J. Cent. South Univ. (2013) 20: 1277−1284
DOI: 10.1007/s11771-013-1612-x

Key-insulated encryption based group key management for wireless sensor network

QIU Wei-dong(邱卫东)[1], ZHOU Yao-wei(周耀伟)[1], ZHU Bo(朱博)[2], ZHENG Yan-fei(郑燕飞)[1], GONG Zheng(龚征)[3]

1. School of Information Security Engineering, Shanghai Jiao Tong University, Shanghai 200240, China;
2. Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China;
3. School of Computer Science, South China Normal University, Guangzhou 510631, China

Abstract: The key exposure problem is a practical threat for many security applications. In wireless sensor networks (WSNs), keys could be compromised easily due to its limited hardware protections. A secure group key management scheme is responsible for secure distributing group keys among valid nodes of the group. Based on the key-insulated encryption (KIE), we propose a group key management scheme (KIE-GKMS), which integrates the pair-wise key pre-distribution for WSN. The KIE-GKMS scheme updates group keys dynamically when adding or removing nodes. Moreover, the security analysis proves that the KIE-GKMS scheme not only obtains the semantic security, but also provides the forward and backward security. Finally, the theoretical analysis shows that the KIE-GKMS scheme has constant performance on both communication and storage costs in sensor nodes.

Key words: wireless sensor network; data encryption; group key management; forward security; key-insulated encryption

1 Introduction

Wireless sensor network (WSN) has attracted widely attentions due to its promising applications, such as military, environmental monitoring and health-care industry [1−2]. When sensor networks are deployed in a hostile environment, security becomes extremely important to resist different types of malicious activities. In WSNs, an adversary can easily sniffer the traffic, impersonate nodes of the network, or intentionally mislead nodes. A secure WSN communication should provide information confidentiality, integrity and authenticity. How to build a secure communication among sensor nodes, via setup secret keys, becomes a hot topic in the WSN-security community.

The above problem is known as the key management problem [3], which has been widely studied in general network environments. Since sensor nodes are resource constraint, they need an efficient scheme that helps reducing communication and computational overheads. One widely-accepted method to solve this problem is the pair-wise key pre-distribution scheme [2−6]. However, there are two special scenarios in WSN which need group key to protect the information. Firstly, one shares security information with more than one node in the group. The communication costs will be high if encrypting the messages with different pair-wise keys. Secondly, when encrypted message transmits in multi-hops chain, the message has to be encrypted and decrypted from node to node in the chain. Such operations are costly on computation and power resource. However, if there is a group key, the message can be encrypted once, while decrypted by all group members. Therefore, group communication becomes one of vital applications of WSN and helps decreasing the communication overhead [7].

Recently, many publications have been proposed on the problem of secure group key management in WSN. ZHANG et al [8] have presented a cluster-based group key management scheme. It reduces the communication overhead and storage cost of sensor node. However, there is no authentication for sensor node joining in their scheme. ZHANG et al [9] proposed B-PCGR and C-PCGR schemes which update the compromised group keys to prevent the compromised nodes from understanding the communications between non-compromised nodes or injecting false data. KHALID and HUSSAIN [10] have proposed a group secure re-keying scheme with compromised nodes revocation in WSN. In

Foundation item: Project(61100201) supported by National Natural Science Foundation of China; Project(12ZZ019) supported by Technology Innovation Research Program, Shang Municipal Education Commission, China; Project(LYM11053) supported by the Foundation for Distinguished Young Talents in Higher Education of Guangdong Province, China; Project(NCET-12-0358) supported by New Century Excellent Talents in University, Ministry of Education, China
Received date: 2012−02−28; **Accepted date**: 2012−05−06

Corresponding author: GONG Zheng, PhD, Associate Professor; Tel: +86−20−85211353−405; E-mail: cis.gong@gmail.com

their scheme, revocations of compromised group nodes do not rely on re-grouping or re-initialization.

With the improvement of hardware, public-key cryptography has been involved in some approaches of WSN security. GONG et al [11] introduced a scheme in which public key cryptography is used to establish a secure link between sensor nodes and gateway. The public key is used to build a session key. AMIN et al [12] also analyzed public-key cryptography for wireless sensor networks. They evaluated time and power consumption of public key cryptography algorithm for signature and key management by simulation. Based on ECC, JIANG [13] introduced ecliptic curve cryptosystem and identity-based authentication mechanism into WSN.

In Eurocrypt'02, DODIS et al [14] initially introduced the key-insulation encryption (KIE) to deal with the key exposure problem of public key cryptography system. Generally, in the key-insulated system, the lifetime of the system is broken into discrete periods 1, ⋯, *N*. While the private key is divided into two parts: a temporary private key, held by the user on a powerful but insecure device (e.g. a mobile device), and a helper key, stored in a physically secure but computationally limited device named "helper". The public key remains unchanged throughout the lifetime of the system, while the temporary private key will be updated at every period via the interactions between the user and the helper. Decryption operations in a given period only involve the corresponding temporary private key without further access to the helper, so that the exposure of up to *t* of the *N* periods, chosen adaptively by the adversary, still keeps any unexposed period secure. The optimal number of *t* achieved by some of the schemes is *N*−1 where the remaining period is secure. Even if the helper key is compromised, the security is still ensured as long as none of the temporary key is exposed. As a result, the damage caused by key exposure is minimized.

Based on KIE, QIU et al [15] proposed a new pair-wise key pre-distribution scheme for WSN that shows a higher security and constant costs on both of the storage and communication overheads. In their scheme, a sensor's ID is used as the KIE time information. Under QIU et al's scheme, all nodes share a public key and each owns a private key related to its ID. It supposes that Node A wants to set up a secret key with Node B, whose ID is ID_{B} and the private key is sk_{B}, Node A chooses a random key *K*, encrypting it with Node B's ID and public key under the function *C*=*Enc*(*pk*, ID_{B}, *K*). When receiving the cipher text C, Node B decrypts it with its private key under the function *Dec*(*C*, sk_{B}) and gets the key *K*. Only Node B has the decryption key related to the ID_{B}, and other nodes can't decrypt the cipher-text to get the secret key *K*.

In this work, we go further to introduce the key-insulated encryption scheme into the WSN secure group key management, which is called KIE-GKMS. There are three type keys in the scheme: group public key (GPK), group private key (GSK), and group key (GK). All nodes in the same group will share the GPK and GSK, which are used to establish the GK. Each group member also stores the updating authentication information (USK) for updating the corresponding group private key GSK. In our scheme, the group header just needs to broadcast the GK, which is encrypted by the GPK. After receiving the encrypted message, all the group members can obtain the GK by decrypting it with the GSK. If a new node joins the group, the GK will be updated for providing the backward security. If one node has been compromised, both the GSK and GK need to be updated for providing the forward security. A group rekeying protocol will be presented to update the GSK. First, the group header generates the new GSK, computes and unicasts the key updating help information (HSK) to the group members. Second, the group members compute the new GSK by combining the USK and HSK. Thinking that the sensor is power and resource constraint, this work combines the pair-wise and group key management together to improve the key management efficiency.

2 KIE-WSN model

Based on the key-insulated encryption, QIU et al [15] have proposed a KIE-WSN model, which is used to construct the KIE-WSN key pre-distribution scheme for the wireless sensor network.

2.1 Bilinear map

Let Z_q denote the set {0, 1, 2, ⋯, *q*−1} and Z^*_q denote $Z_q\backslash\{0\}$. For a finite set *S*, $x \xleftarrow{R} S$ denotes that one randomly chooses an element *x* from *S*.

Negligible: A function *f*: *R*→*R* is negligible if for any *d*>0 there exists *n*, (*n*>0), when *k*>*n*, we have $|f(k)|<1/k^d$.

Let G_1 be an additive group and G_2 be a cyclic multiplicative groups with the same prime order *q*. An admissible bilinear map is a map $\hat{e}: G_1 \times G_1 \rightarrow G_2$ with the following properties:

1) Bilinear: $\forall P, Q \in G_1, \forall a, b \in Z_q^*$, we have $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$;

2) Non-degeneracy: There is $P, Q \in G_1$, such that $\hat{e}(P, Q) \neq 1$;

3) Computability: There is an efficient polynomial-time algorithm to compute $\hat{e}(P, Q)$ for $\forall P, Q \in G_1$.

2.2 Bilinear Diffie-Hellman assumption

Decision Diffie-Hellman: The Decision Diffie-Hellman problem (DDH) [16] in G_1 is to distinguish between the distributions $<P, aP, bP, abP>$ and $<P, aP, bP, cP>$ where a, b, c are random in Z_q^* and P is random in G_1^*. JOUX and NGUYEN [17] pointed out that DDH in G_1 is easy.

Bilinear Diffie-Hellman problem (BDH): Let G_1 and G_2 be two groups of prime order q. Let $\hat{e}: G_1 \times G_1 \rightarrow G_2$ be an admissible bilinear map and let P be a generator of G_1. The BDH problem in $<G_1, G_2, \hat{e}>$ is as follows: Given $<P, aP, bP, cP>$ for some $a, b, c \in Z_q^*$, compute $W = \hat{e}(P,P)^{abc} \in G_2$. An algorithm A has the advantage ε in solving BDH in $<G_1, G_2, \hat{e}>$ if

$$\Pr[\Phi(P, aP, bP, cP) = \hat{e}(P,P)^{abc}] \geq \varepsilon$$

Here, the probability is over the random choice of $a, b, c \in Z_q^*$, the random choice of $P \in G_1^*$, and the random bits of A.

BDH parameter generator: A randomized algorithm ς is a BDH parameter generator if 1) ς takes $k \in Z^+$ as a security parameter, 2) ς runs in polynomial time in k, and 3) ς outputs a prime number q, the description of two groups G_1 and G_2 of order q, and the description of an admissible bilinear map $\hat{e}: G_1 \times G_1 \rightarrow G_2$. The algorithm ς is denoted by $\varsigma(k) = <q, G_1, G_2, \hat{e}>$. The security parameter k determines the size of q. For i=1, 2, we assume that the description of the group G_i contains polynomial time algorithms for computing the group action in G_i and contains a generator of G_i. The generator of G_i enables us to generate uniformly random elements in G_i. Similarly, we assume that the description of $\hat{e}$ contains a polynomial time algorithm for computing $\hat{e}$.

BDH assumption: Let ς be a BDH parameter generator. An algorithm A has advantage $\varepsilon(k)$ in solving the BDH problem for ς if for sufficiently large k:

$$Adv_{\varsigma,A}(k) = \Pr[A(q, G_1, G_2, \hat{e}, P, aP, bP, cP) =$$

$$\hat{e}(P,P)^{abc} \left| \begin{array}{l} <q, G_1, G_2, \hat{e}> \leftarrow \varsigma(1^k) \\ P \leftarrow G_1^*, a, b, c \leftarrow Z_q^* \end{array} \right.] \geq \varepsilon(k)$$

ς satisfies the BDH assumption if $Adv_{\varsigma,A}(k)$ is negligible function for any randomized polynomial time algorithm Φ. BDH is hard in groups generated by ς, if ς satisfies the BDH assumption.

2.3 KIE-WSN model

There are several different ways to construct the KIE scheme after DODIS et al's [14] introduction of the idea key-insulated encryption. Considering the wireless sensor network's particularity, QIU et al constructed the KIE-WSN model based on the SKIE-OT scheme [18]. In the KIE-WSN model, there are four polynomial-time algorithms, which can be denoted by $KIE-WSN = \{PG, KG, Enc, Dec\}$.

PG: The PG algorithm takes a security parameter k as input and generates the system parameters $p_s = <q, G_1, G_2, \hat{e}, P, H_1, H_2>$.

Step 1: Input the security parameter k, run a polynomial-time algorithm $\Im$: $(q, G_1, G_2, \hat{e}) \leftarrow \Im(k)$ to get a safe prime q [19], two cyclic multiplicative groups G_1 and G_2, and the Bilinear map $\hat{e}: G_1 \times G_1 \rightarrow G_2$;

Step 2: Choose a random generator $P \xleftarrow{R} G_1$;

Step 3: Choose a proper n as the output range, and two cryptographic hash functions $H_1: \{0,1\}^* \rightarrow G_1^*$, $H_2: G_2 \rightarrow \{0,1\}^n$.

KG: The KG algorithm takes every sensor node's ID as input. It computes the system public key and the sensor's private key. For example, sun spot sensor node has a 48 bit length IEEE address. If setting the former 24 bit with the same value and the later 24 bit with different values to distinguish the sensor nodes in the sensor network, the later 24 bit can be used as the sensor's ID.

Step 1: Choose a random $s \xleftarrow{R} Z_q^*$ as system security parameter, and generate the public key $P_{\text{pub}} = sP$;

Step 2: Input sensor node ID=i, and compute the private key $sk_i = s \cdot Q_i = s \cdot H_1(i)$.

Enc: The Enc algorithm $C = Enc(P_{\text{pub}}, ID, K)$ takes the randomly chosen communication key and the other node's ID as input. It generates the cipher-text of this communication key.

Step 1: Choose a random symmetric key $K \xleftarrow{R} M$ as their common secret key between Node A and Node B.

Step 2: Suppose Node B's ID=i, and then compute $Q_i = H_1(i)$;

Step 3: Choose a random $r \xleftarrow{R} Z_q^*$;

Step 4: Compute the cipher-text $C = <U, V> = <rP, \quad K \oplus H_2(g_i^r)>$, where $g_i = \hat{e}(Q_i, P_{\text{pub}})$.

Dec: The Dec algorithm $K = Dec(C, sk)$ takes the cipher and private key as input, and outputs the secret symmetric key. If the input is the right private key corresponding to the node ID, the right K can be obtained; otherwise it will get invalid $\perp$. The Dec algorithm looks like below:

$$K' = V \oplus H_2(\hat{e}(sk_i, U)) = K$$

The correctness of decryption can be proved as follows:

$$\because \hat{e}(sk_i, U) = \hat{e}(sQ_i, rP) = \hat{e}(Q_i, P)^{sr} =$$

$$\hat{e}(Q_i, sP)^r = \hat{e}(Q_i, P_{\text{pub}})^r = g_i^r$$

$$\therefore K' = V \oplus H_2(\hat{e}(sk_i, U)) = K \oplus H_2(g_i^r) \oplus H_2(g_i^r) = K$$

3 KIE-WSN based group key scheme

3.1 Network and security assumptions

WSN sensors are assumed to collaborate with each other via unicast and broadcast channels. The nodes are able to get accesses to the broadcast messages and communicate with each other as a group. Figure 1 describes a randomly deployed WSN network model. In the network, a based station (BS) is responsible for generating a group event and deploying system parameters to the network. A group header (GH) is a node taking responsibility of all the key management tasks in the group, who has more powerful computing and storage resource. Dotted circle shows the broadcast range of GH, which allows a node to join a group if the node is in its vicinity. The group members coming under GH are at the distance of single hop from it. BS may send the message to a group header, which broadcasts the messages in its group members on behalf of BS. Let *p* be the probability that a sensor node can communicate with another sensor only by single hop, and *N* be the number of network nodes, $d=p(N-1)$ would be the expected connective degree of a node (i.e., the average number of edges connecting that node with its graph neighbors) [3]. Considering GH is more powerful than any general sensor node, we assume there are about $m=d-2d$ nodes in one group.

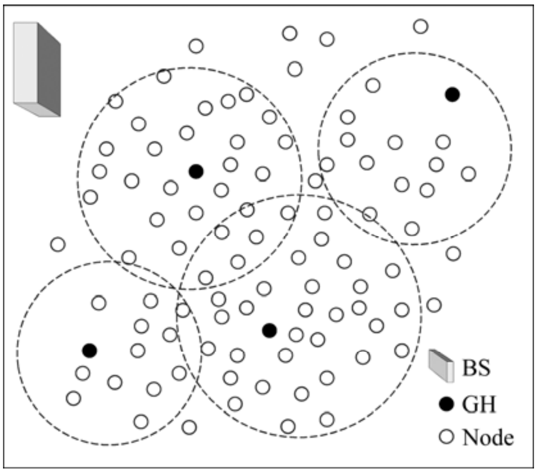

Fig. 1 A randomly deployed WSN network model [10]

The model is assumed to be static after the randomly deployment, i.e., sensor nodes are not mobile. BS, acting as a controller (or key server), is assumed to be a laptop-like device and supplied with long-lasting power. We hypothesize that all node's information will be leaked if compromised. For simplicity, we suppose that BS will not be compromised. Moreover, we assume that each sensor can detect the compromised neighbors and can exclude them at the key generation procedure.

Previous researches on group key management assume that users can dynamically join or leave the group [20]. When a node joins (or leaves) a group, the GK should be updated for the backward (or forward) security [21]. For nodes joining, we will pre-load some key management parameters to the newly deployed sensor nodes. We assume that no nodes will leave group voluntarily and only consider the node leaving event as the exclusion of a compromised node. The backward and the forward securities are defined as follows.

Forward security: When a node revokes from a group, it is unable to decrypt the future group messages with the keys it owns during the time as a legitimate group member.

Backward security: When a new node joins the group, it is unable to decrypt the previous group messages with keys it has now.

Here, we will define the *usk*, *hsk* and *ihsk*, which are used in the Group Rekeying protocol to update the GSK.

usk: The GSK updating authentication information, which will be saved in the sensor to compute the new GSK.

hsk: The GSK updating help information, which will be saved in the GH node.

ihsk: The GSK updating help information, whose value $ihsk=Q\cdot hsk$ is computed during the group rekeying.

3.2 Group key agreement scheme

The scheme is constructed of three phases: system parameters generation, group construction and group key agreement. The group header (GH), who is responsible for the GK generating and updating, is assumed to have strong computing and power resource.

System parameters generation: The system parameters generation phase consists of three off-line steps, namely generating of the system parameters, computing sensor's pair-wise key, and loading the parameters into the sensor nodes.

Step 1: Run the KIE-WSN's PG algorithm to get the system parameters $p_s = < q, G_1, G_2, \hat{e}, P, H_1, H_2 >$;

Step 2: Run the KG algorithm to get the pair-wise key parameters $p = < s, P_{\text{pub}} >$ and the sensor node private key sk_i, whose $ID=i$.

Step 3: Load the parameters $< P, P_{\text{pub}}, sk_i >$ into sensor node *i*.

Group construction: This phase builds the WSN group after the deployment, which consists of four steps.

Step 1: GH broadcasts notify message to find group members, who are in the broadcast range of GH shown in the Fig. 1. We suppose that there are *m* nodes wanting to join this group. The node can join more than one group if it is also in the other GH's broadcast range;

Step 2: GH sends request to the BS. BS runs KG algorithm to compute the secure GK parameters $p_{\mathrm{g}} =< s_{\mathrm{g}}, P_{\mathrm{gpub}} >$, and send them to the GH;

Step 3: GH chooses a random $ID \xleftarrow{R} N$, and computes the GSK $sk_{\mathrm{g}} = s_{\mathrm{g}} \cdot H_1(ID)$. It also computes its updating authentication information *usk* and help information *hsk*, which is used to update the GSK.

$$usk_j \xleftarrow{R} Z_q^*, hsk_j = s_{\mathrm{g}} - usk_j,\ 1 \le j \le m$$

Step 4: GH sends the $< usk_j, sk_g > (1 \le j \le m)$ by the security way to its group members;

Group key agreement: This phase is initiated by the GH and has three steps between the GH and the group members as shown in Fig. 2.

Step 1: GH chooses a random $K \xleftarrow{R} M$ as the *GK*, and then calls the KIE-WSN's Enc algorithm to compute the cipher-text $C =< U, V >= Enc(P_{\mathrm{gpub}}, ID, K)$;

Step 2: GH broadcasts the cipher-text *C* to group members;

Step 3: After receiving the cipher-text *C*, the group members call the KIE-WSN's Dec algorithm to obtain the GK using the GSK $K = Dec(C =< U, V >, sk_{\mathrm{g}})$.

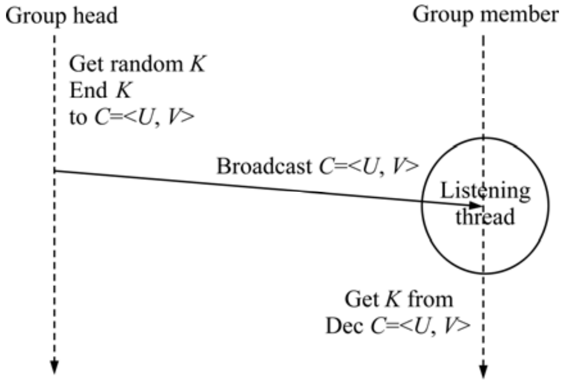

Fig. 2 Group key agreement protocol

3.3 Group rekeying protocol

Whenever one or several new sensors is added to the network, or one or several compromised sensors are forced to leave the network, the GK should be updated. To provide the forward and backward security, our KIE-GKMS scheme will update the GSK by three interactive steps as shown in Fig. 3.

SON et al [20] proved that if a node wants to deliver the same message to its different neighbors, it is more efficient on energy to deliver it by multicast. However, if a node wants to deliver different messages to its different neighbors, it is more efficient to deliver them one-by-one rather than multicast. To benefit from both the wireless multicast and unicast advantages, we first multicast the same part of the rekeying messages to them, and then unicast the different parts to nodes sequentially. Figure 3 shows the detail of the group rekeying protocol.

Step 1: GH chooses a random $ID' \xleftarrow{R} N$, $ID' > ID$, runs the KG algorithm to compute $Q{=}H_1(ID')$ and gets the new GSK: $sk'_{\mathrm{g}} = s_{\mathrm{g}} \cdot H_1(ID')$. Finally, the GH computes the GSK updating help information $ihsk_j = hsk_j \cdot H_1(ID'),\ 1 \le j \le m;$;

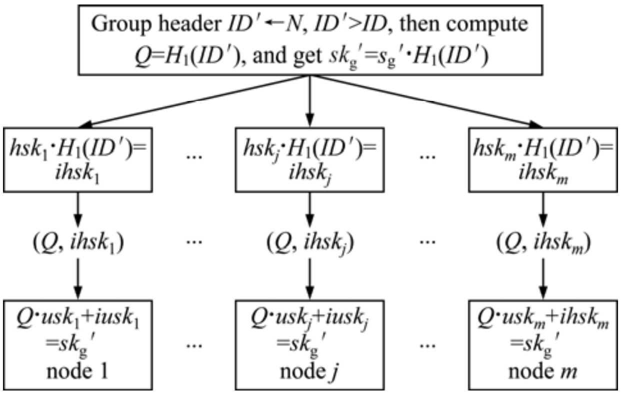

Fig. 3 Group rekeying protocol

Step 2: GH multicasts *Q*, and unicasts the message $ihsk_j$, $1 \le j \le m$ to the right group members;

Step 3: Receiving the GSK updating help information, the group members obtain the new GSK by the follow computation:

$$Q \cdot usk_j + ihsk_j = usk_j \cdot H_1(ID') + hsk_j \cdot H_1(ID')$$

$$= (usk_j + hsk_j) \cdot H_1(ID') = s_g \cdot H_1(ID') = sk'_g$$

4 Analysis

4.1 Security analysis

KIE-WSN semantic security: QIU et al [15] also defined the semantic security for KIE-WSN schemes with an IND-WSN-CPA game, which is identical to the IND-CPA game. KIE-WSN scheme is semantically secure if no polynomial-bounded adversary *A* has a non-negligible advantage against the challenger in the IND-WSN-CPA game [15].

Theorem 1 [15]: Suppose the hash functions H_1 and H_2 are random oracles. The KIE-WSN is a semantically secure key-insulated encryption scheme assuming BDH is hard in groups generated by ς. Concretely, suppose there is an IND-WSN-CPA adversary *A* that has advantage $\varepsilon(k)$ against the scheme KIE-WSN. Suppose *A* makes at most q_E>0 private key extraction queries and $q_{H_2} > 0$ hash queries to H_2, there exists an algorithm *B* that solves BDH in groups generated by ς with a minimal advantage:

$$Adv_{\varsigma,B}(k) \ge \frac{2\varepsilon(k)}{\mathrm{e} \cdot q_E \cdot q_{H_2}}$$

where *e*≈2.71 is the base of the natural algorithm. The running time of *B* is *O*(time(*A*).

KIE-(*t*, *N*): In the KIE scheme, if there are only *t* period private keys compromised, the left *N*−*t* period private keys are still safe.

Optimal KIE-(*t*, *N*): For a KIE-(*t*, *N*) safety scheme, if it has an optimal threshold, which means that we don't need to set the size of *N*, we call it optimal KIE-(*t*, *N*) safety.

QIU et al [15] have proved that the KIE-WSN model has the optimal KIE-(*N*−1, *N*) security, which means that even if *N*−1 node's private key has been compromised the left one node private key is still secure. At the same time, the threshold *N* is optimal.

Theorem 2 [15]: The KIE-WSN scheme has the optimal KIE-(*N*−1, *N*) secure.

Group confidentiality: If Node A is not the member of a group, then A doesn't know what the GSK is, which means A can't decrypt the cipher text of the GK during the group key agreement protocol. For this reasons, the confidentiality of the group is ensured.

Based on the optimal KIE-(*N*−1, *N*) security of KIE-WSN model, our group key management can easily provide the forward and backward security for the group members by updating the GSK. Even if *N*−1 GSKs are compromised, one rest GSK is still safe for the group.

When the GH is compromised, the whole group is destroyed. The BS will re-assign the new GH and run the group construction protocol to build a new group. For the group member, the scheme provides the forward and backward security as below.

Forward security: If a node, not the GH, is compromised, we provide the forward security by two steps.

Step 1: GH runs the group rekeying protocol but does not generate the updating help information for the compromised node. Therefore, without the new GSK's *ihsk*, the compromised node cannot compute the new GSK.

Step 2: GH runs the group key agreement protocol to update the GK.

Hence the compromised node cannot decrypt the future group message. By these two steps, the secure group key management scheme obtains the forward security.

Backward security: When a new node wants to join the group, we need to update the GK to support the backward security. In our assumption, suppose the new node *ID*=*t*, we will pre-load $<P, P_{\text{pub}}, sk_t>$ into the new sensor. There are four steps to update the new GK.

Step 1: The new node sends the join request to GH with its *ID*=*t*;

Step 2: GH computes the usk_i, and send it to the new node;

Step 3: GH runs group-rekeying protocol to update the GSK.

By this way, the new added node knows nothing about the former GSK, which guarantees the backward security of the group.

4.2 Communication analysis

Compared with RSA, ECC has advantages in the computational and storage costs with the same security level. ECC-160 (ECC-224) has the same security performance with RSA-1024 (RSA-2048) [12]. Here, let the element of G_1 be 160 bits, and the safe prime |*q*|=512 bit. We know that $s\in Z_q^*$, $usk\in Z_q^*$, $hsk\in Z_q^*$, and $sk\in G_1, sk_{\text{g}}\in G_1, P_{\text{pub}}\in G_1, P_{\text{gpub}}\in G_1$.

Group construction: For the GH, it receives $p_{\text{g}}=<s_{\text{g}}, P_{\text{gpub}}>$ from the BS, sends $<usk_j, sk_{\text{g}}>(1\le j\le m)$ to the group member. As a result, the communication traffic for GH is $|s_{\text{g}}|+|P_{\text{gpub}}|+(|usk|+|sk_{\text{g}}|)m=672(m+1)$ bit. While, for the sensor node, it will receive the $<usk_j, sk_{\text{g}}>$, so the communication traffic is 672 bit.

Group key agreement: During the key agreement phase, our scheme only needs to broadcast the cipher-text $C=<U,V>$ to the group members. *U* is the element of the group G_1, whose length is $|G_1|$=160, and *V* is the key length, and set |*V*|=*n*, so the communication cost equals to *n*+160. It is natural to set *n*=64 because of the symmetric cryptography algorithm, for example, DES, used in the WSN security communication. The communication traffic of our group key agreement protocol is only 224 bits.

Group rekeying: When there is a node compromised, we need update the GSK. Suppose the help information sequence number of the compromised node is *t*. In the proposed scheme, we will broadcast the common part *Q* and unicast the different part $ihsk_j$, $(1\le j\le m, j\ne t)$ to every group member. $|Q|=|H_1(ID')|=160$, and $ihsk_j$, $(1\le j\le m, j\ne t)$ are elements of the Z_q^*. The total communication cost for the GH is $(m-1)|ihsk|+|Q|=512m-352$ bit. However, for general group members, the communication cost is only $|Q|+|ihsk|=572$ bits, which is a constant value.

The comparison of communication cost of our scheme with PCGR is listed in Table 1. In the B-PCGR and C-PCGR [9] schemes, they also give the communication performance analysis. We can compare our scheme with their data. In PCGR scheme [9], *n* is the

Table 1 Communication cost compare with PCGR [9]

Scheme	Communication cost/bit
B-PCGR	*nL*
C-PCRG	2*nL*
KIE-GKMS's GH node	512*m*−352
KIE-GKMS's general node	572

average number of trusted neighbors that a node has, which has same meaning with *m* in our scheme. *L* is the length (in bits) of a group key. We suppose the group key length |*K*|=64 in our scheme.

It is clearly that, for the general group members, our scheme has higher performance during group rekeying. For the GH node, our scheme has similar performance with PCGR, and all of them have the linear performance.

Now the rekeying communication overload for the GH depends on the group size *m*, which can be evaluated by analyzing the connectivity of network with the help of random-graph theory [22]. A random graph *G*(*N*,*p*) is a graph of *N* nodes and the probability that a link exists between two nodes is *p*. When *p*=0, the graph does not have any edge, whereas when *p*=1, the graph is fully connected. ERDŐS and RÉNYI [22] showed that, for monotone properties, there is a value *p* such that the property moves from "nonexistent" to "certainly true" in a very large random graph. The function defining *p* is called the threshold function of a property. Given a desired probability P_c for graph connectivity, the threshold function *p* is defined as follows:

$$P_{\text{c}} = \lim_{n\to\infty} P_r[G(N,p)\text{is connected}] = \mathrm{e}^{-\mathrm{e}^{-c}}$$

where

$$p = \frac{\ln(N)}{N} + \frac{c}{N}, \ c \text{ is any real constant.}$$

Therefore, given *N*, we can find *p* and *d*=*p*(*N*−1), so that the resulting graph is connected with desired probability P_c. Figure 4 shows the plot of the expected degree of node *d*, which is a function of the network in size of *N*, for various values of P_c. From Fig. 5, if let $\Pr \le 0.999\ 999$, we can find the largest value of *d* is 24, and *m* will be 24−48, which means that the communication cost will be less than 12−24 kbit for the GH, when the total network size is not larger than 10 000.

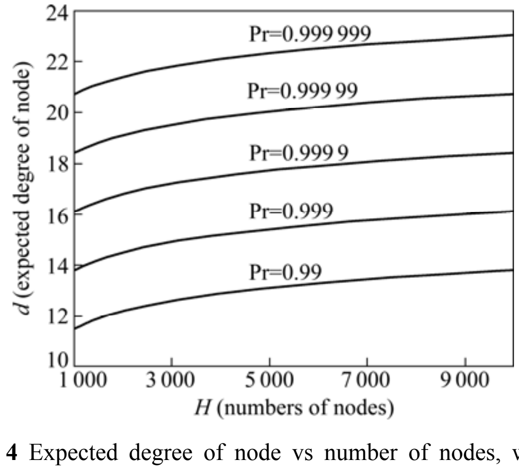

Fig. 4 Expected degree of node vs number of nodes, where P_c=Pr[*G*(*n*, *p*)] is connected [3]

4.3 Storage analysis

In our proposed scheme, for general group members, only $<P, P_{\text{pub}}, sk_i, sk_{\text{g}}, usk>$ is saved in the sensor memory. Consequently, our scheme has the constant performance in the memory usage, which can largely enhance the scalability of sensor network. For all the elements of $<P, P_{\text{pub}}, sk_i, sk_{\text{g}}>$ are the parameters of the group G_1 and |*usk*|=*q*, the memory cost of group members will only be 1 152 bit. However, for GH, it needs to load $<P, P_{\text{pub}}, sk_l, ID, P_{\text{gpub}}, sk_{\text{g}}>$ and $hsk_j, (1 \le j \le m)$ into memory. In the KIE-WSN mode, only the later 24 bit of IEEE addresses are used as ID, i.e., the |*ID*|=24. As a result, the storage cost of GH is 512*m*+824 bit.

We also compare the storage cost with the B-PCRG and C-PCRG [9] schemes, the results are listed in Table 2. In the PCRG [9] scheme, *s* is the degree of *g*′-polynomial *g*′(*x*).

Table 2 Storage cost compare with PCGR [9]

Scheme	Storage cost/bit
B-PCGR scheme	(*n*+1)(*s*+1)*L*
C-PCGR scheme	(2*n*+1)(*s*+1)*L*
KIE-GKMS's GH node	512*m*+824
KIE-GKMS's general node	1 152

Same with the communication cost comparison, for the general group member, our scheme has better performance than PCGR in the storage cost. In the PCGR's storage analysis, it gives example parameters that, *n*=20, *s*=30, *L*=64 bit. If use these parameters and set *m*=20 for KIE-GKMS scheme, the cost storage is shown in Table 3.

Table 3 Storage cost comparison under special system parameters

Scheme	Node's storage cost/bit	Group storage cost(*m*=20)/bit
B-PCGR scheme	41 664	833 280
C-PCGR scheme	81 344	1 626 880
KIE-GKMS's GH node	11 064	32 952
KIE-GKMS's general node	1 152	

First, GH storage costs are only 26.6% of B-PCGR and 13.6% of C-PCGR. Second, for the general nodes, the storage costs are only 2.77% of B-PCGR and 1.41% of C-PCGR. Finally, for the whole group, KIE-GKMS scheme's storage costs are only 3.95% of B-PCGR and 2.03% of C-PCGR. From this comparison, we can find that our KIE-GKMS saves a lot of storage for the sensors.

5 Conclusions

1) Scheme achieves both the forward security and the backward security.

2) Compromised nodes are unable to compute the updated GSK and GK. By updating the GK, the newly added group members cannot decrypt the former group messages. As the GSK is updated instead of discarding, the scheme reduces the communication and computing cost brought by the re-grouping or reinitializing the group, also its flexibility and scalability are improved.

3) For general sensor node of the group, analysis shows that the communication and storage performance is a constant value. However, for the group header, the communication and storage performance is $O(m)$.

4) Our scheme has constant performance for the general group members and lineal performance for the group header.

5) In future, a practical work is to improve the computing performance of the proposed scheme in highly constrained environments.

References

[1] AKYILDIZ I F, SU W, SAMKARASUBRAMANIAM Y, CAYIRCI E. Wireless sensor network: A survey [J]. Computer Networks, 2002, 38(4): 393−422.

[2] AZARDERKHSH R, REYHANI-MASOLEH A and ABID Z. A key management scheme for cluster based wireless sensor networks [C]// The IEEE/IFIP International Conference on Embedded and Ubiquitous Computing (EUC'08), Washington DC: IEEE Computer Society, 2008: 222−227.

[3] ESCHENAUER L, GLIGOR V D. A key management scheme for distributed sensor networks [C]// Proc of the 9th ACM Conference on Computer and Communication Security. New York, USA: ACM Press, 2002: 41−47.

[4] ZHANG Jun-qi, VIJAY V. A new security scheme for wireless sensor networks [C]// Global Telecommunication Conference, New Orleans: IEEE "GLOBECOM", IEEE Press, 2008: 1−5.

[5] MUSFIQ R, SRINIVAS S. A robust pair-wise and group key management protocol for wireless sensor network [C]// IEEE Globecom Workshop on Web and Pervasive Security, Miami: IEEE Press, 2010:1528−1232.

[6] CHOI S J, YOUN H Y. An efficient key pre-distribution scheme for secure distributed sensor networks [C]// Embedded and Ubiquitous Computing (EUC'05 Workshops), Nagasaki: Springer-Verlag, 2005, LNCS 3823: 1088−1097.

[7] WANG Y, RAMAMURTHY B. Group rekeying schemes for secure group communication in wireless sensor network [C]// The IEEE International Conference on Communications, Glasgow: IEEE Press, 2007: 3419−3424.

[8] ZHANG Yuan, SHEN Yong-luo, LEE Sang-ken. A cluster-based group key management scheme for wireless sensor networks [C]// The 12th International Asia-Pacific Web Conference. Busan: IEEE Press, 2010: 386−388.

[9] ZHANG Wen-shen, ZHU Sen-cun, CAO Guo-hong. Predistribution and local collaboration-based group rekeying for wireless sensor network [J]. Ad Hoc Networks, Elsevier, 2009, 7(6):1229−1242.

[10] KHALID A, HUSSAIN M. A secure group rekeying scheme with compromised node revocation in wireless sensor network [C]// ISA 2009, Berlin Heidelberg: Springer-Verlag, 2009, LNCS 5576: 712−721.

[11] GONG Zheng, TANG Qiang, LAW Yee-wei, CHEN Hong-yang. KALwEN+: Practical Key Management Schemes for Gossip-Based Wireless Medical Sensor Networks [C]// The 6th China International Conference on Information Security and Cryptology (Inscrypt 2010), LNCS 6584, Shanghai: Springer, 2011: 268−283.

[12] AMIN F, JAHANGIR A H, RASIFARD H. Analysis of public-key cryptography for wireless sensor network security [J]. World Academy of Science, Engineering and Technology 41, 2008: 531−534.

[13] JIANG Jian-wei, LIU Jian-hui. Research on key management scheme for WSN based on elliptic curve cryptosystem [C]// Network digital technologies (NDT '09), Ostrava: IEEE Press, 2009: 536−540.

[14] DODIS Y, KATZ J, XU S, YUNG M. Key-insulated public key cryptosystems [C]// Advances in Cryptology-EUROCRYPT'02, London,UK: Springer-Verlag, 2002, LNCS 2332: 65−82.

[15] QIU Wei-dong, ZHOU Yao-wei, ZHU Bo, ZHENG Yan-fei, WEN Mi, GONG Zheng. Key-insulated encryption based key pre-distribution scheme for WSN [C]// ISA 2009, Berlin Heidelberg: Springer-Verlag, 2009, LNCS 5576: 200−210.

[16] BONEH D. The decision Diffie-Hellman problem [C]// Algorithmic Number Theory Symposium, Berlin Heidelberg: Springer-Verlag, 1998, LNCS 1423: 48−63.

[17] JOUX A, Nguyen K. Separating decision Diffie-Hellman from Diffie-Hellman in cryptographic groups [J]. Journal of Cryptology, 2003, 16(4): 239−247.

[18] BELLARE M, PALACIO A. Protecting against key-exposure: Strongly key-insulated encryption with optimal threshold [J]. AAECC, 2006, 16(6): 379−396.

[19] MICHAEL S. Computing the Tate Pairing [C]// Topic in Cryptology-CT-RSA 2005, Berlin Heidelberg: Springer-Verlag, 2005, LNCS 3376: 293−304.

[20] SON J H, LEE J S, SEO S W. Energy efficient group key management scheme for wireless sensor network [C]// The 2nd International Conference on Communication Systems Software and Middleware, Bangalore: IEEE Press, 2007: 1−9.

[21] REN Yi, OLESHCHUK V, LI Frank-Y. An efficient chinese remainder theorem based node capture resilience scheme for mobile WSNs [C]// IEEE International conference on Information Theory and Information Security (ICITIS), Beijing: IEEE Press, 2010: 689−692.

[22] SPENCER J. The strange logic of random graphs, algorithms and combinatorics [M]. Berlin Heidelberg: Springer-Verlag 2001: 68−85.

(Edited by HE Yun-bin)