A Study on the Key Management Strategy for Wireless Sensor

Hoon Ko[a], Kitae Bae[b], Goreti Marreiros[c], Haengkon Kim[d], Hyun Yoe[e] and Carlos Ramos[c]

[a] The Department of Informatics, J. E. Purkinje University.
[b] Department of New Media, Korean German Institute of Technology
[c] Institute of Engineering Polytechnic of Porto
[d] Department of Computer and Communications, Catholic University of Daegu
[e] Department of Information and Communication Engineering, Sunchon National University

KEYWORD

Group key
Key Management
Key Strategy
Low-Cost
Session Key

ABSTRACT

Many users who are in a cyber-space usually want to join the social group to have or to share their information. Now, there are two ways to join the group, the group manager invites them, and the users who want to join ask the owner. These days the group polices usually follow this way. But, it can be faced a security problem when the manager send group messages in near future because they don't have any securities. Therefore, the security modules to join groups will be needed when they join the group or when they read the group messages. To set the security, we have to think how to keep the key such as a generation /an update/an arrangement, because all users need the key to join the groups or to read the group messages by decrypting. The key are going to be used to joining the group when it dynamically changes such as frequent group joining and leaving. If it applies or uses the existing methods in the smart cities which consider the users who will move globally, it could easily assume that the overhead/the cost of CPU will be increased and it follows capacity down because of lots of the key updates. So, to let them down, we suggest three key strategies, a group key, a subgroup key and a session key in this paper.

1 Introduction

Wireless Sensor Networks (WSN) usually consists of many sensors nodes which are physical environment; there are lots environmental conditions such as temperature, sound and pressure including some applications which involve them. Also Social Networks (SN) can be in the applications in a physical environment. The groups which establish in Social Networks (SN) where use in Wireless Social Network usually share their information with all group members by sending or by posting their opinions to all members. That is, the users who are joined in the SN group send their information to all members to share. In the cyber-space, there are lots of attackers who will try to get users' personal information which will flow an inside of the groups by attacking in weaknesses area. If some of them succeed to join the group with the hijacked one, then all members of the group will be exposed to the attacker. Therefore, there need the security methods in a social socitey to protect the messages which are sending or receiving. Also, we have to know their requirements for the application services in secure group communications to study the security methods. These days the group communications or the multicast communications usually don't use in social groups, just only some systems are using them limitedly. Although the social networks for commerce in a system are using the security functions, they have just the simple functions such as an authentication step or an authorization step with user IDs to have access controls that the system uses. Because all users have each ID in the most of social networks, the joining step will be easy if they only get the key to joining the group in a social network. Therefore, how safely they arrange the group/subgroup key to accept a group manager and the security key to encrypt and to decrypt are more important [H.-C. Shih, 2013][C.-M. Chen, 2012]. Absolutely, the members who left the social group are not supposed to read the messages as soon as they left. The server in the system has to update the key to avoiding that the left members can read them, so the updated session key has to be sent to all members who are still joining the group [Q. Shi, 2013]. However, in case the group which has dynamic members who leaves and joins frequently/repeatedly, the server has to have the group key/the session key updating/generating and to send to all members, then the processing time of the server will be decreased. Therefore, it makes the cost down and process time increases [S. U. Khan, 2010]. Especially, if there are many members and they do leave and join many times in the social group, the key generation process and the key arrangement process will be going into more complexity [S. U. Khan, 2010]. So, it is needed to study to find more efficient way how to generate the group key, how to update the group key, and how to arrange. To solve it, we decided to study the group key to accept if they join a social group and the session key to encrypt and to decrypt the messages to solve this problem through this paper [H.-C. Shih, 2013][C. M. Chen, 2013][Y. Lin, 2012]. This paper follows, in section 2, it defines security problems, and it writes the suggesting model in section 3. The result of an experiment and the discussion will be in section 4; last, section 5 is for a conclusion that it puts the results and future work.

2 Security Problem

Ko had published a paper that he had suggested the distributed/hierarchical structure to minimize the system's cost, and also it had tried to reduce the number of the key and the number of key updates including a group key and a subgroup key. This paper also had proof that his idea went better than the existing papers which it has been using the centralized control methods [S. H. Jokhio, 2012]. In existing methods, when a member joins a group again, it has to get the group key from a group manager. The users only could use this key to rejoin, so, to take this process to rejoin, some delay had happened. He insisted through [S. H. Jokhio, 2012] that there were some subgroup managers, and a subgroup manager only kept the mission to transfer the messages to all members, they don't have the manage functions. So the mission looks simple so that it could follow the cost and overhead down. Finally, the subgroup manager could react efficiently to their asking to join a group, it could be their benefits. It got the network performance and happened network overhead too much because of the security resource to provide safe and efficient multicast services and the structure in this model. He also explained his other idea in [Hoon Ko, 2006]. It showed that the distributed/hierarchical scheme minimize the overhead and reduce the number of key and key updating operation than other multicast schemes. In this paper, when a member joins a group, it gets a group key from group manager and joins a subgroup using the group key. Therefore, a subgroup manager just transfers data to its members and it makes our scheme simpler. This provides more safe and efficient multicast services and gives information to reduce network overload to performance decline due to security resources. However, this paper has some weaknesses to mobility users in a security. A multicast system that one sender usually sends their message to group members who are widely located has many weaknesses than a unicast system because the multicast system sends. That is, the multicast system passes through many communication links, so they will be attacked from their identification camouflage, a traffic monitoring, Denial of Service (DoS) and repudiation and so on when they pass the links [R. H. Weber, 2010]. To keep the security of the sending message, the multicast system generally sends the message after encrypting with the shared key which keeps on all members. A group key which is a symmetry key will be kept by all group members, whenever it changes, it should be updated. It means that the left member doesn't read or understand the messages; also it forbids joining again the group after it left [R. Roman, 2013][T. Winter, 2012].

There are some problems in big multicast systems like next, If the applications in the network don't help, they can't establish optimized tree to communicate. And finally it follows packet losses, tree depth problem to set to communicate, locality recovery trouble when they have network problems, error recovery delay and delay increasing, and the cost problem in a buffer control to resend. Also, the user who connects to the network only can receive the social messages after they join the social group without the network control access. At this time, the attackers can easily do the DoS attack on purpose or with another attack tool. Not only this affects to group members, but also they potentially affect all users who will join the social group in near future including who are connecting with networks [F. Zhu, 2012]. In addition, the number of communication links which are sent in global multicast system is more than the number of unicast links which is connected to a single source and a single destination. Therefore, the multicast system offers many opportunities to intercept the transferring message to the attackers who try to attack [Y. Liu, 2012].

Next, it surveys the requirements how it manages the key to protecting the forward secrecy and backward secrecy. Forward secrecy is designed to prevent the compromise of a long-term secret key from affecting the confidentiality of past conversations. However, forward secrecy cannot defend against a successful cryptanalysis of the underlying ciphers being used, since a cryptanalysis consists of finding a way to decrypt an encrypted message without the key, and forward secrecy only protects keys, not the ciphers themselves [W. Hu, 2010]. The attacker can capture a conversation whose confidentiality is protected through the use of public-key cryptography and wait until the underlying cipher is broken. This would allow the recovery of old plaintexts even in a system employing forward secrecy. It should be noted that such attacks are currently believed to be theoretical. Backward Security, Future keys outputs, after refreshment, they cannot be disambiguated by the adversary even if the adversary knew the state of the keys before it was refreshed.

3 The Proposed Method

3.1 Multilevel-SN Model

The Fig. 1 shows the simulation model and table 1 defines the simulation parameters, also the key generation and transmission model is in the Fig. 2. Usually all data transfer through WSN, as the same way, all data which is from all users who are in social networks, they can be shared in WSN. However, the user can send the data by themselves in a social network, at that same time; it means they also receive all data because they already joined in social groups. To experiment, it defines all parameters in the Table 1. The model looks like a Mesh types and there are social networks in the model, the distance (d) and the number of keys will be increased or decreased according to the members who are in social groups.

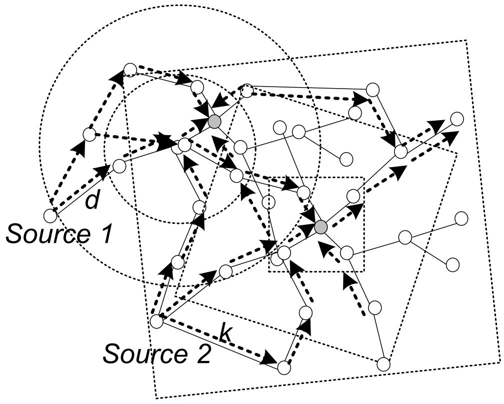

Fig. 1. Simulation Model in WSN

$n = (n_1, n_2, \ldots, n_x), C(\emptyset)_{n1} = (k_1 d_l)_{n1},$

$\boldsymbol{C}(\emptyset)_{\boldsymbol{T}}$ is the total Cost (1), $\boldsymbol{C}(\emptyset)_{\boldsymbol{A}}$ is the average cost (2) of the users who are joining in social groups,

$$C(\emptyset)_T = \sum_{n=1}^{x}(C(\emptyset)_n)\times\alpha \text{ ....(1)}$$

$$C(\emptyset)_A = \frac{C(\emptyset)_T}{n} \text{........................(2)}$$

And it defines the number of generating key of the proposed model in notation (3)

$$k = 2(\mathrm{n} - 1)$$

.........................(3)

Each node, (totally 100 notes in this model) is by-directionally connected, finally it can express in distance vector *(d_1, d_2,...,d_n)* from the fixed node (Not moving node) to mobile nodes (4).

$$d_i^2 = (x_0 - x_i)^2 + (y_0 - y_i)^2 \text{ ....(4)}$$

It makes Model (5) with (1) (2) (3) and (4).

$$M(k,d) = \sum_{n=1}^{x}\{n(k+d)C(\emptyset)_T\}\times\alpha, n \in G_{sn}..(5)$$

Parameters	Values
n,	The number of nodes 100
t	Simulation time 1000 sec
k	The number of Keys
d	Distance ($1 \leq d \leq 2$)
M	Model
$C(\emptyset)_T$	Total Cost
$C(\emptyset)_A$	Average Cost in each Node
G_{sn}	Group in Social Network
α	Variable value ($0.1 \leq \alpha \leq 0.5$)
Message Size	1000

Table 1. . Simulation Parameters

And, $M_{Key}(k,d)$, it defines the model which is in WSN. $C(\emptyset)_{n1}$, it defines the cost which happens in the Node1, with this cost, it can know the distance of each node and k. The model,$M_{Key}(k,d)$, includes all values that it mentioned. Table 1 shows the simulation parameters.

3.2 Light key management

It suggests the two-depth key tree *(gk_i:Group key -> (sgk_i:Sub-Group key||sk: Security Key))* to solve the existing trouble of the key tree in a big social group like a big multicast group [Fig. 1]. It tries to offer the benefits of the two depth key tree for a key generation and for a key arrangement. Also, the server normally generates the key tree with their secret information which was received from group members to avoid the exposure of the group key; the server sends the generated key *(Sub-Group key or Security Key)* to all group members after encrypting. Then, the group members update the key as soon as they receive it from the server. Because a lot of users are in a social group, they will be asked to use the multicast system to share the updated subgroup keys.

Next, it suggests the security way how they keep for the forward secrecy and the backward secrecy which have the requirement of the multicast system to keep the keys safely [Table 2]. Before analyzing it, it writes down their definitions and requirements. Their qualification is that forward secrecy is a property of key-agreement protocols ensuring that a session key derived from a set of long-term keys cannot be compromised if one of the long-term keys is compromised in the future.

Items	Contents
Forward secrecy	A public-key system demonstrates a property referred to as perfect forward secrecy when it: - It generates random public keys per session for the purposes of key agreement, - It does not use any sort of a deterministic algorithm in doing so. This means that the compromise of one message cannot lead to the compromise of others, and also that there is not a single secret value which can lead to the compromise of multiple messages.
Backward secrecy	The attackers can't use the keys that they already will be used.

Table 2. Forward Secrecy vs Backward Secrecy

The key which is used to protect transmission of data must not be used to derive any additional keys, and if the key that it used to protect transmission of data is derived from some other keying material, then the material must not be used to derive more keys. The reason why it has to process is to permit the compromise of single key permits to access the data protection by the single key. On the other hand, backward secrecy should be that the key can't use by the attackers after they left.

Next, the notation explains the speed of the message sends by following in an experiment time.

$$\mathrm{S} = \frac{\sum_1^{sg} d_n}{\sum_r^{1000} t_r}\text{............................(6)}$$

3.3 Hierarchical Key Structure

We describe the hierarchical key structure for the group key generation and for the key update in a big social group [Fig. 2].

The way how to get the group key and depth will be increased whenever the members of the social group increase in a binary tree, on the other hand, in this suggest model, the number of them will be not over than 2-level (as set default, 2), although the members increase continually. The algorithm to establish the key tree in 2-level in table 2 is going to be used. However, according to how far they are distributed, the sgk_i decides who controls the sgk_{i-1} which is far away from gk_i. In case sgk_{12}, it is far way, so sgk_1 decides to control sgk_{12}.

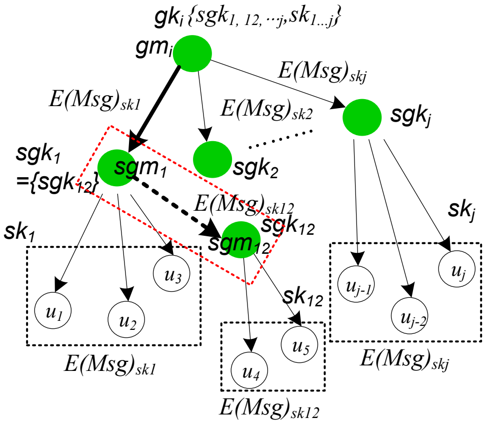

Fig. 2. Hierarchical Key Structur

Table 3 explains the hierarchical key structure algorithm.

```
Input: the number of edges, X
Output: hierarchical key structure

Procedure define Structure (HKS)
Begin
    Define $gk_1=\{sgk_{1,12,\ldots,j}, sk_{1,\ldots,j}\}$
    Define $sgk_1=\{sgk_{12}\}$, $sgk_2$, $sgk_j$
    //$sgk_1$ controls $sgk_{12}$
    Define $X$
    //it depends on the distributed location of sg
    Calculate $X+1$
    Calculate $X*2$
    Send $n$
End
```

Table 3. Hierarchical Key Structure Algorithm

3.4 Group key generation

We suggest the way how setting the key tree with the security key that the group members have. In this model, no matter how many the member changes dynamically, the tree depth always keeps in 2-level *(gk_i: Group key -> (sgk_i: Sub-Group key||sk_i: Security Key))*. The existing system that it usually uses the binary tree generation generates the group key *(gk_i)* which integrates into the previous group key with the one-way function from left node to upper direction with the key tree information.

In this case, if they want to generate the group key in members, they need them as much as the number of depths. Therefore, this model has a limitation about the key tree which has to have 2-level to minimize the cost. And they try to get the Shared Key *(Srd_K)* which keeps only for all group members to share safely all keys with all members. The server generates the group key and sends it after encrypting [Table 4].

The member who received the encrypted message can decrypt them with the session key which the member already keeps. Because the partial information that it needs to make the key is the secret information which is located on upper of the member and secret key around them, it generates the group key with the shared key that only they need. In addition, this generates the group key for all group members only with the information of a server without the group key sending, so we could say that the key can keep safely. Also, although the partial key is exposed, the attackers have to know the last information of key to understanding the messages. However, the attackers can't get the last information, so it is unable to decrypt them without the last information.

```
Procedure generate GroupKey($gk_i$)
Incase message(M) is from members
Do
    Decrypt (M) with (gk)
    Generate HKS with (gk)
    Define Multicast Message <- (M) into HKS
    Encrypt Multicast Message (MM) with (gk)
End
Multicast MM to all group members
//MM: Multicast Message
```

Table 4. Group Key Generation Algorithm

They have weakness in the existing or in previous methods such as the key exposure when they arrange their key in this paper; also this paper suggests how they use the encrypted partial information to generate the group key which it would be sent to the server for social members. Also, this involves how it updates the group key, that is, the group key would not send to all group members, instead of it, each member updates the group key by themselves by using one-way function following beacon signals. In addition, when the member leaves the group, because following the beacon signals from the server, the policy to update the server key is going to process, the dynamic group key management strategy are included in this idea.

The member who received the encrypted message can decrypt them with the session key which the member already keeps. Because the partial information that it needs to make the key is the secret information which is located on upper of the member and secret key around them, it generates the group key with the shared key that only they need. In addition, this generates the group key for all group members only with the information of a server without the group key sending, so we could say that the key can keep safely. Also, although the partial key is exposed, the attackers have to know the last information of key to understanding the messages. However, the attackers can't get the last information, so it is unable to decrypt them without the last information.

3.5 Group key update

They have weakness in the existing or in previous methods such as the key exposure when they arrange their key in this paper; also this paper suggests how they use the encrypted partial information to generate the group key which it would be sent to the server for social members. Also, this involves how it updates the group key, that is, the group key would not send to all group members, instead of it, each member updates the group key by themselves by using one-way function following beacon signals. In addition, when the member leaves the group, because following the beacon signals from the server, the policy to update the server key is going to process, the dynamic group key management strategy are included in this idea [Table 5].

```
Procedure groupkey update
Receive message to update from Servers
If receive messages from server
Begin
    Define update (gk)
    Send ack.message to server
End
```

Table 5. Group Key Update Algorithm

To avoid the key exposure and to decrease the key update time, it considers members who update the key, that is, if the members update it by themselves, it is able to decrease the key update time than the existing method that they receive the new group key from the server after updating it. Then it can protect the key exposure and decrease the key update time. Although the attackers who want to get the group key take the encrypted messages between two users' communications, they are not able to have the key *(Srd_K),* they can't decrypt the encrypted messages. And, after some members left with despiteful purposes, if they try to receive the message with their old key, they can't read them. Because on leaving the group, the key update will be performed in the global system. Therefore, the old key will be expired to decrypt.

4 Discussion

4.1 Security policy in a social group

The security policy in a social group involves their behavior, the access control, the parameters, and the security mechanism which are related to a group security. If there are many receivers in a network system such as multicast systems, it surely will be not efficiency to have a negotiation about a security parameter through two users; senders and receivers. Therefore, overcoming un-efficiency problem is to generate and to arrange for the group manager which controls the multicast session. Next, we define each policy;

- *Sub-group key policy*: The key has to be updated whenever it needs to authenticate group members. So, on leaving the one of members, the sub-group key has to be updated [6].
- *Access-control policy*: the group member gets their authentication with the subgroup key when it joins the group by receiving the subgroup key which is going to be arranged by the group manager [6].
- *Group manager policy*: The subgroup manager sends the subgroup key to the members who want to join the group. And, when the subgroup key wants to receive, it sends the subgroup key [5].

- *Application message*: It defines all security mechanism such as confidentiality, integrity, group authentication, and source authentication which apply following the security demands of an application data [7].

4.2 Hierarchical key managemenet

This paper has a new idea that it tries to solve the existing key management problem by applying the distributed and the hierarchical key management. Because it has a difficulty when it manages in existing methods such as the structure of the key tree, we define the limitation that it sets the two-leveled *(gk$_i$:Group key -> (sgk$_i$:Sub-Group key||sk: Security Key)* depth in our suggestion to solve the difficulties. The limitation means that the key to a group that it belongs to the distributed/hierarchical model usually will be asked to the upper-level group, and the group manager is responsible for the all keys' arrangement [9]. The group manager takes all missions such as a group key generation, a sub-group key generation, and a security key generation/their arrangement, the subgroup manager controls the all sub-group members by requesting the sub-group key. There are two categories to manage the key to showing us the differences between centralized-control method, hierarchical-control method and the proposed method [Table 5]. In the centralized-control method, the group controller is responsible for the joining and the leaving for all group members. Since the number of the encryption will be increased in proportion to the group size, the overhead to process the key management will be increased whenever the members are located in widely dispersion such as the social network group which is a big group. Finally, the suggesting methods could decrease the overhead by using the hierarchical structure to manage the keys.

4.3 Group key update

The group key updates whenever the one of members joins or leaves the group, and the social network that the members are located in wide distribution, this network tries to be satisfied with the requirements of the security such as a forward secrecy and a backward secrecy by updating the dynamic group key. The member who wants to join the group has to not know the previous services after joining, and the left member also have to not realize the group after they left. In case the member joins a group, it receives the encrypted updating key from a group manager. And existing members will receive the new group key *(K_{i+1})* with the previous group key *(K_i)* in a group manage server. The new group key which already encrypted before it sends will be decrypted with the key that the group members have. In case leaving, it has to consider next two assumptions, first, normal leave who the member wants to leave, and second, un-normal leave, in this case, the member leaves from sudden network troubles and so on. So the manager tries to take a look inside of the all groups periodically or non-periodically.

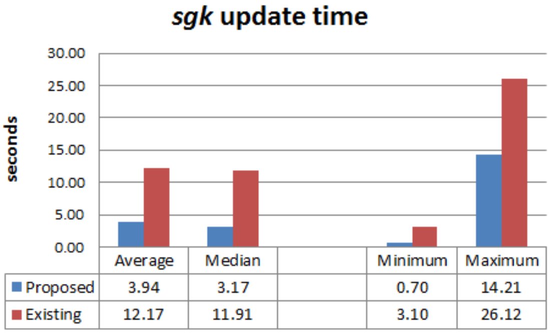

Fig. 3. *sgk* update time

Fig. 3 shows the *sgk* update time. It shows that the existing models are not good in social groups because there are many users who are joining and leaving in social groups, that is, they are so dynamic activation. Therefore, we need to study a new model and a new strategy for them. According to the simulation results, the proposed models took 3.93 sec (Minimum: 0.70 sec) in *sgk* updating, but the existing model took 12.17 sec (Minimum: 3.10 sec). The reason about a big difference between the proposed model and the existing model is when we set the simulation next, in the proposed model; *sgm_1(sgk_1)* takes the *sgk_{12}* for *sgm_{12}*. On the other hand, in existing model, *gm* takes all *sgk*, so the gap between the results looks big.

There are two figures [Fig. 4] and [Fig. 5] to show its experiment for the speed of message send, the speed of key generation. In the message send, it shows that the message in the proposed methods send more quickly than hierarchical method and centralized method.

4.4 Security analysis

In this section, we explain how they accept to the security requirements of a forward secrecy and a backward secrecy that we suggest.

Backward secrecy: When the new member joins a group, it is not supposed to know the previous message after it joins. To do it, the new member sends the message after encrypting the new key after updating the group keys with random numbers that the new member generates. Finally, because the new members don't know the group key of a previous group, they can't decrypt them. So it can say that the new idea in this paper is stronger than existing methods.

Forward secrecy: In case the member leaves the group, it has to not read the messages that they receive after left. To do it, after they left, the group key has to be updated following the suggesting methods in this paper. And the messages should be encrypted the updated key, and they send to all member. Because the messages are encrypted the updated key, the left member can't decrypt this messages. Also, according to this paper, the left members can't catch the all key information because the key is going to be updated following a beacon signal.

4.5 Key generation cost

In existing key graph schemes, when a member joins a group, the existing members encrypt the new group key/new subgroup key with the old group key/the old subgroup key, next it multicasts them to all members. And, the new member will receive them in a unicast system after encrypting the new group key/the new subgroup key with a key that the new member already shared with the group manager or the subgroup manager before their joining. On the other hand, the suggesting method that it generates and shares at that same time, shares the minimized key exchanges or the key transferring and the key sharing with the existing members and new members. The table 6 shows 'the number of rekeying messages in a member-joining (Group manager and subgroup manager)' when a member joins a group, also 'the number of rekeying messages in a member-leaving (Group manager and subgroup manager)' when a member leaves a group.

Table 6. The number of rekeying messages in a member-joining

Category	Centralized Control Method	Hierarchical Methods by Dr. Ko [Hoon Ko, 2008]	The proposed method
The number of key update (Joining case)	n-1 (n: the number of user)	d*(n-1) (Depends on the number of group or subgroup)	2
The number of key update (Leaving case)	n (n: the number of user)		1

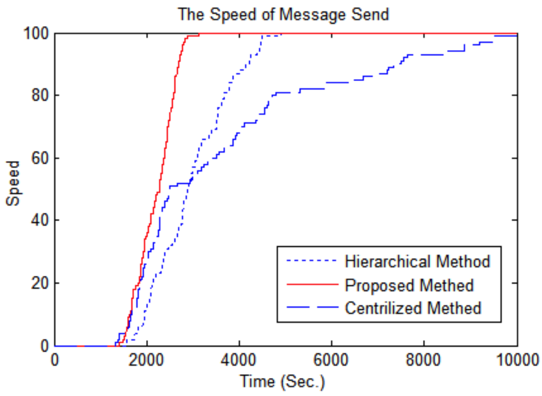

Fig. 4. The speeds of message send

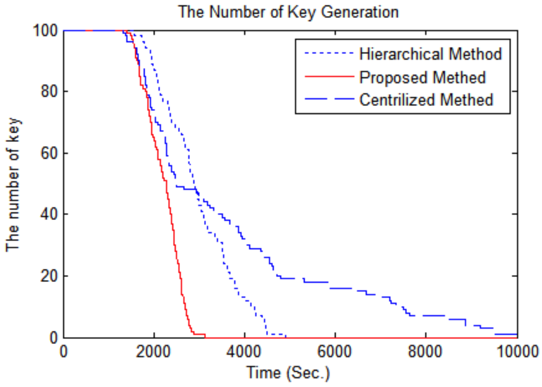

Fig. 5. The number of key generation

In fig. 5, the number of key generation, when it first starts, they begin to generate all keys; the three methods located on the top of the graph. However, because, in the proposed method, it doesn't need to generate the key whenever they move, the number of the key would be down fewer than two existing methods. On the other hand, the centralized method which generates whenever it moves makes many keys than other two methods.

5 Conclusions

In this paper, first we identified the authentication features and the requirements of the smart devices to use in a social network, and then we defined the limitation of the key-depth to authenticate for all members efficiently. It established the hierarchical structural model to do the fast services in a social group, and it sets the group manager and defined new key management strategy who takes all authentications to do the quick authentications. Finally, this paper that we suggested the distributed/the hierarchical social group to solve the existing problems such as cost and processing time, could overcome the weakness such as a key management trouble and a key arrangement trouble. The distributed group has a group manager to manage the subgroups, and subgroup managers to control the local members. This idea lets the model to decide to generate and to shares with the subgroup key generation and with the group key generation at that same time. And this idea could solve the existing problem such as a key generation problem and a key arrangement problem. If the new systems use this algorithm to do the fast authentication process and the minimized key generation/exchange cost, we expect that all members can get the safe and fast services in social network.

6 Acknowledgment

This work has been developed under the project EKRUCAmI - Europe-Korea Research on Ubiquitous Computing and Ambient Intelligence, a project receiving funding from the People Programme (Marie Curie Actions) of the European Union's Seventh Framework Programme FP7/2007-2013/, REA grant agreement No 318878 (PIRSES-GA-2012-318878), and also supported by the International Research & Development Program of the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (Grant number: K 2013079410).

7 References

Shih, H.-C., Ho, J.-H., Liao, B.-Y., and Pan, J.-S. *Hierarchical gradient diffusion algorithm for wireless sensor networks*. Applied Artificial Intelligence in Recent Trends, 2013

Chen, C. M., Lin, Y. H., Chen, Y. H. and Sun, H. M. *Sashimi: secure aggregation via successively hierarchical inspecting of message integrity on wsn*. Journal of Information Hiding and Multimedia Signal Processing, 2013.

Lin, Y., Zhang, J., Chung, H. S.-H., Ip W. H., Li, Y. and Shi, Y.-H. *An ant colony optimization approach for maximizing the lifetime of heterogeneous wireless sensor networks*. IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, 2012.

Shi, Q., Zhang, N., Merabti, M., and Kifayat, K., *Resource-efficient authentic key establishment in heterogeneous wireless sensor networks*. Journal of Parallel and Distributed Computing, 2013.

Chen, C.-M., Lin, Y.-H., Lin, Y.-C. and Sun, H.-M. *RCDA: recoverable concealed data aggregation for data integrity in wireless sensor networks*. IEEE Transactions on Parallel and Distributed Systems, 2012.

Khan, S. U., Lavagno, L. and Pastrone, C. *A key management scheme supporting node mobility in heterogeneous sensor networks*. Proceedings of the 6th International Conference on Emerging Technologies (ICET '10), 2010.

Weber, R. H. *Internet of things—new security and privacy challenges*. Computer Law and Security Review, 2010.

Brachmann, M., Keoh, S. L., Morchon, O. G. and Kumar, S. S. *End-to-end transport security in the IP-based internet of things*. Proceedings of the 21st International Conference on Computer Communications and Networks (ICCCN '12), 2012.

Roman, R., Zhou, J. and Lopez, J. *On the features and challenges of security and privacy in distributed internet of things*. Computer Networks, 2013.

Winter, T., Thubert, P., Brandt, A. et al. *RPL: IPv6 Routing Protocol for Low-Power and Lossy Networks*. IETF RFC 6550, 2012.

Zhu, F., Mutka, M. W. and Ni, L. M. *Private entity authentication for pervasive computing environments*. International Journal of Network Security, 2012.

Liu, Y., Li, J. and Guizani, M. *PKC based broadcast authentication using signature amortization for WSNs*. IEEE Transactions on Wireless Communications, 2012.

Hu, W., Tan, H., Corke, P., Shih, W. C. and Jha, S. *Toward trusted wireless sensor Networks*. ACM Transactions on Sensor Networks, 2010.

Jokhio, S. H., Jokhio, I. A. and Kemp, A. H. *Node capture attack detection and defence in wireless sensor networks*. IET Wireless Sensor Systems, 2012.

Ko, H., Jang, U., Kim, S. and Shin, Y. *An Effective Group Management Method for Secure Multicast Transmission*. The Korean Institute of Information Scientists and Engineers, 2006.

Ko, H., Lee, Y., Sung, K., Oh, H. and Shin, Y. *A Study on an Effective Group Management Scheme for Secure Multicast in MIPv6*. ISA2008, 2008.