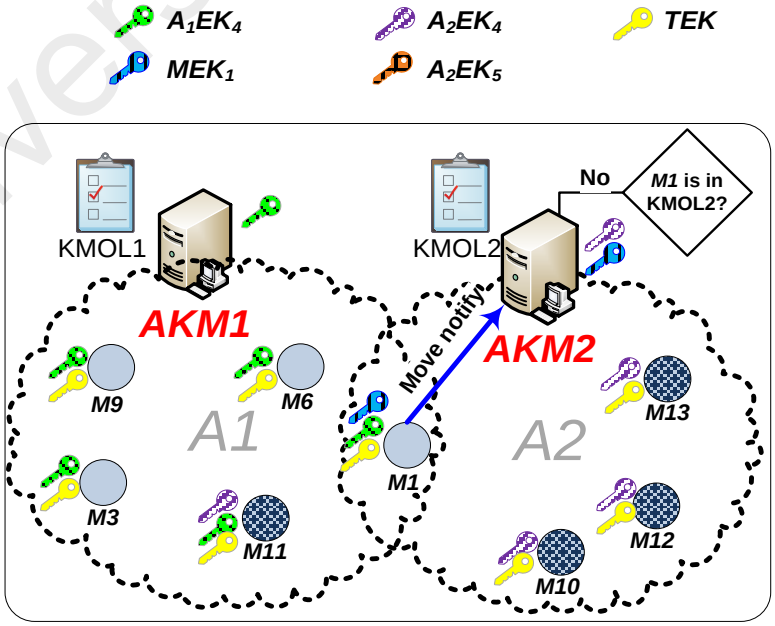

Figure 4.22: The member *M*1 moves from area *A*1 to area *A*2, while is not on *KMOL*2.

The member *M*1 sends a *move notify* message protected and signed with its member encryption key MEK_1 to the key manager of destination area 2. *AKM*2 requires to generates *M*1's member encryption key MEK_1 using the following key derivation function

$$MEK_1 = PRF\text{-}HMAC\text{-}SHA\text{-}256(DEK, ID_{M1}, text)$$

*AKM*2 decrypted *M*1's request and figure out it is first time *M*1 moves into area *A2* as the information of *M*1 has not recorded in *KMOL*2.

It is necessary *AKM*2 checks the time that *M*1 joined the group and compare with the time that A_2EK_4 was generated. *AKM*2 finds out the *M*1 joined the group after the time of A_2EK_4 generation. Thus, *AKM*2 generates new area encryption key A_2EK_5 and delivers it with a unicast message to *M*1 protected under MEK_1 and send it to members *M*10, *M*12, and *M*13 by multicast message encrypted with A_2EK_4.

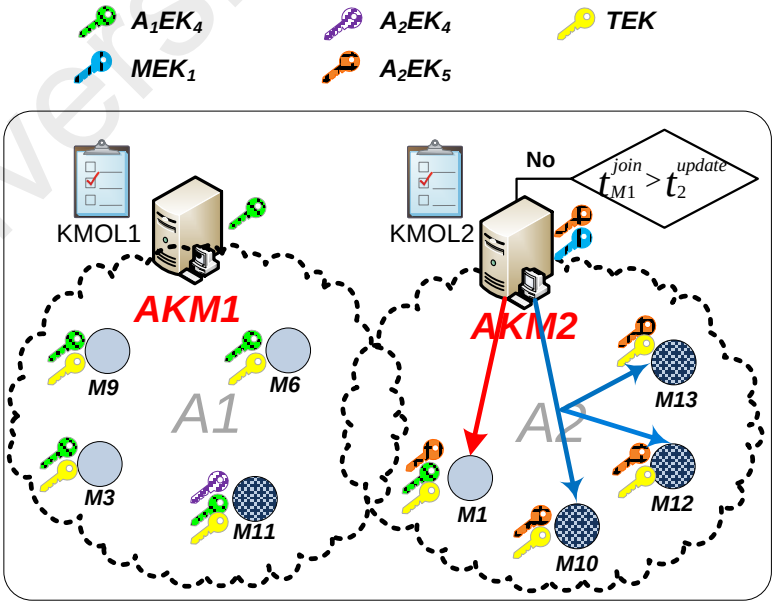

Figure 4.23: *AKM*2 compares the update time of A_2EK_4 with the joining time of *M*1.

Note that the *DKM* is not involved in mobility protocol. Since *M*1 still subscribes to the group session, *AKM*1 does not need to rekey the A_1EK_4.

- Member M_1 leaves the group communication

The member *M*1wants to leave the group while stays in area *An*. Before entering the area *An*, *M*1 has visited the area *A*1, and *A*2 and accumulated the corresponding area encryption keys A_1EK_6 and A_2EK_5 as shown in Figure 4.24. A_1EK_6 has already been replaced with A_1EK_7 in area *A*1, but A_2EK_5 is still valid in area *A*2 and used for key distribution.

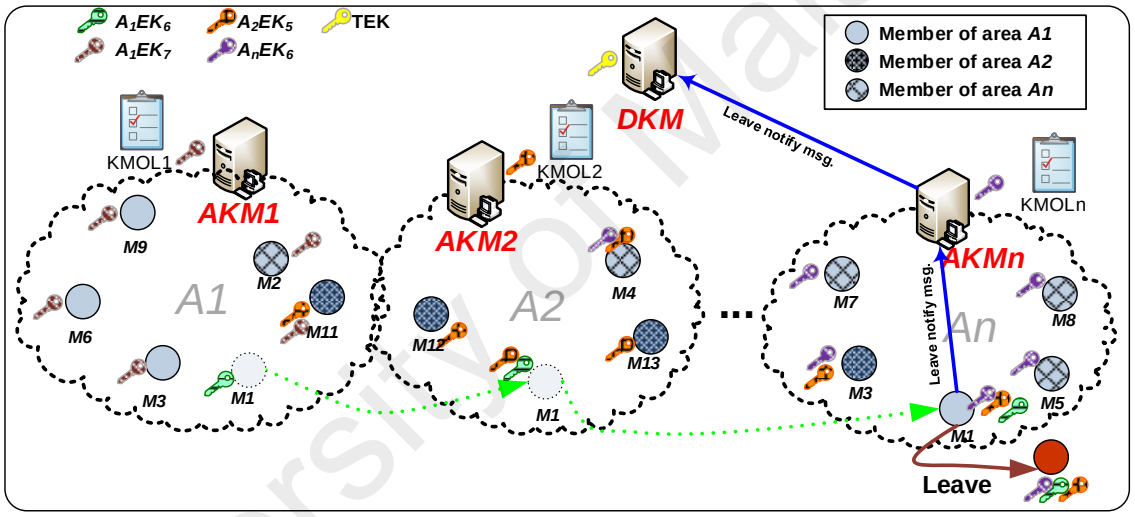

Figure 4.24: The member M_1 leaves the group in area *An*, while it carries expired A_1EK_6 of area *A1* and valid A_2EK_5 of area *A2*.

The member *M*1 informs the *AKMn* by sending a leave notify message as shown in Figure 4.24. *AKMn* also sends the leave notify message to the *DKM*. In order to maintain forward secrecy, the *DKM* generates a new *TEK* and sends it along with a *ready to rekey* message to all *AKM*s in the domain as depicted by Figure 4.25. Since the *M*1 was a member of area *An*, *AKMn* requires to update A_nEK_6 and replace it with A_nEK_7 to achieve forward secrecy at area level. As the area encryption key A_2EK_5 associated with area *A*2 is also

compromised by *M*1, it is necessary *AKM*2 replaces the A_2EK_5 with a new area encryption key A_2EK_6.

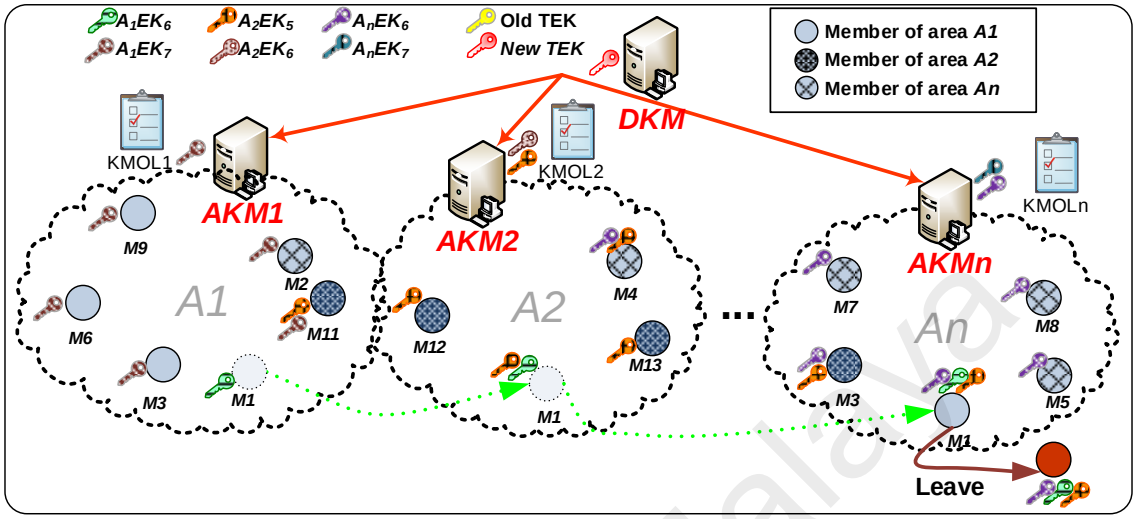

Figure 4.25: Rekeying process conducted with the *DKM* delivers the new *TEK* to all *AKM*s. A_2EK_5 and A_nEK_6 are respectively replaced with A_2EK_6 and A_nEK_7.

Figure 4.26 shows *AKMn* delivers the new *TEK* as well as new A_nEK_7 to the remaining members *M*3, *M*5, *M*7, *M*8 in the area *A*n using multiple unicast messages protected respectively under key MEK_3., MEK_5, MEK_7, and MEK_8. *AKM*2 has to use unicast message to deliver the new *TEK* and A_2EK_6 to all members *M*4, *M*12, *M*13 encrypted with respectively MEK_4, MEK_{12}, and MEK_{13} as the old area encryption key A_1EK_4 is disclosed with leaving member *M*1. A_1EK_7 in area *A*1 is sent by a multicast message encrypted with A_2EK_6 to all members *M*6, *M*9, *M*2, and *M*11.

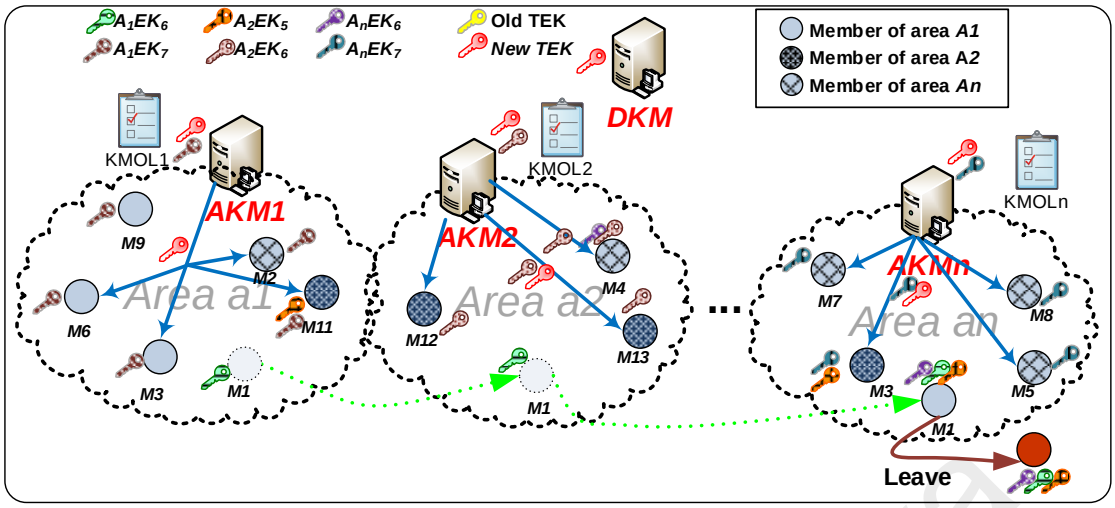

Figure 4.26: *AKM_2* and *AKM_n* require to send *A_2EK_6* and *A_nEK_7* along with new *TEK* using unicast messages in their area respectively *A_2* and *A_n*. *AKM_1* multicasts the new *TEK*.

As a conclusion, the scenarios showed the proposed group key management scheme treated the membership dynamics during join and leave events as well as location dynamic during the move event with the minimum rekeying messages overhead. Since group members are divided into several areas, the impact of rekeying processes is limited to the areas where the events occur. The move event affects only members residing in the visited area whenever it is not on the *KMOL* of the visited area in addition to the member's join time is after the last update of the *AEK* corresponding to the visited area. The single point of failure, possible delays, and signaling loads at the *DKM* is alleviated as the *AKM*s manage the mobility events independently.

4.7 Chapter summary

This chapter determined the scope of the proposed scheme (i.e. HIMOB), and identified the main components of the proposed architecture. The main protocols used for managing different events such as join, move, and leave in HIMOB have been described.

A novel mobility protocol was proposed to remove the limitation of the existing scheme and address host mobility issue.

Next chapter provides the analysis and simulation results of HIMOB.

CHAPTER 5: RESULTS AND DISCUSSION

This chapter provides analysis and evaluation of HIMOB according to metrics identified in Chapter 3 namely, rekeying message overhead and *1-affects-n* phenomenon.

Section 5.1 presents the simulation model, and provides a full analysis of simulation results of HIMOB in comparison with several related schemes from the literature. The size scalability of HIMOB is investigated in Section 5.1.1 and Section 5.1.2, where respectively the impact of the average inter arrival and the average of membership duration on HIMOB are studied. The impact of members' mobility inside the group on HIMOB is studied in Section 5.1.3. Finally, the security requirements of HIMOB in terms of backward and forward secrecy are discussed in Section 5.2.

5.1 Simulation and results

This section presents the results obtained through several simulation experiments. The HIMOB is compared with several related schemes from literature, namely KMGM (Said Gharout et al., 2012), GKMW (Mat Kiah & Martin, 2007), FEDRP (DeCleene et al., 2001), and LKH++ (Pietro et al., 2002). KMGM is a decentralized approach with independent *TEK* per each area (or subgroup), whereas GKMW, and FEDRP employ the decentralized approach with a common *TEK* for the entire group. LKH++ scheme is representative of a centralized approach designed for wireless mobile environments.

A two tier distribution hierarchy with distinct *A* areas (here *A* is equal to five) are designed for HIMOB, GKMW, and FEDRP. One *DKM* in the first tier is responsible for governing all *AKM*s as well as managing the common *TEK* for the whole group. Each area in the second tier is managed by an *AKM*. In KMGM scheme, there is no an explicit *DKM* and its responsibility must be delegated to the all existing *AKM*s in the group. The *DKM* has the main role of key management in LKH++ and entertains all events occurred

in the session. Thereby, the *AKM*s are not involved in key management and all requests are sent directly to the *DKM*.

In different experiment scenarios carried out in the simulation, the rekeying process follows a strict policy so that as soon as any changes occur in the group membership in terms of join or leave, the *TEK* requires to be updated within the entire group or in the affected area. Similar to the *TEK*, this strict policy is also applied for rekeying at area level to replace the old area encryption key with the new one. The rekeying policy for the move event in LKH++, since it does not provide an explicit mobility protocol, is assumed as a leave in the old area and subsequently a join in the new area.

The number of rekeying messages required for achieving backward and forward secrecy as well as the *1-affects-n* phenomenon are studied for all aforementioned schemes with different scenarios experimented in the simulation software. Take note that there is no discrimination between a unicast message and a multicast message for analysis of the rekeying messages overhead. The focus is only given on the number of messages sent for updating keying materials.

The rekeying messages overhead refers to the total messages distributed to update the keying materials for the provision of backward and forward secrecy upon any changes occur in the group membership. This requirement satisfies the bandwidth consumption of wireless networks and devices by HIMOB. The high number of messages transmitted either by unicast or multicast during the performing rekeying process consume enormous network bandwidth, which results in delays in the distribution of the keying materials and disruption in the group communication service.

The *1-affects-n* phenomenon is a scalability requirement for group key management scheme, which represents the total number of members affected with each event so that

need to replace their old keying materials with the new set. The high number of affected members by rekeying process becomes a hurdle to scale the scheme to large groups, particularly in wireless mobile environments where the flurries of key update increases the probability of transient security breaches and confusion.

5.1.1 Simulation model

The network topology is comprised of a grid 2000x1000 meters, which is then divided into 5 areas. The group members are located randomly in each area and join the group session (i.e. group inter arrival) according Poisson process with rate λ (arrivals / time unit). Each area is selected randomly with the same probability for entering the members. The join time rate can be constant (i.e. 10 seconds) or varying (i.e. 1 second to 15 seconds with incremental value 1) depend on the experiment scenario. If the inter arrival is set up with a constant value, the group communication will be populated with *n* members. Otherwise, the group population varies from small to large depend on the arrival rate specified by the user.

The join time is of 10 seconds based on the data analysis carried out on the information captured from MBone (Almeroth & Ammar, 1997). It was shown that the average inter arrival time into a group application with long running session such as Internet radio or Internet video is about 10 seconds (Almeroth & Ammar, 1997). Nevertheless, in the group based applications with a short session duration such as multimedia seminar, most of participants increasingly join the session just after the start time of the program's schedule (Almeroth & Ammar, 1997). In such group based applications, the participation strongly correlates to the start time and end time of transmission.

The group session duration for all experiments in this work is assumed constant and approximately proportional with 30 minutes (common duration for a program broadcasted by Internet video/radio) (Chan & Chan, 2002).

Once a member joins the group, its membership duration (or session sojourn time) follows an exponential distribution with a mean duration $1/\mu$ time unit. Analysis was done on two different sessions, one with long duration, such as the STS-63, and another one with short duration, such as the UCB seminar. The average membership duration is about 50% of the whole session duration (Almeroth & Ammar, 1997). With that, the membership duration is assigned with a constant value 15 minutes for different simulation experiments. This parameter is also used to study the impact of group size variation as one of the scalability requirements on the scheme rekeying message overhead and *1-affects-n* phenomenon while changing from 10 minutes to 25 minutes.

The member remains in each area for a determined time (referred to as area dwell time), and then moves to the other areas with the same probability of selection. In order to study the impact of member's mobility on the performance overhead, the area dwell time vary between 1 and 15 seconds ([1s: 15s]). Reducing the area dwell time leads to increasing the members' mobility rates among areas. The velocity of members is set constant for all experiments which is 5 m/s. This velocity is assumed for foot speed or urban vehicle speed.

5.1.2 Impact of the group size

The scalability size of HIMOB is studied by changing the value of two controlling parameters in the simulation:

1. the average inter arrival into the group session (i.e. a sequence of time duration after which a member joins the group), and
2. the average membership duration in the session (also called session sojourn time), which is a time length that a member remains in the group communication.

For the first scenario, the average inter arrival value varied in the simulation experiments from 2 second to 30 seconds ([2s : 30s]), which make a group respectively with a big size and small size. The first simulation experiment was conducted for a session in which the members arrive after each 2 seconds. This experiment again repeated for the other average inter arrival till the value of inter arrival reaches to 30 seconds. Then, the number of rekeying messages required for updating the keying materials and the number of affected members because of any event occurred in the group is measured to compare the performance of HIMOB with the others. The simulation parameters are summarized in Table 5.1 for this experiment.

Table 5.1: Simulation parameters for the experiment scenario of varying inter arrival time.

Parameters	Value
Number of Area	5
Session time	30 Min
Inter-arrival	[2 : 30] Sec.
Session sojourn time	15 Min
Area dwell time	10 Sec.
Velocity	5 m/Sec

The two (2) seconds minimum inter arrival time is chosen to reflect a member joining a group at the start time of the group session where this scenario normally covers the multimedia seminar or webinar. And maximum inter arrival time of 30 seconds is considered for the group based such as IPTV or Internet video system. Hence, in this work [2 : 30] seconds inter arrival time are used.

Another controlling parameter that influences the size of a group is the average membership duration (i.e. session sojourn time). The increase of membership duration causes the rise of the remaining time length in the session for all group members. As a result, the number of leave events reduces and consequently the group population

increases. In the second scenario, the membership duration is first assigned with a value of 10 minutes. The membership duration value increases one minute more in subsequent simulation experiments until it reaches to 25 minutes. The 25 minutes is roughly equal to 85% of the entire session time which is acceptable for a member to stay in a particular session (Almeroth & Ammar, 1997; P. Guo, 2013; P. J. Guo, Kim, & Rubin, 2014; ON24 Inc., 2016). The simulation parameters and the corresponding values used in this experiment are summarized in Table 5.2.

Table 5.2: Simulation parameters in the experiment scenario of varying the membership duration.

Parameters	Value
Number of Area	5
Session time	30 Min
Inter-arrival	10 Sec
Session sojourn time	[10 : 25] Min
Area dwell time	10 Sec
Velocity	5 m/Sec

5.1.2.1 Impact of inter arrival variation on average number of events

The average cumulative number of events such as join, move, and leave occurred during a session is illustrated in Figure 5.1. The average rate of join gradually reduces when the time that members join the session (inter arrival period) increases. The average number of participants in the group session reaches to 600, when the inter arrival value equals 2 seconds. This rate reduces with the decrease of the inter arrival rate (increase of inter arrival time) to achieve an average of 60 members joining the session.

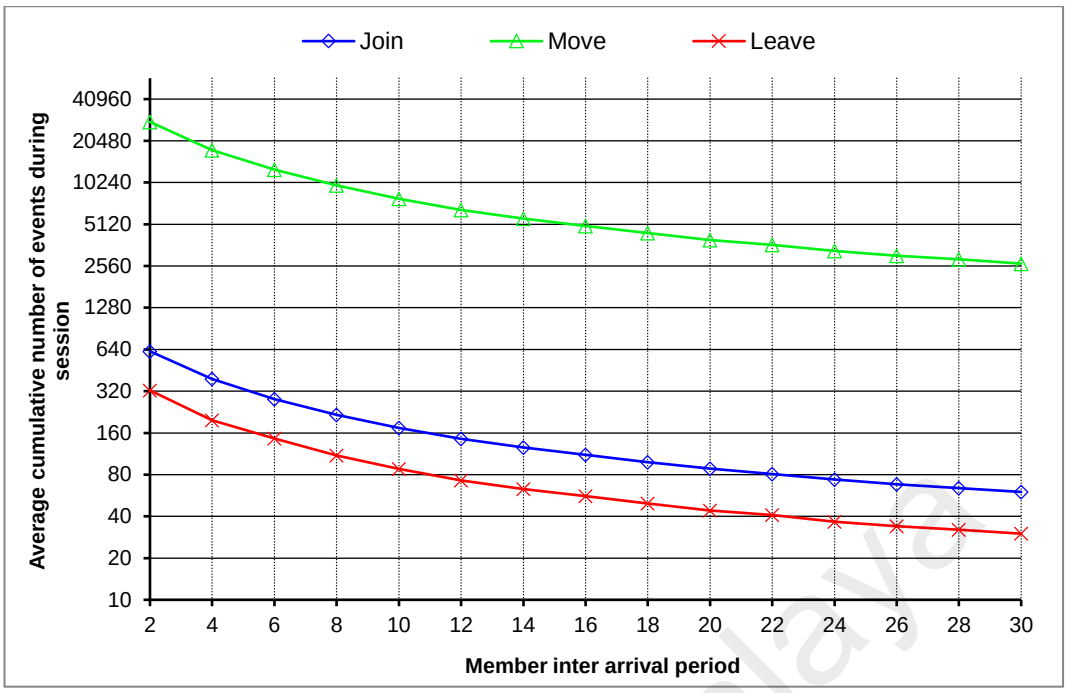

Figure 5.1: Average number of events occurred in the session.

Similar to the join rate, the average rate of leaves from 324 per session gradually decreases to 30 leaves per session while the time duration for joining the group session increases. As the membership duration (i.e. session sojourn time) is constant in this experiment, the leave rate follows the join trends. The area dwell time value remains steady equal to 10 seconds, so the inter moves between areas varies according the population of members in the group. The mobility rate reaches 28000 when the group has the biggest size (the inter arrival is equal 2 second), whereas the mobility rate achieves 5600 when the group population has the smallest size.

5.1.2.2 Impact of inter arrival variation on rekeying messages overhead

Figure 5.2 depicts the ratio of rekeying cost for HIMOB, FEDRP, GKMW, and LKH++ varying the time of member arriving in the session from 2 seconds to 30 seconds [2s : 30s]. It can be easily seen that the average number of messages per each event during the session for HIMOB is smaller than the other solutions. From Figure 5.2, the schemes introducing a mobility protocol show better performance with reducing the average

number of rekeying messages even when group size becomes large and mobility rate increases.

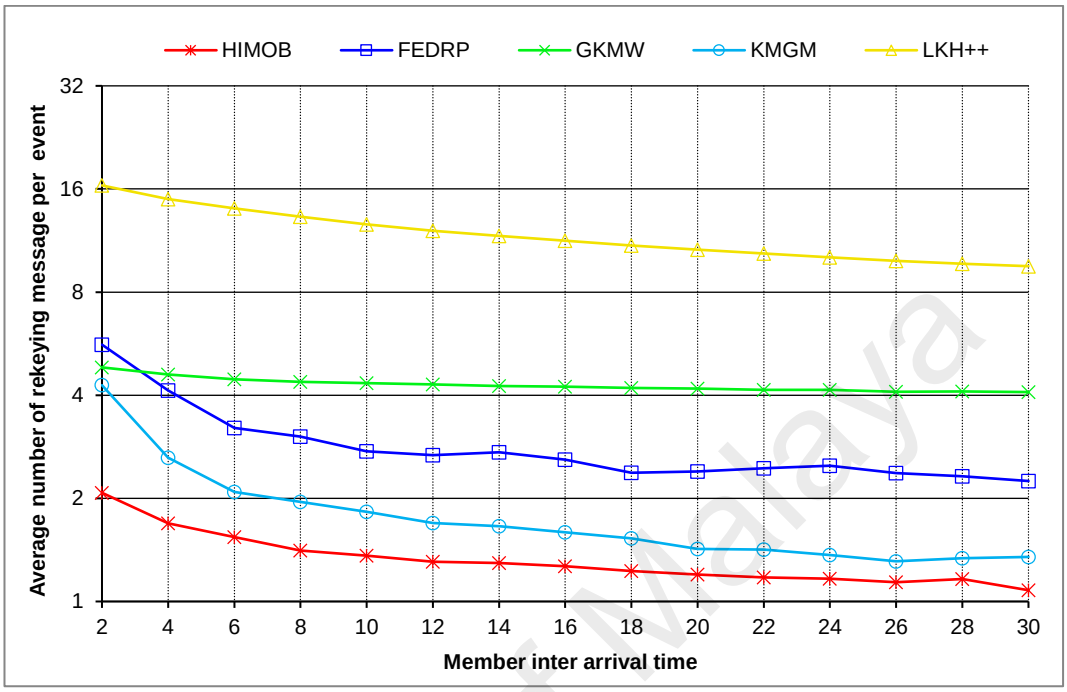

Figure 5.2: Impact of inter arrival variation on average number of rekeying messages per event.

The number of rekeying messages is influenced by the group size in LKH++. When the group population grows up, the rekeying messages required for managing the move event significantly increase as this scheme treats a member's move as a leave in the old area and a join in the new area. In contrary, HIMOB minimizes the rekeying messages overhead in comparison to the other schemes. It is due to the authentication mechanism used by the *AKM*s for verifying moving members, which avoids initiating extra rekeying messages.

Figure 5.3 depicts the total number of rekeying messages sent during the session. The total rekeying messages consist of all messages that deliver the keying materials, including the *TEK* and *AEK*s to all members of a group.

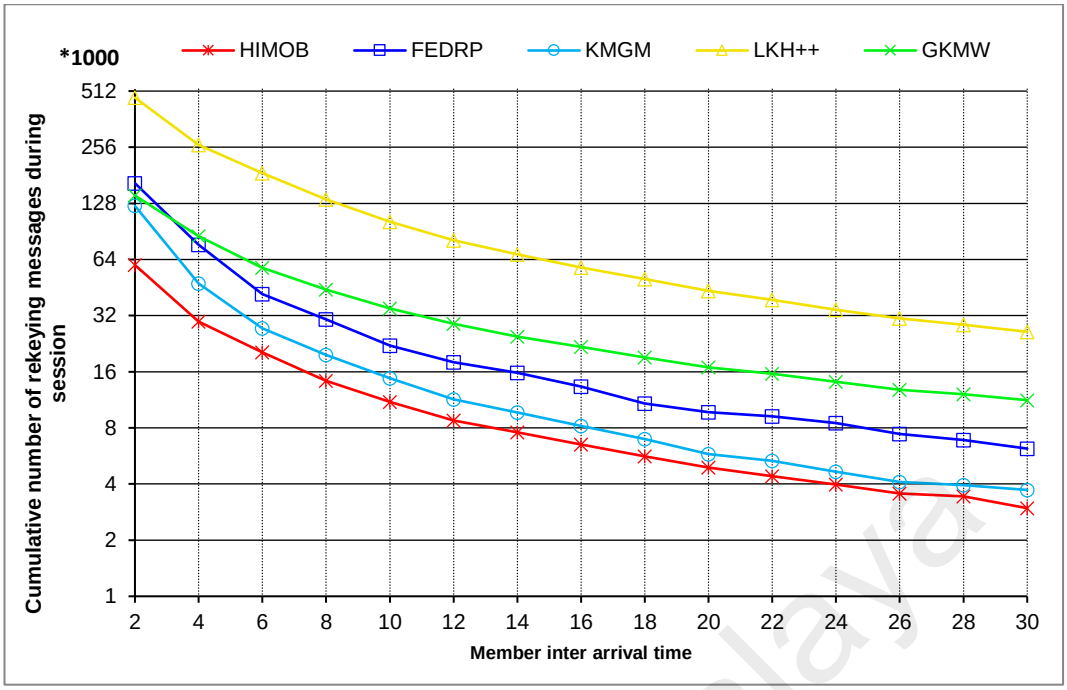

Figure 5.3: Impact of inter arrival variation on average cumulative number of rekeying messages during session.

5.1.2.3 Impact of inter arrival variation on *1-affects-n* phenomenon

The average number of affected members per occurred events in the session is illustrated in Figure 5.4 for the HIMOB, KMGM, FEDRP, GKMW, and LKH++ while the join period varies from 2 seconds to 30 seconds. In Figure 5.4, HIMOB treats the *1-affects-n* phenomenon with the low overhead in comparison to other schemes particularly in the group with a large size. However, the number of affected members are approximately the same for HIMOB, KMGM, and GKMW while the group members joining the session in each 15 seconds or above.

The average of affected members is quite higher for LKH++ as the mobility event is treated as a leave in the old area and subsequently a join in the visited area. Therefore, number of members who need to update their keying materials increases. FEDRP shows fairly higher average of affected members between the schemes that have a protocol for managing the member mobility. It is because that FEDRP empties the mobility list of

each area where there is a group membership change. Thus, in some cases returning back members are treated as members that visit the area for the first time, and extra rekeying processes are performed to preserve backward secrecy, which result in more members being affected in the visited area with move event.

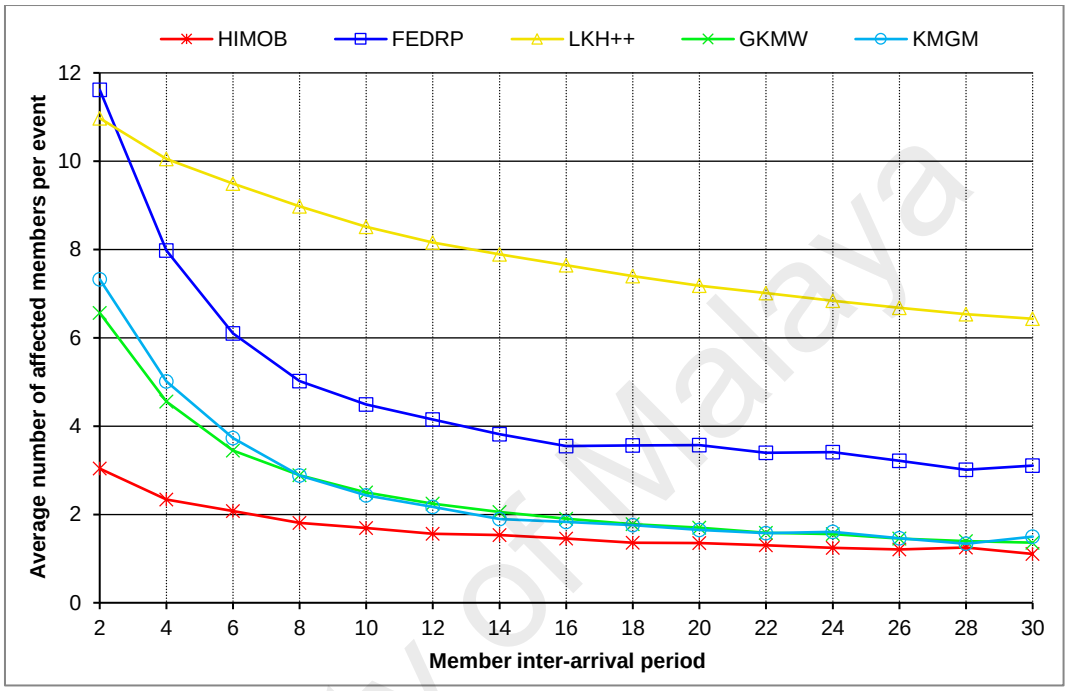

Figure 5.4: Impact of inter arrival variation on average number of affected members per event.

Figure 5.5 depicts a comparison of total members affected by the rekeying process in the session. This comparison illustrates that the average number of affected members considerably drops when the inter arrival time increases. HIMOB, KMGM, and GKMW show approximately close results when the inter arrival rate goes over 20 seconds. In spite of this, the strength of HIMOB is to significantly minimize 1-affects-n phenomenon in the group where members enter with the inter arrival time ranging from 1 second to 10 seconds. Therefore, HIMOB can scale to large group better than other schemes.

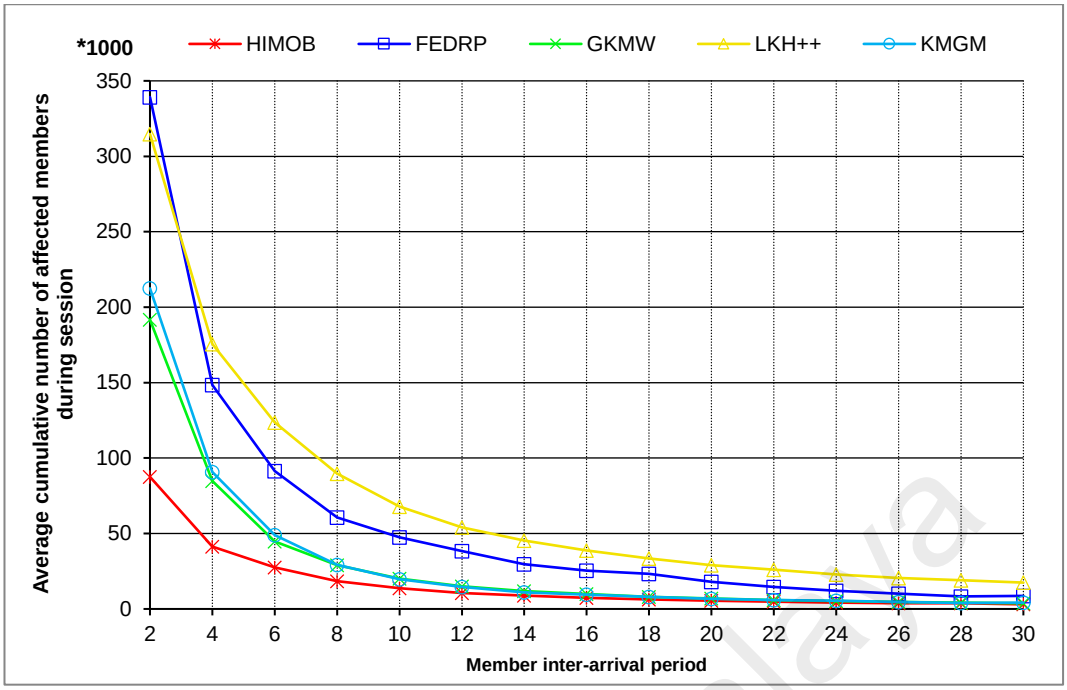

Figure 5.5: Impact of inter arrival variation on average cumulative number of affected members during session.

5.1.2.4 Impact of membership duration on average number of events

The average cumulative number of events such as join, move, and leave taken place during a session is illustrated in Figure 5.6 while the average membership duration (i.e. the time unit duration that a member remains in a group communication) increases ranging from 10 minutes to 25 minutes. The membership duration (i.e. session sojourn time) is the number of time units (in minutes) after which a member leaves the session. The membership duration is the reverse of leave rate so that the increase of membership duration result in reduction in the leave rate.

In Figure 5.6, when membership duration is equal to 10 minutes, the average rate of leaves in the session reaches to about 230. This rate gradually declines to achieve an average of 50 when the membership duration reaches to 25 minutes. Nevertheless, the average rate of join remains steady about 330 during session as the members join the group in each 10 seconds.

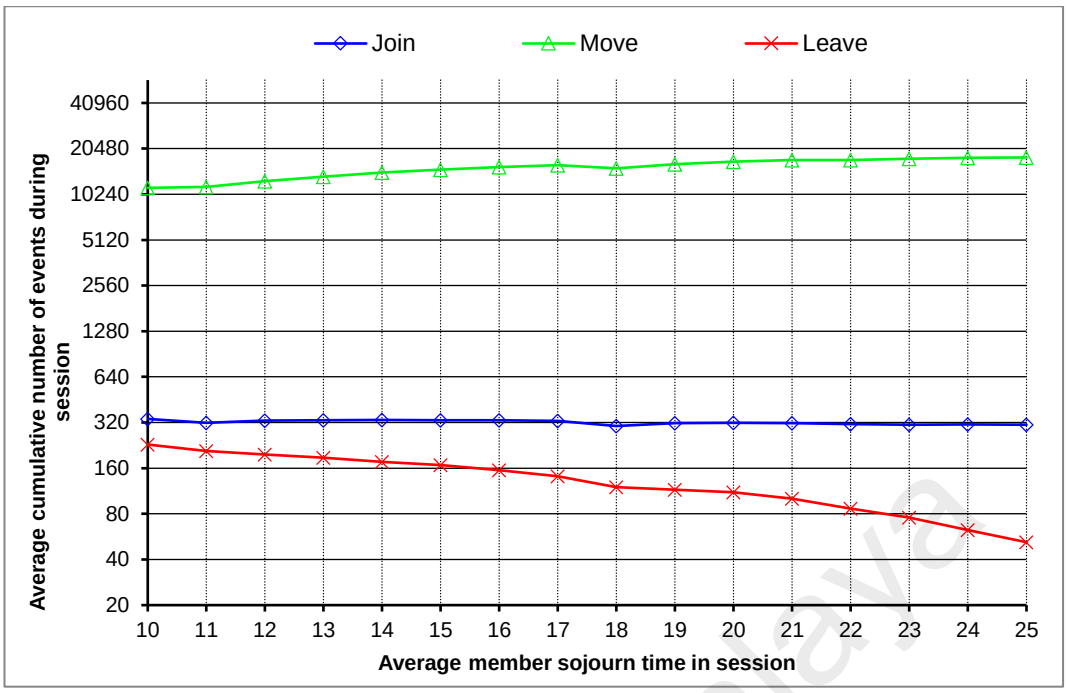

Figure 5.6: Average number of events occurred in the session.

The total average number of mobility rate reaches to around 10000 when the membership duration is 10 minutes, and the rate moderately rises up with increase of membership duration to 25 minutes to achieve an average of 18000. The increase of mobility rate is a result of the reduction in the leave rate, which leads to more members remaining in the session.

5.1.2.5 Impact of membership duration on rekeying overhead

Figure 5.7 depicts the ratio of rekeying overhead for HIMOB, KMGM, FEDRP, GKMW, and LKH++, with average membership duration ranging from 10 minutes to 25 minutes ([10 mins: 25 mins]). In Figure 5.7, the schemes that take address the mobility issue such as the HIMOB, KMGM, GKMW, and FEDRP show less rekeying messages overhead in comparison to LKH++. Due to the lack of a key management protocol for member mobility in LKH++, whenever a move event occurs in the session the rekeying process must be performed twice as the move event is treated as a leave and a new join. Since the increase of membership duration increases the group size as well as mobility

rate in the session, LKH++ induces more rekeying messages overhead as the number of rekeying processes increases according the growth of group population and the mobility rate.

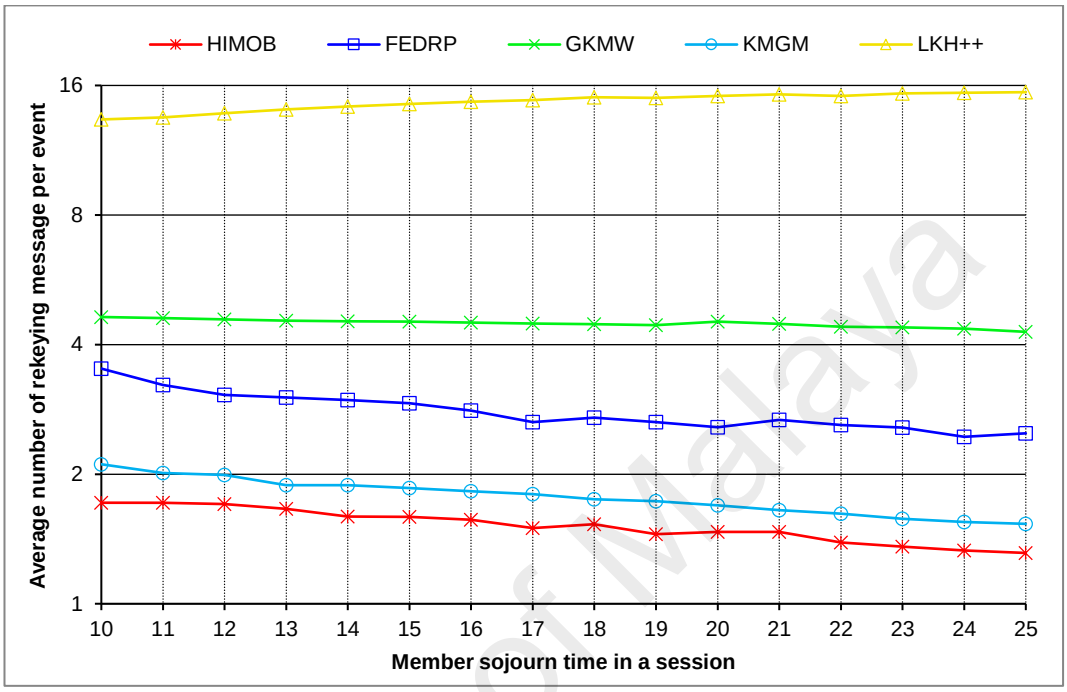

Figure 5.7: Impact of membership duration variation on average number of rekeying messages per event.

HIMOB performs the smallest rekeying process in comparison to the other solutions as it only updates the keying materials particularly in move event on condition that the join time of a member is after the last time of updating keying materials in the visited area.

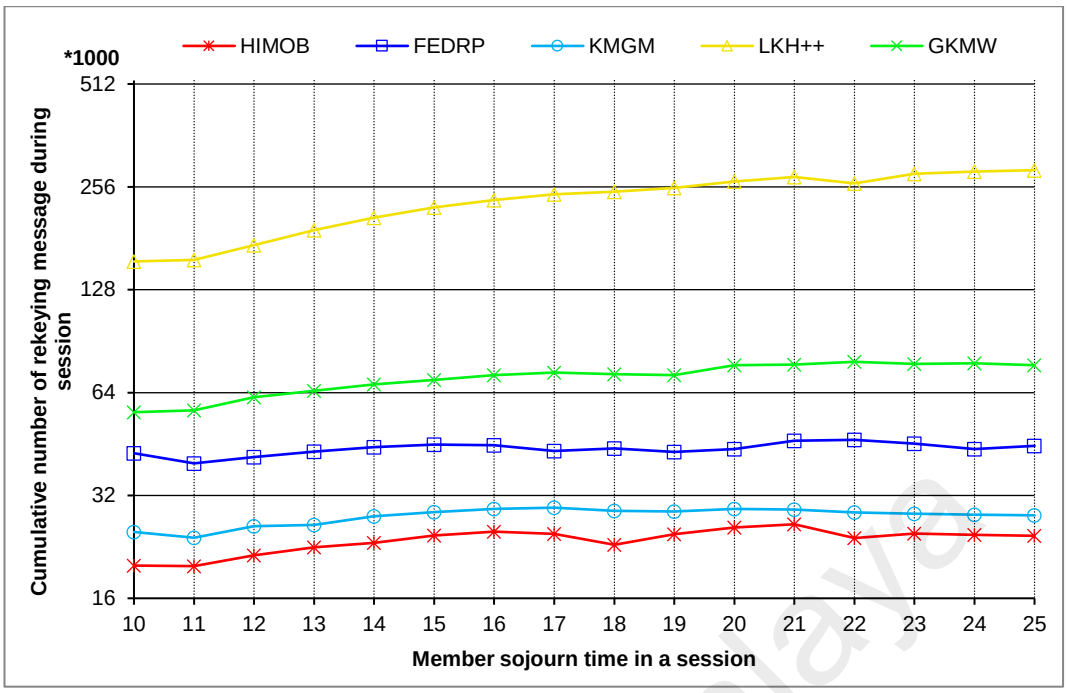

Figure 5.8: Impact of membership duration variation on average cumulative number of rekeying messages during session.

Figure 5.8 illustrates the total number of rekeying messages sent to update the keying material when any event such as join, move, and leave occurs in the session. It can be remarked the total number of rekeying messages in HIMOB, KMGM, FEDRP, and GKMW is significantly less than LKH++. This is due to the fact that LKH++ involves the both old and visited area in move event in rekeying processes.

5.1.2.6 Impact of membership duration on *1-affects-n* phenomenon

The impact of membership duration variation on *1-affects-n* phenomenon is depicted in Figure 5.9 and Figure 5.10. Figure 5.9 illustrates the average number of affected members per event in the session. It is remarked that HIMOB and KMGM show the least overhead on *1-affects-n* phenomenon. This is because that HIMOB and KMGM minimize performing rekeying process during move event. HIMOB carries out the rekeying process when the move member is not on the mobility list and its join time is after the last keying

materials update time in the visited area. KMGM avoids rekeying process in move event, however it suffers from backward secrecy in such events.

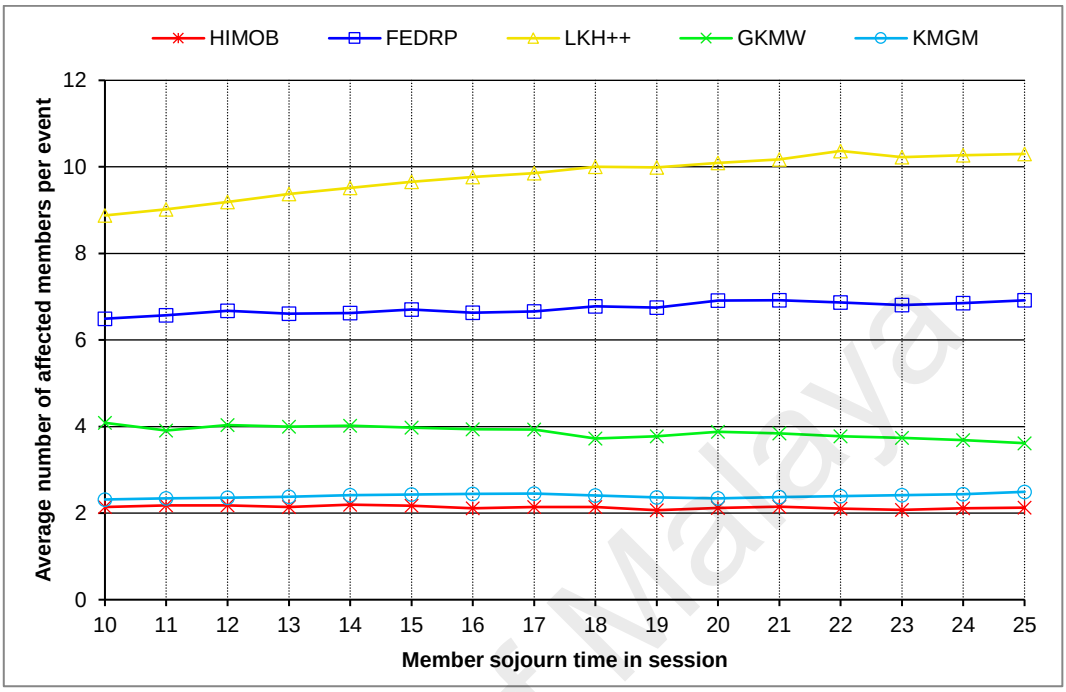

Figure 5.9: Impact of membership duration variation on average number of affected members per event.

GKMW shows less overhead in terms of *1-affects-n* phenomenon than FEDRP because it keeps the track of moving members in the mobility list and does not empty the mobility lists. When a member moves back to an area where s/he has previously visited, the *AKM* avoids to run extra rekeying process for the returning member. Thereby, the number of group members affected by move event reduces. But FEDRP empties the mobility list on each event, which causes the *AKM*s miss the track of moving member and initiates unnecessary rekeying process which affects more members. The lack of protocol for managing move event shows the high average number of affected members in LKH++ as the members are affected twice when a move event occurs.

Figure 5.10 depicts the average total number of affected members in the session varying membership duration from 10 minutes to 25 minutes. HIMOB and KMGM show

better scalability to the large group when the membership duration varies as the average number of members affected in different events is the smallest. The average number of affected members significantly increases in LKH++ with increasing the size of group since all members residing in the old area and the new area are affected in each move event.

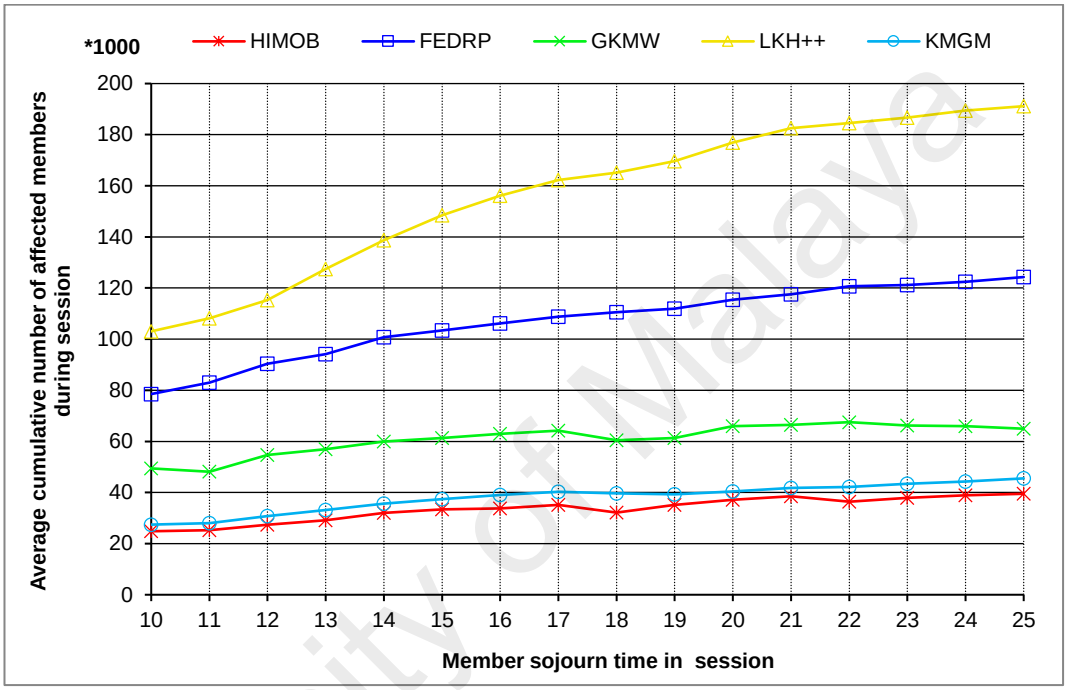

Figure 5.10: Impact of membership duration variation on average cumulative number of affected members during session.

5.1.3 Impact of mobility rate

This section investigates the impact of member mobility on the rekeying messages overhead and *1-affects-n* behavior. To study the impact of mobility rate variation, the area dwell time varies between range from 1 second to 15 seconds ([1s : 15s]) in the simulation experiments. The area dwell time can be called mobility period which is the number of time units (in seconds) after which a member changes its location. The mobility period is the reverse of mobility rate (equal to $\frac{1}{\text{mobility rate}}$) which is the average number of moves on time unit. The smallest value of the area dwell time corresponds to a high mobility

scenario; which members are completely considered as mobile with no tarriance in each area.

In order to reduce the number of mobility events occurring in the session for this experiment, the area dwell time (i.e. mobility time) varies between 1 second and 15 seconds. This is due to the fact that in tactical group based applications established in battlefield or disaster area scenario, a group of troops or a rescue team members may experience different dwell time in different areas during the accomplishment of the assigned task. The dwell time equal with 1 second reflects the scenario that the group members move to the target without any stop for achieving their mission. If the team members have some stop in different areas, it can be experimented with dwell time equal to 15 seconds.

In this scenario, the network topology is a 1000x1000 grid. The members join the group in each 10 second (i.e. [inter arrival time = 10 seconds]) in all experiments. The membership duration in the session is the same for all group members equal to 15 minutes. The members move between areas according random way point model with velocity 5m/s. The simulation parameters used in this scenario experiments are summarized in Table 5.3.

Table 5.3: Simulation parameters in experiment scenario of varying mobility rate.

Parameters	Value
Number of Area	5
Session time	30 Min
Inter-arrival	10 Sec
Session sojourn time	15 Min
Area dwell time	[1 : 15] Sec
Velocity	5 m/Sec

5.1.3.1 Impact of mobility rate variation on number of events

Figure 5.11 depicts the average number of total events such as join, move, and leave occurred in a session with the mobility period ranging from 1 second to 15 seconds ([1s : 15s]). When the mobility period value is equal to 1 second, the average rate of moves in the session reaches to about 19000. In other words, each area in the experiment is approximately visited with 3800 members. The number of move event achieve about 5900 in the session when the mobility period reaches to 15 seconds.

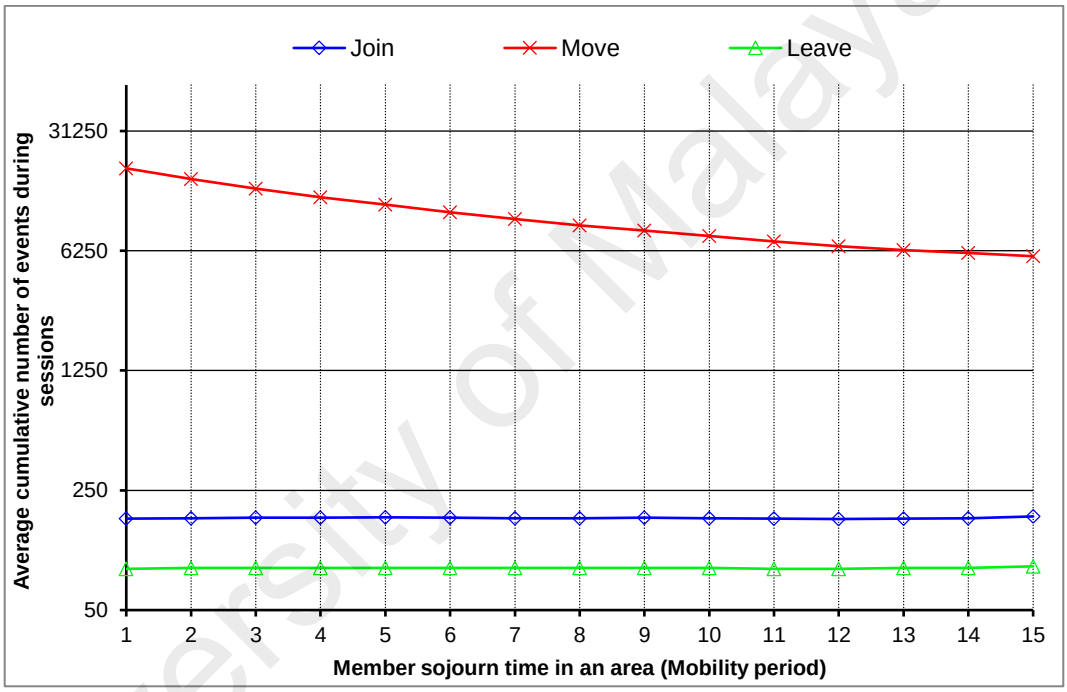

Figure 5.11: Average number of events occurred in the session.

In spite of mobility rate variation during the session, the rate of joins and leave remain steady. A new member joins the session whenever the inter arrival timer (which is equal to 10 seconds) expires. Thus, the average rate of joins in a session is the same for all experiments about 172. When the membership duration of a member expires, it has to leave the session. Since the membership duration is a constant value equal to 15 minutes, the average number of leaves events in all experiments is about 88.

5.1.3.2 Impact of mobility rate variation on rekeying overhead

Figure 5.12 illustrates the ratio of rekeying messages overhead for HIMOB, KMGM, GKMW, FEDRP, and LKH++ with mobility period ranging from 1 seconds to 15 seconds ([1s : 15s]). The increase of mobility rate has a minimum influence on HIMOB and KMGM since the rekeying messages overhead is less than other schemes. This is because of the authentication mechanism used for verification of the moving members which avoid initiating extra rekeying process in move events. However, KMGM suffers from backward secrecy in move event as the member may access to the security information of the visited area which has been valid before the time the moving member joined the group.

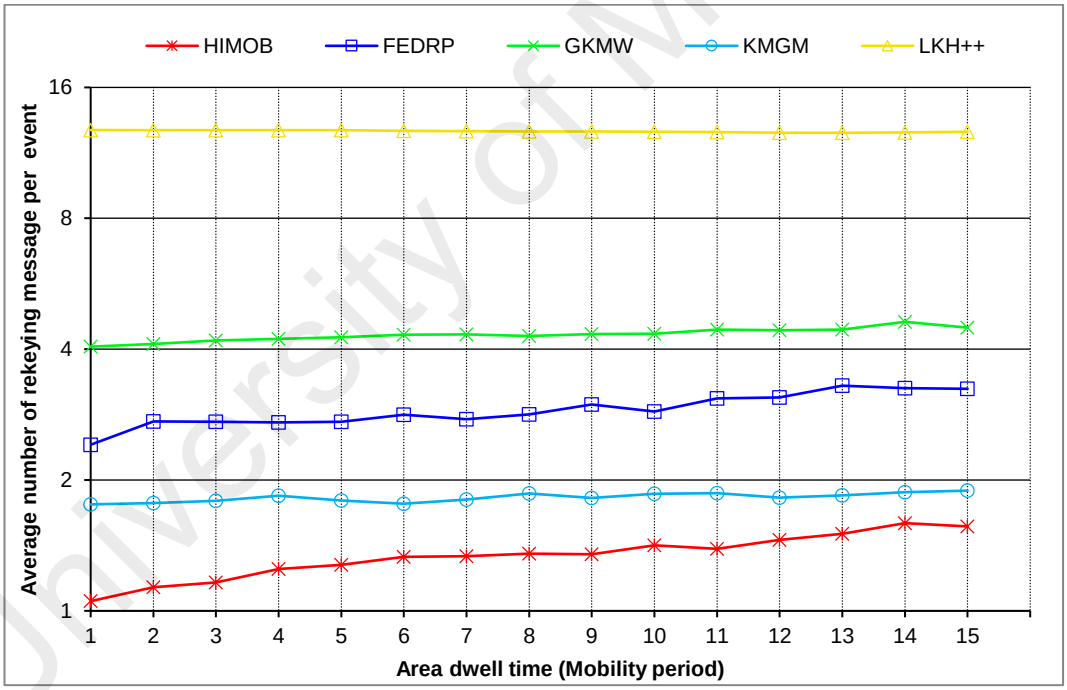

Figure 5.12: Impact of mobility rate variation on average number of rekeying messages per event.

GKMW presents higher rekeying messages overhead in comparison to the other schemes taking address the mobility issue. It is because that the moving member needs to establish a session key with the *AKM* of the visited area before movement, which leads

to increasing the number of messages. The *DKM* is responsible for generating this key and sending it to the moving member as well as the visited *AKM*. LKH++ shows the highest rekeying messages overhead again because of the naïve solution for managing move event.

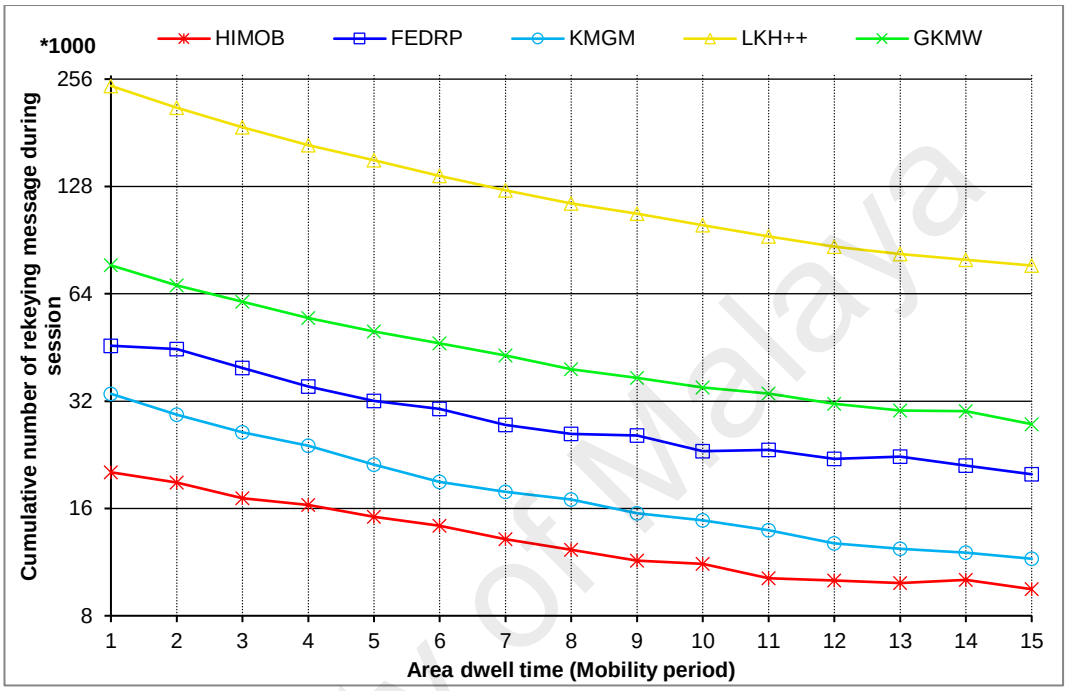

Figure 5.13: Impact of mobility rate variation on average number of cumulative rekeying message during session.

Figure 5.13 illustrates the total number of rekeying messages distributed throughout the domain in order to update the keying material when any event occurs in the session. It can be easily seen the total number of rekeying messages dramatically reduces when the moving members have a tendency to become motionless. The impact of mobility variation on the total rekeying messages overhead is significant in LKH++ particularly in more dynamic environments where the members tends to be more mobile since the increase of mobility escalates the number of keying materials updates. The authentication mechanism used for verification of moving members in HIMOB reduces considerably the rekeying messages overhead.

5.1.3.3 Impact of mobility rate on *1-affects-n* phenomenon

Figure 5.14 depicts the ratio of affected members in a session for HIMOB, FEDRP, GKMW, and LKH++ with the mobility time ranging from 1 second to 15 seconds. Adopting the same strategy by introducing the use of mobility list for keeping track of moving members in HIMOB, KMGM, GKMW, and FEDRP eliminates needs of performing rekeying process in the old area during move events, which result in minimizing the *1-affects-n* phenomenon overhead. This is not the case of LKH++ which suffers the lack of a strategy to manage the move events while affecting the group members twice.

It is remarked that the impact of mobility rate variation on *1-affects-n* phenomenon in HIMOB is considerably smaller than the other solutions because the rekeying process is performed in the visited area as long as the visiting member is not on the mobility list or its join time is after the last time of keying materials update in the visited area.

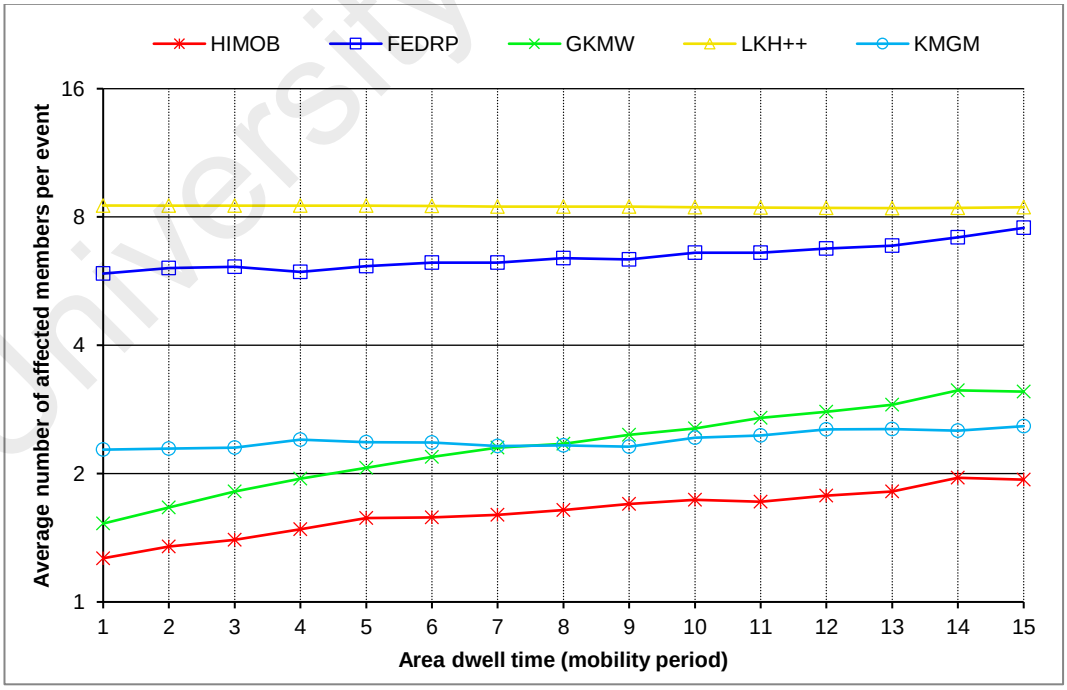

Figure 5.14: Impact of mobility rate variation on average number of affected members per event.

Figure 5.15 depicts the average total number of affected members in the session with mobility period ranging from 1 second to 15 seconds. The total number of affected members gradually reduces in all solutions when the mobility time increases. However, this reduction becomes noteworthy in FEDRP and LKH++. Applying a naïve solution for move event in LKH++ so that such events are treated as a leave in the old area and a new join in the new area shows high *1-affects-n* phenomenon overhead particularly in very dynamic environments where mobility period is less than 5.

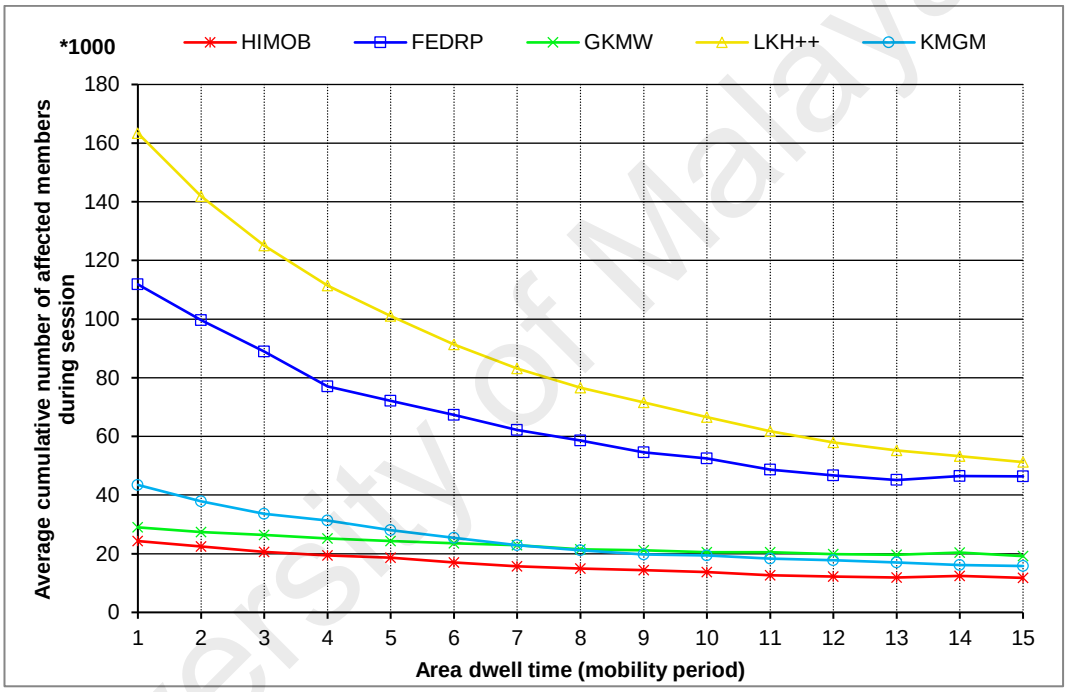

Figure 5.15: Impact of mobility rate variation on average cumulative number of affected members during session.

The number of affected members on session controlled with GKMW, KMGM are slightly close to HIMOB because of the similar strategy for maintaining the mobility list in all aforementioned schemes. The mobility list is not emptied during the session in GKMW unless the policy defines otherwise thus, *AKM*s do not lose the track of members and avoid performing extra rekeying process especially when a moving member return back to an area. KMGM empties the mobility list only when a moving member leaves the

group and its track is on the mobility list. HIMOB affects minimum members in comparison to the other because it affects the members in the visited area by performing rekeying process on condition that the join time of the moving member is after the update time of *AEK* in the visited area.

5.2 Security analysis

This section presents the analysis of HIMOB security requirements in terms of backward secrecy, and forward secrecy. The backward secrecy is needed to ensure that either a new member who joins the group or a moving member who changes its area is prevented from having access respectively to messages that were sent in the group or in the visited area prior to its membership. In contrary, the forward secrecy ensures the leaving member is denied access to future messages sent to the group and different areas not only in the area where the leave occurs, but also in the other area where has already been visited by the leaving and their keying materials are possessed by the leaving member. However, forward secrecy is not required in the old area (i.e. where a member is moving from) since the member is still in the session. These properties are important as they provide the secrecy of traffic in the group.

Each of which is discussed further in the following subsections.

5.2.1 Backward secrecy

The proposed scheme precludes any eavesdropping opportunity when a moving member changes its location. The provision of secrecy with respect to backward secrecy is achieved with performing rekeying process in the visited area, which result in the moving member being unable to discover the service security information before it joined the group in the visited area. This also leads to all group members residing in the visited area obtaining new area encryption key *new_AEK*.

AKM_j, area key manager of the visited area *j*, generates the new area encryption key *new_AEK* when the moving member M_i is not in the $KMOL_j$ and its joining time is after the last update of AEK_j. AKM_j sends *new_AEK* to group members in area *j* with a multicast message protected under the old area encryption key *old_AEK*. To securely deliver new_AEK_j to the visiting member M_i, AKM_j is able to independently derive the MEK_i of the visiting member upon receiving M_i's move notification, and then use it to encrypt the *new_AEK*. If the last update of old_AEK_j had been performed at t_1 and the visiting member joined the group in area *i* at time t_2, the visiting member cannot access to the security information of area *j* between time t_1 and t_2.

When a new member joins the group communication, rekeying process is performed in the area where join event occurs. This results in the new member and other members in the area receive new keys *new_AEK*, and *new_TEK*. This also results in other group members across the domain receiving the *new_TEK*. Therefore, the new member is prevented from having access the previous security information in the area and the domain.

HIMOB has provided this option for backward secrecy, which can be managed using member mobility protocol as well as member joining protocol.

5.2.2 Forward secrecy

Preserving forward secrecy for the transfer of a group member from one area to another is not necessary. When a member moves from one area to another, the area where the member is moving from (old area) does not need to perform rekeying process for updating area key encryption. This is due to that the moving member is still remaining in the group communication despite of chaining its location from one area to another within a domain. Performing the rekeying process in the old area for provision of forward secrecy results in generating extra rekeying messages and increasing overhead during the move event.

As a result, the mobility protocol of HIMOB avoids preserving forward secrecy in the old area during move event.

When a member leaves the group, the rekeying process needs to be performed for provision of forward secrecy. The leave protocol of HIMOB provides this option in the area where a leave event occurs as well as all areas where have been already visited with the leaving member and their *AEK*s are still valid since the last member's visit. This results in all remaining members in an area where the leave event occurs and all members in the areas where the leaving member has previously visited and holds their valid *AEK*s obtaining *new_AEK*. This also results in all remaining group members in the domain obtaining a new traffic key *new_TEK*.

When AKM_i receives the leave notify message from M_i, it initiates rekeying process for its area encryption key AEK_i. The new_AEK_i is sent to all remaining members in area *i* via unicast messages protected under *MEK*s. This results in the leaving member preventing from having access to future security information in area *i* (i.e. provision of forward secrecy). If the leaving member has been logged in other area managers' $KMOL_p$ and it also holds a valid AEK_p, AKM_p initiates rekeying process for the area encryption key AEK_p. This results in all group members in areas *p* obtaining new_AEK_ps, and also achieving forward secrecy in area *p*. On receipt of the leave notify message from AKM_i, *DKM* performs rekeying process for *TEK*. This results in all group members in the domain obtaining a new traffic key *new_TEK*. Thereby, forward secrecy is achieved by preventing the leaving member from having access to future group communication.

5.2.3 Security analysis summary

Table 5.4 shows a comparison between HIMOB, KMGM, FEDRP, GKMW, and LKH++ in terms of backward secrecy and forward secrecy. The maintenance of backward or forward secrecy is indicated with a ✓ notation, otherwise it is indicated with a ✗. From

Table 5.4, all schemes except KMGM preserve backward secrecy when a join event or move event occurs in the session. In KMGM, if authentication phase for a moving member in the target area proceeds successfully, the member receives the keying materials of the visited area from the key manager. In some cases, the moving member may have access to the security information of the visited area which is valid before the time the moving member joined the group, which impose an expense of backward secrecy violation.

Table 5.4: Comparison of rekeying *TEK* and *AEK* in move event between different schemes.

Scheme	Forward secrecy		Backward secrecy	
	Leave	Move	Join	Move
HIMOB	✓	×	✓	✓
KMGM	✓	×	✓	×
FEDRP	✓	×	✓	✓
GKMW	×	×	✓	✓
LKH++	✓	✓	✓	✓

GKMW breaches forward secrecy in leave events since the rekeying process is only performed in the area where the leave occurs, while the leaving member may carry the valid keying materials associated with the other areas which has already been visited. Provision of forward secrecy in the old area in the transfer of a member from one area to another is pointless since the moving member is still remaining in the session. HIMOB, KMGM, FEDRP, and GKMW avoid performing rekeying of area encryption key in the old area whereas, LKH++ has to carry out rekeying process. It is due to the fact that LKH++ treats move event as a leave in the old area and a join in the new area. This results in generating unnecessary rekeying messages in the old area and increasing the overhead

of the scheme in terms of rekeying messages overhead and *1-affects-n* phenomenon (as discussed in Section 5.1).

HIMOB effectively achieves security requirement in terms of backward and forward secrecy. The backward secrecy is preserved during both the join and move event, and the forward secrecy is maintained during the leave event in HIMOB. Since the forward secrecy in the old area is not necessary during the move event, it is skipped to increase the efficiency of HIMOB.

5.3 Chapter summary

HIMOB has been assessed in terms of rekeying messages overhead and *1-affects-n* phenomenon in this chapter. The impacts of group size and mobility rate on the aforementioned overheads have been studied on the proposed group key management scheme. In comparison with the other solutions such as KMGM, FEDRP, GKMW, and LKH++, HIMOB has provided better performance and scalability in dealing with member's mobility in secure group communication thanks to reducing both the number of rekeying messages and the number of affected members. HIMOB has shown that can meet the specified security requirements in terms of backward and forward secrecy.

CHAPTER 6: CONCLUSION

This chapter concludes our work and provides direction for future work.

Section 6.1 presents the conclusion of this thesis. The research objective achievements are presented in Section 6.2. The limitations encountered throughout carrying out this research is given in Section 6.3. Finally, several suggestions for future work are outlined in Section 6.4.

6.1 Conclusion

The demands for development of new group based applications such as multimedia conferencing, interactive group games, video on demand, and IP TV has been given rises as kind an efficient communication in conjunction with the advances in Internet technology. Such applications have a strong need for communication security, which is conventionally supplied through the implementation of appropriate cryptography mechanisms.

A group key management scheme is an essential building block to ensure the security of group communications so that only members of the group can read data. Several group key management schemes have been intended for deployment in wired networks, while few attempts have been made to defeat the shortcomings in handling dynamic of a group in wireless mobile environments. Wireless mobile networks have increasingly prevailed recently among networked devices, thus available applications in wired environments should also be made available in wireless environments. With considering the tendency of network devices to have mobility, it is thus important to develop security techniques for secure group communication in wireless environments.

The overall goal of this research is to address the host mobility issue by developing a key management scheme for secure group communication in wireless mobile

environments. The scheme considered providing backward secrecy where mobile members dynamically perform changes in their locations while still remaining in the session. HIMOB used a new rekeying strategy based on key derivation function and KMOL for effectively performing key management and authentication phase respectively during move events. A decentralized approach with the common *TEK* is adopted to avoid extra encryption and decryption at the edge of the area and mitigate computation overhead as well as *1-affects-n* phenomenon. By delegating the key management and authentication phase to the intermediate *AKM*s massively reduced signaling load on the core network, hence giving scalability while preventing bottlenecks.

The investigation was commenced by looking at the existing group key management schemes, and the type of design approaches under which the schemes can be categorized. Three design approaches were identified, such as centralized, decentralized, and distributed (contributory), and the advantages and disadvantages associated with each approach were discussed particularly when they are extended to wireless mobile environments.

While many key management issues for secure group communication are generic to any networking environments, the primary constraints and challenges were identified as critical factors which must be considered for establishing secure group communications in wireless mobile environments.

Based on the identified specifications for group key management in wireless mobile environments, the investigation was continued by looking at the few solutions that addressed the mobility issue of group members, and identified the remaining weaknesses pertaining to each scheme. Based on the evaluation results, a group key management scheme was then proposed and designed for the provision of secure group communication

suitable for infrastructure-based wireless mobile environments. A number of notable features of the proposed group key management scheme are as follows:

It draws on several design properties of previous group key management schemes facilitate scalable key management. In particular, the architecture is based on the notion of the domain and area to mitigate the impact of rekeying processes.

The use of the list was introduced by the proposed scheme to support member mobility. The *KMOL* enables a key manager to keep track of a moving member, and manages member mobility more efficiently. When a member moves into an area, the area key manager can determine whether the moving member is a returning member or a first time visitor in the area. In the case where a member is moving back into a previously visited area, rekeying process is not necessary.

The trusted domain key manager securely delegates the key management security parameters and the authentication of moving members to the intermediate trusted area key managers during initial registration setup. This property massively reduces performance hurdles such as authentication delays, and *1-affects-n* phenomenon in the entire network when the group members dynamically change point of attachment to the network areas. Furthermore, the *DKM* is less susceptible to face with a single point of failure in comparing to conventional schemes as the *AKM*s reduce signaling loads at the core network.

All area key managers are able to verify moving members without involving the domain key manager. In particular, the key manager of the visited area using a key derivation function independently derives the member key pertaining to each moving member. So, the key manager and the visiting member share a common key with the minimum communication overhead.

It provides backward secrecy not only in the area where a new member joins the group, but in the visited area where a mobile member moves into. It also preserves forward secrecy in the area where a leaving member departs the session and the areas which a leaving member carries their valid area key.

The simulation analysis has been conducted on HIMOB to assess impact of group size variation as well as member mobility variation on *1-affects-n* phenomenon and the average rekeying messages overhead in dynamic environments. HIMOB in overall has lower rekeying messages overhead and the minimum average of affected members in comparison with the FEDRP, GKMW, and LKH++. When the size of a group varies between small and large by changing the inter arrival rate and membership duration, the evaluation result depicted that HIMOB can better scale to large size as the average of rekeying messages and the average of affected members are smaller than the other solutions. In dynamic environments where the mobility rate increases, HIMOB performs more inexpensively compared to the other solutions.

6.2 The achievement of research objectives

The objectives of this study were defined in Chapter 1. Objective 1 has been achieved in Chapter 2 that covers the design approaches for group key management schemes, the constraints and challenges for the extension of a group key management to wireless mobile environments, and scrutinizing the existing schemes taking address the host mobility issue in secure group communication. A group key management scheme designed for wireless mobile environments so that takes address the host mobility issue has been proposed in Chapter 4, which is the accomplishment of second objective. Third objective has been achieved in Chapter 5 which covers the performance analysis of HIMOB in comparison with a number of existing schemes with regard to the rekeying

message overhead and *1-affects-n* phenomenon. The all objectives stated have been successfully achieved throughout this research work.

6.3 Research limitations

Each individual research project typically encounters with some limitations, for example, lacks the time and resources. Several limitations faced during the development of this research and some avenues that may have been of interest are explored in this section to provide future researchers some lessons and suggestions which enabled them to better manage their work.

Any simulator implementation has both advantages and disadvantages. Several simulation applications were explored during the development of the HIMOB. While existing simulators provide a ready to use environment for implementation of the proposed solution, dealing with unnecessary underlying protocols which required to be configured for the deployment of the HIMOB requested extra time and attempts. The SGCSim as the simulator used in this research was programmed from scratch. Although all modules, classes, and protocols have been under control during the development, implementing the proposed protocols in code was roughly challenging.

C# used as the main programming language for developing the SGCSim. Although C# has its own advantages such as simplicity in coding as it is a high-level programing language as well as and its cross-platform compatibility is a benefit for disseminating and repeating this work, it is fairly resource intensive, for example, high memory usage, and not optimized for large test run which may result in the expected result not being obtained. For future research, C# should be examined with a greater evaluation and analysis if it is used as the main programming language in any re-implementation of the SGCSim.

While an acceptable implementation of the HIMOB has been conducted in simulation environments, further deployment of the HIMOB in a real environment would provide the ultimate test of the proposal's viability. Overall scalability and efficiency of HIMOB implementation in the simulator may be pretty different than the real environments. This discrepancy can be explained by the different underlying protocols and constraints corresponding to the real environment that are required to be considered before deploying HIMOB. Lack of appropriate infrastructures, wireless devices, and financial constraints were the main obstacles, which prevented this work being implemented in real environments. Furthermore, implementation in real environments requires a greater time resource than was available in the study period covered by this thesis, but provides a natural next step.

6.4 Future Works

This work provides a useful contribution to the development of group key management in a wireless environment. There are a few challenges for further investigation of this subject. The future directions of this research can be as follows:

6.4.1 Different type of wireless mobile environments

This research includes a generic group key management scheme for establishing secure group communication in infrastructure-based wireless mobile environments. It could thus be used as a basis for developing other schemes for a wider variety of network topologies and scenarios in particular infrastructure-less environments. Particular design changes that would cause alternative schemes to extend the protocols to operate on MANETs (Mobile Ad-hoc Networks), which would lead to giving greater versatility for this protocol and allowing mobile UAVs, vehicles, and personnel to utilize this particular group key management protocol.

6.4.2 Computation cost analysis

The performance of the proposed was analyzed with evaluating *1-affects-n* phenomenon and the communication cost in terms of the number of rekeying messages. These analyses could be extended by measuring the consumption of network bandwidth used by the rekeying messages sent in different events. More analyses can be conducted on different protocols of HIMOB to evaluate the computation overhead when any changes occur in a group membership.

Several parameters other than inter arrival time, membership duration, and area dwell time can also be examined to provide further investigation of their impacts on *1-affects-n* phenomenon and rekeying messages overhead. For example, changing the member velocity value would vary the mobility rate, which could provide a greater understanding of communication costs as well as the average of affected members in the session. Additionally, varying the number of areas in a domain would provide better understanding of scalability of HIMOB for the given scenarios.

6.4.3 Key manager mobility

The effects of member mobility can be considerably mitigated by keeping track of moving members at the area manager level. It is assumed that the key managers have fixed network infrastructure and they are constantly available during the session to all group members. The design of a group key management scheme can be more challenging if not only the group member can move within the network, but also the group managers are considered as mobile entities. The mobility of managers requires a mechanism that can transfer the security services from the old area manager to the new area manager as well as select a new manager for the area from which the area manager has moved out.

6.4.4 Optimization of performance in terms of communication overhead

To be efficient, a scheme should optimize group performance in terms of communication and computation costs, particularly when a moving member visits consecutive areas and suddenly leaves the group. Updating the keying materials in all visited areas at the same time incurs significant overhead as the manager must use secure unicast messages to deliver the new keys.

6.4.5 Internet of Things as an emerging global internet-based information architecture

The Internet of Things is expected to offer a dynamic global network infrastructure by embedding intelligence into the environment and connecting the everyday objects (Gubbi, Buyya, Marusic, & Palaniswami, 2013; Sundmaeker, Guillemin, Friess, & Woelfflé, 2010). Future research on the group key management scheme is engineered to support the heterogeneity of wireless networks in these kind of infrastructures are required.

REFERENCES

Abd-Alhameed, R., Mapoka, T., & Shepherd, S. (2014). A new multiple service key management scheme for secure wireless mobile multicast. *Mobile Computing, IEEE Transactions on*, *PP*(99), 1-14. doi:10.1109/TMC.2014.2362760

Al-Sultan, S., Al-Doori, M. M., Al-Bayatti, A. H., & Zedan, H. (2014). A comprehensive survey on vehicular Ad Hoc network. *Journal of Network and Computer Applications, 37*(0), 380-392. doi:http://dx.doi.org/10.1016/j.jnca.2013.02.036

Almeroth, K. C., & Ammar, M. H. (1996). *Collecting and modeling the join/leave behavior of multicast group members in the mbone.* Paper presented at the High Performance Distributed Computing, 1996., Proceedings of 5th IEEE International Symposium on.

Almeroth, K. C., & Ammar, M. H. (1997). Multicast group behavior in the Internet's multicast backbone (MBone). *Communications Magazine, IEEE, 35*(6), 124-129.

Amir, Y., Nita-Rotaru, C., Stanton, S., & Tsudik, G. (2005). Secure spread: An integrated architecture for secure group communication. *Dependable and Secure Computing, IEEE Transactions on, 2*(3), 248-261.

Angamuthu, M., & Ramalingam, A. (2012). Balanced key tree management for multi-privileged groups using (N, T) policy. *Security and Communication Networks, 5*(5), 545-555.

Aschenbruck, N., Gerhards-Padilla, E., & Martini, P. (2008). A survey on mobility models for performance analysis in tactical mobile networks. *Journal of Telecommunications and Information Technology, 2*, 54-61.

Bajaj, S., Breslau, L., Estrin, D., Fall, K., Floyd, S., Haldar, P., . . . Huang, P. (1999). *Improving simulation for network research* (99-702). Retrieved from http://www.cs.usc.edu/assets/003/83131.pdf

Banerjee, S., & Bhattacharjee, B. (2002). A comparative study of application layer multicast protocols. *Network, 4*, 3.

Baugher, M., Canetti, R., Dondeti, L., & Lindholm, F. (2005). *Multicast Security (MSEC) Group Key Management Architecture. RFC 4046*. Internet Engineering Task Force.

Benyamina, D., Hafid, A., & Gendreau, M. (2012). Wireless Mesh Networks Design A Survey. *Communications Surveys & Tutorials, IEEE, 14*(2), 299-310. doi:10.1109/SURV.2011.042711.00007

Bouassida, M. S., Chrisment, I., & Festor, O. (2008). Group key management in MANETs. *International Journal of Network Security, 6*(1), 67-79.

Bruschi, D., & Rosti, E. (2002). Secure multicast in wireless networks of mobile hosts: protocols and issues. *Mob. Netw. Appl., 7*(6), 503-511. doi:10.1023/a:1020781305639

Buchmann, J. (2013). *Introduction to cryptography*: Springer Science & Business Media.

Burmester, M., & Desmedt, Y. (2005). A secure and scalable Group Key Exchange system. *Information Processing Letters, 94*(3), 137-143. doi:http://dx.doi.org/10.1016/j.ipl.2005.01.003

Cao, J., Liao, L., & Wang, G. (2006). Scalable key management for secure multicast communication in the mobile environment. *Pervasive and Mobile Computing, 2*(2), 187-203.

Cavalcanti, D., Agrawal, D., Cordeiro, C., Bin, X., & Kumar, A. (2005). Issues in integrating cellular networks WLANs, AND MANETs: a futuristic heterogeneous wireless network. *Wireless Communications, IEEE, 12*(3), 30-41. doi:10.1109/MWC.2005.1452852

Challal, Y., Bettahar, H., & Bouabdallah, A. (2004). SAKM: a scalable and adaptive key management approach for multicast communications. *SIGCOMM Comput. Commun. Rev., 34*(2), 55-70. doi:10.1145/997150.997157

Challal, Y., Gharout, S., Bouabdallah, A., & Bettahar, H. (2008). Adaptive clustering for scalable key management in dynamic group communications. *International Journal of Security and Networks, 3*(2), 133-146.

Challal, Y., & Seba, H. (2005). Group key management protocols: A novel taxonomy. *International Journal of Information Technology, 2*(1), 105-118.

Chan, K.-C., & Chan, S.-H. G. (2002). Distributed servers approach for large-scale secure multicast. *IEEE J.Sel. A. Commun., 20*(8), 1500-1510. doi:10.1109/jsac.2002.803966

Chang, Y. F., Chen, C. S., & Zhou, H. (2009). Smart phone for mobile commerce. *Computer Standards & Interfaces, 31*(4), 740-747. doi:http://dx.doi.org/10.1016/j.csi.2008.09.016

Chen, X., Ma, B. N. W., & Yang, C. (2007). M-CLIQUES: Modified CLIQUES key agreement for secure multicast. *Computers & Security, 26*(3), 238-245. doi:http://dx.doi.org/10.1016/j.cose.2006.11.001

Chen, Z.-Z., Feng, Z., Li, M., & Yao, F. (2008). Optimizing deletion cost for secure multicast key management. *Theoretical Computer Science, 401*(1–3), 52-61. doi:http://dx.doi.org/10.1016/j.tcs.2008.03.016

Chlamtac, I., Conti, M., & Liu, J. J. N. (2003). Mobile ad hoc networking: imperatives and challenges. *Ad Hoc Networks, 1*(1), 13-64. doi:http://dx.doi.org/10.1016/S1570-8705(03)00013-1

Cho, J.-H., Chen, I.-R., & Wang, D.-C. (2008). Performance optimization of region-based group key management in mobile ad hoc networks. *Performance Evaluation, 65*(5), 319-344.

Chockler, G. V., Keidar, I., & Vitenberg, R. (2001). Group communication specifications: a comprehensive study. *ACM Comput. Surv., 33*(4), 427-469. doi:10.1145/503112.503113

Chung Kei, W., Gouda, M., & Lam, S. S. (2000). Secure group communications using key graphs. *Networking, IEEE/ACM Transactions on, 8*(1), 16-30. doi:10.1109/90.836475

Cisco Systems. (2001). Internet Protocol IP Multicast Technology *Cisco Internetworking Technology Handbook*: Cisco DocWiki.

Cisco Systems. (2012). Internet Protocol Multicast. Retrieved from http://docwiki.cisco.com/wiki/Internet_Protocol_Multicast

Cisco Visual Networking Index. (2016a). *Global mobile data traffic forecast update, 2015-2020*. Retrieved from http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

Cisco Visual Networking Index. (2016b). *The Zettabyte Era—Trends and Analysis*. Retrieved from http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html

Crow, B. P., Widjaja, I., Kim, J. G., & Sakai, P. T. (1997). IEEE 802.11 wireless local area networks. *Communications Magazine, IEEE, 35*(9), 116-126.

Daghighi, B., Mat Kiah, M. L., Shamshirband, S., & Rehman, M. H. U. (2015). Toward secure group communication in wireless mobile environments: Issues, solutions, and challenges. *Journal of Network and Computer Applications, 50*(0), 1-14. doi:http://dx.doi.org/10.1016/j.jnca.2014.11.001

DeCleene, B., Dondeti, L., Griffin, S., Hardjono, T., Kiwior, D., Kurose, J., . . . Zhang, C. (2001). *Secure group communications for wireless networks*. Paper presented at the Military Communications Conference, (MILCOM 2001). .

Desmond Ng, W. H., Cruickshank, H., & Sun, Z. (2006). Scalable balanced batch rekeying for secure group communication. *Computers & Security, 25*(4), 265-273. doi:http://dx.doi.org/10.1016/j.cose.2006.02.006

Dhurandher, S. K., & Singh, G. V. (2005, 23-25 Jan. 2005). *Weight based adaptive clustering in wireless ad hoc networks.* Paper presented at the Personal Wireless Communications, 2005. ICPWC 2005. 2005 IEEE International Conference on.

Di Pietro, R., Mancini, L. V., Yee Wei, L., Etalle, S., & Havinga, P. (2003, 6-9 Oct. 2003). *LKHW: a directed diffusion-based secure multicast scheme for wireless sensor networks.* Paper presented at the Parallel Processing Workshops, 2003. Proceedings. 2003 International Conference on.

Dierks, T., & Rescorla, E. (2008). *The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246*. Internet Engineering Task Force.

El-Sayed, A., Roca, V., & Mathy, L. (2003). A survey of proposals for an alternative group communication service. *Network, IEEE, 17*(1), 46-51. doi:10.1109/MNET.2003.1174177

Fisher, R. (2007). *60 GHz WPAN standardization within IEEE 802.15. 3c.* Paper presented at the 2007 International Symposium on Signals, Systems and Electronics.

Floyd, S., Jacobson, V., Liu, C.-G., McCanne, S., & Zhang, L. (1997). A reliable multicast framework for light-weight sessions and application level framing. *IEEE/ACM Trans. Netw., 5*(6), 784-803. doi:10.1109/90.650139

Frankel, S., & Kelly, S. G. (2007). Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec: Internet Engineering Task Force.

Freier, A., Karlton, P., & Kocher, P. C. (1996). *The SSL Protocol: Version 3*. Internet Engineering Task Force.

Friedhelm, H. (2002). *GSM and UMTS: The Creation of Global Mobile Communication*: John Wiley & Sons, Inc.

Gartner. (2015). Gartner Says Worldwide Sales of Cellular-Embedded Mobile PCs, Tablets and Mobile Hot Spot Devices Will Exceed 112 Million in 2015 [Press release]. Retrieved from http://www.gartner.com/newsroom/id/3064718

Gharout, S., Bouabdallah, A., Challal, Y., & Achemlal, M. (2012). Adaptive Group Key Management Protocol for Wireless Communications. *Journal of Universal Computer Science, 18*(6), 874-898.

Gharout, S., Challal, Y., & Bouabdallah, A. (2008). Scalable delay-constrained multicast group key management. *International Journal of Network Security, 7*(2), 142-156.

Goldsmith, A. J., & Wicker, S. B. (2002). Design challenges for energy-constrained ad hoc wireless networks. *Wireless Communications, IEEE, 9*(4), 8-27. doi:10.1109/MWC.2002.1028874

Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). Internet of Things (IoT): A vision, architectural elements, and future directions. *Future Generation Computer Systems, 29*(7), 1645-1660. doi:http://dx.doi.org/10.1016/j.future.2013.01.010

Guo, P. (2013). Optimal video length for student engagement. *edX blog*.

Guo, P. J., Kim, J., & Rubin, R. (2014). *How video production affects student engagement: An empirical study of mooc videos.* Paper presented at the Proceedings of the first ACM conference on Learning@ scale conference.

Guo, X., & Zhang, J. (2010). Secure group key agreement protocol based on chaotic Hash. *Information Sciences, 180*(20), 4069-4074. doi:http://dx.doi.org/10.1016/j.ins.2010.06.013

Gupta, A. K. (2008). *Challenges in Mobile Computing.* Paper presented at the Proceedings of 2nd National Conference on Challenges and Opportunities in Information Technology (COIT-2008). Mandi Gobindgarh, India: RIMT-IET.

Gupta, S. K. S., & Cherukuri, S. (2003, 16-20 March 2003). *An adaptive protocol for efficient and secure multicasting in IEEE 802.11 based wireless LANs.* Paper presented at the Wireless Communications and Networking, 2003. WCNC 2003. 2003 IEEE.

Hardjono, T., & Cain, B. (1998, 14-16 Dec 1998). *Secure and scalable inter-domain group key management for N-to-N multicast.* Paper presented at the Parallel and Distributed Systems, 1998. Proceedings. 1998 International Conference on.

Hardjono, T., Cain, B., & Monga, I. (2000). Intra-Domain Group Key Management Protocol. Retrieved from http://tools.ietf.org/html/draft-irtf-smug-intragkm-00

Harney, H., & Muckenhirn, C. (1997). *Group Key Management Protocol (GKMP) Architecture. RFC 2094.* Internet Engineering Task Force.

Harney, H., & Muckenhirn, C. (1997). Group Key Management Protocol (GKMP) Specification. RFC 2093.: Internet Engineering Task Force.

Heba K, A. (2004). A scalable and distributed multicast security protocol using a subgroup-key hierarchy. *Computers & Security, 23*(4), 320-329.

Hernandez Serrano, J., Pegueroles, J., & Soriano, M. (2005). *GKM over large MANET.* Paper presented at the Sixth International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing and First ACIS International Workshop on Self-Assembling Wireless Network.

Heydari, M. H., Morales, L., & Sudborough, I. H. (2006). *Efficient Algorithms for Batch Re-Keying Operations in Secure Multicast.* Paper presented at the System Sciences, 2006. HICSS '06. Proceedings of the 39th Annual Hawaii International Conference on.

Hiertz, G. R., Denteneer, D., Max, S., Taori, R., Cardona, J., Berlemann, L., & Walke, B. (2010). IEEE 802.11s: The WLAN Mesh Standard. *Wireless Communications, IEEE, 17*(1), 104-111. doi:10.1109/MWC.2010.5416357

Holzer, A., & Ondrus, J. (2011). Mobile application market: A developer's perspective. *Telematics and Informatics, 28*(1), 22-31. doi:http://dx.doi.org/10.1016/j.tele.2010.05.006

Hosseini, M., Ahmed, D. T., Shirmohammadi, S., & Georganas, N. D. (2007). A Survey of Application-Layer Multicast Protocols. *Communications Surveys & Tutorials, IEEE, 9*(3), 58-74.

Hur, J., & Yoon, H. (2009). A Decentralized Multi-Group Key Management Scheme. *IEICE Transactions on Communications, 92*, 632-635.

Hyytiä, E., & Virtamo, J. (2007). Random waypoint mobility model in cellular networks. *Wireless Networks, 13*(2), 177-188. doi:10.1007/s11276-006-4600-3

Je, D.-H., Lee, J.-S., Park, Y., & Seo, S.-W. (2010). Computation-and-storage-efficient key tree management protocol for secure multicast communications. *Computer Communications, 33*(2), 136-148.

Jong-Hyuk, R., & Kyoon-Ha, L. (2006, 20-22 Feb. 2006). *Key management scheme for providing the confidentiality in mobile multicast.* Paper presented at the Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference.

Judge, P., & Ammar, M. (2003). Security issues and solutions in multicast content distribution: a survey. *Network, IEEE, 17*(1), 30-36. doi:10.1109/mnet.2003.1174175

Jun, Z., Yu, Z., Fanyuan, M., Dawu, G., & Yingcai, B. (2006). An extension of secure group communication using key graph. *Information Sciences, 176*(20), 3060-3078. doi:http://dx.doi.org/10.1016/j.ins.2005.12.008

Kamat, S., Parimi, S., & Agrawal, D. P. (2003). *Reduction in control overhead for a secure, scalable framework for mobile multicast*.

Kim, Y., Perrig, A., & Tsudik, G. (2004a). Group Key Agreement Efficient in Communication. *IEEE Trans. Comput., 53*(7), 905-921.

Kim, Y., Perrig, A., & Tsudik, G. (2004b). Tree-based group key agreement. *ACM Trans. Inf. Syst. Secur., 7*(1), 60-96.

Kobayashi, M., Nakayama, H., Ansari, N., & Kato, N. (2009). Reliable application layer multicast over combined wired and wireless networks. *Trans. Multi., 11*(8), 1466-1477. doi:10.1109/tmm.2009.2032692

Konstantinou, E. (2011). Efficient cluster-based group key agreement protocols for wireless ad hoc networks. *Journal of Network and Computer Applications, 34*(1), 384-393.

Koodli, R. (2009). RFC 5568 - Mobile IPv6 fast handovers. : Internet Engineering Task Force.

Lenstra, A. K., & Verheul, E. R. (2001). Selecting cryptographic key sizes. *Journal of Cryptology, 14*(4), 255-293.

Li, C., & Xu, C. (2013). Scalable group key exchange protocol with provable security. *COMPEL: Int J for Computation and Maths. in Electrical and Electronic Eng., 32*(2), 612-619.

Li, J. H., Bhattacharjee, B., Yu, M., & Levy, R. (2008). A scalable key management and clustering scheme for wireless ad hoc and sensor networks. *Future Generation Computer Systems, 24*(8), 860-869.

Li, L., Jun-Hong, C., Gerla, M., & Maggiorini, D. (2005, 13-17 March 2005). *A comparative study of multicast protocols: top, bottom, or in the middle?* Paper presented at the INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE.

Lin, J. C., Huang, K. H., Lai, F., & Lee, H. C. (2009). Secure and efficient group key management with shared key derivation. *Computer Standards & Interfaces, 31*(1), 192-208.

Lv, X., Li, H., & Wang, B. (2012). Group key agreement for secure group communication in dynamic peer systems. *Journal of Parallel and Distributed Computing, 72*(10), 1195-1200. doi:http://dx.doi.org/10.1016/j.jpdc.2012.06.004

Magliveras, S., Wandi, W., & Xukai, Z. (2008). *Notes on the CRTDH Group Key Agreement Protocol*. Paper presented at the Distributed Computing Systems Workshops, 2008. ICDCS '08. 28th International Conference on.

Mapoka, T. T. (2013). Group key management protocols for secure mobile multicast communication: a comprehensive survey. *International Journal of Computer Applications, 84*(12).

Martin, J., & Haberman, B. (2008). Internet Group Management Protocol Version 3 (IGMPv3)/Multicast Listener Discovery Version 2 (MLDv2) and Multicast Routing Protocol Interaction: Internet Engineering Task Force.

Mat Kiah, M. L., & Martin, K. M. (2007). *Host Mobility Protocol for Secure Group Communication in Wireless Mobile Environments.* Paper presented at the Proceedings of the Future Generation Communication and Networking

Mehdizadeh, A., Hashim, F., & Othman, M. (2014). Lightweight decentralized multicast–unicast key management method in wireless IPv6 networks. *Journal of Network and Computer Applications, 42*(0), 59-69. doi:http://dx.doi.org/10.1016/j.jnca.2014.03.013

Meyer, D. (1998). *Administratively scoped IP multicast*. (RFC 2365).

Min-Ho, P., Young-Hoon, P., & Seung-Woo, S. (2010). *A Cell-Based Decentralized Key Management Scheme for Secure Multicast in Mobile Cellular Networks.* Paper presented at the Vehicular Technology Conference (VTC 2010-Spring), 2010 IEEE 71st.

Mittra, S. (1997). Iolus: a framework for scalable secure multicasting. *SIGCOMM Comput. Commun. Rev., 27*(4), 277-288.

Mohapatra, P. (2005). *AD HOC NETWORKS: technologies and protocols*: Springer Science & Business Media.

Mortazavi, S. A., Pour, A. N., & Kato, T. (2011, 23-24 Feb. 2011). *An efficient distributed group key management using hierarchical approach with Diffie-Hellman and Symmetric Algorithm: DHSA.* Paper presented at the Computer Networks and Distributed Systems (CNDS), 2011 International Symposium on.

MSEC. (2011). Charter charter-ietf-msec-03. *Multicast Security (MSEC) Group of Internet EngineeringTask Force(IETF).* Retrieved from http://datatracker.ietf.org/doc/charter-ietf-msec/

Nemaney Pour, A., Kumekawa, K., Kato, T., & Itoh, S. (2007). A hierarchical group key management scheme for secure multicast increasing efficiency of key distribution in leave operation. *Computer Networks, 51*(17), 4727-4743.

Ng, W. H. D., Howarth, M., Sun, Z., & Cruickshank, H. (2007). Dynamic Balanced Key Tree Management for Secure Multicast Communications. *Computers, IEEE Transactions on, 56*(5), 590-605.

ON24 Inc. (2016). *ON24 Webinar Benchmarks Report 2016*. Retrieved from http://communications.on24.com/webinar-benchmarks-report-wp-2016.html

OPNET Technologies Inc. (2014). OPNET Modeler. Retrieved from http://www.riverbed.com/products/performance-management-control/network-performance-management/network-simulation.html

Pahlavan, K. (2011). *Principles of Wireless Networks: A Unified Approach*: John Wiley & Sons, Inc.

Park, Y., Je, D., Park, M., & Seo, S. (2014). Efficient Rekeying Framework for Secure Multicast with Diverse-Subscription-Period Mobile Users. *Mobile Computing, IEEE Transactions on, 13*(4), 783-796. doi:10.1109/TMC.2013.40

Paul, S. (2012). *Multicasting on the Internet and its Applications*: Springer Science & Business Media.

Perrig, A., Canetti, R., Song, D., & Tygar, J. (2001). *Efficient and secure source authentication for multicast.* Paper presented at the Network and Distributed System Security Symposium, NDSS.

Piao, Y., Kim, J., Tariq, U., & Hong, M. (2013). Polynomial-based key management for secure intra-group and inter-group communication. *Computers & Mathematics with Applications, 65*(9), 1300-1309. doi:http://dx.doi.org/10.1016/j.camwa.2012.02.008

Pierre, S. (2001). Mobile computing and ubiquitous networking: concepts, technologies and challenges. *Telematics and Informatics, 18*(2–3), 109-131. doi:http://dx.doi.org/10.1016/S0736-5853(00)00024-1

Pietro, R. D., Mancini, L. V., & Jajodia, S. (2002). *Efficient and secure keys management for wireless mobile communications*. Paper presented at the Proceedings of the second ACM international workshop on Principles of mobile computing, Toulouse, France.

Quinn, B., & Almeroth, K. (2001). IP multicast applications: Challenges and solutions - RFC 3170. *Internet Engineering Task Force*.

Rafaeli, S., & Hutchison, D. (2003). A survey of key management for secure group communication. *ACM Comput. Surv., 35*(3), 309-329.

Rappaport, T. S. (2001). *Wireless communications: principles and practice* (Vol. 2): prentice hall PTR New Jersey.

Ratnasamy, S., Ermolinskiy, A., & Shenker, S. (2006). Revisiting IP multicast. *SIGCOMM Comput. Commun. Rev., 36*(4), 15-26. doi:10.1145/1151659.1159917

Romdhani, I., Kellil, M., Hong-Yon, L., Bouabdallah, A., & Bettahar, H. (2004). IP mobile multicast: Challenges and solutions. *Communications Surveys & Tutorials, IEEE, 6*(1), 18-41. doi:10.1109/comst.2004.5342232

Rossi, A., Pierre, S., & Krishnan, S. (2010). An Efficient and Secure Self-Healing Scheme for LKH. *Journal of Network and Systems Management, 18*(3), 327-347.

Sakarindr, P., & Ansari, N. (2007). Security services in group communications over wireless infrastructure, mobile ad hoc, and wireless sensor networks. *Wireless Communications, IEEE, 14*(5), 8-20. doi:10.1109/mwc.2007.4396938

Sakarindr, P., & Ansari, N. (2010). Survey of security services on group communications. *Information Security, IET, 4*(4), 258-272. doi:10.1049/iet-ifs.2009.0261

Sathiaseelan, A., & Crowcroft, J. (2012). Internet on the move: challenges and solutions. *SIGCOMM Comput. Commun. Rev., 43*(1), 51-55. doi:10.1145/2427036.2427046

Savola, P. (2008). Overview of the Internet Multicast Routing Architecture. *Internet Engineering Task Force*.

Schmidt, T., Waehlisch, M., & Fairhurst, G. (2010). Multicast mobility in mobile IP version 6 (MIPv6): problem statement and brief survey. RFC 5757: Internet Engineering Task Force.

Sherman, A. T., & McGrew, D. A. (2003). Key establishment in large dynamic groups using one-way function trees. *Software Engineering, IEEE Transactions on, 29*(5), 444-458.

Shin, Y., Choi, M., Koo, J., & Choi, S. (2013). Video multicast over WLANs: Power saving and reliability perspectives. *Network, IEEE, 27*(2), 40-46. doi:10.1109/MNET.2013.6485095

Shin, Y., & Hur, J. (2012). Scalable and efficient approach for secure group communication using proxy cryptography. *Wireless Networks, 18*(4), 413-425. doi:10.1007/s11276-011-0408-x

Srinivas, V., & Lu, R. (2009, 15-19 June 2009). *An efficient reliable multicast protocol for 802.11-based wireless LANs.* Paper presented at the World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a.

Steiner, M., Tsudik, G., & Waidner, M. (1996). *Diffie-Hellman key distribution extended to group communication.* Paper presented at the Proceedings of the 3rd ACM conference on Computer and communications security, New Delhi, India.

Steiner, M., Waidner, M., & Tsudik, G. (1998, 26-29 May). *CLIQUES: A New Approach to Group Key Agreement.* Paper presented at the Proceedings of the The 18th International Conference on Distributed Computing Systems, Amsterdam, Netherlands.

Sun, Y., Trappe, W., & Liu, K. J. R. (2004). A scalable multicast key management scheme for heterogeneous wireless networks. *IEEE/ACM Trans. Netw., 12*(4), 653-666. doi:10.1109/tnet.2004.833129

Sundmaeker, H., Guillemin, P., Friess, P., & Woelfflé, S. (2010). Vision and challenges for realising the Internet of Things. *Cluster of European Research Projects on the Internet of Things, European Commision*.

Taferner, M., & Bonek, E. (2013). *Wireless internet access over GSM and UMTS*: Springer Science & Business Media.

Varga, A., & Hornig, R. (2008). *An overview of the OMNeT++ simulation environment*. Paper presented at the Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops, Marseille, France.

Waldvogel, M., Caronni, G., Dan, S., Weiler, N., & Plattner, B. (1999). The VersaKey framework: versatile group key management. *Selected Areas in Communications, IEEE Journal on, 17*(9), 1614-1631.

Wallner, D., Harder, E., & Agee, R. (1999). Key Management for Multicast: Issues and Architectures. RFC 2627.: Internet Engineering Task Force.

Wang, C., Li, Y., Han, L., & Ma, J. (2009, 18-20 Oct. 2009). *A new reliable multicast scheme for multimedia applications in wireless environment.* Paper presented at the Broadband Network & Multimedia Technology, 2009. IC-BNMT '09. 2nd IEEE International Conference on.

Wang, N. C., & Fang, S. Z. (2007). A hierarchical key management scheme for secure group communications in mobile ad hoc networks. *Journal of Systems and Software, 80*(10), 1667-1677.

Weingartner, E., vom Lehn, H., & Wehrle, K. (2009, 14-18 June 2009). *A Performance Comparison of Recent Network Simulators.* Paper presented at the Communications, 2009. ICC '09. IEEE International Conference on.

Wu, Q., Mu, Y., Susilo, W., Qin, B., & Domingo-Ferrer, J. (2009). Asymmetric Group Key Agreement. In A. Joux (Ed.), *Advances in Cryptology - EUROCRYPT 2009* (Vol. 5479, pp. 153-170): Springer Berlin Heidelberg.

Yan, S., & Liu, K. J. R. (2007). Hierarchical Group Access Control for Secure Multicast Communications. *Networking, IEEE/ACM Transactions on, 15*(6), 1514-1526.

Yeo, C. K., Lee, B. S., & Er, M. H. (2004). A survey of application level multicast techniques. *Computer Communications, 27*(15), 1547-1568. doi:http://dx.doi.org/10.1016/j.comcom.2004.04.003

Yi-Ruei, C., Tygar, J. D., & Wen-Guey, T. (2011, 10-15 April 2011). *Secure group key management using uni-directional proxy re-encryption schemes.* Paper presented at the INFOCOM, 2011 Proceedings IEEE.

Yick, J., Mukherjee, B., & Ghosal, D. (2008). Wireless sensor network survey. *Computer Networks*, *52*(12), 2292-2330. doi:http://dx.doi.org/10.1016/j.comnet.2008.04.002

Young, E., & Hudson, T. (2015). OpenSSL cryptography and SSL/TLS toolkit.

Zhang, C., DeCleene, B., Kurose, J., & Towsley, D. (2002). Comparison of inter-area rekeying algorithms for secure wireless group communications. *Performance Evaluation*, *49*(1-4), 1-20.

Zheng, S., Manz, D., & Alves-Foss, J. (2007). A communication-computation efficient group key algorithm for large and dynamic groups. *Computer Networks*, *51*(1), 69-93.

LIST OF PUBLICATIONS AND PAPERS PRESENTED

Journal publications:

Daghighi, B., Kiah, M. L. M., Shamshirband, S., Iqbal, S., & Asghari, P. (2015). Key management paradigm for mobile secure group communications: Issues, solutions, and challenges. *Computer Communications, 72*, 1-16. [ISI-Q1]

Daghighi, B., Mat Kiah, M. L., Shamshirband, S., & Rehman, M. H. U. (2015). Toward secure group communication in wireless mobile environments: Issues, solutions, and challenges. *Journal of Network and Computer Applications, 50*(0), 1-14. [ISI-Q1]

Daghighi, B., & Mat Kiah, M. L. (2014). A lightweight key management for secure group communication in wireless mobile environment. *ScienceAsia, (Accepted)* [ISI-Q3].

Daghighi, B., Mat Kiah, M. L., Keith, M., & Iqbal, S. (2016). Host mobility key management for dynamic secure group communication. *Journal of Network and Computer Applications.* (Submitted) [ISI-Q1]

Conferences:

Daghighi, B., Mat Kiah, M. L., & Iqbal, S. (2015). *Host mobility key management for dynamic secure group communications*. Paper presented at the 10th International Conference on Broadband and Wireless Computing, Communication and Applications, Krakov, Poland.

Daghighi, B., & Mat Kiah, M. L. (2013). *A Group Key Management Scheme for Wireless Mobile Environments*. Paper presented at the International Conference on Computer Engineering & Mathematical Sciences (ICCEMS 2013), Kuala Lumpur, Malaysia.