A Novel Cluster-based Key Management Scheme to Improve Scalability in Wireless Sensor Networks

Seyed Reza Nabavi[1*], **Seyed Morteza Mousavi**[2]
1- Department of Computer Engineering, Arak Branch, Islamic Azad University, Arak, Iran

Abstract
Wireless sensor networks are mobile ad-hoc networks in which sensors have constraint resource and communication. Providing data and connection security needs appropriate encryption protocols. Key management is a basic security mechanism for wireless sensor networks. While several key management schemes have been proposed for sensor networks, it is still a challenging issue in these networks. Secure key management in wireless networks such as wireless sensor networks is one of the issues attracting researchers' attention in terms of energy consumption and processing load on sensor nodes, as well as security problems. Due to storage limit in these networks, scalability is one of the most important components discussed in key management. In this paper, we proposed a scalable key management scheme based on clustering for wireless sensor networks, which is consists of two setup and maintenance phases and supports node mobility. The results show that our proposed solution improves the network's scalability, while providing high secure connection and total improved efficiency. Moreover, our proposed scheme has lower computational overhead, energy consumption and delay compared to modern schemes.

Keywords:
Security, Clustering, Key Management, Wireless Sensor Networks

1. Introduction

Wireless sensor network may be considered as a network consisting of sensor nodes which communicate with each other using the radio channel. Today, wireless sensor network is used in real time or mission-critical applications [1]. Using wireless sensor network in mission-critical applications may create a new requirement for application of these networks. These requirements include security and mobility (dynamic) requirements. To protect the network from malicious attacks, considering the security issue is critical and to increase the network access domain, assurance mobility characteristic is necessary. The aim of this paper is studying and resolving the security issues of wireless sensor networks in mobile and dynamic scenarios. Wireless sensor networks' security is much more complex, in comparison with conventional security mechanisms. Extensive studies done in wireless sensor networks have focused on encryption solutions for security. Encryption solutions focus on key management issues. Different key management algorithms have been proposed in this regard, considering network management and key sharing between different nodes. In this regard, a large number of key management algorithms have been proposed, in which only a flat, static network is considered, and mobility and dynamic in the network are not considered [2]. The important role of this paper is proposing a key management solution in cluster-based dynamic environment. Cluster-based networks are efficient in terms of scalability and energy efficiency. These kinds of network arrangement help to improve key management efficiently and reduce fast penetration rate of security attacks in the network. Existing sensors in these networks are limited in terms of energy resource and therefore, key management algorithm for the sensor should have the lowest overhead for message transfer and computations. Cluster-based mechanism may be helpful in this regard.

In this paper, a novel key management algorithm has been proposed, considering challenges discussed so far. The existing challenges are responded by a novel scheme, as well.

In this scheme, a cluster-based wireless sensor network has been considered. Here, we assume that cluster head and nodes are both mobile, meaning that the managers of key and nodes are mobile. The suggestion proposed here is devolving the key management responsibility to another node in the cluster, meaning that we make a new key manager or cluster head. In this work, we assumed that when a cluster head or key manager is approaching the cluster boundary, it transfers key management responsibilities to another cluster head, for which we use cluster head selection algorithm.

Here, the algorithm considers two phases: the first phase is the setup phase, used in clustering and key distribution in the network. The second phase is responsible for controlling or maintaining keys during mobility.

Moreover, to exchange information, ECDSA encryption algorithm has been used, which provides a fast encryption with low energy consuming.

2. Related Work

Key management may be defined as a set of techniques and procedures that support establishment and maintenance of keying relationships between authorized parties. Key management technique for a secure application must

Manuscript received July 5, 2016
Manuscript revised July 20, 2016

minimally incorporate authenticity, confidentiality, integrity, scalability, and flexibility [1].
Table 1 shows the key management mechanism comparison. Key management mechanisms are compared considering scalability level, node authentication and their deployment knowledge. This comparison shows that many algorithms have proper functionality, but their efficiency degrades by the mobility of nodes and they do not have a well performance in scalable environments. The idea behind this is to propose a scalable key management which properly works under mobility of wireless sensor network.

Table 1 Key Management Mechanisms Comparison

Scheme	Scalability	Node Authentication	Deployment Knowledge
BROSK [3]	High	No	No
LKMS [4]	Moderate	No	No
Full pair-wise [5]	Low	Yes	No
Q-Composite [6]	Moderate	No	No
Multipath key reinforcement [6]	Moderate	No	No
Pair-wise key establishment [7]	Moderate	No	No
RGM [8]	Moderate	No	No
Blom's Scheme [9]	Moderate	Yes	No
Multiple space key [10]	Low	Yes	No
Gird-based [11]	Moderate	Yes	No
DMBS [12]	Moderate	No	No
GQ design [11]	Moderate	No	No
Group-based deployment [1]	High	No	Yes
Closet pairwise keys [11]	High	Yes	Yes
HGKM [13]	Moderate	Yes	Yes
Matrical Closet Pairwise [14]	Moderate	Yes	Yes
EDDK [15]	High	Yes	Yes
CMKMS [16]	High	Yes	Yes

3. Cluster-based Key Management Scheme

3.1. System Model

Figure 1 presents the proposed system model to design an efficient key management scheme for wireless sensor network. The system model shows a network which is divided into number of clusters. This cluster is used to improve the scalability and energy efficiency of system. Each cluster consists of cluster head which aggregates information from all sensor nodes in the cluster and transfers the aggregated information to the other cluster head or the base station. node to node communication is an intra-cluster communication, done by node to node link.

Transfer between cluster head and cluster head or cluster head and base station is an inter cluster communication which is done by cluster head to cluster head link. The key management algorithm considers the cluster head as key manager. This is based on the assumption that a node may move from one location to another one but the cluster head and the base station are fixed in a location.

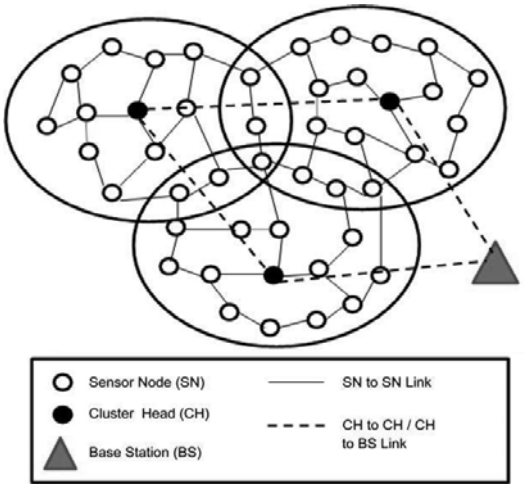

Fig 1 System Model for Key Management

Table 2 shows notations that used to explain the scheme.

Table 2 notations

Notation	Significance
N_i	Node ID
C_i	Cluster ID
K_h	Home Key
K_f	Foreign Key
N_c	Average Number of Nodes Per Cluster
n_c	Number of Cluster in The Network
l	Average Number of Cluster Neighbor
MAC_k	Hash function

3-2- Working Mechanism

The working mechanism of this algorithm is divided into two sections, as follows.

3-2-1- Setup Phase

This phase consists of two parts, which organizing the network into clusters and setting up a cluster keys for each cluster. It is responsible for establishing secure link between clusters to make the whole network connect securely. Here, consider that a unique ID is assigned to each node, which

identifies them distinctly in a network. The algorithm considers that each node maintains two keys (foreign key and home key). K_h or the home key is used to communicate inside its own cluster and cluster head and K_f or the foreign key is used to perform communication with foreign cluster head or nodes during node mobility. These keys will be used for secure information exchange in between nodes.

In the first part (Figure 2) after deployment, each node waits a random time before broadcasting the following Hello message to declare its decision to become a cluster head $E_k\{N_i|K_h.N_i|K_f.MAC_k(N_i|K_h).MAC_k(N_i|K_f)\}$. When a Hello message is received, if the node has decided its role, it rejects all messages to avoid becoming cluster head and member at the same time. If the node has not decided yet, it responds only to the Hello message, cancels timer, sends ACK back and joins the cluster of the node that sent the message. The ACK message contains its ID which is encrypted using both home and foreign keys. Then node set $C_i = N_i$ and set k_h and k_f as cluster keys. The cluster head constructs the polynomial by using $h(x) = f(x) + (K_{hc}^i \oplus K_{fc}^i)$.

The aim of second part (Figure 3) is to make the whole network connect securely. In the algorithm, nodes store cluster key of other neighbor clusters in the form of foreign key K_f. When a node is compromised, it's all neighbor clusters must be evicted from network. To solve this problem, the algorithm generates a unique pairwise key for each neighbor nodes pair. For example, nodes 1 and 2 are neighbor nodes pair in a different cluster, they can establish pairwise key. The pairwise key is generated as follows. Node 1 and 2 exchange their foreign keys encrypted by home key: $E_k\{C_i|K_h|K_f.MAC_k(C_i|K_h|K_f\})$. Nodes in the same cluster will ignore the message, while any nodes from neighboring clusters store $\{C_i|K_h|k_f\}$. The nodes located in two different clusters can compute their pairwise key, such as nodes 1 and 2; they compute their pairwise key by computing: $khf_{ci.cj}^{1.2} = Khf_{ci}^1 \oplus Khf_{cj}^2$. Then neighbor clusters can establish secure link in a network.

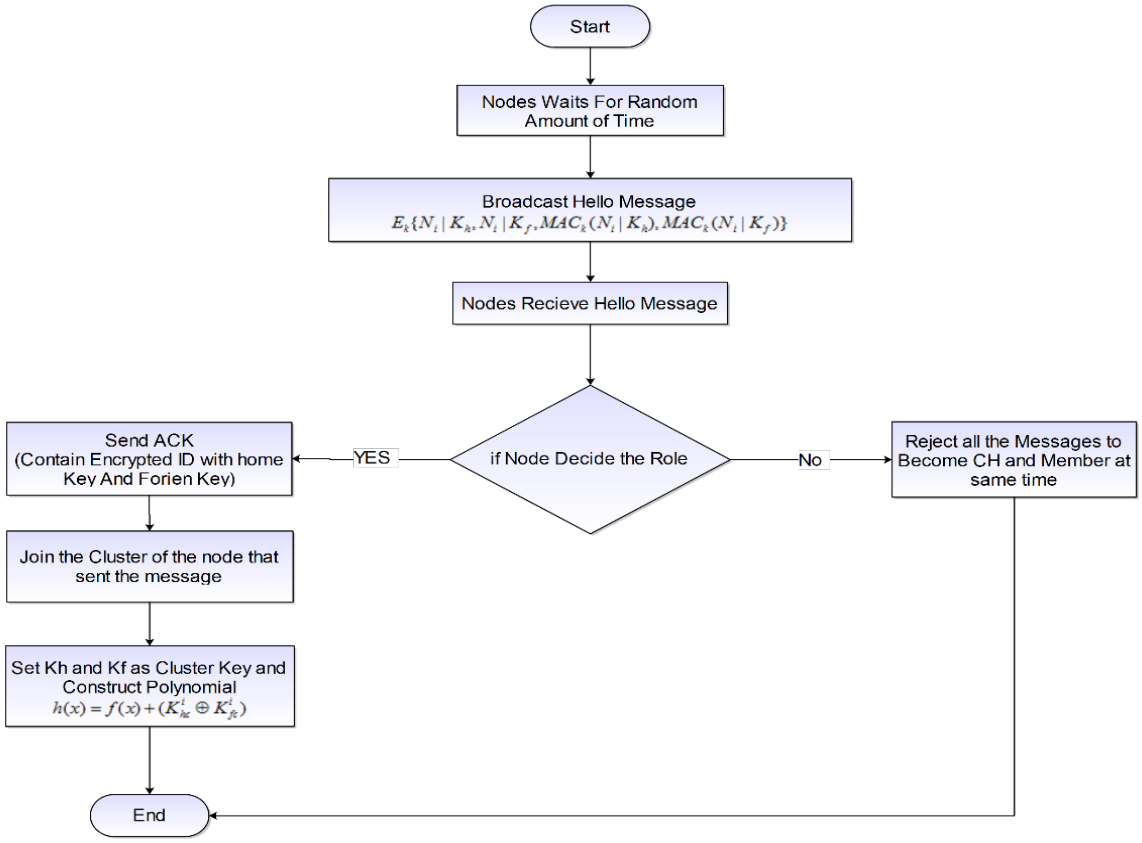

Fig 2 First Part of Setup Phase

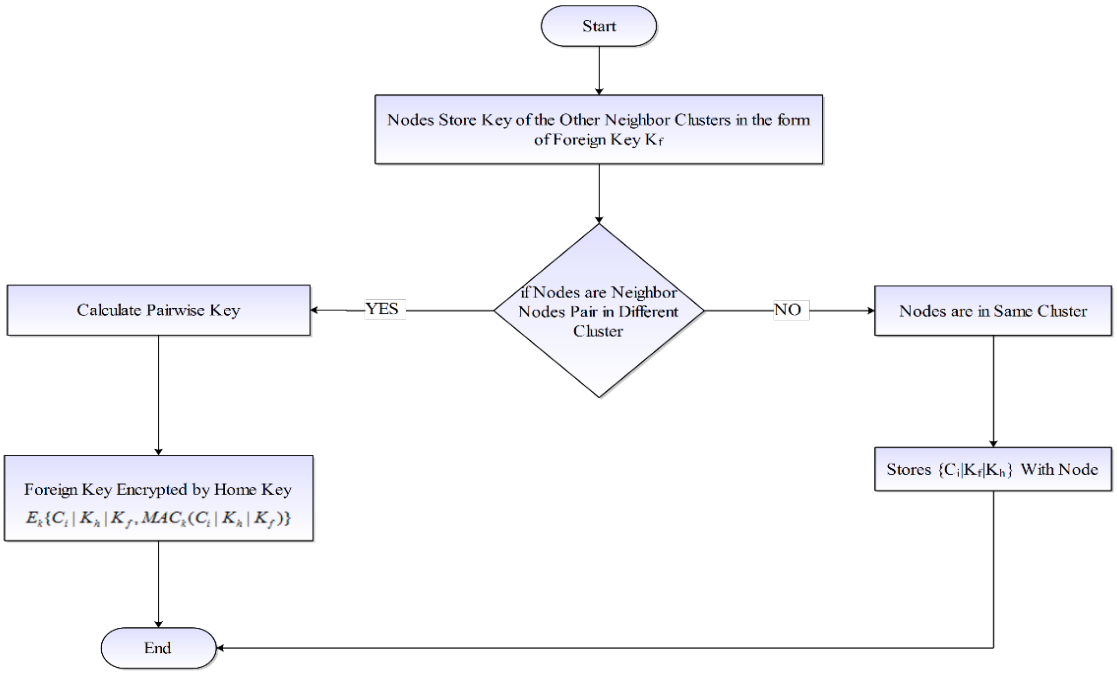

Fig 3 Second Part of Setup Phase

3-2-2- Maintenance Phase

The key maintenance phase tries to maintain keys in following different situations:

- Case 1: when new node joins to the cluster.
- Case 2: when any node moves from one cluster to the other cluster.

As shown in Figure 4, when new members are supposed to join the cluster, it will beacon the message with its ID. The beacon message is received by some neighboring nodes and forwards it to the cluster head or it may also receive it by cluster head directly. Cluster head broadcasts this message to other members and to other cluster heads. When a new member joins the cluster, it will get the home and foreign keys by running setup phase.

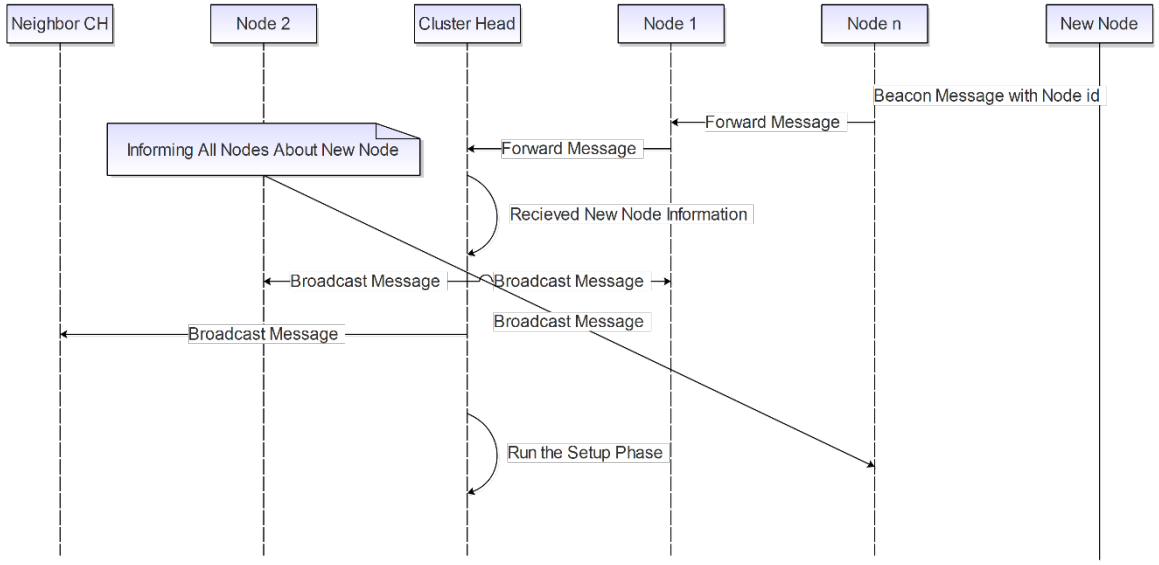

Fig Case 1 Sequence Diagram

As shown in figure 5, when a node wants to move from its cluster, it sends a beacon message to cluster head that it wants to move from one cluster to other. Cluster head updates the information to its members and to the neighboring cluster head that the node is moving from its territory to another one. Therefore, the neighboring cluster is getting understanding that the particular node will communicate with us using other cluster foreign key.

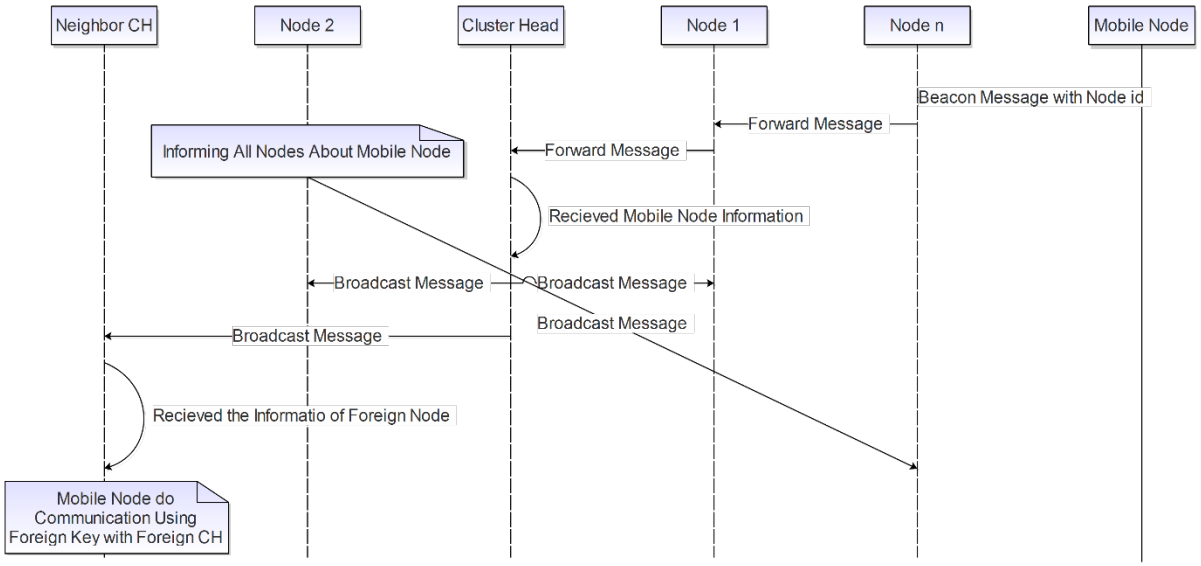

Fig 5 Case 2 Sequence Diagram

4- Simulation and Comparative Evaluation

4-1- Simulation Details

The implementation is performed by using NS-2 simulator [17]. The parameters set during simulations are shown in Table 3. The implementation uses ECDSA [18] to implement encryption and decryption algorithms.

Table 3 Simulation Parameters

Parameter	Value
Network Interface Type	Wireless Physical : 802.15.4
Radio Propagation Model	Two-ray Ground
Antenna	Omni-directional antenna
Channel Type	Wireless Channel
Link Layer	LL
Interface Queue	Priority queue
Buffer Size of IFq	50
Mac	802.15.4
Routing Protocol	Ad-hoc routing
Energy Model	EnergyModel
Initial Energy	100 J
Number of Nodes	From 25 to 250
Node Placement	Random

The performance of the proposed algorithm is compared with the new algorithms EDDK [15] and CMKMS [16], for wireless sensor networks. The simulation of EDDK, CMKMS and the proposed scheme are performed by considering same simulation and node parameters. The performance of the proposed mechanism is measured by using three parameters: computational overheads, average energy consumption and average delay.

4-2- Results and Evaluation

In this paper, the proposed scheme has been simulated using the NS-2 network simulator and its performance is compared with two modern key management solutions EDDK and CMKMS. Major results have been obtained based on computational overhead, average energy consumption and average delay.

In case of key management algorithms, a large number of overheads are incurred during network initialization includes the encryption and authentication of the local broadcast messages, the verification and decryption of the received message from neighbors and calculation of functions. The overheads are majorly measured in terms of computation overheads, which show number of packets transmitted for initialization, average energy and delay incurred for it.

Figures 6 to 8 show the computation overhead in terms of packet transmission, average energy consumption in joules and average delay in millisecond respectively. These three results are measured by varying the number of nodes in the network from 25 to 250.

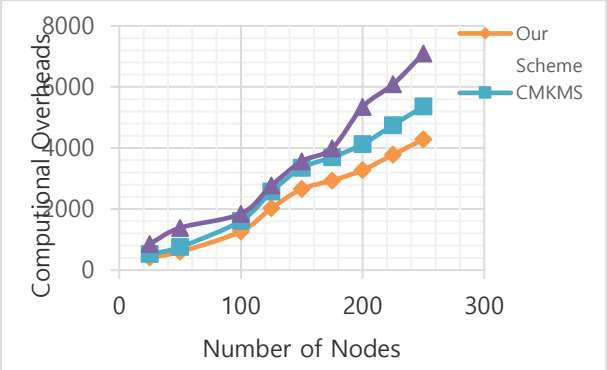

Fig 6 Computational Overhead

The main reason for lower efficiency of EDDK than proposed scheme and CMKMS is EDDK considers local cluster key and pairwise keys for each node, while our proposed scheme and CMKMS consider local and foreign keys for each cluster nodes and pairwise keys only for the common nodes in between clusters. The overheads incurred for establishing local cluster key and pairwise key is more than establishing keys in the proposed scheme.

The reason of lower efficiency of CMKMS than proposed scheme is CMKMS using RC5 algorithm for encryption and decryption, which has lower efficiency compared to the ECDSA algorithm used in our proposed scheme. Moreover, one reason for improved efficiency of the proposed scheme compared to EDDK and CMKMS is that the proposed scheme uses functions with less complexity.

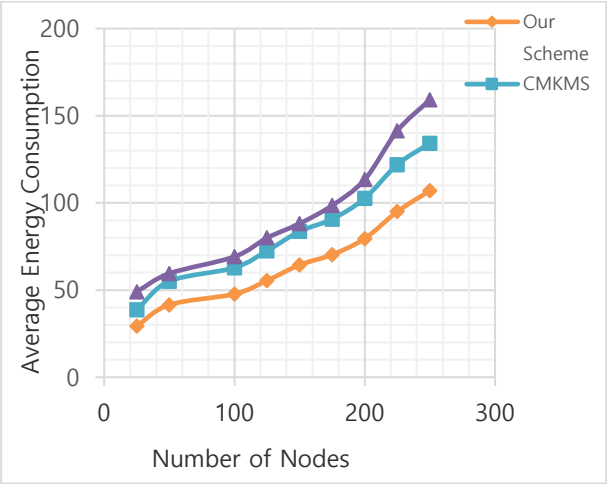

Fig 7 Average Energy Consumption

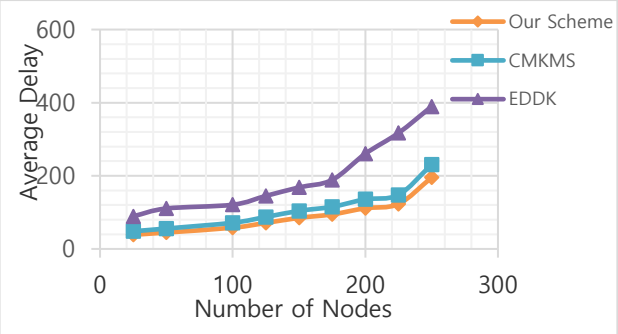

Fig 8 Average Delay

The main reasons of increasing performance overhead when cluster head goes mobile are as follows:

- It changes it neighborhood and the change in neighborhood effect on calculating of keys, pairwise and individual keys.
- It also reflects in reselection of cluster heads.

Here, the performance of EDDK is lower than the proposed scheme and CMKMS. The major reason of lower performance of EDDK is that it calculates the pairwise keys and change in neighborhood effect on calculation of pairwise keys, which may give wrong instance of pairwise keys and need the recalculation of pairwise keys.

5- Conclusion

Due to the sensitivity of wireless sensor networks applications and lack of support of constant infrastructure and also storage restrictions such as processing speed, storage size and energy in wireless sensor nodes, performing security mechanisms such as key management are mentioned as challenging issues in wireless sensor networks. One of the main concerns in designing a key management scheme in wireless sensor networks is scalability.

The paper surveys the different key management algorithm in wireless sensor network and proposes a novel key management scheme to improve scalability, security and mobility requirements by reducing the computational complexity of the algorithm. The scheme runs in two phases, first phase is setup the cluster and assign the home and foreign keys to each node. The second phase maintains the key during the node and cluster head mobility. Moreover, to reduce energy consumption and computational overhead and to improve encryption speed ECDSA encryption algorithm has been used.

Our proposed scheme has lower computational overhead, average energy consumption, and average delay as compared with new solutions of EDDK and CMKMS. Our proposed scheme shows 15-25 percent improvements compared with other algorithms.

References

[1] Y. Xiao, V. K. Rayib, B. Sun, D. Xiaojiang, F. Hu and M. Galloway, "A Survey of Key Management Schemes in Wireless Sensor Networks," Computer Communications, vol. 30, no. 12, pp. 2314-2341, 2007.

[2] J. Zhang and V. Varadharajan, "Wireless Sensor Network Key Management Survey And Taxonomy," Network and Computer Applications, vol. 33, no. 2, pp. 63-75, 2010.

[3] B. Lai, S. Kim and I. Verbauwhede, "Scalable session key construction protocol for wireless sensor networks," in IEEE Workshop on Large Scale Real-Time and Embedded Systems (LARTES), Washington, 2002.

[4] B. Dutertre, S. Cheung and J. Levy, "Lightweight key management in wireless sensor networks by leveraging initial trust," Technical Report SRI-SDL-04-02, 2004.

[5] H. Chan, V. D. Gligorg, A. Perrig and G. Muralidharan, "In the distribution and revocation of cryptographic keys in sensor networks," Dependable and Secure Computing, IEEE Transactions on, vol. 2, no. 3, pp. 233-247, 2005.

[6] H. Chan, A. Perrig and D. Song, "Random key pre-distribution schemes for sensor networks," in Security and Privacy 2003, Proceedings of IEEE Symposium, Washington, 2003.

[7] Z. Sencun, X. Shouhuai, S. Setia and S. Jajodia, "Establishing pairwise keys for secure communication in ad hoc networks: a probabilistic approach," in Network Protocols 2003, Proceedings of the 11th IEEE International Conference, Washington, 2003.

[8] M. Ergun, A. Levi and E. Savas, "A resilient key predistribution scheme for multiphase wireless sensor networks," in Computer and Information Sciences, ISCIS 2009, 24th International Symposium, Washington, 2009.

[9] R. Blom, "An optimal class of symmetric key generation systems," in EUROCRYPT 84 Workshop on Advances in Cryptology: Theory and Application of Cryptographic Techniques, New York, 1985.

[10] D. Wenliang, J. Deng, Y. S. Han and P. K. Varshney, "A pairwise key pre-distribution scheme for wireless sensor networks," in Computer and communications security CCS 2003, Proceedings of the 10th ACM Conference, New York, 2003.

[11] D. Liu, P. Ning and R. Li, "Establishing pairwise keys in distributed sensor networks," Journal ACM Transactions on Information and Syststem Security, vol. 8, no. 1, pp. 41-77, 2005.

[12] D. Wenliang, J. Deng, Y. S. Han, S. Chen and P. K. Varshney, "A key management scheme for wireless sensor networks using deployment knowledge," in INFOCOM 2004, Proceedings of 23rd Annual Joint Conference, Los Alamitos, 2004.

[13] N. T. Canh, Y. K. Lee and S. Lee, "HGKM: a group-based key management scheme for sensor networks using deployment knowledge," in the Sixth Annual conference, Los Alamitos, 2008.

[14] Y. Zhen and Y. Guan, "A key management scheme using deployment knowledge for wireless sensor networks," Journal of Parallel and Distributed Systems, IEEE Transactions, vol. 19, no. 10, pp. 1411-1425, 2008.

[15] X. Zhang, J. He and Q. Wei, "EDDK: Energy-Efficient Distributed Deterministic Key Management forWireless Sensor Networks," Wireless Communications and Networking, vol. 2011, no. 12, pp. 1-11, 2011.

[16] S. Dilip Babar, N. Rashmi Prasad and R. Prasad, "CMKMS: Cluster-based mobile key management scheme for wireless sensor network," International Journal of Pervasive Computing and Communications, vol. 10, no. 2, pp. 196-211, 2014.

[17] "NS-2," The network simulator-ns-2, 2005. [Online]. Available: www.isi.edu/nsnam/ns/. [Accessed 21 07 2016].

[18] S. Vanstone, "Responses to NIST's proposal," CACM, vol. 35, no. 7, pp. 50-52, 1992.

Seyed Reza Nabavi received his B.S. degree in software engineering from Amirkabir college of technology, Arak, in 2014. He received his M.S. degree in software engineering from Arak Branch, Islamic Azad University, in 2016. He was leader of multiple research projects, the translator of multiple textbooks in Persian, multiple journal and conference papers. He is a member of Young Researchers and Elite Club of Islamic Azad University. His research interests include sensor networks, load balancing algorithm, real time, distributed, parallel and fault-tolerant systems.

Seyed Morteza Mousavi received his B.S. degree, M.S. degree and Ph.D. degree in computer system architecture from Iran. He has been working as a lecturer and a faculty member with the department of computer engineering, Arak Branch, Islamic Azad University. He was leader of multiple research projects. He has also published many papers on international conferences and journals and is the author of several books in the field of computer engineering. His research interests include sensor networks, network security, computer architecture and cryptography.