TELKOMNIKA Indonesian Journal of Electrical Engineering
Vol.12, No.5, May 2014, pp. 3969 ~ 3978
DOI: http://dx.doi.org/10.11591/telkomnika.v12i5.4241

An Improved Key Management Scheme for Hierarchical Wireless Sensors Networks

Abdoulaye Diop*, Yue Qi, Qin Wang
School of Computer and Communication Engineering
University of Science and Technology Beijing, Beijing, China.
Corresponding author, e-mail: adiop.ustb@gmail.com*, qiyuee@ustb.edu.cn, wangqin@ies.ustb.edu.cn

Abstract

Key management play a central role for protecting communication in WSNs. Due to the limited memory resources and energy constraints, complex security algorithms cannot be used in sensor networks. Therefore, to secure data communication and well balance between the security level and the associated energy consumption is a challenging task. In this paper, we present an Improved Key management Scheme for Securing communication in Hierarchical Wireless Sensors Networks (IKS). The proposed technique based on symmetric key mechanism, generates and distributes the keys within a cluster efficiently and updates periodically keys to mitigate the node compromise attack. We provide a detailed security analysis of our IKS protocol and show its advantages in avoiding node capture attack and several serious attacks from malicious nodes. Finally, using NS-2 simulator, the results shows that IKS provides energy saving and has low communication overhead and end to end delay compared to existing key management schemes.

Keywords*: wireless sensor network, hierarchical networks, key management, security, attacks.*

1. Introduction

With the design and development of micro devices, the communication technology enabled the design and development of WSNs with low cost, low energy consumption and high utilization. WSNs have lot of applications in military, health and other industrial sectors. Because of the characteristics of WSNs, sensor nodes are usually characterized by limited power, low bandwidth, memory size and limited energy [1, 5].

Cluster based sensors networks is one of the main research areas in WSNs and behave better in performance and reliability than traditional flat WSNs (FSNs). Cluster topology is used to extend the lifetime of WSNs and can provide scalability, good organization and energy efficient. Many routing protocols for cluster-based WSNs have been proposed by researchers [2, 6]. In cluster-based routing protocols, network is divided into cluster and each cluster has its own Cluster Head (CH). Further, CHs are responsible for relaying of messages from ordinary nodes to the Base Station (BS).

To achieve both security and efficiency for WSNs, several key distribution and management schemes have been proposed in WSNs. The first key pre-distribution scheme was presented by Eschenauer and Gligor [7]. They propose a probabilistic key pre-distribution technique, where two sensor nodes need to identify the common keys they share, to establish a pair-wise key between them. However this scheme cannot provide sufficient security when the number of compromised nodes increases. Zhu et al. [9] propose Localized Encryption and Authentication Protocol (LEAP), which establishes four types of keys that must be stored in each sensor. One weakness of this approach is that once the initial key is compromise, an adversary can deduce all the pair-wise keys installed in the network [8]. Y. Cheng and D. Agrawal propose IKDM (an Improved Key Distribution Mechanism) [10] based on hierarchical network architecture and bivariate polynomial-key pre-distribution mechanism.

In IKDM, Only two pair-wise keys are preloaded in each sensor node to reduce the key storage overhead. For securing LEACH (Low-Energy Adaptive Clustering Hierarchy) presented by Heinzelman et al. [2], some secure routing protocols have been proposed, such as SecLEACH [11], GS-LEACH [12] and SLEACH [13].

Received October 13, 2013; Revised December 20, 2013; Accepted January 9, 2014

SecLEACH show how a random key pre-distribution can be used for secure communication in cluster-based protocols. However, GS-LEACH, SLEACH and SecLEACH present some security vulnerabilities caused by the random key pre-distribution scheme and are also vulnerable to key collision attacks. Most of these schemes are vulnerable to a number of security threats [3]. Further, if a node gets compromised, it is possible for the adversary to know all the keys stored in the node.

In this paper, we propose an Improved Key management Scheme for Securing communication in Hierarchical Wireless Sensors Networks (IKS) to overcome the limitations of current key distribution and management schemes. Based on the hierarchical network structure, IKS use different kind of keys to ensure basic security requirements. Data encryption is performed for secure communication. One way hash function and Message Authentication Code (MAC) are also used to provide authentication and message integrity. The proposed technique based on symmetric key mechanism, generates and distributes efficiently the keys within a cluster and updates periodically keys to mitigate the node compromise attack [4].

Indeed, if an intruder manages to capture a node, an encryption mechanism should be present to restrict the access of intruder to the message history of node [5, 16]. Therefore, after key establishment phase, a key updating phase should be used to update the keys regularly. After certain time interval new nodes are selected as CH, and BS generates a new key using the hash function and the current key. This procedure ensures that intruders cannot acquire the keys easily, hence avoid different types of attacks from malicious nodes, because only legitimate nodes can join the network.

The rest of the paper is organized as follows. Section 2 describes the network model. Section 3 explains the proposed key management scheme in details. In Section 4, we present the security analysis and simulation results of the proposed key management scheme. Finally, we conclude our work and present some future research directions in section 5.

2. Network Model

We focus on hierarchical structure of sensor network [10], as illustrated in Figure 1.

a) BS is considered trustworthy with unlimited resources and is located in a safe place. BS has authentication system for any node in the network [15], a node member table of all nodes in the network and an intrusion detection system.

b) Sensors nodes collect information of surrounding environment and transmit them to the cluster head.

c) CH is responsible for collecting data within a cluster and transmission to the BS.

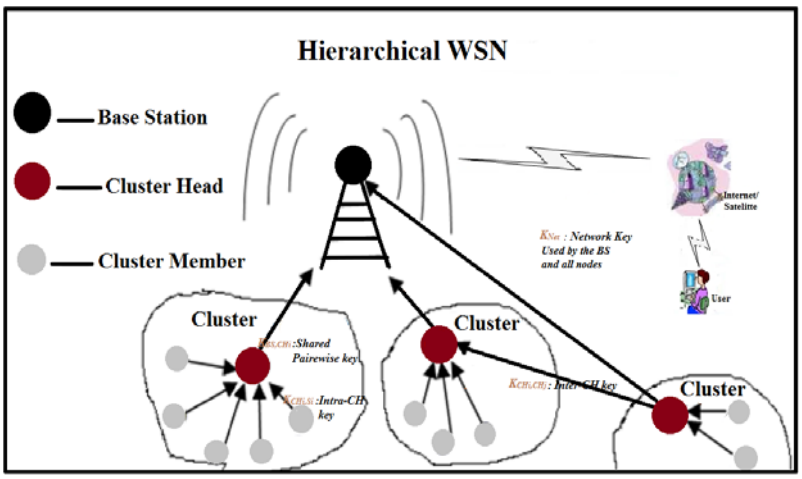

Figure 1. Proposed Scheme architecture.

Descriptions of the notations used in the proposed key management technique are listed in Table 1.

Table 1. Notation Description used in IKS

S. No	Notation	Description
1.	idSNi	Identification Number of node i
2.	idCHi	Identification Cluster Head i
3.	idBS	Identification Base Station
4.	KNet	Network key
5.	KBS,CHi	Shared Pair-wise Key
6.	KCHi,Si	Intra-cluster key
7.	KCHi,CHj	Inter-cluster key
8.	KI	Initial key
9.	EK(M)	Encryption of message M with symmetric key K
10.	V	An array of node ids
11.	H ( )	One-way hash function
12.	MACK()	The message authentication code of message M using symmetric key K
13.	⊕	Bit wise XOR operation
14.	N	Nonce

Initially, we consider that WSNs are homogeneous and symmetric.

Sensor nodes keep stationary after deployment during the network operation. To distinguish between them, each node has a unique id with enough length.

In our scheme, we use Low-Energy Adaptive Clustering Hierarchy (LEACH) [2] to randomly choose CHs. Sensors nodes choose their cluster head according some parameters such as the strongest signal received [2]. As shown in Figure 1, there is no communication between sensors nodes. After certain time interval new nodes are selected as CH to provide energy saving of a cluster head [2].

We assume that when CH is located far from the BS, CHs can communicate each other. In this case, CH sends the aggregated sensing data to the relay CH near the BS to save energy.

In this network model, each exchanged message has a timestamp that guarantee the freshness of information. We also consider a minimum time "Tmin" after deployment, in which a node cannot be compromised.

Further, as threat model we assume that an adversary can eavesdrop on the traffic, inject new messages, spoof other identities, replay and modification of old messages. However we consider that an adversary need at least time "Tcapture" to compromise a node.

3. The Proposed Hierarchical Key Management Scheme

3.1. Key pre-distribution phase

In our proposed scheme, each node are preloaded with one unique secret key K_I, shared with the BS before they are deployed. Nodes must authenticate themselves with the BS using their corresponding unique key K_I. Hence node send request to the BS consisting of sensor node's id, nonce, and MAC, where MAC is calculated by using K_I. The BS authenticates the nodes by verifying the MAC.

Afterwards, if authentication is successful, the BS generates a network key K_{Net} for sensor node with id_{SNi} and loads each node with this Network key. K_{Net} will be used during the initial cluster formation phase. Note that all members should prove their validity to the BS. Further K_I is deleted from SN's memory after joining the network.

3.2. Key Establishment

Shared Pair-wise Key Establishment ($K_{BS,CHi}$): After the deployment, some nodes are randomly selected as CH, hence BS needs to establish pair-wise key with each CH to secure the communication between them. The BS generates an array V of all sensors nodes id_{SNi} in the network. The BS first using the network key K_{Net} encrypts a threshold value T(n), generates a MAC and broadcasts these information with a nonce to all sensor nodes. Node generates a random number R between 0 and 1. If R is less than a given threshold T(n), the node acts as a cluster head.

T(n) is calculated as:

$$T(n) = \begin{cases} \dfrac{p}{N - p * \left(r \bmod \frac{N}{p}\right)} & n \in G \\ 0 & other \end{cases} \quad (1)$$

Where p is the percentage of cluster heads, r is the current round number, G is the set of nodes which haven't been elected as cluster-heads in the last r mod (N / p) rounds.

a) When a node SNi becomes CH first time, it sends an authentication packet to the BS By inserting its id and encrypting message using K_{Net}.

$$CHi \rightarrow BS : \quad idCHi , idBS \,||\, E\,K_{Net}(M|N) \,||\, MAC\,K_{Net}\,(M|N)$$

Where MAC ensures data integrity and authentication, the timestamp N avoids message replay attack. M is the cluster head's message $(M= id_{CHi}|id_{BS}|\, K_{Net})$.

b) Upon receiving the CH information, the BS authenticates M and verifies the MAC. If CH is a valid node, BS computes a new key $K_{BS,CHi}$ by using a keyed one-way hash function $H_K(val)$. BS encrypts M and $K_{BS,CHi}$ using network key K_{Net} and sends it to CHi.

$$BS \rightarrow CHi : \quad idCHi , idBS \,||\, EK_{Net}(M|N|\, K_{BS,CHi}) \,||\, MACK_{BS,CHi}\,(M|N)$$

Intra-cluster key establishment ($K_{CHi,Si}$): Each CH needs to establish a shared key with its cluster member SNi to ensure secure communication between them (as illustrated Figure 2). This key establishment can be briefly described as follows:

a) First, Each CH broadcasts an advertisement message M using KNet, id_{CHi} and a timestamp N to avoid replay attack.

$$CHi \Rightarrow SNi: \quad idCHi \,||\, EK_{Net}\,(M|N) \,||\, MACK_{Net}\,(M|N)$$

b) Node SNi authenticates CHi by verifying the MAC, using the network key K_{Net}. A node SNi joins a cluster based on the received signal strength [2]. Then, for membership of this cluster, a node generates a message M as follows: $M= id_{SNi}|id_{CHI}|K_{Net}$

c) Now node SNi encrypts the message M using K_{Net}, includes the timestamp N and sends the encrypted message to the selected CHi.

$$SNi \rightarrow CHi: \quad idSNi, idCHi \,||\, EK_{Net}\,(M|N) \,||\, MAC\,K_{Net}\,(M|N)$$

d) Afterward, CHi sends the identity list (idList) of each node member in the cluster to the BS.

$$CHi \rightarrow BS: \quad idCHi, idBS \,||\, EK_{BS,CHi}\,(M|N \,|\, idList\,) \,||\, K_{BS,CHi}\,(M|N)$$

Where idList = {idSN1,idSN2,…… ,idSNk-1}, k is the number of node in the cluster and M is the cluster head message.

e) BS computes the cluster key $K_{CHi,Si}$ using a one-way hash function and the pair-wise key $K_{BS,CHi}$ and send it to the CH with an authentication response message.

$$BS \rightarrow CHi: \quad idBS, idCHi \,||\, EK_{BS,CHi}\,(M|N|\, K_{CHi,Si}) \,||\, MACK_{BS,CHi}\,(M|N)$$

f) Each CH is responsible for distributing this shared key to its cluster member SNi. Therefore CHi authenticates BS, decrypts the message and send the intra-cluster key $K_{CHi,Si}$ to all cluster members .

$$CHi \rightarrow SNi: \quad idCHi \,||\, EK_{Net}\,(M|N \,|\, K_{CHi,Si}\,) \,||\, MACK_{Net}\,(M|N)$$

g) The cluster members authenticate M by verifying the MAC, if authentication is valid, $K_{CHi,Si}$ will be act as the shared key between CH and cluster members.

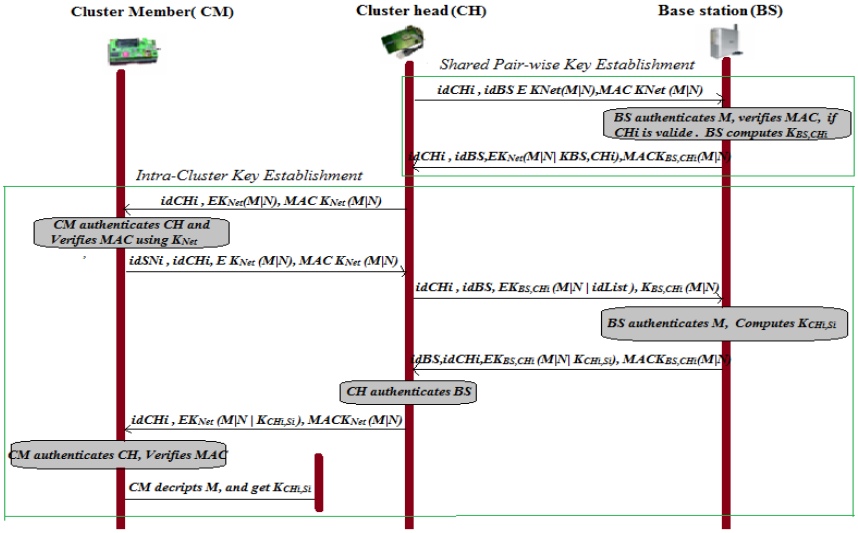

Figure 2. Keys Generation and Distribution in IKS

Inter-cluster key establishment ($K_{CHi,CHj}$): In Inter cluster communication, the source CH (located far from the BS), communicates with BS through the relay CH along the path. The key sharing scheme contains the following steps:

Source CH sends a request for inter-cluster key generation to the BS with the list of CHs along the path of communication.

Upon receiving the request from the initiating CH, the BS authenticates the CH's message and generates shared key for inter-cluster communication between CHi and CHj. Afterward BS encrypts $K_{CHi,CHj}$ using $K_{BS,CHi}$ and sends to the correspondings CHs. Thus CHs use the inter-cluster key to communicate in a secure way.

3.3. Data Transmission Phase

a) This phase mainly consists of two distinct steps in hierarchical sensor network. Firstly, sensor nodes send encrypted data packets to it corresponding CH as follows:

$$SNi \rightarrow CHi: \quad idSNi, idCHi \,||\, E\,K_{CHi,Si}(M) \,||\, MACK_{CHi,Si}(M)$$

Where M is the sense data.

b) The confidentiality of the message is ensured by using $K_{CHi,Si}$. Afterward CH sends encrypted aggregate data packets to BS for processing, encrypted by the pair-wise key $K_{BS,CHi}$.

$$CHi \rightarrow BS: \quad idCHi \,||\, EK_{BS,CHi}(H(M1, Mj, \ldots.., Mn))$$

3.4. Key Updating Phase

To reduce the risk of node capture attacks, it is essential to update the keys [9]. Hence the network key K_{Net} is updated periodically. This key is valid only for a limited time period that is less than the predicted time required for node compromise ($T_{capture}$). That period of time is dependent on the network environment.

Thus BS generates and sends the new network key K_{Net+1} to CHi encrypted with $K_{CHi,BS}$.

$$BS \rightarrow CHi: \quad idBS, idCHi \,||\, EK_{BS,CHi}(M|N|\,K_{Net+1}) \,||\, MACK_{BS,CHi}(M|N)$$

Upon receiving the message, CHi authenticates BS, decrypts and broadcasts the informations to its cluster members.

$$CHi \rightarrow SNi: \quad idCHi, idSNi \,||\, EK_{CHi,Si}(M|N|\,K_{Net+1}) \,||\, MACK_{CHi,Si}(M|N)$$

The legitimate cluster members receive the broadcast message, authenticates CHi by verifying MAC, decrypt it using the current network key and get the new network key.

The intra-cluster keys can also be refreshed periodically. In this case BS using the hash function (H) and the current cluster key, generates a new cluster key. The messages are encrypted with $K_{CHi,BS}$ and sent to CHi.

$$BS \rightarrow CHi: \quad idBS, idCHi \ || \ EK_{BS,CHi}(M|N|\ \boldsymbol{K'}_{CHi,Si}) \ || \ MACK_{BS,CHi}(M|N)$$

CHi authenticates BS and transmits the new intra-CH key to its cluster members, encrypted with the current $K_{CHi,S}$ only known by the legitimate cluster members.

$$CHi \rightarrow SNi: \quad idCHi, idSNi \ || \ EK_{CHi,Si}(M|N|\ \boldsymbol{K'}_{CHi,Si}) \ || \ MACK_{CHi,Si}(M|N)$$

It is worth noting that, sensor nodes SNi or CHi cannot uncover the new cluster key since it does not know the current ones.

3.5. Node Compromise and Cluster Re-organization

In WSNs, in the case of a node compromise, it is necessary to preserve the shared keys secrecy, hence avoid that the number of compromised nodes reached a critical value. In this scheme, upon identifying a compromised sensor node, CH broadcasts a notification to its cluster members, and removes the compromised node from its cluster member table.

If a CH is compromised, a re-clustering of cluster member of the compromised CH among the remaining CH needs to take place. In this case, BS informs its cluster members and other CHs. Cluster member of the compromised CH are distributed among other uncompromised CHs. It is worth noting that to distribute the nodes among themselves, the CHs use the clustering algorithm as discussed in [2, 14]. Afterward BS initiates the key update mechanism, however nodes discards its currents keys and uses a new network key and cluster key for future communication. Further, the news keys are used to secure all the communications in the network between the communicating parties.

It is worth noting that in IKS, as long as the duration of an epoch is less than the key compromising time, adversary can not compromise the keys, therefore confidentiality and integrity will be still guaranteed in IKS.

In this proposed model of key management technique, we consider that cluster heads are rotated after certain time interval [2], and all nodes get a chance to be a cluster head equal number of times. This approach allows balancing the energy consumption among all nodes in the network.

BS broadcast a packet to all CHs at the end of cluster duration to remove its member table. New cluster head create a table of its member node when a new cluster goes on, as described in intra-cluster key establishment, forward it to the BS and continue its process.

4. Security Analysis and Simulation Results

Here, we evaluate the security properties and network performance of our IKS and we compare it with some of existing schemes.

4.1. Security Analysis of IKS

In IKS, assuming the BS is trustworthy, key can be safely established between the CH and the BS, and between CH and cluster members. Before the transmission of the message, encryption is performed to secure the communication with the help of one way hash function. One-way hash function is used to provide authentication and message integrity.

Protocols in cluster based WSNs mitigate most attacks except Sinkhole, Wormhole attacks.

In IKS, each cluster members encrypts information using $K_{CHi,Si}$, avoiding eavesdropping attacks. Only legitimate CH that owns the cluster key can decrypt the message. IKS provides freshness using time interval, time-stamps and nonce. The nonce N is very important since it prevents a replay attack and ensures the integrity of the message. Hence an external attacker cannot modify or inject routing information without being detected. Further to

know the origination of the message for further action, BS checks the CH id which is attached to the message. To prevent a malicious node to attemp keys establishment, the BS authenticate CH by verifying the MAC calculated using K_{Net}. Since K_{Net} is only known by the BS and legitimate nodes. The MAC ensures the data integrity and authentication of sensing data. It is worth noting that after the keys establishment, keys will be used to encrypt the transmitted data for the next transmission, hence will ensure secure communication. Further, if we expect that the attacker requires a fixed amount of time to compromise the node, the keys would have changed to a new one before the attacker could use the compromised keys.

Hence, changing periodically the keys, avoids eavesdrop attack and provides secure communication in our Scheme.

Further, if compromised node sends data using the previous key, BS will reject all data that have received from the malicious nodes. The whole process ensures that malicious nodes will not be authenticated by the BS. Hence, as long as the duration of an epoch is less than the key compromising time, our proposed scheme is secure. Therefore, sinkhole attack, wormhole attack, selective forwarding attack fail against IKS.

Table 2. Keys Storage and Security Mechanisms

Keys Storage & Security Mechanisms	Protocol Name SLEACH	SecLEAH	IKS
Cryptography Schme	Symmetric Cryptography	Symmetric Cryptography	Symmetric Cryptography, AES
Key Management Scheme		Random key predistribution scheme	Key Management for Hierrarchical WSNs
Authentication Scheme	MAC	Don't provide broadcasts authentication	MAC
Key Storage overhead	m keys × key size	m keys × key size	(2 keys for CM + 4 keys for CH) × key size

Table 3. Security and Performance requirement Comparaison

Attacks Types & Performances Requirements	Protocol Name GSLEACH	SLEACH	SecLEAH	IKS
Node Capture attack	×	×	×	√
Sinkhole attack	×	√	×	√
Wormhole attack	×	×	×	√
Seclective forwarding	×	√	√	√
Energy Efficient	Good	Medium	Medium	Good
Storage Load	High	High	High	Low
Connectivity	Medium	Medium	Medium	Full
Scalability	Medium	Medium	Medium	Good
Robustness	Limited	Limited	Limited	Good

IKS as compared to other key pre-distribution schemes such as SecLEACH [11], GS-LEACH [12] and SLEACH [13] based on LEACH and random key pre-distribution provides efficient security. GS-LEACH, SLEACH and SecLEACH present some security vulnerabilities caused by the random key pre-distribution scheme and are also vulnerable to key collision attacks. Security mechanisms, resilience against attacks and Performance requirements Comparaison are presented in Table 2 and Table 3.

IKS allows every entity in the network to be confirmed or authenticated continuously and reduces the chances of node compromise. Further, IKS satisfies general security requirements, such as confidentiality with encryption, message integrity with MAC, node authentication as mentioned before, message freshness of messages exchanged in the network with nonce and full confidentiality.

4.2. Simulation Results

To evaluate the proposed Key Management technique, we compare our IKS scheme based on hierarchical network with LEACH protocol using the NS2 simulator [17]. We consider

a random network of 250 sensor nodes deployed in an area of 200x200m.The sink node is assumed to be near the sensing. Simulation time was 400 seconds. Advanced Encryption Standard (AES) [18] (block size of 128 bits) was used to implement the encryption/decryption algorithm. Due to execution time and energy consumption requirement of AES is much less than other cryptography algorithms [18].

In order, to evaluate the performance of the security overhead, we consider two metrics: the energy consumption and the End to End delay. We measured the average energy consumption of sensor nodes following different network size.

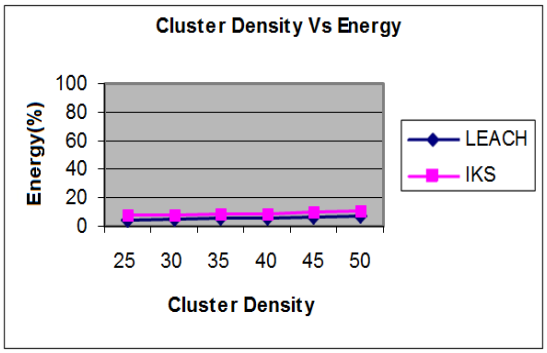

Figure 3. Cluster Density Vs %Energy Consumed by a CH

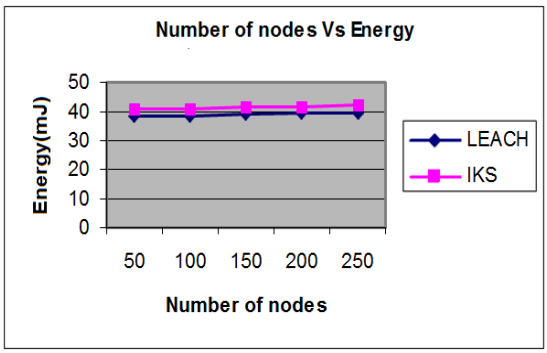

Figure 4. Number of nodes Vs Energy Consumed by Sensor Nodes

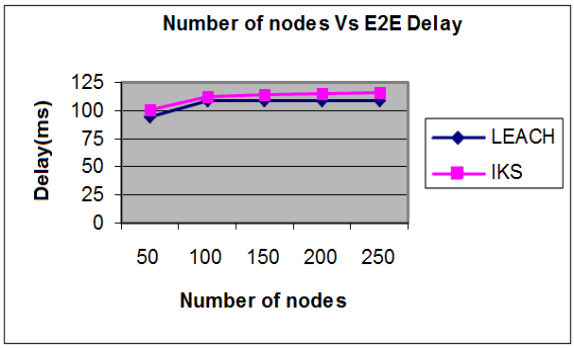

Figure 5. Number of Nodes Vs End to End Delay

Figure 3 shows that with the increase number of nodes from 50 nodes to 250 nodes, the energy consumption of IKS protocol is slightly greater by 1.45% to 1.65% when compared with LEACH because of the communication overhead. The graph also indicates that for both schemes, the average energy is almost constant for varying network sizen and the gap between IKS and LEACH is extremely low and practically identical. This result was expected because in IKS, cluster members communicate only with the cluster head, each ordinary node sends one message and receives one message. This is because, in IKS data are transmitted via one hop between cluster members and CH. Therefore, provides energy saving.

Figure 4 shows the Average energy percentage consumed by a cluster head over cluster densities. Both models show that the energy consumption of CH increases with cluster density. This is because increasing the number of messages add significant cost to the cluster energy. In a network of 250 sensors and cluster density of 25 nodes, the average energy consumption of CH in IKS is more than 2.15% compared to LEACH, because of the computation overhead.

Figure 5 illustrates the average end-to-end delay time following the number of nodes. Defined as the time taken by a packet to reach the destination from the source. We can notice that the E2E delay in our IKS scheme is slightly greater compared to LEACH. Therefore we can conclude that the security mechanism used in IKS does not take a lot of time. We also notice

that, increasing the number of node in LEACH as well as in IKS, the E2E delay remains almost unchanged. This is because non-CH communicates only with its corresponding CH with one hop.

5. Conclusion

In this paper, we presented an Improved Key management Scheme for Securing communication in Hierarchical Wireless Sensors Networks (IKS). We find that the overhead which the IKS protocol leads to be acceptable with low memory overhead and E2E delay. Based on the hierarchical network structure, IKS distribute the keys within a cluster and provides rekeying process to enhance network security avoiding node-capturing problem and assure that only legitimate nodes send data for processing.

Before message encryption, communicating parties need to be in agreement on a key, hence provides continuous authentication of nodes in the network. IKS mechanism is scalable with few messages unlike to random key predistribution schemes based on key pools which generate a lot of message, with high storage overhead. Simulation and analysis has shown that IKS approach not only achieves efficient security, but also provides energy saving. Our Future works may concentrate on developping a complete security scheme for cluster based WSNs under mobility to deal with varied and complex attacks.

Acknowledgements

The work reported in this paper was supported by National Natural Science Foundation of China (No. 61172049, 61003251), National High Technology Research and Development Program of China (Grant No. 2011AA040101-3), Doctoral Fund of Ministry of Education of China (No. 20100006110015).

References

[1] J Zhang, V Varadharajan. Wireless sensor network key management survey and taxonomy. *Journal of Network and Computer Applications*. 2010; 33(2): 63-75.
[2] W Heinzelman, A Chandrakasan, H Balakrishnan. *Energy-efficient communication protocol for WSNs,* Proc. of the 33rd Hawaii International Conference on System Sciences, Washington. 2000.
[3] A Diop, Y Qi, Q Wang, S Hussain. An Advanced Survey on Secure Energy-Efficient Hierarchical Routing Protocols in Wireless Sensor Networks. *International Journal of Computer Science Issues.* 2013; 10(1-2): 490-500.
[4] J Lee, V Leung, K Wong, J Cao, H Chan. *Key management issues in wireless sensor networks: current proposals and future developments*. IEEE Wireless Communications. 2007; 14(5): 76 –84.
[5] MA Simplício Jr, PS Barreto, CB Margi, TC Carvalho. A Survey on Key Management Mechanisms for Distributed Wireless Sensor Networks. *Computer Networks*. 2010; 54(15): 2591-2612.
[6] P Zhu, F Jia. A New Approach to Sensor Energy Saving Algorithm. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(5): 2485-2489.
[7] L Eschenauer, VD Gligor. *A key management scheme for distributed sensor networks*. Proc. of the 9th ACM conference on Computer and communications security, New York. 2002: 41-47.
[8] AH Sodhro, Y Li, M Ali Shah. Novel Key Storage and Management Solution for the Security of Wireless Sensor Networks. *TELKOMNIKA Indonesian Journal of Electrical Engineering.* 2013; 11(6): 3383-3390.
[9] S Zhu, S Setia, S Jajodia. *LEAP: Efficient Security Mechanisms for Large-Scale Distributed Sensor Networks*. Proceedings of the 10th ACM conference on Computer and communications security, New York. 2003: 62–72.
[10] Y Cheng, D Agrawal. An improved key distribution mechanism for large-scale hierarchical wireless sensor networks. *Ad Hoc Networks (Elsevier)*. 2007; 5(1): 35–48.
[11] LB Oliveira, A Ferreira, MA Vilaca, et al. SecLEACH-on the security of clustered sensor networks. *Signal Processing*. 2007; 87(12): 2882-2895.
[12] P Banerjee, D Jacobson, SN Lahiri. *Security and performance analysis of a secure clustering protocol for sensor networks*. Proc. 6th IEEE Intl. Symposium on Network Computing and Applications. 2007: 145-152.
[13] AC Ferreira, MA Vilaca, LB Oliveira, E Habib, HC Wong, AA Loureiro. *The security of cluster-based communication protocols for wireless sensor networks*. Proc. 4th IEEE International Conference on Networking (ICNŠ05). 2005: 449–458.

[14] PK Sahoo, JJ Chen, Pi Sun. *Efficient security mechanisms for the distributed wireless sensor networks*. Proceedings of the IEEE Third International Conference on Information Technology and Applications (ICITA'05). 2005; (2): 541-546.
[15] D Liu, P Ning, *Efficient distribution of key chain commitments for broadcast authentication in distributed sensor networks*. Proc. of the 10th Annual Network and Distributed System Security Symposium, California. 2003.
[16] C Karlof, D Wagner, *Secure routing in wireless sensor networks: Attacks and countermeasures*, In Proceedings of First IEEE International Workshop on Sensor Network Protocols and Applications. 2003: 113-127.
[17] NS-2 web site, http://www.isi.edu/nsnam/ns
[18] J Daemen, V Rijmen. The Design of Rijndael: AES - the Advanced Encryption Standard. Springer-Verlag New York. 2002.