Efficient and Non-Interactive Hierarchical Key Agreement in WSNs

Hyunsung Kim

Dept. of Cyber Security, Kyungil University
Kyungsan, Kyungbuk 712-701, Korea
kim@kiu.ac.kr

Abstract

Wireless sensor networks (WSNs) have many applications, vary in size, and are deployed in a wide variety of areas. They are often deployed in potentially adverse or even hostile environment so that there are concerns on security issues in these WSNs. Sensor nodes used are resource-constrained, which make security applications a challenging problem. Key agreement is a fundamental security service in WSNs; it 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, Guo et al. proposed an efficient and non-interactive hierarchical key agreement protocol applicable to mobile ad-hoc networks, which has good properties including non-interactive, hierarchical, resilient, etc. The purpose of this paper is to propose a non-interactive hierarchical key agreement protocol over the hierarchical WSNs, which is a revision of Guo et al.'s protocol for the WSNs due to their protocol's good properties. Our revision inherit advantages from Guo et al.'s protocol and well suited to the hierarchical WSNs.

Keywords: *Wireless Sensor Network Security, Hierarchical Key Agreement, Non-interactive Protocol, Identity-based Encryption*

1. Introduction

Wireless sensor networks (WSNs) have received a lot of attention recently, particularly with the proliferation of the advances in wireless communications and electronics [1-4]. Example applications include military target tracking and surveillance [5-6], natural disaster relief [7], biomedical health monitoring [8-9], and hazardous environment exploration and seismic sensing [10]. In military target tracking and surveillance, a WSN can assist in intrusion detection and identification. Specific examples include spatially-correlated and coordinated troop and tank movements. With natural disasters, sensor nodes can sense and detect the environment to forecast disasters before they occur. In biomedical applications, surgical implants of sensors can help monitor a patient's health. For seismic sensing, ad hoc deployment of sensors along the volcanic area can detect the development of earthquakes and eruptions.

Smart sensor nodes are low power devices equipped with one or more sensors, a processor, memory, a power supply, a radio, and an actuator. A variety of mechanical, thermal, biological, optical, and magnetic sensors may be attached to the sensor node to measure properties of the environment. Since the sensor nodes have limited memory and are typically deployed in difficult to access locations, a radio is implemented for wireless communication to transfer the data to a base station [1].

Unlike traditional networks, a WSN has its own design and resource constraints. Especially, sensor nodes used to form these networks are resource-constrained, which make security applications a challenging problem. It is thus, more than obvious that the diversified environments necessitate the employment of security mechanisms at different levels. In summary as from [11], the most important reasons for that are the following

- The computation power available in embedded systems is limited and may be insufficient for the processing of security algorithms,
- The battery capacity is also limited and their life duration is strongly connected to the quantity of computation executed in the embedded processor,
- The quantity of storage is limited also,
- WSN systems can be accessed more easily than fix systems by attackers. Indeed they must be secure against malicious entities,
- The range of attack increases very quickly so at the same time the WSN must be flexible enough to support the rapid evolution of security mechanisms and standards.

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, pairwise 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 pairwise key establishment techniques such as public key cryptography and key distribution center.

To enable legitimate nodes to communicate securely, shard secret information is needed. Without such information, attacker is unable to take any useful information from the network messages encrypted with the key. The goal is to provide this advantage to legitimate nodes in the WSNs [12]. However, the question is: How can secrets be established if an adversary can eavesdrop on every message exchange? Many solutions to this question are given, but most of them consider the case of wired networks, which requires computationally strong devices. One possibility of key management is the use of public key cryptography as it makes distribution of keys easy [13-16]. Since the introduction of the bilinear pairing by Boneh and Franklin, pairing has become one of the most attractive topics in cryptographic research [17]. In the context of WSNs, pairing has been proven as practical in resource-constrained platforms with its efficient in key management, strong security aspect, less memory requirement for key storing and economy communication overhead [16].

Recently, Guo, *et al.*, proposed an efficient and non-interactive hierarchical key agreement protocol applicable to mobile ad-hoc networks [18]. Guo, *et al.*'s protocol is based on the pairing cryptography and satisfies the desired properties mentioned in [19] 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, this paper proposes a non-interactive hierarchical key agreement protocol, named as HKAP, over the hierarchical WSNs, which is a revised version of Guo, *et al.*'s protocol for the WSNs and extended version of a paper that was presented at SecTech 2012 [20]. To devise a new protocol, we first design a naïve HKAP to support features of non-interactive, hierarchical and resilient. We further design a privacy supporting HKAP based on the naïve one, which supports anonymity.

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 revision could support security and robustness over the hierarchical WSNs.

This paper is organized as follows. Section 2 reviews bilinear pairings and Sakai, *et al.*'s non-interactive identity-based key agreement protocol. Two new HKAPs are proposed over the hierarchical WSNs in Section 3. Security analyses are provided in Section 4. Finally, Section 5 gives a brief conclusion.

2. Related Works

This section introduces the background used in our protocol. We give basic definition and properties of bilinear pairings and Sakai et al.'s non-interactive identity-based key agreement protocol in [21].

2.1. Bilinear Map and Security Assumption

Bilinear map: The admissible bilinear map $\hat{e}$ is defined over two groups of the same prime order p denoted by G and G_T in which the computational Diffie-Hellman problem is hard. More formally, we have the following definition:

Definition 1: Let G is an additive group of prime order q and G_T a multiplicative group of the same order. Let P denote a generator of G. An admissible pairing is a bilinear map $\hat{e}: G \times G \rightarrow G_T$ which has the following properties :

- Bilinear : given $Q, R \in G$ and $a, b \in Z^*_q$, we have $\hat{e}(aQ, bR) = \hat{e}(Q, R)^{ab}$
- Non-degenerate : $\hat{e}(P, P) \neq 1_{G_T}$
- Computable : $\hat{e}$ is efficiently computable

G is a subgroup of the group of points on an elliptic curve over a finite field. G_T is a subgroup of a multiplicative group of a related finite field. Typically, the map $\hat{e}$ can be derived from either the Weil pairing or Tate pairing on an elliptic curve over a finite field. The computational effort of the Tate pairing is less than the Weil pairing. A more comprehensive description of how these groups, pairings, and other parameters should be selected in practice for efficiency and security can be found in [17]. Throughout this paper, we will simply use the term "bilinear map" to refer to the admissible bilinear map.

2.2. Non-interactive Identity-based Key Agreement

Sakai, *et al.*, proposed a non-interactive (but not hierarchical) identity-based key agreement scheme [21]. In Sakai, *et al.*'s scheme, the central authority firstly chooses two cyclic groups G and G_T and the bilinear map $\hat{e}: G \times G \rightarrow G_T$ to setup the parameters for an identity-based public key system. Moreover, it chooses a cryptographic hash function $H: \{0,1\}^* \rightarrow G$. It then chooses a secret key $s \in Z_q$ and generates the secret key $S_{ID} = sH(ID) \in G$ for a node with identity ID.

Suppose two nodes with identities ID_1 and ID_2 want to establish a shared secret key. The shared key between them is $K = \hat{e}(H(ID_1), H(ID_2))^s \in G_T$, which party ID_1 computes as $K = \hat{e}(S_{ID_1}, H(ID_2))$ and ID_2 computes as $K = \hat{e}(H(ID_1), S_{ID_2})$. The security of this scheme

can be reduced to the decisional bilinear Diffie-Hellman assumption in the random-oracle model.

3. Non-interactive Hierarchical Key Agreement Protocol

This section proposes a non-interactive hierarchical key agreement protocol, named as HKAP, using pairings over the hierarchical WSNs, which is a revision of Guo, *et al.*'s protocol for the WSNs. The HKAP falls 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 establishing a secure session key between any two nodes in the hierarchical WSN.

It is assumed in the proposed protocol 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 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.

The goal of the HKAP over the hierarchical WSNs is to supporting the following features mentioned in [19] for the WSNs.

- Non-interactive: any two nodes can compute a unique shared secret key without an interaction. This property is used to save bandwidth
- Identity-based: each node computes the shared secret key only using its own secret key and the identity of its peer. This property is used to avoid coordination and therefore support ad-hoc communication
- Hierarchical: the scheme is decentralized through a hierarchy where intermediate nodes in the hierarchy can derive the secret keys for each of its child nodes. This property is used to allow flexible provisioning of nodes
- Resilient: the scheme is fully resilient against any number of malicious nodes

To design a protocol with supporting the above mentioned features, we propose a Na ̈ve HKAP. However, probably the most important problem in ubiquitous computing environments poses serious privacy risks. By watching everything a node does, these systems have the potential to leak all our actions, preferences, and locations to others unknown to us, now or in the future. Privacy could be defined as following feature, which is focused on only anonymity.

- Privacy: The identity of the origin and/or the destination of a conversation is hidden from adversaries unless it is intentionally disclosed by the user. Anonymity impacts location privacy, because as long as a user is anonymous, location privacy is preserved. Anonymity mechanisms should allow the node to use the network services while protecting the identity or other identification information from possible abuse. For keeping the node anonymous, there should not be possibility to link any parameters of the node identity with any context-based information.

To support the privacy issue, we further propose a privacy supporting HKAP by using amplified identities not using real identities of nodes.

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. 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
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

Hierarchical Key Settlement Phase: This phase is based on the ID-based cryptosystem, which uses identities of each node for their public key. The main goal of 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 an administrator, which starts key generation and setup with it's private key set (S_1, S_2, S_3). We assume that each entity has a different role in a network, which 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. In the hierarchical key settlement phase, we distinguish the privilege of each entity with the possession of how many elements of private key the entity has from the sink's key set. Figure 1 shows a procedure of a hierarchical key settlement for the hierarchical WSN. To set up keys, the protocol 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 $ID_S \cdot S_1$. 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.

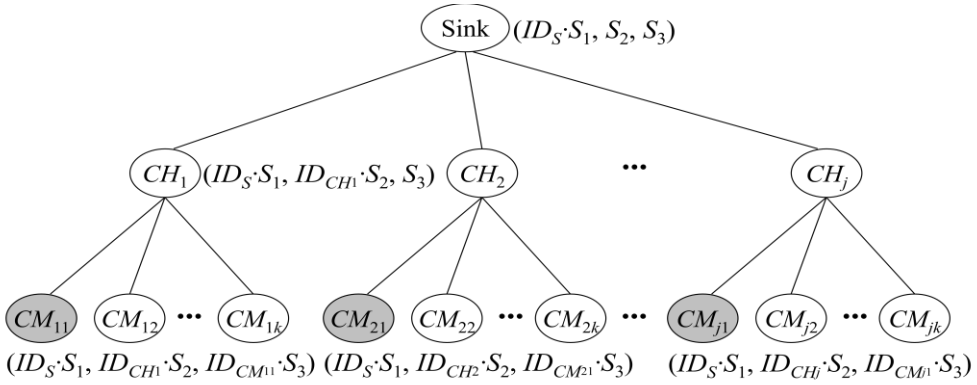

Figure 1. Hierarchical Key Settlement Model for Naïve HKAP

The proposed key settlement phase for the naïve HKAP has a good advantage in the perspective of that the leaf nodes in Figure 1 does not need any other information to set up session keys with their ancestor nodes in the tree because they already have the required information. However, the other cases of two communication parties in the outside of their hierarchy in the tree require additional information to set up session keys, *i.e.*, the case between CM_{11} and CM_{21} in Figure 1. They should have to know the amplified IDs, which are considered as the public keys on ID-based cryptosystem, of their counterparts in different hierarchy. However, we could easily assume that each party could exchange the amplified IDs in their hierarchy with others before they are deployed because of the basic nature of public key itself. Thereby, they do not need to communicate with the counterpart node in their hierarchy to set up the session key. It is a very important aspect in WSN due to their limitations.

Session Key Agreement and Secure Communication Phase: The purpose of this phase is to establish a secure channel by establishing a secure 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 $MAC = h(sk \| \text{the encrypted data packet})$ 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*.

The proposed session key agreement and secure communication phase has a good advantage in the perspective of that node does not need any other communication to agree on a session key.

3.2. Privacy Supporting HKAP

To support the privacy issue, this subsection further proposes a privacy supporting HKAP 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 Supporting HKAP are very similar with the Naïve one. Figure 2 shows a procedure of a hierarchical key settlement for the privacy supporting HKAP. To set up keys, the protocol 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.

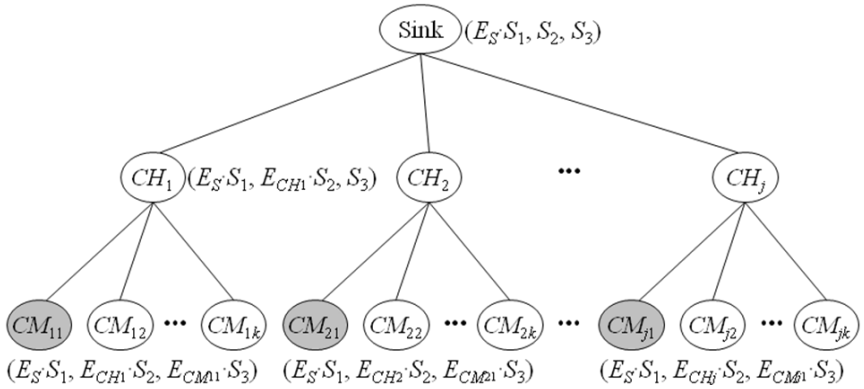

Figure 2. Hierarchical Key Settlement Model for Privacy Supporting HKAP

The proposed key settlement phase has a good advantage in the perspective of that the leaf nodes in Figure 2 does not need any other information to set up session keys with their ancestor nodes in the tree because they already have the required information.

However, the other cases of two communication parties in the outside of their hierarchy in the tree require additional information to set up session keys, i.e. the case between CM_{11} and CM_{21} in Figure 1. They should have to know the amplified IDs, which are considered as the public keys on ID-based cryptosystem, of their counterparts in different hierarchy. However, we could easily assume that each party could exchange the amplified IDs in their hierarchy with others before they are deployed because of the basic nature of public key itself. Thereby, they do not need to communicate with the counterpart node in their hierarchy to set up the session key. It is a very important aspect in WSN due to their limitations.

Session Key Agreement and Secure Communication Phase: The purpose of this phase is to establish a secure channel by establishing a secure 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 ($E_S \cdot S_1$, $E_{CH_i} \cdot S_2$, $E_{CM_{ij}} \cdot S_3$) computes $sk = \hat{e}(E_S \cdot S_1, E_S') \cdot \hat{e}(E_{CH_i} \cdot S_2, E_{CH_k}') \cdot \hat{e}(E_{CM_{ij}} \cdot S_3, E_{CM_{kl}}')$ by using the amplified ID set of the counterpart node CM_{kl}, which is $\{E_S', E_{CH_k}', E_{CM_{kl}}'\}$. Independent with CM_{ij}, CM_{kl} with it's private key set ($E_S \cdot S_1$, $E_{CH_k} \cdot S_2$, $E_{CM_{kl}} \cdot S_3$) computes $sk = \hat{e}(E_S \cdot S_1, E_S') \cdot \hat{e}(E_{CH_k} \cdot S_2, E_{CH_i}') \cdot \hat{e}(E_{CM_{kl}} \cdot S_3, E_{CM_{ij}}')$ by using the amplified ID set of the counterpart node CM_{ij}, which is $\{E_S', E_{CH_i}', E_{CM_{ij}}'\}$. CM_{ij} and CM_{kl} can compute the same shared session key due to $sk = \hat{e}(E_S \cdot S_1, E_S) \cdot \hat{e}(E_{CH_i} \cdot S_2, E_{CH_k}) \cdot \hat{e}(E_{CM_{ij}} \cdot S_3, E_{CM_{kl}}) = \hat{e}(E_S \cdot S_1, E_S) \cdot \hat{e}(E_{CH_k} \cdot S_2, E_{CH_i}) \cdot \hat{e}(E_{CM_{kl}} \cdot S_3, E_{CM_{ij}}) = \hat{e}(E_S, E_S)^{S_1} \cdot \hat{e}(E_{CH_k}, E_{CH_i})^{S_2} \cdot \hat{e}(E_{CM_{kl}}, E_{CM_{ij}})^{S_3} = \hat{e}(E_S, E_S)^{S_1} \cdot \hat{e}(E_{CH_i}, E_{CH_k})^{S_2} \cdot \hat{e}(E_{CM_{ij}}, E_{CM_{kl}})^{S_3}$.

Step 2. The source node CM_{ij} sends an encrypted data packet and the message digest $MAC = h(sk \| \text{the encrypted data packet})$ 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.

The proposed session key agreement and secure communication phase has a good advantage in the perspective of that node does not need any other communication to agree on a session key.

4. Analyses

This section provides analyses mainly focused on the security concerns only for the privacy supporting HKAP. Although it is important to provide a formal security proof on any cryptographic protocols, the formal security proof of the protocols remains one of the most challenging issues for cryptography research. Until now, a simple, efficient and convincing formal methodology for correctness analysis on security protocols is still an important subject of research and an open problem. Because of these reasons, most security protocols have been demonstrated with a simple proof. Therefore, we follow the approaches used in [22] for comparison purpose. This section overviews the computational problems that the privacy supporting HKAP is based on and then gives 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 [17]. 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 supporting HKAP including passive and active attacks as follows.

Proposition 1. The proposed privacy supporting HKAP is secure against passive and active attacks.

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 ID 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 supporting HKAP is secure against passive attack.

Proposition 2. The proposed privacy supporting HKAP 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 supporting HKAP is secure against active attack.

4.3. Performance Analyses

Key management demands extra storage space to store the required keys for the secure communication. The performance comparison 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 feature comparisons between the proposed privacy supporting HKAP and Guo, *et al.*'s protocol.

Table 2. Comparisons between Related Protocols

Features / Protocols	Size of key set	Number of pairing operations
Guo et al.'s	log *n*	log *n*
Privacy supporting HKAP	Constant	Constant

There is an increase in the size of key set in Guo, *et al.*'s protocol if the number of nodes in the network denoted as *n* in Table 2 is increased. However, the proposed privacy supporting HKAP keeps the same size of key set with constant of 3 even if it varies. Furthermore, the computational overhead does affect to the size of key set it Guo, *et al.*'s protocol, which is very important property especially in WSN with the limits of battery life or resource constraints.

5. Conclusion

To enable legitimate nodes to communicate securely, shard secret information is needed. Without such information, attacker is unable to take any useful information from the network messages encrypted with the key. However, the question is: How can secrets be established if an adversary can eavesdrop on every message exchange? Many solutions to this question are given, but most of them consider the case of wired networks, which requires computationally strong devices. 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, this paper proposed the privacy supporting non-interactive hierarchical key agreement protocol over the hierarchical WSNs, which is a revised version of Guo, *et al.*'s protocol for the WSNs. The proposed protocol is secure against the corruption of any number of nodes at any level in the hierarchy. Compared with other existing protocols, the proposed protocol offers much better performance on the bandwidth consumption, the computational cost, and the storage cost.

Acknowledgements

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

References

[1] J. Yick, B. Mukherjee and D. Ghosal, "Wireless sensor network survey", Computer Networks, vol. 52, **(2008)**, pp. 2292–2330.
[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] T. Arampatzis, J. Lygeros and S. Manesis, "A survey of applications of wireless sensor and wireless sensor networks", Proceedings of IEEE International Symposium on Mediterrean Conference on Control and Automation, **(2005)**, pp. 719-724.
[4] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks", IEEE Communications Magazine, vol. 40, no. 8, **(2002)**, pp. 104–112.
[5] A. Dukeman and M. Anderson, "A CSP solution to multi-camera surveillance and target tracking", Proc. of 2011 IEEE International Conference on Systems, Man, and Cybernetics, **(2011)**, pp. 97-102.
[6] P. Skoglar, "Tracking and Planning for Surveillance Applications", Linköping University, Thesis, **(2012)**.
[7] E. Cayirci and T. Coplu, "SENDROM: Sensor Networks for Disaster Relief Operations Management", Journal of Wireless Networks, vol. 13, no. 3, **(2007)**, pp. 409-423.
[8] P. Kumar, S. -G. Lee and H. -J. Lee, "E-SAP: Efficient-Strong Authentication Protocol for Healthcare Applications Using Wireless Medical Sensor Networks", Sensors, vol. 12, **(2012)**, pp. 1625-1647.
[9] 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.
[10] F. Ehlers, D. Sofge, M. Chitre and J. Potter, "Distributed Mobile Sensor Networks for hazardous Applications", International Journal of Distributed Sensor networks, vol. 2012, **(2012)**, pp. 1-3.
[11] N. R. Prasad and M. Alam, "Security Framework for Wireless Sensor Networks", Wireless Personal Communications, vol. 37, **(2006)**, pp. 455–469.
[12] M. Tiwari, K. V. Arya, R. Choudhari and K. S. Choudhary, "Designing Intrusion Detection to Detect Black Hole and Selective Forwarding Attack in WSN Based on Local Information", Proc. of 2009 Fourth International Conference on Computer Sciences and Convergence Information Technology, **(2009)**, pp. 824-828.
[13] B. Kadri, D. Moussaoui, M. Feham and A. Mhammed, "An efficient key management scheme for hierarchical wireless sensor networks", Wireless Sensor Network, vol. 4, **(2012)**, pp. 155-161.
[14] L. B. Oliveira, R. Dahab, J. Lpez, F. Daguano and A. A. F. Loureiro, "Identity-based encryption for sensor networks", Proc. of 5th IEEE International Conference on Pervasive Computing and Communications Workshops, **(2007)**, pp. 290-294.
[15] P. Kalyani and C. Chellappan, "Enhanced RSACRT for energy efficient authentication to wireless sensor networks security", American Journal of Applied Sciences, vol. 9, no. 10, **(2012)**, pp. 1660-1667.
[16] 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.
[17] D. Boneh and M. Franklin, "Identity-based encryption from the weil pairing", Lecture Notes in Computer Science, vol. 2139, **(2001)**, pp. 213-229.
[18] 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.

[19] 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.
[20] 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.
[21] R. Sakai, K. Ohgishi and M. Kasahara, "Cryptosystems based on pairings", Proc. of Symposium on Cryptography and Information Security 2000, **(2000)**.
[22] H. S. Kim, "Location-based authentication protocol for first cognitive radio networking standard", Journal of Network and Computer Applications, vol. 34, **(2011)**, pp. 1160-1167.

Author

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, VLSI, authentication technologies, network security and ubiquitous computing security.