Research Article

A Secret Sharing-Based Key Management in Hierarchical Wireless Sensor Network

Yiying Zhang,¹ Chunying Wu,² Jinping Cao,¹ and Xiangzhen Li³

¹ State Grid Information & Telecommunication Company Ltd., Beijing 100761, China
² Computer Science School of Shaanxi Normal University, Xi'an, Shaanxi 710062, China
³ State Grid Electric Power Research Institute, Nanjing, Jiangsu 210003, China

Correspondence should be addressed to Yiying Zhang; zhangyiying1973@hotmail.com

Received 31 January 2013; Revised 13 May 2013; Accepted 13 May 2013

Academic Editor: Muhammad Khurram Khan

Copyright © 2013 Yiying Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Wireless sensor networks (WSNs) are subject to various attacks because of the vulnerable environment, limited recourse, and open communication channel. To protect WSNs, in this paper, we present a Secret sharing-based key management (SSKM). SSKM utilizes the advantages of hierarchical architecture and adopts two-level key management and authentication mechanism, which can efficiently protect the all over network communication security and survivability. Different from previous works, the SSKM distributes keys based on secret sharing mechanism by the clustered architecture, which not only localizes the key things but also keeps scalability. The SSKM provides various session keys, the network key for base station (BS) and cluster heads (CHs); the cluster key between the cluster head and member nodes. The SSKM dynamically generates different keys based on different polynomials from BS in different periods which can protect the network from the compromised nodes and reduce the high probability of the common keys. The security analysis shows that the SSKM can prevent several attacks effectively and reduce the energy consumption.

1. Introduction

Due to the development of internet of things (IoT) and cyber physical system (CPS), wireless sensors have been deployed in many applications, such as in smart grid, national security, intelligent transportation, forest detection, or chemical harmful gas monitoring [1]. However, wireless sensor networks (WSNs) usually consist of tiny sensors which have low computational capability, small storage, and limited energy; that is, the WSNs are often subject to a variety of attacks, such as eavesdropping attack and flood attack and so on. Once a sensor is compromised by adversaries, the information materials of the sensor become non secretive and intercepted by enemy, and the entire network is threatened [2].

Therefore, security mechanisms in WSN are required to provide data confidentiality, integrity, freshness, availability, and authentication [2]. Moreover, in view of the excellent performance of the clustering algorithm in WSN, the hierarchical architectures are often used in WSN applications [3]. Normally, cryptographic methods of securing a network are the key management strategy, and it has been intensively studied in the literature of WSNs [4–13]. Therefore, some literatures adopt hierarchical architecture to deploy key system on them to protect the communication in WSNs [4–10].

In [5, 6], the authors employed the secret sharing mechanism to distribute keys into nodes, which can effectively generate and assign keys. However, in these schemes, the network must exchange many messages to establish key system, which consumes lots of energy. In this paper, we present a novel secret sharing-based key management (SSKM).

In SSKM, considering that the energy efficiency is a dominant consideration problem of WSNs, we firstly employ the maximum energy cluster head (MECH) protocol to form cluster. Different from other hierarchical architectures, MECH protocol limits the size of cluster to generate uniform cluster. In each cluster, there is a sensor, called cluster head (CH), collecting information from other cluster member nodes and forwarding the processed information to the base station.

Therefore, to protect the communication channel from CH to BS, we present a network key. Firstly, the BS encrypts
the network key with a secret, puts the secret as constant of
the polynomial, and divides the secret into shares based on
Lagrange interpolation formula. To reconstruct the poly-
nomial of \((t - 1)\) degree, any \(t\) or more parameters \((ID, f(ID))\)
combination can recover and obtain the secret. Therefore, our
solution tries to avoid the adversary intercept and capture
sufficient parameters.

Also, similar to network key, we design a cluster key
to protect the communication between the cluster head and
member nodes. Unlike the BS, CHs have no sufficient
energy to broadcast messages. Thus, the BS deploys key
material to sensors in advance, such as polynomials and
revoked list. Then, the CH just exchanges parameters to adjust
polynomials to generate/cancel keys.

Compared to previous works [5, 6], the salient advantages
of our work are as follows:

1. SSKM establishes a relocatable key mechanism based
on the secret sharing theory, which hides keys into
secret and recovers them when needed;
2. SSKM adopts hierarchical architecture which is suit-
able for the secret sharing mechanism and localizes
the security and reduces energy consumption. It
makes the SSKM key management feasible;
3. SSKM presents an authentication mechanism based
on the secret sharing theory, which supports the
scalability (join or leave).

The rest of this paper is organized as follows. Section 2
describes the related work, Section 3 presents the system
model and assumption, Section 4 describes the secret sharing
key management in detail, and Section 5 evaluates SSHM
using security analysis. Finally, we end the paper with a
conclusion as well as the further work in Section 6.

2. Related Work

In 1979, Shamir [14] and Blakley [15] proposed the secret
sharing method based on the Lagrange interpolation formula,
and the nature of the vector space, respectively. Proposition:
given \(n + 1\) points \((x_0, f(x_0))\) on a polynomial \(f(x)\) of degree
\(n\), one can identify a uniquely polynomial by calculating:

\[
f(x) = \sum_{i=1}^{n} \left( f(x_i) \prod_{1 \leq k \neq i \leq n} \frac{x - x_k}{x_i - x_k} \right). \tag{1}
\]

One also defines the Lagrange coefficient \(\Delta_{i,s}\) for \(i \in Z_p\)
and a set \(S \subseteq Z_p\):

\[
\Delta_{i,s}(x) = \prod_{s \in S \neq i} \frac{x - j}{i - j}. \tag{2}
\]

A \((t, n)\) threshold secret sharing scheme is as follows:
given \(n\) points \((x_i, f(x_i))\) on a polynomial \(f(x)\) of degree
\((t - 1)\), randomly picking \(t\) points of \((x_i, f(x_i))\) from
\(n\) points can construct the polynomial. When constructing
the scheme, the credibility of the parties splits initial secret \(S\)
into \(n\) shares (points) and assigns them to users safely. Any
\(t\) or more users combining their share can reconstruct the
secret \(S\), but any \(t - 1\) user group or less cannot reconstruct
the secret. This secret sharing method provides the security
scheme in many applications, such as key distribution, secure
computation, and information safe storage.

In [5], the authors present a low-cost secret-sharing
scheme for sensor network. This paper provides basic
building blocks to establish secure communication through
exchanging secret keys between neighbor nodes without
any cryptography methods. In [5], authors also design a
second algorithm which extends the secret key establishment.
However, due to the exchange happening among sensors,
it consumes lots of energy. Moreover, the authentication
between neighbor nodes also needs to exchange large mes-
gages, which makes it unsuitable for wireless sensor network.

In [6], authors presented some schemes to secure data
aggregation based on secret sharing and information dis-
peral. In these schemes, sensor nodes split messages into
subshares and forward them among several disjoint paths
to defend DoS attack, eavesdropping attack, and tampering
attack. They design a secret multipath aggregation (SMA)
mechanism which applies secret sharing to create shares to
deal with security under the contingency of node compro-
mise. However, these schemes are not feasible for heavy
energy consumption. On one hand, they want data aggre-
gation using secret sharing; on the other hand, they have to
distribute key things and messages to confuse the enemy, so
that the adversary cannot find the real route, which needs a
large number of messages exchange.

Comparing with previous works, our solution adopts
the hierarchical network and localizes the communication
and security. Also, we ingeniously use the base station to
carry complicated things out, which can reduce the energy
consumption.

3. System Model

3.1. Network Model. The wireless sensor network is energy
sensitive. Therefore, we adopt the maximum energy cluster
head (MECH) protocol for our network architecture [1]. The
MECH is an LEACH-like protocol (LEACH: low energy
adaptive clustering hierarchy) [2] which divides the network
into clusters.

As shown in Figure 1, in the MECH architecture, the
sensors self-organize into some clusters and act as two types
of roles: cluster heads and member nodes. In each cluster,
one node as a CH manages the cluster and deals with information
from member nodes forward to the base station (BS). MECH
constructs clusters based on radio range and the number of
cluster members. The cluster topology in the network is
distributed more equally through our cluster constructing;
that is, nodes in each cluster do not exceed a certain threshold.

3.2. Assumptions. In the considered network, we consumed
the following.

(i) All sensor nodes are static.
(ii) Each sensor has a unique ID assigned by the base
station.
4. The Secret Sharing-Based Key Management

In this section, we describe the secret sharing-based key management (SSKM) in detail. After deployment, the base station assigns each sensor an initial key $K_{init}$ similar to LEAP+ [12]. And then, the BS broadcasts the key materials to the network to build the network key and the cluster key, respectively. The key architecture is shown in Figure 2.

4.1. Preliminaries. After deployment, the BS randomly chooses an integer, relatively primes with $p - 1$ and $q - 1$ ($p$ and $q$ are big primes); the base station assigns each sensor an initial key $K_{init}$ similar to LEAP. Since the BS is credible and has sufficient capacity and energy, it keeps all IDs and keys. Meanwhile, BS chooses polynomials and the corresponding number of polynomial values for each cluster. Assume that there are $m - 1$ clusters $C_i$ ($i = 1, \ldots, m - 1$), and each cluster has a cluster head $CH_i$ and $k$ ($k \geq t$) member nodes.

Shamir’s $(t, n)$ threshold scheme based on Lagrange interpolating polynomial states that there are $n$ shareholders $U = \{U_1, \ldots, U_n\}$ and mutually trusted dealer $D$. The scheme consists of two steps [16]:

(1) share generation phase: dealer $D$ randomly selects a polynomial $f(x)$ of $(t - 1)$ degree:

$$f(x) = S + a_1x + \cdots + a_{t-1}x^{t-1} \pmod{P}, \tag{3}$$

in which the secret $S = f(0)$, all coefficients $S, a_1, a_2, \ldots, a_{t-1}$ are in finite field $F_p = GF(P)$ with $p$ elements, and dealer computes all shares $S_i = f(x_i)$ for $i = 1, \ldots, n$; then it distributes each share $S_i$ to corresponding shareholder $U_i$ privately;

(2) secret reconstruct phase: any $t$ shares $(S_1, S_2, \ldots, S_t)$ of $n$ shares as input, and we can reconstruct the secret $S$ as

$$f(x) = \sum_{i=1}^{t} S_i \left( \prod_{j \neq i} \frac{x - x_j}{x_i - x_j} \right) \pmod{P}, \tag{4}$$

$$f(0) = \sum_{i=1}^{t} S_i \left( \prod_{j \neq i} \frac{x_j}{x_i - x_j} \right) \pmod{P}.$$

We find that the above scheme satisfies the basic security requirement of secret sharing scheme: any $t$ shares or more than $t$ shares can reconstruct the secret $S$; fewer than $(t - 1)$ shares cannot reconstruct the secret $S$. Shamir’s scheme is information theoretically secured [17]. However, there are some requirements [18] in this situation:

(iii) Each sensor has the same capabilities in energy, computation, radio range, and so forth.

(iv) If a node is compromised, all of the key things in the node are revealed [7].

(v) Each sensor is in, and only in, one cluster.

(vi) The BS can communicate with all sensors in the network.

3.3. Notations. In Table 1, we list some notations used in this paper.

| Notation | Explains |
|----------|----------|
| $K_{init}$ | The initial key shared by all nodes |
| ID | ID of sensor |
| BS | Base Station |
| $C_i$ | The $i$th cluster |
| $CH_i$ | The $i$th cluster head |
| $S_{Ch_i}^l$ | The secret sharing between CH and members in session $l$ |
| $K_{Ch_i}$ | The secret sharing between CH and $i$th cluster head |
| $S_{Ch_i}$ | The secret sharing between CH and members in session $l$ |
| $K_{Ch_i}$ | The secret key among clusters during session $l$ |
| $R$ | Revoked set before session $l$ |
| $l$ | Session period |
| $w$ | The number of nodes in WSN |

Figure 1: The network system.

Figure 2: The key architecture.
There must be a secure channel for delivering shares between dealer and users;

(2) \( x_i \) and \( f(x_i) \) are made publicly known. However, in key transfer protocol, for security reason, we need to keep \( x_i \) and \( f(x_i) \) as each user’s secret. So we adopt the discrete logarithm in finite field and DDH difficulty assumption to ensure the security in the unsecure communication channel.

4.2. Initial Phase. Once wireless sensor network has been deployed and sensors self-organized into clusters, BS starts to form the key system as follows.

(1) Firstly, BS chooses two big primes \( p_i \) and \( q_i \); let \( p = 2p_i + 1 \) and \( q = 2q_i + 1 \), \( N = pq \); it is computationally intractable to solve the factor \( N \) without \( p, q \). Meanwhile, BS selects a generator \( g \) (\( g \in [N^{1/2}, N] \)) and another prime \( Q \) \( (Q > N) \). And then, BS broadcasts the three triple \((N, g, Q)\) to sensors in the network.

(2) Assume that during each session period \( l \) \((l = 1, \ldots, M)\), BS randomly and uniformly chooses \( m \) polynomials \( f(x) \) of \((t - 1)\)-degree, where \( m - 1 \geq t \). And one of polynomials is as follows:

\[
 f_{C_n}^l(x) = S_{C_n}^l + d_{1,C_n}^l x + \ldots + d_{t-1,C_n}^l x^{t-1} \mod Q. \tag{5} 
\]

Equation (5) is used to key distribute between BS and cluster head. And other \( m - 1 \) polynomials are utilized to key distribute among the cluster and member nodes as follows:

\[
 f_{CH_i}(x) = S_{CH_i}^l + d_{1,CH_i}^l x + \ldots + d_{t-1,CH_i}^l x^{t-1} \mod Q. \tag{6} 
\]

(3) BS independently selects \( M \) session keys \( \{K_{C_n}^l\}_{l=1}^M \) and \( \{K_{CH_i}^l\}_{i=1}^M \) from \( \text{GF}(Q) \) in the finite field \( Q \) and hides the session keys with secret \( S_{C_n}^l \) and \( S_{CH_i}^l \), namely, \( Z_{C_n}^l = \{K_{C_n}^l + S_{C_n}^l\} \) in network key management and \( Z_{CH_i}^l = \{K_{CH_i}^l + S_{CH_i}^l\} \) in cluster key management. The algorithm of initial phase is shown in Figure 3.

4.3. Network Key Management. The network key is the session key between the BS and cluster heads to protect their communication. The key shares distribution process is as follow.

(1) During the session period \( |l| \in [1, \ldots, M], \) the BS counts out each cluster head’s (CH’s) share \( f_{C_n}^l(\text{ID}_{CH_i}^l) \) by their ID and gets the two tuples \( \{\text{ID}_{CH_i}^l, f_{C_n}^l(\text{ID}_{CH_i}^l)\} \), for the security between BS and CHs. Firstly, the BS randomly chooses \( x_i^l \) (\( x_i^l \in [2, n] \)) which relatively primes with \( p - 1 \) and \( q - 1 \), and let \( y_0^l = g^{x_i^l} \mod N \); meanwhile, CHs also chooses \( x_i^l \) (\( x_i^l \in [2, N] \)) and lets \( y_i^l = g^{x_i^l} \mod N \). And then, the CH sends \( (\text{ID}_{CH_i}^l, y_i^l) \) \((i = 1, \ldots, m - 1)\) to the BS, and BS unicasts \( (y_0^l, (\text{ID}_{CH_i}^l, f_{C_n}^l(\text{ID}_{CH_i}^l))(y_i^l)^{x_i^l} \mod N) \) to each ID_{CH_i}. Note that the BS must ensure if \( \text{ID}_i \neq \text{ID}_j \) then \( y_i^l \neq y_j^l \); otherwise, it should regenerate until success.

(2) Given that \( R \) indicates the set of revoked CHs during the session period \( l \) and before, let \( R = R_2 \cup \cdots \cup R_{l-1} \cup R_l \) where \(|R| \leq t \). In session \( l \), the BS selects a group of users \( V_l = \{\text{ID}_{CH_i}^l, \ldots, \text{ID}_{CH_i}^l\} \) which meet \( ID_R \subseteq V_l \) and \( \text{ID}_{CH_i} \cap V_l = \emptyset \).

(3) BS broadcasts the information \( \{Z_{C_n}^l, Z_{C_n}^l, \ldots, Z_{C_n}^l\} \) to each cluster node.

The network key process is shown in Figure 4.

The session key recovery process is as follows.

(1) Having received the key materials, cluster heads calculate their individual share \( f_{C_n}^l(\text{ID}_{CH_i}^l) \cdot (y_0^l)^{x_l^l} / (y_i^l)^{x_l^l} = f_{C_n}^l(\text{ID}_{CH_i}^l) \) with the private key \( x_i^l \) and public key \( y_0^l \). According to the information \( \{\text{ID}_{CH_i}^l, f_{C_n}^l(\text{ID}_{CH_i}^l)\} \), any \( t \) sensors or more than \( t \) sensor can recover the secret \( S_{C_n}^l \) with (7) as follows:

\[
 f_{C_n}^l(x) = \sum_{j=1}^{t} \left( \prod_{l \neq j} \frac{x - \text{ID}_{CH_i}^l}{x - \text{ID}_{CH_i}^l} \right) f \left( \text{ID}_{CH_i}^l \right) \mod Q, 
\]

\[
 S_{C_n}^l = \sum_{j=1}^{t} \left( \prod_{l \neq j} \frac{\text{ID}_{CH_i}^l}{\text{ID}_{CH_i}^l - \text{ID}_{CH_i}^l} \right) f \left( \text{ID}_{CH_i}^l \right) \mod Q. 
\]

(7)
Figure 4: The network in session $l$. 

(2) Using $Z_{C_{in}}^l$ and $S_{C_{in}}^l$ users can get the secret $K^l = Z_{C_{in}}^l - S_{C_{in}}^l$.

4.4. Cluster Key Management. In this phase, the protocol establishes the cluster key between CH and members. Similar to the network key, the cluster key can be generated as follows.

(1) Firstly, cluster head chooses $x_{ch}^l$ ($x_{ch}^l \in [2, N]$) randomly which relatively primes with $p - 1$ and $q - 1$, and CH sends it to BS. Then, BS counts out $y_{ch}^l = g^{x_{ch}^l}$ and sends $(ID_{CH}^l, y_{ch}^l)$ to sensor node in cluster $CH_l$; meanwhile, sensor node picks $x_{I}^l$ randomly which relatively primes with $p - 1$ and $q - 1$, computes $y_{I}^l = g^{x_{I}^l}$ mod $N$, and then sends $(ID_{I}^l, y_{I}^l)$ ($i = 1, \ldots, m - 1, r = 1, \ldots, k$) to the BS. The BS ensures that if $ID_{I}^l \neq ID_{CH}^l$, there should be no $y_{I}^l = y_{ch}^l$; otherwise it reselects until success.

Furthermore, the BS utilizes CH's ID$_{CH}^l$ and members' ID$_{I}^l$ ($r = 1, \ldots, k$) to count out the share $f_{CH}^l(ID_{CH}^l)$ and $f_{CH}^l(ID_{I}^l)$, respectively.

(2) Given that $R$ indicates the set of revoked sensors during the session period $l$ and before, let $R = R_1 \cup \cdots \cup R_{l-1} \cup R_l$, where $|R| \leq t$. During session $l$, the CH selects a group of users $V_l = \{ID_{1}^l, \ldots, ID_{k}^l\}$, where $ID_R \subseteq V_l$ and $ID_R \cap V_l = \emptyset$, while BS unicasts $(ID_{CH}^l, f_{CH}^l(ID_{CH}^l) \cdot (y_{I}^l)^{x_{I}^l} \mod N)$ to sensor node in $CH_l$ and sends $(ID_{I}^l, f_{CH}^l(ID_{I}^l) \cdot (y_{I}^l)^{x_{I}^l} \mod N)$ to $CH_l$.

(3) Then, BS independently selects $M$ session keys $\{K_i\}_{i = 1}^{M}$ from GF($Q$), in finite field $Q$, hiding the $K^l$: $Z_{CH}^l = [K_i + S_{CH}^l]_{i = 1}^{M}$, with the secret $S_{CH}^l$. The purpose is not to leak the $K^l$. The cluster key process is shown in Figure 5.

4.5. Secret Recovery. Depending on the received information from base station, public generator, node's private key $x_{ch}^l$, and cluster node's own key $x_{ch}^l$, cluster head and members can obtain their share through the following formulas:

$$
\frac{f_{CH}^l(ID_{CH}^l) \cdot (y_{I}^l)^{x_{I}^l}}{(y_{ch}^l)^{x_{ch}^l} \mod N} = f_{CH}^l(ID_{CH}^l),
$$

(8)

$$
\frac{f_{CH}^l(ID_{I}^l) \cdot (y_{I}^l)^{x_{I}^l}}{(y_{ch}^l)^{x_{ch}^l} \mod N} = f_{CH}^l(ID_{I}^l).$$

$\frac{f_{CH}^l(ID_{CH}^l) \cdot (y_{I}^l)^{x_{I}^l}}{(y_{ch}^l)^{x_{ch}^l} \mod N}$, $\frac{f_{CH}^l(ID_{I}^l) \cdot (y_{I}^l)^{x_{I}^l}}{(y_{ch}^l)^{x_{ch}^l} \mod N}$ is broadcast information; $y_{I}^l$ and $y_{ch}^l$ are public information. So cluster head and common nodes can obtain their own shares, respectively, and then members send $(ID_{I}^r, f_{CH}^l(ID_{I}^l), f_{CH}^l(ID_{I}^l))$ to CH. CH uses $(ID_{CH}^l, f_{CH}^l(ID_{CH}^l))$ and $r - 1$ sensors' $(ID_{I}^l, f_{CH}^l(ID_{I}^l))$ to recover the secret $S$. According to (7), we can carry $S_{CH}^l$ out. Furthermore, CH can get $Z_{CH}^l = K^l + S_{CH}^l$, and then we can calculate $K^l$ to unicast $K^l$ to $r - 1$ sensors $(ID_{I}^l, K^l)$.

4.6. Scalability. In our solution, we also consider the scalability of network.

4.6.1. New Member Join. When a new member $P_u$ ($u \neq 1, \ldots, n$) wants to join during session period $l$, $P_u$ should randomly choose an integer $x_u$ ($x_u \in [2, N]$) and count out $y_u = g^{x_u} \mod N$. And then, $P_u$ keeps $x_u$ secretly and chooses randomly an ID$_u$ (ID$_u > n$); $n$ is the largest node identity in network, and then sends $(ID_u, y_u)$ to the BS. The BS will authenticate $P_u$. If ID$_u > N$ and $P_i$ ($i = 1, \ldots, N$) and $y_i \neq y_u$, then $P_u$ is acceptable and can join the network.

4.6.2. Node Isolation. Once CH or neighbor nodes find a compromised node $P_u$, the CH sends its information ID$_u$ to BS. Meanwhile, BS and CH add their IDs into $R : \{R\} \cup \{ID_u\}$.
5. Security Analyses

Due to the unreliable wireless environment, dynamic clustering cluster key distribution scheme is subject to a variety of attacks, such as eavesdropping, tampering, and replay attacks. Compared to previous works, the salient advantage of our solution is that we addressed challenging runtime security issues using localizing key things and group key management based on secret sharing mechanism.

5.1. Robustness. In the recovery phase, for any user \( P_i \in P \), if anyone wants to recover \( K_l \), they must obtain both \( Z^l_{\text{CH}} \) and \( S^l_{\text{CH}} \), which makes it very difficult to recover keys.

Furthermore, assume that any set \( F \subseteq P \) and \(|F| \leq t - 1 \), if an unrevoked user \( P_i \in F \), any other user collusions in \( F \) cannot get information about the \( P_i \)'s personal secret \( S_i \). Because in each session \( l \), user \( P_i \)'s secret \( S^l_i = f^l_{\text{CH}}(0) \) or \( S^l_i = f^l_{\text{CM}}(0) \) is a 1-degree polynomial \( f^l_{\text{CH}}(x) \) or \( f^l_{\text{CM}}(x) \), the users in \( F \) only know \( t - 1 \) values about \( f^l_{\text{CH}}(x) \) or \( f^l_{\text{CM}}(x) \). And the difficulty to reconstruct a polynomial \( f^l_{\text{CH}}(x) \) or \( f^l_{\text{CM}}(x) \) by \( t - 1 \) values is equivalent to breaking a Shamir's \((t,n)\) secret sharing problem, which is not feasible in computation. Therefore, user collusions in \( F \) have no ability to obtain user \( P_i \)'s secret \( S^l_i \).

Moreover, because the cluster session key \( K_l \) is selected from a uniform distribution, and independent of the user's personal secret, no one can obtain information about the session key \( K_l \) separated from personal secret collection. Also, in each session \( l = 1, \ldots, M \), because of \( Z^l_{\text{CH}} = K_l + S^l_{\text{CH}} \), \( Z^l_{\text{CH}} \) hides session key \( K_l \) with personal secret \( S^l_{\text{CH}} \), and adversary has no ability to obtain any useful information just from the collection of broadcast messages.

5.2. Tolerance. The normal user \( P_i \notin R \) (unrevoked user or normal node) can utilize the broadcast messages and private secret to recover the session key \( K_l \); however, the revoked user can only obtain \( t - 1 \) values from the broadcasted polynomial \( f^l(x) \); thus, they have no ability to reconstruct \( t - 1 \) degree polynomial \( f^l(x) \) as mentioned above. Therefore, the user in \( R \) cannot get \( f^l(0) \). Moreover, because of \( K_l = Z^l_{\text{CH}} - f^l(0) \), it is not feasible to recover personal secret \( K_l \) by \( Z^l_{\text{CH}} \) and \( S_{\{P_i \in R\}} \).

5.3. Security. Our solution also has both \( t - 1 \) forward secrecy and \( t - 1 \) backward secrecy.

5.3.1. \( t-1 \) Forward Secrecy. Let \( R \subseteq P \), \(|R| \leq t - 1 \), and each \( P_i \in R \) is a revoked user before session \( l \). Even if user collusions know all cluster keys before the session \( l \), they cannot obtain any information of current session key \( K_l \), because they cannot recover \( f^l(0) \) with just \( t - 1 \) values of \( f^l(x) \). Therefore, the solution is \( t - 1 \) forward secrecy.

5.3.2. \( t-1 \) Backward Secrecy. Let \( J \subseteq P \), \(|J| \leq t - 1 \). Each user \( P_i \in J \) joined the group before session \( l \). Even if user collusions in \( J \) know all cluster keys \( K_l \) before the session \( l \), they cannot obtain any information of current session key \( K_l \), \((l_i < l)\). Because if a user wants to get \( K_l \), the user \( P_i \in J \) at least recovers \( t \) points of \( f^l(x) \) for \( f^l(0) \). However, each user \( P_i \) after session \( J \) at least obtains \( t - 1 \) value from \( t - 1 \) degree \( f^l(x) \) and has no ability to reconstruct \( f^l(x) \); that is, the solution is \( t - 1 \) backward secrecy.

5.4. Complexity Analysis. In this section, we discuss the complexity of our scheme from computation complexity, communication complexity, and storage cost needed by common node and cluster.

1. Computation complexity: we assume that base station has a large computation capacity, the pickup of polynomial and share distribution as well as the choice of generator. Common node only needs to compute division, and cluster head needs to reconstruct the polynomial beside division.

2. Communication costs include the broadcast cost: \( j(\log^3 q) \) and download the publishing information from publish board: \( t \ast j \ast (\log^3 q) \), where \( t \) indicates a session period, \( j \) is the number of nodes in one cluster, and \( q \) is an enough secure prime.

3. Storage cost: in our scheme, we only need to save a private respective key, which is \( \log^3 q \).

6. Conclusion and Future Work

In this paper, we propose a secret sharing-based key management scheme (SSKM) to enhance network security and survivability. Different from previous works, although we employ the hierarchical architecture, we limited the size of clusters to balance the overall energy consumption of the network. In contrast to other clustered architectural security solutions, the salient advantage of this work is that we addressed challenging security issues by localizing key things based on secret sharing theory. We present the network key and cluster key and generate new keys from various polynomials by Lagrange interpolation formula. Also, we present a rekey mechanism in the cluster head selection with low energy consumption. Meanwhile, SSKM has an authentication mechanism to ensure the scalability, which cannot only authenticate the new sensor but also can isolate the compromised node. The security analysis shows that our solution cannot only reduce the energy consumption effectively but also enhance the security level. In the future, we will focus on how to enhance security in mobile and scalable WSNs.

Acknowledgments

This work was supported by China Postdoctoral Science Foundation Funded Project (2012M510367); the National Basic Research Program of China (973 Program) (2011CB302900).
References

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "A survey on sensor networks," IEEE Communications Magazine, vol. 40, no. 8, pp. 102–114, 2002.

[2] Y. Zhou, Y. Fang, and Y. Zhang, "Securing wireless sensor networks: a survey," IEEE Communications Surveys and Tutorials, vol. 10, no. 3, pp. 6–28, 2008.

[3] K. Vivek, C. Narottam, and S. Surender, "A survey on clustering algorithms for heterogeneous wireless sensor networks," International Journal of Advanced Networking and Applications, vol. 2, no. 4, pp. 745–754, 2011.

[4] X. Du, Y. Xiao, M. Guizani, and H. H. Chen, "An effective key management scheme for heterogeneous sensor networks," Ad Hoc Networks, vol. 5, no. 1, pp. 24–34, 2007.

[5] M. Bertier, A. Mostefaoui, and G. Trédan, "Low-cost secret-sharing in sensor networks," in Proceedings of the IEEE 12th International Symposium on High Assurance Systems Engineering (HASE '10), pp. 1–9, November 2010.

[6] T. Claveirelo, M. Dias De Amorim, M. Abdalla, and Y. Viniotis, "Securing wireless sensor networks against aggregator compromises," IEEE Communications Magazine, vol. 46, no. 4, pp. 134–141, 2008.

[7] H. N. Seyed, H. J. Amir, and D. Vaneasa, "A distributed group rekeying scheme for wireless sensor networks," in Proceedings of the 6th International Conference on Systems and Networks Communications (ICSN C '11), pp. 127–135, 2011.

[8] Y. Y. Zhang, X. Z. Li, J. M. Liu, J. C. Yang, and B. J. Cui, "A secure hierarchical key management scheme in wireless sensor network," The International Journal of Distributed Sensor Networks, vol. 2012, Article ID 547471, 8 pages, 2012.

[9] Y. Y. Zhang, X. Z. Li, J. C. Yang, Y. A. Liu, N. X. Xiong, and A. V. Vasilakos, "A real-time dynamic key management for hierarchical wireless multimedia sensor network," Multimedia Tools and Applications, 2012.

[10] Y. Y. Zhang, W. C. Yang, K. B. Kim, and M. S. Park, "An AVL tree-based dynamic key management in hierarchical wireless sensor network," in Proceedings of the 4th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IH-MSP '08), pp. 298–303, August 2008.

[11] M. Luk, G. Mezzour, A. Perrig, and V. Gligor, "MiniSec: a secure sensor network communication architecture," in Proceedings of the 6th International Symposium on Information Processing in Sensor Networks (IPSN '07), pp. 479–488, April 2007.

[12] S. Zhu, S. Setia, and S. Jajodia, "LEAP+: efficient security mechanisms for large-scale distributed sensor networks," ACM Transactions on Sensor Networks, vol. 2, no. 4, pp. 500–528, 2006.

[13] M. Eltoweissy, M. Moharrum, and R. Mukkamala, "Dynamic key management in sensor networks," IEEE Communications Magazine, vol. 44, no. 4, pp. 122–130, 2006.

[14] A. Shamir, "How to share a secret," Communications of the ACM, vol. 22, no. 11, pp. 612–613, 1979.

[15] G. Blakley, "Safeguarding cryptographic keys," in Proceedings of the National Computer Conference (AFIPS '79), pp. 313–317, AFIPS Press, New York, NY, USA, 1979.

[16] S. Agrawal, "Verifiable secret sharing in a total of three rounds," Information Processing Letters, vol. 112, pp. 856–859, 2012.

[17] C. Hua, X. Liao, and X. Cheng, "Verifiable multi-secret sharing based on LFSR sequences," Theoretical Computer Science, vol. 445, pp. 52–62, 2012.
Submit your manuscripts at
http://www.hindawi.com