International Journal of Security and Its Applications
Vol.8, No.1 (2014), pp.81-92
http://dx.doi.org/10.14257/ijsia.2014.8.1.08

Freshness Consideration of Hierarchical Key Agreement Protocol in WSNs

Sung-Woon Lee[1] and Hyunsung Kim[2]

[1]*Dept. of Information Security, Tongmyong University*
Namgu, Busan 608-711, Korea
[2]*Dept. of Cyber Security, Kyungil University*
Kyungsan, Kyungbuk 712-701, Korea
[1]*staroun@tu.ac.kr*, [2]*kim@kiu.ac.kr*

Abstract

Wireless sensor networks (WSNs) are often deployed in potentially adverse or even hostile environment so that there are concerns on security issues in these WSNs. Sensor nodes with resource-constrained make security applications a challenging problem. Key agreement is a fundamental security service in WSNs, which enables sensor nodes to communicate securely with each other using cryptographic techniques. However, due to the resource constraints on sensor nodes, it is infeasible to use traditional key management techniques such as public key cryptography and key distribution center. Recently, Kim proposed two efficient and non-interactive hierarchical key agreements in WSNs, which have good properties including non-interactive, hierarchical, resilient, etc. However, Kim's protocols do not support freshness of the established session key that key agreement protocol should supports. Thereby, we propose two freshness preserving hierarchical key agreement protocols over the hierarchical WSNs, named as HKAP_FP. Our two HKAP_FPs inherit advantages from Kim's protocols and well suited to the hierarchical WSNs.

Keywords: *Wireless Sensor Network Security, Hierarchical Key Agreement, Freshness, Identity-based Encryption*

1. Introduction

The wireless sensor networks (WSNs) have recently emerged as a promising computing model for many civilian and military applications such military target tracking and surveillance, natural disaster relief, biomedical health monitoring, and hazardous environment exploration and seismic sensing. It usually consists of a large number of low-cost, battery-powered sensor nodes that are of limited computation and communication capability. While these nodes are left unattended after deployment, they can adaptively form a routing graph and continuously collect data for events of interests and deliver the data to a designated sink node, a node that is usually resource-abundant and trustworthy [1-3].

The unattended nature of a WSN makes it vulnerable to varying forms of security attacks such as a compromised node injecting false data reports [4-7]. Security services such as authentication and key management are critical to secure the communication between sensors in hostile environments. As one of the most fundamental security services, pair-wise key establishment enables the sensor nodes to communicate securely with each other using cryptographic techniques. However, due to the resource constraints on sensor nodes, it is not feasible for sensors to use traditional pair-wise key

ISSN: 1738-9976 IJSIA

establishment techniques such as public key cryptography and key distribution center [8-9].

Recently, Guo *et al.*, proposed an efficient and non-interactive hierarchical key agreement protocol applicable to mobile ad-hoc networks [10]. Guo *et al.'s*, protocol is based on the pairing cryptography and satisfies the desired properties mentioned in [11] for authenticated key agreement protocol for mobile ad-hoc networks and tactical networks. However, their protocol could not be applied to the WSNs as it is due to the WSN's uniqueness. Thereby, Kim proposed two non-interactive hierarchical key agreement protocols, named as the naïve HKAP and the privacy HKAP, over the hierarchical WSNs, which are a revised version of Guo *et al.'s*, protocol for the WSNs [9]. Kim's protocols are secure against the corruption of any number of nodes at any level in the hierarchy. However, Kim's protocols do not support freshness of the established session key that key agreement protocol should supports.

The purpose of this paper is to devise two freshness preserving hierarchical key agreement protocols over the hierarchical WSNs to solve the problem in Kim's protocols. To devise new protocols with fresh provision, we first design a naïve HKAP_FP to support features of non-interactive, hierarchical, resilient, and session freshness. We further design a privacy HKAP_FP based on the naïve one, which supports anonymity and session freshness. They fall into two phases, a hierarchical key settlement phase and a session key agreement and secure communication phase. The first phase is for setting up the system and the other one is to communicate by using a secure channel after agreeing on a secure session key between any two nodes in the WSN. Our two revisions could support security, robustness, and session key freshness over the hierarchical WSNs.

This paper is organized as follows. Section 2 reviews Kim's non-interactive hierarchical identity-based key agreements. Two new HKAP_FPs are proposed over the hierarchical WSNs in Section 3. Security analyses are provided in Section 4. Finally, Section 5 gives a brief conclusion.

2. Kim's Non-Interactive Hierarchical Identity-based Key Agreement

This section reviews Kim's non-interactive hierarchical identity-based key agreement, named as HKAP, which is focused on the naïve one [9]. Furthermore, we show that the HKAP does not provide the session freshness, which is one of important features for the key agreement protocol.

2.1. Kim's Naïve HKAP

Similarly to other identity-based authenticated key agreement protocols in [12-14], Kim's naïve HKAP requires a private key generator (PKG) and consists of two phases: key settlement phase and session key agreement and secure communication phase [9]. Let k be the security parameter, G and G_T be two cyclic groups of prime order q, and $\hat{e}$: $G \times G \rightarrow G_T$ be a bilinear pairing. He denotes by G^* the non-identity elements set of G and assumed that public keys (identities or IDs) at depth l are vectors of elements in $(G^*)^l$. The j-th component corresponds to the identity at level j. The system later extends the construction to public keys over $\{0, 1\}^*$ by first hashing each component I_j using a collision resistant hash $H_1 : \{0,1\}^* \rightarrow G$. Notations used in the protocol are listed in Table 1.

Table 1. Notations

Notation	Description
CH_i	Cluster head i
CM_{ij}	Member node j in the cluster head i
ID_i	Entity i's identifier
E_i	Amplified identity of ID_i
(S_1, S_2, S_3)	Private key set of sink, $S_i \in Z_q^*$
sk	Session key established between two entities
r_1	Random number
G, G_T	Cyclic groups of prime order q
P	Denote a generator of G
$\hat{e}$	bilinear map $G \times G \rightarrow G_T$
$h()$	One way hash function $h : \{0, 1\}^* \rightarrow G^*$
$\cdot$	Multiplication
$\|\|$	Concatenation

Key Settlement Phase: This phase is to make that each node has a pair of keys for the public key cryptosystem, a public key and a private key. Sink node performs the role of a PKG. It is assumed that sink has a super power than the other nodes, cluster heads have superior than sensor nodes but lower than sink, and sensor nodes have the lower rights than any other nodes, and the role of nodes is pre-allocated before the phase. Figure 1 shows the concept of Kim's key settlement phase for the hierarchical WSN. To set up keys, the protocol performs the following operations

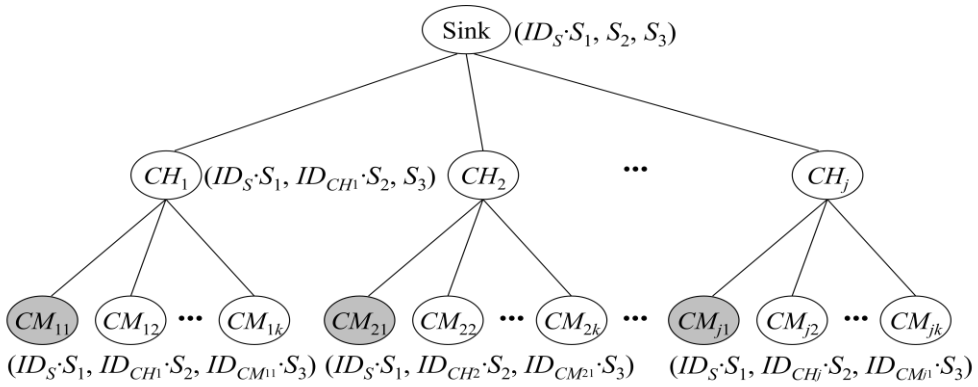

Figure 1. Hierarchical Key Settlement Model for Kim's Naïve HKAP

Step 1. Sink with identities ID_S creates a private key set (S_1, S_2, S_3) for a WSN and computes $ID_S \cdot S_1$. After that, sink stores the information in it's memory and sends $\{(ID_S \cdot S_1, S_2, S_3), ID_S\}$ to cluster heads.

Step 2. When a cluster head with identities ID_{CHi} receives the message, it computes $ID_{CHi} \cdot S_2$. After that, the cluster head stores the information in it's memory and sends $\{(ID_S \cdot S_1, ID_{CHi} \cdot S_2, S_3), ID_S, ID_{CHi}\}$ to it's member nodes.

Step 3. When a sensor node with identities ID_{CMij} receives the message, it computes $ID_{CMij} \cdot S_3$. After that, the node stores the information $\{(ID_S \cdot S_1, ID_{CHi} \cdot S_2, ID_{CMij} \cdot S_3), ID_S, ID_{CHi}, ID_{CMij}\}$ in it's memory.

Session Key Agreement and Secure Communication Phase: The purpose of this phase is to establish a secure channel by establishing a session key between any two nodes in the WSN. To establish a shared session key, CM_{ij} and CM_{kl} conduct the following tasks

Step 1. CM_{ij} with it's private key set ($ID_S \cdot S_1$, $ID_{CHi} \cdot S_2$, $ID_{CMij} \cdot S_3$) computes $sk=\hat{e}(ID_S \cdot S_1, ID_S') \cdot \hat{e}(ID_{CHi} \cdot S_2, ID_{CHk}') \cdot \hat{e}(ID_{CMij} \cdot S_3, ID_{CMkl}')$ by using the amplified ID set of the counterpart node CM_{kl}, which is $\{ID_S', ID_{CHk}', ID_{CMkl}'\}$. Independent with CM_{ij}, CM_{kl} with it's private key set ($ID_S \cdot S_1$, $ID_{CHk} \cdot S_2$, $ID_{CMkl} \cdot S_3$) computes $sk=\hat{e}(ID_S \cdot S_1, ID_S') \cdot \hat{e}(ID_{CHk} \cdot S_2, ID_{CHi}') \cdot \hat{e}(ID_{CMkl} \cdot S_3, ID_{CMij}')$ by using the amplified ID set of the counterpart node CM_{ij}, which is $\{ID_S', ID_{CHi}', ID_{CMij}'\}$. CM_{ij} and CM_{kl} can compute the same shared session key due to $sk=\hat{e}(ID_S \cdot S_1, ID_S) \cdot \hat{e}(ID_{CHi} \cdot S_2, ID_{CHk}) \cdot \hat{e}(ID_{CMij} \cdot S_3, ID_{CMkl}) = \hat{e}(ID_S \cdot S_1, ID_S) \cdot \hat{e}(ID_{CHk} \cdot S_2, ID_{CHi}) \cdot \hat{e}(ID_{CMkl} \cdot S_3, ID_{CMij}) = \hat{e}(ID_S, ID_S)^{S_1} \cdot \hat{e}(ID_{CHk}, ID_{CHi})^{S_2} \cdot \hat{e}(ID_{CMkl}, ID_{CMij})^{S_3} = \hat{e}(ID_S, ID_S)^{S_1} \cdot \hat{e}(ID_{CHi}, ID_{CHk})^{S_2} \cdot \hat{e}(ID_{CMij}, ID_{CMkl})^{S_3}$.

Step 2. The source node CM_{ij} sends an encrypted data packet and the message digest $MAC=h(sk||$the encrypted data packet) to it's counterpart node CM_{kl}, which is encrypted by using the agreed session key sk.

Step 3. After receiving the encrypted message, the destination node CM_{kl} checks the validity of MAC by using the agreed session key sk. Only if the validity check is successful, CM_{kl} accepts the message from CM_{ij}, which means that the encrypted message is successfully transferred by using the agreed secure channel based on sk.

2.2. No Key Freshness Support Problem in Kim's HKAP

A key establishment or agreement process among the participants should guarantee that each shared session key is fresh, i.e. has not been reused by one of the participants [15]. This also means that a key used in one cryptographic association has not been used in another association. Thus, the session key needs to be changed over time since a key may be compromised during pre-deployment or operational phases of communication networks.

In Kim's naïve HKAP, each party computes sk between any two entities in the WSN, which depends on both of their own private key and identity tuples but not on the session dependent random value. Thereby, the naïve HKAP does not provide key freshness. No freshness support means that the established session keys in different sessions are always the same, which could provide some means or useful information to attacker. One of serious effects is traffic analysis attack, which is focused on traffic flow identification, traffic flow tracking, or disclosing application-level information.

3. Hierarchical Key Agreement Protocol with Freshness Property

This section proposes two new hierarchical key agreement protocols with freshness property, named as naïve HKAP_FP and privacy HKAP-FP, using pairings over the hierarchical WSNs. They fall into two phases: a hierarchical key settlement phase and a session key agreement and secure communication phase. The first phase is for setting up the system which is the same as Kim's, and the other one is to communicate by using a secure channel after establishing a fresh session key between any two nodes in the hierarchical WSN.

We assume the same assumptions in Kim's HKAP that the network is formed in hierarchy, one hop is considered between sensor nodes and a head in a cluster and multiple hops are assumed between cluster heads and the sink over the network. Thereby, this paper also follows the hierarchy of WSN and considers a hierarchical tree with depth 3. For the tree construction, it is assumed that the degree of sink node is u and the degree of cluster head is v, respectively, which are determined by the number of nodes n in a WSN and the protocol uses the related previous schemes to form equally distributed clusters in the network.

3.1. Naïve HKAP

To establish a shared key between two nodes in a WSN, it is necessary to pre-establish secret keys. The purpose of key settlement phase is to establish necessary secret keys before they are deployed. Nodes in WSNs indeed have met before their deployment because all these nodes usually belong to the same administrative entity. This is a major difference between WSN environments and the other mobile network environments. In many WSN applications, sensor nodes do know and trust each other before the deployment. In other words, before their deployment, sensor nodes are in a benign environment where they can exchange information in plaintext and thus establish trust relationships among themselves

Hierarchical Key Settlement Phase: This phase is the same as in Kim's HKAP, which is described in Section 2.

Session Key Agreement and Secure Communication Phase: The purpose of this phase is to establish a secure channel by establishing a secure fresh session key between any two nodes in the WSN. To establish a shared session key, CM_{ij} and CM_{kl} conduct the following tasks

Step 1. CM_{ij} with it's private key set ($ID_S \cdot S_1$, $ID_{CH_i} \cdot S_2$, $ID_{CM_{ij}} \cdot S_3$) chooses a random number r_1, computes $R_1 = r_1 \cdot ID_{CM_{ij}}$, a fresh session key $sk = \hat{e}(ID_S \cdot S_1, ID_S') \cdot \hat{e}(ID_{CH_i} \cdot S_2, ID_{CH_k}') \cdot \hat{e}(ID_{CM_{ij}} \cdot S_3, ID_{CM_{kl}}')^{r_1}$ by using the amplified ID set of the counterpart node CM_{kl}, which is $\{ID_S', ID_{CH_k}', ID_{CM_{kl}}'\}$, and $MAC_1 = h(sk, R_1)$, and sends $\{ R_1, MAC_1 \}$ to CM_{kl}.

Step 2. When CM_{kl} receives the message, it with it's private key set ($ID_S \cdot S_1$, $ID_{CH_k} \cdot S_2$, $ID_{CM_{kl}} \cdot S_3$) computes a fresh session key $sk = \hat{e}(ID_S \cdot S_1, ID_S') \cdot \hat{e}(ID_{CH_k} \cdot S_2, ID_{CH_i}') \cdot \hat{e}(ID_{CM_{kl}} \cdot S_3, R_1)$ by using the amplified ID set of the counterpart node CM_{ij}, which is $\{ID_S', ID_{CH_i}', ID_{CM_{ij}}'\}$. CM_{ij} and CM_{kl} can compute the same shared fresh session key due to $sk = \hat{e}(ID_S \cdot S_1, ID_S) \cdot \hat{e}(ID_{CH_i} \cdot S_2, ID_{CH_k}) \cdot \hat{e}(ID_{CM_{ij}} \cdot S_3, ID_{CM_{kl}})^{r_1} = \hat{e}(ID_S \cdot S_1, ID_S) \cdot \hat{e}(ID_{CH_k} \cdot S_2, ID_{CH_i}) \cdot \hat{e}(ID_{CM_{kl}} \cdot S_3, R_1) = \hat{e}(ID_S, ID_S)^{S_1} \cdot \hat{e}(ID_{CH_k}, ID_{CH_i})^{S_2} \cdot \hat{e}(ID_{CM_{kl}}, ID_{CM_{ij}})^{S_3 r_1} = \hat{e}(ID_S, ID_S)^{S_1} \cdot \hat{e}(ID_{CH_i}, ID_{CH_k})^{S_2} \cdot \hat{e}(ID_{CM_{ij}}, ID_{CM_{kl}})^{S_3 r_1}$. CM_{kl} assures the correctness of the established fresh session key only if the validity check of MAC_1 is successful by comparing it with $h(sk, R_1)$.

Step 3. CM_{kl} sends back an encrypted data packet with the message digest $MAC_2 = h(sk \| \text{the encrypted data packet})$ to it's counterpart node CM_{ij}, which is encrypted by using the agreed session key sk.

Step 4. After receiving the encrypted message, CM_{ij} checks the validity of MAC_2 by using the agreed session key sk. Only if the validity check is successful, CM_{ij}

accepts the message from CM_{kl}, which means that the encrypted message is successfully transferred by using the agreed secure channel based on *sk*.

The proposed session key agreement and secure communication phase has a good advantage in the perspective of communication cost to agree on a fresh session key.

3.2. Privacy HKAP_FP

To support the privacy issue, this subsection further proposes a privacy supporting HKAP_FP based on the Naïve one by using amplified identities not using real identities of nodes.

Hierarchical Key Settlement Phase: The assumptions and steps of this phase for the privacy HKAP_FP are the same as Kim's privacy HKAP in [9]. Fig. 2 shows a concept of the hierarchical key settlement phase for the privacy HKAP_FP. To set up keys, the phase performs the following operations

Step 1. Sink with identities ID_S creates a private key set (S_1, S_2, S_3) for a WSN and computes $E_S=h(ID_S)$ and $E_S \cdot S_1$, where $h()$ is a one-way hash function. After that, sink stores the information in it's memory and sends $\{(E_S \cdot S_1, S_2, S_3), E_S\}$ to cluster heads.

Step 2. When a cluster head with identities ID_{CHi} receives the message, it computes $E_{CHi}=h(ID_{CHi})$ and $E_{CHi} \cdot S_2$. After that, the cluster head stores the information in it's memory and sends $\{(E_S \cdot S_1, E_{CHi} \cdot S_2, S_3), E_S, E_{CHi}\}$ to it's member nodes.

Step 3. When a sensor node with identities ID_{CMij} receives the message, it computes $E_{CMij}=h(ID_{CMij})$ and $E_{CMij} \cdot S_3$. After that, the node stores the information $\{(E_S \cdot S_1, E_{CHi} \cdot S_2, E_{CMij} \cdot S_3), E_S, E_{CHi}, E_{CMij}\}$ in it's memory.

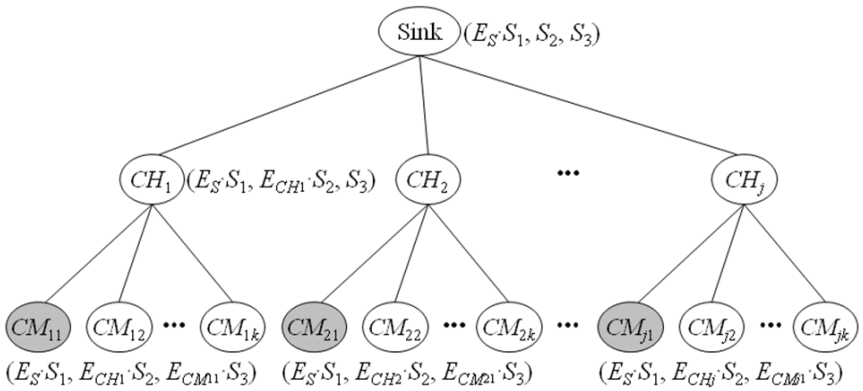

Figure 2. Hierarchical Key Settlement Model for Privacy HKAP_FP

Session Key Agreement and Secure Communication Phase: The purpose of this phase is to establish a secure channel by establishing a fresh session key between any two nodes in the WSN. To establish a shared fresh session key, CM_{ij} and CM_{kl} conduct the following tasks

Step 1. CM_{ij} with it's private key set ($E_S \cdot S_1$, $E_{CHi} \cdot S_2$, $E_{CMij} \cdot S_3$) chooses a random number r_1, computes $R_1=r_1 \cdot E_{CMij}$, a fresh session key $sk=\hat{e}(E_S \cdot S_1, E_S') \cdot \hat{e}(E_{CHi} \cdot S_2, E_{CHk}') \cdot \hat{e}(E_{CMij} \cdot S_3, E_{CMkl}')^{r_1}$ by using the amplified ID set of the

counterpart node CM_{kl}, which is $\{E_S', E_{CHk}', E_{CMkl}'\}$, and $MAC_1=h(sk, R_1)$, and sends $\{ R_1, MAC_1 \}$ to CM_{kl}.

Step 2. When CM_{kl} receives the message, it with it's private key set ($E_S \cdot S_1$, $E_{CHk} \cdot S_2$, $E_{CMkl} \cdot S_3$) computes $sk=\hat{e}(E_S \cdot S_1, E_S') \cdot \hat{e}(E_{CHk} \cdot S_2, E_{CHi}') \cdot \hat{e}(E_{CMkl} \cdot S_3, R_1)$ by using the amplified ID set of the counterpart node CM_{ij}, which is $\{E_S', E_{CHi}', E_{CMij}'\}$. CM_{ij} and CM_{kl} can compute the same fresh shared session key due to $sk=\hat{e}(E_S \cdot S_1, E_S) \cdot \hat{e}(E_{CHi} \cdot S_2, E_{CHk}) \cdot \hat{e}(E_{CMij} \cdot S_3, E_{CMkl})^{r_1}=\hat{e}(E_S \cdot S_1, E_S) \cdot \hat{e}(E_{CHk} \cdot S_2, E_{CHi}) \cdot \hat{e}(E_{CMkl} \cdot S_3, R_1)=\hat{e}(E_S, E_S)^{S_1} \cdot \hat{e}(E_{CHk}, E_{CHi})^{S_2} \cdot \hat{e}(E_{CMkl}, E_{CMij})^{S_3 r_1}=\hat{e}(E_S, E_S)^{S_1} \cdot \hat{e}(E_{CHi}, E_{CHk})^{S_2} \cdot \hat{e}(E_{CMij}, E_{CMkl})^{S_3 r_1}$. CM_{kl} assures the correctness of the established fresh session key only if the validity check of MAC_1 is successful by comparing it with $h(sk, R_1)$.

Step 3. CM_{kl} sends back an encrypted data packet with the message digest $MAC_2=h(sk||$the encrypted data packet$)$ to it's counterpart node CM_{ij}, which is encrypted by using the agreed session key sk.

Step 4. After receiving the encrypted message, CM_{ij} checks the validity of MAC_2 by using the agreed fresh session key sk. Only if the validity check is successful, CM_{ij} accepts the message from CM_{kl}, which means that the encrypted message is successfully transferred by using the agreed secure channel based on sk.

The proposed session key agreement and secure communication phase has a good advantage in the perspective of node communication cost to agree on a fresh session key.

4. Analyses

This section provides only security analyses focused on the privacy HKAP_FP. We follow the approaches used in [16] for comparison purpose. This section gives computational problems, which are based on the security of our protocols and provides various security analyses.

4.1. Computational Problems

Bilinear map captures an important cryptographic problem, *i.e.*, the Biliniear Diffie-Hellman (BDH) problem, which was introduced by Boneh and Franklin in [13]. The security of our protocol relies on a variant of the BDH assumption.

Let G and G_T be two groups of a prime order q. Suppose that there exists a bilinear map $\hat{e}$: $G \times G \rightarrow G_T$. We consider the following computational assumptions

- Bilinear Diffie-Hellman (BDH) : For a, b, and $c \in_R Z_q^*$ and given aP, bP, and cP, computing $\hat{e}(P, P)^{abc}$ is hard
- Decisional Bilinear Diffie-Hellman (DBDH) : For a, b, and $c \in_R Z_q^*$, differentiating (aP, bP, cP, $\hat{e}(P, P)^{abc}$) and (aP, bP, cP, $\hat{e}(P, P)^r$) is hard

4.2. Security Analyses

Our security analysis is focused on verifying the overall security requirements for the proposed privacy HKAP_FP including passive and active attacks as follows.

Proposition 1. The proposed privacy HKAP_FP provides entity anonymity.

Proof: In the privacy HKAP_FP, the anonymity of entity is obtained by hash function and the BDH problem. The key settlement phase and key agreement and secure communication phase use the amplified identity of each entity by applying hash function from the real identity. Only sink can get the real identity of cluster heads and sensor nodes. Assume that the attacker has captured the messages { R_1, MAC_1 } and {an encrypted data packet, MAC_2} from the key agreement and secure communication phase. However, there are no ways that the attacker can derive the real identity of communication parties.

Proposition 2. The proposed privacy HKAP_FP could not reveal the private key or the generated session key to outsiders.

Proof: If we consider the confidentiality of the private key set, the key set is combinations of the amplified identity and secret value. This indicates that the attacker has to know both of them to know the private key set. However, there are no ways that the attacker can derive the secret value or the amplified identity from the private key set even if the attacker is registered to the sink. Also, in order to obtain the session key *sk*, the attacker must try to derive *sk* from any intercepted messages { R_1, MAC_1 } and {an encrypted data packet, MAC_2}. However, there are no ways that the attacker can derive the shared key due to the BDH and DBDH problems even if the attacker is registered entity.

Proposition 3. The proposed privacy HKAP_FP provides session key freshness and thereby can prevent the replay attack.

Proof: The fresh nonce used in the session key agreement and secure communication phase guarantees the freshness of the session keys. To achieve freshness, session initiator uses a nonce r_1 along with MAC_1 to generate session key *sk*. There are no ways that the attacker can generate session key due to the BDH problem even if the attacker is registered entity. Furthermore, the privacy HKAP_FP is strong against the replay attack due to the session key freshness.

Proposition 4. The proposed privacy HKAP_FP is secure against passive attack.

Proof: We assume that an adversary is success if the adversary could learn some useful information from the intercepted messages. We show that probability to succeed in learning them is negligible due to the difficulty of the underlying cryptosystem, the BDH problem, and the DBDH problem.

1. A completeness of the key agreement protocol is already proven by describing the run of the protocol in section 3.
2. If the adversary is passive adversary, all the adversary can gather are as follows: the amplified identity set {E_S', E_{CHj}', E_{CMk}'} and the message digest *MAC*. However, it is negligible to find the key related information from them due to the difficulty of the underlying cryptosystem, the BDH problem, and the DBDH problem.

Finally, we could say the proposed privacy HKAP_FP is secure against passive attack.

Proposition 5. The proposed privacy HKAP_FP is secure against active attack.

Proof: We assume that an adversary is success if the adversary finds the session key *sk* or the session key related information {S_1, S_2, S_3}. Therefore, we show that probability to succeed in finding them is negligible due to the difficulty of the underlying cryptosystem, the BDH problem, and the DBDH problem.

1. The acceptance by all entities means that each *MAC* in the corresponding message is successfully verified. That is, *MAC* is decrypted and verified successfully by using the correct session key *sk*. We show that if it is the case that entities accept the messages and continue the session, then the probability that the adversary have modified the messages being transmitted is negligible. And the only way for the adversary to find the session key or security related information is to solve the difficulty of the underlying cryptosystem, the BDH problem, and the DBDH problem.
2. Now, we consider the active adversary with following cases.
   (a) There is no way that an adversary could get the secret information $\{S_1, S_2, S_3\}$ due to the difficulty of the BDH problem and the DBDH problem.
   (b) An adversary cannot impersonate CM_{ij} or CH_i to cheat the sink. That is the attacker cannot generate valid messages without deriving the correct session key *sk*, since the attacker cannot pass the verification of *MAC* in the protocol.
   (c) An adversary cannot impersonate the sink to cheat CM_{ij} or CH_i. As described above, only the legal sink can form the legal messages by including the proper check sum, which needs to be properly matched with the information from CM_{ij} or CH_i in the protocol steps. Even if the attacker could pass the verifications at the protocol steps, the attacker still cannot get any useful information from the encrypted messages due to the difficulty of the underlying public-key cryptosystem and cannot generate the consequent valid messages.

Finally, we could say the proposed privacy HKAP_FP is secure against active attack.

4.3. Comparison

Key management in WSNs demands extra space to keep the required keys for the secure communication. The performance comparisons provided in this subsection presents the space and computation overhead that are related with the size of the key set. A simple node stores at least three key related information from $\{S_1, S_2, S_3\}$. Table 2 shows the feature comparisons between the proposed privacy HKAP_FP, Kim's HKAP in [9] and Guo *et al.'s*, protocol in [10].

Table 2. Comparisons between Related Protocols

Features / Protocols	Size of key set	Number of pairing operations	Session key Freshness supporting
Guo et al.'s	log *n*	log *n*	No
Privacy supporting HKAP	Constant	Constant	No
Privacy HKAP_FP	Constant	Constant	Yes

The size of key set in Guo *et al.'s*, protocol is dependent with the number of nodes in the network but not in HKAP and our HKAP_FP, which keeps the same size of key set with constant of 3. Furthermore, the computational overhead in our HKAP_FP does not affect to the size of key set but Guo *et al.'s*, protocol does affect. This is very important property especially in WSN with the limits of battery life or resource constraints.

5. Conclusion

Recently, Guo *et al.*, proposed an efficient and non-interactive hierarchical key agreement protocol applicable to mobile ad-hoc networks. However, their protocol could not be applied to the WSNs as it is due to the WSN's uniqueness. Thereby, Kim proposed two privacy supporting non-interactive hierarchical key agreement protocols over the hierarchical WSNs, which is a revised version of Guo *et al.'s*, protocol for the WSNs. Kim's protocols are secure against the corruption of any number of nodes at any level in the hierarchy. However, Kim's protocols do not support freshness of the established session key that key agreement protocol should supports. Thereby, we proposed two freshness preserving hierarchical key agreement protocols over the hierarchical WSNs. Our revisions inherit advantages from Kim's protocols and well suited to the hierarchical WSNs.

Acknowledgements

This work was supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2010-0021575) and also was supported by the National Research Foundation of Korea Grant funded by the Korean Government (MEST) (NRF-2011-0008890).

References

[1] A. Hadjidg, M. Souil, A. Bouabdallah, Y. Challal and H. Owen, "Wireless sensor networks for rehabilitation applications: Challenges and opportunities", Journal of Network and Computer Applications, vol. 36, **(2013)**, pp. 1-15.
[2] G. Mao, B. Fidan and B. Anderson, "Wireless sensor network location techniques", Computer Networks: The International Journal of Computer and Telecommunications Networking, vol. 29, **(2007)**, pp. 2529-2553.
[3] J. Yick, B. Mukherjee and D. Ghosal, "Wireless sensor network survey", Computer Networks, vol. 52, **(2008)**, pp. 2292-2330.
[4] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: attacks and countermeasures", Proc. of IEEE international workshop on sensor network protocols and applications, **(2003)**, pp. 113-127.
[5] P. Kumar and H.-J. Lee, "Security Issues in Healthcare Applications Using Wireless Medical Sensor Networks: A Survey", Sensors. vol. 12, **(2012)**, pp. 55-91.
[6] N. R. Prasad and M. Alam, "Security Framework for Wireless Sensor Networks", Wireless Personal Communications, vol. 37, **(2006)**, pp. 455-469.
[7] M. Kim, H. Kim and S. Lee, "User Authentication for Hiererchical Wireless Sensor Networks", Proc. of the 14th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, **(2013)**, pp. 203-208.
[8] H. Kim, "Non-interactive hierarchical key agreement protocol over hierarchical wireless sensor networks", Communications in Computer and Information Science, vol. 339, no. 5, **(2012)**, pp. 86-93.
[9] H. Kim, "Efficient and Non-Interactive Hierarchical Key Agreement in WSNs", International Journal of Security and Its Applications, vol. 7, no. 2, **(2013)**, pp. 159-170.
[10] H. Guo, Y. Mu, Z. Lin and X. Zhang, "An efficient and non-interactive hierarchical key agreement protocol", Computers & Security, vol. 30, **(2011)**, pp. 28-34.
[11] R. Gennaro, S. Halevi, H. Krawczyk, T. Rabin, S. Reidt and S. D. Wolthusen, "Strongly-resilient and non-interactive hierarchical key-agreement in MANETs", Lecture Notes in Computer Science, vol. 5283, **(2008)**, pp. 49-65.
[12] R. Rosli, Y. M. Yusoff and H. Hashim, "A review on pairing based cryptography in wireless sensor networks", Proc. of 2011 IEEE Symposium on Wireless Technology and Applications, **(2011)**, pp. 48-51.
[13] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing", Lecture Notes in Computer Science, vol. 2139, **(2001)**, pp. 213-229.
[14] R. Sakai, K. Ohgishi and M. Kasahara, "Cryptosystems based on pairings", Proc. of Symposium on Cryptography and Information Security 2000, **(2000)**.
[15] H. Kim, "Privacy Preserving Security Framework for Cognitive Radio Networks", IETE Technical Review, vol. 30, no. 2, **(2013)**, pp. 17-24.

[16] H. Kim, "Location-based authentication protocol for first cognitive radio networking standard", Journal of Network and Computer Applications, vol. 34, **(2011)**, pp. 1160-1167.

Authors

Sung-Woon Lee, he is a professor at the Department of Information Security, Tongmyong University, Korea. He received the B.S. and M.S. degrees in Computer Science from Chonnam National University, Korea in 1994 and 1996, respectively, and the Ph.D. degree in Computer Engineering from Kyungpook National University, Korea, in 2005. He was with the Korea Information System as a researcher, Korea, from 1996 to 2000. His research interests include cryptography, network security, and security protocol.

Hyunsung Kim, he is an associate professor at the Department of Cyber Security, Kyungil University, Korea from 2012. He received the M.S. and Ph.D. degrees in Computer Engineering from Kyungpook National University, Republic of Korea, in 1998 and 2002, respectively. From 2000 to 2002, he worked as a senior researcher at Ditto Technology. He had been an associate professor from 2002 to 2012 with the Department of Computer Engineering, Kyungil University. His research interests include cryptography, crypto hardware design, authentication technologies, network security and ubiquitous computing security.