International Journal of Innovative Research in Advanced Engineering (IJIRAE) ISSN: 2349-2163
Issue 11, Volume 2 (November 2015) www.ijirae.com

A Generic Group Key Management Framework model for wireless Mobile Environments

Umi Salma B.
Research Scholar / Computer Science
Bharathiar University ,Coimbatore

Dr. A.Arul Lawrence Selvakumar
Dean & Professor / CSE
RGIT, Bangalore

Abstract – In wireless environments with the growing usage of mobile devices and the advent of multicast communications, are widely researched and the development of secure multicast is pertinent in wired networks compared to implementation in mobile environments. Key management is widely being adopted in securing group communication for both wired and wireless networks. Wireless networks bring additional challenges due to member mobility and increase in the number of member's nature of mobile devices. This paper presents a generic design issues and GKMF model in wireless mobile environments with protocols. 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: Wireless Network, Group key management, Security, Re-keying

1. Group Key Management framework (GKMF): Design Challenges in Wireless Mobile Environments

1.1 Types of Wireless Mobile Environments

While issues concerning group communication are widely researched and the development of secure multicast becoming more pertinent in wired networks, its implementation in mobile environments (wireless networks) is still in its infancy. Applications and services which are available in wired networks should also be made available in wireless networks and vice versa. There are similar expectations for providing secure and reliable communication in both environments. Main types of WMobE can be categorized as:

(a) ***Fixed-based networks***

A collection or a group of wireless mobile nodes communicating or using the services usually provided by corporate enterprise networks or small office home offices (SOHO) over wireless mediums such as wireless local area networks (WLANs), or cellular-based networks such as GSM or UMTS (Lin and Chlamtac, 2001), (Hillebrand, 2002). This kind of group communications operates with the help of fixed infrastructures such as base stations, access points or satellites.

(b) ***Non Fixed-based networks***

Operating without the help of any infrastructure, non fixed-based networks can be further classified into:

• *Adhoc networks*. A collection of wireless mobile nodes communicating among themselves, possibly over multi-hop paths, without the help of any infrastructure such as base stations or access points (IIyas, 2003), (Michiardi and Molva, 2006).

• *Sensor networks*. A special form of adhoc networks, sensor networks consist of a collection of individual nodes (usually battery operated), each of which transmits data signals also without the help of any infrastructure. Typical use of an individual sensor is for collecting specific data such as sudden changes in climate across a geographical area (Wadaa et al., 2004). We note that the WMobEs concerned with our work rely on fixed network infrastructures where reliable entities such as domain key managers and area key managers are assumed available. We will not consider ad-hoc or sensor networks in this Journal.

1.2.1 Non-fixed and Wireless Network Connectivity In comparison to fixed wired networks, wireless mobile networks have some very different security issues:

• *More susceptible to security attacks*. Compared to wired data transmissions, wireless data transmission is more susceptible to attacks, particularly passive attacks such as eavesdropping and monitoring of data traffic. These attacks are more acute in WMobEs since they are easier to carry out in such environments because an adversary does not have to physically tap into a network (Vines, 2002) and (Nichols and Lekkas, 2002). This could result in further security breaches if certain measures are not in place. For example, messages or group data that are not encrypted can easily be read and understood by an adversary who is eavesdropping on the group data traffic.

• *Trust within foreign networks*. As mobile environments allow members to move around, changes in location may require group members to occasionally communicate via foreign networks that cannot always be trusted.

© 2014- 15, IJIRAE- All Rights Reserved

This affects the amount of trust to impart on governing entity(s) within foreign networks (including how much information they want to share). Likewise, foreign networks may want to consider the amount of trust they want to place on the visiting mobile hosts, since they may gather information about the local security services for the areas they visit, which can lead to security threats if not contained.

• *Susceptible for disconnections*. Compared to wired networks, transmissions in wireless networks have high tendencies for disconnection. Host mobility in such environments aggravates this issue further. Frequent disconnections during data transmission may cause unnecessary disruptions to group communication. In this case, minimal delay during key management processes may be useful in such environments.

1.2.2 Nature of Mobile Devices

Many mobile devices such as mini PDAs or smart phones exhibit special characteristics with regards to having limited power consumption (or energy supply), limited data (key) storage as well as having lower processing speed, which hinder intensive and heavyweight processing. With such constraints, the design of a GKMF should employ:

• As little cryptographic computation as possible.
• Minimal messaging bandwidth usage (in other words, minimum message exchange between communicating entities).
• Efficient storage (normally associated with minimizing numbers of keys that a mobile device needs to store).

1.2.3 User (Host) Mobility

Wireless mobile environments allow user (host) mobility. As group members are allowed to move between areas, the mobility issue exhibits problems that do not occur in wired networks such as:

• *Hand-off operations*. When group members move from one area to another, some kind of hand-off operation from the current area to the visited area (the area where the member is moving to) is required.
• *Management of keying material*. Problems pertaining to management of cryptographic keys needed during host mobility include deciding who governs the moves and who keeps track of keying material.
• *Network disconnection*. As discussed in *Section 1.2.1*, network transmissions between communicating entities may change over time, and be prone to failure (disconnection).

The issues discussed so far are, to an extent, the common security problems associated with wireless mobile networks. The implementation of multicast group communications in such environments introduces further issues that need to be addressed by a GKMF. We look at these issues in the next section.

1.3 Issues Specific to Group Communications in WMobEs

In this section we present specific security issues concerning group communications for deployment in wireless mobile networks.

1.3.1 New Reasons for Joins and Leaves

Group members no longer just join and leave a multicast group, but are also allowed to join and leave as they move between areas while still remaining in a group session. A specific protocol is required to govern members' movement between areas, which is also referred to as *host mobility*. The process of a member moving to other areas may be treated as a *leave* from one area followed by a *join* to another. Any member who wishes to move will need to notify a key manager, prior to moving. Similarly to a member joining and a member leaving a member moving to another area may require the provision of backward and/or forward secrecy (since different areas may have their own security requirements). In this case, controlling access to visited areas may be necessary. In dynamic environments, group members may be allowed to freely move between areas while still remaining in a group session.

1.3.2 Additional Key Management to Support Mobility

The generation of new keying material may be required in order to support host mobility. For example:

(a) Moving members may still hold cryptographic keys of the areas they visited even after they leave a multicast group, which may lead to compromise.

(b) Host mobility may require group members to occasionally communicate via a foreign network (the visiting area) that may not be fully trusted. Thus, it is important to ensure that group members that are moving from one area to another are protected (via different sets of keys).

(c) Group members that move between areas may gather the area's local security information. It is imperative to ensure that the area is protected from members who are moving from one area to another in order to collect the security information (keys) of each area for malicious purposes.

1.3.3 Performance Requirements

Security and key management requirements apply for designing a GKMF for WMobEs. In this section, we discuss the performance requirements of a GKMF in a WMobE. The performance requirements can be divided into several categories as follows: Computation, energy and storage requirements. All computations related to group security should be resource efficient, since mobile nodes are usually equipped with limited power and storage, which restricts their ability to do rigorous computations and to store a large amount of data. Therefore, any security-related operations to be done by mobile devices are to be kept as low as possible. This also includes the number of keys that each member needs to store throughout a multicast group session.

© 2014- 15, IJIRAE- All Rights Reserved

• Communication and bandwidth requirements. In WMobEs, since frequent connection cut-offs are likely to happen and the bandwidth available is limited compared to wired environments, the number of messages exchanged (such as between key manager entities sending and/or receiving keys and other security related information) should be kept as low as possible. As battery life is also a big problem in many mobile devices, less messages that need to be processed and sent may be necessary since more messages will drain batteries life.
• Mobility requirement. All security-related information associated with each member during his mobility, including the distribution of new keys or key updates, needs to be managed in a proper manner so that the performance (quality of service) of the framework should not be significantly reduced by host mobility.

2. A Generic Group Key Management Framework Model for Wireless Mobile Environments

2.1 Main Components

In this section we describe the main functional components that a GKMF should have for secure multicast group communication. This includes the general architecture of typical entities that form a GKMF, their relationships and cryptographic keys used. It also includes other aspects of the framework such as options for assigning key managers, and group membership policy.

2.1.1 General Architecture

The main components that comprise architecture can be listed as follows:
(a) Main entities
The main entities identified within a framework for group key management are typically:
• *Server(s).* Typical roles of servers are as key managers, group managers, group controllers, or as supporting nodes for multicast group communication.
• *Hosts.* Hosts or group members are the lowest level entities within a GKMF. Also referred to as *key users*, group members engage in the actual group communication. Group members consist of at least one *sender* of data, and one (or more) *recipients* of data. *Figure 2.1* illustrates a basic model for group key management with a central entity. From the figure, a key server (which can also be referred to as group manager) is the central managing entity for group members; one as a sender and another as a recipient. Both members need to establish security parameters prior to group communication, and this is done through key and security association (SA) management (denoted by ←→ arrows). Any subsequent communications such as key updates that need to occur are done through a control channel between the key server and the group members (denoted by *dashed* arrows). The actual group communication among the group members is denoted by a bold −→ arrow.

(b) Domain(s) and Area(s)
Domain(s) and area(s) in wireless mobile environments (WMobEs) can be described, as follows:
• *Domains.* Logically, a domain can be viewed as a bigger system which comprises a group of subsystems, which can consist of hundreds or thousands of services. From another perspective, a domain can cover a physical large geographic area (such as a UK region), which consists of smaller areas (such as counties). Each domain may have its own purpose, common goals and objectives.

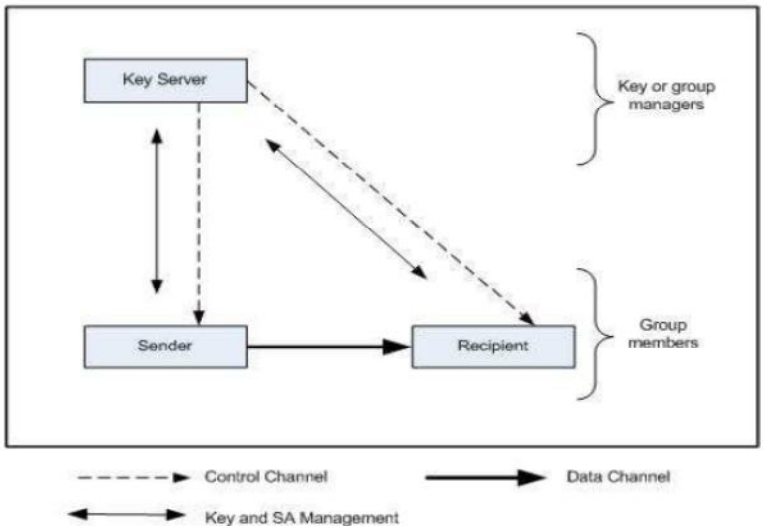


Figure 2.1: A basic model for group key management with central entity.

Areas. Smaller versions of domains, areas can be viewed logically as subsystems which operate under the governance of bigger systems (such as *domain* system). Similarly, an area can cover a physical geographic area (such as one or more counties) which can be part of larger system (such as a region).

© 2014- 15, IJIRAE- All Rights Reserved

While all areas in a domain may share the domain's general goals and objectives, each area may have its own unique requirements. Typically for the purpose of key management, in a wireless mobile network (such as GSM network), a domain can be one large physical area which may be governed by physical entity (such as a group controller or a key manager). For efficient key management, the domain can be further divided into smaller manageable physical areas, each of which may have its own governing entity.

WMobEs allow group members to move from one place to another while still participating in a group session. For ease of managing host mobility in such environments, group members are typically placed in these manageable areas. As WMobEs can consist of multiple domains and areas that can be overlapping logically and physically, we use *intra-domain* to describe relationships within a domain, and *inter-domain* to describe relationships between two
(Or more) distinctive domains. An example of domains and areas is depicted in *Figure 2.2*. Domain DA and DB each consists of several areas. The *inter* and *intra* domain relationships are shown between and within the domains.

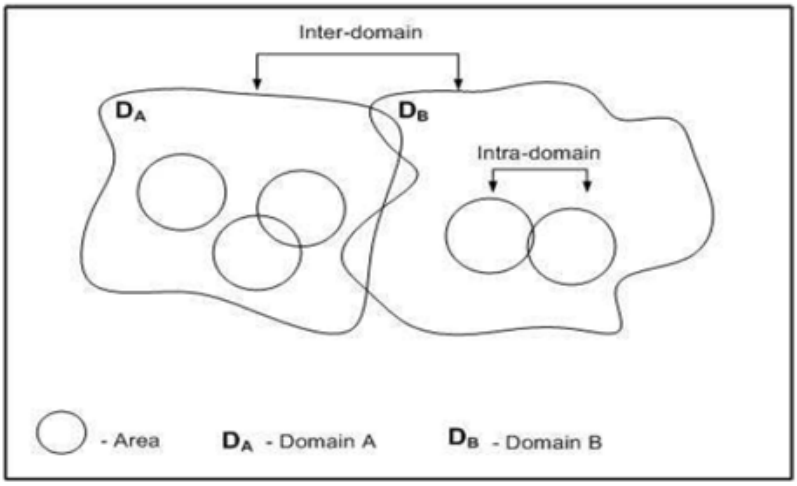


Figure 2.2: An example of domains and areas.

(c) Placement of entities
The entities in the framework identified in (a) must be placed in the domains and areas identified in (b). As discussed in (b), a domain can be further divided into smaller (or multiple) areas (see *Figure 2.2*). Thus, these areas need to be placed within a domain and group members within these areas. Domain(s) and areas may have their own managing entity. Typical placement of entities based on a centralized framework is illustrated in *Figure 2.3*. Here, domain j is depicted to consist of at least two independent areas, which are *Area a* and *Area b*, with one managing entity at the domain level referred to as the *domain entity*. Similarly, at the area level, each area consists of one managing entity referred to as an *area entity*. Group members M can be positioned in any of the areas in domain j, and each area entity is responsible for group members residing in its area.

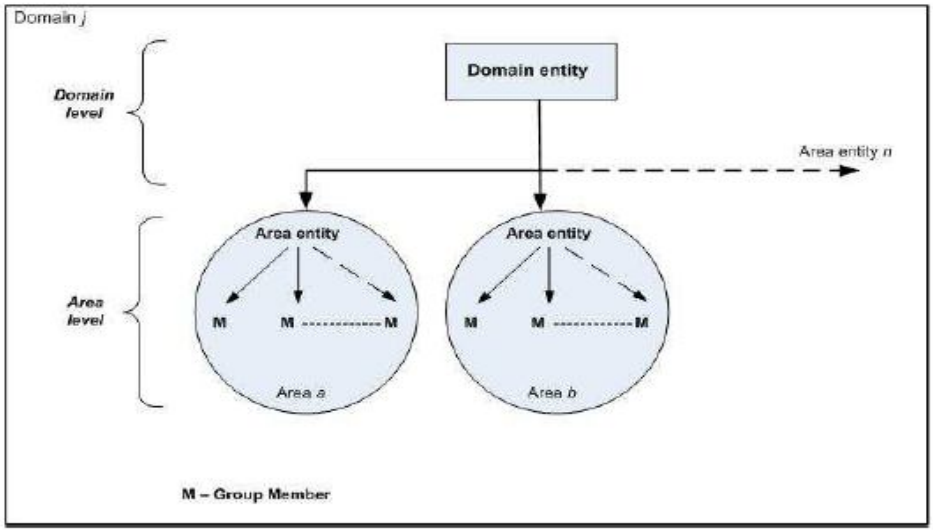


Figure 2.3: An example of placement of entities in a domain j.

© 2014- 15, IJIRAE- All Rights Reserved

(d) Trust Relationships

Trust relationships between the entities need to be specified. Typically, trust relationships within a GKMF revolve around key managers (as the main key distributors). Key managers are often configured and maintained by human administration and implemented using secure technology, since they represent the best point of attack. All entities involved in multicast group communication in a centralized scheme trust the centralized key manager in the domain, which is the primary source of security parameters (such as cryptographic keys) needed for group communication. When the need arises, this entity can also take on the role of a certification authority. On the other hand, distributed schemes require having multiple entities acting as key managers, who are jointly trusted for the generation and distribution of the security parameters needed for the group communication.

(e) Types of Key

The types of key used in the framework need to be specified, cryptographic keys can be symmetric, asymmetric or a combination of both. Where symmetric keys are used, they can typically be classified into:

• *Traffic keys*. Primarily used for securing actual data communications (also referred to as *traffic encryption keys* or *traffic protection keys*), and shared amongst all group members of a particular multicast group. Typically, a traffic key is unique for a multicast group.
• *Domain keys*. These keys are typically shared and used by key managers for secure distribution of *traffic keys* within a domain (also referred to as *group keys* or *key encryption keys*). In WMobEs where multiple domains may exist, every domain should have its own domain key.
• *Area keys*. These keys are typically unique to an area and shared between key managers and group members residing in the area (also referred to as *area control keys*). Used for secure distribution of keys by an area key manager to members within an area. They also form auxiliary keys (at the area level) which may be used by members to send secure messages amongst one another within that particular area.
• *Individual keys*. These keys are long-term secrets between key managers and group members. These keys typically need to be established prior to any request to create multicast groups. In cases where asymmetric keys are used, all entities are normally assigned asymmetric key pairs. Each member is assumed to have a copy of the public keys of relevant key managers. Key managers are also assumed to have copies of public keys of any other key managers they need to communicate with.

2.1.2 Group Membership Policy

Group membership can be categorized as:
(a) *Static or closed membership;*
(b) *Dynamic or open membership.*
We normally assume that dynamic membership is supported in any GKMF for WMobEs.

2.1.3 Design Approach

(a) *Centralized scheme,*
(b) *Distributed scheme, or*
(c) *Hybrid scheme.*

2.2 Main Protocols

In this section we describe the main protocols needed for designing a GKMF for deployment in a WMobE. While many of these are generic GKMF protocols, we also need a protocol for *host mobility* to govern the movement of group members between areas throughout their membership of a multicast group.

2.2.1 Protocol for Creating New Group

This protocol governs the creation of new multicast groups. It also includes the initial registration of group members to a multicast group and initial distribution of keys to that particular member. Note that, typically, any host who requests the creation of a multicast group is considered to be the first host to register (or join) the group. At this point, the information related to group membership policy, as well as the relevant keys (see *Section 2.1*) necessary for group communication, are determined.

2.2.2 Protocol for New Member Joining

This process is similar to the initial registration of group members. In particular, this protocol governs new joins of group members into a multicast group. It also includes the distribution of all cryptographic keys needed for the group communication to the newly joined members. Different membership policy affects the way we manage the cryptographic keys. Thus, this protocol introduces two options which can be adopted in admitting any hosts to become group members of a multicast group:

(a) *Members joining with backward secrecy*

© 2014- 15, IJIRAE- All Rights Reserved

In the first option, typically when a newly joined member is to be prevented from accessing previous group traffic or old group keys, all cryptographic keys associated with the multicast group, including the area where the group member is residing, need to be re-keyed.

(b) *Members joining without backward secrecy*

In the second option, no such restriction applies, and re-keying may not need to occur. New group members are given the same set of keys that are currently used by existing members.

2.2.3 Protocol for Existing Member Leaving

This protocol governs existing members leaving a multicast group, members leaving can be:

(a) *Voluntary, or*

(b) *Non-voluntary.*

Like members joining, two options can be adopted due to members leaving:

(a) *Members leaving with forward secrecy*

In the first option, if controlling access to future group communication is necessary then re-keying needs to occur whenever an existing member leaves a multicast group. All relevant keys will need to be generated and distributed to all remaining group members.

(b) *Members leaving without forward secrecy*

In the second option, no such restriction applies. Re-keying may not need to occur.

2.2.4 Protocol for Member Moving to Other Areas

We need a protocol for *host mobility*. This protocol governs members moving to other areas. Hand-off operations between affected areas (areas where group members are moving from and moving to) may need to occur prior to members moving in order to establish relevant keys. As the security requirements for a member moving to another area is quite similar to that of a new member joining, two options can be adopted due to host mobility:

(a) *Members moving with backward secrecy*

In the first option, typically if a moving member is to be denied access to an area's security information (the area where the member is moving to), all cryptographic keys associated with that area need to be re-keyed.

(b) *Members moving without backward secrecy*

In the second option, no such restriction applies, and re-keying within an area may not need to occur. Moving members are given the same set of keys that are currently used in that area.

2.2.5 Protocol for Re-keying

Re-keying may need to occur due to:

(a) *Group membership change.*

(b) *Periodic re-keying.*

(c) *Expiration of cryptographic keys.*

(d) *Compromised keys.*

WMobEs may also require re-keying due to *host mobility*. Each of these processes results in all group members obtaining the new cryptographic keys needed for secure group communication. In WMobEs where domains and areas may have their own managing entities, re-keying can be divided into two levels:

(a) *At the domain level*

Re-keying at the domain level is normally initiated and controlled by a domain manager. For example, for the provision of backward secrecy, whenever a new host joins a multicast group, the group's traffic key must be re-keyed. The domain governing entity will have to initiate re-keying of the traffic key by generating a new traffic key and delivering it to all area manager entities and group members.

(b) *At the area level*

Re-keying at the area level is initiated and controlled by an area manager governing that area. Similarly, for the provision of backward secrecy, the area key of an area (where the new join occurs) must be re-keyed. The area manager will have to initiate the re-keying of its area key by generating a new area key and sending it to group members in that area (including the newly joined member).

3 Existing Group Key Management Framework (KMFS)

We look at existing GKMFs and identify where they are deficient for our purpose of establishing a GKMF for a WMobE. None of them provide the full functionality we require, but this exercise will influence our design of a specific GKMF in the remaining Journal.

3.1 Introduction

We look at existing architectures that are relevant in identifying certain attributes that are useful for our purpose. A number of GKMFs have been proposed in the literature, including (Mittra, 1997), (Harney and Muckenhirn, 1997),

© 2014- 15, IJIRAE- All Rights Reserved

(Wong et al., 1998), (Waldvogel et al., 1999), (Wallner et al., 1999), (Hardjono et al., 2000a), (Hardjono et al., 2000b), (Baugher et al., 2003), (Hardjono and Weis, 2004) and (Baugher et al., 2005).
We will present a brief review of the architectures of (Harney and Muckenhirn, 1997), (Mittra, 1997), (Wong et al., 1998), (Hardjono et al., 2000a), (Hardjono et al., 2000b) and (Baugher et al., 2003), since they particularly influenced our later design.

We choose these proposals instead of others (as mentioned) because of the following reasons:
• GKMP-A (Harney and Muckenhirn, 1997) was the first GKMF proposed for multicast communication. As such, it provides a benchmark against which subsequent research can be measured.
• Key-graph (Wong et al., 1998) was the first framework to propose the notion of secure group communication based on a hierarchy of keys. Its idea of having high level keys to protect low level keys is interesting from the perspective of designing scalable frameworks.
• Iolus (Mittra, 1997) was the first GKMF to introduce the idea of subgroups to mitigate scalability problems, due to re-keying events that may need to occur during group operations.
• F-MSEC (Hardjono et al., 2000a) proposed the idea of regions, where each region associates with a different cryptographic key. The idea of multiple regions is natural for most wireless mobile networks and so FMSEC is relevant to our study.
• Intra-domain GKMP (Hardjono et al., 2000b) is based on the F-MSEC (Hardjono et al., 2000a) framework, and extends the idea of regions by introducing domains and areas, each of which is independently managed. This concept fits well with the types of wireless mobile architecture that we have in mind.
• GKM-A (Baugher et al., 2003) is an extension of GKMP-A, and introduced the idea of exchanging multiple security associations (SAs) to establish secure group communication. It also includes improvements in processes such as registration of group members that are of relevance to our requirements.

3.2 Related Frameworks

In this section we present the existing frameworks relevant to our purpose:

3.2.1 GKMP-A

GKMP-A (Harney and Muckenhirn, 1997) was the first proposal to introduce protocols for group key management in multicast communication. The framework can operate in two different modes; *sender initiated*, or *receiver initiated*. In *sender initiated* mode, the sender will act as the group key controller of the multicast group. Otherwise, it will be one of the receivers of the multicast group who controls the key. The sender-initiated model operates in the form of a group key management application, operating on behalf of the originator as well as the group key controller of a multicast group. In the receiver-initiated model, a group member is made responsible for initial group key establishment, as well as the periodic generation and dissemination of new keys.

3.2.1.1 Main Architecture

GKMP-A does not require a centralized key manager (with the exception made for an entity who manages the distribution of security certificates). The main entities involved in the protocol are: (a) *Group key controller (GKC)* The GKC is appointed by the group key management application to be the main group key controller of the group. Either the sender or one of the receivers will be the GKC, depending on the mode of operation. The GKC manages the aspects of key management, including the generation and distribution of cryptographic keys to all members of the group.
(b) *Security manager*
Responsible for creating and distributing authentic identification and security information (such as certificates needed for communicating entities). We observe that because of the introduction of two operational modes, which then determine who will be the GKC for the multicast group, GKMP-A avoids having a centralized key manager as the only parties involved in key management are the same parties (group members) who will be participating in the group communication. GKMP-A does not address other aspects of a framework such as placement of entities in the architecture, group membership policy, and trust relationships.

3.2.1.2 Keys

Keys that are used in the protocol are classified into three categories:
(a) *Group key packet (GKP)*, which contains two keys:
• *Group traffic encrypting key (GTEK)* is the current traffic key for encrypting group data traffic.
• *Group key encrypting key (GKEK)* is the future key used for secure distribution of a new GTEK.
(b) *Session key package (SKP)*, which is unique for each member and contains two keys:
• *Session traffic encrypting key (STEK)* is the key used for encrypting unicast data traffic between a member and the key controller.
• *Session key encrypting key (SKEK)* is the key used for secure distribution of a new STEK to a particular member.
(c) *Group rekey package (GRP)*

© 2014- 15, IJIRAE- All Rights Reserved

The GRP contains a new GKP encrypted with a key encrypting key (GTEK or SKEK) and signed by the originator's private signature key. A GKP is created by the application prior to the establishment of an SKP and
GRP. Although GKMP-A claims to use a combination of both symmetric and asymmetric cryptography, we note that the main focus is on symmetric cryptography, and asymmetric cryptography is only used to provide the originator's (GKC) certificate. It is assumed that all group members have a means to verify the originator's signature.

3.2.1.3 Main Processes

Based on the two aforementioned operational modes, two processes are introduced:

(a) *Generation and distribution of keys*

For *sender initiated* operation, group keys (see *Section 3.2.1.2*) are generated by the application in the form of a GKP. The GKP is distributed to one of the group members selected by the application, which then becomes the GKC of the multicast group. The GKC then contacts each member of the group, creates an SKP (which is unique to each member), creates a GRP (which contains the GKP) and sends these to each of the group members.

For *receiver initiated*, a GKP is created by the application with the first member to initiate contact, which is then distributed to the member. The rest of the operation is similar to *sender initiated*, where other group members are each keyed with an SKP and a GRP.

(b) *Re-keying of group keys*

Re-keying of group keys was briefly addressed. For both modes of operation, when a re-key is required, the application selects a member, creates a new GKP and a new GRP, and distributes the keys to other members. With the exception of these processes, GKMP-A does not address other GKMF protocols such as creation of multicast group, or member joining and leaving protocols.

3.2.2 Key-graph

Key-graph (Wong et al., 1998) was the first proposal to conduct secure group key management by employing a hierarchy of keys in order to reduce the number of messages required for re-keying. The notion of a secure group is organized in the form of a logical tree, as illustrated in *Figure 3.1*.

From *Figure 3.1*, there are two main groups of *nodes*; K-nodes and U-nodes. Every K-node in the tree is assigned a key, and each user (U1. U5) is associated with a U-node and placed as a leaf node. Each user is assigned keys along the path from its leaf to the root node. For example, user U1 is assigned the keys K1, K123, K12345.

Figure 3.1: An example of Key Tree Graph.

Figure 3.1: An example of Key Tree Graph.

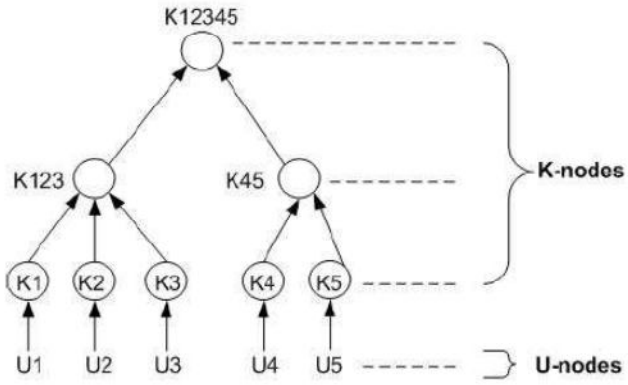


Figure 3.1: An example of Key Tree Graph.

3.2.2.1 Main architecture

The main entities involved in this architecture are:

• *User* : Consisting all group members.

• *A server (s)*: With primary responsibilities for managing group members and keys, it is assumed that the server also acts as the group controller for all group users. Key-graph does not address other aspects of a GKMF such as placement of these entities in the architecture, trust relationships, and group membership policies. We observe that Key-graph adopts a centralized scheme in its design.

3.2.2.2 Keys

Symmetric keys that are used in this architecture are:

(a) *Individual keys*

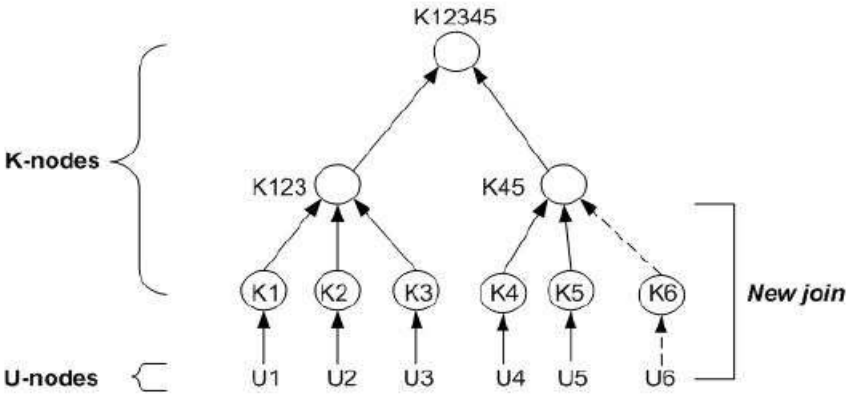
Also referred to as *leaf keys*, an individual key is shared only between a user and the key server (or the group controller). Used primarily for communication between a user and the key server. For example, K1, K5 are individual keys, each associated with user U1..U5 respectively (see *Figure 3.1*)

© 2014- 15, IJIRAE- All Rights Reserved

(b) *Subgroup keys*
Also referred to as *internal keys* (keys which are in-between leaf and root nodes) or *key encrypting keys* (since they are normally used for encrypting other keys). For example, K123 and K45 are subgroup keys, associated with subgroups {U1, U2, U3} and {U4, U5} respectively (see *Figure 3.1*).
(c) *Group keys*
Also referred to as *root keys* (the highest nodes in the tree hierarchy), a group key is shared between all users and the key server. A group key is primarily used for encrypting group data and is also called a *traffic encryption key*. For example, K12345 is a group key shared by all users (see *Figure 3.1*).
These symmetric keys are generated and distributed to users prior to group communication.

3.2.2.3 Main Processes

The main processes described in the framework are re-keying due to hosts joining and leaving. When a new user joins a group, the K-nodes along the path from the new leaf (which is added to assign the new user) to the root are re-keyed with new keys.
Based on *Figure 3.2*, when U6 joins the subgroup {U4, U5}, subgroup key K45 and group key K12345 need to be re-keyed as new K456 and K123456 keys. This results in all users of the subgroup (where the new join occurs) including the new user U6, obtaining new K456 and K123456 keys, and all other subgroup users {U1, U2, U3}, obtaining the new K123456 key. Figure 3.2: Key Tree Graph: A new join by user U6 into the group. When a user leaves, a similar re-keying process applies. For example, when user U6 leaves its subgroup {U4, U5, U6} (see *Figure 3.2*), all users in its subgroup (where the leave occurs) obtain a new subgroup key K45, and all users excluding the leaving user U6, obtain a new group key K12345.



With the intention of reducing the number of messages sent during re-keying, three approaches for re-keying are suggested:
(i) *User-oriented*
In this approach, for each user, the key server sends a re-key message that contains the new key, encrypted with a key held by the user. For example, during a new join of user U6 (see *Figure 3.2*), the server needs to send users:

• U1, U2, U3, new group key K123456 encrypted with K12345.
• U4, U5, new group key K123456 and new subgroup key K456 encrypted with K45.
• U6 (new user), new group key K123456 and new subgroup key K456 encrypted with K6.

(ii) *Key-oriented*
In this approach, each new key is encrypted individually (except keys for the joining user). A user may have to get multiple rekey messages in order to get all the new keys it needs. For example, as in the *user oriented* case, when a user U6 joins the group, the server sends a new group key K123456 and new subgroup key K456 to users U4 and U5 separately, as follows:
• New group key K123456, encrypted with K45.
• New subgroup key K456, encrypted with K45. It is observed that in order to save re-key costs, these re-key messages can be combined and sent to users as one message. Note that these keys need to be sent to the new user U6 separately.
(iii) *Group-oriented*
In this approach, each new key is sent to users (except keys for the joining user) as a *group*. In this case the server sends a single rekey message containing all new keys via multicast to the entire group. For example, as in the *user* and *key-oriented* cases, when a new user U6 joins, the server needs to send users:
• U1..U5, new group key K123456 encrypted with K12345.
• U4 and U5, new subgroup key K456, encrypted with K45.

© 2014- 15, IJIRAE- All Rights Reserved

Note that these keys are sent in one combined multicast message. As with the *key-oriented* case, the new user U6 still needs to be sent these keys individually.

3.2.3 Iolus

While Key-graph (Wong et al., 1998) (see *Section 3.2.2*) introduced the notion of secure group key management based on a hierarchy of keys, Iolus (Mittra, 1997) introduced the notion of secure group communication based on a hierarchy of nodes. Iolus proposed a framework for securing multicast communication by addressing two scalability problems; *1 affect n* and *1 does not equal n*, where n is the number of group members of a multicast group. The first scalability problem occurs when an action of one member affects the entire group. For example, from a key management perspective, when a new host joins a multicast group, an old group key may need to be replaced with a new one. If that is the case, we observe that a *join* operation exhibits the *1 affect n* scalability problem because the protocol requires all members to process the change of obtaining a new key when the new join occurs.

Nevertheless, the basic problem is that each member must be considered individually every time a leave occurs. Thus, we observe that a *leave* operation exhibits both of the aforementioned scalability problems.

3.2.3.1 Main Architecture

Iolus addresses these problems by adopting a number of smaller multicast *subgroups*.
The main entities that are involved in the Iolus framework are:

(a) *Group security controller (GSC)*

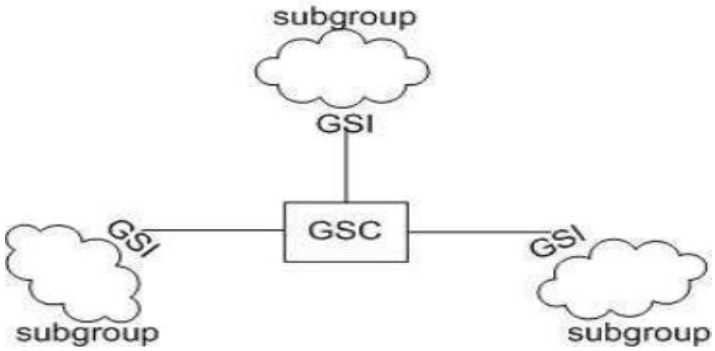


Figure 3.3: An example of hierarchy of nodes in Iolus.

GSC is the root of the hierarchy that maintains and controls the top level subgroup entities.

(b) *Group security intermediaries (GSIs)*

GSIs (also referred to as *group security agents, (GSAs)*) are the top level subgroup entities. One per subgroup, GSIs manage and connect each of the other subgroups. We illustrate the hierarchy of nodes consisting of these main entities of Iolus in *Figure 3.3*. Each subgroup is treated independently from another subgroup in the sense that actions within a subgroup do not affect other subgroups. Iolus is thus a hybrid scheme by design. We observe that although it is not clearly addressed, it is assumed that all group members trust GSIs and GSIs trust the GSC for multicast communication. We also assume that Iolus considers dynamic policy for its group membership.

3.2.3.2 Keys

The Iolus framework uses two symmetric keys, which are:
(a) *GSI-Member key*
This key is unique to every member of the multicast group and shared only with the member's GSI. Used for unicast communication between the GSI and the member, including secure distribution of the subgroup key to the particular member.
(b) *Subgroup key*
Primarily used for data communication, this key is unique to a subgroup and is shared by all group members within the subgroup.

3.2.3.3 Main Processes

The main process described in the framework is for establishing secure multicast data transmission, and Iolus views the whole framework as one virtual group. Iolus does not address creation of multicast groups. For starting secure multicast, Iolus requires a GSC to initiate, which then decides who can or cannot join the group via an access control list (ACL). Once the GSC is established, other hosts such as GSIs and group members can join the group. It is assumed that the *GSI-Member key* and *Subgroup key* are established during the joining process of a particular member to the group. Iolus limits the re-keying that needs to occur due to host joining and leaving to the subgroup level.

© 2014- 15, IJIRAE- All Rights Reserved

When a new join occurs, a member joins its local subgroup. In order to protect access to previous data traffic from the newly joined member (i.e. backward secrecy), the subgroup key needs to be changed. We observe that Iolus tackles the *1 affect n* scalability problem by limiting the effect of re-keying at the subgroup level. Similarly, when a member leaves the group, the subgroup key needs to be rekeyed, and the new key needs to be sent to the remaining group members by excluding the leaving member.

3.2.4 F-MSEC

Motivated by three aspects specific to multicast security; multicast application, scalability and trust relationships among entities, Hardjono et al. (Hardjono et al., 2000a) described a simple framework (referred to here as F-MSEC) for group key management from two aspects:

(a) *Network infrastructure plane*, consisting of entities and functions which define the network aspect of the framework, such as routers, hosts, as well as the routing protocols that they use.

(b) *Key management plane*, consisting of entities and functions which define the security aspect of the framework such as key managers, policy servers, as well as the security protocols that they use.

3.2.4.1 Main Architecture

The key management plane is further divided into 'regions' consisting of:

(a) *Trunk regions*

A trunk region contains only key managers, each of which is associated with at least one leaf region.

(b) *Leaf regions*

A leaf region is where all members are defined to exist. Each leaf region is associated with one key manager. Each key management region consists of *one trunk region* and *one or more leaf region(s)*. Hardjono et al. (Hardjono et al., 2000a) claim that the purpose of introducing trunk and leaf regions is for scalability of the framework, by allowing regions to be defined according to the underlying network infrastructure, as well as the multicast applications that are under consideration, the main entities introduced in the framework are:

(a) *Key managers (KMs)*

Responsible for key management in trunk and leaf regions respectively.

(b) *Key translators (KTs)*

One KT for each leaf region, it translates (from encryptions under different keys) transmission payload across regions. *Figure 3.4* depicts the notion of *trunk* and *leaf* regions, along with the placement of entities in the regions. It is assumed that all group members trust KMs and KTs for group key management.

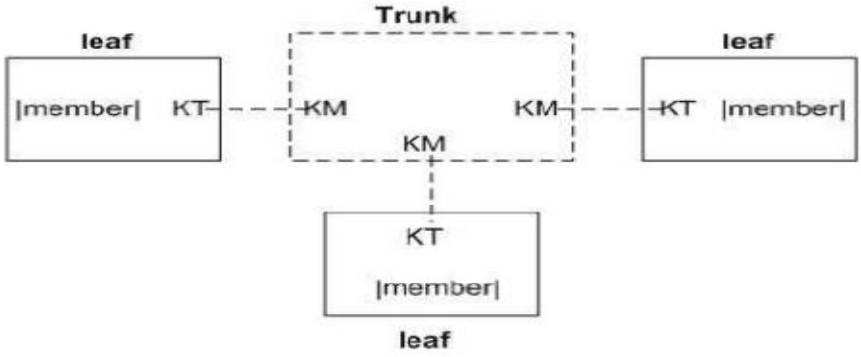


Figure 3.4: An example of notion of trunk and leaf regions in F-MSEC.

Although it is not mentioned whether F-MSEC is centralized or distributed in its design, by having KMs at each of leaf regions we observe that F-MSEC's design is based on centralized scheme. F-MSEC briefly addresses both open and closed policy for its group membership.

3.2.4.2 Keys

In a key management region, F-MSEC introduces two symmetric keys, which are:

(a) *Trunk key*, unique for each trunk region.

(b) *Leaf key*, unique for each leaf region.

F-MSEC makes two interpretations in its view of regions, as well as from the application of cryptographic keys (the trunk and leaf keys) as follows:

(a) *Regions for delivering a group key*

Regions can act as *secure channels* for the purpose of the distribution of a group key. The group key is then used for protecting the multicast data.

© 2014- 15, IJIRAE- All Rights Reserved

(b) *Regions for delivering multicast data*

Different keys that associate with regions (trunk and leaf regions) are used to protect multicast data as it transits across regions. Thus, each region applies a different key to the multicast data, and translations of multicast data across the regions occur in the form of decryption and re-encryption.

3.2.4.3 Main Processes

We observe that F-MSEC does not clearly address any particular GKMF protocols such as creation of new groups, initial registration of keys, distribution of keys, as well as host joining and/or leaving. However, F-MSEC briefly mentions the periodic re-keying that may occur in trunk and leaf regions, and that re-keying of trunk keys (if necessary) does not need to occur when a member at the leaf region leaves a multicast group, because trunk and leaf regions employ different keys.

3.2.5 Intra-domain GKMP

In (Hardjono et al., 2000b), Hardjono et al. proposed Intra-domain GKMP for intra-domain key management for IP multicast security. This work is based on the earlier F-MSEC (Hardjono et al., 2000a) framework. We observe that Intra-domain GKMP extends the idea of *regions* to introduce the notion of a *domain*, where the domain is further divided into a number of smaller *areas*. The definition of domain is viewed from an administrative perspective, where a domain is administered and controlled by one body. In order to distinguish the multicast groups for key management from the multicast group for data, Intra-domain GKMP refers to the latter as *data groups*, and the former as *control groups*. Control groups are area-wide and managed by the area key distributors. Another control group that exists at the domain level is the *All-KD-group*. This consists of all key distributors in the domain. There is only one All-KD-group in a domain. Intra-domain GKMP places group members in one of the areas. Hardjono et al. (Hardjono et al., 2000b) claim that the purpose of placing group members in areas is to achieve flexible and efficient key management, especially when dealing with changes in group membership due to host joining (or host leaving) that may occur during the lifetime of a multicast group.

Thus, like Iolus and F-MSEC, key management operations such as generation and distribution of new keys to new members can be contained within a manageable area.

3.2.5.1 Main Architecture

The main entities that take part in Intra-domain GKMP across a domain are:

(a) *Domain key distributor (DKD)*

DKD manages and controls key management at the domain level.

(b) *Domain multicast address allocation entity (DMAAE)*

DMAAE allocates multicast addresses at the domain level.

(c) *Router/translating entity (R/TE)*

R/TE governs any multicast group that originates from outside of the domain.

(d) *Area key distributor (AKD)*

One AKD per area, which manages and controls key management at the area level.

(e) *Area multicast address allocation entity (AMAAE)*

One AMAAE per area, which allocates multicast addresses at the area level. Trust relationships in the framework revolve around the key distributors (DKD and AKD). All group members trust these key distributors. Although it is not clearly mentioned, we observe that Intra-domain GKMP addresses both policies for its group membership. With the introduction of domains and areas, each of which has its own key managing entity, we note that Intra-domain GKMP is a hybrid scheme in its design.

3.2.5.2 Keys

Intra-domain GKMP uses both symmetric and asymmetric cryptography. However, for simplicity, it only uses asymmetric cryptography for key distribution (by DKD and AKDs), as well as by the multicast address allocation entities (DMAAE and AMAAE) for signing information for the multicast group members. Intra-domain GKMP uses the following group-oriented symmetric keys:

(a) *Multicast key*

Selected by the DKD, the multicast key is unique to a multicast group. Its primary use is to secure group data traffic.

(b) *Area group key*

Selected by the AKD, the area group key is unique for each area and multicast group pair within that area. It is used for secure distribution of the multicast key to all group members of a particular multicast group in an area.

(c) *All-KD-Key*

Assigned by the DKD, this key is used to secure data traffic amongst all key distributors in the domain. Intra-domain GKMP also uses long-term symmetric keys that are issued prior to the distribution of the group-oriented keys, which are:

(a) *Member-private-key*

Unique for each member, this key is shared with the AKD of an area and is established prior to members joining the multicast group.

(b) *AKD-Private-Key*

© 2014- 15, IJIRAE- All Rights Reserved

Unique for each AKD, this key is shared with the DKD and is established before any multicast group exists in the domain. We observe that both of the long-term keys can be used to assist secure distribution of the group-oriented keys.

3.2.5.3 Main Processes

The main processes described in the framework are:

• *Creation of multicast groups*

Intra-domain GKMP described two scenarios in the creation of multicast groups, each of which includes generation and distribution of multicast keys and area group keys of the new multicast group:

(a) *Group initiation from internal-origin*

When an initiator wishes to form a multicast group, it first initiates the creation of a new multicast group with the help of DMAAE at the domain level, which then assigns a multicast group address. The initiator then notifies its local AKD of the new multicast group, and requests a multicast key. The request is then passed to DKD by the AKD via a secure channel (which is encrypted by a secret AKD shares with DKD). DKD then generates the multicast key, and sends the key to every AKD in the domain. Upon receiving the multicast key from DKD, AKD generates and distributes its area group key to the initiator. Although it is not clearly mentioned, we assume that the multicast key is sent to the initiator by the AKD along with the area group key.

(b) *Group initiation from external-origin*

Group initiation from outside the domain can be done by the initiator via a border router, which then notifies the DKD of the request to create a multicast group. Alternatively, the initiator can make itself known to the AKD or the DKD directly. Although it is not clearly mentioned, we assume that the rest of the group initiation process (i.e. generation and distribution of the multicast key and area group key) is similar to the group initiation that occurs internally.

• ***Re-keying***

Intra-domain GKMP addresses re-keying of multicast keys, area group keys, and the All-KD-Key. Re-keying of multicast keys is initiated and controlled by the DKD.When it is necessary to re-key the multicast key, the DKD generates and distributes a new multicast key to all AKDs in the domain, which then deliver it to members in their areas. Re-keying of area group keys is similar to multicast keys, except that it is initiated and controlled by the AKD of an area. We observe that area group keys are only known to the AKD and members in that area. Re-keying of the All-KD-Key is initiated and controlled by the DKD. When it is necessary to re-key the All-KD-Key, the DKD generates and distributes the new All-KD-Key to all AKDs in the domain.

• *Host joining*

Intra-domain GKMP addresses two scenarios pertaining to a host joining a multicast group:

(a) *Host joining with backward confidentiality:* If backward confidentiality is required, re-keying of the multicast key and area group key is conducted. When a new host joins, rekeying of the multicast key is initiated by the DKD, and re-keying of the area group key is performed by the AKD.

(b) *Host joining without backward confidentiality:* If no provision of backward confidentiality is required, re-keying does not need to occur, and the newly joined member is given keys that are currently in use by the group.

• *Host leaving*

Intra-domain GKMP addresses host leaving from the perspective that rekeying must occur to preserve forward confidentiality. Like host joining, re-keying of the multicast key and area group key (where the member leaving resides) is conducted, which results in all AKDs and group members of the multicast group (excluding the leaving member) in the domain obtaining new keys. Intra-domain does not consider host leaving without forward confidentiality.

3.2.6 GKM-A

This proposal (Baugher et al., 2003) is an extension of GKMP-A (see *Section3.2.1*). GKM-A provides a common architecture for group key management protocols in multicast security that supports a variety of protocols. Each protocol can be specialized for different purposes, depending on the type of application under consideration. GKM-A introduces the notion of a *Group Security Association* (GSA), which consists of a group of conventional security associations (SA) (which is a set of security parameters that two entities share, including encryption keys, authentication and integrity keys, as well as other attributes such as cryptographic policy of the keys and a reference index of the SA), and needs to be established prior to group communication.

GKM-A uses three types of SA to establish common keys amongst the group members, which are: (a) *Registration protocol SA*

This SA protects the registration protocol, which occurs between the group controller and each group member. (b) *Data protocol SA*

This SA is to protect the data communication between the communicating entities.

(c) *Re-key protocol SA*

This SA is optional because cryptographic keys may not need to be updated. The architecture also suggests that all keys could be delivered by the registration protocol.

3.2.6.1 Main Architecture

To facilitate establishment of these SAs, the main entities involved in the architecture are:

© 2014- 15, IJIRAE- All Rights Reserved

(a) *Group controller and key server (GCKS)*
A separate and logical entity that performs member authentication and authorization according to the group policy set by the group owner.
(b) *Policy infrastructure*
A separate entity that dictates group policies for multicast groups.
(c) *Authorization infrastructure*
A separate entity that provides credentials and certificates needed by any communicating entities. We observe that GKM-A adopts a centralized scheme in its design and addresses both policies for its group membership.

3.2.6.2 Keys

In the architecture, GKM-A uses two types of symmetric key:
(a) *Key encrypting key (KEK)*
One (or more) KEKs are used to protect the TPKs and other KEKs.
(b) *Traffic protection key (TPK)*
TPK is used for securing data traffic. Like the aforementioned SAs, these keys are generated and established during the registration protocol and/or re-key protocol. Although not clearly addressed in the proposal, we observe that GKM-A also considers asymmetric cryptography through the use of GCKS' credential (certificate) by the group members. We assume that the credentials are issued and managed by the authorization infrastructure.

3.2.6.3 Main Processes

As we have noted, the main processes involved are:
• *Registration protocol*
In this protocol, both entities (GCKS and group members) establish a registration SA, where entities authenticate each other, and information (such as which multicast group a member is registered to, group policy, and keys) is obtained. During this protocol, other SAs such as a data SA (for securing group data traffic) and a re-key SA (if necessary) are established. This includes generation and distribution of a KEK and TPK to the member.
• *Re-key protocol*
This is an optional protocol, as dictated by a group policy in cases where keys (KEK and/or TPK) may need to be updated whenever there is a change in group membership due to new members joining and/or members leaving a multicast group, creation of new keys by GCKS, or when keys are given expiration periods. We observe that the GCKS is responsible for managing the re-key protocol. Apart from these protocols, GKM-A does not clearly address other GKMF protocols such as creation of new multicast groups, new members joining, or members leaving.

4. GKMF for WMobE: Scope and Requirements

4.1 Scope of Proposal

Regarding the scope of our GKMF specification, it is important to note the following:
• Infrastructure-based environment. The framework relies on an infrastructure based environment with a basic underlying cellular architecture (Bhargava et al., 2000), (Lin and Chlamtac, 2001) and (Park et al., 2002) as its networking platform. We do not intend to extend its usage to non-infrastructure environments such as wireless adhoc networks, or wireless sensor networks.
• Group key management. Our proposal focuses solely on the GKMF, whose main goal is to provide fundamental security support by providing all communicating entities with the necessary cryptographic keys, and providing a means to distribute these keys for the purpose of group communication. It is not the aim of the proposal to specify the details of the real data communication and how keys are used during the course of such communication.
• Key distributions and key updates. The aspects of key management that the framework is primarily concerned with are *key distribution* and *key updates (or, re-keying)*. We do not consider other aspects of key management in detail (such as the generation, storage and the disposal of cryptographic keys). Each of these is important and should be conducted in a proper and secure manner as required by the multicast application in place. We do not treat these here because they can be handled by generic techniques that are not specific to multicast group communication.
• Type of Multicast Applications. Multicast applications can be categorized as *one-to-many* or *many-to-many* relationships, depending on whether a single (or, many) sender(s) transmit data traffic to many receivers (group members) in the multicast group communication. Since the scope of the proposal is primarily concerned with key management and is not concerned with the real data communication, the type of multicast application in place does not matter and does not affect the proposal design. Therefore, our proposal does not impose any restriction on the type of multicast application in place.
• Generic model. The framework proposed is stated in sufficient abstraction that it can easily be made compatible with existing network protocols, as well as application-layer security protocols to allow for practical implementation for group communication in WMobEs. In order to further support this, we suggest the use of mechanisms and techniques that are based on standards (such ISO/IEC and Internet Standards).

© 2014- 15, IJIRAE- All Rights Reserved

4.2 Properties and Design Requirements

We divide the requirements into three categories; general requirements, security and trust requirements, and performance requirements.

4.2.1 General Requirements

• *Dynamic group membership policy*. Group members should be allowed to join and/or leave a multicast group at any time.

• *Provision for host mobility*. A protocol should be provided for allowing group members to move to other areas while still remaining in a group session.

• *Scalability*. A GKMF should be scalable enough to expand its group size over time.

• *Reliable and trustworthy key manager*. All key managers managing the cryptographic keys and governing group operations should be trusted by group members.

4.2.2 Security and Trust Requirements

The list the main security and trust requirements for our proposal:

• *Entity authentication*. Both group members and key manager entities should be able to authenticate and verify each other's identities.

• *Backward and forward Secrecy*. From the aspect of key management, rekeying needs to occur whenever there is a change in group membership (either due to new joins, member leaves, or member moves) for provision of backward and forward secrecy.

• *Data (Message) integrity and authentication.* Both group members and key manager(s) should be able to verify the integrity of data received.

• *Secure data exchange*. Communications amongst group members should be protected (and remained confidential), where authorized access is given only to group members.

• *Key distribution.* All necessary keys should be securely distributed to all group members prior to group communication.

• *Key updates (re-keying).* Key updates should be done in a secure and proper manner. Group members should be informed when an update of keys is required.

• *Additional key management during host mobility*. Generation, distribution and re-keying of new cryptographic keys may be required to support host mobility.

• *Trust model*. A trust model is crucial for a GKMF in order for it to properly function. This includes a security architecture that is compatible with existing network protocols, scalable and transparent to higher level applications and services.

4.2.3 Performance Requirements

• *Computation, energy and storage requirements.* Many mobile devices have limited processing power, as well as storage capacity, and these should be kept to minimal. This includes requirements for:

(a) Keeping the number of encryptions (or decryptions) involved to a minimum.

(b) Using encryption (or decryption) techniques that are not computationally intensive.

(c) Keeping the number of keys that a member (device) needs to store to a minimum.

• *Communication and bandwidth requirements*. As wireless mobile networks are susceptible to frequent disconnection, and available bandwidth is limited, the communication overhead between group entities should be kept to a minimum

4.3 Main Architecture

We use the main structural components to describe the architecture in detail. We first determine the aspects that influenced our design decision.

4.3.1 Design Influence

(a) *Domains and Areas*

We will adopt the notion of *domains* and *areas* as the main structural components in the framework architecture. This idea facilitates scalable and efficient distribution of keys to all group members, as group members are defined to exist in individual areas that are locally managed by a trusted entity. We will extend these ideas by introducing *inter-domain* relationships for group communications that originate from other domains. This was not considered in existing proposals.

(b) *Subgroups*

By placing group members in individual areas, we can associate them with the concept of subgroups. In other words, group members are *sub grouped* in particular areas. By doing so, we seek to overcome scalability problems that may occur whenever there is a change in group membership. When a new member joins (or an existing member leaves) a multicast group, it joins (or leaves) it's local area and does not affect the other *subgroups* (in other areas) in the domain. Host mobility between areas can also be managed in a more efficient way.

(c) *Symmetric cryptography*

We follow previous GKMF proposals, and adopt symmetric cryptography in our proposal. This is primarily due to reasons pertaining to the nature of the wireless mobile environments that our framework is intended for. See *Section 4.3.6* for more details.

(d) *Key hierarchies*

© 2014- 15, IJIRAE- All Rights Reserved

Hierarchies of keys are very useful for group communication in WMobEs where group members may move between areas that may have their own security requirements. We will adopt the concept of key hierarchies in our framework (see *Section 4.3.6*).

(e) *Domain key manager (DKM)*

At the domain level, a DKM is defined to exist, whose main responsibility is generating, distributing, storing and deleting all keying materials that may be required. We also assume that the DKM plays the role of group controller, which includes managing group policies, group membership, re-keying events and security policies. However, in practice, the domain key manager and group controller can be separate entities, as observed in (Harney and Muckenhirn, 1997), (Mittra, 1997), (Hardjono et al., 2000b) and (Baugher et al., 2003). We assume that there is only one DKM per domain, and that it is the main reference for security parameters for other key managers in the domain. In collaboration with other key managers, the DKM governs host mobility across the domain.

4.3.3 Domain(s) and Area(s)

In this section, we look more closely at the domain(s) and area(s) within the architecture. We also discuss multiple domain relationships, in the cases where inter-domain group communication is permitted. The notion of domain(s) and area(s) provides a means to have an administratively manageable environment for group communication to take place, most importantly for efficient key management. In our proposal, a *domain* can be logically or physically defined. Either way, it is controlled and managed by a trusted entity operating under one system, for instance the *Global System for Mobile Communications* (GSM) operator's network (Lin and Chlamtac, 2001).

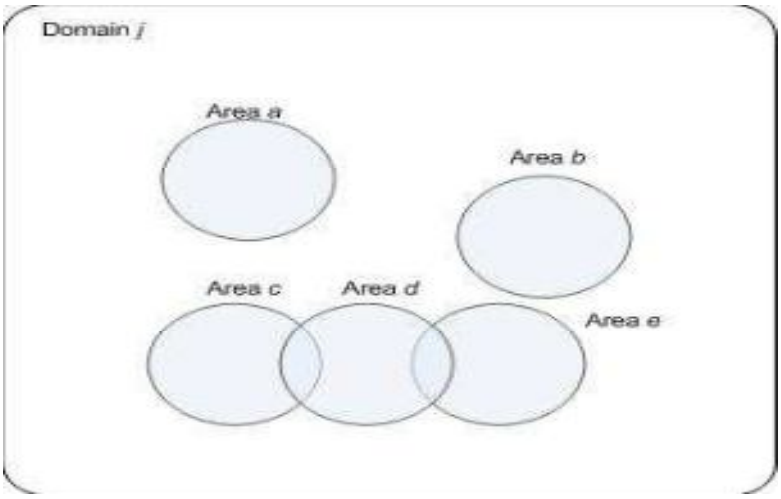


Figure 4.1: An example showing the notion of domain and area.

The domain is further divided into a number of smaller manageable areas, each of which is managed by an AKM, operating closely with the DKM with regards to key management. We illustrate the notion of domain and areas in *Figure 4.1* where, Domain j is further divided into several areas labeled Area a to e, each of which can be logically or physically overlapping with one another. Since a domain is controlled by one DKM, all corresponding entities across a domain should be able to interface successfully with one another. Although areas within a domain may be using similar systems for interoperability, this does not change the fact that each area is unique and that a group member that moves from its local area to another area must obtain security information (i.e. keys) associated with that area prior to, or during, the move.

(a) *Local areas*

The term *local area* is used to refer to the area where hosts (potential group members) first join a multicast group.

(b) *Visited areas*

The term *visited area* is used to refer to other areas in a domain, where group members may or may not *move to* (during host mobility) throughout the lifetime of their group membership.

4.3.3.1 Inter-domain Relationship

In this section, we introduce the idea of *inter-domain* relationships. This is useful in cases where group operations originating from outside the local domain are permitted (for example, when a host or potential member wishes to join a multicast group that is managed by other domain). This kind of request is called a *cross-domain request*. There are two approaches that can be adopted in dealing with inter-domain communication:

(a) *Use of an intermediate entity*

The first approach is to have a separate entity, which can be in the form of a server or a router, to deal with any inter-domain communication (if it occurs). For example, any request to join a multicast group that is not in the current domain where the request originates from is transferred to the aforementioned entity, which deals with the inter-domain requests.

© 2014- 15, IJIRAE- All Rights Reserved

This entity thus acts as an intermediate node or a bridge between the two distinctive domains. An intermediate host may cater for two or more inter domains communications depending on its hardware or software capacity.

(b) *Cross-referencing between DKMs*

An alternative approach is the cross-referencing between DKMs. In this case, DKMs from both affected domains are the ones governing the inter domain requests. For example, let Di and Dj denote domain i and domain j. Any host who wishes to join a multicast group outside its local domain Di (the point of where it is residing at the time of the request), is managed by its domain key manager (DKM). The DKM then liaises with the DKM in Dj to govern the request, including any security relationship exchange that may occur during the course of the host request. Both DKMs need to collaborate in order to realize inter-domain communication.

4.3.4 Placement of Entities

In this section, we place the entities specified in *Section 4.3.2* in the domains and areas specified in *Section 4.3.3.* As mentioned in *Section 4.3.2*, a DKM oversees key management at the domain level, and an AKM oversees key management at the area level. We illustrate placement of entities in two instances in *Figure 4.2* and *Figure 4.3.*

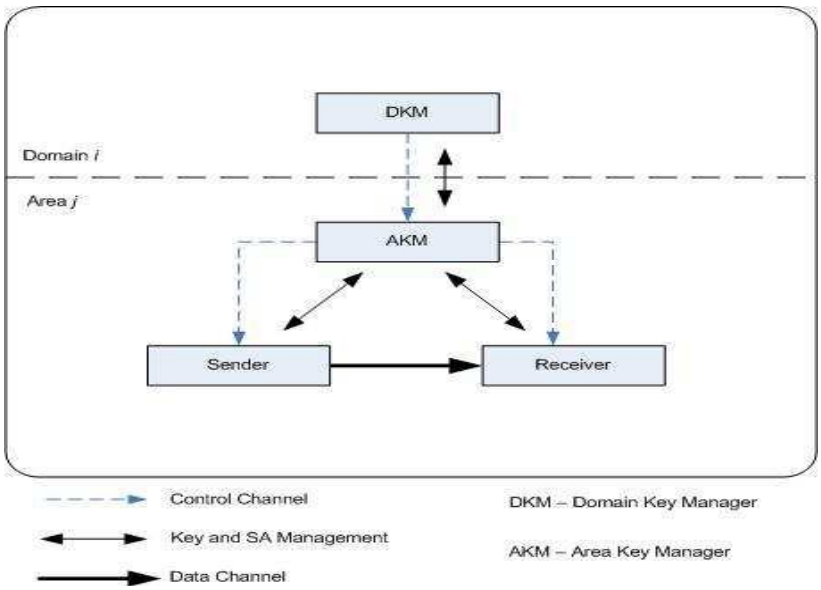


Figure4.2: Placement of entities in domain i and area j.

Figure 4.2 shows placement of entities in domain i and area j. From the illustration, while DKM is the main key manager of domain i and AKM is the key manager of the area j, we assume there is a sender and a receiver to represent the group members of the multicast group in place. The horizontal dotted-line shows a logical division between domain i and area j. The *dotted-arrow* lines from DKM to AKM, as well as from AKM to both sender and receiver, show *control channels* which can be used by the DKM and AKM for transmitting control messages, such as notification on re-keying that has taken place, or acknowledgement of messages received. While *double-arrow* lines show the exchange of key and SA management between pairs of entities, the *single-arrow* line from sender to receiver shows the data channel of real group communication which may take place after the

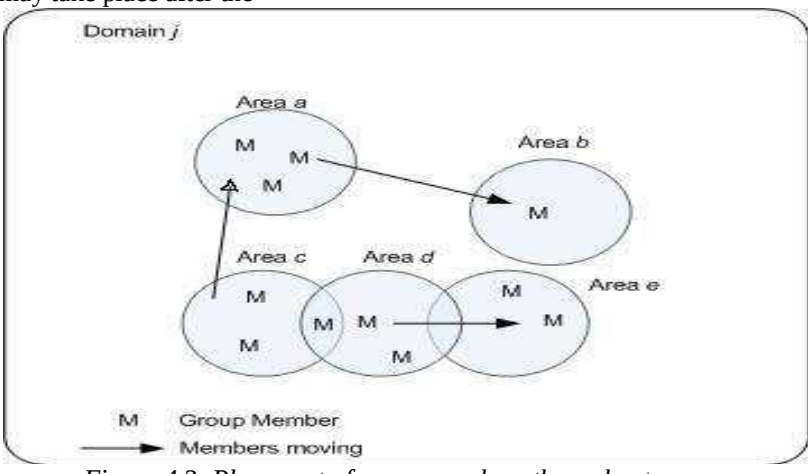


Figure 4.3: Placement of group members throughout areas.

© 2014- 15, IJIRAE- All Rights Reserved

Exchange of key and SA management occurs. On the other hand, *Figure4.8.3* shows placement of group members M across a domain j, where distribution of members occurs throughout the areas a to e. The *arrows* denote the movement of group members between the areas.

4.3.5 Trust Relationships

Trust relationships often revolve around key managers, who are the main key distributors. In our framework, we assume that all key managers (DKM and AKMs) in a domain are trustworthy and reliable. All group members of multicast groups trust these key managers, particularly for providing secure key management services. There are two levels of trust relationships:

(a) *At the domain level*

At the domain level, all AKMs trust the DKM as the primary key distributor, as well as the main group manager for various multicast groups operating in that domain.

(b) *At the area level*

At the area level, all group members (residing in that area) trust their AKM as the main reference point for security parameters needed for group communication.

4.3.6 Types of Key

In this section, we look at another of the fundamental components of the proposed architecture, namely the cryptographic keys. We base our framework on symmetric key cryptography because most mobile devices that operate in WMobEs exhibit special characteristics, which benefit from the computationally faster and less complex techniques offered by the symmetric approach. In the following sections, we look at the symmetric keys used in our framework, which are categorized into two main groups; *long-term* and *short-term* secret keys. We assume that all symmetric keys used are of an appropriate recommended length (at the time of writing we recommend 128 bits), allowing the use of standard algorithm such as AES (FIPS, 2001). We end this section by looking at the aspects of key management that we assume are in place and available for the purpose of group key management, and that we will thus not fully specify in the framework.

4.3.6.1 Long-term Keys

Long-term keys are assumed to have been established prior to any host joining a multicast group. Entities in the framework use these keys to securely initiate secure group communications, including when disseminating short-term keys. There are three types of long-terms key in the framework; *Domain-Area keys, Domain keys* and *Area-Member keys*. The following subsections describe each key in turn, and their details are summarized in Table 4.1.

Key	Generated by	Held by	Function
Domain-Area key	DKM	DKM, AKM	(a) Unique to DKM and a specific AKM. (b) Supports *unicast* communication between DKM and AKM. (c) Supports secure distribution of the *domain key*.
Domain key	DKM	DKM, AKM	(a) Common key for DKM and all AKMs. (b) Supports *multicast* communication amongst DKM and AKMs. (c) Supports secure distribution of the *traffic key*.
Area-Member key	DKM	DKM, AKM, M	(a) Unique to AKM and a specific M. (b) Supports *unicast* communication between AKM and M. (c) Supports secure distribution of the *area key* in an area.

• Domain-Area Key, DAi−Key

The Domain-Area key is the unique long-term key shared between the DKM and a specific AKM in a domain. More precisely, DAi−Key corresponds to the symmetric key shared between DKM and the area key manager AKMi of area i. This unique key is established with every AKM in the domain prior to any request to create multicast groups in the domain. Each key is generated and distributed by the DKM to every AKM by an appropriate secure means (see *Section 4.3.6.3*).

• Domain Key, D−Key

The Domain key D−Key is the long-term key shared by all key managers (i.e. DKM and all AKMs) in a domain. Like the Domain-Area key, the domain key is established prior to any request to create multicast groups in the domain. D−Key is generated and distributed by the DKM to all AKMs via secure channels. One such channel is created by unicasting under the appropriate *Domain-Area* keys. Since the group membership of all key managers is static, D−Key is also fixed and valid until the policy determines otherwise. A new domain key must be generated and distributed to all AKMs to replace the old one on its expiry.

• Area-Member Key, AiM−Key

© 2014- 15, IJIRAE- All Rights Reserved

Another long-term key used is a unique Area-Member key shared between the AKM of an area and every group member residing in that particular area. More precisely, AiM−Key corresponds to a symmetric key shared between area key manager AKMi and a group member M. This key is obtained during the first contact that a host (who soon will become a member of a multicast group) makes to the AKMof a particular area to register with a particular multicast group. More precisely, at the domain level, the DKM uses the Domain-Area key to secure the distribution of the Area-Member key to a particular AKM of an area. Then, at the area level an AKM uses a *secure means* (see *Section 4.3.6.3*) to distribute the key to the group member M. Since group membership can be dynamic, we assume that the Area- Member key remains valid throughout the lifetime of a particular multicast group, or until the group member has ceased to be a member of that particular multicast group.

4.3.6.2 Short-term Keys

Short-term symmetric keys are assumed to have been established after group members join multicast groups (or after the long-term secrets have been established). As group members are already in possession of long-term keys, these are often used to assist secure distribution of short-terms keys. There are three short-terms keys introduced in the framework; *traffic encryption keys, area keys* and *session mobility keys.* The following sections describe each type of key in turn, and at the end their details are summarized in Table

Traffic Encryption Key, T−Key

The Traffic Encryption Key (or *traffic key*) T−Key is a short-term key shared by all group members of a particular multicast group in a domain. There is a unique traffic key for a specific multicast group. T−Key is generated and distributed by the DKM to all AKMs in the domain, which then in turn disseminate this key to all group members residing in their area. The establishment of the key takes place only after the first host creates (and joins) a multicast group. There are several options for securing the distribution of the traffic key to all group members in the domain. One option is for the DKM to use *unicast,* by protecting the key under each Domain-Area key and thus sending it to every AKM independently.

• Area Key, A−Key

The Area key is a short-term key unique to an area. Every area has a different A−Key. An area key is generated and distributed by the AKM of a particular area, and shared only by group members residing in that area. An area key is established with a group member after it joins a multicast group. The dissemination of the area key to all group members is done by the AKM via *unicast* methods using each Area-Member key. The main purpose of having an area key, which is unique to an area, is:

(a) *To securely manage host mobility across areas in the domain* without proper control, group members that are moving from one area to another may collect security information, including old keys not authorized to them. In addition, different areas may have different access control pertaining to their local information. By having adequate access control, one can prevent such security violations by unauthorized moving members.

(b) *To provide efficient and scalable re-keying*

Unique area keys are useful for efficient re-keying and promote scalability, since group members within an area are managed under one unique key. Note that while group members of a multicast group can be dispersed across the domain due to host mobility, each group member may have in his possession a different set of area keys.

• Session Mobility Key, Sm−Keyiv

The Session mobility key Sm−Keyiv is a short-term symmetric key shared between an AKMand a moving group member. More precisely, Sm−Keyiv is a session mobility key shared between an area key manager AKMv and a group member Mi. This key is only used for host mobility and exchanged during the hand-off operation.

This key is established between a moving member and the AKM of a visited area prior to host mobility. The generation and initial distribution of this key is conducted by the DKM in a domain, then delivered to the group member via an AKM in the local area (where the member is currently residing). The same key is then delivered to the AKM of a visited area by the DKM. The delivery of the session mobility key must be done using secure channels (see *Section 4.3.6.4*). This key is used for any *unicast* communication that may occur between the AKM of the visited area and the mobile member throughout its residence period in that area. This includes the secure distribution of the visited area's area key to the mobile member.

© 2014- 15, IJIRAE- All Rights Reserved

Key	Generated by	Held by	Function
Traffic key	DKM	DKM, AKM, M	(a) Common key to all Ms in a group. (b) Unique to a specific multicast group. (c) Secures the actual data communication.
Area key	AKM	AKM, M	(a) Unique to an area. (b) Supports *multicast* communication amongst AKM and all Ms in a group. (c) Supports secure distribution of the *traffic key* to Ms in a group.
Session Mobility key	DKM	DKM, AKM, M	(a) Unique to AKM and M. (b) Supports *unicast* communication between AKM and M. (c) Supports secure distribution of *area key* for host mobility.

Table 4.2: Summary of short-term keys and their functions

4.3.6.3 Assumptions on Aspects of Key Management

As mentioned in *Section 4.1*, we will focus on distribution and updating of (mainly short-term) cryptographic keys. Other operations are not treated here in detail because their provision can be achieved by generic solutions which are not specific to group communication. The following assumptions regarding key management operations are made:

(a) Key generation is conducted in a secure and proper manner by all key managers (DKM and AKMs) in a domain. We assume that key manager's use recognized key generators to generate keys as randomly as possible.

(b) Long-term key distribution is conducted in a secure manner, prior to any group being established. These keys can be established using various key establishment methods in ISO/IEC 8732 (ISO, 1988), ISO/IEC 11770- 1 (ISO, 1996a) and ISO/IEC 11770-2 (ISO, 1996b). There are several accepted methods that can be used to distribute keys to the communicating entities, including via physical (manual) delivery techniques, using other key to encrypt keys (key encrypting keys), or using a trusted third party.

(c) Key storage is managed in environments equipped with secure technology. For example, tamper-resistant hardware can be used to increase the level of security of the stored keys.

(d) Key installation is performed securely by all key managers (DKM and AKMs). We assume that a DKM is responsible for key installation at the domain level, and an AKM is responsible for key installation at the area level.

(e) Key revocation is conducted in a secure and proper manner by all key managers in a domain. We assume that a DKM governs any process to revoke keys at the domain level, and likewise an AKM at the area level. (f) Key disposal is handled in a secure and proper manner by all key managers such that no other information can be used to recover the disposed keys. We assume that DKMs and AKMs manage key disposal processes.

4.3.6.4 Secure Channels

We use the term *secure channel* to mean that communication between group entities (key managers and group members) within the GKMF is protected by careful application of symmetric keys. This is achieved as follows:
(a) Key managers (DKM and AKMs) at the domain level secure the communications between them by using either a common key such as *domain key* for protecting communications between all key managers, or *Domain-Area* keys for secure communications between the DKM and each AKM separately.

(b) Area key managers and group members in the same area secure communications either by using a common *area key* for protecting communications between the AKM and all group members (residing within that area), *Area-Member* keys between the AKM and a group member separately, or *session mobility* keys between an AKM and a mobile member. Secure channels are created when group entities (DKM, AKMs and group members) use these keys in the course of group communication.

4.4 Protocol Functionalities

Creating New Group and Initial Distribution of Keys This protocol governs the creation of a multicast group and describes the initial key distribution to all key managers and the first host (group member) to create a multicast group. The main functional requirements of this protocol are to:

• create a multicast group for the purpose of group communication.

© 2014- 15, IJIRAE- All Rights Reserved

• distribute a traffic key T−Key by the domain key manager to all area key managers.
• distribute a traffic key T−Key and an area key A−Key by the area key manager (of an area where the host residing) to the first host to create a multicast group.
The main security requirements of this protocol are to ensure that:
• Only an authorized host is allowed to create a multicast group.
• Communication between the domain key manager and area key managers is secure.
• Communication between the area key manager and the host (or group initiator) is secure.
• The distribution of the traffic key T−Key and the area key A−Key to the area key manager and the host is protected.
The main information disseminated during this protocol includes:

• *Backward and forward secrecy*
The need for these confidentiality services is indicated by a *2-bit binary* flag value, as shown in *Table 4.3*, where these services are presented as *on* for a *required* service, or *off* for a *not required* service.

Flag value	Confidentiality Service		Security Requirement
	Backward Secrecy	*Forward Secrecy*	
00	Off	Off	None
01	Off	On	Only forward secrecy
10	On	Off	Only backward secrecy
11	On	On	Both

Table 4.3: Requirements for backward and forward secrecy

This flag value is indicated in the protocol message contents as *conR* (*confidentiality requirement*).

8.4.2 New Member Joining
For a group communication to take place, a multicast group should have at least two (or more) entities in place. This protocol governs the *join* (or registration) of group members into the multicast group, when a host wishes to join a multicast group, two approaches can be adopted based on whether providing backward secrecy is necessary or not. If providing backward secrecy is required then any new host that wishes to join the group must not be allowed to access previous traffic prior to its admission to the group. Otherwise, no such restriction applies. Thus, this protocol will consider both options of hosts joining without and with backward secrecy.

The main functional requirements of this protocol are categorized in two parts, corresponding to the aforementioned *join* options:
(a) For host joining *without* backward secrecy, the main functional requirements are to:
• add new group members into a multicast group.
• deliver a set of keys (which are traffic key T−Key and area key A−Key) to the newly joined member.
(b) For host joining *with* backward secrecy, in addition to those listed in (a), other functional requirements are to:
• initiate a re-keying of a traffic key T−Key in the domain.
• initiate a re-keying of an area key A−Key in the area (where the join occurs).

4.4.3 Existing Member Leaving
Dynamic environments allow group members to leave multicast groups at anytime. This protocol governs the leaves (or de-registration) of a group member from a multicast group, two options can be adopted depending on whether the provision of forward secrecy is necessary, or not. If it is, then past group members (including the ones who are leaving) must not be allowed to access future communications. Otherwise, no such restriction applies. This protocol will consider both options of members leaving with and without forward secrecy. Unlike members joining, leave cases can be *voluntary*, or *involuntary*. Involuntary leaves in particular may require protection from the ejected members (in such cases forward secrecy is necessary). As in members joining, while members leaving without forward secrecy will not affect the current group keys (no updates of keys are necessary), members leaving with forward secrecy require the protocol to re-key the group with a new set of keys. In the event that a leave notification originates from outside the current domain, the request is managed accordingly by the governing entity in place. The main functional requirements of this protocol depend on the aforementioned *leave* options, and are listed as follows:
(a) For members leaving *without* forward secrecy, the main functional requirement is to remove existing group members from a multicast group.
(b) For members leaving *with* forward secrecy, in addition to those listed in (a), the other main functional requirements are to:
• initiate a re-keying of a traffic key T−Key in the domain.

© 2014- 15, IJIRAE- All Rights Reserved

• initiate a re-keying of an area key A−Key in the area where the leave occurs, as well as in area(s) visited by the leaving member.
The main security requirements of this protocol are to ensure that:

4.4.4 Member Moving to Other Areas

In dynamic wireless mobile environments, group members are not just allowed to join (and/or leave) a multicast group, but are also allowed to move between areas while remaining in a group session. As each area may have different security requirements, as well as keeps its own security information (such as old keys), members moving to other areas can be viewed as new joins, each of which may require protection of its local information from an entering member. Because a moving member may accumulate information for each area it visits, two options can be adopted depending on whether the provision of backward secrecy is necessary or not for controlling access to an area's past security information (which could be used for malicious purposes). This protocol will thus consider both options of members moving with and without backward secrecy.
The main functional requirements of this protocol are divided into the two *moving* options, as follows:

(a) For members moving *without* backward secrecy, the main functional requirements are to:
• transfer a group member from one area to another area.
• deliver an area key of a visited area to a moving member.
(b) For members moving *with* backward secrecy, in addition to those specified in (a), the other main functional requirements are to:
• initiate a re-keying of an area key of a visiting area.
(c) The main functional requirements of this protocol are to:
• establish a session mobility key between a moving member and an AKM of a visited area.
• deliver a session mobility key to a moving member and to an AKM of a visited area.
The main security requirements of this protocol are to ensure that:
• Only transfers from authorized group members are processed.

4.4.5 Re-keying

This section describes re-keying protocols of the cryptographic keys that are needed for a multicast group communication, in particular the traffic key
T−Key and the area key A−Key.
Re-keying may need to occur if the provision of backward and/or forward secrecy is necessary for a multicast group communication. In addition, dynamic mobile environments may also require a multicast group to re-key (or update) group members with a new set of keys.
We explicitly describe re-keying operations for the following two types of key:
(a) Re-keying the traffic key
This protocol governs re-keying of a traffic key of a multicast group. In particular, re-keying of a traffic key occurs to satisfy the design requirements of *Protocol II(b): Member joining with backward secrecy*, and *Protocol III(b): Member leaving with forward secrecy.* .Note that re-keying of a traffic key does not always need to occur during a host mobility protocol.

The main security requirements of this protocol are to ensure that:
(b) Re-keying the area key
This protocol governs re-keying of an area key. In particular, re-keying of an area key occurs not only to satisfy the design requirements of *Protocol II(b)* and *Protocol III(b)*, but also *Protocol IV(b): Member moving with backward secrecy*. The main functional requirements of this protocol are to distribute a new area key to:

REFERENCES

[1]. Rafaeli, S. and Hutchison, D. (2003). A Survey of Key Management for Secure Group Communication. ACM computing Surveys, vol. 35, no. 3, pp. 309-329.
[2]. 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.
[3]. S. Mittra, "Iolus: a framework for scalable secure multicasting," SIGCOMM Compute. Common. Rev., vol. 27, pp. 277-288, 1997.
[4]. Wallner, D.,E.J. Harder, and Agee, R. (1998). Key Management for Multicast: Issues and Architectures, Internet Draft (work in progress), draft-wallner-key-arch-01.txt, Internet Eng. Task Force.
[5]. Steer, D., Strawczynski, L., Diffie, W., and Wiener, M. (1990). A secure audio teleconference system. In Advances in Cryptology -- CRYPTO'88.

© 2014- 15, IJIRAE- All Rights Reserved

[6]. Chan, H., Perrig, A., and Song, D. (2003). Random Key Pre-distribution Schemes for Sensor Networks. To appear in Proc. Of the IEEE Security and Privacy Symposium.
[7]. Du, W., Deng, J., Han, Y., and Varshney, P. (2003). A Pairwise Key Pre-distribution Scheme for Wireless Sensor Networks. In Proc. of 10th ACM Conference on Computer and Communications Security (CCS), Washington DC.
[8]. Chan, H. and Perrig, A. (2005). PIKE: Peer Intermediaries for Key Establishment in Sensor Networks, In Proceedings of IEEE INFOCOM.
[9]. Chan, A. (2004). Distributed Symmetric Key Management for Mobile Ad hoc Networks, IEEE INFOCOM.
[10]. Shamir, A. (1979). How to Share a Secret. Communications ACM 1979; 22(11), pp. 612–613.
[11]. Wong, T., Wang, C., and Wing, J. (2002). Verifiable Secret Redistribution for Threshold Sharing Schemes. Technical Report, CMU-CS-02-114-R, School of Computer Science, Carnegie Mellon University.
[12]. Stadler, M. (1996). Publicly Verifiable Secret Sharing. Proceeding of Eurocrypt'96. pp. 190-199.
[13]. Luo, H., Zerfos, P., Kong, J., Lu, S., and Zhang, L. (2001). Providing Robust and Ubiquitous Security Support f or Mobile Ad-hoc Networks. Proceeding of The 9th International Conference on Network Protocols.
[14]. Wu, B. and Wu, J. (2007). An Efficient Group Key Management Scheme for Mobile Ad Hoc Networks. Accepted to appear in International Journal of Security and Networks (IJSN).
[15]. P. Judge and M. Ammar, "Security issues and solutions in multicast content distribution: a survey, "Network, IEEE, vol. 17, pp. 30-36, 2003.
[16]. Y. Challal and H. Seba, "Group Key Management Protocols: A Novel Taxonomy. ," Enformatika, International Journal of Information technology, vol. 2, 2005.
[17]. 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.
[18]. 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.
[19]. 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.
[20]. C. Perkins, "RFC3344: IP Mobility Support for IPv4,". IETF RFC. Status: Proposed Standard. August 2002.
[21]. R. Koodli, "RFC5568: Fast Handovers for Mobile IPv6," IETF RFC.Status: Proposed Standard. July 2009.
[22]. D. Johnson, C. Perkins, and J. Arkko, " RFC3775: Mobility Support in IPv6," IETF RFC. Status: Proposed Standard. June 2004.
[23]. T. Hardjono and L. R. Dondeti, Multicast and Group Security : Artech House, 2003.
[24]. 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.
[25]. 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.
[26]. K. Brown and S. Singh, "RelM:Reliable Multicast for Mobile Networks," Computer Communications, vol. 21, pp. 1379-1400, 1998.
[27]. 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.
[28]. C. K. Wong, M. Gouda, and S. S. Lam, "Secure Group Communications Using Key Graphs," ed. ACM SIGCOMM, 1998.
[29]. D. A. McGrew and A. T. Sherman, " Key Establishement in Large Dynamic Groups using One-way Function Trees," Technical Report TR-0755,May 1998.
[30]. 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.

© 2014- 15, IJIRAE- All Rights Reserved