Host Mobility Protocol for Secure Group Communication in Wireless Mobile Environments

Miss Laiha Mat Kiah
Faculty of Computer Science & Information Technology
University of Malaya
Kuala Lumpur, Malaysia.
misslaiha@um.edu.my

Keith M. Martin
Information Security Group, Department of Mathematics
Royal Holloway, University of London
United Kingdom
keith.martin@rhul.ac.uk

Abstract

Secure group communication allows a set of nodes (or devices) to communicate securely amongst each other over unprotected and open networks. Provision of security for group communication is based on cryptographic services, which relies on careful management of cryptographic keying material. Securing group communication in wired networks is fairly well understood, however wireless networks introduce further challenges as group members may move from one place to another while still remaining in a group session. In this paper we propose a host mobility protocol to govern group member movement in wireless mobile environments. We introduce the use of lists as part of our protocol design in order to facilitate host mobility.

1. Introduction

The advance in Internet technology, in particular *multicast* functionality [8] (which efficiently enables group communication), has increased the demand and popularity of group-based applications such as multimedia conferencing, news or stock updates and virtual classrooms.

Both end-users and content (or service) providers have similar security expectations for multicast applications as in traditional point-to-point (*unicast*) applications. Thus a vital component of any security architecture for group communication is the design of a *group key management framework* (GKMF), within which to govern the management keying material.

Previous GKMF proposals, such as those in [1][2][5][7][10][11][13][14][19][22][24] were not designed with wireless mobile environments explicitly in mind. While some efforts have been made to extend GKMFs to mobile environments, such as [4][6][8][20][21], most do not explicitly address host mobility issues. Providing mechanisms to address this specific problem is fundamental if secure group communication is to be deployed in wireless mobile environments. In this paper we propose and analyze a host mobility protocol.

The rest of the paper is organized as follows. Section 2 describes the main components of a GKMF. Section 3 identifies specific issues that need to be addressed in a GKMF for wireless mobile environments. In the following two sections we present our proposal for supporting host mobility. Finally, we conclude in Section 6.

2. Group key management frameworks

A GKMF is an infrastructure comprising the basic entities and functions necessary to provide common cryptographic keys to all communicating entities (group members) in a network supporting group (multicast) communication. The main objectives for group communications are similar to traditional unicast communications, namely the provision of *confidentiality, integrity* and *authentication* security services. A GKMF provides all the management tasks to maintain and protect the keys required to implement all of these services.

The main components of a GKMF can be divided into two parts, each of which is discussed in the following subsections.

2.1. GKMF architecture

The GKMF architecture specifies:

- the main entities within the GKMF, which typically consist of key managers, group managers and group members;
- the placement of these entities within the architecture;
- the trust relationships between the entities;
- the (types of) cryptographic key used within the architecture.

2.2. Key management processes

The essential processes identified within a GKMF for secure group communication are as follows:

- *Formation of multicast group*. This includes *group creation* and *initial registration of group members*. Group creation can consists of a host sending a request to the network via the Internet Group Management Protocol (IGMP)[7][23]. Registration of group members is typically performed by a host sending a *join* request to a group manager (perhaps asking for a specific internet service). At this point all the information related to a group, such as group membership policy and the required cryptographic keys, is determined.
- *Generation and distribution of cryptographic keys*. Cryptographic keys can be symmetric, asymmetric or a combination of both, depending on the security objectives or preferences of particular applications. Most GKMFs employ symmetric keys because symmetric algorithms have lower computational complexity and are faster than asymmetric algorithms [9][15]. Typical key types are:
  - *Individual keys*. Also referred to as *long-term* keys, these keys are unique to every host (potential group member) and are typically generated by, and shared with, a key or group manager. These keys are established prior to the group creation.
  - *Group keys*. Also referred to as *traffic encryption keys* (TEKs), group keys are generated by a key manager, and shared by all group members of a multicast group. Primarily used for securing the actual data communication, group keys are usually distributed to every member of a multicast group under the protection of individual keys.

An auxiliary key may be needed for the secure and efficient distribution of a group key to the group members [11][17][24]. Instead of having to send the group key separately under the protection of individual keys of group members, it can be sent once via a multicast message protected under the auxiliary key.

Where asymmetric cryptography is used, all entities involved in the group communication are assigned asymmetric key pairs [1][11][12].

- *New member joins*. Similarly to initial registration, any host who wishes to join a group will need to send a *join* request message to a governing entity such as a group manager. If the member is granted permission to join the group then relevant keys need to be delivered to the newly joined member.
  Provision of *backward secrecy* [11][19] may require re-keying of cryptographic keys whenever a new member joins a group in order to control access to the previous group traffic from the new member. Re-keying will result in all group members including the newly joined member obtaining a new group key.
- *Existing member leaves*. This process is initiated by sending a *leave* request message to a governing entity such as a group manager. If provision of *forward secrecy* [11][19] is required then re-keying will need to occur in order to update the group with a new set of group keys, and to control access of future group traffic from the leaving member.
  Unlike member joins, member leaves can be voluntary or non-voluntary. While the former occurs at the request of a group member, the latter can occur accidently (such as when a member is disconnected), or when a member is evicted from a group. An eviction of a group member usually requires re-keying to occur.
- *Re-keying*. This may occur due to group membership change (new joins for backward secrecy, and member leaves for forward secrecy), re-keying may also occur due to:
  - *Periodic re-keying*. A pre-determined plan to re-key a multicast group after a certain interval (which is often dictated by a group policy, as well as security requirements of a particular application).
  - *Expiration of cryptographic keys*. When a key has reached the end of its validity period.
  - *Compromised keys*. When a key used is believed (or suspected) to have been compromised and is no longer considered safe to use.

Re-keying events are normally initiated by governing entities such as group or key managers.

3. GKMFs for Wireless Mobile Environments (WMobEs)

The GKMF components identified in the previous section are generic to any networking environment. Wireless mobile environments have several special characteristics that need to be taken into account when designing a suitable GKMF. The two most important are identified in the following subsections. These have been incorporated into a generic GKMF model for wireless mobile environments [18].

3.1. New reasons for *joins* and *leaves*

As well as the reasons discussed in Section 2.2, group members may join and leave groups as they move between areas, while still remaining in a group session. The process of a member moving to another area can be treated as a *leave* from one area followed by a *join* to another. Moving member will need to notify a key manager, prior to moving. Such a move may require the provision of backward and/or forward secrecy (since different areas may have their own security requirements). Thus a specific protocol is required to govern host mobility.

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:

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

4. A GKMF supporting host mobility

In the next two sections we propose a method of supporting host mobility group communication in wireless mobile environments. In this section we briefly outline the GKMF and in the next section we provide details of the host mobility protocol.

4.1. Scope of proposal

It is important to note the following:

- *Infrastructure-based environment*. The framework for the proposed protocol relies on an infrastructure based environment with a basic underlying cellular architecture [3][16][20] as its networking platform.
- *Key distributions and key updates*. The aspects of key management that the protocol is primarily concerned with are *key distribution* and *key updates (or, re-keying)* which may occur during host mobility. Other aspects of key management such as generation, registration as well as deletion of keys are implicitly assumed to be available.
- *Simplified protocol descriptions*. Our protocol descriptions will be simplified in order to highlight how the keys in the GKMF are utilized to provide confidentiality services. Provisions of other security services (such as data integrity and message authentication) are implicitly assumed.

4.2. Main architecture

We adopt the general cellular architecture based on the notion of domain(s) and areas as the basic framework [11][18]. The reason for adopting logically or physically defined domains and areas is to provide a means to have an administratively manageable environment for group communication to take place within.

- **Main entities**. The main entities involved are:

  - *Domain key manager (DKM)*: responsible for generating, distributing, storing and deleting all keying materials that may be required at the domain level, as well as playing the role of group controller, which includes managing group policies, group membership, re-keying events and security policies.
  - *Area key manager (AKM)*: one per area, operating under the DKM's jurisdiction and responsible for key management within the area, including for group members residing within that area.
  - *Group member(s)*: senders and receivers, defined to reside within one area at any given time.
- **Placement of entities**. Figure 1 shows placement of entities in domain *i* and area *j*, with DKM as the main key manager of domain *i* and AKM as the key manager of the area *j*.

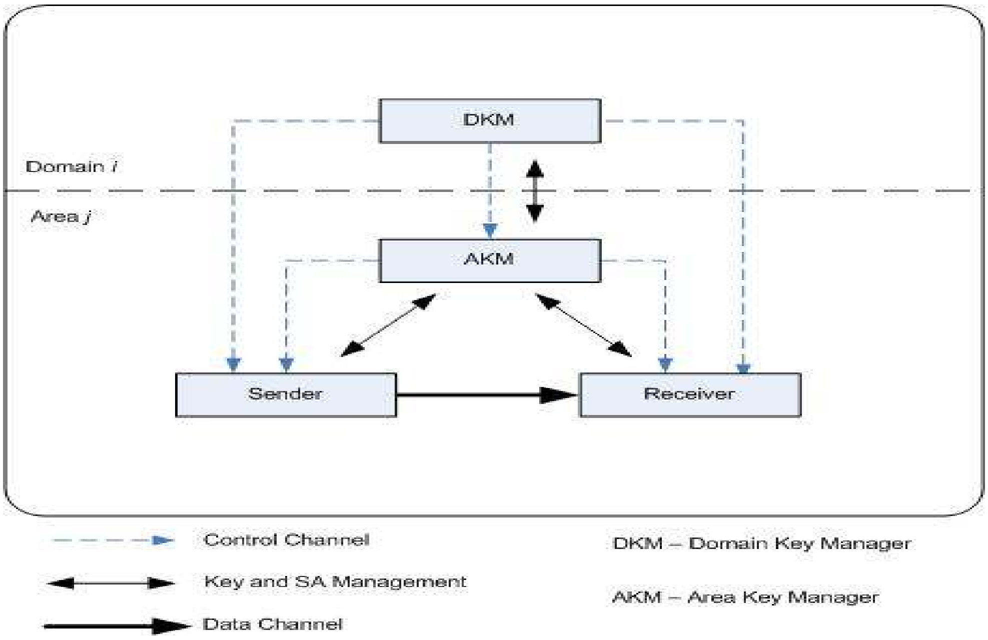

Figure 1. Placement of entities in domain *i* and area *j*.

Figure 2 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.

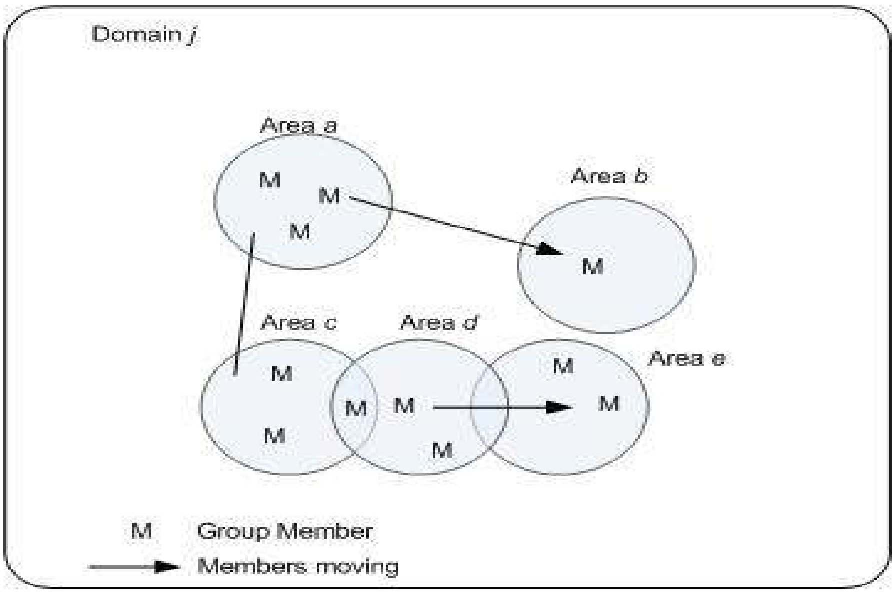

Figure 2. Placement of group members throughout areas.

- **Trust relationships**. We assume that all key managers (DKMs and AKMs) are trustworthy and reliable. All group members trust these key managers. There are two levels of trust relationships:
  - *At the domain level*. All AKMs trust the DKM as the primary key distributor, as well as the main group manager for various groups operating in that domain.
  - *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.
- **Types of key**. The symmetric keys used are:
  - *Domain-Area Key, DA_i_Key*. A unique long-term key shared between the DKM and AKM_i of area *i*. Generated by the DKM, the function of each Domain-Area key is restricted to unicast communication between the DKM and a particular AKM.
  - *Area-Member Key, A_iM_Key*. A unique key shared between AKM_i and group member *M*. Generated by the AKM, the function of each Area-Member key is restricted to unicast communication between AKM and the group member.
  - *Area Key, A_Key*. A group key is unique to an area. Generated by each AKM, the main purpose of having an area key is for managing host mobility and for efficient and scalable re-keying.

4.3. Host mobility protocol functionality

Since a moving member may accumulate information for each area it visits, provision of backward secrecy is necessary for controlling access to an area’s past security information (which could be used for malicious purposes). This requires re-keying to occur whenever host mobility occurs. Thus the main functional requirements of our host mobility protocol are to:

- transfer a group member from one area to another area;

- initiate a re-keying of an area key of a visiting area;
- deliver a new area key of a visited area to a moving member and to group members residing in that area.

The main security requirements of the protocol are to:

- ensure that only transfers from authorized group members are processed;
- secure communications between the group member and the area key manager;
- secure communications between the area key manager and the domain key manager;
- protect the distribution of area key of the visited area to the moving member.

4.4. List(s) management

In this section, we introduce an important concept that we will use as part of our protocol design. As group members may frequently move between numbers of areas while still remaining in a group session, every time a member moves, re-keying of an area key may need to occur.

As frequent re-keying may cause disruption of group communication, it may be necessary to keep track of the mobility of a highly dynamic group member. This can be useful to avoid frequent re-keying of an area key. To facilitate this, we propose the use of a list referred to as a *mobility list* (*MobList*). This is securely maintained by key managers (DKM and AKMs) in a domain and contains information on group members that move from one area to another (and also indicates how many area keys the member possesses). Each time a member moves from one area to the next, the following information is logged in *MobList*:

- ID of the moving member,
- ID of the multicast group joined by the member,
- ID of the area that a member is moving from,
- ID of the visited area that a member is moving to.

MobList can be used to keep track of host mobility and frequent re-keying can be avoided every time a member moves back into an area that it recently visited. This is because when the same member moves back into that area, the AKM of the visited area can determine (by looking up its *MobList*) whether the member is a returning member who is just moving back into the area, in which case re-keying of the area's key may not need to take place.

5. Host mobility protocol

We now specify our host mobility protocol.

5.1. Notation

We use the following notation:

- DKM. Domain key manager.
- ID_D. Identity (ID) of DKM.
- AKM. Area key manager.
- AKM_i. AKM of area *i*.
- ID_{Ai} . ID of AKM_i.
- M_i. Group member of an area *i*.
- ID_{Mi} . ID of M_i.
- ID_G. ID of a multicast group G.
- DA_i_Key. Domain-Area key between DKM and AKM_i.

- A_iM_i*_Key*. Area-Member key between AKM$_i$ and M_i.
- *A_Key$_i$*. Area key of area *i*.
- *A_Key$_{inew}$*. New area key of area *i*.
- *Sm_Key$_{ji}$*. Session mobility key between AKM$_i$ and M_j.
- ||. Concatenation operator.
- $\{m\}_k$. Encryption of message (or data) *m* with a symmetric algorithm using the key *k*.
- *text*. A field in the message content which may contain optional information.
- *a* →*b*. *Unicast* transmission from entity *a* to entity *b*.
- *Mov_Token$_D$*. A *move token* from DKM →AKM containing security parameters associated with a multicast group during *host mobility*.
- *Mov_Token$_A$*. A *move token* from AKM →*M* containing security parameters associated with a multicast group during *host mobility*.

5.2. Important assumptions

For ease of design, we make the following assumptions:

- Availability of secure encryption algorithms.
- Implicit use of secure entity and data origin authentication mechanisms such as use of message authentication codes (MACs).
- Symmetric keys specified in Section 4.2 are assumed to have been established securely prior to the commencement of the host mobility protocol.
- Use of some form of *time variant parameter* such as a time stamp in the *text* field within protocol messages for freshness checking.
- The membership of all key managers (DKM and AKMs) in a domain is predetermined and fixed. Thus, each Domain-Area key and the Domain key are static and valid until the policy determines otherwise.
- Availability of secure storage of cryptographic keys for all group communication entities.
- Availability of secure mechanisms for managing the *MobList*.

5.3. The protocol

This protocol describes transfer of a group member from one area to another with consideration to secure access to previous keys and group data traffic (backward secrecy). The protocol also includes the delivery of a new area key *A_Key$_{vnew}$* to the moving member, as well as to the group members residing in that area.

Throughout this protocol, we make the following assumptions:

- An established multicast group has already been created.
- Moving member M_i (currently in area *i*) and AKM$_v$ (in the visited area *v*) have securely established a shared short-term session mobility key *Sm_Key$_{iv}$* prior to moving.

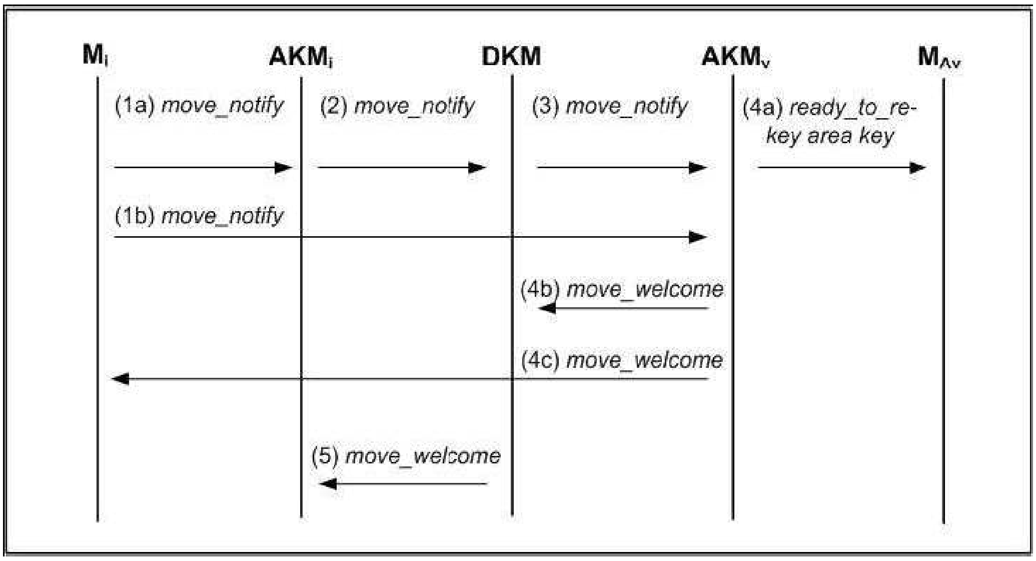

Figure 3. Member Moving with Backward Secrecy protocol message flow.

The message flow of this protocol is depicted in *Figure 3*. The steps involved are described as follows:

1. A group member M_i that wishes to move into another area sends a *move_notify* message along with the ID of the area that he is moving into (ID_{Av}) to:

   (a) its current area key manager AKM$_i$ protected under Area-Member key $A_iM_i\_Key$:

   $$M_i \rightarrow AKM_i : ID_{Mi} || \{ID_G||ID_{Ai}||ID_{Av}||ID_{Mi}||text\}A_iM_i\_Key.$$

   (b) the area key manager of the visited area AKM$_v$ protected under a session mobility key Sm_Key_{iv}:

   $$M_i \rightarrow AKM_v : ID_{Mi}||\{ID_G||ID_{Ai}||ID_{Av}||ID_{Mi}||text\}Sm\_Key_{iv}.$$

2. Upon receiving the *move_notify* message from M_i, AKM$_i$ checks the message by decrypting it with $A_iM_i\_Key$ and passes the message to DKM protected under Domain-Area key DA_i_Key:

   $$AKM_i \rightarrow DKM : ID_{Ai}||\{ID_G||ID_{Ai}||ID_{Av}||ID_{Mi}||text\}DA_i\_Key.$$

3. Upon receiving the message from AKM$_i$, DKM checks the message by decrypting it with DA_i_Key, and sends the *move_notify* message to AKM$_v$ along with the ID of M_i in the form of Mov_Token, where Mov_Token_D = $\{ID_G||ID_{Ai}||ID_{Mi}||ID_{Av}||text\}$ protected under the Domain-Area key DA_v_Key it shares with AKM$_v$:

   $$DKM \rightarrow AKM_v : ID_D||\{Mov\_Token_D||text\}DA_v\_Key.$$

4. Upon receiving the *move_notify* message from DKM and M_i, AKM$_v$ does the following:

   (a) checks the message from DKM by decrypting it with the Domain-Area key DA_v*_Key* it shares with the DKM;
   (b) checks the message from M_i by decrypting it with the session mobility key *Sm_Key*$_{iv}$ it shares with M_i;
   (c) assuming that the checking is valid, AKM$_v$ looks up its MobList$_v$ and if M_i is not in the list (meaning that this is M_i's first time to enter the area), AKM$_v$ must re-key its area key *A_Key*$_v$. To do so, AKM$_v$ generates a new area key and sends it via the *ready_to_rekey* message. This results in all group members MA_v in that particular area obtaining the new area key *A_Key*$_{vnew}$.
   (d) AKM$_v$ sends the new area key *A_Key*$_{vnew}$ via a *move_welcome* message in the form of *Mov_Token*$_A$ = $\{ID_G||ID_{Ai}||ID_{Mi}||ID_{Av}||A\_Key_{vnew}||text\}$, to:
       - M_i, protected under the session key *Sm_Key*$_{iv}$;
       - DKM, protected under the Domain-Area key DA_v*_Key*.
   (e) If M_i is already on MobList$_v$, AKM$_v$ will need to check whether there has been any re-keying of its area key since M_i's last visit to the area. If none, AKM$_v$ sends a *move_welcome* message to M_i in the form of *Mov_Token*, along with its current area key *A_Key*$_v$ (as in *Step 4(d)*).
   Otherwise, AKM$_v$ sends an updated area key *A_Key*$_{vnew}$ to M_i.

5. Upon receiving the *move_welcome* message from AKM$_v$, DKM informs AKM$_i$ of the successful move of member M_i via a *move_welcome* message, protected under a *Domain-Area* key.

   Note that re-keying of the area key due to host mobility only needs to occur within the visited area (which the member is moving to) and does not affect other areas.

Having the new information concerning member M_i, DKM and AKMs (AKM$_i$ and AKM$_v$) will need to update their *MobList* in order to keep track of member M_i's mobility, along with the number of area keys that may have been kept by M_i.

In cases where provision of backward secrecy is not necessary, no update of keying material will need to occur during host mobility, and a moving member is given the same key(s) currently in use in the visited area.

5.4. Analysis of protocol

In this section, we provide a basic analysis of the proposed protocol.

- A member M_i who wishes to move into another area must first establish a short-term session mobility key with the AKM of the visited area, and we have assumed that this was done securely (see Section 5.3).
- After obtaining the session mobility key, M_i initiates the *move* protocol by sending a *move_notify* message to its local area key manager AKM$_i$, protected under the Area-Member key, and to the visited area key manager AKM$_v$, protected under the session mobility key. If an adversary gets hold of the enciphered messages between the entities, he has no way of deciphering the message as he has no access to either of the keys (Area-Member key or session mobility key).

- A member M_i uses the session mobility key to secure communications with the AKM$_v$. If an adversary wants to masquerade as some moving member in order to get hold of the area key of the visited area, he will not be able to do so because he has no access to the session mobility key shared only between the moving member and the AKM$_v$.
- We have implicitly assumed the provision of data origin authentication (such as using MACs). Thus, we can conclude that if an adversary wants to masquerade as some moving member in order to get hold of A_Key_v or A_Key_{vnew}, the adversary will not able to do so because he has no access to the MAC key. Other entities (DKM, AKM and M_i) can easily check the integrity of messages received via the same process.
- After receiving the *move_notify* message from AKM$_i$, DKM notifies AKM$_v$ of the move, protected under the Domain-Area key. Similarly, if an adversary gets hold of the enciphered message between DKM and AKM$_v$, he has no way of deciphering the message as he has no access to the Domain-Area key.
- After receiving the *move_notify* message from DKM and M_i (and if M_i is not in the *MobList*), AKM$_v$ initiates the re-keying of its area key A_Key_v. This results in all members residing in the visited area, including the moving member M_i, obtaining the new area key A_Key_{vnew}. AKM$_v$ can send this key to group members (excluding M_i) in the area either via multicast, protected under the old area key A_Key_v, or via unicast, protected under the Area-Member keys. AKM$_v$ sends A_Key_{vnew} to M_i, protected under the session mobility key. If an adversary wants to get hold of A_Key_{vnew}, he will not be able to do so because he has no access to the keys (A_Key_v, Area-Member keys, or session mobility key).
- We have implicitly assumed the provision of data origin authentication, so we can conclude that if an adversary wants to masquerade as some moving member in order to get hold of A_Key_{vnew}, the adversary will not be able to do so as he has no access to the MAC keys.
- On a member's first move into an area, the area needs to be re-keyed with a new area key, and information about the moving member is logged in *MobList*. If it is necessary to control the number of area keys that are kept by a group member (which corresponds to the number of areas that he visited), *MobList* may need to be reset for that particular member after a period of time, for example when the number of area keys collected by a member (as he moves from one area to another) has reached a threshold limit. In this case, re-keying of the area key may need to occur when the member moves into an area. This will be determined by the group security policy at the creation of a multicast group, prior to the commencement of group communication. This is useful to avoid a group member moving from one area to another with intent to collect all the area keys. If colluding members want to exchange security information, such as area keys, to gain unauthorized access to different areas, this could also be prevented (periodically).
- All affected key managers (DKM, AKM$_i$ and AKM$_v$) update their *MobList*, and area(s) visited are logged. We assume that these lists are maintained and kept securely by the key managers.

6. Conclusion

We have proposed a protocol for facilitating member moves with consideration for backward secrecy. Using the protocol, a group member M_i moves from an area managed by an area key manager AKM$_i$ (where it is currently residing), to another area managed by

AKM$_v$, while still remaining in the group session. The *move* is managed by the DKM via AKM$_i$. For provision of backward secrecy, when a member moves to a visited area, the area key of visited area needs to be re-keyed. This results in group members (including the moving member) residing in that particular area obtaining a new area key. All affected key managers during host mobility (such as DKM, AKM$_i$ and AKM$_v$) need to update their *MobList*, and new information regarding the moving member is logged into the lists.

The proposed protocol features a mechanism (MobList) that allows for efficient processing of members who are returning to recently visited areas during host mobility.

7. References

[1] M. Baugher, R. Canetti, L. Dondeti, and F. Lindholm. *Group Key Management Architecture*. Internet Draft IETF MSEC WG, 2003.http://www2.tools.ietf.org/html/draft-ietf-msec-gkmarch-04.

[2] M. Baugher, R. Canetti, L. Dondeti, and F. Lindholm. *Multicast Security (MSEC) Group Key Management Architecture*. RFC 4046, 2005.

[3] B. Bhargava, S. B. Kamisety, and S. K. Madria. Fault-tolerant authentication and group key management in mobile computing. Technical report, Center for Education and Research in Information Assurance and Security, and Department of Computer Science Purdue University, 2000. http://www.cs.purdue.edu/homes/bb/cs690b/report.ps.

[4] D. Bruschi and E. Rosti. Secure multicast in wireless networks of mobile hosts: Protocols and issues. *Mobile Networks and Applications*, 7(6):503–511, Dec 2002.

[5] R. Canetti, J. Garay, G. Itkis, D. Micciancio, M. Naor, and B. Pinkas. Multicast security: A taxonomy and some efficient constructions. In *Proceeding of IEEE Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOMM)'99*, 1999. http://citeseer.ist.psu.edu/canetti99multicast.html.

[6] B. Decleene, L. Dondeti, S. Griffin, T. Hardjono, D. Kiwior, J. Kurose, D. Towsley, S. Vasudevan, and C. Zhang. Secure group communications for wireless networks. In *Proceedings of IEEE MILCOM'01*, pages 66–73, 2001.

[7] S. Deering. *Host extensions for IP multicasting*. RFC 1112, 1989.

[8] L. Gong and N. Shacham. Multicast security and its extension to a mobile environment. *Wireless Networks*, 1(3):281–295, 1995.

[9] R. A. Gove. Fundamentals of cryptography and encryption. In *Information Security Management Handbook, 4th Edition by H. F. Tipton and M. Krause*. Auerbach, 2000.

[10] T. Hardjono, B. Cain, and N. Doraswamy. *A Framework for Group Key Management for Multicast Security*. Internet Draft IETF, 2000. http://www3.ietf.org/proceedings/00jul/I-D/ipsec-gkmframework-02.txt.

[11] T. Hardjono, B. Cain, and I. Monga. *Intra-Domain Group Key Management Protocol*. Internet Draft IETF, 2000. http://www.securemulticast.org/draft-ietf-ipsec-intragkm-03.txt.

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

[13] T. Hardjono and G. Tsudik. IP Multicast Security: Issues and Directions. *Annales de Telecom*, pages 324–340, 2000. http://citeseer.ist.psu.edu/hardjono99ip.html.

[14] H. Harney and C. Muckenhirn. *Group Key Management Protocol (GKMP) specification*. RFC 2093, 1997.

[15] J. Ikbal. An introduction to cryptography. In *Information Security Management Handbook, 4th Edition by H. F. Tipton and M. Krause*. Auerbach, 2003.

[16] Y. Lin and I. Chlamtac. *Wireless and Mobile Network Architectures*. John Wiley & Sons, Inc, 2001.

[17] M. L. MatKiah and K. M. Martin. Group communication: Design challenges in the development of key management frameworks in wireless mobile environments. In *Proceedings of International Conference on Security and Management SAM'05*, pages 385–390. CSREA Press, 2005.

[18] M. L. MatKiah and K. M. Martin. A generic group key management framework for group communication in wireless mobile environments. In *Proceedings of the Sixth International Network Conference INC2006*, pages 347–354. University of Plymouth, 2006.

[19] S. Mittra. Iolus: A framework for scalable secure multicasting. In *Proceedings of ACM SIGCOMM*, pages 277–288, Cannes, France, 1997.

[20] J. Park, Y. Suh, and S. Kang. Supporting mobile multicast in mobile networks by considering host mobility. In *IDMS/PROMS 2002:Proceedings of the Joint International Workshops on Interactive Distributed Multimedia Systems and Protocols for Multimedia Systems*, pages 263–273. Springer-Verlag, 2002.

[21] R. Shankaran, V. Varadharajan, and M. Hitchens. Secure multicast extensions for mobile networks. In *LCN '99: Proceedings of the 24th Annual IEEE Conference on Local Computer Networks*, page 106. IEEE Computer Society, 1999.

[22] D. Wallner, E. Harder, and R. Agee. *Key Management for Multicast: Issues and Architectures*. RFC2627, 1999.

[23] B. Williamson. *Developing IP Multicast Networks*. Cisco Press, 2000. Chapter 43.

[24] C. K. Wong, M. G. Gouda, and S. S. Lam. Secure group communications using key graphs. In *Proceedings of the ACM SIGCOMM'98 conference on Applications, technologies, architectures, and protocols for computer communication*, pages 68–79. ACM Press, 1998. http://citeseer.ist.psu.edu/article/wong98secure.html.

Authors

Dr. Miss Laiha Mat Kiah BSc. Comp. Sc. (Hons) (Malaya), MSc (London) PhD (London), joined the Faculty of Computer Science & Information Technology, University of Malaya, Malaysia as a tutor in 1997. She was appointed as a lecturer in 2001. She received her BSc. (Hons) in Computer Science from the University of Malaya in 1997, a MSc from Royal Holloway, University of London UK in 1998 and a PhD also from Royal Holloway, University of London in 2007. Between 1999 and 2003 before pursuing her study, she was primarily involved in academic teaching and research in University of Malaya. Her current research interests include key management, group communication and wireless mobile security. She is also interested in routing protocols.

Prof Keith M. Martin B.Sc. (Glasgow), PhD (London), CMath FIMA joined the Information Security Group, Department of Mathematics Royal Holloway, University of London UK as a lecturer in January 2000. He received his BSc (Hons) in Mathematics from the University of Glasgow in 1988 and a PhD from Royal Holloway in 1991. Between 1992 and 1996 he held a Research Fellowship in the Department of Pure Mathematics at the University of Adelaide, investigating mathematical modeling of cryptographic key distribution problems. In 1996 he joined the COSIC research group of the Katholieke Universiteit Leuven in Belgium where he was primarily involved in an EU ACTS project concerning security for third generation mobile communications. He has also held visiting positions at the University of Wollongong, University of Adelaide and Macquarie University. Keith's current research interests include cryptography, key management and wireless sensor network security. Keith is also interested in e-learning and is a co-developer of the distance learning MSc Information Security.