A New Hierarchical Key Management Scheme for Secure Clustering in Wireless Sensor Networks

Mohamed-Lamine Messai, Hamida Seba, Makhlouf Aliouat

▶ **To cite this version:**

Mohamed-Lamine Messai, Hamida Seba, Makhlouf Aliouat. A New Hierarchical Key Management Scheme for Secure Clustering in Wireless Sensor Networks. 13th International Conference on Wired/Wireless Internet Communication (WWIC), May 2015, Malaga, Spain. pp.411-424. hal-01728801

HAL Id: hal-01728801

https://inria.hal.science/hal-01728801

Submitted on 12 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution 4.0 International License

A New Hierarchical Key Management Scheme for Secure Clustering in Wireless Sensor Networks

Mohamed-Lamine Messai[1,2], Hamida Seba[2], and Makhlouf Aliouat[3]

[1]Department of computer science, faculty of exact sciences, University of Bejaia, 06000 Bejaia, Algeria
[2]University of Lyon 1, CNRS, LIRIS, UMR5205, F-69622, France
[3]Department of computer science, faculty of sciences, University of Setif 1, 19000 Setif, Algeria
messai.amine@gmail.com,hamida.seba@univ-lyon1.fr,aliouat_m@yahoo.fr

Abstract. In Wireless Sensor Networks (WSNs), clustering is the suitable topology to save the energy of sensor nodes. In this paper, we provide a secured cluster formation by proposing a new symmetric key management scheme for hierarchical WSNs. The new scheme is called EAHKM (Energy Aware Hierarchical Key Management in WSNs). EAHKM needs the pre-distribution of only three keys in each sensor node before deployment, and it ensures a secure cluster formation after deployment. EAHKM assures the establishment of a pairwise key between each sensor node and its cluster head, thus the establishment of a broadcast key in each cluster in the network. Simulation results show that EAHKM provides an energy-efficient, flexible and scalable solution to the key management problem in hierarchical WSNs, and it presents a good resilience to node compromising attacks than other hierarchical key management schemes.

Key words: Secure clustering, hierarchical wireless sensor networks, key management, Energy saving

1 Introduction and motivation

Nowadays, Wireless Sensor Networks (WSNs) become the key technology for ubiquitous computing. They can be integrated in any application that collects data in environments [1]. A WSN consists of many tiny sensing devices called sensor nodes. A great number of sensor nodes are deployed in the field of interest either randomly or in predefined positions. Each sensor node has a sensing and a wireless communication capabilities, which enable it to sense data from the environment and send them to other sensor nodes in the network or to a base station (BS) [1].

A sensor node is equipped with limited resources and is manufactured to be a low cost device. As a result of its resource constraints, applying existing security solutions that are applied in wired networks or mobile ad hoc networks is

infeasible. Security in this context becomes a challenging task and requires the design of appropriate key management schemes [2, 3]. In WSNs, most proposed key management schemes are based on symmetric cryptography for its resource savings [4]. However, the challenge in using the symmetric cryptography is how to distribute, maintain and refresh pairwise keys between communicating sensor nodes in a network. The pairwise key establishment is needed to provide the authentication, the confidentiality and the integrity of exchanged messages between each neighboring sensor nodes. Key pre-distribution is the technique used to ensure key sharing after deployment of sensor nodes. It consists of pre-loading symmetric keys into memories of sensor nodes before their deployment. It is a practical method to deal with the key distribution problem in WSNs [5].

For energy saving purpose (in particular in long lived WSNs), a WSN is usually organized into a cluster-based topology that forms an hierarchical network. Hierarchical WSNs are more often used in real applications because they can improve the network scalability and reduce the energy consumption [6, 7]. Therefore, the hierarchical organization is a promising technique in WSNs to ensure the key management process.

An example of hierarchical topology of a WSN is illustrated in Figure 1. In a cluster-based topology, sensor nodes are randomly scattered in the sensing area and transmit their messages to a BS through cluster heads. Each sensor node in the network is either selected as cluster head (CH) or it is a cluster member by choosing a neighboring sensor node as a CH.

The existing hierarchical key management schemes [8–14] suppose that the WSN is previously organized into clusters by applying one of the existing clustering algorithms. So, the proposed key management solutions relay on hierarchical WSNs and do not take into account the security of cluster formation where an attacker can easily disturb this phase. As a result, existing schemes of key management do not ensure a secure cluster formation phase.

This paper propose a new scheme called EAHKM (Energy Aware Hierarchical Key Management Scheme for Wireless Sensor Networks) with a secure phase of cluster formation. EAHKM has two phases; (1) key pre-distribution phase and (2) cluster formation and key establishment phase. In the first phase, each sensor node S_i is pre-distributed with three keys : $K_{BS,i}$, $K_{i,BS}$ and K_N. $K_{BS,i}$ and $K_{i,BS}$ are two pre-distributed keys to encrypt messages (e. g. node readings) from sensor node S_i and the BS. K_N is the network key shared by all sensor nodes. This key will be deleted after cluster formation and key establishment phase. EAHKM uses a symmetric crypto-system and supports in the second phase an energy efficient formation of clusters with the establishment of two keys : K_{c_i} is a cluster key shared between sensor nodes of a cluster i, and K_{i,CH_j} is the pairwise key between a sensor node S_i and its cluster head S_j.

The rest of the paper is organized as follows. Section 2 discusses the related works on hierarchical key management schemes in WSNs. Section 3 details our proposed scheme EAHKM. Section 4 presents the security analysis of EAHKM. In Section 5, we compare and evaluate by simulation EAHKM with some existing hierarchical key management schemes. Conclusion is presented in Section 6.

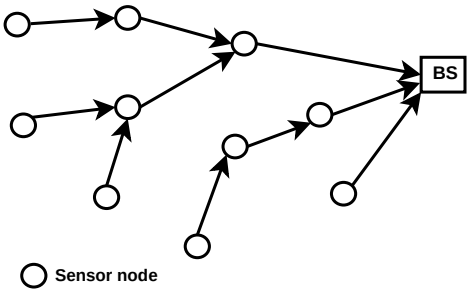

Fig. 1. An example of an hierarchical WSN.

2 Related work

Recently, several key management schemes using symmetric crypto-systems and based on key pre-distribution have been proposed to establish secure communications between sensor nodes in WSNs. However, few works consider key management in an hierarchical topology of WSNs.

In this section, we discuss some proposed hierarchical key management schemes in WSNs. We classify the existing schemes into centralized and distributed as presented in Figure 2.

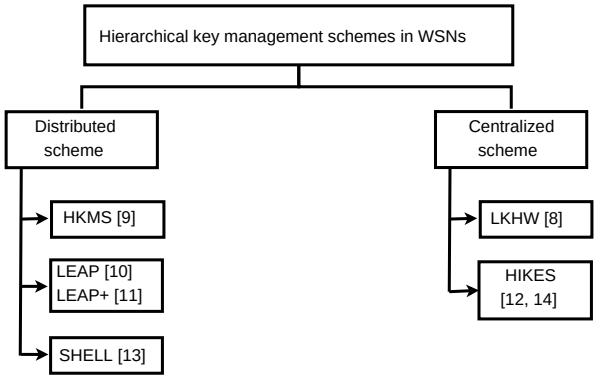

Fig. 2. Classification of existing hierarchical key management schemes in WSNs.

Among the first proposed centralized hierarchical key management schemes in the literature, we find LKHW [8]. This scheme is based on a logical key hierarchy. In LKHW [8], the BS distributes keys over a previously constructed tree. The BS is the root of the tree and it plays the role of a key distribution center to nodes. LKHW [8] is flexible, it allows new nodes to join the network.

The main drawback of this scheme is that the BS is the single point of failure. If the BS of the network fails, key management cannot take place.

In [9], authors presented a distributed hierarchical key management named HKMS for clustered WSNs. HKMS [9] is designed for homogenous WSNs, it does not employ special sensor nodes (e.g. sensor nodes with high energy or high computational capability) to distribute or to establish keys. In HKMS [9], sensor nodes are pre-distributed with an initial key before their deployment. The selection of *CHs* is done according to an algorithm proposed in [15] without any security mechanism. HKMS [9] limits the size of clusters by using a TTL (Time To Leave) for broadcasted messages. The *CHs* broadcasts their messages encrypted by the pre-distributed initial key. The authors fix the value of the TTL to three without any argument or justification. The security of HKMS [9] is based on an initial key. If an adversary compromises a sensor node immediately after its deployment and discovers the initial key, he can compute any established pairwise key in the network.

In [10, 11], a distributed hierarchical key management scheme called Localized Encryption and Authentication Protocol (LEAP) is proposed. LEAP [10, 11] establishes five keys for each node: a global key, a pairwise key, a cluster key, an individual key and a group key. Each sensor node is initially pre-distribute with a global key K_I, an individual key shared with the BS, and pseudo-random function f. Using K_I and f, a sensor node can generate its master key and those of other sensor nodes. LEAP [10, 11] is a scalable scheme but it has a low resilience to node compromising attacks. The entire security is based on a global initial key K_I. Once a sensor node is compromised before a threshold time, an adversary can obtain all pairwise keys. K_I is erased immediately after the neighbor discovery phase and the pairwise key establishment phase, it is not possible to add new sensor nodes to the network. Therefore, LEAP [10, 11] has a poor flexibility.

A new scheme called HIKES (Hierarchical Key Establishment Scheme) is proposed in [12, 14]. In HIKES [12, 14], the BS plays the role of the Trust Authentication (TA) and assigns a part of its role to CHs. HIKES [12, 14] uses a partial key escrow table that enables any sensor node, selected as a CH, to generate all the cryptographic keys needed to authenticate other members within its cluster. However, storing partial key escrow table requires an additional memory space. When an adversary obtains a partial key escrow table through a node compromise of a CH, he can deduce the pairwise keys between this CH and its members.

In order to underwrite the scalability metric, the authors in [13] proposed a scheme called SHELL (Scalable, Hierarchical, Efficient, Location-aware, and Lightweight). SHELL [13] employs the Exclusion Basis System (*EBS*) method [16] and performs location-based key assignment in the clusters to decrease the number of keys revealed by a node compromising attack. According to *EBS* [16], there are $k+m$ keys and each sensor node knows a distinct set of k keys. When a sensor node is compromised, the m keys (not known to the compromised sensor node) are used to replace the k compromised keys. It is notable that a large k

increases the storage requirements at the sensor node, while a large m increases communication overhead for the revocation of compromised keys.

Above schemes are applied in hierarchical WSNs. They do not consider the security aspect in the phase of cluster formation of deployed nodes. This phase is the first step to perform by the deployed nodes. For this reason, we propose EAHKM to secure the cluster formation and ensure the key management in clustered WSNs.

3 Energy Aware Hierarchical Key Management for Wireless Sensor Networks

3.1 Network, energy and threat models

The following properties are assumed in regard to the WSN model:

- Each sensor node has a unique identifier S_i.
- N homogeneous sensor nodes are randomly dispersed in a $m \times m$ field.
- Sensor nodes are homogeneous : they have the same capabilities of processing, memory and battery.
- The sensor nodes can use power control to change the amount of transmit power.
- The sensor nodes and the BS are static after deployment.
- The sensor nodes are unaware of their locations after deployment.
- The BS has no constraint on energy.
- Communications are symmetric: meaning if a node S_i can listen to a node S_j, S_j can listen to S_i.

In our work, we use the radio model proposed in [17]. According to this model, the energy cost to transmit or receive $k\,bits$ between two sensor nodes that are separated by a distance d is given by the following equations:

$$E_{Tx}(k,\, d) = k \times E_{elec} + k \times \varepsilon_{fs} \times d^2, d \leq d_0 \tag{1}$$

$$E_{Tx}(k,\, d) = k \times E_{elec} + k \times \varepsilon_{amp} \times d^4, d > d_0 \tag{2}$$

$$E_{Rx}(k) = k \times E_{elec} \tag{3}$$

Equations 1, 2 and 3 present the energy consumption of the radio module. Depending on the transmission distance d_0 (the threshold distance), both the free space channel ϵ_{fs} and the multi-path fading channel ϵ_{amp} models are used. When receiving, the radio expends E_{Rx} according to equation 3.

We assume that an adversary can listen to all broadcasted messages in the network. However, in the bootstrapping step (immediately after deployment of sensor nodes) adversaries cannot be present in all regions. When a sensor node is compromised, we assume that all its security material (keys, cryptographic algorithm, etc.) will be disclosed to the adversary. The BS cannot be compromised.

3.2 Proposed scheme

Our approach has two phases: a key pre-distribution phase, and a cluster formation and key establishment phase. The notations used in our work are listed in Table 1.

Table 1. Notations.

Notation	Description
S_i	i^{th} sensor node in the network
$\{M\}_k$	Encryption of the message M with the key k
$BS \longrightarrow * : M$	The BS broadcasts the message M
$MAC_k(M)$	Message Authentication Code of M using the key k
K_{c_i}	Cluster key when the sensor node S_i is the CH
K'_N	Refreshed key of K_N
N_i	A nonce generated by the sensor node S_i
CH_i	Cluster head S_i
$H_k(M)$	A one-way hash function applied to M using the key k

Key pre-distribution phase: Sensor nodes are pre-distributed with the following three keys before their deployment into the sensing area: K_N, $K_{S_i,BS}$ and K_{BS,S_i}. K_N is the key shared by all sensor nodes in the network. This key is deleted from memories of nodes after the second phase. $K_{S_i,BS}$ and K_{BS,S_i} are two keys shared between S_i and the BS for encryption of messages. S_i uses $K_{S_i,BS}$ to encrypt its messages sent to the BS. Similarly, the BS uses K_{BS,S_i} to encrypt messages sent to S_i. By employing these two pairwise keys, a cryptanalysis attack becomes very hard to do. The BS stores K_N and $2 \times n$ pairwise keys in its memory where n is the number of nodes.

Cluster formation and key establishment phase: After deployment, the BS initiates the construction of clusters by broadcasting a Hello message as follows:

$$BS \rightarrow \{Hello, BS, Level = 0, Energy = \infty, MAC_{K_N}(BS, 0, \infty)\}_{K_N}.$$

Then, each sensor node executes the steps of cluster formation and key establishment given by Algorithm 1. When a sensor node receives the Hello message of the BS, it chooses the BS as a CH. If the node receives Hello messages from its neighbors, it chooses as CH the node with the lowest level that has the highest energy value. The selection of *CHs* is modified at regular intervals.

After the cluster formation and key establishment phase, each CH generates a cluster key K_{c_i} and sends it to each member node of its cluster encrypted by the established pairwise keys. So, each sensor node in the network stores a

cluster key K_{c_i} and a pairwise key employed to secure the communication with its CH in addition of the three pre-distributed keys.

After key establishment, the CH generates a Time Division Multiple Access (TDMA) schedule for its members. The schedule message is encrypted with the shared cluster key K_{c_i}. Each sensor node transmits its messages to its CH in its assigned time slots encrypted with K_{i,CH_j} and it keeps its state in sleep mode during the remained slots. The CHs and the BS exchange messages by using a Code Division Multiple Access (CDMA).

Algorithm 1 Cluster formation and key establishment

Receive $\{Hello, Sender\_ID, Sender\_Level, E_{Sender\_ID}, CH_{Sender\_ID}, MAC_{K_N}(Sender\_ID, Sender\_Level, E_{Sender\_ID})\}_{K_N}$
If ($Sender\_ID$:= BS)
$CH_i := BS$; /* cluster head is the identifier of the BS. */
$Level_i := 1$;
/*A shared key already exists.*/
Else
Begin
Repeat
Add E_{Sender_ID} to E_List_i
Add $Sender\_Level$ to $Level\_List_i$
Receive $\{Hello, Sender\_ID, Sender\_Level, E_{Sender\_ID}, CH_{Sender\_ID}, MAC_{K_N}(Sender\_ID, Sender\_Level, E_i)\}_{K_N}$
Until (receiving all neighbors messages)
$CH_i :=$ The identifier of the node with the greatest ratio $E_{Sender}/Level_{Sender}$;
$Level_i :=$ Level of its cluster head + 1;
/* Compute a pairwise shared key with the cluster head */
$K_{S_i,CH_i} := H_{K_N}(S_i||CH_i||Level_i)$;
end;
EndIf
$S_i \rightarrow \{Hello, S_i, Level_i, E_i, CH_i, MAC_{K_N}(S_i, Level_i, E_i)\}_{K_N}$

Addition of Sensor nodes. In most cases, new sensor nodes are added by a post-deployment to a deployed WSN to assure network connectivity, to replace dead nodes or to cover more regions in an area of interest. EAHKM is flexible by allowing addition of new sensor nodes. New sensor nodes will be able to establish pairwise keys with previously deployed sensor nodes. Adding a new node S_n is achieved as follows:

1. Before deploying S_n, the BS sends a new generated K'_N to all nodes in the network over the CHs.
2. S_n is pre-distributed with $K_{S_n,BS}$, K_{BS,S_n} and the current K'_N.
3. After being deployed, S_n generates a nonce N_n and broadcasts a Join message as follows: $S_n \rightarrow \{JOIN, S_n, N_n, MAC_{K_N}(S_n, N_n)\}_{K'_N}$. The Join message is encrypted with K'_N, the shared key by all sensor nodes.

4. When receiving the join message, every cluster head CH_i in the transmission range of the new sensor node generates a nonce N_i and responds with the following message : $\{JOIN-Ack, S_i, N_n, N_i, Level_i, E_i, MAC_{K_N}(S_i, N_i)\}_{K'_N}$.
5. S_n declares its CH, the source of the received message with high ratio $E_{CH_i}/Level_{CH_i}$ and diffuses the following message : $\{CH\_ID, S_n\}_{K'_N}$.
6. The CH node adds S_n in its cluster member list.
7. The CH and S_n compute their shared keys: $K_{S_n,CH_i} := H_{K'_N}(S_i||CH_i||Level_i)$;
8. The CH sends to S_n the cluster key K_{c_i} encrypted with K_{S_n,CH_i}

Key refresh. To prolong the lifetime of the network, it is necessary to change the CH role among the sensor nodes. In EAHKM, the rekeying operation refreshes the clusters and the keys. Key refresh is on demand and it is initiated by the BS. Before key refresh, the BS sends to all sensor nodes through *CHs* a new generated network key K'_N. Then, it broadcasts the same diffused message after the initial deployment of sensor nodes encrypted by K'_N :
$BS \rightarrow \{Hello, BS, Level = 0, Energy = \infty, MAC_{K_N}(BS, 0, \infty)\}_{K'_N}$.
All sensor nodes re-execute Algorithm 1 (cluster formation and key establishment) given above. After the end of the algorithm execution, we will get new clusters with new established keys.

4 Security analysis

In this section, we analyze the security of our solution. An outsider adversary, who does not know the key K_N, cannot discover the meaning of broadcasted messages by the BS and sensor nodes after their deployment. Nevertheless, an adversary can compromise one or more sensor nodes, so he becomes an insider adversary. The keys of compromised sensor nodes can be used to forge wrong useless messages to waste the energy of nodes. We note here that the compromised nodes cannot induce any damage on other communications in the network. Inside a cluster, the established pairwise keys and the cluster key are used to encrypt and authenticate messages. Therefore, adversaries who do not know these keys cannot obtain the clear messages. In the following, we present how EAHKM prevents specific attacks in WSNs; Hello flooding, Sybil and node replication attacks.

Hello flooding attack: in EAHKM, sensor nodes discover their neighbors by sending Hello messages encrypted with the key K_N. An attacker cannot launch a Hello Flooding attack without knowing the key K_N.

Sybil attack: in the cluster formation and key establishment algorithm, a MAC of the sensor node identifier, its level, and its CH identifier is calculated to authenticate the sender and the receiver. Therefore, a sensor node cannot play a role of an other sensor node.

Node replication attack: this attack is detectable by the CH if the replication is in the same cluster. However, node replication may affect EAHKM when an adversary duplicates compromised sensor nodes in different clusters.

5 Comparison and simulation results

In this section, we first define the metrics for performance evaluation. Then, we present a comparison of EAHKM and the schemes discussed in Section 2. After that, we use simulation to compare EAHKM, LEAP+ [11] and HKMS [9].

5.1 Evaluation Metrics

The aim of a key management scheme is to provide the basic security objectives in terms of integrity, confidentiality and authentication. In addition to these objectives, a key management scheme is evaluated in accordance to the following metrics:

- Efficiency: It refers the energy consumption, storage requirement and calculation needed by the key management scheme.
- Flexibility: It is the adaptability of the key management scheme to add and revoke sensor nodes.
- Resiliency or resistance to node compromising attacks: It measures the impact of capturing nodes on the links of non-compromised nodes in the network.
- Key connectivity (KC): It is the probability that two adjacent sensor nodes share a key after their deployment.
- Scalability: It measures the ability of the key management scheme to support a large number of deployed sensor nodes.

5.2 Comparison

In EAHKM, each sensor node is pre-distributed initially with three keys before deployment. After deployment, if the sensor node is a CH, it computes a number of keys that is equal to the number of its cluster members. The analysis of the communication complexity for the construction of clusters is measured by the number of messages received and issued by each sensor node. Each sensor node sends a message, receives d messages from its neighbors and receives (or sends, if this sensor node is CH) a message containing the cluster key, that is, $d + 2$ messages by each sensor node. Table 2 shows that EAHKM has a low communication complexity over other schemes. In addition, EAHKM ensures a secure cluster formation phase that makes it adaptable for real-world applications. Table 3 summarizes whether the scheme supports: key refresh, cluster key establishment, addition of new sensor nodes and if the scheme necessitates that sensor nodes are aware of their locations after deployment.

EAHKM allows key refresh and nodes addition which makes it dynamic and flexible. The establishment of a cluster key allows a secure broadcast of messages in the cluster and sensor nodes are unaware of their locations after deployment.

Table 2. Comparison 1.

Schemes	communication complexity	Memory efficiency	KC	secured clusters formation
LKHW [8]	Tree formation	h+1	1	No
HKMS [9]	m (depend on TTL value)	d'+1	1	No
LEAP[10]	(2xd)+1	(3xd)+2+key chain of TESLA	1	No
LEAP+ [11]	(2xd)+1	(3xd)+2+key chain of TESLA	1	No
HIKES [12]	6xd	partial key escrow table+7 keys+encrypted nonce	1	No
SHELL [13]	d+1	k keys+key's identifier	1	No
EAHKM	d+2	4 + number of cluster members	1	Yes

d: number of neighbors. d': number of neighbors within the same cluster. h: height of the tree.

Table 3. Comparison 2.

Schemes	Key refresh	Node addition	Cluster key establishment	Location Acknowledgment
LKHW [8]	Yes	Yes	Yes	No
HKMS [9]	Yes	No	Yes	No
LEAP[10]	Yes	No	Yes	No
LEAP+ [11]	Yes	Yes	Yes	No
HIKES [12]	No	No	Yes	No
SHELL [13]	Yes	Yes	No	Yes
EAHKM	Yes	Yes	Yes	No

5.3 Simulation results

We implemented EAHKM, LEAP+ [11] and HKMS [9] using the MATLAB framework[1]. Simulation parameters are cited in Table 4.

Table 4. Simulation parameters.

Network area	250 m x 250 m
Base station location	(50 m, 50 m)
Initial energy	0,5 Joule
Packet size	4000 bits
E_{elec}	50 nJ/bit
E_{amp} $(\alpha = 2)$	$10pj/bit/m^2$
E_{amp} $(\alpha = 4)$	$0.0013pj/bit/m^4$
Distance d_0	87 m
Key size	128 bits

Figure 3 shows the communication overhead for different networks size, which consists of the average number of received messages by each sensor in the key

[1] MATLAB for MATrix LABoritory is a matrix-based system for scientific and engineering calculation.

establishment phase. For instance, in EAHKM each node receives 49 messages in a WSN of 200 nodes, which is a less cost than LEAP+ [11] and HKMS [9]. In LEAP+ [11], each sensor node broadcasts a message, receives d messages from its neighbors and then sends d messages to its neighbors to establish a cluster key. For HKMS [9], each CH broadcasts a number of messages that depend on the TTL value and receives d messages from its neighbors. Also, a node that is not a CH receives different messages from surrounding CHs and sends one message to a chosen a CH.

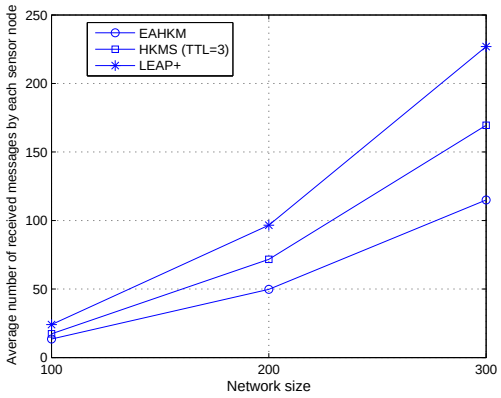

Fig. 3. Communication overhead.

Figure 4 presents the total storage requirement in different networks size. From this Figure we can see that EAHKM has a nearly total memory value than HKMS [9] and a very small value compared to LEAP+ [11]. In EAHKM, each sensor node has to store only three keys in its memory before deployment. After deployment, in a cluster of m nodes there are $m-1$ established keys. EAHKM is a memory-optimal scheme adapted to sensor nodes. When the network size increases, the total memory space in the network linearly increases in EAHKM.

Due to the energy constraint of sensor nodes, key management schemes should be energy-efficient. Figure 5 evaluates energy consumption during key establishment by calculating the average remaining energy in a sensor node. In HKMS [9], a cluster member node sends its messages to its CH through several hops which consumes more energy than a one hop transmission. However, in EAHKM each sensor node conserves its energy by choosing as a CH the sensor node S_i with the highest value $E_{S_i}/Level_{S_i}$, this means choosing the nearest to the BS (in number of hops) and the highest energy node to be a CH.

CHs perform more encrypt and decrypt operations and receive more messages than cluster member nodes. In EAHKM, sensor nodes with high energy

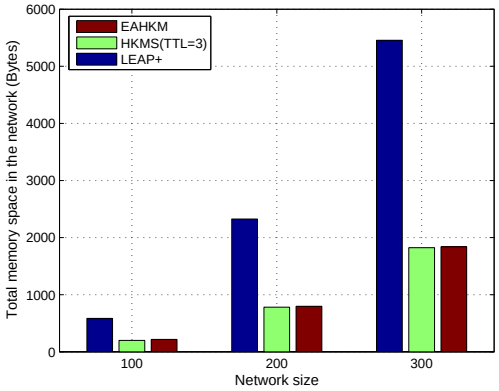

Fig. 4. Memory overhead vs. network size.

budgets are selected as CHs, this prolongs the lifetimes of low energy sensor nodes which extends the lifetime of the network.

Suppose an adversary that captures randomly sensor nodes in a network of 100 nodes. Resiliency is measured by the fraction of compromised links on the rest of communications in the network. Figure 6 illustrates the resiliency of EAHKM, LEAP+ [11], and HKMS [9] against node compromising attacks when an adversary captures randomly 1 to 5 sensor nodes. It is noted that EAHKM outperforms the other schemes. In EAHKM, if a compromised node is a cluster member, it does not affect other communications within its cluster and if the node is a CH, it does not affect communications in other clusters.

From the presented comparison and simulation results, EAHKM not only ensures a secure cluster formation, but also provides better performance in terms of energy efficiency, communication overhead, memory overhead, scalability and resilience to node compromising attacks.

6 Conclusion

Organizing WSNs in hierarchical topologies is a promising technique to save energy. This organization needs to be constructed in a secure way. Nevertheless, the use of wireless channels, the energy-constraint and the large number of deployed sensor nodes complicate the task of key management in WSNs. This work proposed a new hierarchical key management scheme called EAHKM. Our comparison and simulation results indicate that EAHKM presents a better performance than other key management schemes. In addition, EAHKM offers a secure bootstrapping to sensor nodes. This desired feature is not warranted by the discussed schemes.

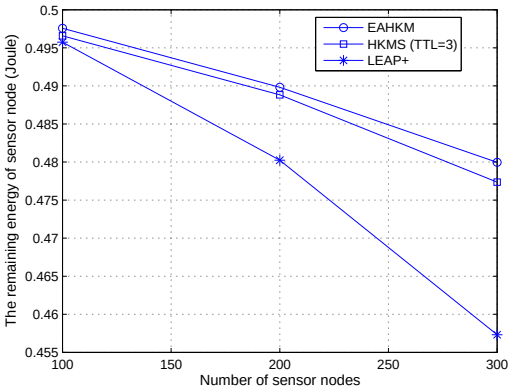

Fig. 5. Average energy consumed by sensor node during key establishment (including cluster heads).

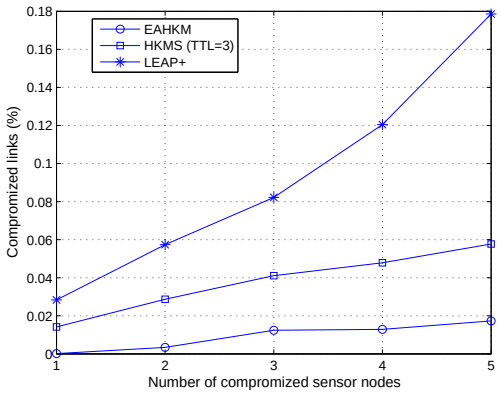

Fig. 6. Resilience to node compromising attack.

References

1. Zhou, Y., Fang, Y., Zhang, Y.: Securing wireless sensor networks: a survey. Communications Surveys & Tutorials, IEEE **10**(3) (2008) 6–28
2. Cardei, M., Ibriq, J., Ilyas, M., Mahgoub, I.: Encyclopedia of wireless and mobile communications. (2007)
3. Chen, C.Y., Chao, H.C.: A survey of key distribution in wireless sensor networks. Security and Communication Networks (2011)
4. Messai, M.L., Aliouat, M., Seba, H.: Tree based protocol for key management in wireless sensor networks. EURASIP Journal on Wireless Communications and Networking **2010** (2010) 59
5. Cheng, Y., Agrawal, D.P.: An improved key distribution mechanism for large-scale hierarchical wireless sensor networks. Ad Hoc Networks **5**(1) (2007) 35–48
6. Ya-nan, L., Jian, W., He, D., Li-jun, S.: Intra-cluster key sharing in hierarchical sensor networks. IET Wireless Sensor Systems **3**(3) (2013) 172–182
7. Reegan, A.S., Baburaj, E.: Key management schemes in wireless sensor networks: A survey. In: 2013 International Conference on Circuits, Power and Computing Technologies, IEEE (2013) 813–820
8. Di Pietro, R., Mancini, L.V., Law, Y.W., Etalle, S., Havinga, P.: Lkhw: A directed diffusion-based secure multicast scheme for wireless sensor networks. In: Proceedings of the International Conference on Parallel Processing Workshops, IEEE (2003) 397–406
9. Zhang, Y., Li, X., Liu, J., Yang, J., Cui, B.: A secure hierarchical key management scheme in wireless sensor network. International Journal of Distributed Sensor Networks **2012** (2012)
10. Zhu, S., Setia, S., Jajodia, S.: Leap: efficient security mechanisms for large-scale distributed sensor networks. In: Proceedings of the 10th ACM conference on Computer and communications security, ACM (2003) 62–72
11. Zhu, S., Setia, S., Jajodia, S.: Leap+: Efficient security mechanisms for large-scale distributed sensor networks. ACM Transactions on Sensor Networks (TOSN) **2**(4) (2006) 500–528
12. Ibriq, J., Mahgoub, I.: A hierarchical key establishment scheme forwireless sensor networks. In: 21st International Conference on Advanced Information Networking and Applications, IEEE (2007) 210–219
13. Younis, M., Ghumman, K., Eltoweissy, M.: Location-aware combinatorial key management scheme for clustered sensor networks. IEEE Transactions on Parallel and Distributed Systems **17**(8) (2006) 865–882
14. Ibriq, J., Mahgoub, I.: Hikes: Hierarchical key establishment scheme for wireless sensor networks. International Journal of Communication Systems **27**(10) (2014) 1825–1856
15. Chen, J.S., Hong, Z.W., Wang, N.C., Jhuang, S.H.: Efficient cluster head selection methods for wireless sensor networks. journal of networks **5**(8) (2010) 964–970
16. Eltoweissy, M., Heydari, M.H., Morales, L., Sudborough, I.H.: Combinatorial optimization of group key management. Journal of Network and Systems Management **12**(1) (2004) 33–50
17. Min, X., Wei-Ren, S., Chang-Jiang, J., Ying, Z.: Energy efficient clustering algorithm for maximizing lifetime of wireless sensor networks. AEU-International Journal of Electronics and Communications **64**(4) (2010) 289–298