Secure Multicast Group Communication Scheme in Wireless IPv6 Networks

Abbas Mehdizadeh[1],
Fazirulhisyam Hashim[2], and **Raja S. Azmir Raja Abdullah**[3], Non-members

ABSTRACT

Key management is one of the challenging issues in group communications. It is generally used to secure multicast data transmission as well as preventing potential eavesdropping by malicious attackers. Group security key should be maintained for data encryption, while group key update and dissemination processes are required when a new user joins or leaves the group, which eventually lead to high communication and computation cost. Since eavesdrop activities can be initiated by capturing the disseminated keys, higher communication and computation cost due to frequent updates also increase the possibility of attack of multicast transmission. In this paper, a key management scheme for IPv6 networks is proposed to reduce communication and computation cost and therefore, fewer security risks. The obtained results from test-bed implementation show the efficiency of proposed scheme in terms of communication and computation cost, number of updated paths and security index due to key updating, while at the same time achieving both forward and backward secrecy.

Keywords: Group Communication Security, Key Management, Wireless IPv6 Test-bed, Path Probability of Attack

1. INTRODUCTION

Multicast is an efficient way to distribute data from one sender or multiple senders to multiple receivers. It is sometimes called as one-to-many or many-to-many communications [1–3].

One of the main challenges in multicast networks is to protect data which is multicast to many participants. In general, the security method should provide *authenticity*, *confidentiality* and *data integrity.* To achieve this, several key management protocols have been proposed to generate, distribute and update the key for data encryption. A common design for key management protocol is to support the security of network, transport and application layer [4]. When a change in group membership occurs, the new key needs to be generated and then sent to all members of the group [5, 6]. It means one user/member can affect all members to be updated, which is referred to *1-affect-n* problem. When a user joins the group, it must be prevented to access the past data which is called *backward security.* Similarly when a member leaves the group, it must be prevented to access the further data transmitted to the group, called *forward security* [1, 7, 8].

Group key management protocols can be divided into three categories, namely, *centralized*, *decentralized*, and *distributed.* In centralized schemes, a Group Controller (GC) or server handles the key generation, distribution and update process [1, 9, 10]. In distributed schemes, these actions are taken by each member and server. In decentralized schemes [11], one large group is divided into some subgroups with separate key management services. The advantages and disadvantages of the existing methods will be discussed in the next section.

Some of the existing methods mentioned above are based on key tree graph. The server can use logical key tree graph to handle key management procedure to achieve lower rekeying cost. Key management protocols based on the hierarchical tree significantly reduce the communication and computation cost [12, 13] which was proposed in [9, 14] and later was enhanced in [15–18]. The communication and computation cost of key updating is proportioned logarithmically to the size of the group [2].

In this paper, a decentralized multicast-unicast key management scheme is proposed to reduce the security risk, communication and computation cost of key updating in IPv6 networks while at the same time providing forward and backward secrecy. Specifically, we extend our previous work in [19], and consider the different types of DoS attacks, and improvement of the framework of the proposed scheme. In our proposed scheme, the key update process of multicast network is divided into two levels: multicast level from the server to the Access Points (APs) and unicast level from APs to the end users. Here, each subgroup under the AP (i.e., unicast mode) is responsible for key management including distribution and

Manuscript received on November 22, 2012 ; revised on March 21, 2013.

Final manuscript received October 21, 2013.

[1] The author is with Department of Computing, Faculty of Engineering, Science and Technology, Nilai University, 71800 Nilai, Negeri Sembilan, Malaysia.E-mail: mehdizadeh@ieee.org

[2,3] The authors are with Department of Computer & Communication Systems Eng., Faculty of Eng., Universiti Putra Malaysia (UPM), 43400 Serdang, Selangor, Malaysia. E-mail: fazirul@upm.edu.my and rsa@upm.edu.my

update process. Whenever a user joins or leaves the group, the entire member of the group does not need to be updated, and therefore, the *1-affect-n* problem is prevented. Therefore, the communication and computation cost due to arrival and departure of users can be decreased since the key update process can be handled by the AP or locally within the subgroup. Our proposed scheme also solves the problem of central key server failure in centralized scheme and unsafe key derivation in distributed schemes. In addition to key management, the same multicast-unicast paradigm can also be used for multicast data delivery. In [20, 21], we have shown that such a multicast-unicast approach is able to increase the quality of multicast services is wireless IPv6 networks.

The rest of the paper is organized as follow. Section II provides an overview of several existing key management methods, specifically on their advantages and disadvantages. The multicast-unicast key management is explained in Section III. Finally, the test-bed implementation and performance evaluation of our proposed multicast-unicast scheme is discussed in Section IV, followed by some concluding remarks and future extensions of our work.

2. RELATED WORKS

A summary of existing key management protocols are described in this section.

Wallner *et al.* [9] proposed a Logical Key Hierarchy (LKH) for Group Key (GK), Key Encryption Key (KEK), and Individual Key (IK) of each user to provide secure key management scheme for multicast networks. A server constructs a virtual key tree and considers all members as leaves of tree. Each member holds all the keys up to the root of tree (server), which is the group key. The intermediate nodes of the tree are *KEKs*, which are used to encrypt the *IKs*. Fig 1 shows the construction of key tree from communication network of n users $(U_1...U_n)$. Whenever a new user join or leave the group, the server re-construct the tree and then updates the members with new keys to provide forward and backward secrecy. For example, if U_n joins or leaves the group, KEK_m and GK need to be changed and sent to the members who have access to these keys. This update process is called rekeying procedure. LKH reduces the number of required keys of rekeying, which depends on the height of tree (h) and the type of tree. The type of tree is considered as the number of children of each node (denoted as d) in the tree, hence, the tree is called *d-ary* tree where $d > 2$, and binary tree where $d = 2$. The communication cost of join and leave operations are $2*h$ and $d*h$, respectively, where $h = 1 + log_d^n$. Now we derive the computation cost, which refers to the number of key encryption (E), decryption (D), derivation (F), and random generation (G) in the key management procedure. For join operation in LKH, two encryptions for group key and individual key of the new member, and one key generation are required depend on the height of the logical tree. Therefore, the computation cost is $h(2E + G)$. But in leave operation, the d encryption is required, therefore the computation cost is $h(dE + G)$.

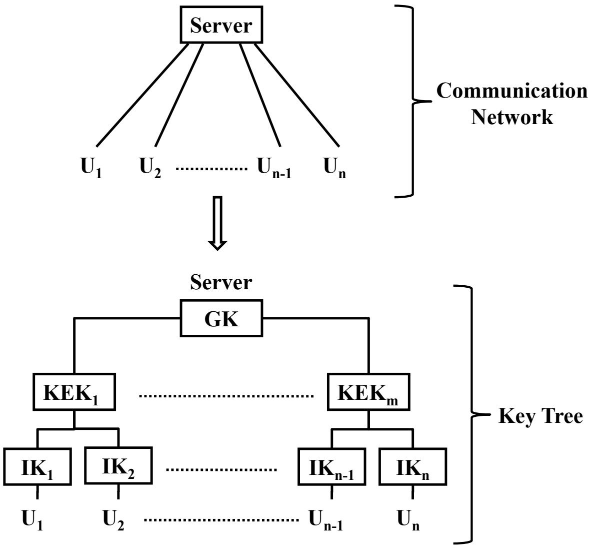


Fig.1: *Key tree construction from communication network of* n *users.*

In SKDC [1], a key server has built to implement and perform rekeying for join and leave procedures based on the key graphs. Specifically, a different key hierarchy is proposed to solve the scalability problem of large groups, and a user-key relation maintained by trusted server is introduced. Suppose there are nine users $(u1 - u9)$; three users in three subgroups $(sub1 - sub3)$. Three keys are given to each user, namely, individual key, group key, and subgroup key. When a user leaves the group (let say $sub1$, which now has two users), the server sends the new group key to the users of $sub1$ and $sub2$ by encrypting with the existing subgroup keys of $sub2$ and $sub3$. The users in $sub1$ receives the new subgroup key which is encrypted by individual key of each user, and new group key which is encrypted with new subgroup key. $d * log_d(n)$ encryption is needed by the server to perform rekeying procedure, where n is the number of members and is a power of d. In this method, a server construct rekey messages and distribute to the members based on the key graph. When join operation occurs, only 1 key is needed for new member, while in leave operation, $(n - 1)$ keys for $(n - 1)$ members are needed. In regards to computation cost, two encryptions for group key and individual key of the new member, and one key generation are required in join operation. However in leave operation, one key generation is done by the server with $(n-1)$ encryptions for the existing multicast members.

Recently, Shared Key Derivation (SKD) for group key management was proposed by Lin *et al.* [2], which can be classified as distributed scheme. Generally it is proposed to reduce the communication and computation cost of rekeying procedure. Members can derive the new key with shared key derivation and therefore, server does not need to generate and distribute the new keys when membership changes due to member join and leave. However, the communication cost for join and leave operations depends on the h and d, and equal to h and $(d-1)*h$, respectively. The existing cryptographic function is used for key derivation function, which has the following criteria: *i)* by giving derivation key, easy to compute key derivation function, but by giving key derivation function, it is impossible to compute derivation key. *ii)* by giving the derivation key of all members, if computing the derivation key is impossible, then computing the key derivation function is impossible. Therefore, it is impossible to predict the new key without derivation key. Strong encryption should be used to achieve a secure multicast group.

Sherman and McGrew [10] proposed a scalable centralized method based on One-way Function Trees (OFT) for large dynamic groups. The rekeying procedure is done from bottom to the up level of tree. The communication cost is reduced to $1+lg(n)$ (which is equal to h) for join and leave operations, where n is the number of members in the group. The key tree is binary in this method, which means each node of key tree has exactly two children. In addition, each node has two cryptographic value, namely, *node secret* and *node key*. Here, the cryptographic function has the same criteria as SKD. A node can compute the node key from the node secret, but by giving the node key, it is impossible to compute the node secret. The node secret is supposed to be distributed to the members during group initialization stage, which consist of three steps; *i)* establishment of shared key between server and members, *ii)* OFT key tree construction by server, and placing the members into the tree, *iii)* each member compute the group key by using node secret, which is encrypted by node key and broadcasted to all members by server. When a join or leave operation occur, the server broadcast the new node secret. Then all affected members (which are in the path between the join/leave member and the server) and server compute the new group key separately.

The communication cost of join and leave operations for the OFT are equal to the height of the tree. The computation cost also depend on the height of the tree and can be summarized as $h(2E+2F)+2G$ and $h(E+2F)$ for join and leave operation, respectively.

In summary, the centralized approaches suffer from circumventing a single point of vulnerability problem, as well as 1-affect-n problem. The decentralized approach (e.g., Iolus [11]) is not well-defined and it is difficult to implement in real network. In Iolus, the subgroups are connected to each other by group security intermediaries by bridging between them. The decryption and re-encryption happen when transmitted data pass through a subgroup. With increasing the number of subgroups connected to each other, many encryptions/decryptions are required and therefore, the computational cost increases. And finally, in distributed approaches, since the members are involved in the key derivation process, each member should have enough information to derive the new key. The key derivation is done using cryptographic function, and thus a high computation cost with high complexity is expected.

Therefore, we proposed a lightweight decentralized approach that solves the above mentioned problems experienced by the existing methods, reducing the communication and computation cost, as well as providing a secure multicast communication.

3. THE PROPOSED MULTICAST-UNICAST SCHEME

In this section, the theoretical framework of the proposed scheme is explained and to show the efficiency of our scheme, the following issues are outlined (which make our scheme more apparent); path probability of attacks, scalability, intra-domain mobility, preventing the inside-group attacks, and many-to-many consideration.

The proposed key management scheme is based on key generation, distribution and updating for multicast in wireless IPv6 networks. The aim is to reduce the communication and computation cost while increasing the security of multicast over wireless IPv6 networks. Whenever a join or leave occurs, only a subgroup is affected for updating. The following steps summarizes the theoretical framework of the join operation in our proposed method. The notation $A \rightarrow$ B(m) denotes A sends message m to B.

1. *New User → AP → Server (Join)*
2. *AP keeps the new user's join record*
3. *Server authenticates new user*
4. *Server informs AP*
5. *AP updates routing table by adding new user's MAC*
6. *AP → New user (Individual key)*
7. *AP → Subgroup users (New multicast key)*

Similar description applies to a leave operation. The following steps depict the leave operation of our proposed method.

1. *User → AP → Server (Leave)*
2. *AP keeps the user's leave record*
3. *Server → AP → User (Acceptance)*
4. *AP updates routing table by deleting user's MAC*
5. *AP → Existing users (New multicast key)*

We have separated the communication network into two levels, namely, multicast level from the server to lowest level (i.e., AP), and unicast level from AP

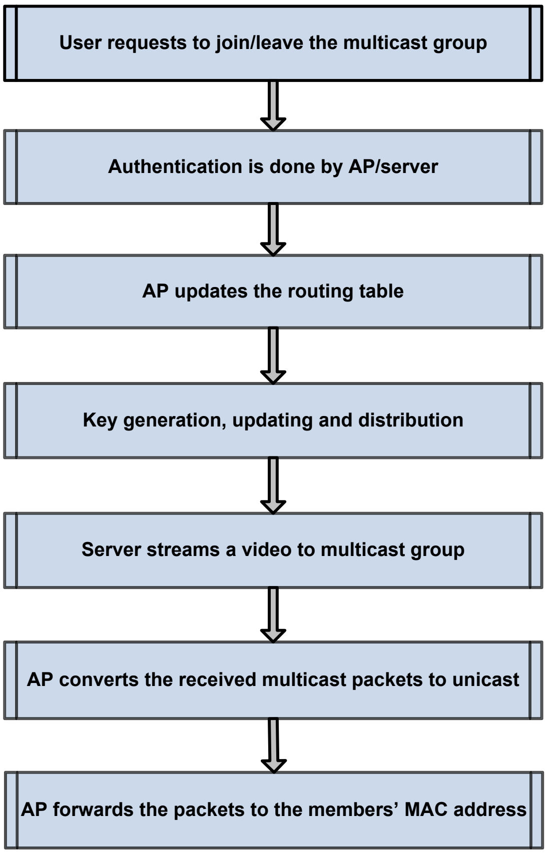

User requests to join/leave the multicast group
Authentication is done by AP/server
AP updates the routing table
Key generation, updating and distribution
Server streams a video to multicast group
AP converts the received multicast packets to unicast
AP forwards the packets to the members' MAC address

Fig.2: *Work Procedure of the Proposed Key Management Scheme.*

to the end users. In the proposed scheme, the whole group is divided to some subgroups which are managed with one manager. The managers that can be APs in wireless intra-domain networks are responsible for key updating and unicast data delivery to the end users instead of multicasting. Each AP, re-encrypt the packets and send to the MAC address of wireless members. Fig 2 shows the work procedure of the proposed key management and data delivery scheme. The reason of converting multicast packets to unicast MAC packets are; firstly, AP is layer 2 device which has access up to the second layer of OSI model (MAC layer), secondly, by changing the MAC address of multicast packets to unicast MAC address without changing the multicast IPv6 address, the end-to-end multicasting is achieved. It should be noted that our proposed scheme is viable with IPv4 as well. The implementation of IPv6 protocol is considered in this paper, which has been introduced to replace the conventional IPv4 [22]. Therefore, the characteristics of IPv6 in wireless network is taken into account by using IPv6 address for all members in our test-bed and transmission of IPv6 packets. Moreover, the IPv6 test-bed considers the multicast transmission of real-time traffic such as video application for evaluation of the proposed scheme (shown in Fig 2).

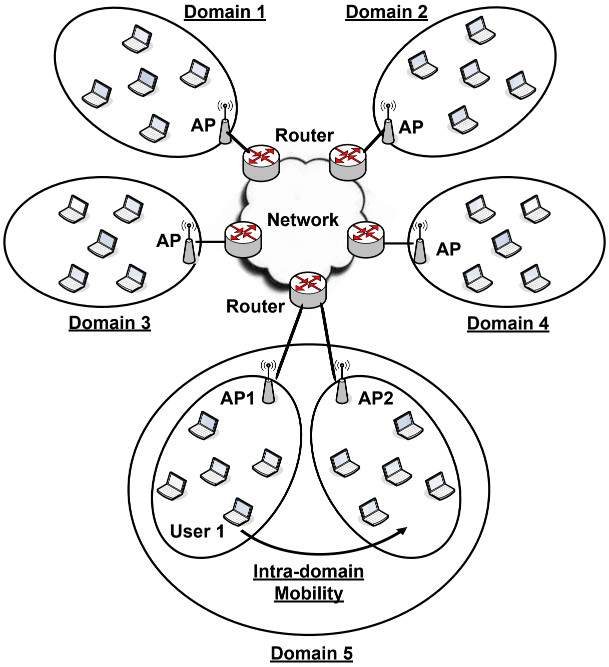


Fig.3: *An example of multicast network.*

By dividing the multicast network into two separated level, key updating is not needed for the whole group when a join and leave operation of node are occurred and therefore the number of key transmission, and required bandwidth are minimized. The novelty of our proposed method is the combination of multicast and unicast transmission and use in multicast networks. By integrating both multicast and unicast, the proposed scheme inherits their monumental advantages. The bandwidth advantage of multicast is used from the server to the AP, and cost-reduced and secured advantages of unicast transmission are used from the AP to the members in wireless networks.

In our proposed scheme, only valid members can access to the keys and data, only during the period of their memberships to the group, hence, the forward and backward secrecy are provided. In addition, the proposed scheme can protect the group keys from inside-group and outside-group attackers, which the existing methods did not consider it.

In the following we describe an example that can adopt multicast-unicast scheme to improve the security of multicast over Wireless Local Area Network (WLAN), while reducing the communication and computation cost of rekeying procedure. Suppose in a university, there are 4 faculties and each faculty has 4 departments. Each department has different labs and lab areas, where students and lecturers with mobile devices can connect to the network through APs. To support real-time lecturing or video-conferencing, a server is located at the university or outside which sends multicast streams to groups of users. When a mobile user wishes to join

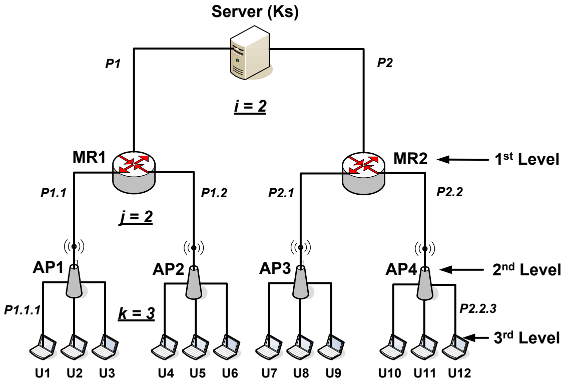


Fig.4: *An example of communication network.*

or leave the group, a rekeying procedure is needed to achieve forward and backward secrecy. To end that, the server needs to update all the security keys which lead to *1-affect-n* problem and therefore high communication and computation cost, and high security risk. We can divide a large group (entire university) to subgroups (each AP) based on the location area. Each AP is responsible for key updating, therefore, when a user join or leave the group, the server does not need to update all members by new key.

The example described is illustrated in Fig 3. There are 5 domains representing of faculties in a university. Each domain is connected to the network through AP and router. When a server (located at the network) sends a multicast stream, it goes to router and is forwarded to AP. Each AP handles the key management and rekeying procedure. Whenever a user (let say a user from domain 1) joins or leaves the group, the rekeying procedure is needed only for its domain.

3.1 Path Probability of Attack

Fig 4 illustrates a balanced communication network, where server connected to two multicast router (MR1 and MR2). Each router has two AP connection and 3 mobile users are connected to each AP. Balanced communication network is considered for simplicity, means that all the nodes in the same level have the same outgoing paths or connection. For example in 3^{rd} level, each AP has connection to 3 mobile users. Suppose a server key is K_s, when a join or leave occurs, the server has to send a new key (i.e., K_s') to the all users. It means all the paths are affected (by key updating) and the path probability of attack is equal to the number of all affected paths.

Three levels from the server to end users are considered here. Let us denote i, j, and k, as the number of outgoing paths of each node in 1^{st}, 2^{nd}, and 3^{rd} level, therefore, i=*2*, j=*2*, and k=*3* in this example. The number of paths in 1^{st}, 2^{nd}, and 3^{rd} levels are i, i*j, and i*j*k, respectively. For example *P1* and *P2* in 1^{st} level, and *P1.1*, *P1.2*, *P2.1*, and *P2.2* in 2^{nd} level, and so on. Let denote N_p as the total number of the paths in the network. Therefore, $N_p = i + (i * j) + (i * j * k) = i * (1 + j * (1 + k))$.

In this paper, we consider the probability of attack based on the number of the paths which needs to be updated. In the existing key tree-based approaches, when a join or leave operation occurs, all the paths are affected by new server key (i.e., *i+(1+j*(1+k))* paths). But in our proposed scheme, when a join or leave occurs, only k paths are affected. This is mainly due to dividing the network into two separate levels, namely, multicast level from server to AP, and unicast level from AP to end users.

We define two parameters to evaluate the path probability of attack, namely, Security Index (SI) and Risk Index (RI). By reducing the risk of attack, the security of the network is increased, therefore, $SI = 1/RI$. SI is calculated as follow: $SI = \rho/N_P$, where ρ is the probability of attack. By decreasing the number of updated paths (N_P), the SI is increased, meaning that our network is more secured. When there is definitely an attack, then $\rho = 1$, and when there is no attack, then $\rho = 0$.

3.2 Scalability

The proposed scheme can solve the scalability problem by setting a trust center in each subgroup. Trust center is an entity or device to be used for the implementation of the proposed scheme. This trust center can be implemented in AP, extra entity connected to AP, or in a higher level of communication network (e.g., faculty level or university level in our example, or in base station in cellular network).

In fact, the situation here is same as when there are many users try to connect to the AP using WiFi at the same time. Since the AP can handle only 30 users at particular time [23, 24], to avoid unsuccessful connection of users, the administrator should set and install more APs in that area to allow more users' connectivity.

3.3 Intra-domain Mobility

Suppose there are two intra-domains in one domain, (see domain 5 in Fig 3). AP1 and AP2 can communicate and update each other about their memberships. Whenever a user moves from one intra-domain to another (e.g., user1 moves from AP1 to AP2), there is no need for authentication again and rekeying update for leave and join operation since AP2 has the record of the user. The advantage of this issue is more apparent when user1 is traveling between AP1 and AP2 frequently. The user can join or rejoin the intra-domain very quickly without waiting for authentication each time, which led to less packet lost for real-time applications.

3.4 Inside Group Attacks

AP is able to detect Denial-of-Service (DoS) or IP-Spoofing attacks from any group members, and can simply prevent the forged user to launch attack by removing the MAC address of user from its routing table. The attacker cannot connect to another intra-domain where the intra-domain APs update each other and therefore, they have the record of the attackers. Refer to Fig 3, suppose user1 (in domain 5) connects to AP1 and launches attack. AP1 is able to detect and update AP2 (and other intra-domain if exist) about this attack. After detection, AP1 terminate the connection of attacker to prevent it from sending forged packets to the network. If attacker moves to AP2, AP2 will reject this user from registration to its group. As far as the attacker remains in this domain, it is prevented to access the network.

We show that how our proposed method can detect and prevent the different types of DoS attacks in Section 4.2.

4. EXPERIMENTAL RESULTS AND EVALUATION

The performance evaluation and feasibility of our scheme are provided by implementation of real test-bed. We setup a test-bed based on the communication network in Fig 4. Note that, all of the nodes including server, routers, APs, and mobile users are PC-based, which gives us an ability to run desired program and self configurations. PC-based mobile users are equipped with wireless card, which can connect to the particular AP and working as wireless user. We use Linux operating system (which can access to the kernel and easy to run our programs) for all nodes with global IPv6 address.

In the following subsections, we evaluate the communication and computation cost of the proposed key management scheme. Then, the detection and prevention of different types of DoS attacks are discussed, followed by the results and discussion on the path probability of attack.

4.1 Communication and Computation Cost

We evaluate the communication and computation cost of the existing methods based on the explanation of each method in Section 2, and compare with our proposed scheme. From the test-bed we measure the communication and computation cost by implementing the proposed and existing (LKH, SKDC, SKD, and OFT) methods. Therefore, the required programs are written in *gcc* on Linux. Fig. 5 shows the communication cost of join operation for multicast-unicast, LKH, SKDC, SKD, and OFT. LKH has the highest communication cost compared to other methods, while multicast-unicast and SKDC offer the lowest communication cost equal to 1 key only. By increasing the number of mobile users, the communication cost is increased for LKH, OFT, and SKD, while it is constant in multicast-unicast and SKDC methods.

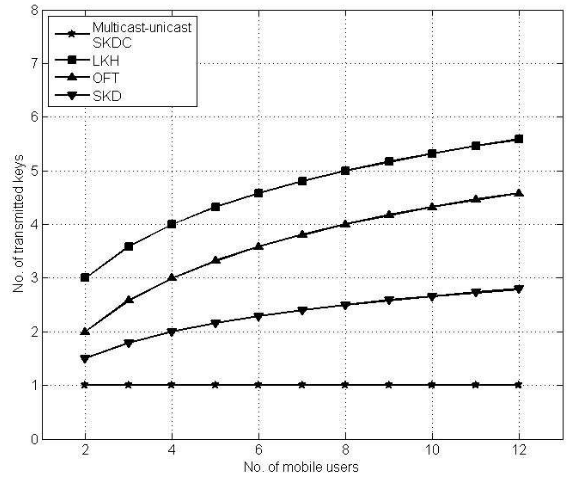


Fig.5: *Communication cost of join operation for multicast-unicast, LKH, SKDC, SKD, and OFT.*

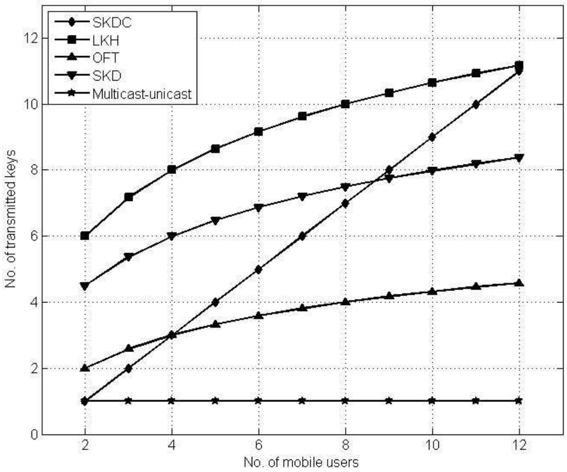


Fig.6: *Communication cost of leave operation for multicast-unicast, LKH, SKDC, OFT, and SKD.*

Fig. 6 illustrates the communication cost of leave operation. As can be seen, our proposed multicast-unicast method has lowest communication cost compared to other methods. The communication cost of leave operation for SKDC, LKH, OFT and SKD are increased due to increasing of number of mobile users, and SKDC offers the highest communication cost because all the members need new key when leave operation occurs. In both join and leave operations, the communication cost of our proposed multicast-unicast method is constant due to the dividing the communication network to two separate levels.

The computation cost of the server for the existing and proposed method for join and leave operations

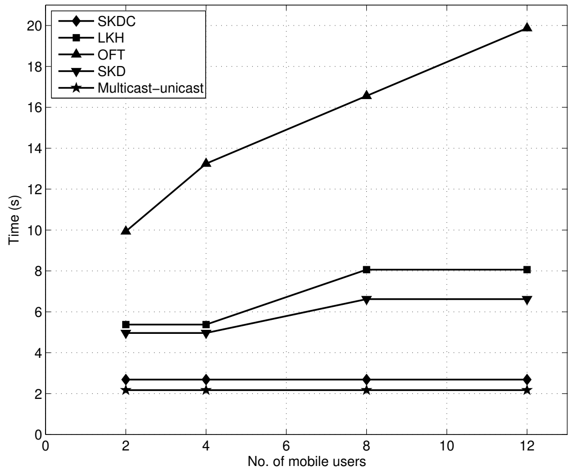


Fig.7: *Computation cost of join operation for multicast-unicast, LKH, SKDC, SKD, and OFT.*

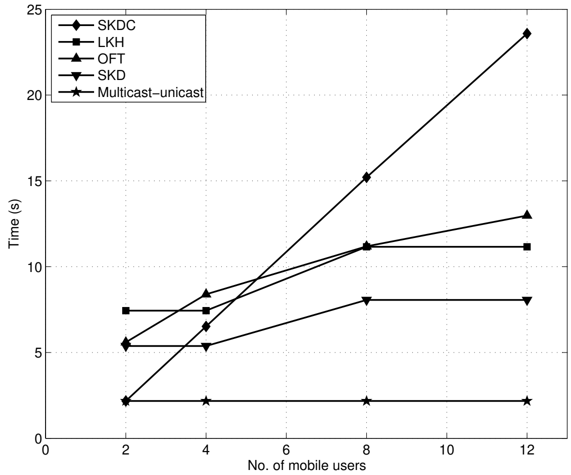
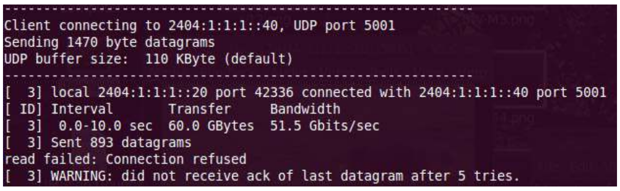


Fig.8: *Computation cost of leave operation for multicast-unicast, LKH, SKDC, OFT, and SKD.*

are shown in Fig. 7 and Fig. 8 , respectively. As can be seen, our proposed method experiences the lowest computation cost compared to the other methods. OFT has the highest cost in join operation due to the key derivation process. However in leave operation, the computation cost of SKDC is increased by increasing the number of users, and is higher than the OFT, SKD, LKH, and our proposed method due to key generation and distribution processes in the server, which is required for all members in the group.

4.2 DoS Attack Detection

We wrote and installed a program on user1 to simulate DoS attack by sending malicious data (by using UDP port 5001) to the server through the AP1 (based on the network shown in Fig 4). This kind of attack represents the inside group attacks described in Section 3.4. The IPv6 addresses of user1 (attacker here) and AP1 are *2404:1:1:1::20* and *2404:1:1:1::40*, respectively, as illustrated in Fig 9. The bandwidth usage is increased to 51.5 Gbits/s for the interval of 10 second due to DoS attack. Then the AP1 detects the attack and prevents the attacker to send malicious data, which is shown in Fig 9, *“connection refused”*. The MAC address of the attacker is blacklisted by AP1, and therefore, the attacker cannot connect to the AP1 anymore. Moreover, AP1 updates other APs (e.g., AP2) in the same domain. From then, the attacker is not able to connect to the network through this domain.



```
------------------------------------------------------------
Client connecting to 2404:1:1:1::40, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size:  110 KByte (default)
------------------------------------------------------------
[  3] local 2404:1:1:1::20 port 42336 connected with 2404:1:1:1::40 port 5001
[ ID] Interval       Transfer     Bandwidth
[  3]  0.0-10.0 sec  60.0 GBytes  51.5 Gbits/sec
[  3] Sent 893 datagrams
read failed: Connection refused
[  3] WARNING: did not receive ack of last datagram after 5 tries.
```

Fig.9: *DoS attack detection by AP.*

To prevent the attack, we set a maximum threshold for each user based on the bandwidth usage. If the bandwidth usage reaches the specified threshold, the AP behaves with it as attacker. Note that in the case of many-to-many communications which user needs more bandwidth usage, the user is not detected as attacker if AP authenticates the user to be as a server.

It is worthwhile to highlight that the DoS attack mentioned above is considered as inside group attacks and it is not specially for multicast transmission. However, a possible DoS attack for multicast transmission would be one where a node repeatedly joins and leaves a secured group at a rate so fast that rekeying takes longer than the join/leave sequence, meaning that there are never any valid keys for the subnet. Our proposed scheme can detect the so-called “DoS-Key-Request” attack and prevent the attacker by ignoring the join/leave request from it. The AP keeps the record of the requests of each users and behave the user as attacker if a node repeatedly sends the request.

It should be noted that in this case, even if the attacker sends many join/leave requests, the malicious requests affect the subgroup manager instead of making the server busy due to the separated levels of communication network in our proposed method. This kind of attacks can be considered as one of dangerous DoS attacks against multicast communication, which can be detected and prevented in our proposed method.

4.3 Path Probability of Attack

Referring the explanation of path probability of attack in Section 3.1, we consider different communication networks based on different value of parameters i, j, and k. Recall that i, j, and k are the number of

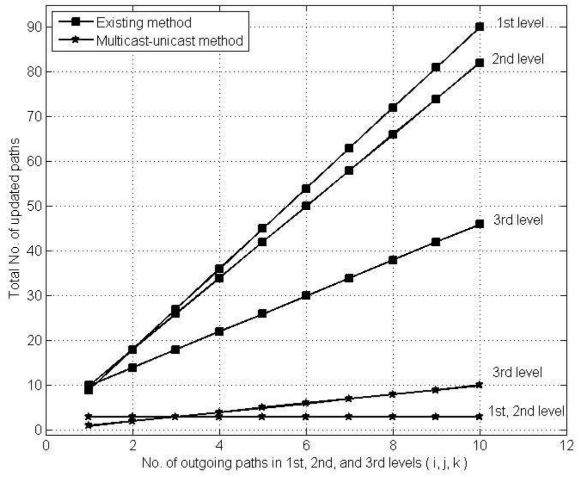


Fig.10: *Total number of updated paths.*

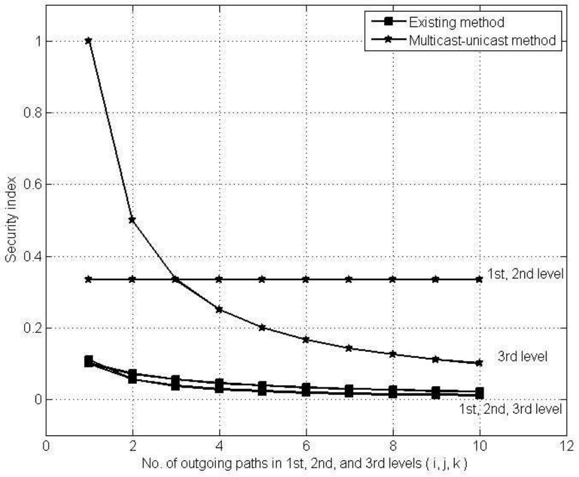


Fig.11: *Security index.*

outgoing paths in 1^{st}, 2^{nd}, and 3^{rd} level of the communication network (shown in Fig. 4), respectively. Two parameters are constant while one is increasing, then we measure the number of updated paths (as illustrated in Fig. 10) and security index of the network (as illustrated in Fig. 11).

Fig. 10 illustrates the number of update paths (N_P) of the existing and proposed methods for 1^{st}, 2^{nd}, and 3^{rd} levels. The N_P for 1^{st} level means i varies from 1 to 10 when $j = 2$, $k = 3$. As mentioned before, the number of update paths is equal to k in multicast-unicast scheme, and therefore, the N_P is constant when i and j vary. But in the existing method, N_P is increased.

The N_P for 2^{nd} level means $i = 2$, $j = 2$, and k varies from 1 to 10. The N_P is increasing slowly in multicast-unicast method when k is increased, while in the existing method, the N_P is increasing dramatically when i, j, and k are increased (3^{rd} level). When N_P is increased, the security index of the network is reduced due to increasing the risk index and path probability of attack.

Fig. 11 illustrates the security index (SI) of the existing and proposed multicast-unicast methods. When i, and j are increased, the SI of multicast-unicast method is constant, but in the existing method, the SI is decreased.

When k is increased, the SI of both methods is decreased. But multicast-unicast method achieves higher SI compared to the existing method due to separating the communication network to two levels. Recall that, whenever a join or leave operation occurs, only a subgroup of users under an AP require key updating which lead to small number of updated paths and therefore, higher SI.

5. CONCLUSION AND FUTURE EXTENSIONS

This paper proposed a new key management scheme for multicast over wireless IPv6 networks. The proposed multicast-unicast scheme, which divides the multicast network to two separate levels, solves the *1-affect-n* problem, and reduces the communication and computation cost of key updating. Moreover, the proposed scheme is able to detect and prevent different types of DoS attacks, and also improves the security of the network by decreasing the risk probability and increasing the security index compared to the existing ones. It is implemented in terms of experimental test-bed and the efficiency of the scheme is apparent by show its superiority in real time network.

The proposed key management scheme is able to outperform the existing methods by significantly reducing the communication and computation cost, while at the same time providing secure multicast communication.

At present, we are investigating the efficiency of our proposed scheme in terms of computation and storage cost or rekeying procedure for further improvement. At the same time, we are working on implementation of our proposed scheme in larger experimental environment.

References

[1] C. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," IEEE/ACM Transactions on Networking, vol 8, no. 1, pp. 16–30, 2002.

[2] J. C. Lin, K. H. Huang, F. Lai, and H. C. Lee, "Secure and efficient group key management with shared key derivation," Computer Standards & Interfaces, vol. 31, no. 1, pp. 192–208, 2009.

[3] W. Trappe, S. Jie, R. Poovendran, and K. J. R. Liu, "Key management and distribution for se-

cure multimedia multicast," IEEE Transactions on Multimedia, vol. 5, no. 4, pp. 544–557, 2003.

[4] M. Baugher, R. Canetti, L. Dondeti, and F. Lindholm, "Multicast security (MSEC) group key management architecture," IETF, RFC 4046, 2005.

[5] H. Ko, Y. Lee, K. Sung, H. Oh, and Y. Shin, "A Study on an effective group management scheme for secure multicast in MIPv6," International Conference on Information Security and Assurance (ISA 2008), 2008.

[6] A. Mehdizadeh, S. Khatun, and M. Borhanuddin, "Distinctive key management method to secure multicast IPv6 networks" IEEE 9th Malaysia International Conference on Communications (MICC), pp. 301-304, 2009.

[7] D. H. Je, J. S. Lee, Y. Park, and S. W. Seo, "Computation-and-storage-efficient key tree management protocol for secure multicast communications," Computer Communications, vol. 33, no. 2, pp. 136–148, 2010.

[8] R. Srinivasan, V. Vaidehi, R. Rajaraman, S. Kanagaraj, R. Kalimuthu, and R. Dharmaraj, "Secure group key management scheme for multicast networks," International Journal of Network Security, vol. 11, no. 1, pp. 30–34, 2010.

[9] D. Wallner, E. Harder, and R. Agee, "Key management for multicast: Issues and architectures," IETF, RFC 2627, 1999.

[10] A. Sherman and D. McGrew, "Key establishment in large dynamic groups using one-way function trees," IEEE Transactions on Software Engineering, pp. 444–458, 2003.

[11] S. Mittra, "Iolus: A framework for scalable secure multicasting," ACM SIGCOMM Computer Communication Review, vol. 27, no. 4, pp. 288–, 1997.

[12] Z. Jun, Z. Yu, M. Fanyuan, G. Dawu, and B. Yingcai, "An extension of secure group communication using key graph," Information Sciences, vol. 176, no. 20, pp. 3060–3078, 2006.

[13] S. Rafaeli and D. Hutchison, "A survey of key management for secure group communication," ACM Computing Surveys (CSUR), vol. 35, no. 3, pp. 309–329, 2003.

[14] H. Harney and E. Harder, "Logical key hierarchy protocol," draft-harney-sparta-lkhp-sec-00. txt, IETF Internet Draft (work in progress), 1999.

[15] R. Poovendran and J. S. Baras, "An information-theoretic approach for design and analysis of rooted-tree-based multicast key management schemes," IEEE Transaction on Information Theory, vol. 47, no. 7, pp. 28242834, 2001.

[16] M. P. Howarth, S. Iyengar, Z. Sun, and H. Cruickshank, "Dynamics of key management in secure satellite multicast," IEEE Journal on Selected Areas in Communications, vol. 22, no. 2, pp. 308319, 2004.

[17] Y. Sun, W. Trappe, and K. J. R. Liu, "A scalable multicast key management scheme for heterogeneous wireless networks," IEEE/ACM Transactions on Networking, vol. 12, no. 4, pp. 653666, 2004.

[18] W. T. Zhu, "Optimizing the tree structure in secure multicast key management," IEEE Communications Letters, vol. 9, no. 5, pp. 477479, 2005.

[19] A. Mehdizadeh, R. Abdullah, R. S. Azmir, and F. Hashim,"Secure group communication scheme in wireless IPv6 networks: An experimental test-bed," International Symposium on Communications and Information Technologies (ISCIT), pp. 724-729, 2012.

[20] A. Mehdizadeh, F. Hashim, R.S.A. Raja Abdullah, B. Mohd ali, and M. Othman, "Quality-improved and secure multicast delivery method in mobile IPv6 networks," The 16th IEEE symposium on Computers and Communications (IEEE-ISCC), Jun. 28-Jul. 1, 2011.

[21] A. Mehdizadeh, F. Hashim, R. S. A. R. Abdullah, B. M. Ali, M. Othman, and S. Khatun, "Multicast-Unicast Data Delivery Method in Wireless IPv6 Networks," Journal of Network and Systems Management, pp. 1-26, 2013.

[22] Bao, C., Boucadair, M., Bagnulo, M., Huitema, C., Li, X.: IPv6 Addressing of IPv4/IPv6 Translators. RFC6052, 2010.

[23] J. Wang, Y. Fang, and D. Wu, "Enhancing the performance of medium access control for WLANs with multi-beam access point," IEEE Transactions on Wireless Communications, vol. 6, pp. 556-565, 2007.

[24] E. Lopez-Aguilera, M. Heusse, Y. Grunenberger, F. Rousseau, A. Duda, and J. Casademont, "An asymmetric access point for solving the unfairness problem in WLANs," IEEE Transactions on Mobile Computing, vol. 7, pp. 1213-1227, 2008.

Abbas Mehdizadeh obtained his M.Sc. in IT & Multimedia Systems, and Ph.D. in Communication and Network Engineering from Universiti Putra Malaysia (UPM), in 2008 and 2012, respectively. He is currently Senior Lecturer at Nilai University Malaysia. He also served as researcher at MIMOS Bhd Malaysia, in 2008. He was the proud recipient of the Best of the Best and Gold awards of Malaysia Technology Expo 2009 (MTE'09 - The largest Invention and Innovation Expo in Malaysia). Abbas is a Senior Member of IEEE, member of IEICE-Japan, International Engineering Consortium (IEC), and International Association of Engineers. His research interest includes wireless communications and networks, network security, multicast networks, and IPv6.

Fazirulhisyam Hashim holds a M.Sc. degree from Universit Sains Malaysia and a Ph.D. in Telecommunications Engineering from the University of Sydney, Australia. His research primarily focuses on network security and QoS of heterogeneous mobile and cellular networks. He is a TPC member/reviewer for many international conferences and IEEE journals. Fazirulhisyam is a member of the IEEE, and currently a lecturer at the Universiti Putra Malaysia.

Raja S. Azmir Raja Abdullah received the BEng. (2000) in Electronic and Electrical Engineering and MSc. (2001) in Communication System Engineering from The University of Birmingham, U. Kingdom. He then received his PhD in 2005 also from The University of Birmingham majoring in Radar and Microwave System. His research focuses on the Microwave, Radar Systems and Wireless Sensor Network and has been involved in the development of hardware and software for Radar sensors and Wireless Sensor Network (WSN). He his among the pioneer on developing advanced practical Forward Scattering Radar (FSR) System and the system has been adopted for various applications including civil, military and medical. With addition of intelligent system, the advanced FSR system capable of automatically detect and also classified most ground and air targets. Recent development is on the high resolution radar networked (RSN) which can be applied for small object detection and classification. By adapting the WSN advantages the drawback of traditional radar can be compensated. The future radar will has a size of computer mouse compared to huge radar set mounted on the tower. In 2006 he became the founding Head of the Microwave, Millimeter wave and Radar Systems Laboratory (M2aRS). The laboratory specialized in the radar systems, signal processing, microwave devices and sensor networks. Currently he is a director for Wireless and Photonic Networks Research Center (WiPNET) at the Universiti Putra Malaysia.