Paper: ASAT-14-148-CM

14th International Conference on
AEROSPACE SCIENCES & AVIATION TECHNOLOGY,
ASAT - 14 – May 24 - 26, 2011, Email: asat@mtc.edu.eg
Military Technical College, Kobry Elkobbah, Cairo, Egypt
Tel: +(202) 24025292 –24036138, Fax: +(202) 22621908

Secure, Scalable, and Efficient Key Management Scheme for Wireless Sensor Networks (WSNs)

T. Soliman*, A. Abdelhafez†, and W. Anis‡

Abstract: In Wireless sensor networks (WSNs), key management is the core issue of any security approaches. The main challenge in WSNs key management is how to distribute and manage secret keys within WSN nodes. Several schemes are proposed for this purpose. Another challenge which was not tackled by most of the proposed schemes is the behavior of network in case of nodes capturing, since once any node is captured by an adversary, all the information stored in its memory could be compromised. In this paper we propose a key management scheme for hierarchical WSNs. In addition; we propose a mechanism concerning adding of new sensor nodes, and keys revocation and re-keying processes to overcome nodes capturing crisis. Then we will evaluate the secrecy of our scheme by analyzing the security properties of our proposed scheme with other previous proposed schemes.

1. Introduction

WSNs are being deployed for a wide variety of applications. There are two different network models used for WSNs with each model having its own characteristics. A flat distributed network in which all nodes have the same capabilities and a hierarchical network in which some nodes that called Cluster Heads (*CHs*) have higher capabilities than others that called normal Sensor Nodes (*SNs*) [1]. The hierarchical (heterogeneous) network model has more operational advantages than the flat homogeneous model for *SNs* with their inherent limitations on power and processing capabilities [2, 3]. Many applications of WSNs require secure data communications, especially in a hostile environment as in military field. So the increased attention of key management in WSNs which is the most important and difficult issue in security is obviously seen recently. In general the key management includes the processes of generation, distribution, encryption, and revocation of secret key. For WSNs, symmetric keys are more preferred for data encryption due to the resource constraints of *SNs*, since public key infrastructure is considered to be not suitable to provide security for WSNs because of complexity and power consumption overhead. But some studies on elliptic curves cryptography indicate that algorithm based on this kind of cryptography could be a potential choice [4]. There are three keying models in WSNs, network keying in which the entire network has the same shared key, group keying in which each group in a network has a shared key, third is the pairwise keying in which each pair of nodes in a network has a unique shared key [5].

Concerning key distribution, due to unpredictable network topology and lack of infrastructure support, key pre-distribution mechanism could be a practical solution to these constraints in

* Faculty of Engineering, Ain Shams University, tsoliman69@yahoo.com.
† Egyptian Armed Forces, aabdelhafez@gmail.com.
‡ Prof. Faculty of Engineering, Ain Shams University.

WSN. The basic idea of key pre-distribution scheme is preloading some secret or some keys information into sensor nodes (SNs) before they are deployed. After deployment, the shared keys will be detected using preloaded secret keys or keys information [6].

The rest of this paper is organized as follows; section **2** presents an overview of previous proposed key management schemes for WSNs. Sections **3** and **4** include our proposed key management scheme and re-keying mechanism respectively. Section **5** introduces an evaluation of our proposal according to other previous proposed schemes. Section **6** will conclude the paper, followed by the proposed future work.

2. Overview of Previous Proposed Key Management Schemes for WSNs

Several key management schemes have been proposed recently in WSN. These schemes can be classified according to several aspects. One of the important aspects is the key establishment techniques in which schemes are classified according to the two kinds of approaches that they use, either probabilistic, or deterministic [7]. Probabilistic approach is characterized by the use of key chains, which are randomly selected from a large key pool and preloaded into sensor nodes, and then a shared key between each pair of nodes will be detected after deployment. While in deterministic approach; deterministic processes are used after network deployment to generate the shared key between each pair of nodes using the related preloaded information. Here we will introduce an overview on previous proposed key management schemes according to the homogeneity of WSN (i.e.; flat or hierarchical WSN) taking into account key establishment classifications.

2.1 Key Management Schemes of Flat WSNs

Eschenauer and Gligor (E&G) introduced a probabilistic key predistribution scheme [8], which is considered the basic scheme for key management in WSN. This scheme suffers from shortage of authentication [9]. Chan et al. proposed a "***q***-composite" probabilistic scheme which is very similar to the basic scheme with differences in the size of key pool, and using multiple keys (***q***) to establish communication link between two sensor nodes instead of one key [10]. Du, et al. proposed a probabilistic scheme using deployment knowledge technique [11]. This scheme, exploits deployment knowledge only guaranteeing that any two neighboring nodes can find a common secret key with a certain probability (***p***). Based on location aware technique, Huang et al. proposed a probabilistic scheme [12]. Location aware based key pre-distribution schemes can be used only for some specific applications that have static and well-known configuration like light traffic sensors. Based on polynomial symmetric function *(f(x, y) = f(y, x))* of ***t***- degree threshold, Du et al. proposed a deterministic scheme in which to establish a pairwise key, each node evaluates the polynomial at the identity (***id***) of the other sensor node [13]. Based on Combinatorial Design Theory (CDT), Campte proposed another deterministic scheme which is using block design technique in CDT [14]. This scheme tries to increase key sharing probability among the sensor nodes by designing the key chains.

2.2. Key Management Schemes of Hierarchical WSNs

Localized Encryption and Authentication Protocol (LEAP) [15] is considered the basic hierarchical key management scheme in WSNs. It is a deterministic key management scheme in which four types of secret keys are used as follows; individual key to communicate between each node in a network and *BS*, pairwise key to communicate between each pair of nodes, cluster key to communicate between a group of nodes (cluster members), and network key which is a shared key between all nodes in a network including BS. Du, et al. presented the Asymmetric Pre-distribution (AP) scheme for Heterogeneous Sensor Networks (HSN)

[16]. It is a deterministic scheme. The basic idea of the AP key management scheme is to pre-load a large number of keys in each CH while only pre-loads a small number of keys in each SN. Shen et al. proposed the recent deterministic hierarchical scheme which is based on polynomial key calculations mechanism such that each network node is preloaded with a number of polynomial coefficients ($\boldsymbol{g_j}$) [17]. A Low-Energy Key Management (LEKM) protocol is a probabilistic key management scheme proposed by Gaurave et al. [18]. Kausar et al. proposed the recent probabilistic key management scheme for WSNs, which is based on a random key pre-distribution, where each *SN* and *CH* in a network is preloaded with different numbers of generation keys (g_{ki}) [19]. Then exchanges of information are done to detect the secret keys between each *SN* with each of neighbor and with its *CH*.

3. Our Proposed Key Management Scheme for WSNs

3.1 Network Model

Network model in our scheme is based on three tier hierarchical architecture as shown in Fig. 1. Base Station (*BS*), Cluster Head node (*CH*) and Sensor Node (*SN*). Our proposal considers that a large number of *SNs* and an optimum number of *CHs* will be randomly distributed in a specific area and the *BS* is located in a well-known protected place such that it is trusted and not able to be captured by any adversary. A Pseudo Random Number Generator (PRNG) is preloaded in each *CH* to generate an initial random number (r_{CH}). There are two different hash functions; first is a common hash function ($\boldsymbol{H_C}$) that is used in all network devices (*SN*, *CH*, *BS*), which is used to generate the pairwise key between each pair of *CHs*, *CH* with each of its cluster members (*SNs*), and the *BS* with each of *CH* and *SN* in a network. Second, a private hash function (***H***) for each of *CH* in a network to calculate a private random value (R_{CH}) which is used in generation of pairwise keys as discussed later on .

The used notations are defined in Table 1;

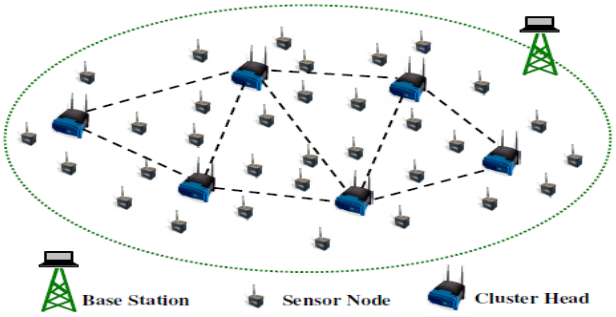

Fig. 1 Network Model in Our Proposed Key Management Scheme

3.2 Phases of Our Proposed Scheme

Our approach has four phases, key pre-distribution phase, initialization and cluster formation phase, intra and inter-cluster pairwise keys establishment phase, and erasure phase.

3.2.1 Key Pre-distribution Phase

It is an off-line phase in which some parameters as node identity (*id*) and others should be preloaded in each node before deployment in WSN. These parameters are loaded into *SNs* and *CHs* by a Key Distribution Server (KSD) which could be the *BS* itself. In our proposal each *SN* and *CH* will be preloaded by its identity (***id***) and other parameters as follows:

Table 1 Used Notations

Notation	Definition
$\boldsymbol{id_{Chi}}$	Identity of Cluster Head ***i***
$\boldsymbol{id_{SNi}}$	Identity of Sensor Node ***i***
$\boldsymbol{K_{X,Y}}$	A shared key between X, and Y
$\boldsymbol{E_{KX,Y}(m \| n)}$	An encrypted message including ***m*** and ***n*** data using the shared key between X and Y for encryption.
X ⇒ Y: *m*	X sends a message *m* to Y.
X ⇐ Y: *m*	X receives a message *m* from Y.
X ⇒ * : *m*	X broadcasts a message *m*.

a) *Master key (K_M)*; which is preloaded in all network devices (*SN*, *CH*, and *BS*) to communicate with each other after deployment in secure manner.
b) BS random value (R_{BS}): each of *CH* in network is preloaded with a different *BS* random value (i.e. $R_{BS,CHi}$ with a CH_i) using to generate the pairwise key between the *BS* and each of *CH* in a network.
c) *SN random value (R_{SN})*: each SN_i in a network is preloaded with a different random value (R_{SNi}) which is used to generate its pairwise key with the *BS* and its *CH*. The *SN* random value (R_{SNi}) is assigned through two steps as shown in Fig. 2. First, input the SN_i identity ($\boldsymbol{id_{SNi}}$) as seed toPRNG of a large enough period to produce a sequence of ***n*** numbers (e.g. PRNG ($\boldsymbol{id_{SNi}}$) = $\boldsymbol{r_{1SNi}, r_{2SNi}, r_{3SNi}, \ldots r_{nSNi}}$). Second, input the selected random number $\boldsymbol{r_{mSNi}}$ (where m = 1: n) into *BS* private hash function ($\boldsymbol{H_{BS}}$) with the master key ($\boldsymbol{K_M}$) to produce the SN_i random value ($\boldsymbol{R_{SNi}}$).

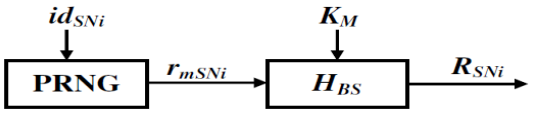

Fig. 2 Process of *SN* Random Value (R_{SN}) Generation

d) *Pairwise key of BS with SN ($K_{BS,SN}$)*: each *SN* in a network should be preloaded with its pairwise key with *BS*. $K_{BS,SN}$ is calculated using the common hash function (H_C) with the exclusive or of *BS* random value (R_{BS}) and *SN* random value (R_{SN}) as shown in (1):

$$K_{BS,SNi} = H_C (R_{BS,SNi} \oplus R_{SNi}) \quad (1)$$

3.2.2 Initialization and Cluster Formation Phase

It is the first phase after nodes deployment in a field, in which the essential information are exchanged, and clusters are configured. Many algorithms are presented to configure the clusters as in [20], [21], [22] which is out of scope of our study. In cluster formation processes of our proposal, each *CH* in a network selects its random value (R_{CH}). The same two steps as in *SN* random value generation is done to the *CH* taking into account that second step may be repeated several times as shown in Fig .3.

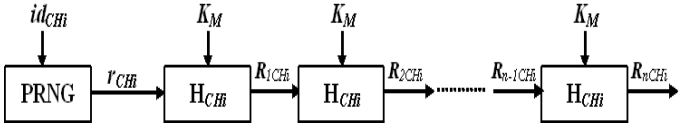

Fig. 3 Process of CH_i Random Value (R_{CHi}) Generation

Then each *CH* broadcasts an encrypted hello message using master key ($\boldsymbol{K_M}$) including its identity ($\boldsymbol{id_{CH}}$) and its random value ($\boldsymbol{R_{CH}}$). Each *SN* receives hello messages from several *CHs*, decrypts them and records in its list the identities of the first three *CHs* ($\boldsymbol{id_{CH}}$) according to their Received Signal Strength Indicator (RSSI) as in [23]. After that *SN* extracts and records the selected random value ($\boldsymbol{R_{CH}}$) of the only first *CH* in its list which is considered its *CH*. Then each *SN* broadcasts an encrypted hello message including its identity ($\boldsymbol{id_{SN}}$), its preloaded random value ($\boldsymbol{R_{SN}}$), and a list of recorded *CHs* identities. Each *CH* receives *SNs* hello messages, designates its order in *SN* list, and arrange *SNs* sequentially after designation its cluster members. Then records *SNs* identities with only its cluster members random values ($\boldsymbol{R_{SNi}}$)

3.2.3 Inter-Cluster Pairwise Keys Establishment Phase

In this phase pairwise keys will be generated between a *CH* and each of its cluster members (*SNs*), each pair of *CHs* in a network, and *BS* with each of *CH* in a network using the common hash function ($\boldsymbol{H_C}$) as in (2), (3), and (4).

$$K_{SNi,CHi} = K_{CHi,SNi} = H_C\,(R_{CHi} \oplus R_{SNi}) \quad (2)$$

$$K_{CHi,CHj} = K_{CHj,CHi} = H_C\,(R_{CHi} \oplus R_{CHj}) \quad (3)$$

$$K_{BS,CHi} = K_{CHi,BS} = H_C\,(R_{BS,CHi} \oplus R_{CHi}) \quad (4)$$

After that each *CH* sends an encrypted message to *BS* using the sharing key ($\boldsymbol{K_{CH,\ BS}}$) including all information about its cluster.

3.2.4 Erasure Phase

All preloaded and generated initial information ($\boldsymbol{K_M}$, $\boldsymbol{R_{BS,CH}}$, $\boldsymbol{R_{SN}}$, and $\boldsymbol{R_{CH}}$) will be erased from each *CH* and *SN* in a network.

Summary of our proposal is presented in Fig. 4.

4. Adding New SNs and Re-keying Mechanism

There are two cases are worthy to be solved in WSNs key management. First, adding of new *SNs* which may be needed for enlarge the network or replacing of depleted *SNs*. Second, key revocation and re-keying mechanism, which is needed in case of node capturing.

In our proposal, the necessary step in both cases is the dissemination of new master key (K_M) to all *CHs* in secured messages using pairewise keys between *BS* and *CHs* as in (5).

$$BS \Rightarrow CH_i : E_{KBS,CHi}(K_M) \quad (5)$$

4.1 Adding of New SNs

The added *SNs* should be preloaded with the new K_M . After deployment, each added *SN* broadcasts a secured message including its identity ($\boldsymbol{id_{SNi}}$). Consequently, *CHs* in range of its transmission area will generate a new random value ($\boldsymbol{R_{CHi}}$) as shown in Fig. 3, and broadcast a secured message including their $\boldsymbol{id_{CHi}}$, and the generated $\boldsymbol{R_{CHi}}$. Then the processes of pairwise key generation between SN_i and CH_i will be consumed according to our original proposal in Fig. 4.

4.2 Revocation and Re-keying Mechanism

Our proposal for re-keying mechanism includes four stages; detection of compromised node, dissemination of compromised node identity, revocation of compromised keys, and re-keying.

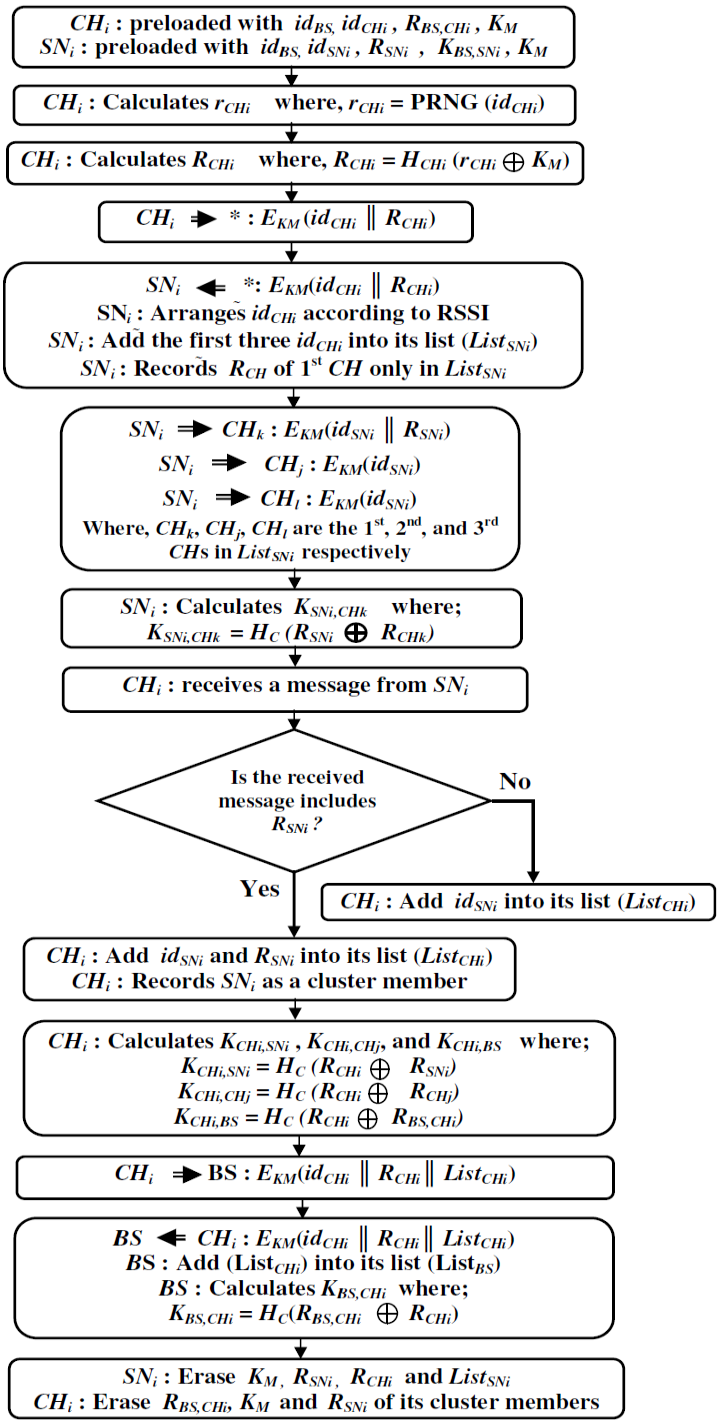

Fig. 4. Flowchart of our proposal

4.2.1 Detection of Compromised Node Stage

There are several proposed algorithms to discover and designate the compromised node, which is out of scope of our study.

4.2.2 Dissemination of Compromised Node Identity Stage

In case of sensor node compromising (SN_c), *BS* will send secured caution messages including the compromised node identity (***id***$_{SNc}$) to all *CHs* which are relevant to SN_c to inform them about SN_c. While in case of *CH* compromising, *BS* will send a caution messages to all *SNs* connected to compromised cluster head (CH_c) and to all other *CHs* in a network.

4.2.3 Revocation of Compromised Keys Stage

In case of *SN* compromising, *BS* and *CH* will revoke their pairewise keys with SN_c, also id_{SNc} will revoked from all relevant *CHs* lists. In case of *CH* compromising, its pairewise keys with all other network *CHs* and its cluster members (*SNs*) will be revoked.

4.2.4 Re-keying Stage

In case of *SN* compromising, there is no need to re-keying because the only pairwise keys with *BS* and with its *CH* will be revoked, so there is no more actions. In case of *CH* compromising (CH_c), cluster members (*SNs*) of CH_c should re-key their pairewise keys with other non-compromised *CHs* in a network, so the following steps will be preceded:

a) The second *CH* in *SN* list ($List_{SN}$) will generate a new R_{CH}, and send it to *BS* after receiving the caution message.
b) *BS* will generate a new $\boldsymbol{R_{SN}}$ for each of *SN* in a compromised cluster, and then send a secured message to each of *SN* in compromised cluster including its new generated random value ($\boldsymbol{R_{SN}}$), and new *CH* information ($\boldsymbol{id_{CH}}$ and $\boldsymbol{R_{CH}}$). In addition, *BS* will send the information of new *SNs* members ($\boldsymbol{id_{SN}}$ and $\boldsymbol{R_{SN}}$) to *CH*.
c) Then *CH* and each of new *SN* in its cluster will calculate their pairewise key using common hash function ($\boldsymbol{H_C}$) as in original algorithm.

Summary of key revocation and re-keying mechanism is presented in Fig. 5.

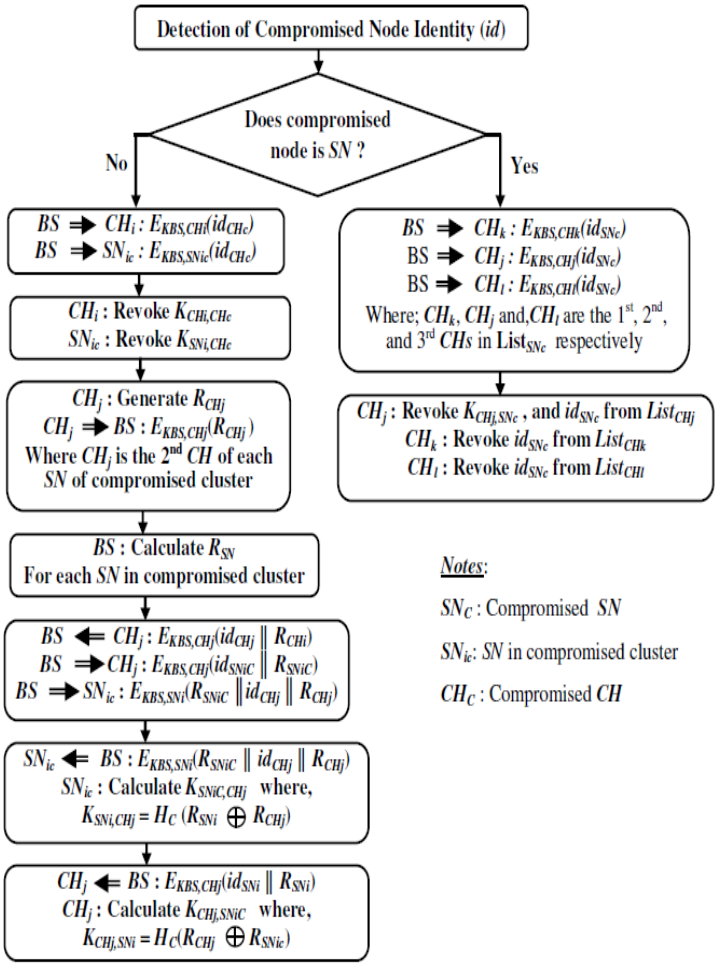

Fig. 5 Key Revocation and Re-keying Mechanism

5. Evaluation of Our Proposed Scheme

To explore the advantages of our proposed key management scheme, it should be evaluated or in other words should be compared with other previous proposed schemes. In this section, we will analyze the security properties of our proposed scheme comparatively with LEAP scheme [15] which is considered the basic hierarchical key management scheme in WSNs, and with both of Kausar et al. scheme [19] and Shen et al. scheme [17] which are the most recent hierarchy schemes in WSNs.

5.1 Security Analysis

A key establishment technique is not judged solely based upon its ability to provide secrecy of transferred messages after establishment of secret keys, but must also meet certain other criteria for efficiency in light vulnerability to adversaries including the protection of exchanged information during secret key establishment phases, key revocation, and network resiliency [24].

There are two aspects to be discussed, first, the achieving of basic security requirements (confidentiality – authentication – authorization) during secret key establishment phases. Second, network resiliency which is one of the important security analysis aspects.

5.1.1 Achieving of Security Requirements

In our proposed scheme, all exchanged information in all phases are encrypted by a preloaded temporary key (K_M) which is erased after establishment of secret keys, so all basic security requirements (confidentiality – authentication – authorization) are achieved.

Concerning LEAP scheme [15], in dependency on a needed short period to establish the secret keys, LEAP exchanges the initial information in plain, while the acknowledgement is authenticated using a preloaded initial key. So the basic security requirements are not achieved since confidentiality and authorization services are not achieved, while the authentication is achieved in acknowledgement messages only.

For Kausar et al. scheme [19], all exchanged information in cluster formation phase are in plain, while in other key establishment phases are authenticated. Then we can globally say that, there are no achieving of confidentially and authorization security services.

For Shen et al. scheme [17], there is no proposed vision for cluster formation and all exchanged information for secret key establishment are in plain.

Table 2 summarizes the above comparison of security requirements achievement between schemes.

Table 2 Security Requirements Achievements Comparison

Item	Our proposal	LEAP	Kausar	Shen
Mode of Transmission	Encrypted	Plain	Plain	Plain
Achieved Security Requirements	Confidentiality, Authentication, Authorization	Authentication for Ack only	Authentication only	None

5.1.2 Network Resiliency

In WSN, if a node is captured, the key establishment technique should ensure that secret information about other nodes is not revealed. A network resiliency is calculated as the compromising probability of communication links of non-compromised nodes versus the number of compromised nodes.

In our proposed scheme, *SN* communicates only with *BS* and its *CH* through different secret (pairwise) keys. So *SN* capturing will compromise only the communication links the compromised *SN* with *BS* and its *CH* and cannot effect on other network communication links. While for *CH* capturing, all relevant communication links of compromised *CH* with *BS*, its cluster members (*SNs*), and other network *CHs* will be compromised. Our proposed scheme includes a mechanism for compromised key revocation and re-keying. This mechanism includes the necessary processes to retrieve the network security efficiency in cases of *SN* or *CH* capturing.

For LEAP scheme, one of the major drawbacks is that it assumes that the secret keys have never been disclosed and then it is not resilient for compromised nodes [25]. So in case of *SNs* or *CHs* capturing, the group key will be revealed, thereby all the entire network communication links will be compromised (i.e., the probability of compromised communication links = 1). In addition, there is no proposed mechanism for keys revocation and re-keying.

For Kausar et al. scheme, the probability of total number of compromised keys, or network compromised communication links as in (6);

$$P = 1 - (1 - r/M)^{n`} \quad (6)$$

where, $n`$ is the number of compromised *SNs*, r is the number of preloaded generation keys, M is the number of chains of generation keys. The similar case for CH capturing as in (7);

$$P = 1 - (1 - s/M)^{m`} \quad (7)$$

where, s is the number of preloaded generation keys, $m`$ is the number of compromised *CHs* in network. In addition, the proposed mechanism for key revocation and re-keying is only including the actions of *SN* capturing, and not include any actions for *CH* capturing [26].

For Shen scheme, it is similar to our proposed scheme but the difference or in other words the major drawback of Shen scheme is that there is no proposed mechanism for key revocation and re-keying [26].

Figures 6 and 7 show the compromising probability of non-compromised nodes communication links (network resiliency) versus the number of compromised *SNs* (n`) and *CHs* (m`) respectively for compared schemes. We assumed that the network is uniformly distributed (i.e., each cluster has *n/m* members, where *n* is the total number of *SNs* in network and *m* is the total number of *CHs* in network). For Kausar scheme, we assumed the preloaded generation keys (*r*) and (*s*) are equal ***10*** and ***150*** keys respectively which achieve the highest probability of key sharing among nodes (the probabilities = 0.99) with key chains (*M*) equal *400* chains as in [19] .

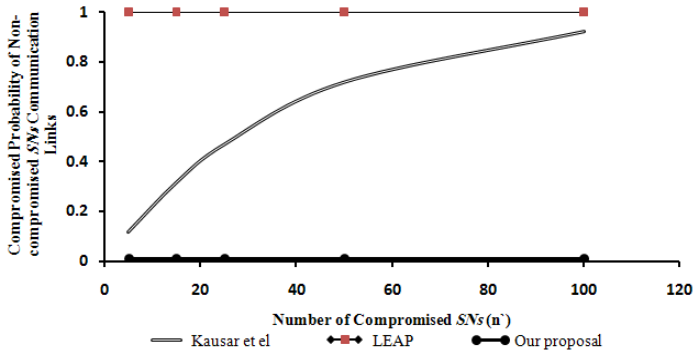

Fig. 6 *SNs* Resiliency versus Number of Compromised *SNs* (n`)

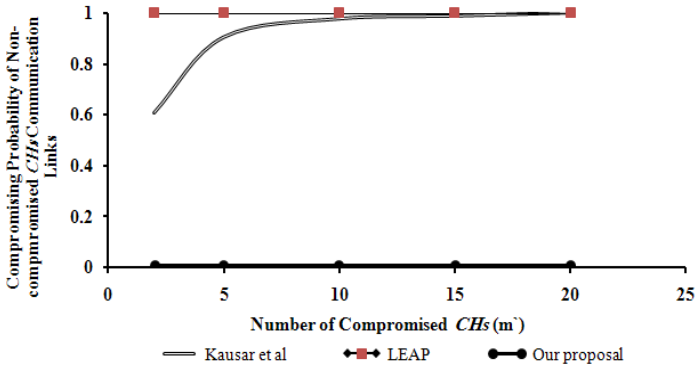

Fig. 7 *CHs* Resiliency versus Number of Compromised *CHs* (m`)

From the above discussions and Figs. 6 and 7, it is shown that our proposal has the strongest network resiliency whatever the number of captured (compromised) nodes, while LEAP is the weakest one. For Kausar et al. scheme, it depends on the number of both of captured nodes, key chains (*M*), and preloaded generation keys. Note that for all studied schemes, the network resiliency does not depend on the number of network size.

6. Conclusion and Future Work

In this paper, we proposed a secure, scalable, simple, and efficient key management protocol for WSNs based on a three-tier network hierarchy (*BS*, *CH*, and *SN*) with simple computations and low communications. In addition, we proposed a key revocation and re-keying mechanism which is considered a complementary feature for our proposed scheme. Then we analyze the security property of our proposal comparatively with other recent previous schemes to explore the features of our proposal.

In our future work we will implement our proposed scheme in a test bed to measure its performance in terms of key storage overhead and communication overhead, and some other metrics as power consumption, simplicity, and scalability. Also we will study our proposed scheme immunity against various security attacks.

References

[1] K. M. Martin, M. Paterson, "An Application-Oriented Framework for Wireless Sensor Network Key Establishment", *Computer Science 192*, 2008, *pp. 31–41*.

[2] B. Liu, Z. Liu, and D. Towsley, "On the capacity of hybrid wireless networks," *In Proceedings of IEEE Infocom 2003, San Francisco, CA*, Apr. 2003.

[3] S. Zhao, K. Tepe, I. Seskar, and D. Raychaudhuri, "Routing Protocols for Self-Organizing Hierarchical ad-hoc Wireless Networks," *in Proceedings of IEEE Sarnoff 2003 Symposium*, 2003.

[4] F. Amin, A. H. Jahangir, and H. Rasifard, "Analysis of Public-Key Cryptography for Wireless Sensor Networks Security", *Proceedings of World Academy of Science, Engineering and Technology Volume 31 July 2008 ISSN 1307-6884*.

[5] Johnson. C. Lee and Victor C. M. Leung, "Key Management Issues in Wireless Sensor Networks: Current Proposals and Future Developments", IEEE journal of Wireless Communications, Vol. 14, No. 7, 2007, pp. 76 - 84.

[6] Yang. Xiao, "Security in Sensor Networks", *Auerbach Publications, Taylor& Francis Group, LLC*, Edition, 2007.

[7] S. A. Camtepe, and B¨ U. Yenr, "Key Distribution Mechanisms for Wireless Sensor Networks: *a Survey", Technical Report TR-05-07, Department of Computer Science, Rensselaer Polytechnic Institute*, March 23, 2005.

[8] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," *in Proceedings of the 9th ACM Conference on Computer and Communications Security,* Nov, 2002.

[9] R. M. S. Silva, N. S. A. Pereira, M. S. Nunes, "Probabilistic Key Management Practical Concerns in Wireless Sensor Networks", *Journal of Networks, Vol. 3, No. 2,* Feb. 2008, pp. 29 -37.

[10] H. Chan, A. Perrig, and D. Song, "Random Key Predistribution Schemes for Sensor Networks", *Proceedings of the 2003 IEEE Symposium on Security and Privacy,* May 11-14, 2003, pp. 197 – 213.

[11] W. Du, J. Deng, Y. S. Han, S. Chen, and P. K. Varshney, "A Key Management Scheme for Wireless Sensor Networks Using Deployment Knowledge", *IEEE Transaction on Dependable and Secure Computing,* VOL. 3, NO. 1, January- March 2006, pp. 62 -77.

[12] D. Huang, M. Mehta, D. Medhi, and L. Harn, "Location aware Key Management Scheme for Wireless Sensor Networks". *Proceedings of ACM Workshop on Security of Ad Hoc and Sensor Networks (SASN'04),* October 2004, pp. 29-42.

[13] Cheng Y, Agrawal DP, "An improved key distribution mechanism for large-scale hierarchical wireless sensor networks", *Elsevier Journal of Ad Hoc Networks,* Vol. 5, 2007, pp. 35 – 48.

[14] S. A. Çamtepe, and B. Yener, "Combinatorial Design of Key Distribution Mechanisms for Wireless Sensor Networks", *IEEE/ACM Transmission on Networking, Vol. 15, No. 2,* April 2007, pp. 346 -358.

[15] S. Zhu, S. Setia, and S. Jajodia, " LEAP: Efficient Security Mechanisms for LargeScale Distributed Sensor Networks",. *Proceedings of 10th ACM Conference on Computer and Communications Security (CCS '03),* October 2003, pp. 62 - 72.

[16] X. Du, Y. Xiao, M. Guizani, and H. H. Chen, "An Effective Key Management Scheme for Heterogeneous Sensor Networks,", *Elsevier journal of Ad Hoc Networks*, Vol. 5, No 1, jan.2007, pp 24-34.

[17] A.N. Shen, S. Guo1, H. Y. Chien, and M. Guo," A scalable key pre-distribution mechanism for large-scale wireless sensor networks", Concurrency and Computation: Practice and Experience, Vol. 21, No. 10, 2009, pp. 1373–1387.

[18] G. Jolly, M. C. Kuscu, P. Kokate and M. Yuonis, "A low-energy key management protocol for wireless sensor networks," *in Proceeding of the Eighth IEEE International Symposium on Computers and Communication (ISCC'03),* USA, 2003, pp. 335 - 340.

[19] F. Kausar , S. Hussain, L. T. Yang ·and A. Masood, "Scalable and efficient key management for heterogeneous sensor networks", *Journal of Supercomputing* Volume 45 , No. 1, July 2008, pp. 44 – 65.

[20] S. Banerjee and S. Khuller, "A Clustering Scheme for Hierarchical Control in Multi-hop Wireless Networks," *in Proceedings of IEEE INFOCOM,* 2001, Anchorage, AK, USA, 2001, pp. 1028 - 1037.

[21] Y. P. Chen , A. L. Liestman ,and J. Liu," Ad Hoc and Sensor Networks", *Nova Science Publishers,* 2004, available at [http: www.cs.mun.ca/~yzchen/papers/chapter04.pdf].

[22] A. A. Abbasi, and M. Younis, "A survey on clustering algorithms for wireless sensor networks", *Elsevier journal of Computer Communications*, Vol. 30, No. 14, June, 2007, pp. 2826–2841.

[23] X. Du, Y. Xiao, "Energy efficient chessboard clustering and routing in heterogeneous sensor network", *International Journal of Wireless and Mobile Computing*, Vol. 1, No. 2, 2006, pp.121 - 130

[24] Y. Xiao, V. K. Rayi, B. Sun, X. Du, F. Hu, and M. Galloway, "A Survey of Key Management Schemes in Wireless Sensor Networks", *Computer Communication journal, Special Issue on Security on Wireless Ad Hoc and Sensor Networks*, Vol. 30, No. 11-12, 2007, pp. 2314 - 2341.

[25] Y. Wang, G. Attebury, and, B. Ramamurthy "A Survey of Security Issues in Wireless Sensor Networks", *IEEE journal of Communications Surveys*, Vol. 8, No. 2, 2006, pp. 2 - 23.

[26] T. Soliman, and A. Abdelhafez " Efficient Key Management Schemes for Wireless Sensor Networks: A Comparative Study" *Al-Azhar University Engineering Journal, JAUES, Vol. 5, No. 3,* Dec. 2010, pp. *343–356.*