A Secure Routing Protocol for Cluster-Based Wireless Sensor Networks Using Group Key Management

Kun Zhang
Northeastern University at Qinhuangdao,
Qinhuangdao 066004, China,
Email: lishaoh88@163.com

Cong Wang
Northeastern University,
Shenyang 110004, China,
Email: wangcong81121@163.com

Cuirong Wang
Northeastern University at Qinhuangdao,
Qinhuangdao 066004, China
Email: wangcr@mail.neuq.edu.cn

***Abstract*—Wireless sensor networks routing protocols always neglect security problem at the designing step, while plenty of solutions of this problem exist, one of which is using key management. Researchers have proposed many key management schemes, but most of them were designed for flat wireless sensor networks, which is not fit for cluster-based wireless sensor networks (e.g. LEACH). In this paper, we investigate adding security to cluster-based routing protocols for wireless sensor networks which consisted of sensor nodes with severely limited resources, and propose a security solution for LEACH, a protocol in which the clusters are formed dynamically and periodically. Our solution uses improved Random Pair-wise Keys (RPK) scheme, an optimized security scheme that relys on symmetric-key methods; is lightweight and preserves the core of the original LEACH. Simulations show that security of RLEACH has been improved, with less energy consumption and lighter overhead.**

I. Introduction

Routing protocols for wireless sensor networks (WSNs) [1] are classified as data-centric (flat) protocols, hierarchical (cluster-based) protocols and location-based protocols. Routing protocols for cluster-based WSNs [2, 3, 4] propose that, all of nodes are divided into a certain number of clusters, and each cluster chooses a cluster head (CH). Member nodes in each cluster send data collected to the CH, and the CH aggregates this coming data followed by sending it to base station (BS). The selection of CH on routing protocols for cluster-based WSNs always follows the "round" strategy, which improves network's energy efficient.

From the security's whole ground of view, LEACH (Low Energy Adaptive Clustering Hierarchy) is not secure enough, since it has no ability to resist some attacks in network layer. In this paper, we add security to LEACH by using key management scheme. Researchers have proposed a number of key management schemes [5, 6, 7, 8, 9] for WSNs. Most of the key management schemes assume the relationship between nodes is fixed, while clusters as well as the relationship between nodes in hierarchical protocol are dynamical, so these schemes designed for flat networks are not suitable for routing protocol for cluster-based WSNs.

In this paper, we show how random pairwise keys (RPK) scheme [6] can be used in secure communication in hierarchical (cluster-based) routing protocols such as LEACH. RPK scheme 1): is safer by realization of node-to-node authentication, and 2): saves energy. Although RPK scheme has above metrics, we can not directly introduce it into LEACH. This is because RPK scheme is a strategy based on probability, which has no ability to ensure all the adjacent nodes to share the key, and RPK scheme is limited by network scale. We substitute one-way hash function for the keys to resolve the two problems above of RPK scheme, making it well suitable for LEACH.

We focus on finding a way to ensure security and efficiency problem during communication between member nodes and CH in LEACH. In this paper, we propose RLEACH, a modified version of LEACH that bootstraps its security from improved random pair-wise key scheme. We then give a detailed analysis and performance evaluation of our scheme. Our main contributions are: 1) an effective solution to communication's security in LEACH is provided; 2) RPK scheme is improved to be fit for routing protocol for cluster-based WSNs.

The rest of this paper is organized as follows. In part 2, some main secure problems in LEACH are expatiated and we improve RPK scheme, making it fit for LEACH. In part 3, RLEACH is proposed, and its performance is evaluated in part 4. Finally, related work and conclusion are presented respectively in part 5 and part 6.

II. LEACH AND ITS VULNERABILITIES

A. Protocol Description

LEACH [2] is a self-organizing, adaptive clustering routing protocol. The key idea is to reduce the number of nodes communicating directly with the base station. LEACH balances nodes energy consumption in network by choosing cluster header randomly. Nodes organize several clusters automatically, and communicate with cluster header. The cluster process data aggregation and communicate with base station. LEACH performs local data fusion to "compress" the amount of data being sent from the clusters to the base station, further reducing energy dissipation and enhancing system lifetime.

The operation of LEACH is broken up into rounds, which begin with a set-up phase, when the clusters are organized,

978-1-4244-2108-4/08/$25.00 © 2008 IEEE

followed by a steady-state phase, when data transfer to the BS occur.

B. Security Vulnerabilities

Like most routing protocols for WSNs, LEACH is vulnerable to a number of security attacks, including spoofed, altered, and replayed information, selective forwarding, sinkholes attacks, sibyl attack, wormholes attack and HELLO flood attack etc. LEACH is more robust against attacks than most other routing protocols. In contrast to more conventional multi-hop schemes where nodes around the BS are especially attractive for compromise, CHs in LEACH communicate directly with the BS, can be anywhere in the network, and change from round to round. All these characteristics make it harder for an adversary to identify and compromise strategically more important nodes.

However, because it is a cluster-based protocol, relying fundamentally on the CHs for data aggregation and routing, attacks involving CHs are the most damaging. If an intruder manages to become a CH, it can stage attacks such as selective forwarding, the sibyl attack and HELLO flood attack.

Key management is an effective method to improve WSNs' security, some key management schemes have been specifically designed for WSNs. While they are well-suited for network organizations they were designed for, they are inadequate for others. These schemes typically assume that a node interacts with a quite static set of neighbors and that most of its neighborhood is discovered right after the deployment. However, clusters in LEACH are formed dynamically (at random) and periodically, which changes interactions among the nodes and requires that any node needs to be ready to join any CH at any time. In this paper, we choose RPK scheme as adding security to LEACH protocol.

C. RPK Schemes

RPK scheme is proposed by Chan-Perrig- Song. It is a scheme based on probability, a certain number nodes' shared-key are stored in each node's memory, ensuring network's connectivity by keeping probability of connection. The analysis of connection in RPK scheme is analogous to E-G protocol. Node's IDspace is introduced into this protocol for easy configuration. Not only key information but also according identifiers are conserved in node. For existing shared-key between any two nodes, RPK scheme can realize nodes authorization. Node-to-node authorization can realize a lot of security functions.

RPK scheme has many advantages, but doesn't adapt to LEACH protocol. Its main problem is that RPK scheme is a kind of probability-based scheme, which can keep the whole network's connectivity, but can not ensure all pairs of nodes sharing key-pair. Therefore, it is possible that some nodes could not join a cluster for their having no shared-key. We call these nodes orphans. The issues of orphans in LEACH were discussed in [10]. In this following, we improved RPK scheme to reduce orphans by enhancing connectivity of networks.

D. Improved RPK Scheme

A key pool's generation scheme was proposed by Kui Ren in [11]. The main point of this is that: key pool *K* in WSNs is composed of L different key chain C_i, $k = \bigcup_i C_i (i = 0, ..., L-1)$ and $C_i \bigcap C_j = \emptyset (i \neq j)$. Each key chain C_i is iteratively generated by a one-way hash function using key g_i and public seed. Thereby, the lth key of C_i can be computed: $k_{c_i,l} = H^l(seed, g_i)$. $H^l(seed, g_i) = H(H^{l-1}(seed, g_i), g_i)$, (C_i, l) is according key.

In RPK scheme, for nodes allocation in WSNs, we always assume every node's neighbor selection is random and all nodes' distribution is even. Yet, in practical application, all things are just the opposite. In many wireless network applications, thousands of sensor nodes can not be deployed artificially, and are diffused randomly by airplane or ship. Obviously, for the whole networks, the adjacent groups are tended to build security links, since the communication ability of sensor node is limited. The farther the group is, the less probability the secure connection is built, even impossible. We can make an assumption, there are n nodes in a sensor network, divided into g groups in terms of some rules, with n_a nodes in each group, where $g = n/n_a$. All the sensor nodes are scattered on group and distributed not even according to the former depiction. But from the whole ground point of view, their deployment is even.

[9]discussed deployment model of WSNs, which assumes establishing two dimensional Gaussian distribution (also called Normal distribution) after nodes' deployment. We also consider nodes' distribution problem. But different from above model, we divide all nodes of networks into group g_i, its probability to be neighbor node in the same group is the highest, the neighbor group $(g_{i+1}, g_{i-2}, g_{i+2}, g_{i-2}...)$ takes the second place, and the farther between two groups, the less probability to be neighbor nodes. Therefore, when choosing m neighbor node, we tend to choose from the neighbor group. If the two groups are apart from a certain distance, we will not choose it as neighbor node. In this way, since much more node's IDs of same group or neighbor group are stored, building a secure link between neighbor nodes is more conceivable.

The same as RPK scheme, we need to distribute a unique node ID_i for each node before deployment. Using node identifiers, we can realize node-to-node identifier in sensor networks. The difference is that, node identifier in RPK scheme is distributed randomly. We distribute node identifier in terms of its group.

For a sensor networks having n nodes, is divided into g groups. For $g = 1$ means all the sensor nodes are in the same group. Due to they have same hash and seed, any two nodes can be linked and compute the shared-key by the same one-way hash function, this make our approach evolve to a RPK scheme. Obviously, the network has high connectivity at this time but poor security. On the other side, $g = N$ means all sensor nodes are distributed randomly, our scheme has evolved to $pair - wise$ key scheme, which has high security and poor

connectivity.

Therefore, we can resize group to adapt to various requirements and form different sensor networks. For sensor networks application with high security requirement, we can increase the number of groups to improve security. For other sensor networks with high connectivity requirement, we can decrease the number of groups to attain.

III. RLEACH PROTOCOL

A. Description of RLEACH

The operation of RLEACH is composed of five phases:

1) ***Pre-distribution Phase****:* In this phase, each node is pre-distributed with its ID_i and an original key K_i which is related with ID_i, chooses m keys randomly from the key pool. The node can use those keys as the shared-key with the other nodes, and register the corresponding ID of the shared-key. The BS needs to record every node's ID and the whole net's key pool and one-way hash function $F()$. (Defining a one-way hash function $Hash(Seed_i, ID_x, ID_y)$, which is stored in each node's memory, and a public seed $Seed_i$ for the nodes, which are in the same group, for example: g_1 group is related with $Seed_i$, g_2 group is related with $Seed_2$ etc). ID_x, ID_y are two node identifiers which want to build a secure link. Then the node chooses m neighbor nodes from its neighbor group and distributes m keys to those nodes respectively. If the nodes are in the same group, we don't need to pre-distribute pair-key for them because they have the same public seed $Seed_i$ and function $F()$ in common. Nodes in the same group can use them to calculate the shared-key for their communication. If nodes are in different groups, they can use function $Hash(Seed_i, ID_x, ID_y)$ to compute the shared-keys and store the related node identifiers at the same time.

2) ***Shared-key discovery phase****:* In this phase, node broadcasts its ID (supposed as ID_i) to its neighbor nodes. Assume node B receives node A's identifier ID_i, node B judges that whether or not node A is in the same group at first. If they are, node B may use its identifier (supposed as ID_j) to calculate the conversation key ($K_i = Hash(Seed_i, ID_i, ID_j)$, and return the key K_i and ID_j to node A. Node A validates the key using the same hash function, if they have the same result, they can found their shared-key at last; otherwise, if these two nodes are in different groups, node B examines its key ring to discover whether or not they have a shared-key, if have, then through a shake hand process to affirm.

3) ***Cluster set-up phase****:* RLEACH is the same as LEACH, its operation is based on the "round". First, each node determines whether it is a CH of the current round through the calculation. Once the node determines that it is a CH in the current round, it will broadcast an advertisement message, which contains its node identifier and relevant information. Each node may receive from many advertisement messages of CHs, and it chooses the CH by the following principles, 1) whether or not the CH has a shared-key, and 2) if many CHs have shared-keys, according to the signal strength of the broadcasting advertisement message from CHs.

4) ***Schedule creation phase****:* After a period of time, CHs with all member nodes set up secure links. Based on the number of the nodes, CHs will generate a TDMA schedule, and broadcast it to their member nodes. TDMA mechanism makes all the members of the cluster nodes no data collisions and conflicts, and when a node to transmit data to its CH, the other member nodes can close their wireless devices, in order to save energy.

5) ***Data transmission phase****:* This phase mainly consists of two components, the member nodes sending data to CH and the CH sending data to BS. Each member node can sleep to save energy. CH must keep its receiver open to receive the data from its member nodes. In the process, in order to ensure the security of data transmission, the CH and the member nodes use the shared-key for authentication each other. The CH integrates and compresses received data to a new signal, and then sent it to BS with its ID. BS will use the original key K_i to validate whether the data is effective.

B. Analysis of RLEACH Protocol Security

LEACH has high-security character, which can resist many attacks such as spoofed, altered, or replayed information, sinkholes attacks and wormholes attacks. This is because, in LEACH, all the transmission from BS to CH, and from CH to common nodes, is only single-hop. After cluster congregation finished, data goes from member nodes to CH, and finally to BS, malicious node's spoofed, altered, or replayed information is meaningless. Analogously, wormholes generate a lingering link to attract traffics in the network. Nevertheless, information in LEACH is transmitting direct-aimed and single-hoped, nodes have no "interest" in short-delayed path. Therefore, LEACH can resist this kind of attack. LEACH has high ability to against simple sinkholes attacks. For nodes in cluster only transmit data to CH in LEACH, without through internal node, nodes in cluster will not generate sinkholes. In addition, "round" working in CH makes itself isolated from sinkholes attacks.

RLEACH protocol in principle based on the design of clusters with LEACH protocol. Therefore, RLEACH protocol can resist above three attacks. RLEACH protocol is on the basis of LEACH protocol to establish security mechanisms, so it can resist selective forwarding, the sibyl attacks and HELLO flood attack.

1) ***Against the Selective Forward Attack****:* For the LEACH protocol, selective forward attack often happen with the HELLO flood attack combination. One or more malicious nodes can send high-power radio signal, and attract the communication flows of the network. They have disposed of all (or part of) the messages, making the network work abnormal. If a malicious node becomes a CH, it may be bring on selective forward attack. In RLEACH protocol, the CHs and all member nodes in the cluster establish shared-key, and malicious nodes in the network need to capture many nodes to achieve the objective that they become CHs, which requires pay a high price, so RLEACH protocol can resist select forward attacks.

*2) **Against the Sibyl Attack**:* According to the CH mechanism of LEACH, malicious node can use Sibyl attack. A malicious node in the network may present many different identities to other nodes, and announce that there are a lot of energy, in order to increase the probability that it will be selected as a CH. RLEACH protocol uses the improved RPK scheme, through the use of the corresponding shared-key, and achieves node-to-node authentication. Because of different identities of the same node can not be certified, and the CH to the BS data transmission, BS also need the node ID and its initial key to verify their identity, so RLEACH can resist the sibyl attack.

*3) **Against the Hello Flood Attack**:* Because of the competition in CH process of the LEACH protocol, the member node joins the cluster according to signal strength of the CH, the malicious node can be easily attacked by HELLO flood attack. Malicious node broadcasts a high-power radio, wants to make a large number of nodes added to the its cluster. Then malicious node can be used in other attacks, such as selective forward, modify data packets, and so on. In the RLEACH protocol, inter-node communications need for certification, if not passed certification, it can not be the network nodes.

IV. PERFORMANCE EVALUATION

In this section, we analyze the performance and security of our scheme. We present our analytical results on the following three metrics:

Network connectivity. We use local connectivity to refer to the probability of any two neighboring nodes sharing at least one key space. We use P interchangeably to refer to the local connectivity. The local connectivity directly affects the performance of the scheme.

Memory occupation. To address the limited memory constraint, small number of keys should be promised while supporting same or higher level of security.

Network lifetime. Time when the last node is dead, which is an important parameter of measuring energy consumption for the network.

A. Analysis of Network Connectivity

In this paper, we assume n to be the number of nodes in the network, and the network is divided into g groups, then the number of nodes in each group is $n_a = n/g$, P is the probability of between two nodes sharing a shared-key, and the incident A and B were defined as follows:

A: Two nodes do not share any key

B: Two nodes in different groups

$$P = 1 - P_r(A) = 1 - P_r(AB) = 1 - P_r(A|B)\,P(B) \quad (1)$$

In Equation (1), for $P_r(A|B)$ and $P(B)$, consider the probability of worst-case scenario, obtained respectively:

$$P_r(A|B) = \left(1 - \frac{m}{n - n_a}\right)^2 \quad (2)$$

$$P(B) = \frac{\binom{n/n_a}{2} n_a^2}{\binom{n}{2}} = \frac{n - n_a}{n - 1} \quad (3)$$

From (1)(2)(3) we have

$$P = 1 - \frac{(n - n_a - m)^2}{(n - n_a)(n - 1)} \quad (4)$$

When the network is large-scale, we have $n \approx n-1$. Thus, Equation (4) can be simplified

$$P = \frac{1}{g} + 2\frac{m}{n} - \frac{g}{g-1}\left(\frac{m}{n}\right)^2 \quad (5)$$

RPK scheme of network connectivity can be used $P = m/n$ to compute probabilities, compared with the previous Equation, as $\frac{1}{g} - \frac{g}{g-1}\left(\frac{m}{n}\right)^2 > 0$, therefore our scheme's connection probability is at least 2 times larger than RPK scheme.

B. Analysis of Memory Occupation

As our scheme divides sensor nodes into groups, if the same group nodes are adjacent, its connective probability is very high (equivalent to 1), and the same group nodes need not to store shared-keys. Compared to RPK scheme, our scheme needs less memory in the same network size. Fig.1 shows the relationship between the number of keys and the connectivity probability of networks, the size of the network in RPK scheme and our scheme (were taken at 5 and 10) is 1000. Note that in the same situation, our scheme needs to store keys fewer, that is, our scheme reduces the memory overhead.

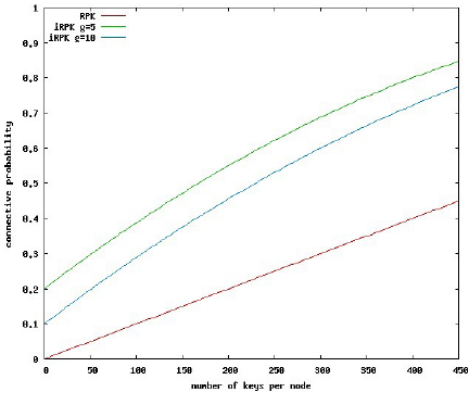

Fig. 1. Analysis of Node Memory Occupation

C. Analysis of Network Lifetime

Wireless sensor networks can be regarded as a graph with vertices set V and edges set E, which can be described as $G = \{V, E\}$ [12]. V represents the sensor nodes including base station, E represents the connection among sensor nodes. We do not consider the energy consumption of the base station, because base station need to communicate with other networks and bear a large number of calculation works. If two vertices of an edge do not include the base station, the energy consumption of any edge in G is:

$$C_{ij}(k) = 2E_{elec} \cdot k + \varepsilon_{amp} \cdot k \cdot d_{ij}^2 \quad (6)$$

If a node communicates with base station, then:

$$C_i(k) = E_{elec} \cdot k + \varepsilon_{amp} \cdot k \cdot d_{ib}^2 \quad (7)$$

C_{ij}Can be regarded as the energy cost between nodes i and j here. C_i is the energy cost between node i and base station. d_{ij} is distance between nodes i and j. d_{ib} is distance between node i and base station.

We use NS-2 discrete-event simulator to evaluate the performance of the proposed routing protocol. The following are the important simulation assumptions. Sensing field configuration is $100 \times 100m^2$ with 100 sensor nodes and 1 base station, packet length is 500 bytes, $E_{elec} = 50nJ/bit$, $\varepsilon_{amp} = 10pj/bit/m^2$. We set an initial energy of $3J$ for each node (except base station) in all our policies.

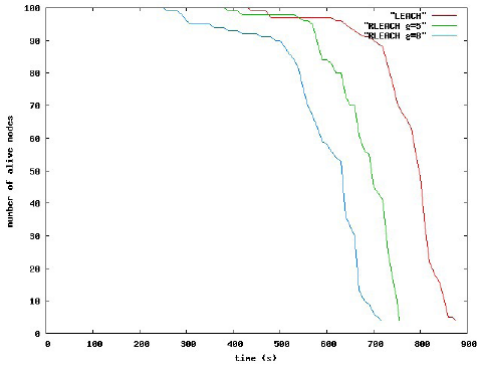

Fig. 2. Analysis of Network Lifetime

Fig. 2 shows the network lifetime for LEACH and RLEACH in different groups. Note that the lifetime is longer in LEACH than in any group of RLEACH, and larger values of g lead to shorter lifetime. The reason is that RLEACH establishes security mechanism by using key management. Then it could increase the energy consumption and reduce the lifetime of the network. In RLEACH, the smaller groups, the higher probability to share keys between the nodes. And it could decrease the energy consumption of establishing shared-key, and could prolong the lifetime of the network (comparing with the larger groups), but it will reduce the security performance via the analysis in section 2.4. Hence we should balance the network security and lifetime, according to the requirements of application.

V. RELATED WORK

For resource-constrained WSNs security problems have been widespread concerned, we focus on the cluster-based routing protocol's security issues by using the key management schemes. Most of them are designed for the multi-hop communications networks, and they are not well suitable for clusters-based routing protocols in WSNs.

Among those specifically targeted of cluster-based sensor networks, Bohge et al. [13] proposed an authentication framework for a concrete 2-tier network organization, in which a middle tier of more powerful nodes between the BS and the ordinary sensors were introduced for the purpose of carrying out authentication functions. In their solution, only the sensor nodes in the lowest tier do not perform public key operations. More recently, Oliveira et al. [14] proposed a solution that relies exclusively on symmetric key protocols and is suitable for networks with an arbitrary number of levels; and Leonardo B. Oliveira et al. proposed SecLEACH [10], which we discussed in Section 2.

VI. CONCLUSION

This paper focuses on security issues in cluster-based WSNs routing protocols, and proposes RLEACH, the protocol is based on the LEACH, and enhances communications security between nodes. RLEACH uses the improved random key management scheme for security. Analysis shows that RLEACH has better security features, and can resist a variety of network attacks. Through performance evaluation, we find the overhead which the RLEACH protocol leads to is acceptable, and it improves the network connectivity, and reduces the memory overhead.

REFERENCES

[1] Kemal A and Mohamed Y. A Survey on Protocols for WSNs. Communications Magazines, 2002, 40(8), pages 102-114.

[2] W. Heinzelman, A. Chandrakasan, H. Balakrishnan, Energy-efficient communication protocol for WSNs, in: Proceeding of the Hawaii International Conference System Sciences, Hawaii, January 2000.

[3] S. Lindsey, C.S. Raghavendra, PEGASIS: power efficient gathering in sensor information systems, in: Proceedings of the IEEE Aerospace Conference, Big Sky, Montana, March 2002.

[4] M. Younis, M. Youssef, K. Arisha, Energy-aware in cluster-based sensor networks, in: Proceedings of the 10th IEEE/ACM International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS2002), Fort Worth, TX, October 2002.

[5] Eschenauer L, Gligor V D. A key management scheme for distributed sensor networks. In Proceedings of the 9th ACM Conference on Computer and Communication Security. USA: ACM, pages 41-47, 2002.

[6] Chan H, Perrig A, Song D. Random key pre-distribution schemes for sensor networks. In Proceedings of the IEEE Computer Society Symposium on Security and Privacy. Piscataway, USA: IEEE, pages 197-213, 2003.

[7] Kejie Lu, Yi Qian, Mohsen Guizani and Hsiao-Hwa Chen. A framework for a distributed key management scheme in heterogeneous WSNs. Wireless Communications, IEEE Transactions on Volume 7, Issue 2, Pages: 639- 647, February 2008.

[8] Yun Zhou and Yuguang Fang. A Two-Layer Key Establishment Scheme for WSNs. Mobile Computing, IEEE Transactions on Volume 6, Issue 9, Pages: 1009 - 1020, Sept. 2007.

[9] Wenliang Du, Jing Deng, Yunghsiang S.Han, Pramod K.Varshney. A Key Predistribution Scheme for Sensor Networks Using Deployment Knowledge. IEEE Transactions on Dependable and Secure Computing, Vol.3, NO.1, pages 62-77, January-March 2006.

[10] Leonardo B. Oliveira, Hao C. Wong, M. Bern, A. A. F. Loureiro. SecLEACH-A Random Key Distribution Solution for Securing Clustered Sensor Networks. The Fifth IEEE International Symposium on Network Computing and Applications (NCA'06), 2006.

[11] Kui Ren, Kai Zeng, Wenjing Lou. On efficient key pre-distribution in large scale WSNs. Military Communications Conference, 2005. MILCOM 2005. IEEE 2005, 10, pages 20-26.

[12] Zhou Zude, Xu Chao, Liu Quan. Energy Modeling and HMST-Based Wireless Sensor Networks Routing Protocol. Industrial Electronics and Applications, Pages: 827-830. ICIEA, 2nd IEEE Conference on 23-25 May 2007.

[13] M. Bohge and W. Trappe. An authentication framework for hierarchical ad hoc sensor networks. In 2003 ACM workshop on Wireless security, pages 79-87, 2003.

[14] L. B. Oliveira, H. C. Wong, and A. A. F. Loureiro. Lha-sp: Secure protocols for hierarchical WSNs. In 9th IFIP/IEEE International Symposium on Integrated Network Management (IM'05), pages 31-44, 2005.