Group Key Management Protocols for Secure Mobile Multicast Communication: A Comprehensive Survey

Trust Tshepo Mapoka
University of Bradford
United Kingdom

ABSTRACT

Key management is equally important as compared to any other security measure such as encryption and authentication. With the growing usage of mobile devices and the advent of multicast communication, there has been a significant amount of work carried out in developing an optimum group key management protocol for mobile multicast systems. Key management is widely being adopted in securing group communication for both wired and wireless networks. Securing group communication over wired networks is fairly well established; however, wireless networks bring additional challenges due to member mobility and increase in the number of members. This paper presents a comprehensive survey of group key management protocols in wireless mobile environments that employ multicast communication. They are classified into network dependent and independent protocols and further categorized into tree-based and cluster-based key management protocols. The survey clearly outlines the characteristics of each protocol along with highlighting their advantages and limitations with respect to real-world systems. The paper is concluded with a taxonomy of the individual protocols with respect to the defined requirements which provides a strong source of literature reference for researchers in this field.

Keywords: Mobile multicast, security, group key management, wireless network.

1. INTRODUCTION

In recent years multicast communications from both from the Internet Service Providers (ISPs) and content or media providers and distributors have gained popularity. IP multicast is increasingly used as an efficient communication mechanism for delivery of group-oriented applications such as video conferencing, interactive group games, video on demand (VoD) and etc. over the internet to multiple members. Transmitting multimedia content over wireless environment to a group of members with low bandwidth requires multicast. Though multicast provide attractive features it has some security vulnerabilities [1]. In order to counteract the vulnerabilities for secure multicast, a key management system is required. Group key management is one of the basic building blocks in securing group communications. Many efforts have been conducted to address the issues related to group key management such as: Access control, confidentiality, data integrity and authentication, etc. Data confidentiality is pronounced to limit access to the multicast content by encrypting it with the group key knows as the Traffic Encryption Key (TEK). The TEK is shared to all legitimate members prior to start of the multicast session by the Key Distributor usually. Whenever the service provider sends the multicast content, it is encrypted using the TEK which the legitimate members can decrypt on multicast content delivery. The illegitimate members may receive the content too, but are restricted from accessing the content without a valid TEK. So the communications among participating members of the multicast group should be secure. However, during the lifetime of the multicast session there are still some problems that need to be addressed. The members in the group may be changed frequently due to dynamic membership change caused by member joins/leaves. This requires the TEK to be updated to a new one for forward and backward secrecy. This process is known as rekeying. Forward secrecy ensures that the leaving member cannot access next communication key while backward secrecy ensures that the new joining member to a group cannot access the previous communication key. Therefore rekeying is obligatory in any group key management protocol. In some instances rekeying may occur when the validity of the session/key expires. A member can only participate in the session after receiving the new TEK. How to generate and distribute the new TEK securely without service disruption and without degrading the network performance remains a big challenge. Previous work [2-4] surveys the group key management protocols for addressing rekeying when dynamic group memberships occurs in wired networks without considering host mobility in wireless environment. Work in [5] surveys the mobility challenges and solutions in wireless mobile multicast environments without considering the complexity of key management during member mobility. Since researchers are now keeping an eye on mobility, members will be able to access their multimedia content ubiquitously using portable devices without service disruptions. Work in [6-8] address the mobility issue in wireless environment where members perform handoff across networks with improved service latency. However the complexity of key management rises too because a moving member may not be unknown in the new area it visits even though a fast handover mechanism occurs. Complexity arises due to whether to perform rekeying in both old and new areas or not because a member moving is considered as evicted member in old area and as joining member in the target area while remaining as a valid member in the session. Thus, a further research is required to enforce the deployment of efficient group key management protocol to deal not only with dynamic group membership (join and leave) but also with this additional complexity, dynamic member location.

The rest of the paper is organised as follows: Section 2 presents the group key management classification and requirements. Section 3 and 4 presents the network independent and dependent approaches in wireless systems. Section 5 concludes the paper.

2. GROUP KEY MANAGEMENT CLASSIFICATION AND REQUIREMENTS

The security of group communication depends on the protection of the TEK. Group key management is necessary to control key generation, distribution and updating. The crucial problem is to achieve a scalable rekey technique which is triggered after each membership change. The number of TEK update messages may be important during frequent join, leave and move operations in wireless networks. Work in [2-4] surveys the existing group key management protocols for wired networks. They classify them according to centralized, decentralized and distributed frameworks. Work in [2] further categorize the group key management schemes in to either common TEK and independent TEK per subgroup. The various group key management protocols are classified as shown in Figure 1. To design an efficient key management protocol for secure mobile multicast, some miscellaneous requirements have to be considered. The requirements from three facts of view are summarized into: security requirement, performance requirements and operational efficiency as follows:

2.1. Security Requirements

Forward secrecy: Forward secrecy guarantees that a group member who has left the group is denied access to any future group keys. This guarantees that an evicted member cannot decrypt data after it leaves the group.

Backward secrecy: Backward secrecy guarantees that a new member joining the multicast group is denied access to any previous group keys. This guarantees that a joining member cannot decrypt data transmitted before it joins the group.

Key independence: Key independence requires that entire keying materials used should be absolutely independent from each other. This feature ensures that the disclosure of one key should not compromise other keys in any secure wireless group key management system.

Collusion freedom: Collusion freedom requires that a set of departing members/fraudulent members should not cooperate to deduce the current TEK. This is usually done by applying the old keying materials known to the evicted members.

Trust Relationship: Trust relationship ensures that any secure wireless group key management protocol should not place trust in a huge number of communicating entities such as any intermediate or third party.

Resilience: It is important to include the threat model and the security analysis when deploying and evaluating the group key management protocol. Both network-based attacks and service-related attacks should be considered.

2.2. Performance Requirements

1-affect-n phenomenon: A Key management protocol suffers from the 1-affects-n where a single group membership change join/leave results in a re-keying process that disturbs entire group members to update the TEK as discussed in [2].

Scalability: Scalability is used to efficiently determine the issues such as capability to handle variable group sizes, highly dynamic membership changes and broadly scattered memberships.

Acceptable delays: Typically multimedia applications are very sensitive to jitters and delays in packet delivery.

Service availability: In the event a single entity fails in the key management protocol, this must not prevent the affect the normal operation of the entire multicast session.

Minimal Bandwidth: As wireless mobile networks are susceptible to frequent disconnection, and the available bandwidth is limited, the communication overhead between group entities should be kept to a minimum.

Multiple membership change: This feature should satisfy scalability issue. Key management protocol in wireless mobile environment should handle multiple membership changes without adding rekey overheads.

2.3. Operational Efficiency Requirements

Communication cost: The key management scheme must not induce high number of signalling messages transmitted (either unicast or multicast) during rekeying process by the Key Distributor Server (KDS). This requirement satisfies the bandwidth requirements of the system.

Storage Cost: The total number of keys stored by the communicating entities (KDS and mobile devices) should be kept as low as possible. This requirement enables fast execution and fast accessibility of stored keys.

Computation Cost: The cost of ciphering and deciphering operations in the group key management protocol should be low in order to obtain the updated TEKs efficiently. This requirement enables processing speed and time of the communicating entities.

3. NETWORK INDEPENDENT APPROACH TO WIRELESS NETWORKS

Extension of network independent group key management protocols to wireless environment could encounter challenges that prevent them from performing efficiently since wireless networks are bandwidth limited. It also consists of resource constraint and widely dispersed mobile devices with less computational capability, less storage capacity and limited power supply. This section analyses the performance challenges of network-independent group key management protocols if extended to wireless mobile networks.

3.1. Centralized schemes challenges

Centralized schemes can encounter the following challenges when applied to wireless networks:

Lack of scalability: Wireless networks offers flexibility to group members by allowing them to receive multicast sessions ubiquitously. With the ever increasing number of widely dispersed members in wireless network, this induces a huge number of rekeying messages. For large and highly dynamic group application, frequent rekeying may overwhelm the capability of a single Key server hence triggering failure of entire group key management operations.

Operational inefficiency: In wireless networks it is difficult to track the location of members without the base station like in cellular wireless networks since members are highly dynamic. This enforces rekeying messages to be multicast to the entire wireless domain.

Inability to support multiple membership change: The key management structure for centralized schemes is specific to a particular multicast session. Therefore a separate key tree is established for each multicast session. If a member is enforced to participate in multiple sessions, this will enable a member to register with several key management trees. In this case huge number of keys needs to be stored and managed by the member. On multiple membership change, several key trees and all the members residing in the tree will be affected hence inducing large number of rekeying messages to the wireless network.

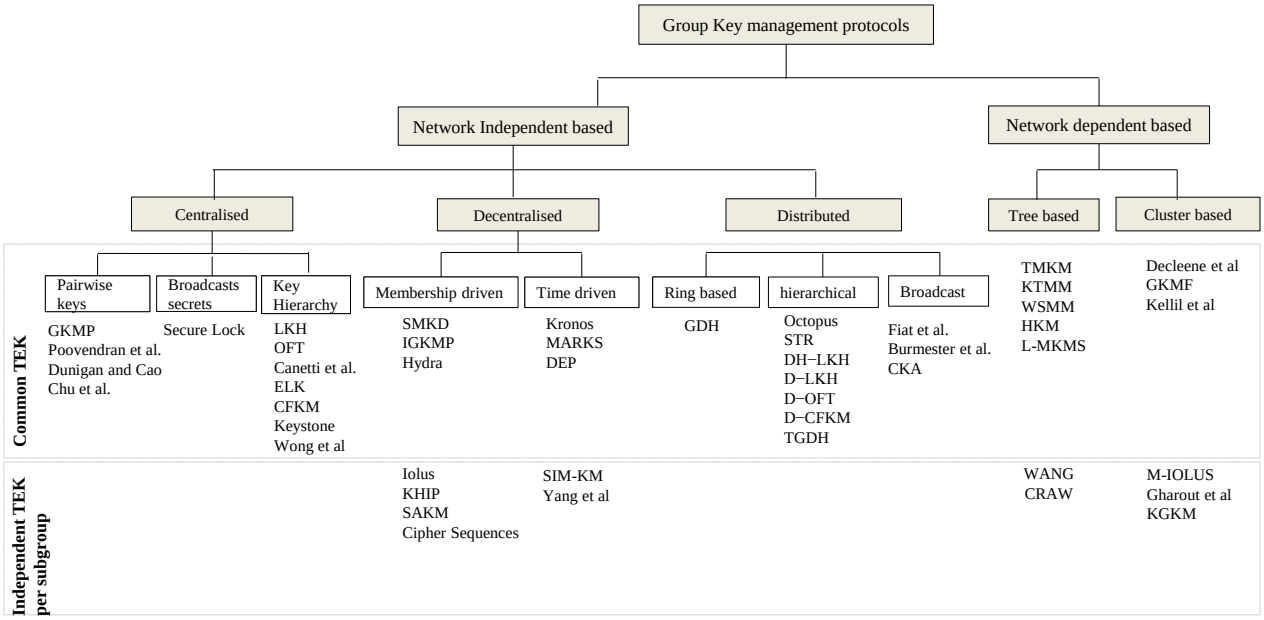

Figure 1: Classification of Group Key management Protocols

3.2. Decentralized schemes challenges

Decentralized schemes can encounter the following challenges when applied to wireless networks:

Collaboration with efficient Key distribution protocols: Traditional decentralized schemes only propose a large scale framework for group key management without considering an approach for efficiently distributing keying materials to group members within the subgroup. In this challenge, decentralized schemes need to collaborate with other efficient group key management schemes to build an integrated solution for wireless networks.

Establishment of trust relationship: The existence of third party entities in decentralized schemes is a critical security concern that needs to be considered by wireless group key management protocols [9].There should be some form of security association between the third parties to maintain the trust level.

Integration of authentication with group key management: Decentralized schemes consist of subgroups which may belong to same or different networks. To enable the flexibility of members by accessing group content ubiquitously across wireless network subgroups, members have to be verified by authentication before they could obtain the necessary key material used in the target subgroup. The group key management protocol should also consider an efficient group authentication mechanism in case of multiple moves.

3.3. Distributed schemes challenges

Distributed schemes can encounter the following challenges when applied to wireless networks:

Operational inefficiency: The fault-tolerance feature in distributed schemes sacrifices operational efficiency at the cost of communication and computation cost. The use of Diffie-Hellman key exchange protocol which widely applied in distributed schemes to compute the TEK is computationally expensive due to its exponentiation. Moreover, it takes a long time for all collaborating members to reach the TEK. As the group size gets large, the conjunction time for key generation linearly increase with expanding group size.

4. NETWORK DEPENDENT APPROACH

Group key management faces some complexities when extended to wireless networks. In order to deploy an efficient group key management protocol over wireless networks, group key management protocols should be network dependent. Network dependent protocols rely on the features of the underlying network infrastructure for them to operate efficiently. Network dependent group key management protocols support mobile multicast whereby members move over a widely distributed wireless network and continue receiving their subscribed multicast services. When a host moves it changes its point of attachment from one access area to another and this induces more complexity on handling key management and traffic destined for the mobile host regardless of whether a fast handover mechanism is used. The main challenge with respect to member mobility in a wireless network is that needs to be addressed is how to handle re-keying efficiently and securely while mobile members in intra-group inter-area join/leave the group such that the data remains secure and undisrupted while the overall impact on system performance is minimized. The mobile multicast security group key management protocols is listed and categorized in Table 1 and Figure 2 respectively. They are classified into tree based or cluster based; wherein the tree-based approach adopts the LKH scheme [37] and the cluster-based or hierarchy-based approach partition groups into a hierarchy of subgroup/cluster managers.

4.1 Tree based Schemes

This section lists all the tree-based mobile multicast group key management protocols for wireless networks. They are listed and briefly described as follows:

4.1.1 Topology Matching Key Management Tree (TMKM):

In this solution the authors of [10] [11] adapt the traditional key tree (LKH key tree) to the cellular wireless network model of [12] and matches the key management tree to a three-level topological structure. The model consists of network entities such as the mobile users, base stations (BS) and a supervisor host (SH). The BSs perform key management inside its scope then multicasts the useful keying material to its members and they

operate under the jurisdiction of the SH. The SH handles most of the routing and protocol detail for mobile users, and manages the keying materials including the group key (TEK) and the supporting keys to protect group communication. When users move between cells, an efficient handoff mechanism handles the relocation of that user in TMKM tree. Each cell has a corresponding Wait to be removed list (WTB) that tracks previous and current cell members. As member α moves from cell *i* to cell *j* it is treated as leaving cell *i* and rejoining cell *j*. The major benefit of using TMKM approach is that it has low communication overheads by broadcasting rekey messages to the useful member unlike network independent centralized group key management approaches that needs to broadcast rekeying messages within the whole wireless domain. However it suffers from the 1-affect-n phenomenon since it uses common TEK approach, hence rekeying affects the entire group. In the user sub tree, the BS needs to multicast all rekeying messages to the members within the cell to achieve key updating. TMKM lacks trust relationship because both the BSs and SHs are provided by a third party. TMKM uses centralized structure, by relying on GKC which is single point of failure. If the GKC is out of service, the entire group key management system becomes unavailable. Moreover, the authors claim that dividing the network into two separate levels, wired and wireless networks, improves the overall performance of re-keying in wireless networks. Multiple-layer structure increases the number of levels which complicate key management. The capacity of the GKC may be overwhelmed by a large number of rekeying requests which results in to poor performance in the actual development for highly dynamic membership changes. Highly mobile members will definitely face service disruptions due to delays in key updating. In TMKM, the physical location of a mobile user affects the user's position on the key management tree. When a mobile user moves from one cell to another cell, the mobile user needs to be relocated on the TMKM tree. When the users move faster, the TMKM tree leads to decrease in efficiency, which is the situation when frequent handoffs occur.

4.1.2 A Hybrid Key Management Scheme (HKM):

The authors of [13] propose a hybrid key management (HKM) tree suitable for mobile IP environments with topology matching (TM) sub trees and topology independent (TI) sub trees similar to TMKM [10] [11]. HKM is a network dependent protocol which is inspired by the topology of hierarchical cellular networks. In order to minimize the communication rekeying overheads caused by member handoffs, the HKM tree adopts unique key management trees (TIKM trees and TMKM trees) then combine them to manage high mobility users and low mobility users. These schemes enables rekeying messages for low mobility users to be delivered to the specified location and rekeying messages for high mobility users only need to be broadcast when the users leave the group regardless of any number of handoffs happen**.** The HKM tree is partitioned into two sub-trees: the TI sub tree for high mobility users with velocity $(V>V_0)$ and the TM sub tree for low mobility users with velocity $(V<V_0)$, where V_0 is the threshold velocity. Individual users store the private key, the session key and a set of KEKs on the path from itself to the root node of the key tree. HKM perform two operations; rekeying operation for high mobility users which is triggered by only when join/leave in the group occurs, and locates the delivery of the rekeying messages to low mobility users who do not undergo frequent handoffs. Rekeying operation for HKM is unique from the traditional rekeying method LKH [14] [15].

HKM tries to solve the problem of TKMM by making sure the physical location of a mobile user does not affect the user's position on the key management tree. It adopts a combination of advantage of both TIKM trees and TMKM trees to manage low mobility users and high mobility users. Bandwidth is not wasted in HKM because it is able to locate the delivery of rekeying messages for low mobility users. However bandwidth is wasteful for high mobility users because HKM broadcast the rekey messages to all the BSs. Even though rekeying messages are reduced in HKM, it uses common TEK approach, which suffers from 1 affect n phenomenon. Update of the group key Ks affects both the TI sub tree and TM sub tree for low mobility and high mobility respectively hence the protocol cannot handle highly dynamic environment with multiple membership change. This would induce more signalling messages which cannot be handled by the central GM which is a single point of failure.

4.1.3 WANG Scheme:

Authors of [16] and [17] propose a distributed network dependent group key management protocol which divides group members into leader units and general member units. The leader units participate in key management on behalf of general member units. The authors also propose null area re-keying in both old and new ones on member movement across two areas. For this purpose, a handoff member mechanism is defined to handle the member mobility. The scheme consists of 2-tier logical framework in order to match the topology of cellular networks which consists of the key server to significantly reduce the communication cost during the key updating. The network entities involved are: Group key server (GKS) in the first level which generate group key and distribute it by multicast it to control units in the second level, Independent Cell Key Servers (CKS) per cell in the second level which receives the group key from the GKS and further distributes it to the members in its wireless cell. In the logical structure, the CKS partition group members into 2 roles: The leader unit which is at the leader level and general member unit which is at the user level. The member in the leader unit is allocated as leader of the member unit, i.e. one leader for one member unit. Leaders help the CKS to take part in the key distribution. In the user level, there are many member units to accommodate general group members. Rekey process happens in these units on membership change. This rekey process is known as the micro key management.

The protocol uses decentralized framework for scalability and takes the advantage of the network independent centralized group key management tree to reduce rekeying overhead and workload from the server. In addition, the key management operations can be performed in parallel in all the cells simultaneously. However it does not satisfy backward and forward secrecy by maintaining the area keys in both sides for mobile members. The distributed two tier logical structure is not matched to the cellular network topology which makes the WANG scheme not suitable for wireless implementation. Moreover, it is complicated to select the leader when member evict at the leader unit. The huge number of keys stored by the mobile device which is resource constrained device can affect the operational performance of the mobile device. The protocol does not consider the huge number of messages that can overwhelm the server which is also a single point of failure when multiple membership change occurs. This protocol cannot support highly mobile users who require fast rekeying due authentication delay caused by contacting the previous area server every time a member moves. Increased number of level in WANG scheme complicates key management and delays packet delivery.

4.1.4 Combination of Rekeying and Authentication in Wireless Networks (CRAW):

In order to achieve an efficient rekeying method the authors of [18] proposed this protocol which operates by combining member authentication procedure with group key management. They use Simple and Password Authentication protocol (SAS)

[19] for member authentication and use an efficient network independent key management protocol known as CKC (Code for Key Calculation) [20] for group key management in wireless networks. This combination provides a simple and secure mechanism while mobile members join/leave a group or move inter-area. The CRAW scheme adopts a decentralized cellular wireless network framework with the main server which distributes the multicast content to individual Area Wireless Server (AWS). Main server maintains the main list which contains member information regarding join/leave and movement. AWS which performs member authentication process generates, sends the Area group key, and forwards the content to the mobile members. CRAW protocol collaborates with CKC which is more efficient than LKH to manage rekeying in each subgroup. CKC is an improved version of logical key hierarchy (LKH). It reduces the workload from the server by enabling members to compute necessary u-node and k-node keys of the tree using node codes and one-way hash function after receiving the updated group key after membership change. The security of the keys in CKC relies on one way hash function. The CRAW operation is partitioned in to two phases: authentication phase when the member first joins the group and when the member moves, and the group key management phase which enables the update of group key and area group key on membership change due to join/leave and update of the area keys only when a member moves.

The CRAW protocol provides a scalable decentralized framework with Independent TEK per subgroup. CRAW takes advantage of CKC which reduces the workload from the AWSs since members are allowed to participate in generating their essential keys. CARW also take advantage of the refreshed one-time password by using it a member individual key. AWS does not need to generate the individual keys for members when they move to different areas. However it suffers from 1 affect n phenomenon within the clusters area due to using CKC with common TEK approach. The protocol requires member re-authentication on handoff which relies on the main server which is a single point of failure and also the attack target. If the main server is far from the serving area, this will induce unnecessary delays. The protocol does not address multiple authentications at join or move which can overwhelm the main server. Therefore CRAW cannot handle multiple membership change and lacks efficient multiple authentication mechanism. If the main server is out of service members cannot be recognized in the new area hence high possibility session disruption. The protocol tries to use area group keys to reduce rekeying overheads however they can easily be compromised. Member participating in multiple sessions cannot handle the high computations overheads.

4.1.5 Key Tree in Mobile Multicast (KTMM):

Authors of [21] propose two Key management protocols suitable for securing group communications in mobile multicast. These include KTMM (Key Tree in Mobile Multicast) and WSMM (Wireless Subgroup in Mobile Multicast) protocols. KTMM is a network dependent protocol which works by matching the key management tree to the cellular wireless network topology. KTMM adopts a decentralized framework which divides the key management tree in to subgroup cells managed by the base station. Each k-node occupied by the BS in the tree becomes the subgroup key of that cell to localize the delivery of rekeying messages. The group key is at the root of the tree which shared by all group members. Hence KTMM uses common TEK approach. When the source sends multicast content, it first encrypts it using common group key known to all group members then all members access the content by decrypting using the same group key. A member *(MN_x)* performing handoff is authenticated by the joining cell and receives the move registration request provided**.** After the handoff, the *MNx* holds the subgroup keys of both the previous cell *Bi* and visited cell *Bj* thus modifying the tree.

KTMM takes the advantage of matching the centralized LKH into decentralized framework to provide scalability and reduce rekey overheads. This enables TKMM as a good choice for high bandwidth applications. However, the protocol suffers from 1-affect-n phenomenon since it uses the common TEK approach. It relies on GM which is a single point of failure. KTMM does not address the issue of rekeying on member mobility however it violates the working operation of LKH tree by modifying its structure on member mobility. A moving member knows the subgroup keys of both the previous and current cells. This is an insecure way to provide forward and backward secrecies. For highly dynamic membership, KTMM is not suitable because for a departing member who has roamed many wireless regions, the leaving latency increases due to many subgroup keys that need to be updated hence degrading the overall performance. For multiple memberships, KTMM will not meet the operational efficiency needs because it will induce more signalling messages that can overwhelm the GM and cause storage complexity on resource constraints mobile devices. This protocol does not address an efficient way of authenticating members before they obtain the subgroup keys in the visited cell.

4.1.6 (WSMM):

This group key management protocol is also network dependent which relies on the underlying cellular wireless network topology. It matches the key management tree (LKH) in to a decentralized cellular wireless and the framework is divided into two parts; the wireless and wired part. The wireless part consists of subgroups each with a base station which generates a common subgroup key *(SubKey)* and manages its own rekeying in its subgroup on dynamic membership and on member handoff. In the wired network, LKH scheme is employed by treating the base station as a multicast member wherein the GM generates the group key *(wKey)* and shares it with BSs. Dynamic membership request for mobile hosts are handled by the GM, but not the position of mobile host. Both the wired and wireless parts of the network are managed independently in WSMM. The BS decrypts the multicast content received from the source and encrypts it with wired group key then multicasts the content. Alternative BSs with group members receives the content and decrypts it using wired group key. BS encrypts the multicast content using its wireless subgroup key. Therefore two re-encryptions are needed in multicast data transmission in WSMM. Upon handover of member *(MNx)* from subgroup *Bi* to subgroup *Bj* the joining BS authenticates the joining member *MNx* and creates its individual key *IKjx* and shares it with *MNx. Bj* shares its updated wireless subgroup key *($wsubKey_j$)* with *MNx* encrypted with its individual key *(IKjx).* If *T0* denotes the time when *MNx* joined the multicast group *G*, and if *T1* denotes the time when Bj recently updates the *wsubKeyj*, then rekeying is required if *T0< T1*. Therefore, after a total handoff of *MNx Bi* creates a new *subKeyi* to ensure forward secrecy, encrypts it with old *subKeyi* then multicasts it to existing members *MNs* under *Bi*. WSMM offers a scalable decentralized framework with independent subgroup keys to localize rekeying which is only limited to the affected subgroup when membership changes. It also provides good choice for dynamic membership by providing forward and backward secrecies. However, BSs cannot be trusted by the content provider to open the multicast content. Authors in [9] propose Dual Encryption protocol to resolve the trustworthy issue. Data transformations at the BSs also induce unacceptable delays on real time packet delivery for the moving member hence causing unnecessary session disruptions. WSMM suffers the same problem as TKMM which is the 1-affect-n phenomenon if

the wired group key is changed. Membership requests requires the involvement of the GM which is a single point of failure, this become a performance hurdle on the GM when multiple memberships occur.

4.1.7 Multicast Key Management scheme (MKMS):

The authors of [23] proposed two multicast key management schemes for secure group communications based on LMA based and MAG based multicast schemes [24] in Proxy Mobile IPv6 (PMIPv6) networks [25]. The schemes operate by modifying the LKH scheme in order to satisfy the forward and backward secrecy requirements and reduce the 1-affects-n problem with lower communication costs. PMIPv6 is a network based mobility management protocol introduced by IETF NETLMM working group to reduce the handover latency caused by host based mobility management schemes [7] [8] and [26]. PMIPv6 network supports mobile multicast whereby a mobile node (MN) move from one MAG to another MAG under the same localized mobility domain (LMD) without changing its IP Address. In MAG-based method the MN joins the multicast group directly and receives multicast content through the current MAG without passing through the LMA hence providing lower end to end transmission delay compared to LMA based. While in LMA-based method the MN joins the multicast group through the LMA and receives the multicast content via the MAG-LMA tunnel. The MN does not rejoin the group as it moves around the same LMD because it joins the multicast group through the LMA hence providing lower join and handoff delay. However, the LMA-based method is suitable for high speed environment and the MAG-based method is suitable for stable environment. The decentralized PMIPv6 framework consists of the following network entities: Mobile Access Gateways *(MAGs)* which perform mobility signalling with the LMA on behalf of the MN and detect the movement of the MN, Local Mobility Anchor *(LMA)* which operates similar to the home agent (HA) in MIPv6 [8]. It maintains the binding cache entries for currently registered MNs, Authentication, Authorization, and Accounting *(AAA)* server which authenticates the MN, Key Distributor Centre *(KDC)* which generates, distributes and update the group key and the Service provider *(SP)* which obtains the group key from the KDC and encrypt the multicast content before delivering the encrypted multicast content to the MNs either via MAG-based or LMA-based multicast methods. The authors consider handoff operation on L-MKMS rather than in M-MKMS because when MN moves between MAGs under the same LMD it does not rejoin the multicast group as compared to M-MKMS hence lower join and handoff delay. Moreover, in the L-MKMS method, the LMA acts as a key node and a member node at the same time. The SP encrypts the multicast content with the GK and sends it to the LMA, which decrypts data with its GK. The LMA then encrypts the data with Domain-GK, and multicasts it to all members via the MAG-LMA tunnel. Finally, the recipients use the Domain-GK to decrypt and obtain the data. The strength of the key management is that it tries to mitigate the "one affects all" problem in the LKH scheme with lower communication cost. However the protocol itself does not adjust itself to the network environment and the characteristics of the user. It also lacks trustworthy relationship due to decryption and encryption processes which can expose the multicast content to eavesdropping opportunities. The data transformations may also lead to unacceptable delays which are not tolerable by delay sensitive applications to packet delivery. Authentication of the MN relies on the previous untrusted MAG which may be out of service hence restricting the mobile node from accessing the session on the next MAG. If multiple members move, the previous MAG cannot handle such multiple authentication requests hence delaying packet delivery to the mobile members. The protocol does not address rekeying in old and new area on member mobility which does not satisfy backward and forward secrecies. If more members move, the protocol will suffer from tunnel convergence problem.

4.2 Cluster-Based Schemes

This section lists all the cluster-based mobile multicast group key management protocols for wireless networks which are as follows:

4.2.1 Micro-Grouped Iolus Scheme:

The authors of [27] proposed an improved version of the scheme proposed in [28] which supports member mobility known as micro-grouped Iolus (M-Iolus) which further divides subgroups into micro-groups. M-Iolus adopts a decentralized approach similar to Iolus with independent TEK per subgroup. The network entities involved include the central Group Security Controller (GSC) which manages all the trusted Group Security Intermediaries (GSI) linked to it and the GSI which manages the key management of members within its subgroup. The protocol reduces overhead by introducing the concept of time-stamp association in each sub-group and micro-group key update maintained by each GSI for its group. When MNx moves form micro-group 1 to micro group 2 under the same GSI, GSI detects the area that the MNx has moved in and then the moving member notifies the GSI by sending a move request. The GSI in this case does not need to change its subgroup key K_{SGRP} because MNx is still under its control. GSI records the change of location from micro group 1 to micro-group 2 and timestamps the change. GSI eventually send the micro-group 2 key $K_{2\text{-}MICRO}$ to the moved member MNx, encrypted by K_{MBR} of MNx in order to receive the control information messages in the future. M-Iolus supports member mobility with reduced control overheads because the author proposes null rekeying on host mobility. The scheme reduces the effect of 1-affect-n phenomenon and controls mobile nodes. Null rekeying cost on member mobility is at the expense of raising a forward and backward secrecy violation. Moreover the protocol does not consider other important aspects such as speed of the mobile host, security association between the previous GSI and target GSI. The scheme increases the number of encryptions along with the storage of multiple keys. Another weakness is the time synchronization problem between network entities which may lead to unnecessary update of keys on member leaves. The protocol also lacks trust relationship due to data transformations which can expose the data to eavesdropping. Multiple members moving will be a performance hurdle to the previous GSI which has to deal with multiple authentication requests. If the previous GSI fails, members moving will face service disruptions. If multiple members who had moved between several GSIs leave the group, this will trigger rekeying in all the affected areas hence adding more control overheads which wastes bandwidth.

4.2.2 DeCleene et al:

The authors of [29] and [30] proposed a hierarchy framework and key distribution algorithms for dynamic environment. The authors focus on how keys and trust relationships are transferred when members move across areas in the hierarchy. Furthermore, they make comparable study of rekeying algorithms involved every time a member moves from area to area. It relies on the central server known as Domain Key Distributor (DKD) at the domain level for Key generation, key updating and key distribution. Each area is managed by their controllers known as the Area Key Distributors (AKD) which operate under the jurisdiction of the DKD. AKDs distribute the group key to members under its area securely by encrypting it with the area

local key held by each AKD. The main security keys involved are the group key (data key) held by the DKD which encrypts the actual multicast traffic before is it sent and the Area local key held by the AKDs to securely send the encrypted traffic to its members. Figure 2 illustrate the several rekeying algorithms proposed to minimize the need of rekeying in decentralized framework.

In Static Rekey (SR), shown in Figure 2(a), the AKD maintains the keys unchanged when member *MNx* moves from area *i* to area *j*. Thus, movement of a member triggers null re-keying either in previous area or the new one. The AKDi of area *i* still become responsible for delivering its service when *MNx* moves to area *j* without any registration with the new local *AKDj*. Thus, mobility of MNx affects neither the previous area rekeying nor the new area re-keying. Baseline Rekeying (BR), shown in Figure 2(b), is a direct approach to address mobility whereby location change of a member from one area to another is treated as a leave from the previous area followed by a join to the new area. For this purpose, both the group key and the area key are updated twice in both previous area and new one to ensure forward and backward secrecy respectively. However BR suffers service disruptions in both old and new area during rekeying process. Moreover BR is not able to differentiate between each join/leave of a member from its movement across the areas. Immediate Rekeying (IR), shown in Figure 2(c) improves the baseline algorithm by adding explicit semantics for a hand-off mechanism between areas. When a member moves from area i to area j, it synchronously sends a signalling message to both areas. Both the AKDs concurrently update their area keys while the group key remains unchanged. With IR, unlike the baseline algorithm, there is no need to generate/update the group key and the data transmission continues undisrupted. However it suffers high re-keying overhead especially if member moves rapidly across the areas because this requires repeated local rekeying. Delayed Rekeying (DR), as shown in Figure 2(d), the local rekeying is delayed until a particular criterion is satisfied. Members moving between several areas may collect multiple area keys and reuse these keys when they return to a previously visited area. The idea is based on the probability of returning the moving members to their previous area soon. Thus, the area re-keying in old areas is delayed for a threshold. The old area key remains unchanged for reuse because that member may return promptly. The drawback of this approach is that backward secrecy is not assured till the next join/leave. Periodic rekeying (PR) approach requires update of the area key to done periodically irrespective of member mobility. By periodically re-keying, no area key remains valid for more than a fixed pre-defined period of time. This proposal can be used in conjunction with other inter-area re-keying algorithms like First Entry Delayed Rekey + Periodic (FEDRP) [30]**.** FEDRP combines Delayed and Periodic re-keying algorithms. When a member moves intra-group from area i to area j, it concurrently transfer two signalling messages to both previous AKDi and target AKDj synchronously. Member MNx is added to EKOL in old area. Whenever period expires or join/leave occurs, EKOL is reset. In this case FEDRP improves the inter-area re-keying significantly by keeping old area keys unchanged. The drawback of this method is that keeping EKOL seems to add additional storage overhead to previous area because it is more probable that a member enters a new area for the first time than is back to previous area. Generally immediate rekeying outperforms delayed rekeying because delayed rekeying must rekey multiple areas when the member departs rather than a single area. Since the data transmission cannot resume until all the visited areas have been rekeyed and a new data key distributed, the delayed algorithm remains off-line longer than the immediate algorithm. Logical tree-based intra-area algorithms can help reduce the time required to each area and thereby reduce the difference between the two algorithms. For highly mobile users who remain in the session for a reasonable duration of time, delayed rekeying can provide significant improvements by reducing the number of keys generated and distributed. However, FEDRP seems to be the best rekeying algorithm to reduce communication overheads, provide scalability, low communication overheads and support highly dynamic membership however the area keys maybe compromised easily. If common TEK approach is used, the key management protocol will suffer 1-affect-n phenomenon which will affect the performance of the rekeying algorithms proposed when multiple membership changes occur. The authors here do not address the authentication of users and the speed of the moving user which can be part of the delay factor. Trust relationship between the communicating entities is not addressed. This protocol relies on the single DKD which is a single point of failure. FEDRP can suffer time synchronization problem which can trigger unnecessary rekeying.

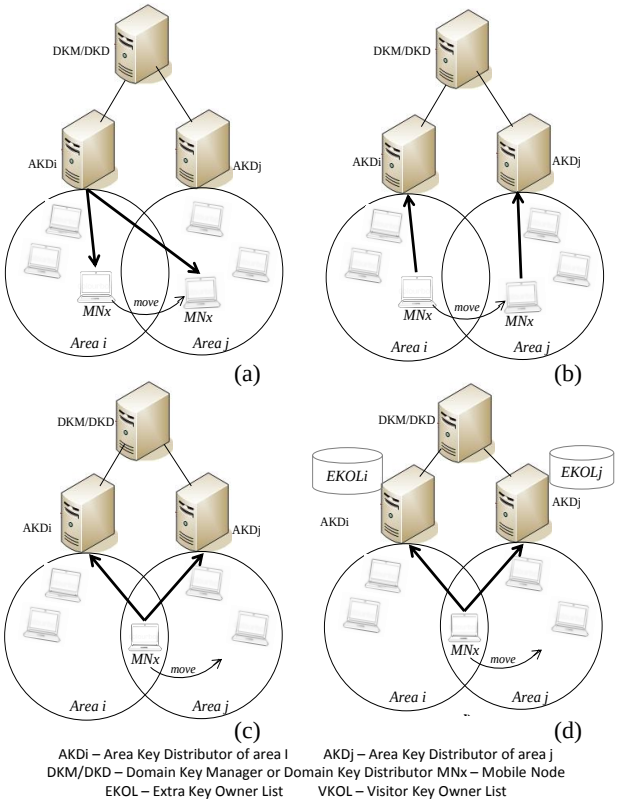

Figure 2: Rekeying algorithms

4.2.3 Kellil et al:

Authors of [31] propose area re-keying algorithm to address member mobility in mobile multicast communication. The specific key called Visitor Encryption Key (VEK) is introduced for moving members. The scheme adopts a decentralized framework similar to cellular wireless topology. The entities involved include *Domain Group Controller Key Server (Domain GCKS)* which generates and distribute the Traffic Encryption Keys (TEK) to a number of local Group Controller Key Servers (local GCKS). *Local GCKS* for forwarding the TEK protected under Key Encryption Keys (KEKi, KEKj) specific to each area managed by GCKSs (GCKSi, GCKSj) respectively to its group members. Each GCKS maintains lists called Extra Key Owner List (EKOL) and Visitor Key Owner List (VKOL). When moving member Mi moves from area i to area j, it synchronously sends two signalling messages to both previous GCKSi area and

new GCKSj to inform them about its mobility (similar to IR and FEDRP). The target GCKSj sends its local VEKj to Mi via a secure channel. VEKj acts like local area key within the area j. The area controllers maintain two kinds of owner lists: EKOL for stationery member's records as well as a Visitor-Key Owner List, VKOL, for mobile member's records. VKOL maintain a list of members holding a valid VEK but have left the area of that VEK. EKOL maintains a list of members holding the KEK (local area key). This method separates re-keying of mobile members from non-mobile members. Therefore, no area re-keying is needed when a mobile group member moves between two areas. Both backward and forward secrecies are assured along the mobility between areas. The rekeying algorithm triggers null local rekeying (KEKJ or VEKJ) when a member moves between two areas even though both backward and forward secrecies are ensured. However, it separates rekey due to membership dynamism from rekeying due member mobility which is not a reasonable way to reduce rekey overheads. The mechanism also suffers from 1-affects-n problem since it falls in the category of common TEK approach. This means that each join/leave to the group causes re-keying for visitor members as well as area non-mobile members. The mechanism cannot handle highly dynamic and highly mobile members due to multiple rekey requests. Introduction of the extra VKOL actually add storage overhead to AKDs which can induce processing delays for highly dynamic membership.

4.2.4 Group Key Management Framework (GKMF):

The authors of [32] [33] and [34] propose a network dependent based group key management protocol which address the issue of member mobility protocol for Secure Group Communication in Wireless Mobile Environments. The protocol also introduces the use of lists to simplify mobility management of members. Provisions of other security services (such as data integrity and message authentication) are implicitly assumed in GKMF. In GKMF, the nodes of a network are physically or logically classified to main entities and placement of entities, so that all nodes are divided to two levels: domain level and area level. GKMF consider a decentralized framework similar to [29] which relies on cellular wireless network with common TEK approach. The entities involved are the DKM (Domain key Manager), the main key manager of domain i which generates, distribute, store, and delete all the key material that may be required at the domain level and the AKM (Area Key Manager), the key manager of the area j inside a domain which perform key management in its area and manages group members residing in its area. Both the entities maintain a list so-called MobList to keep track of mobile members. Every time there is a member handoff, the protocol records the information such as the IDs of the moving member, multicast group G joined by the member, area that a member is moving from, ID of the target area that a member is moving to. The GKMF relies on the MobList to keep track of mobile members and can easily determine whether rekeying is necessary in the visited areas. In GKMF the trustworthy relationship between the communicating entities is ensured by introducing secure association at different levels using shared symmetric keys which are fast to process. However the biggest drawback of this protocol is the large number of used keys which result in to storage complexity at the resource constraint mobile devices. The protocol does not address how leave re-keying is performed in the previous area since there is no re-keying while moving hence forward secrecy is violated. The use of common TEK gives rise to the 1-affect-n phenomenon. A member moving with backward secrecy can suffer join latencies due to rekey of both area key and TEK which are done independently. A member leaving after rapidly moving between the areas requires update of both the area key and the traffic in all the areas that a member had visited.

4.2.5 Gharout et al:

The authors [35] propose new key management protocol considering node mobility which secures group communications in mobile environment with a null rekeying cost. The protocol follows the concept of independent TEK per subgroup to prevent 1-affect-n phenomenon. The entities involved are the Domain Key Distributors (DKD) which manages all the Area Key Distributors (AKDs) under it and AKDs which limit key management procedure only to its area. Each cluster is controlled by one DKD at the Domain level and at least one AKD at the area level. AKDs belonging to the same DKD use common TEK hence no rekeying is required when a member moves from one area to another within the same cluster. The area levels consist of member nodes which are either dynamic or moving from one area to another. Each AKDi maintains two lists: a list of members (denoted by LMi) which contains identities of current area i members; and a list of old members (denoted LOi) which contains identities of previous members. This protocol has an advantage of optimizing rekeying messages by avoiding renewing the TEK when the member intra cluster move across areas belonging to the same DKD. This violates the forward and backward secrecies required by any group key management protocol. It reduces the workload from the DKD by allowing the mobile members to be authenticated by the AKDs. Though it introduces null rekeying intra-move it has a drawback of storage overheads at the member because a moving member participating in multiple sessions needs to store multiple MEKs. The protocol does not address inter cluster movement which may cause data transformation problem same as in Iolus [28]. A moving member can experience join latency due to data transformation processing delays when it inter moves.

4.2.6 Key Management to secure Group Communications in Mobile Environments (KMGM):

The authors of [36] propose an adaptive group key management protocol for wireless communications known as KMGM which introduces mobility support to Adaptive clustering for Scalable key management in dynamic Group communications protocol (ASGK) in [37]. The scheme is improved by introducing both the intra-cluster mobility and inter-cluster area mobility then categorizing the AKDs belonging to the same cluster as passive agents or active agents using the heuristic described in [37]. Verification steps of moving members on the target AKD is performed similar to [35]. KGKM forms a hybrid scheme which takes advantage of both the common TEK and Independent TEK per subgroup approaches. KGKM adopts a decentralized framework whereby a group is partitioned into a hierarchy of administrative areas managed by an Area Key Distributors (AKDs). The internal AKDs of a cluster which are considered passive receive and forward the messages to their respective area members with no data transformation. Both the AKDs maintain two lists LMi and LOi as in Gharout et al to keep track of mobility members. The operation of KMGM is partially based on the same protocol proposed in [35] therefore it inherits some strengths and drawbacks of the protocol. Introducing more micro-groups like in M-Iolus [27] within the cluster add number of encryption areas which may delay the packet delivery to the members.

Table 1 and 2 summarizes if whether the above mentioned requirements are met for the respective group key management protocols in this survey for common TEK and independent TEK per cluster respectively.

Table 1: Comparative study of requirements for common TEK schemes		
Evaluation criterion	**Tree-Based**	**Cluster-Based**
Key Independence	Yes to all	Yes to all
1 affect n phenomenon	Yes except KTMM	Yes to all
Trust relationship	No except TMKM	GKMF only
Local rekeying	Yes except TMKM and HKM	Yes to all
Data transformation	Yes except TMKM and HKM	None
Support multiple membership changes	No except KTMM	None
Support highly dynamic membership	Yes except TMKM and L-MKMS	None
Forward secrecy	Yes except KTMM and L-MKMS	None
Backward secrecy	Yes except KTMM and L-MKMS	Yes to all
Integration of security services (authentication with key management)	Yes except TMKM and HKM	None
Support scalability	Yes to all	Yes to all
Support for heterogeneous wireless networks	None	None
Single point of failure	Yes except WSMM	Yes to all
Fault tolerance	None	None
Considers single authentications at join/move	Yes except TMKM and HKM	GKMF only
Support multiple authentications at join/move	None	None
Address member velocity environments	No except HKM and WSMM	None
Use of list(s) to manage mobility(number of lists)	No except TMKM (1)	Decleene et al (1), Kellil et al (2), GKMF (2)
Number of layers/levels (number)	TMKM (multiple), HKM(multiple), KTMM(2), WSMM (0), L-MKMS (2)	2 for all
Number of keys held by the mobile device	NS for all	Decleene et al, (3), Kellil et al (4), GKMF (4)
Number of keys held by the Domain Key Distributor	NS for all	Decleene et al, (2), Kellil et al (2), GKMF (3)
Number of keys held by the Area Controllers	NS for all	Decleene et al, (3) Kellil et al (3), GKMF (5)
Rekeying Communication cost	TMKM($2\log_2$+1), HKM(NS), KTMM(NS), WSMM(NS), L-MKMS(0)	Decleene et al $(O(n)+1))$, Kellil et al $(O(n+1))$, GKMF $(O(n+1))$
Computation cost	NS for all	NS for all

Table 2: Comparative study of requirements for independent TEK schemes			
Evaluation criterion		**Tree-Based**	**Cluster-Based**
Key Independence		Yes WANG scheme only	Yes to all
1 affect n phenomenon outside the cluster		None	None
1 affect n phenomenon inside the cluster		Yes for all	Yes except M-Iolus
Trust relationship		Yes for WANG scheme only	None
Local rekeying		Yes for all	Yes to all
Data transformation		None	Yes to all
Support multiple membership changes		Yes except CRAW	None
Support highly dynamic membership		Yes except CRAW	No except M-Iolus
Forward secrecy		Yes for all	No except M-Iolus
Backward secrecy		Yes for all	Yes to all
Integration of security services (authentication with key management)		Yes for all	Yes except M-Iolus
Support scalability		Yes for all	Yes to all
Support scalability within the cluster		Yes for all	Yes except Gharout et al
Support for heterogeneous wireless networks		Yes except CRAW	None
Single point of failure		None	No to all
Fault tolerance		Yes except CRAW	Yes to all
Considers single authentications at join/move		Yes for all	Yes to all
Support multiple authentications at join/move		None	None
Address member velocity environments		None	None
Use of list(s) to manage mobility(number of lists)		Yes for all (1)	M-Iolus (1), Gharout et al (2), KGKM (3)
Number of layers/levels		2 for all	2
Storage cost	At the mobile device	CRAW ($\log_2$(n)-1), WANG ($\sum_{i=1}^{j} \log 2(n)+1$)	M-Iolus (3), Gharout et al (4), KGKM (4)
	At the Area Controllers	CRAW $(O(2n-1))$, WANG(not specified)	M-Iolus (nTEK + 1), Gharout et al (2), KGKM (nTEK + 1)
	At the Domain Key distributor	CRAW $(O(\log 2(n)))$, WANG(2n-1)	M-Iolus (3), Gharout et al (5), KGKM (5)
Rekeying Communication cost on member mobility		CRAW (1+$\log_2$(n)), WANG($2\log_2$(n)+2)	M-Iolus (2O(n)), Gharout et al (0), KGKM (0)
Computation cost on member mobility		CRAW (1+$\log_2$(n)), WANG($(2\log 2(n)+1)^2 + (\sum_{1=1}^{nTEK(affected)} i \times n(i) - 1))$	Not specified for all

5. CONCLUSION

Access control in multicast communication is applied by encrypting the multicast content with a shared group key (TEK). The security of multicast communication depends on the safety of the TEK. An efficient group key management is required to control the key generation, key distribution and key updating. This paper has provided a strong review on the existing group key management approaches and a broad classification of them according to network independent based and network dependent based protocols. Network independent approaches which rely on the underlying network infrastructure are further classified in to three subcategories namely the centralized, decentralized and distributed group key management types. Since these protocols were designed for stationery nodes, a review and comparison for the strength and weaknesses when applied to wireless mobile environment is presented. This paper also reviews and compares the existing network dependent protocols which rely on the underlying wireless fixed infrastructure. These are categorized in to tree based and cluster based approaches. Both of them use either the common TEK or Independent TEK approaches to

address mobility in wireless mobile multicast environment. The proposed common TEK solutions suffer from 1 a affect n phenomenon and the Independent TEK approaches suffers from the important number of decryption / encryption operations which need to be addressed. The proposed solutions lack support for highly dynamic membership change with multiple membership changes. The choice of best key management protocol relies on the application needs: reducing rekeying latency or reducing data multicasting latency.

6. REFERENCES

[1] P. Judge and M. Ammar, "Security issues and solutions in multicast content distribution: a survey," *Network, IEEE*, vol. 17, pp. 30-36, 2003.

[2] Y. Challal and H. Seba, "Group Key Management Protocols: A Novel Taxonomy. ," *Enformatika, International Journal of Information technology*, vol. 2, 2005.

[3] J. Bibo and H. Xiulin, "A Survey of Group Key Management," in *Computer Science and Software Engineering, 2008 International Conference on*, 2008, pp. 994-1002.

[4] S. Rafaeli and D. Hutchison, " A Survey of Key Management for Secure Group Communication," *ACM Computing Surveys*, vol. 35, pp. 309–329, September 2003 2003.

[5] I. Romdhani, M. Kellil, L. Hong-Yon, A. Bouabdallah, and H. Bettahar, "IP mobile multicast: Challenges and solutions," *Communications Surveys & Tutorials, IEEE*, vol. 6, pp. 18-41, 2004.

[6] C. Perkins, "RFC3344: IP Mobility Support for IPv4,". IETF RFC. Status: Proposed Standard.,August 2002.

[7] R. Koodli, "RFC5568: Fast Handovers for Mobile IPv6," IETF RFC.Status: Proposed Standard.,July 2009.

[8] D. Johnson, C. Perkins, and J. Arkko, " RFC3775: Mobility Support in IPv6," IETF RFC. Status: Proposed Standard., June 2004.

[9] T. Hardjono and L. R. Dondeti, *Multicast and Group Security*: Artech House, 2003.

[10] S. Yan, W. Trappe, and K. J. R. Liu, "An efficient key management scheme for secure wireless multicast," in *Communications, 2002. ICC 2002. IEEE International Conference on*, 2002, pp. 1236-1240 vol.2.

[11] S. Yan, W. Trappe, and K. J. R. Liu, "Topology-aware key management schemes for wireless multicast," in *Global Telecommunications Conference, 2003. GLOBECOM '03. IEEE*, 2003, pp. 1471-1475 vol.3.

[12] K. Brown and S. Singh, "RelM:Reliable Multicast for Mobile Networks," *Computer Communications*, vol. 21, pp. 1379-1400, 1998.

[13] L. Lin, L. Xueming, and C. Yong, "HKM: A Hybrid Key Management Scheme for Secure Mobile Multicast," in *Networking, Architecture, and Storage, 2007. NAS 2007. International Conference on*, 2007, pp. 109-114.

[14] C. K. Wong, M. Gouda, and S. S. Lam, "Secure Group Communications Using Key Graphs," ed. ACM SIGCOMM, 1998.

[15] D. A. McGrew and A. T. Sherman, " Key Establishement in Large Dynamic Groups using One-way Function Trees," Technical Report TR-0755,May 1998.

[16] W. Yiling, L. Phu Dung, and B. Srinivasan, "Hybrid Group Key Management Scheme for Secure Wireless Multicast," in *Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on*, 2007, pp. 346-351.

[17] W. Yiling, L. Phu Dung, and B. Srinivasan, "Efficient Key Management for Secure Wireless Multicast," in *Convergence and Hybrid Information Technology, 2008. ICCIT '08. Third International Conference on*, 2008, pp. 1131-1136.

[18] E. Eidkhani, M. Hajyvahabzadeh, S. A. Mortazav, and A. N. Pour, "CRAW: Combination of Re-Keying and Authentication in Wireless Networks for Secure Multicast Increasing Efficiency of Member Join/Leave and Movement," *International Journal of Computer Networks & Communications (IJCNC)*, vol. 4, pp. 107-128, 2012.

[19] M. Sandirigama, S. Akihiro, and M. Noda, "Simple And Secure password authentication protocol," *IEICE Trans. Com*, vol. E83-B, pp. 1363-1365 2000.

[20] M. Hajyvahabzadeh, E. Eidkhani, S. A. Mortazavi, and A. N. Pour, "A New Group Key Management Protocol Using Code for Key Calculation: CKC," in *Information Science and Applications (ICISA), 2010 International Conference on*, 2010, pp. 1-6.

[21] R. Jong-Hyuk and L. Kyoon-Ha, "Key management scheme for providing the confidentiality in mobile multicast," in *Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference*, 2006, pp. 5 pp.-1209.

[22] L. R. Dondeti, S. Mukherjee, and A. Samal, "Scalable secure one-to-many group communication using dual encryption," *Computer Communications*, vol. 23, pp. 1681-1701, 11/1/ 2000.

[23] C. Ming-Chin and L. Jeng-Farn, "MKMS: Multicast key management scheme for Proxy Mobile IPv6 networks," in *Consumer Electronics, Communications and Networks (CECNet), 2011 International Conference on*, 2011, pp. 1402-1405.

[24] G. Jianfeng, Z. Huachun, Z. Hongke, and H. Luo, "Multicast Extension Support for Proxy MIPv6," in *Consumer Communications and Networking Conference (CCNC), 2010 7th IEEE*, 2010, pp. 1-5.

[25] S.Gundavelli, K. Leung, V. Devarapalli, K.Chowdhury, and B.Patil, "Proxy Mobile IPv6," *RFC 5213*, August 2008.

[26] H. Soliman, C. Castelluccia, K. ElMalki, and L.Bellier, "Hierarchical Mobile IPv6 (HMIPv6) Mobility Management," *RFC 5380*, October 2008.

[27] S. Kamat, S. Parimi, and D. P. Agrawal, "Reduction in control overhead for a secure, scalable framework for mobile multicast," in *Communications, 2003. ICC '03. IEEE International Conference on*, 2003, pp. 98-103 vol.1.

[28] S. Mittra, "Iolus: a framework for scalable secure multicasting," *SIGCOMM Comput. Commun. Rev.*, vol. 27, pp. 277-288, 1997.

[29] B. DeCleene, L. Dondeti, S. Griffin, T. Hardjono, D. Kiwior, J. Kurose, *et al.*, "Secure group communications for wireless networks," in *Military Communications Conference, 2001. MILCOM 2001. Communications for*

Network-Centric Operations: Creating the Information Force. IEEE, 2001, pp. 113-117 vol.1.

[30] C. Zhang, B. DeCleene, J. Kurose, and D. Towsley, "Comparison of inter-area rekeying algorithms for secure wireless group communications," *Performance Evaluation*, vol. 49, pp. 1-20, 9// 2002.

[31] M. Kellil, Olivereau, J. C. A., and P. Janneteau, "Rekeying in secure mobile multicast communications," *United States Patent Application Publications*, US 2007/ 0143600 A1 2007.

[32] L. M. Kiah and K. M. Martin, "Host Mobility Protocol for Secure Group Communication in Wireless Mobile Environments," in *Future Generation Communication and Networking (FGCN 2007)*, 2007, pp. 100-107.

[33] M. L. M. Kiah and K. M. Martin., "Host Mobility Protocol for Secure Group Communication in Wireless Mobile Environments," *International Journal of Security and its Applications*, vol. 2, pp. 39-52, January 2008.

[34] M. L. M. Kiah and B. Daghighi, "An Implementation of Secure Group Communication in a Wireless Environment," *International Journal of Computer and Electrical Engineering*, vol. 4, December 2012.

[35] S. Gharout, A. Bouabdallah, M. Kellil, and Y. Challal, "Key management with host mobility in dynamic groups," presented at the Proceedings of the 3rd international conference on Security of information and networks, Taganrog, Rostov-on-Don, Russian Federation, 2010.

[36] S. Gharout, A. Bouabdallah, Y. Challal, and M. Achemlal, "Adaptive Group Key Management Protocol for Wireless Communications," *j-jucs*, vol. 18, pp. 874--898, May 2012.

[37] W. Chung Kei, M. Gouda, and S. S. Lam, "Secure group communications using key graphs," Networking, IEEE/ACM Transactions on, vol. 8, pp. 16-30, 2000.