

Host mobility key management in dynamic secure group communication

Babak Daghighi^{1*}, Miss Laiha Mat Kiah¹, Salman Iqbal¹, Muhammad Habib Ur Rehman², Keith Martin³

¹Faculty of Computer Science & Information Technology, University Malaya, Kuala Lumpur, Malaysia

²COMSATS Institute of Information Technology, Wah Cantt, Pakistan

³Information Security Group, Department of Mathematics, Royal Holloway, University of London, United Kingdom

*Corresponding Author: babak@um.edu.my

Abstract— The key management has a fundamental role in securing group communications taking place over vast and unprotected networks. It is concerned with the distribution and update of the keying materials whenever any changes occur in the group membership. Wireless mobile environments enable members to move freely within the networks, which causes more difficulty to design efficient and scalable key management protocols. This is partly because both member location dynamic and group membership dynamic must be managed concurrently, which may lead to significant rekeying overhead. This paper presents a hierarchical group key management (GKM) scheme taking the mobility of members into consideration intended for wireless mobile environments. The proposed scheme supports the mobility of members across wireless mobile environments while remaining in the group session with minimum rekeying transmission overhead. Furthermore, the proposed scheme alleviates *1-affect-n* phenomenon, single point of failure, and signaling load caused by moving members at the core network. Simulation results shows that the scheme surpasses other existing efforts in terms of communication overhead and affected members. The security requirements studies also show the backward and forward secrecy is preserved in the proposed scheme even though the members move between areas.

Keywords—Secure group communication, group key management, group communication, host mobility

1 INTRODUCTION

The advance in Internet technology during the last few years and the increase of bandwidth in today networks [1,2] give rise to the demand 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 [3-5]. Group based applications provide efficient communication by delivering a single copy of data to the networks elements such as routers and switches making copy as necessary for the receivers, which results in better utilization of network resources such as bandwidth and buffer space [6,7]. Unfortunately, such applications suffer from lack of security [8,9] since members can openly and anonymously join the group[10].

Group based applications deployed either by with IP multicast [11] or overlay multicast [12] or other means suffer from lack of security [8,9]. Due to the open and anonymous membership and distributed nature of group communication [10], the group communication services become vulnerable to various security attack such as eavesdropping, denial of service (DoS), masquerading attacks and others [8,13,14]. To deliver securely the content of group communication only to the eligible recipients, the basic security services such as secrecy, data integrity and entity authentication must be in place depending on the application need. The group communication secrecy requires only legitimate members can read the data regardless of it is disseminated into the entire network [15]. The straightforward way for provision of secrecy in a group communication is the use of a symmetric called *Traffic Encryption Key (TEK)*, which shared between authorized members. Only legitimate group members holding the similar *TEK* can decrypt the group-intended data encrypted with the sender. However, the *TEK* should be updated through rekeying process due to the group membership dynamics caused with member joins or leaves. When an authorized member joins (or leaves) the group, the new *TEK* should be delivered to all group members to deny the access of new member (or leaving member) to previous (or future) group content to satisfy backward secrecy (or forward secrecy) [16,17].

Key management scheme is a fundamental building block for preserving secrecy in a group communication. Its main role is to generate, update, and distribute the *TEKs* to all group members. The challenge of an efficient group key management scheme is to reduce the rekeying communication overhead and the number of affected member each time there is a membership change. The impact of this rekeying process can be measured by number of messages distributed to replace the old *TEK* as well as the number of members affected with this process (referred to as *1-affects-n* phenomenon). Several attempts have been carried out to address secrecy in group communications [18-21]. However, these solutions remain some challenges in terms of efficiency, scalability, and performance. The number of messages can become critical when the rekeying process is triggered after each membership changes, which result in utilizing more network bandwidth

and buffer space. Logical key hierarchy (LKH) scheme efficiently reduce the rekeying cost [22]. To limit the impact of rekeying process on the group members, some solutions organized the group members into several subgroups, which cause the rekeying process restricted only to a subgroup in which an event occurs [23-26].

Wireless networks are generally implemented using radio communication that omits the needs of wire for connection, which enables wireless devices to have movement between different areas of a network. Host mobility [27-29] as a unique property of wireless networks poses a new challenge for securing group communication; how to deliver the keys to the members moving from one area to another while still remaining in the session [30]. Host mobility may result in extra generation of keying materials in secure group communication since the member is not recognized by the new area key manager. Therefore, the group key management scheme must deal not only with the dynamic group membership (join or leave) but also with the dynamic member location (mobility). Therefore, an efficient key management is required for managing host mobility in order to prevent service latency while reducing rekeying latency.

Several existing attempts have been made to address the mobility issue in secure group communication discussed in [20]. KMGW [31] exhibits minimum rekeying cost on member mobility however, it suffers from backward secrecy violation when a moving member enters into a new area. The moving member may be able to have access to the security information which had been generated before it joined the group in the visited area. Meanwhile, GKMW [24] has to burden many signaling messages. It also suffers from breach of forward secrecy in the visited areas, as the keying materials from each area visited by a leaving member are not updated when the moving member leaves the group communication. As a result, there is a need for a group key management scheme so that protects 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.

This paper proposes a hierarchical key management scheme with a novel rekeying strategy for members' mobility in secure group communications (called herein HISCOM) dedicated to infrastructure-based wireless mobile environments. This scheme moves the authentication of individual moving recipients from the domain key manager (*DKM*) to the area key managers (*AKM*), which alleviates single point of failure, unnecessary delays and possible bottlenecks at the *DKM*, and eventually signaling load at the core network. All the *AKMs* are able to generate independently the individual keys of each moving member without involving the trusted *DKM* and the origin *AKM*. In fact, host mobility rekeying is handled locally with the minimum communication overhead to reduce *1-affects-n* phenomenon. A new area encryption key mobile owner list (*AMOL*) is introduced for securely tracking host mobility, and minimizing rekeying transmission overheads in move events. Finally, in terms of security, HISCOM maintains backward and forward secrecy not only when a change occurs in group membership but also when group members change their locations.

The rest of this paper is organized as follows: the existing group key management approaches are presented in Section 2. The proposed scheme including host mobility key management is presented in Section 3. In Section 4, the required protocols for managing different events in terms of join, move, and leave are explained in details. Section 5 discusses the obtained results of the simulation experiments. Finally, the paper is concluded in section 6.

2 RELATED WORK

Traditional group key management schemes addressing rekeying over wired networks have been extensively studied in literature and classified into three design approaches namely, centralized, decentralized and distributed [18,21]. This classification is derived based on the main entities that are responsible for initiating keying materials.

Centralized group key management involves a single entity (i.e. a group controller GC) which is responsible for generating, distributing and updating the *TEK* and auxiliary keys whenever required. 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 [22,32]. The tree of keys in LKH reduces the required number of messages for updating the *TEK* induced by rekeying after membership changes to the logarithm of group members ($\log(n)$). Other existing approaches can be found in, [15,33-36].

In distributed approach (also known as contributory approach), group key management has no explicit key distribution centre (KDC) and all members contribute to manage the traffic key (*TEK*). This scheme helps to uniformly distribute the work load for key management and eliminates the need for central entity. Some distributed group key management schemes have been presented in [37,17,16,38-42].

In decentralized approach, a large group is split into some small subgroups so that they make some hierarchical levels. The responsibility of key management tasks is equally distributed between subgroup managers in each level to achieve scalability. This category can be classified further depending on how the *TEK* is distributed in the scheme into decentralized approach with a common *TEK* used for the whole group such as [43] or decentralized approach with independent *TEK* per subgroup as proposed in Iolus [44]. Although the decentralized scheme with independent *TEK* per subgroup alleviates the *1-affects-n* phenomenon as any changes in subgroup membership affect only the members residing at that specific subgroup, it suffers from computation overhead at the edge of each subgroup since data must be translated when it passes from one subgroup to another. Some improved schemes followed Iolus are such as [43,25,24,45-50].

Even though the design of efficient and scalable group key management scheme is difficult, the problem becomes complicated when host mobility is considered. Wireless mobile devices ranging from mobile hand held devices, notebooks,

and PDA with wireless connectivity can be exploited for opportunistic data transfer without using any fixed network infrastructure. These devices are able to freely move between different sections of a network while following different mobility patterns [51-53]. In a naïve scenario, the move event can induce rekeying process twice since it can be treated as a leave in the old area and a join in the new area. Performing frequent rekeying processes will drain the resources of wireless devices since such devices suffer from resources scarcity [54]. Previous studies such as those discussed in literature are mostly designed for wired environments. Few efforts have been carried out to extend the group key management protocols and address host mobility issues[55,20].

Di Pietro et al. presented LKH++ as an improved version of LKH for wireless mobile environments but it does not treat the mobility issue with an explicit protocol [56]. Two schemes (KTMM and WSMM) were proposed in [57] in order to extend the group key management to the mobile IP environment. The KTMM matches the key management tree to the mobile IP network topology, whereas the WSMM manages the wired and wireless areas separately using different keys in each area for data transmission. The group manager remains as a single point of failure in these solutions. Moreover, WSMM protocol requires to repeatedly update the keying materials if a member rapidly visits different areas which causes a cost of communication overhead in the both old area and new area. M-Iolus [58] is an enhanced version of Iolus, which supports mobility of members in wireless environment. However M-Iolus presents a null rekeying cost on member mobility, it suffers from backward secrecy violation in the visited subgroups.

FEDRP [59] and GKMW [24] maintains a list to keep track of mobile user as well as avoiding frequent rekeying in visited area. FEDRP shows an expense in communication cost when the mobility rate increases. GKMW also has to burden many signaling messages for managing move events. It also did not explain how to manage the leave rekeying since a leaving member might carry some valid area keys associated with the visited area, which cause lack of forward secrecy. Gharout et al. [60] proposed an adaptive key management which supports the mobility of members with a null rekeying cost. To handle the mobility of members, two lists are used in each area: 1) list of current members residing in the area (ListM), and 2) list of old members who already moved to the other areas (ListO). The drawback of this protocol is violation of backward secrecy since the mobile member may have access the security services information in visited area which is valid before the time that the member joined the group.

3 A GROUP KEY MANAGEMENT SCHEME SUPPORTING HOST MOBILITY

This scheme adopts a two tier hierarchical approach with a common traffic key for the whole group communication similar to [43,61]. The first level is the domain level, which consists of the domain key manager (*DKM*) for initial authentication procedure and managing the traffic encryption key. The second is comprised of a number of manageable areas where each one is managed by an area key manager (*AKM*) independently. The areas are indeed made by dividing the domain into a number of administratively scoped regions, which can be defined logically or physically. Each area contains a set of members subscribed to diverse group communications. The members are allowed to freely move between distributed areas. In this architecture, a domain can be viewed as an autonomous system which consists of a group of subsystem, for instance a corporate network, a multicast domain, or a wireless area. Areas can be viewed as subsystems which operate under the governance of bigger systems and follow the goals and objective of the bigger. The aim of placing members in areas is to achieve flexible and efficient management, particularly when there are changes in the group membership due to join, move or leave event. Therefore, the rekeying process is localized within the area and consequently the *1-affects-n* phenomenon is alleviated. Figure 1 shows the main components involved in the scheme architecture.

The role of *DKM* is to ensure the management of the *TEK* triggered due to join or leave events within the domain. The *DKM* is responsible to manage the domain, and closely operates with *AKMs* in regards to key management. The *AKM* is responsible for key management within its area and operates under control of the *DKM*. When an *AKM* receives a message from the *DKM*, it plays a role of a proxy and sends the message to the members residing in the area under its control. Furthermore, the management of members' moves are delegated to *AKMs* to omit the burden of authentication phase at the *DKM*. The *AKMs* are allowed to verify moving members, update and deliver the keying materials. Each *AKM* maintains an area encryption key mobile owner list (*AMOL*) to keep track of moving members and reduce the need for rekeying when a moving member return back the area where it has previously been visited.

All areas in the same domain use a common *TEK* generated by the *DKM* for encrypting/decrypting data flow. Therefore, when messages pass from one area inside the domain to another, the messages are not required to be translated. The *DKM* and the *AKMs* use a number of auxiliary keys in order to securely deliver the *TEK* and the controlling messages to the group members.

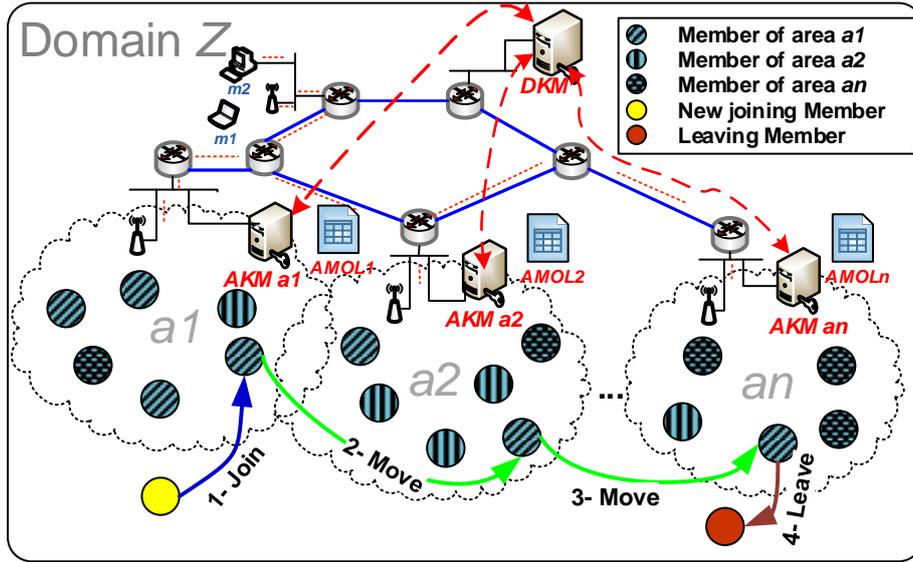


Figure 1. Placement of main components as well as join, move and leave event.

Upon any changes in group membership, the *TEK* and the auxiliary key of the affected areas must be updated for the domain members. The new *TEK* is generated by the *DKM* and delivered to the members of the group through the *AKM* of each area. Meanwhile, the *AKM* of affected area updates the area keying materials and send them to the existing members inside the area.

3.1 Definition and assumption

For more simplicity, an area and a member with identity i is respectively denoted by a_i and M_i . Each member has an individual secret key MEK_i which is shared between the member and its area key manager AKM_i . This key is generated by AKM_i on receiving the join request from the member. Each member M_i also has a certified pair of public and private keys. Moreover, each *AKM* generates an area encryption key AEK_i and shares it with all members residing in its jurisdiction, which is used to distribute the controlling messages to all group members inside area i through a secure multicast communication.

The *DKM* shares a symmetric key referred to as *Domain Encryption Key (DEK)* with all the *AKMs* used to disseminate other keys as well as controlling messages via a secure multicast communication. In addition, *DEK* is utilized as one of the parameters for generating the member encryption key (MEK_i) by every *AKM*. To communicate securely through a unicast channel with a specific *AKM*, the *DKM* generates a unique symmetric key called *Domain Area Key (DAK_i)* and shares with the AKM_i .

The *TEK* is generated by the *DKM* and must be shared among all members throughout the domain with assistance of all *AKMs*. At the domain level, the *DKM* encrypts the new *TEK* with either DAK_i or *DEK* and delivers to the *AKMs*. At the area level, the new *TEK* is delivered to members residing in area a_i protected under either each member MEK_i or AEK_i . The *DEK* and AEK_i are introduced respectively at the domain and the area level to optimize the number of rekeying messages whenever the *TEK* requires to be renewed. In this case, the *DEK* and AEK_i are used to encrypt and deliver the new *TEK* through a single multicast message in turn to the all *AKMs* and members within the domain instead of distributing the new *TEK* via unicast message encrypted in turn under DAK_i and MEK_i .

3.2 Assumption and notation

The design of the HISCOM scheme is based on the following assumption specified in [24].

- The cryptography keys specified in Section 3.1 are already established at initial group setup.
- All key managers (i.e. *DKM* and *AKM*) are trustworthy and reliable and all members trust them.
- The *AKM* has capability of deriving MEK_i without involving the *DKM*.
- Reliable multicast such as [62], [63] are in place to provide a reliable key delivery mechanism.
- Availability of secure storage of cryptographic keys for all group communication entities.
- Availability of secure mechanism for managing *AMOL*.

For more simplicity, the notations used in this solution are described in Table 1.

Table 1. Notation used in group key management scheme

Symbol	Significance	Function
M_i	Member i	Existing moving member in area i .
a_i	Area i	A set of group members using the same AEK and under control of AKM_i .
DKM	Domain Key Manager	The main security point, governing $AKMs$, generating and updating TEK during rekeying process.
AKM_i	Area Key manager of area i	Granting access to M_i , MEK_i derivation, verifying the moving members, and maintenance of the $AMOL$ and $MemL$.
TEK	Traffic Encryption Key	Specific symmetric key for encrypting and decrypting group traffic. The new_TEK and old_TEK are respectively referred to as newly generated TEK , and the currently used TEK .
MEK	Member Key	A symmetric key is used to encrypt messages between AKM and member. Each member M_i has its own MEK shown by MEK_i .
AEK	Area Encryption Key	A symmetric key used for encrypting messages sent to all members residing in an area.
DEK	Domain Encryption Key	A symmetric key shared between DKM and all $AKMs$.
$AMOL$	List of departing member from area	This contains the list of moving members which previously left the area i and moved to other areas.
$MemL$	List of current member in area	This contains the list of current members residing in area a_i .
$ A $	Total number of areas	The number of areas into which the domain is divided.
n_i	Total number of members in area i	The total number of members that stay in area i .
n	Number of group members	The total number of group members have joined the session.
ID_{M_i}	Identity of Member i	It is used by AKM_i to generate MEK_i .
ID_G	Identity of the group communication G	It is used by AKM_i and DKM to identify particular services a member is subscribed to.
$\{m\}K$	Message m is encrypted with symmetric key K	Message (or data) m which is encrypted with the symmetric key K .
$Text$	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.
$A \Rightarrow x$	Multicast transmission	Disseminating a message from entity a to a group of members x using multicast communication.
\parallel	Concatenation	Concatenates different fields of a message.

3.3 Mobility key management

In this scheme, some system security parameters initially setup by the trusted DKM is securely delivered to the $AKMs$ for establishment of each group member MEK . A unique cryptography key DEK shared between the DKM and all $AKMs$ is one of the chosen security parameters that enable each AKM to derive individual key of each member without involving the DKM and other $AKMs$.

The AKM uses a key derivation function like PRF-HMAC-SHA-256 [64] to generate MEK_i of a new 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 the Formula 1 as follows:

$$MEK_i = PRF-HMAC-SHA-256(DEK \parallel ID_{M_i} \parallel ID_G \parallel text) \quad (1)$$

In Formula 1, $Text$ contains other security parameters corresponding to the member. All $AKMs$ require to use the same PRF-HMAC-SHA-256 in order to achieve a coordination throughout the domain for deriving the same MEK in all areas. Therefore, the AKM is able to proceed with the authentication of the visiting member and delivery of AEK without involving the DKM . This authentication mechanism enables all the $AKMs$ to verify the MEK presented by a moving member. For instance, 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 calculates 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 new area. The process of MEK_i^* derivation and comparison with MEK_i is depicted in Figure 2.

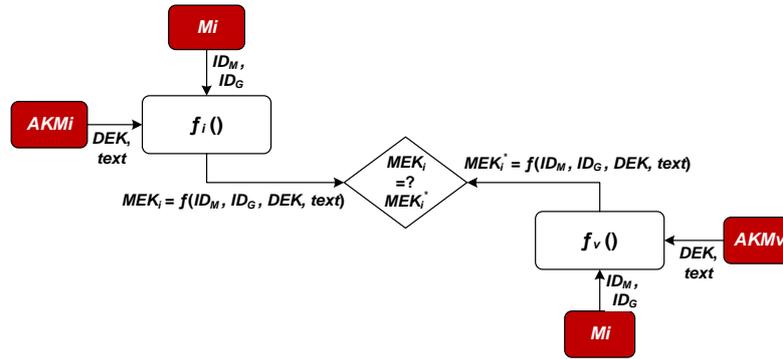


Figure 2. Mobility key management process.

The advantages of using this mechanism are as follows:

- (1) the bottleneck on the DKM is mitigated for managing mobility of dynamic members as the DKM is not swarmed with the multitude singling messages for authentication of moving members.
- (2) the resource constraint mobile devices do not undergo heavy computing process during authentication phase in the visited area, and
- (3) the management of moving members are distributed between all $AKMs$, which result in saving enormous bandwidth utilization during rekeying process.

3.4 List management

An important concept used as part of the mobility protocol design is a managing list(s) referred as Area encryption key Mobile Owner List (AMOL). This list enables each AKM to keep track of mobility of highly dynamic members who may accumulate the keying materials in the visited areas. The advantage of use of this list is to avoid frequent rekeying in visited areas that may cause disruption of group communication. Each area key managers in a domain securely maintains its own $AMOL$ and stores information of group members that move from its managing area to another. Each time a member transits to a new area, the information associated with the moving member such as identity of the moving member ID_{M_i} , identity of group communication ID_G joined by the member, identity of the area the that a member is moving to are logged in $AMOL$.

When a member enters an area, the AKM of the visited area can determine by looking up its $AMOL$ whether the member is a returning member who is just moving back to the area or is a new visiting member. In a case that the member is moving back into the area, the AKM skips to perform the rekeying process.

Another list called $MemL$ maintained with area key managers contains the information of current members residing in the area. The information in $MemL$ is used by key managers in a domain in order to locally handle the update of keying materials within the area upon any changes occur in the group membership.

4 GROUP KEY MANAGEMENT PROTOCOLS

In this scheme, a common TEK is used throughout the entire domain, which must be updated when there is a change in group membership due to join or leave in order to preserve backward and forward secrecy. The move event does not lead to performing the TEK rekeying process since the moving member is still valid in the session. Moreover, join or leave event causes to perform the AEK update in the area where it occurs for provision of backward and forward secrecy at the area level. The backward secrecy is necessary in mobility event to prevent moving member from having access the content before the time it joined the group within the visited area. The forward secrecy is unnecessary in the old area since members maintain session continuity while changing point of attachment to the network. Three cases of rekeying are distinguished as follows:

- Join rekeying: when a new member joins the group, a new AEK_i must be generated and sent to the new member and the other members residing in area i . Moreover, a new TEK must be generated and distributed to the group members and newly joining member. The scenario is shown in action (1) in Figure 1.
- Move rekeying: when a member changes its location from one area to another, the TEK is not changed but the AEK_j in the new area may be changed depend on the member join time and the last AEK_j update time. This scenario is depicted as action (2) and (3) in Figure 1.
- Leave rekeying: when a member leaves the group, the TEK is generated and delivered to the remaining group members. The AEK is also updated in area which their $AEKs$ are still valid and carried by the leaving member. This scenario is illustrated by action (4) in Figure 1.

4.1 Join protocol

In order to join the group session, a member M_i located in area a_i sends a join request message signed with its private key to AKM_i . On receipt of join request, AKM_i verifies the member's request. If the member is authorized to join the group session, AKM_i informs the DKM and concurrently generates MEK_i for the member M_i . The new MEK_i is sent to member protected under public key of member.

$$M_i \rightarrow AKM_i : \{ID_{M_i} \| ID_G \| text\} K_M$$

$$AKM_i \rightarrow DKM : \{ID_{A_i} \| ID_{M_i} \| ID_G \| text\} DAK_i$$

$$AKM_i \rightarrow M_i : \{ID_{A_i} \| ID_{M_i} \| ID_G \| MEK_i \| text\} PK_M$$

Upon receiving the message, the DKM generates a new TEK in order to guarantee backward secrecy throughout the domain. The DKM sends the new TEK to AKM_i manager of area i where the new member has joined the group session as well as other $AKMs$ in the domain. When AKM_i receives the new TEK , it generates a new AEK_i to achieve backward secrecy at area level. AKM_i sends a unicast message containing the new TEK and the new AEK_i encrypted with MEK_i to the new member and multicast this message to the other members residing in area i encrypted with the old AEK_i .

$$DKM \Rightarrow AKM_i : \{ID_{A_i} \| ID_{M_i} \| ID_G \| new_TEK \| text\} DEK$$

$$AKM_i \Rightarrow M_i : \{ID_{A_i} \| ID_{M_i} \| ID_G \| new_TEK \| new_AEK_i \| text\} old_AEK_i$$

The other AKM_s in the domain distribute the new TEK to the member residing in their area t by a secure multicast communication as well.

$$AKM_t \Rightarrow M_i : \{ID_{A_t} \| ID_G \| new_TEK \| text\} AEK_t$$

4.2 Mobility Protocol

This protocol describes the movement of a group member M_i from area i to area v with consideration to prevent access to previous keying materials (backward secrecy) in the visited area. Figure 3 outlines the flow of the mobility protocol in algorithmic form. The following operations are executed upon the movement of a member.

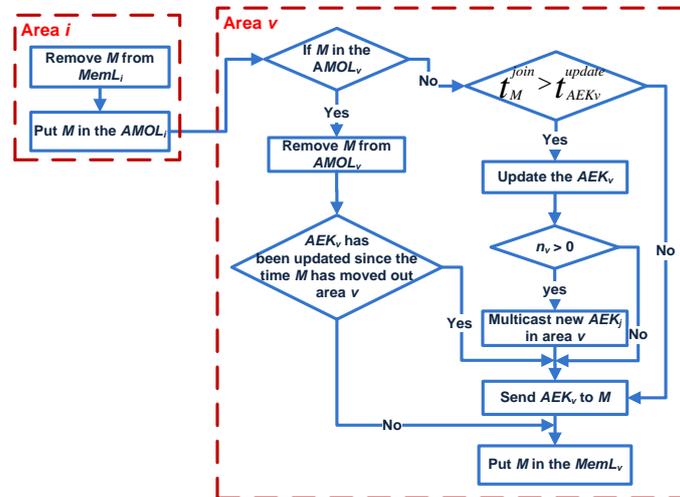


Figure 3. Mobility protocol when a member M moves from area i to area v .

- Member M_i informs simultaneously both its current AKM_i and the target AKM_v by sending a *Move Notify* message protected under M_i_Key . i.e.

$$M_i \rightarrow AKM_i : \{ID_{A_i} \parallel ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel text\}_{MEK_i}$$

$$M_i \rightarrow AKM_v : \{ID_{A_i} \parallel ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel text\}_{MEK_i}$$

- On receiving the move notify message, AKM_i verifies the message and informs the DKM about the member movement.

$$AKM_i \rightarrow DKM : \{ID_{A_i} \parallel ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel text\}_{DAK_i}$$

AKM_i does not require to carry out rekeying process for AEK_i in its area when the member M_i moves out since the moving member is still remaining in the session and therefore, the maintenance of forward secrecy is not necessary.

- As the M_i request reaches the target AKM_v , the member undergoes authentication process in order to be granted access. The AKM_v derives the MEK_i (as stated in Section 3.3), and after that verifies whether M_i is valid member or not. If the member verification is successful, AKM_v does the following:

- It looks for M_i 's identity in its $AMOL_v$. If M_i is not in the list that means this is the first time M_i visits area v and AKM_v must check the join time $t_{M_i}^{join}$ of M_i . If the join time $t_{M_i}^{join}$ is after the last update time ($t_{AEK_v}^{update}$) of AEK_v (i.e. $t_{M_i}^{join} > t_{AEK_v}^{update}$), AKM_v needs to perform key update process to refresh the AEK_v for achieving backward secrecy. It sends the new_AEK_v to M_i encrypted with MEK_i and distributes it between all residing members M_v in its area preferably by a multicast message. Meanwhile, information of the arrival is added to the $MemL_v$.

$$AKM_v \rightarrow M_i : \{ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel new_AEK_v \parallel text\}_{MEK_i}$$

$$AKM_v \Rightarrow M_v : \{ID_{A_v} \parallel ID_G \parallel new_AEK_v \parallel text\}_{old_AEK_v}$$

- In case that the M_i information has been already logged into the $AMOL$, there is no need to rekey the area encryption key AEK_i . AKM_i only requires to send the current area key AEK_v to M_i if it has been updated since the last visit paid by the moving member.

$$AKM_v \rightarrow M_i : \{ID_{A_v} \parallel ID_{M_i} \parallel ID_G \parallel AEK_v \parallel text\}_{MEK_i}$$

- AKM_v notifies the previous AKM_i and the DKM about moving M_i from area i to area v .

$$AKM_v \rightarrow DKM : \{ID_{A_v} \parallel ID_{A_i} \parallel ID_{M_i} \parallel ID_G \parallel text\}_{DEK}$$

$$AKM_v \rightarrow AKM_i : \{ID_{A_v} \parallel ID_{A_i} \parallel ID_{M_i} \parallel ID_G \parallel text\}_{DEK}$$

- AKM_i subsequently removes member information from $MemL_i$ and puts it into $AMOL_i$.

Figure 4 shows the sequence of messages during movement of a member from area i to area v .

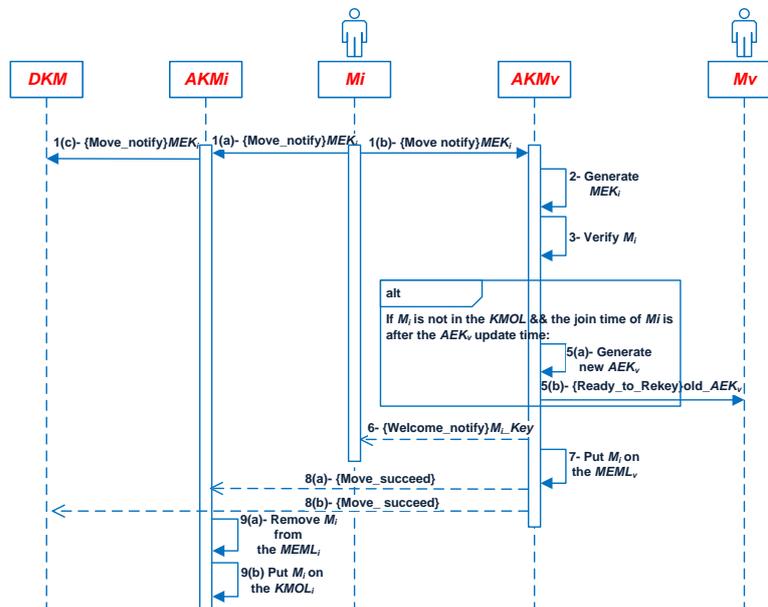


Figure 4. Mobility protocol messages flow.

4.3 Leave Protocol

When a member M_i located in a_i leaves the group session, it informs its area key manager AKM_i by sending a *Leave Notify* message encrypted with MEK_i . Upon receiving the message, AKM_i checks the message and subsequently encrypts and sends it to the DKM . In order to achieve forward secrecy at area level, AKM_i updates the AEK_i inside the area a_i .

The DKM removes the information of departure M_i from the group session by updating its $MemL$. The new TEK is generated and distributed throughout the domain in order to guarantee forward secrecy. The DKM sends the new TEK as well as information of departing member M_i to all the $AKMs$.

$$DKM \Rightarrow AKM : \{ID_{A_i} \parallel ID_{M_i} \parallel ID_G \parallel new_TEK \parallel text\} DEK$$

Upon receiving the new TEK , AKM_i sends the new TEK and new AEK to residue members M_i^* inside area i encrypted with each member MEK_i^* excluding the leaving member M_i . AKM_i removes M_i from $MemL_i$.

$$AKM_i \rightarrow M_i^* : \{ID_{A_i} \parallel ID_{M_i^*} \parallel ID_G \parallel new_AEK_i \parallel new_TEK \parallel text\} MEK_i^*$$

The leaving member might visit other areas inside the domain and accumulate information of each visited area. Therefore, the member M_i knows all the $AEKs$ used in previously visited areas. Thereby, all the $AEKs$ must be refreshed in each area where M_i has already visited. In other areas v ($v \neq i$) where the member M_i has previously visited them or is in the $AMOL_v$, AKM_v must update the AEK_v and send it along with the new TEK to its members in its area encrypted with the secret key MEK_v of each member M_v . Moreover, AKM_v removes information of the leaving member M_i from its $AMOL_v$.

$$AKM_v \rightarrow M_v : \{ID_{A_v} \parallel ID_{M_v} \parallel ID_G \parallel new_AEK_v \parallel new_TEK \parallel text\} MEK_v$$

To distribute the new TEK in the other areas p in the domain, AKM_p sends a multicast message containing the new TEK protected under AEK_p to all members residing in area p .

$$AKM_p \Rightarrow M_p : \{ID_{A_p} \parallel ID_G \parallel new_TEK \parallel text\} AEK_p$$

5 RESULT AND ANALYSIS

The proposed scheme is compared with several related schemes described in Section 2 namely, KMGM [31], GKMW [24], FEDRP [59], and LKH++ [56]. KMGM is a decentralized approach with independent TEK per area or subgroup, whereas GKMW, and FEDRP employ the decentralized approach with a common TEK for the whole group. LKH++ scheme was studied in the analysis as a representative of a centralized approach designed for wireless mobile environments.

5.1 General comparison

The generic comparison and number of rekeying signaling messages flow for each scheme are summarized respectively in Table 2 and Table 3. From Table 2, LKH++ and GKMW require to manage mobility events in synchronization with the DKM . Thus the key derivation of the TEK and the auxiliary keys require to involve the centralized DKM , which causes these schemes to become slower than HISCOM. This is due to the fact that the signaling messages require to traverse a long path to the DKM which could be located far from the $AKMs$. Moreover, the rekeying signaling load at the core network become considerable especially in dynamic mobile environments where the group members tend to change their location frequently, which leads to the lack of scalability.

With cryptographically separate keys at each area in KMGM, group communication requires to be decrypted and re-encrypted at the edge of each area hence increasing the computation overhead. Furthermore, KMGM requires to send a message with bigger size comprising of the auxiliary keys and the TEK in the visited area during move events, which result in more bandwidth consumption in comparison to HISCOM that sends only the AEK to the moving member in the visited area. The AKM_v can derive MEK_i of each moving member independent of the DKM and the origin AKM_i in HISCOM. Thus, host mobility is managed with minimal service disruption and rekeying delays.

The use of list for tracking moving members in HISCOM, KMGM, GKMW, and FEDRP prevent rekeying at the old area i during move events. Due to the lack of mobility list and using a naive mechanism for managing host mobility in LKH++, the rekeying messages overhead increases since the mobility is treated as a leave at the old area and a join in the new area.

Table 2. HISCOM comparison with other schemes.

Evaluation criteria	LKH++	KMGM	FEDRP	GKMW	HISCOM
Decentralized approach	×	✓	✓	✓	✓
Number of layer	1	1	2	2	2
Common TEK per group	✓	×	✓	✓	✓
DKM involved in host mobility key management	✓	×	×	✓	×

Decryption/re-encryption overhead	×	✓	×	×	×
Localized rekeying after move	×	✓	✓	✓	✓
Single point of failure	✓	×	×	×	×
Use of host mobility list	×	✓	✓	✓	✓
Moving member authentication	✓	✓	-	✓	✓

Table 3. Rekeying signaling load during mobility event.

Scheme	$AKM_i \leftrightarrow DKM$	$AKM_j \leftrightarrow DKM$	$M_i \leftrightarrow AKM_i$	$M_i \leftrightarrow AKM_i$	$M_j \leftrightarrow AKM_j$	$M_i \leftrightarrow DKM$
LKH++	-	-	-	-	-	$2 \log n$
KMGGM	-	-	1	1	-	-
FEDRP	-	-	1	2	1	-
GKMW	2	1	2	2	<1	-
HISCOM	-	-	1	<1	<1	-

Table 4 presents the storage overhead incurred by the various entities in the presence of α moves between $|A|$ areas. The number of residing member in area i , and the number of areas in the domain are respectively denoted by n_i and $|A|$. Storage overhead determine number of keys held by the DKM , AKM , and moving member M_i . The low key held by each entity result in fast execution and fast accessibility.

Table 4. Storage overhead at each entity.

Scheme	DKM	AKM_i	M_i
LKH++	$2 \log n + 1$	-	$\log n + 1$
KMGGM	-	$ A + n_i + 1$	$3 + 2\alpha$
FEDRP	$2 + A $	$4 + n_i$	$3 + \alpha$
GKMW	$2 + A + \alpha$	$4 + n_i + \alpha_i$	$4 + \alpha$
HISCOM	$2 + A $	$4 + n_i$	$3 + \alpha$

LKH++ is a centralized scheme and does not employ any AKM in managing keying materials. Therefore, there is no storage overhead on the $AKMs$. KMGGM adopted an independent TEK per subgroup approach, which leads to the elimination of the DKM . Thus, each area key manager needs to keep its own generated TEK in addition the $TEKs$ of the other areas in the domain. It is due to that the AKM is enabled to decrypt the content sent from other subgroup. As a result, more storage complexity is added to the AKM_i as seen in Table 4.

Moreover, the moving member needs to receive both the TEK and AEK associated with the visited in KMGGM because each area has its own independent TEK . Thus, M_i has to incur more storage overhead. In contrary, the moving member M_i needs to receive only the AEK of the visited areas in HISCOM, FEDRP, and GKMW as the TEK is common throughout the group, which reduce the key memory required at the member M_i .

For managing mobility event in GKMW, the DKM generates a session key and share it with the moving member and the visited area key manager. Thus, the storage overhead increase at the DKM , visited AKM and M_i . The number of the session keys that AKM_i needs to hold is denoted by α_i . KMGGM, FEDRP, GKMW, and HISCOM share a member key MEK_i with each member residing in its area hence the storage overhead at the AKM_i increase according the number of member residing member n_i in the area i .

5.2 Analytic model

MListen was a tool created for a project to capture information about join or leave event of a multicast in *MBone* (multicast backbone) session. Almeroth et al. (1998) used this tool to study the characteristics of the membership dynamics of *MBone* and showed that the member arrival into the session follows Poisson process with inter arrival rate λ (arrival/time), and its membership duration in the session is exponentially distributed with mean duration $\frac{1}{\mu}$ time unit [65], [66].

The proposed scheme divides a group communication domain into a number of areas equal with a given number like A . Once a group member arrives, it remains in area i for an amount of time that is exponentially distributed with mean $\frac{1}{\mu_i}$, and then transit to another area v . Since the scheme is modeled based on an open network extension to Jackson's theorem, each area is modeled as a $M/M/\infty$ queue [67]. Figure 5 shows the state diagram of the proposed group key management based on Jackson's network theorem. The arrows indicate the mean rates at which transitions occur between the areas in the domain.

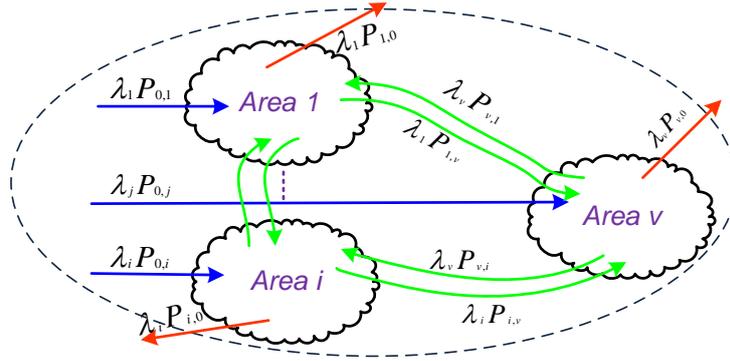


Figure 5. The state diagram of the group key management.

The state of a randomly selected member is modeled as a *Markov* process, where state i ($i \in \{1, \dots, A\}$) denotes a member residing in area i . A virtual state with index zero denotes the state of a member who is outside the group. Thereby, the inter arrival rate $\lambda_{0,i}$ denotes a new member enters the group session from area i . Since the members are permitted to move between areas, let $P = [p_{i,j}]$ ($\forall i, j \in \{0, \dots, A\}$) denote the transition probability among $A+1$ states or areas, and assume $p_{i,i} = 0$; and $p_{i,v} \geq 0$. The probability $p_{0,i}$ is the probability that a member arrives at area i to join the group, and the probability $p_{i,0}$ is the probability that a member leaves the group from area i . Under steady-state conditions, the arrival rate λ_i for area $i=1, \dots, A$ is obtained from the equation 1 ($\forall i, j \in \{1, \dots, A\}$) [68].

$$\lambda_i = \lambda_{0,i} + \sum_{j=1}^A \lambda_j p_{j,i} \quad (2)$$

A member possibly departs the group from the area a_i with the probability $p_{i,0}$. Therefore, the leaving probability of a member at area a_i from the group is obtained from the following equation

$$p_{i,0} = 1 - \sum_{j=1}^A p_{i,j} \quad (3)$$

The overall rate of members joining the group communication is denoted with λ and equal with

$$\lambda = \sum_{i=1}^A \lambda_{0,i} \quad (4)$$

The number of members residing in area a_i is denoted by n_i and calculated by Formula 4. Therefore, the probability of exactly k members residing in area a_i is obtained by Formula 14.

$$n_i = \frac{\lambda_i}{\mu_i} \quad (5)$$

$$P(n_i = k) = \frac{n_i^k}{k!} e^{-n_i} \quad (6)$$

5.3 Simulation model

This section presents the simulation model and some results obtained through several simulation experiments. A two tier distribution hierarchy with distinct five areas was designed for HISCOM, GKMW, and FEDRP. One *DKM* within the first tier is responsible for governing all *AKMs* 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 not an explicit *DKM* and its responsibility is delegated to the all existing *AKMs* in the group. The *DKM* has the main role of key management in *LKH++* and entertains all events occurred in the session, thus, the *AKMs* are not involved in this matter. Therefore, all requests are sent to the *DKM*.

In different experiments 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* is required 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

key with the new one. The rekeying policy for a member's move in LKH++, however, it does not provide an explicit mobility protocol, was considered as a leave in the old area and subsequently a join in the visited area.

The session time was assumed for 30 minutes. All new members enter the group through any of the areas with an inter-arrival average λ equal to 10 seconds. 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 equal 15 minutes. In order to study the impact of group size variation as one of the scalability requirements for the scheme performance, both parameters the inter arrival rate and membership duration can respectively vary [2sec : 30sec] and [10 min: 25 min] in separate experiments. The member remains in each area for a determined time (referred to as area dwell time), and then move to the other areas with the same probability of selection. In order to study the impact of members' mobility on performance overhead, the area dwell time will vary. Reducing the area dwell time will lead to increasing the members' mobility rates among areas. The velocity of members is set constant for all experiments, equal to 5 m/s. Table 5 summaries the simulation parameters and their various values used in different scenarios.

Table 5. Simulation parameters for the experiment scenarios

Parameters	Value
Number of Area	5
Session time	30 min
Inter-arrival	10 Sec (to study the impact of inter arrival variation [2sec : 30sec])
Session sojourn time	15 min (for study impact of membership duration [15 : 25] min)
Area dwell time	10 Sec (to study the mobility rate [1s:15s])
Velocity	5 m/sec

In order to evaluate each protocol, the communication complexity as the main performance metrics for group key management schemes associated with each scheme was assessed in terms of the number of rekeying messages required for updating keying materials in the old and new area when there is a change in group membership. The importance of the communication overhead is because of scarce radio resources used simultaneously by many mobile users. We assumed that there is no difference between unicast and multicast messages. In addition, we studied the *1-affects-n* phenomenon of each simulated scheme. Three parameters such as inter-arrival time, member sojourn time, and area dwell time have been changed to study the impact of group size and the mobility rate on HISCOM in what relates to efficiency in terms of rekeying communication overhead, and scalability in terms of *1-affects-n* behavior.

1. **Impact of the group size:** The scalability size of the proposed scheme is studied by changing the value of two controlling parameters in the simulation: the average inter arrival of members into the group session, and the average membership duration of members in the session. First, the average inter arrival value was varied in the simulation experiments from 2 second to 30 seconds while other simulation parameters were kept constant. The small number of inter arrival time resulted in a high population in the group about 600, whereas the number of group members achieves an average of 60 when the inter arrival time reaches 30 seconds. The average of leave and move also follow the group population trends.

The average membership duration (i.e. session sojourn time) as the other controlling parameter can influence the size of the group. The membership duration value varied from 10 minutes to 25 minutes [10min : 25min]. The membership duration (i.e. session sojourn time) is meant 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 results in reduction of leave rate and consequently increases of the remaining members in the session. The average rate of leave declines with increase of membership duration from about 230 to 50. While the join rate keeps steady about 330, the move event gradually increases.

2. **Impact of the movement rate:** 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 the 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.

When the mobility period value is equal to 1 second, the average rate of moves in the session reaches to about 19000. It gradually declines to achieve an average about 5900 inter moves in the session when the mobility period reaches to 15 seconds. In spite of mobility rate variation during the session, the rate of joins and leave remain steady respectively about 172 and 88.

5.3.1 Communication overhead

This requirement satisfies the bandwidth consumption of the wireless networks and devices. The high number of messages transmitted either by unicast or multicast during the performing rekeying process consume enormous network bandwidth, which result in delays in distributing the keying materials and disruption in the group communication service.

It can clearly be observed from Figure 6, Figure 7, and Figure 8 that LKH++ induces a higher number of rekeying messages than the others due to the lack of a protocol intended for managing the move events. Whenever any move occurs in the group, LKH++ affects the both old and visited area as it needs to perform rekeying process in the previous area as well as the visited area to achieve respectively forward and backward secrecy. Whereas other schemes induce null rekeying in the previous area due to the use of the mobility list and keeping the track of moving members.

Lack of strategy for handling the move event significantly increases the rekeying messages overheads particularly in the group with big size. Figure 6 and Figure 7 obviously depict the ratio of rekeying messages has increased when the group population grows up by increase of either inter arrival rate or membership duration.

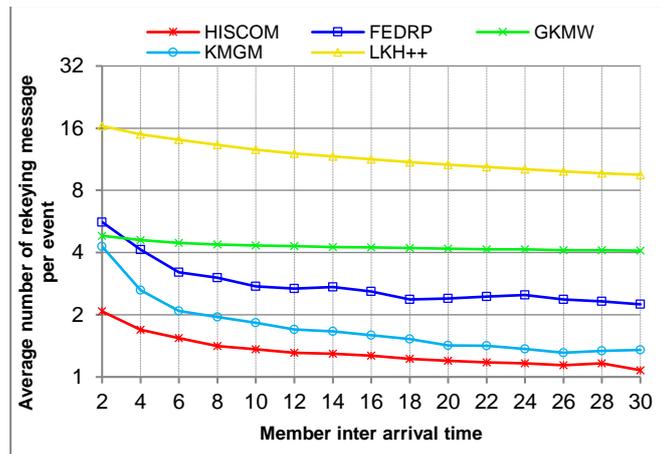


Figure 6. Impact of inter arrival variation on rekeying message overhead.

HISCOM, KMGM, FEDRP, and GKMW introduced the use of mobility list as to record the track of moving members such that the old area induces null communication overhead in move event and the visited area makes minimum communication overhead depending on the scheme design. Using this strategy results in improving bandwidth efficiency of the system while satisfying backward secrecy. Nevertheless, GKMW impose higher communication overhead than the others as depicted in Figure 6, Figure 7, and Figure 8. This is due to the required signaling messages for establishment of session mobility key between the moving member and the destination *AKM*. The session mobility key is generated by the *DKM* on the receipt of request from the *AKM* of the old area. The *DKM* requires to deliver this key to the visited area *AKM*, and the moving member through its old *AKM*. The session mobility is used for encryption/decryption messages between the moving member and the visited *AKM*.

FEDRP uses similar strategy for managing member mobility to GKMW. While GKMW uses a symmetric cryptography key for member authentication in the visited area, FEDRP uses asymmetric cryptography key for authentication of the moving members, which result in huge computation overhead at the member side and delay in obtaining service in the visited area. Furthermore, FEDRP empties the mobility list whenever an event occurs in the area. Thus, when a member returns back to an area where has already been visited, the *AKM* may not be able to find the track of moving back member as the mobility list has previously been emptied due to a change in area membership. The *AKM* treats the returning back member like a member who visits the area for the first time. Therefore, the communication overhead of FEDRP significantly increases, as the keying materials frequently require to be updated.

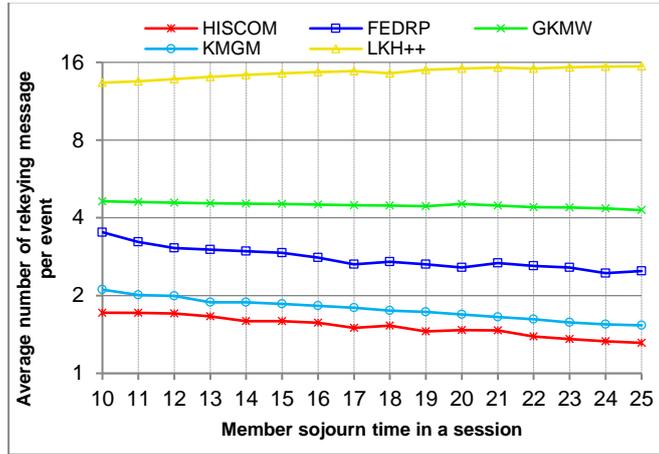


Figure 7. Impact of membership duration variation on rekeying message overhead.

The increase of the group size and the mobility period has a minimum influence on HISCOM and KMGM as the communication overhead remains low in comparison to the other solutions. This is because of the authentication mechanism used to verify the moving member in the visited area, which minimizes the signaling messages and avoids initiating extra rekeying process in move events. Since KMGM used an independent *TEK* per each area, the *TEK* corresponding to the visited area along with the *AEK* of the visited area must be sent to the moving member, which result in the size of messages sent to the member becoming bigger in contrast with other decentralized solutions. HISCOM, GKMW, and FEDRP need to send only the *AEK* of the visited area as the *TEK* is common for all areas. Thereby, rekeying process can become a hurdle for KMGM in a group with large size as the system bandwidth is enormously consumed by signaling messages.

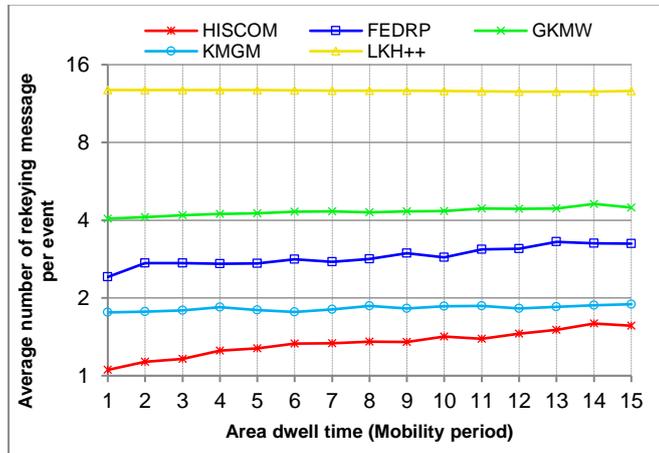


Figure 8. Impact of mobility rate variation on rekeying message overhead.

5.3.2 1-affects-n phenomenon

The *1-affects-n* phenomenon refers to the number of group members affected by a rekeying process due to any changes in the group membership. The high number of members involved in rekeying process on each event becomes a hurdle for the scheme to scale the scope of key management to very large group. Due to the intuitive characteristics of wireless devices in terms of wireless bandwidth limitation, high affected members consume enormous wireless resources which causes failure in receiving the new update of keying materials by some group members.

It can easily be seen from Figure 9, Figure 10, Figure 11, the high number of members are affected on each event in LKH++. The reason is LKH++ does not provide any protocol for move event and treats it as a leave in the old area and a join in the visited area thus, the entire group members are influenced with such an event twice. The average number of member affected by rekeying processes increases in this scheme especially when the group size grows up as depicted in Figure 9, and Figure 10. Since the number of affected member is proportional with logarithm of group members, with increase of group population the average of affected members rises up in the group. Therefore, LKH++ cannot scale to large group size.

HISCOM affects minimum group members by the update process of keying materials even when many members join the group and increase the group population as depicted in Figure 9. While, FEDRP, GKMW, and KMGM show high

overhead on *1-affects-n* phenomenon when the group size grows up particularly in situation the rate of inter arrival is fallen between time period [2s : 10s]. Thereby, the scheme scalability for FEDRP, GKMW, and KMGM becomes a hurdle for groups in which members enter with a short inter arrival time as the size of groups sharply grows up. Although when the rate of inter arrival increases, HISCOM, GKMW, and KMGM involve the minimum number of members in rekeying process.

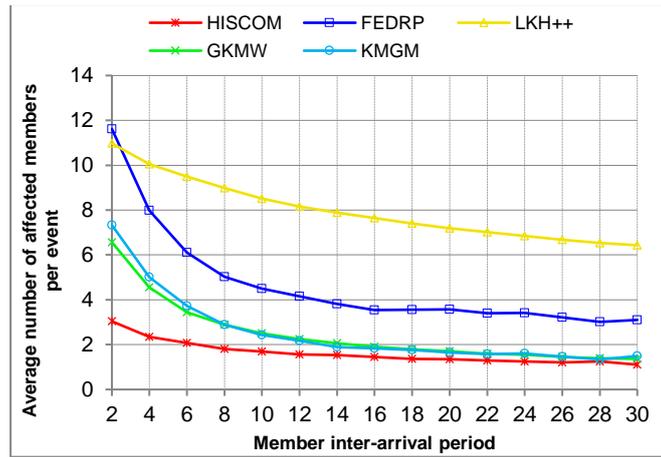


Figure 9. Impact of inter arrival variation on the average number of affected member.

The same strategy adopted by all schemes except LKH++ (i.e. the use of mobility list for keeping track of moving members) eliminates needs of performing rekeying process in the old area when a moving member returns back, which reduces the *1-affects-n* phenomenon overhead in all solutions more than LKH++. In FEDRP, the average of members affected on each event is fairly higher than HISCOM, KMGM, and GKMW. This is due to the fact that the mobility list is emptied whenever any membership changes occur in each area. Thus, in some cases that a moving member returns back to an area where he has previously visited, and the mobility list of the area has been emptied during the absence of the moving back member causes unnecessary rekeying process, which affects all residing members in the visited area.

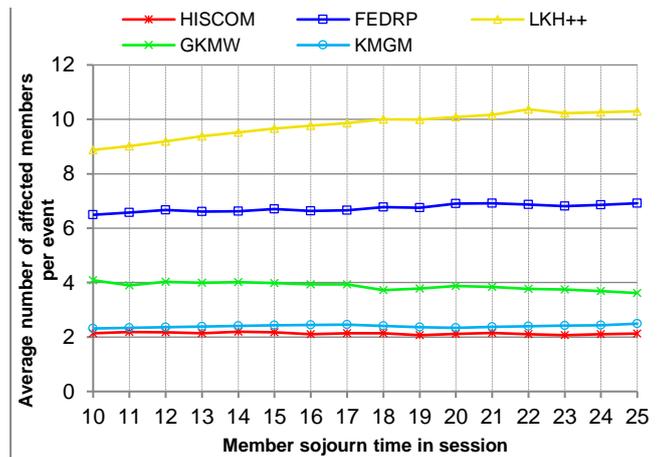


Figure 10. Impact of membership duration variation on the average number of affected member.

Figure 11 depicts a considerable reduction for the *1-affects-n* behavior in HISCOM particularly in highly dynamic environments in compare to the other solutions. It is because that The group members managed by HISCOM are considerably less affected with the group membership changes because the rekeying process is performed in the visited area when a member changes its location as long as the visiting member is not on the mobility list or its join time is after the last update time of keying materials in the visited area. Other solutions such as FEDRP, and GKMW carry out rekeying process if the member is not on the mobility list, which result in likely more updating keying materials as shown in Figure 10 and consequently more members are affected on each event as shown in Figure 11.

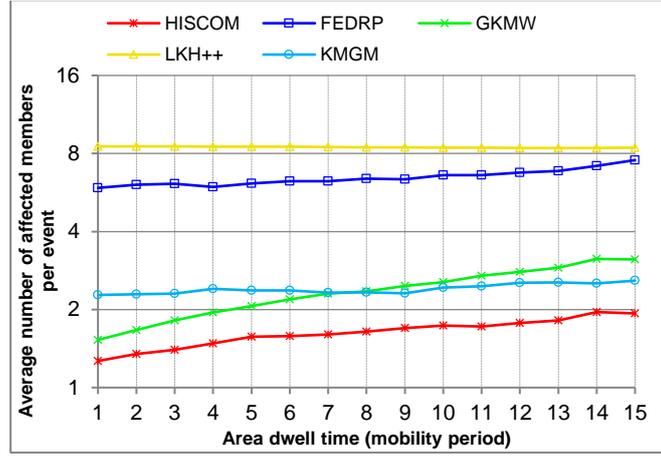


Figure 11. Impact of mobility period variation on the average number of affected member.

5.4 Security analysis

The proposed scheme precludes any eavesdropping opportunity when a moving member change its location. The provision of confidentiality 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 the time it joined the group in the visited area.

Preserving forward secrecy in the area where a member is moving out in order to enter another destination within a domain is certainly pointless because the member still remains in the session, in spite of changing its location. As a result, the mobility protocol does not require to maintain forward secrecy in the old area. Nevertheless, the leave protocol provides this option in the area which leave event occurs as well as all areas of which the leaving member holds their area encryption keys as long as these *AEKs* are still valid on the occurrence of leave event. All *AKMs* corresponds to these areas need to perform updating process for keying materials. Therefore, the forward secrecy requirement is achieved when any leave occur in the group throughout the domain.

Lemma 1: let D , AKM_k , and M_l respectively be the domain, the *AEK* used in area k , and an expelled member from area i . There is

$$\exists a_k \subset D \mid M_l \in AMOL_k \Rightarrow AEK_k \text{ is compromised.} \quad (1)$$

Proof

When M_l arrives in a new area, it receives the keying material (*AEK*) of the new area. The *AEK* of the old area and the *TEK* throughout the domain are not updated (Section 4.2). Thus, the expelled member knows all *AEKs* of areas where it has already visited. If $A = \{a_1, a_2, \dots, a_j\}$ is assumed as all areas visited by M_l , the expelled member is able to decrypt all messages sent to A since it knows *AEKs* of areas in A .

When M_l is expelled from area i , the *DKM* informs all area key managers that M_l has been expelled from the group and thereby the new *AEKs* are generated in areas where equation 2 is verified. The area key manager sends the new *AEK* to each member remaining in area i encrypted with the member key MEK_i^* of member M_i^* ($AKM_i \rightarrow M_i^* (i \neq l) : \{new_AEK_i\} MEK_i^*$). Thereby, each remaining valid member M_i^* in area would be able to decrypt the new *AEK* using its MEK_i^* . The expelled member would not able to access the new *AEK* since its associated member key is not used by AKM_i to encrypt the new *AEK*.

For each area k where equation 2 is verified, to distribute the new *AEK*, AKM_k sends to each member $M_k (k \neq l)$ remaining in its area the new *AEK* encrypted with MEK_k associated with each member. The MEK_l of expelled member is excluded in encryption process. Therefore, M_l will not be able to access to the new *AEK* in area k .

An intruder cannot access to the new *TEK*, and hence to the content, since it does not know any member key MEK_k of a remaining legitimate member, and thereby cannot decrypt the new *AEK*. It also does not know any AEK_i of areas it has not visited yet. Therefore, the intruder is disable to have access to the new *TEK* in the group, as he does not know the *AEK* of every areas.

The impersonation attacks are also impossible as the area key manager and the moving members mutually authenticate each other on every mobility events. The *AKM* verifies each member against the member key *MEK*, which is derived using the security parameters generated and shared by the trusted *DKM* and the information associated with the specific member.

Table 6 provides a comparison between HISCOM, KMGM, FEDRP, GKMW, and LKH++ in terms of backward secrecy and forward secrecy. From the Table 6, 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. In some cases, the moving member may have access to 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.

GKMW breaches forward secrecy in the 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 areas which has already been visited. LKH++ performs unnecessary rekeying process for provision of forward secrecy in the old area since it treats move event as a leave in the old area and a join in the new area despite of the moving member is still remaining in the session.

Table 6. Comparison of rekeying *TEK* and *AEK* in move event between different schemes.

Scheme	Forward secrecy		Backward secrecy	
	Leave	Move	Join	Move
LKH++	✓	✓	✓	✓
KMG	✓	✗	✓	✗
FEDRP	✓	✗	✓	✓
GKMW	✗	✗	✓	✓
HISCOM	✓	✗	✓	✓

6 CONCLUSION

Designing a group key management scheme in wireless mobile environment becomes more complicated since wireless devices can move freely between areas and subnets of networks, which causes more difficulty in key management and member authentication. In this paper, a new scheme HISCOM has been proposed to improve the key management performance in wireless mobile environments. It considered backward secrecy where moving group members dynamically changes their locations while seamlessly maintaining the session. HISCOM used a new rekeying strategy based on member join time and area key update time, and AMOL for effectively performing key management and authentication phase, as well as avoiding renewing the *TEK* respectively during move events. HISCOM adopted decentralized approach with a common *TEK* for all areas to localize rekeying, and alleviate *1-affects-n* phenomenon and path decryption/encryption overheads. By delegating the authentication phase of moving members to the intermediate *AKMs*, the *DKM* is given scalability and preserved from bottleneck as the signaling loads reduce at the core domain. The HISCOM was modeled by analytical formal method for evaluating the communication cost and *1-affects-n* phenomenon. Simulation results depicted HISCOM considerably reduced rekeying overhead and increased the scalability of key management with decreasing the affected members on each event while maintaining backward secrecy in move events.

ACKNOWLEDGEMENT

The authors would like to acknowledge the financial support of eScience fund 01-01-03-SF0786.

REFERENCES

1. Cisco Visual Networking Index (2016). Global mobile data traffic forecast update, 2015-2020. *Cisco white paper: Cisco systems*.
2. Sathiaselvan, A., & Crowcroft, J. (2012). Internet on the move: challenges and solutions. *SIGCOMM Comput. Commun. Rev.*, 43(1), 51-55, doi:10.1145/2427036.2427046.
3. 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.
4. 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>.
5. 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>.
6. Deering, S. E., & Cheriton, D. R. (1990). Multicast routing in datagram internetworks and extended LANs. *ACM Trans. Comput. Syst.*, 8(2), 85-110, doi:10.1145/78952.78953.
7. Cisco Systems (2012). Internet Protocol Multicast. http://docwiki.cisco.com/wiki/Internet_Protocol_Multicast. Accessed 15/08/2016 2016.
8. 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.
9. 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.
10. 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.
11. Savola, P. (2008). Overview of the Internet Multicast Routing Architecture. *Internet Engineering Task Force*.

12. 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.
13. Nguyen, H. L., & Nguyen, U. T. (2008). A study of different types of attacks on multicast in mobile ad hoc networks. *Ad Hoc Networks*, 6(1), 32-46.
14. Iqbal, S., Mat Kiah, M. L., Daghighi, B., Hussain, M., Khan, S., Khan, M. K., et al. (2016). On cloud security attacks: A taxonomy and intrusion detection and prevention as a service. *Journal of Network and Computer Applications*, 74, 98-120, doi:<http://dx.doi.org/10.1016/j.jnca.2016.08.016>.
15. Baugher, M., Canetti, R., Dondeti, L., & Lindholm, F. (2005). Multicast Security (MSEC) Group Key Management Architecture. RFC 4046. Internet Engineering Task Force.
16. Kim, Y., Perrig, A., & Tsudik, G. (2004). Group Key Agreement Efficient in Communication. *IEEE Trans. Comput.*, 53(7), 905-921.
17. Kim, Y., Perrig, A., & Tsudik, G. (2004). Tree-based group key agreement. *ACM Trans. Inf. Syst. Secur.*, 7(1), 60-96.
18. Rafaeeli, S., & Hutchison, D. (2003). A survey of key management for secure group communication. *ACM Comput. Surv.*, 35(3), 309-329.
19. 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.
20. 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>.
21. Challal, Y., & Seba, H. (2005). Group key management protocols: A novel taxonomy. *International Journal of Information Technology*, 2(1), 105-118.
22. 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.
23. Gharout, S., Challal, Y., & Bouabdallah, A. (2008). Scalable delay-constrained multicast group key management. *International Journal of Network Security*, 7(2), 142-156.
24. Mat Kiah, M. L., & Martin, K. M. Host Mobility Protocol for Secure Group Communication in Wireless Mobile Environments. In *Proceedings of the Future Generation Communication and Networking 2007* (Vol. 01, pp. 100-107): IEEE Computer Society. doi:<http://dx.doi.org/10.1109/FGCN.2007.144>.
25. 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.
26. Heba K, A. (2004). A scalable and distributed multicast security protocol using a subgroup-key hierarchy. [doi: 10.1016/j.cose.2003.11.003]. *Computers & Security*, 23(4), 320-329.
27. Schmidt, T., Waehlich, M., & Fairhurst, G. (2010). Multicast mobility in mobile IP version 6 (MIPv6): problem statement and brief survey. RFC 5757. Internet Engineering Task Force.
28. 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.
29. Al-Surmi, I., Othman, M., & Mohd Ali, B. (2012). Mobility management for IP-based next generation mobile networks: Review, challenge and perspective. *Journal of Network and Computer Applications*, 35(1), 295-315, doi:<http://dx.doi.org/10.1016/j.jnca.2011.09.001>.
30. Daghighi, B., Mat Kiah, M. L., Shamshirband, S., Iqbal, S., & Asghari, P. (2015). Key management paradigm for mobile secure group communications: Issues, solutions, and challenges. *Computer Communications*, 72, 1-16, doi:<http://dx.doi.org/10.1016/j.comcom.2015.05.008>.
31. 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.
32. Wallner, D., Harder, E., & Agee, R. (1999). Key Management for Multicast: Issues and Architectures. RFC 2627.: Internet Engineering Task Force.
33. Yan, S., & Liu, K. J. R. (2007). Hierarchical Group Access Control for Secure Multicast Communications. *Networking, IEEE/ACM Transactions on*, 15(6), 1514-1526.
34. 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.
35. 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.
36. Je, D.-H., Lee, J.-S., Park, Y., & Seo, S.-W. (2010). Computation-and-storage-efficient key tree management protocol for secure multicast communications. [doi: 10.1016/j.comcom.2009.08.007]. *Computer Communications*, 33(2), 136-148.
37. Steiner, M., Tsudik, G., & Waidner, M. Diffie-Hellman key distribution extended to group communication. In *Proceedings of the 3rd ACM conference on Computer and communications security, New Delhi, India, 1996* (pp. 31-37): ACM. doi:<http://doi.acm.org/10.1145/238168.238182>.
38. 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.
39. Zheng, S., Manz, D., & Alves-Foss, J. (2007). A communication-computation efficient group key algorithm for large and dynamic groups. [doi: 10.1016/j.comnet.2006.03.008]. *Computer Networks*, 51(1), 69-93.
40. Magliveras, S., Wandl, 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,
41. 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.
42. 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>.
43. Hardjono, T., Cain, B., & Monga, I. (2000). Intra-Domain Group Key Management Protocol. <http://tools.ietf.org/html/draft-irtf-smug-intragkm-00>.
44. Mitra, S. (1997). Iolus: a framework for scalable secure multicasting. *SIGCOMM Comput. Commun. Rev.*, 27(4), 277-288.
45. 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. [doi: 10.1016/j.comnet.2007.07.007]. *Computer Networks*, 51(17), 4727-4743.
46. Cho, J.-H., Chen, I.-R., & Wang, D.-C. (2008). Performance optimization of region-based group key management in mobile ad hoc networks. [doi: 10.1016/j.peva.2007.07.002]. *Performance Evaluation*, 65(5), 319-344.
47. Li, J. H., Bhattacharjee, B., Yu, M., & Levy, R. (2008). A scalable key management and clustering scheme for wireless ad hoc and sensor networks. [doi: 10.1016/j.future.2008.03.007]. *Future Generation Computer Systems*, 24(8), 860-869.
48. 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.
49. Hur, J., & Yoon, H. (2009). A Decentralized Multi-Group Key Management Scheme. *IEICE Transactions on Communications*, 92, 632-635.

50. 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>.
51. Hyttiä, E., & Virtamo, J. (2007). Random waypoint mobility model in cellular networks. *Wireless Networks*, 13(2), 177-188, doi:10.1007/s11276-006-4600-3.
52. Narmawala, Z., & Srivastava, S. (2015). Community Aware Heterogeneous Human Mobility (CAHM): Model and analysis. *Pervasive and Mobile Computing*, 21, 119-132, doi:<http://dx.doi.org/10.1016/j.pmcj.2014.12.008>.
53. Wang, J., Jiang, C., Quek, T. Q. S., Wang, X., & Ren, Y. (2016). The Value Strength Aided Information Diffusion in Socially-Aware Mobile Networks. *IEEE Access*, 4, 3907-3919, doi:10.1109/ACCESS.2016.2600526.
54. Habib ur Rahman, M., Liew, C. S., Wah, T. Y., Shuja, J., & Daghighi, B. (2015). Mining personal data using smartphones and wearable devices: A survey. *Sensors*, 15(2), 4430-4469.
55. 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.
56. 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,
57. Jong-Hyuk, R., & Kyoong-Ha, L. Key management scheme for providing the confidentiality in mobile multicast. In *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference, 20-22 Feb. 2006 2006* (Vol. 2, pp. 1205-1209). doi:10.1109/icact.2006.206187.
58. Kamat, S., Parimi, S., & Agrawal, D. P. Reduction in control overhead for a secure, scalable framework for mobile multicast. In *2003* (Vol. 1, pp. 98-103 vol. 101): IEEE
59. DeCleene, B., Dondeti, L., Griffin, S., Hardjono, T., Kiwiior, D., Kurose, J., et al. (2001). *Secure group communications for wireless networks*. Paper presented at the Military Communications Conference, (MILCOM 2001). ,
60. Gharout, S., Bouabdallah, A., Kellil, M., & Challal, Y. (2010). *Key management with host mobility in dynamic groups*. Paper presented at the Proceedings of the 3rd international conference on Security of information and networks, Taganrog, Rostov-on-Don, Russian Federation,
61. Kiah, M. L. M., & Daghighi, B. (2012). An Implementation of Secure Group Communication in a Wireless Environment. *International Journal of Computer and Electrical Engineering*, 4(6), 850.
62. 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.
63. Srinivas, V., & Lu, R. An efficient reliable multicast protocol for 802.11-based wireless LANs. In *World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a, 15-19 June 2009 2009* (pp. 1-6). doi:10.1109/WOWMOM.2009.5282455.
64. Frankel, S., & Kelly, S. G. (2007). Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec. Internet Engineering Task Force.
65. Almeroth, K. C., & Ammar, M. H. Collecting and modeling the join/leave behavior of multicast group members in the mbone. In *High Performance Distributed Computing, 1996., Proceedings of 5th IEEE International Symposium on, 1996* (pp. 209-216): IEEE
66. Almeroth, K. C., & Ammar, M. H. (1997). Multicast group behavior in the Internet's multicast backbone (MBone). *Communications Magazine, IEEE*, 35(6), 124-129.
67. Nelson, R. (2013). *Probability, stochastic processes, and queueing theory: the mathematics of computer performance modeling*: Springer Science & Business Media.
68. Bolch, G., Greiner, S., de Meer, H., & Trivedi, K. S. (2006). *Queueing networks and Markov chains: modeling and performance evaluation with computer science applications*: John Wiley & Sons.