A Key Management Protocol for Wireless Sensor Networks with Multiple Base Stations

Yong Wang
University of Nebraska-Lincoln, ywang@cse.unl.edu

Byrav Ramamurthy
University of Nebraska-Lincoln, bramamurthy2@unl.edu

Yuyan Xue
University of Nebraska-Lincoln, yxue@cse.unl.edu

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the ICC 2008 proceedings.

ICC '08. IEEE International Conference on Communications, 2008.
Digital Object Identifier: 10.1109/ICC.2008.314 Publication Year: 2008 , Page(s): 1625 - 1629

A Key Management Protocol for Wireless Sensor Networks with Multiple Base Stations

Yong Wang, Byrav Ramamurthy, and Yuyan Xue
Department of Computer Science and Engineering
University of Nebraska-Lincoln
Lincoln, NE 68588-0115 USA
{ywang, byrav, yxue}@cse.unl.edu

***Abstract*— Most of the proposed key management protocols for wireless sensor networks (WSNs) in the literature assume that a single base station is used and that the base station is trustworthy. However, there are applications in which multiple base stations are used and the security of the base stations must be considered. This paper investigates a key management protocol in wireless sensor networks which include multiple base stations. We consider the situations in which both the base stations and the sensor nodes can be compromised. The proposed key management protocol, mKeying, includes two schemes, a key distribution scheme, mKeyDist, supporting multiple base stations in the network, and a key revocation scheme, mKeyRev, used to efficiently remove the compromised nodes from the network. Our analyses show that the proposed protocol is efficient and secure against the compromise of the base stations and the sensor nodes.**

I. Introduction

Wireless sensor networks (WSNs) are promising solutions for many applications and security is an essential problem for WSNs. Among all security issues in WSNs, key management is a core mechanism to ensure the security of network services and applications in WSNs.

Most of existing works in the literature consider a WSN with a single base station. However, as the size of the sensor network grows, the distances between the base station and the associated sensor nodes also become larger. The increase in the distances may result in the following changes:

- The performance of the network degrades due to the increase in the packet loss as the size of the network grows.
- The lifetimes of the sensor nodes are reduced due to the high energy consumption involved in query-flooding and data-retrieval between the sensor nodes and the single base station.
- The lifetime of the sensor network is shortened dramatically because the energy of the sensor nodes close to the base station is exhausted quickly.

The deployment of multiple base stations in a sensor network is a promising solution [1], [2]. By allowing multiple base stations in a sensor field, we can reduce the distances between the sensor nodes and the base stations, and thus reduce the energy consumption in sensor nodes and improve the performance of the sensor network.

In this paper, we consider a WSN with multiple base stations. We consider the situations in which both the base stations and the sensor nodes could be compromised. Since the base stations are usually resource-rich devices, we assume that the base stations can communicate with each other directly using high speed wireless links. The base stations can work cooperatively in data acquisition in the network. To detect a compromised base station or a sensor node, intrusion detection techniques are involved. Intrusion detection is out of the scope of this paper. We assume that there are some methods to detect a compromised base station or sensor node.

A key management protocol designed for a WSN with multiple base stations must satisfy the following requirements:

1) The base stations and the sensor nodes can authenticate the messages from each other.
2) The capture or compromise of any base stations should not affect the security of the sensor network.
3) The capture or compromise of any sensor nodes should not affect the security of the sensor network.
4) The capture or compromise of both a base station and sensor nodes should not affect the security of the sensor network.

In this paper, we propose a key management protocol, mKeying, for the WSNs with multiple base stations. mKeying includes two schemes, a key distribution scheme, mKeyDist, supporting multiple base stations in the network, and a key revocation scheme, mKeyRev, used to efficiently remove the compromised nodes from the network.

The rest of the paper is organized as follows: Section II discusses related work. Section III presents the proposed key management protocol and is followed by Section IV analyzing the security and the performance of the proposed protocol. Section V describes the simulations and results. Section VI finally concludes the paper.

II. Related work

Many key management protocols have been specifically designed for WSNs [3], however, most of the proposed key management protocols consider a single base station in the WSN and assume that the base station is trustworthy. In consideration of multiple base stations and their security, none of the current proposed key management protocols can be directly used. For example, most proposed key management protocols assume that each sensor node is preloaded with a master key, shared by all nodes in the sensor network, or

978-1-4244-2075-9/08/$25.00 ©2008 IEEE

a pairwise key, shared with the base station [3]. A simple clone of a base station to multiple base stations indicates that all the base stations use the same master key or the same key set of pairwise keys. Several issues arise under such assumptions. First, the base station can authenticate the messages from the sensor nodes; however, the sensor nodes cannot authenticate the messages from the base stations. Second, once a base station is captured or compromised, the master key and the whole pairwise key set are exposed. As a result, any adversaries with this key set can utilize the WSN without incurring any cost to deploy the sensor network.

A few papers discussed multiple base stations in the literature [1], [2], [4]. However, most of the papers focus on how to save energy and improve network lifetime by using multiple base stations in the network. None of these works discussed key management issues. In [5], the authors considered a sensor network with multiple mobile base stations and proposed some schemes to restrict a mobile base station without impeding its capability to carry out any authorized operations for an assigned task. However, the proposed revocation schemes were based on local multicast and may be not scalable to large sensor networks.

Next, we present our proposed key management protocol, mKeying, for the WSNs with multiple base stations.

III. mKeying: A key management protocol for WSNs with multiple base stations

In this section, we propose a key management protocol, mKeying, for WSNs with multiple base stations. We assume that there are at most m base stations and n sensors in the network ($m \ll n$). The base stations are deliberately distributed in the field to ensure that each sensor node can be reached by at least γ ($\gamma < m$) base stations within one hop of communication range. The base stations and the sensor nodes are identified by globally unique numbers called *base station identifier* and *sensor identifier* respectively. There is no overlapping between the base station identifier space and the sensor identifier space.

mKeying includes two schemes, a key distribution scheme, mKeyDist, and a key revocation scheme, mKeyRev. In the remainder of this section, we first motivate and present an overview of the mKeying protocol. Then, we introduce security model and demonstrate broadcast authentication, key distribution, revocation, and bootstrap processes using the mKeying protocol.

A. mKeying Overview

Our proposed key management protocol supports the establishment of five types of keys for each sensor node—*pairwise key*, *individual key*, *session key*, *encryption key* and *message authentication code (MAC) key*.

Pairwise key: Each sensor shares a pairwise key with each of its immediate neighbors. The pairwise key is used to generate the encryption key and the MAC key to ensure data confidentiality and authentication.

Individual key: Each node has a unique key shared pairwise with the base station. This key is used for the base station to authenticate individual sensor node.

Session key: This is a global key shared by all nodes in the network. The need for the session key is motivated by the key revocation scheme. We present an efficient key revocation scheme, KeyRev, for a WSN with single base station using a globally distributed session key in [6]. In this paper, we extend the KeyRev scheme to support multiple base stations in the network (Section III-E).

Encryption key and message authentication code key: The encryption key and the MAC key are used to ensure the security of the communication in a sensor network. The encryption key and the MAC key are derived from the pairwise key and the session key (Section III-D).

B. Security model

In literature, most research studies assume that the base station is trustworthy and consider only the compromise of the sensor nodes. However, due to the hostile environments in the deployment area, it is necessary to consider the security of the base stations as well. In this paper, we consider the situations in which both the base stations and the sensors can be compromised.

C. Broadcast authentication

We limit the capability of broadcasting authentication messages only to the base stations. We adopt the practical broadcast authentication protocol proposed in [7] to support multiple base stations in the WSNs. Note that the broadcast authentication protocol for WSNs with multiple base stations must be properly designed. A simple clone of $\mu TESLA$ instance for each mobile base stations does not work. Two schemes were proposed in [7]: a basic approach and a special instantiation for long-lived senders. We use the second one.

D. mKeyDist, key distribution scheme

1) Pairwise key and individual key: The pairwise key and the individual key can be set up using Blundo's theory [8].

Before sensor nodes are distributed, the setup server randomly generates a bivariate *t*-degree polynomial

$$f(x, y) = \sum_{i,j=0}^{t} a_{ij} x^i y^j$$

over a finite field F_q where q is a prime number that is large enough to accommodate a cryptographic key such that it has the property of $f(x, y) = f(y, x)$. For each sensor *i*, the setup server computes a polynomial share of $f(x, y)$, that is, $f(i, y)$, and loads the single-variate polynomial $f(i, y)$ to the sensor *i*. For any two sensor nodes *i* and *j*, node *i* can compute the common key $f(i, j)$ by evaluating $f(i, y)$ at point *j*, and node *j* can compute the same key $f(j, i) = f(i, j)$ by evaluating $f(j, y)$ at point *i*. Thus, both the sensor i and the sensor j can find the shared pairwise key.

The individual key can be set up in the same way. Before the base stations are deployed, each base station j is loaded with a single-variate polynomial $f(j, y)$. By exchanging the identifiers between a base station and a sensor node, both the base station and the sensor node can find the shared individual key.

2) Session key: The session key is a global key shared by all the nodes. We propose to use distributed key schemes such as secure group communication schemes to generate a group key among the base stations first. Then, the session key can be derived from the group key and dynamically distributed to the sensor nodes. The secure group communication schemes are out of the scope of this paper. Please refer to [9] for details.

Let K_g be the group key generated by the secure group communication scheme, the session key K_s^i in i-th session can be derived from the group key using a pseudo-random function $\mathcal{G}$:

$$K_s^i = \mathcal{G}(K_g, i)$$

The distribution of session key is based on the personal key share distribution scheme [10]. Let K_s^i be the current session key in the i-th session, the session key distribution process is described below:

1) Secret share distribution: In the pre-distribution stage, the setup server randomly picks m $2t$-degree masking polynomial, $h_j(x) = h_{j,0} + h_{j,1}x + \cdots + h_{j,2t}x^{2t}$, $1 \leq j \leq m$, over F_q. Each sensor s_u is preloaded with the personal secret, $S_u = \{h_j(u) | 1 \leq j \leq m\}$. Each base station b_j gets the polynomial $h_j(x)$.
2) Distinct share broadcast: Given a set of revoked sensor nodes, $R = \{r_1, r_2, \cdots, r_w\}$, $w \leq t$, the base station b_j randomly picks a t-degree polynomial $p_j(x)$ and constructs $q_j(x) = K_s^i - p_j(x)$. Then, the base station distributes the shares of t-degree polynomial $p_j(x)$ and $q_j(x)$ to non-revoked sensors via the following broadcast message:
$$\begin{aligned} M &= \{R\} \qquad (1) \\ &\cup \{P_j(x) = g(x)p_j(x) + h_j(x)\} \\ &\cup \{Q_j(x) = g(x)q_j(x) + h_j(x)\} \end{aligned}$$
where the revocation polynomial $g(x)$ is constructed as $g(x) = (x - r_1)(x - r_2)\cdots(x - r_w)$. The broadcast authentication is ensured by the scheme in [7].
3) Session key recovery: If any non-revoked sensor nodes s_u receives such a broadcast message, it evaluates the polynomial $P_j(x)$ and $Q_j(x)$ at point u and gets $P_j(u) = g(u)p_j(u) + h_j(u)$ and $Q_j(u) = g(u)q_j(u) + h_j(u)$. Because s_u knows $h_j(u)$ and $g_j(u) \neq 0$, it can compute $p_j(u) = \frac{P_j(u) - h_j(u)}{g(u)}$ and $q_j(u) = \frac{Q_j(u) - h_j(u)}{g(u)}$. s_u finally can compute the new session key $K_s^i = p_j(u) + q_j(u)$. The revoked sensors cannot recover the session key because $g(x) = 0$.

A sensor node can recover multiple copies of the session key since each sensor node can be reached by at least γ base stations. A session key is only accepted when the sensor node restores at least γ copies of the session key.

3) Encryption key and message authentication code key: Let $K_{A,B}$ denote the pairwise key shared between the sensor A and the sensor B, and K_s^i be the current session key. The encryption key K_{encr} and the message authentication code (MAC) key K_{mac} are calculated as follows [6]:

$$\begin{aligned} K_{encr} &= \mathcal{F}(MAC(K_{A,B}, K_s^i), 1) \\ K_{mac} &= \mathcal{F}(MAC(K_{A,B}, K_s^i), 2) \end{aligned}$$

where $\mathcal{F}(K, x)$ is a pseudo-random function and x is an integer 1 or 2 for generating K_{encr} or K_{mac} respectively.

The complete message A sends to B is:

$$A \longrightarrow B : \{M|T_s\}_{K_{encr}}, MAC(K_{mac}, \{M|T_s\}_{K_{encr}}) \qquad (2)$$

where M is the message, and T_s is the timestamp when sending the message. The sensor B can calculate the same K_{encr} and K_{mac} with the sensor A. Thus, B can authenticate and decrypt the message.

E. mKeyRev, key revocation scheme

We consider three situations in the key revocation process: the compromise of a base station, the compromise of a sensor node, and the compromise of both a base station and a sensor node. We assume that each sensor maintains a list: *node revocation list (NRL)*. A *NRL* includes all the entity (base station or sensor node) identifiers which have been revoked in the network. The revocation list is empty initially and will be populated as the time goes by. The revocation list is checked for any incoming and outgoing messages to ensure that only the valid entities are involved in the network.

Case 1: When a base station is compromised, the compromised base station needs to be removed from the network. Since the security of the communication is ensured by the encryption key and the MAC key which are both bound up with the session key, the compromised base station can be removed from the network if it cannot reveal the next session key. To do this, the compromised base station will be forced to leave the group and a new group key is generated. Thus, when the new session key is distributed to the network, the compromised base station will be removed from the network.

Since a sensor node accepts a session key only when it can restore at least γ ($\gamma < m$) session keys for the same session from the distinct share broadcast messages (Equation 1), an attacker must compromise γ base stations to start the revocation attack.

Case 2: If a sensor node is found to be compromised, the compromised sensor can be removed using the session key distribution scheme. After the compromised sensor identifier is added to the revocation list, the revoked sensor cannot recover the new session key since $g(x) = 0$ and thus the revoked sensor cannot reveal the new encryption keys and the MAC keys. Although the compromised sensor still has the pairwise keys with its neighboring nodes, it cannot decrypt any messages and authenticate itself in the network.

Since the capability of broadcasting authentication messages is limited to the base stations only (ensured by the scheme in

[7]), a compromised sensor node cannot start the revocation attack either.

Case 3: In case that both a base station and a sensor node are compromised, the base station needs to be revoked first, followed by the revocation of the sensor node. The detail steps are described in Case 1 and Case 2.

F. Bootstrap

To bootstrap a WSN using mKeying, pre-distributed key materials are required to be loaded on the sensor nodes before they are deployed. These pre-distributed key materials include the polynomials used to set up the pairwise key, the secret shares to recover the session key, and the required materials for broadcast authentication [7]. After the sensor nodes are deployed, the pairwise keys are set up first, followed by the distribution of the session key. Then, the encryption keys and the MAC keys can be set up on the fly.

IV. Security and performance analysis

In this section we first discuss the security of the protocol. Then, we analyze the computation, communication costs and storage requirement of mKeying protocol.

A. Security analysis

Our proposed key management protocol—-mKeying, satisfies the following properties:

Property 1 Only the authorized sensors can communicate in the network. The communication among the sensors is ensured by the encryption key K_{encr} and the MAC key K_{mac} as shown in Equation 2. Unauthorized sensors (outside attackers) cannot participate the communication without proper assigned key materials.

Property 2 The session key distribution process is secure. The distribution of session key is based on the personal key share distribution scheme [10]. A revoked sensor cannot recover the session key because $g(x) = 0$. Since the broadcast authentication is ensured by [7], an outside attacker cannot masquerade a base station disseminating a session key and start a revocation attack either.

Property 3 The key revocation scheme is secure inspite of not removing the pre-distributed key materials at a compromised sensor node. Although, without removing the pre-distributed key materials, the compromised sensor could reveal the pairwise keys, the adversaries cannot figure out the encryption key K_{encr} and MAC key K_{mac} if the session key is updated. In the worst case, an adversary might use chosen plaintext attack to crack the session key, however, the attack itself is also time consuming. As long as the session key lifetime is less than the session key cracking time, the proposed key revocation scheme is secure.

B. Performance analysis

Because base stations are usually regarded as resource-rich nodes, we focus on the performance of sensor nodes:

1) Computation cost: There are five different types of keys in our proposed protocol. To calculate the pairwise key, the individual key, and the session key, polynomial evaluation is required. The computation of polynomial evaluation is efficient. The calculation of the encryption key and the MAC key is based on a pseudo-random function. Thus, the key distribution scheme is efficient in computation.

2) Communication cost: Since the setup of the pairwise key and the individual key does not involve any extra message exchanges among the sensor nodes and the base stations (the base station identifier and the sensor identifier can be send along with a message between the sensor node and the base station), we focus on the distribution of the session key. The session key is distributed to the network using broadcasting message in one round. The maximum size of the broadcast message in bits is decided by $S = (5t + 2) \log q$. Assume B indicate the transmission rate of the base stations, L be the maximum range between the base stations and the sensor nodes. The session key distribution time can be calculated as:

$$t_s = \frac{S}{B} + \frac{L}{3 * 10^8} \approx \frac{(5t + 2) \log q}{B}$$

(the propagation delay is very small compared with the transmission time). In case a sensor node is compromised, no extra messages are required to remove the compromised sensor node.

3) Storage requirement: Let d represent the number of neighboring nodes around a sensor and m be the number of the base stations. Each sensor node need d storage units for the pairwise keys, m storage units for the individual keys, and three storage units for the session key, the encryption key and the MAC key. In addition, to calculate the pairwise key and the individual key, each sensor need $t + 1$ storage units for the t-degree polynomial and each sensor has also to be loaded with m secret shares to recover the session key. Thus, the total storage units of keys required for each sensor is $2m+d+t+4$.

V. Simulations and discussion

The proposed key distribution scheme, mKeyDist, does not require any extra message exchanges among the sensor nodes and the base stations. Thus, the key distribution scheme does not add any extra communication overhead. In this section, we focus on the performance of the key revocation scheme, mKeyRev. We evaluated the performance of the mKeyRev scheme in SENSIM, a component-based discrete-event simulator for sensor networks. We consider two sensor network experimental settings: a small-scale sensor network with 100 nodes uniformly dispersed in a field with dimension $100m \times 100m$ and a large-scale sensor network with 1000 nodes uniformly dispersed in a field with dimension $2000m \times 2000m$. In both the networks, we assume each sensor node can be reached by at least four base stations ($\gamma = 4$).

We compare the mKeyRev scheme with the centralized key revocation scheme in [3] (referred to as EsRev scheme) and the GPSR-based revocation scheme proposed in [5] (referred to as the GPSRRev scheme). Both of the schemes, EsRev scheme and GPSRRev scheme, are designed for single-base-station WSNs. For these two schemes, we assume the base station is located at the center of field. We further assume that the sensor filed in the GPSRRev scheme is divided into four subareas.

The evaluation metrics include the key revocation time and the average energy consumption per node to revoke a compromised sensor in the network. The key revocation time is the time duration from when the key revocation protocol starts until all the uncompromised sensor nodes receive the key revocation message. We consider the mKeyRev scheme operating on a finite field F_q, where q is a 56-bit integer. The polynomial degree t in the mKeyRev scheme is set to four. We use the simulator parameters that represent the Mica2 Mote radio characteristics. For each test, we randomly select one sensor to be revoked and run the simulation ten times. The average value is measured.

Tables I and II show the key revocation time and the average energy consumption to revoke a compromised sensor node in the two sensor networks. As the tables show, the GPSRRev and the mKeyRev schemes are much better than the EsRev scheme in both the key revocation time and the average energy consumption per node. The GPSRRev scheme is better than the mKeyRev scheme in the key revocation time in the small-scale sensor network; however, the mKeyRev scheme is better than the GPSRRev scheme in the large-scale sensor network. Further, in both of the networks, the mKeyRev scheme consumes less energy than the EsRev and the GPSRRev schemes. Note that the key revocation time and the average energy consumption to revoke a single sensor node by using mKeyRev scheme have only a slight difference between the 100-node sensor network and the 1000-node sensor network. It indicates that the mKeyRev scheme is scalable to large-scale sensor networks. The good performance of the mKeyRev scheme is due to the efficient session key distribution scheme shown in Section III-D. The mKeyRev scheme is the best in overall performance for removing a compromised sensor node in the network.

TABLE I

KEY REVOCATION TIME.

Scheme	100-node WSN Time (seconds)	1000-node WSN Time (seconds)
EsRev	49.63	496.06
GPSRRev	1.02	4.04
mKeyRev	2.36	2.48

TABLE II

AVERAGE ENERGY CONSUMPTION TO REVOKE A COMPROMISED SENSOR.

Scheme	100-node WSN Energy (joules)	1000-node WSN Energy (joules)
EsRev	0.71	7.14
GPSRRev	0.19	0.29
mKeyRev	0.04	0.04

Notice that the proposed protocol requires a session key to be distributed to the network during each session. The duration of the session time could be set and adjusted dynamically according to the application to reduce the background traffic in the sensor network. In addition, a few papers [11], [12] have been proposed to address the key management issues in heterogeneous sensor networks which can also be used to improve the performance of the sensor network when the size of the network grows. We are confident that our proposed protocol can also be applied to heterogeneous sensor networks.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we propose a key management protocol, mKeying, for wireless sensor networks with multiple base stations. mKeying includes two schemes, mKeyDist and mKeyRev. Unlike most existing key management protocols for wireless sensor networks which assume that a single base station is used and the base station is trustworthy, the mKeying protocol supports multiple base stations and considers the situations in which both the base stations and the sensor nodes can be compromised. Our analyses show that the proposed protocol is efficient and secure against the compromise of the base stations and the sensor nodes.

In this paper, we assume that an appropriate routing protocol is used to support multiple base stations in the network. Further investigations include the security issues related to data acquisition and the routing issues arising in supporting multiple base stations.

REFERENCES

[1] S. R. Gandham, M. Dawande, R. Prakash, and S. Venkatesan, "Energy efficient schemes for wireless sensor networks with multiple mobile base stations," in *Proc. GLOBECOM'03*, vol. 1, December 2003, pp. 377–381.

[2] E. I. Oyman and C. Ersoy, "Multiple sink network design problem in large scale wireless sensor networks," in *Proc. ICC'04*, June 2004, pp. 3663–3667.

[3] L. Eschenauer and V. D. Gligor, "A key-management scheme for distributed sensor networks," in *Proc. CCS'02*. New York, NY, USA: ACM Press, 2002, pp. 41–47.

[4] A. Das and D. Dutta, "Data acquisition in multiple-sink sensor networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, no. 3, pp. 82–85, 2005.

[5] W. Zhang, H. Song, S. Zhu, and G. Cao, "Least privilege and privilege deprivation: towards tolerating mobile sink compromises in wireless sensor networks," in *Proc. MobiHoc'05*. New York, NY, USA: ACM Press, 2005, pp. 378–389.

[6] Y. Wang, B. Ramamurthy, and X. Zou, "KeyRev: An efficient key revocation scheme for wireless sensor networks," in *Proc. ICC'07*, Glasgow, Scotland, U.K., June 2007.

[7] D. Liu, P. Ning, S. Zhu, and S. Jajodia, "Practical broadcast authentication in sensor networks," in *Proc. MobiQuitous '05*, San Diego, CA, USA, July 2005, pp. 118–129.

[8] C. Blundo, A. D. Santis, A. Herzberg, S. Kutten, U. Vaccaro, and M. Yung, "Perfectly-secure key distribution for dynamic conferences," in *Proc. CRYPTO '92*. London, UK: Springer-Verlag, 1993, pp. 471–486.

[9] X. Zou, B. Ramamurthy, and S. S. Magliveras, *Secure Group Communications Over Data Networks*. Springer, 2005.

[10] D. Liu, P. Ning, and K. Sun, "Efficient self-healing group key distribution with revocation capability," in *Proc. CCS'03*. New York, NY, USA: ACM Press, 2003, pp. 231–240.

[11] P. Traynor, H. Choi, G. Cao, S. Zhu, and T. L. Porta, "Establishing pairwise keys in heterogeneous sensor networks," in *Proc. INFOCOM'06*, 2006.

[12] P. Traynor, R. Kumar, H. B. Saad, G. Cao, and T. L. Porta, "LIGER: Implementing efficient hybrid security mechanisms for heterogeneous sensor networks," in *Proc. MobiSys'06*, Uppsala, Sweden, 2006.