석사학위논문

Master's Thesis

무선 센서 네트워크에서 안전한 센서 노드의 클러스터 형성 기법 연구

A Study on a Secure Clustering Scheme in Wireless Sensor Network

유명한 (柳명한 Yoo, Myunghan)

정보통신공학과

Department of Information and Communications Engineering

한국과학기술원

Korea Advanced Institute of Science and Technology

2010

무선 센서 네트워크에서 안전한 센서 노드의 클러스터 형성 기법 연구

A Study on a Secure Clustering Scheme in Wireless Sensor Network

A Study on a Secure Clustering Scheme in Wireless Sensor Network

Advisor : Professor Kim, Kwangjo

by

Yoo, Myunghan
Department of Information and Communications Engineering
Korea Advanced Institute of Science and Technology

A thesis submitted to the faculty of the Korea Advanced Institute of Science and Technology in partial fulfillment of the requirements for the degree of Master of Engineering in the Department of Information and Communications Engineering

Daejeon, Korea
2009. 12. 18.
Approved by

Professor Kim, Kwangjo
Advisor

무선 센서 네트워크에서 안전한 센서 노드의 클러스터 형성 기법 연구

유명한

위 논문은 한국과학기술원 석사학위논문으로 학위논문심사위원회에서 심사 통과하였음.

2009년 12월 18일

심사위원장 김광조 (인)

심사위원 김명철 (인)

심사위원 이병천 (인)

MICE 20084239

유 명 한. Yoo, Myunghan. A Study on a Secure Clustering Scheme in Wireless Sensor Network. 무선 센서 네트워크에서 안전한 센서 노드의 클러스터 형성 기법 연구. Department of Information and Communications Engineering . 2010. 25p. Advisor Prof. Kim, Kwangjo. Text in English.

Abstract

Wireless Sensor Network (WSN) is considered to be as one of the fundamental technologies for building ubiquitous computing environment. Since WSN consists of many sensor nodes with limited resources (*i.e.*, computation, storage and battery), security primitives can be hard to be implemented. Thus, the network can be vulnerable to various attacks (*i.e.*, denial-of-service attack, sinkhole/wormhole/sybil attack, message forgery attack, *etc.*).

HPEQ (Hierarchical Periodic, Event-driven and Query-based) protocol was purposed for surveillance of emergency events. In this routing protocol, clustering method is originated from LEACH, which is one of the most cited researches for cluster-based WSN. However, not only inherent vulnerabilities exist in HPEQ protocol, due to no provision of security requirements such as confidentiality, integrity of data packets, authentication of sensor nodes, *etc.*, but also new vulnerabilities may occur, by the modification of aggregator selection from LEACH. Several secure variants of LEACH such as SecLEACH, GS-LEACH, *etc.* were suggested. However, these schemes still have some vulnerabilities. Therefore, we present a secure clustering scheme guaranteeing authentication of sensor nodes that are members of a cluster and confidentiality of communication data and the cluster topology.

Contents

List of Tables

List of Figures

1. Introduction

We expect that Wireless Sensor Network (WSN), which is considered to be as one of the most promising technologies for upcoming ubiquitous society, can help us not only in the ordinary life, but also in the severe environment, where human-being cannot enter or need to observe for a long time. Therefore, various researches [2], [3], [4], [6], [13], [14], [15], [16], *etc.*, were presented.

HPEQ (Hierarchical Periodic, Event-driven and Query-based) protocol [4] is believed to be a useful and efficient protocol to monitor wide and dangerous areas. The main objective of HPEQ protocol is to observe critical and physical environments such as fire on a building, leaking of toxic gases, explosions, even military battle field, *etc.* Thus, reliably capturing an event and transmitting a captured event to the sink are important.

However some critical security vulnerabilities are caused by the naive clustering scheme and the data report. During cluster selection, an adversarial node outside a network can join the process and can be an aggregator which is responsible for aggregating sensed data and reporting events occurring in a cluster to the sink. And during the reporting process of the critical event, an adversarial node outside a network can capture the message and modify the message including that an event does not occur.

Thus, we address a secure clustering scheme that provides authentication of all cluster member nodes as well as aggregator, integrity, confidentiality, and freshness of each message. The proposed scheme requires that each node has only two embedded keys and one *Credential*, but while sensor nodes make a cluster, the inspector node, which is responsible for observing misbehavior of the aggregator, should request the sink to authenticate the aggregator and cluster member nodes. It causes additional communication overhead. However we show that some trade-off between the level of security and overhead is required.

The rest of this thesis is organized as follows: In Chapter 2, we examine related work which can be used for critical condition monitoring applications and show the reason why we choose HPEQ protocol. In Chapter 3, we will describe our scheme in detail. We evaluate our scheme from the point of security in Chapter 4. Then, in Chapter 5, we examine the overhead of our scheme. Finally, we will make conclusion and suggest future work.

2. Related work

2.1 Routing Protocol

We need to define requirements of protocols for critical application. According to [4], monitoring critical conditions has the following requirements, simultaneously: periodic, event-driven and query-based reports from sensor nodes to a sink. We believe that people are wondering query-based report requirement. Query-based report requirement needs fast path establishment to subscribe the current situation when an event occurs. For example, a fire breaks out in a building. To rescue people in the urgent situation, a rescue corps needs to know where people are located in a building. Low latency for event delivery and reliability are also important requirements.

Meeting these requirements is quite difficult, due to conflict of requirements. For example, in Directed Diffusion paradigm [8], to mitigate node failures caused by sending packets on a path, transmissions are performed through multi-path, which is probabilistically chosen. However, using multi-path may cause more energy dissipation and packet collisions. In PFR protocol [6], a source node forwards packets to the sink through nodes, in virtually connected zone which is constructed to propagate the data to the sink. The node energy dissipation and cost can be increased by estimating direction of a received packets, since the node has to equip magnetometer module. SW-PFR [16] extended version of PFR uses sleep-awake duration for energy savings. Variable Transmission Range Protocol (VRTP) [2] tries to solve the problems of fault tolerance and energy efficiency by diversifying the range of the data transmission. Network lifetime, then, is prolonged since the nodes away one hop from the sink can sleep. On the other hand, an additional hardware component is needed.

SPIN (SPMS) [13] also focuses on node failures. In this protocol, meta-data exchange is used before data transmissions. SPMS requests and transmits data through the shortest multi-hop path to reduce energy costs and end-to-end delay. The mechanism for dealing with fault-tolerance keeps the shortest and the second shortest paths in the routing table. When sensing node failures in the shortest path, a sender will choose the second shortest path. However, in a huge disaster (*i.e.* explosion), a large number of sensor nodes can be destructed including nodes on the second path.

PEQ [3] builds the shortest path for low latency for event delivery. This protocol uses

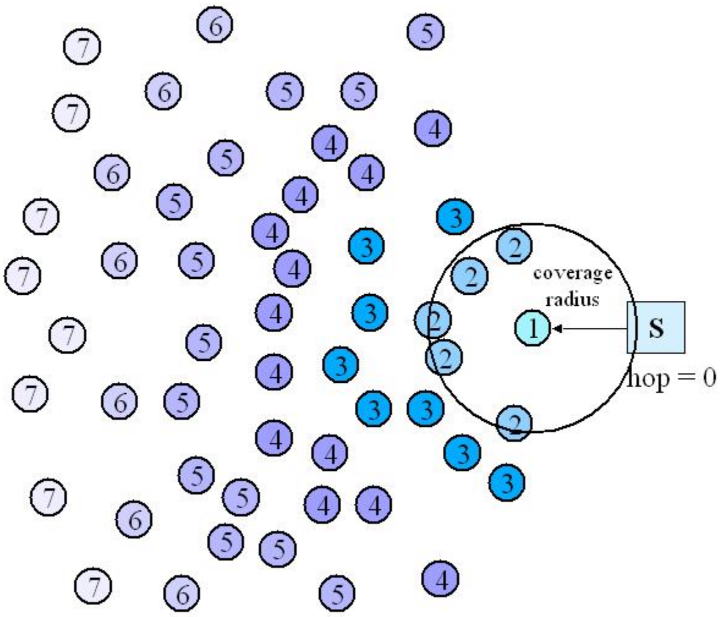


Figure 2.1: PEQ Protocol

the hop count metric for routing mechanism which requires a small amount of information (Figure 2.1). When an event occurs, PEQ utilizes three ways: broadcasting a message to intermediates of a source node to find paths, receiving response including the hop level and identification of the intermediates. As other hop nodes, a node in lower the hop level will be designated. This mechanism is also useful to avoid loop formation. Simulation results show quite good performance of PEQ in terms of delay and delivery ratio. HPEQ [4] is a hierarchical version of PEQ. HPEQ shows more uniform load balance, lower latency and higher delivery ratio than PEQ, by aggregating data from clusters. Several hierarchical protocols exist: APTEEN [15], PEGASIS [14], and Energy-Aware Routing for Cluster-Based Sensor Network [19] which are good solutions for energy efficiency and latency, but complicated. The clustering mechanism of HPEQ is inspired from LEACH [9]. In LEACH, an aggregator is selected based on probabilistic threshold and normal nodes select their aggregator based on signal strength. However, in HPEQ, a selected node based on probabilistic threshold is just a candidate that can be an aggregator, broadcasts a messages to request remaining energy to neighbors. Then, the neighbor replies with their identification and the remaining amount of energy. Finally, a node which has the highest level of energy will be assigned as an aggregator. In LEACH, the communication with the sink is performed by only aggregators with one hop. However, direct communication between the sink and aggregators can cover the small scale of WSN. On the other hands, HPEQ supports multi-hop communication between an aggregator and the sink.

However, due to the naive aggregator selection, clustering and transmission, HPEQ causes several critical security vulnerabilities. From the communication point of view, HPEQ does not guarantee the confidentiality and the integrity of each message. Thus, anybody can eavesdrop and modify every message. For example, when an event, such as fire and appearance of enemies in the battle field, occurs, an adversary can change the message reporting an event to the sink. And, without any compromised node, an adversarial node outside a network which has abundant computational and communicational resources can join a cluster even as an aggregator by overstating the remaining amount of energy in the aggregator selection step. Then, the node selected with probability threshold will appoint the adversarial node outside a network as the cluster aggregator. The adversarial node, then, can selectively transmit messages by dropping messages.

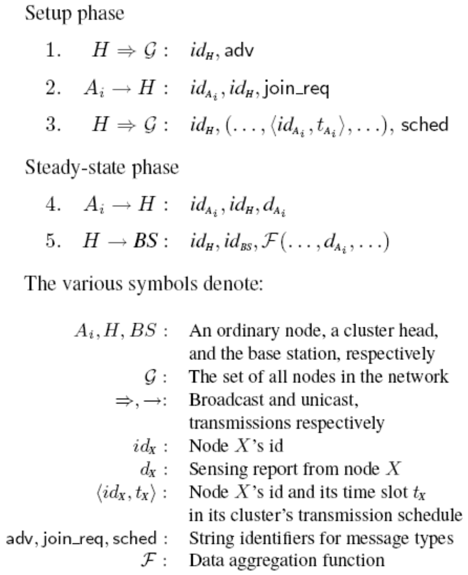
2.2 Clustering Scheme

LEACH [9] is designed to prolong life time of the whole network by distributing energy consumption. LEACH protocol has some assumptions that sensor nodes are uniformly spread over fields and all sensor nodes have enough power to transmit to Base Station(BS) directly.

To reduce energy consumption, most sensor nodes send their sensing data to their aggregator. Aggregators will use data fusion function and transmit fused data to BS. Obviously, aggregators have more energy since, normally, the distance between BS and an aggregator is farther than the distance between an aggregator and a sensor node. Thus, for balancing energy consumption in a whole network, aggregators are randomly selected in each round. Sensor nodes select a node as an aggregator. An aggregator broadcasts to ask neighbors to join their cluster. Other sensor nodes select their aggregator by measuring signal strength from aggregators.

Each round of LEACH is as shown on Figure 2.2. All sensor nodes know when each round starts through time synchronization among all sensor nodes.

LEACH protocol is not considered to be secure. As the result, LEACH protocol is vulnerable to simple attacks, *i.e.*, watching and modifying important messages. Thus, Oliveira *et al.* suggested SecLEACH [17] by combining Eschenauer *et al.*'s random key pre-distribution and LEACH.



Setup phase

1. $H \Rightarrow \mathcal{G}:\ id_H, \mathsf{adv}$
2. $A_i \rightarrow H:\ id_{A_i}, id_H, \mathsf{join\_req}$
3. $H \Rightarrow \mathcal{G}:\ id_H, (\ldots, \langle id_{A_i}, t_{A_i} \rangle, \ldots), \mathsf{sched}$

Steady-state phase

4. $A_i \rightarrow H:\ id_{A_i}, id_H, d_{A_i}$
5. $H \rightarrow BS:\ id_H, id_{BS}, \mathcal{F}(\ldots, d_{A_i}, \ldots)$

The various symbols denote:

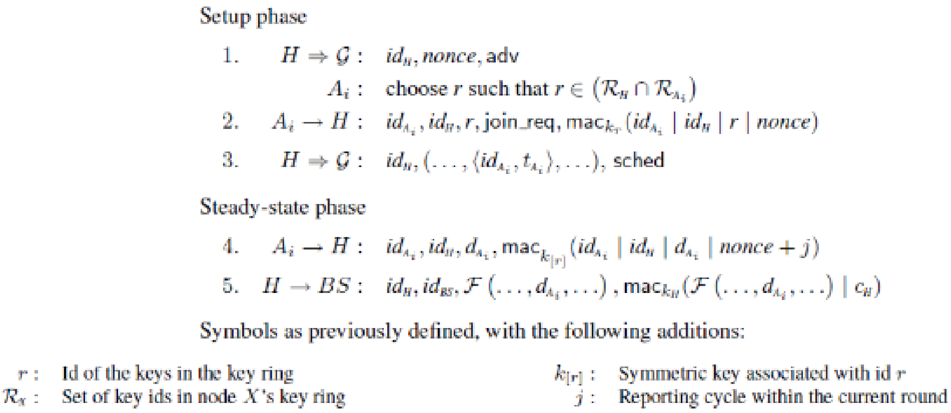
A_i, H, BS :	An ordinary node, a cluster head, and the base station, respectively
$\mathcal{G}$:	The set of all nodes in the network
$\Rightarrow, \rightarrow$:	Broadcast and unicast, transmissions respectively
id_X :	Node X's id
d_X :	Sensing report from node X
$\langle id_X, t_X \rangle$:	Node X's id and its time slot t_X in its cluster's transmission schedule
adv, join_req, sched :	String identifiers for message types
$\mathcal{F}$:	Data aggregation function

Figure 2.2: LEACH Protocol

SecLEACH have a key pool having S keys like Eschenauer *et al.*'s random key predistribution scheme [7] and make IDs of each node. Through a pseudorandom function, BS generates unique ID_X for node X. Then, by using ID_X as a seed of a pseudorandom function, BS operates a pseudorandom function m times. Then, m keys of a key pool are distributed to each node. R_X, a group of ID of keys in node X is mapped between 0 and S-1 by operating modular arithmetic. Finally, all nodes have pairwise keys with BS.

SecLEACH protocol have 5 steps, which look like LEACH protocol. In step 1, aggregator H elected through algorithm broadcasts of LEACH protocol ID of H, ID_H, and a nonce. Next, normal nodes A_i calculate key IDs of aggregator H by using ID_H and select the nearest one as their aggregator, which has shared key $k_{[r]}$. Normal nodes add Message Authentication Code (MAC) keyed with $k_{[r]}$ to a reply message and transmit to H. MAC includes nonce as well as key ID r for protection from replay attacks. In step 3, H broadcasts schedules for nodes' transmission in a round.

To protect communication between member nodes and an aggregator, member nodes' message includes MAC keyed with $k_{[r]}$. The number of report times j to H from member nodes is added to an included nonce in order to prevent replay attacks. In step 5, aggregator H sends a message and MAC keyed with shared key with BS. All steps are



Setup phase

1. $H \Rightarrow \mathcal{G}$: $id_H, nonce$, adv

A_i : choose r such that $r \in (\mathcal{R}_H \cap \mathcal{R}_{A_i})$

2. $A_i \rightarrow H$: id_{A_i}, id_H, r, join_req, $\text{mac}_{k_r}(id_{A_i} \mid id_H \mid r \mid nonce)$

3. $H \Rightarrow \mathcal{G}$: $id_H, (\ldots, \langle id_{A_i}, t_{A_i} \rangle, \ldots)$, sched

Steady-state phase

4. $A_i \rightarrow H$: $id_{A_i}, id_H, d_{A_i}, \text{mac}_{k_{[r]}}(id_{A_i} \mid id_H \mid d_{A_i} \mid nonce + j)$

5. $H \rightarrow BS$: $id_H, id_{BS}, \mathcal{F}(\ldots, d_{A_i}, \ldots), \text{mac}_{k_H}(\mathcal{F}(\ldots, d_{A_i}, \ldots) \mid c_H)$

Symbols as previously defined, with the following additions:

r :	Id of the keys in the key ring	$k_{[r]}$:	Symmetric key associated with id r
$\mathcal{R}_X$:	Set of key ids in node X's key ring	j :	Reporting cycle within the current round

Figure 2.3: SecLEACH Protocol

illustrated on Figure 2.3.

Lastly, GS-LEACH [1] was proposed as one of LEACH families in order to solve problems of SecLEACH. In SecLEACH, normal nodes have to select an aggregator, which has a shared key with normal nodes. If the nearest aggregator does not have a shared key, a normal node must select a next aggregator. Also, If and aggregator having shared key does not exist, a normal node have to transmit directly to BS even though much energy is required for transmission.

GS-LEACH has an assumption of sensor nodes distribution based on a grid. Initially, a sensing field is divided into k squares, and n sensor nodes are deployed in each square. Then, S keys pool composed of k partial groups is made. Next, m keys from a partial group of S are randomly assigned to each square. Each node has a shared key with BS.

GS-LEACH has also 5 steps for clustering and reporting to BS. But, clustering is performed only in a grid. Each grid has an aggregator. If a node, which has not a shared key with aggregator, a node sleeps during a round.

Thus, GS-LEACH provides almost same security strength as SecLEACH, but communication range between an aggregator and normal nodes is shorter than SecLEACH. A node without an aggregator can reduce energy consumption.

3. Our Proposed scheme

3.1 Design of Architecture

Before discussing our proposed scheme, we will examine the architecture of our scheme to describe simply what we have to do. We found out that the architecture of HEPQ [4] consists of three categories: initial configuration, clustering that contains both the aggregator selection and the cluster configuration, and reporting which includes data transmission to the aggregator and to the sink. We have already explained each category in Section 2.1.1.

On the other hand, the architecture of our proposed scheme consists of four parts: initial configuration, secure clustering, key management, and secure reporting. In initial configuration, firstly, the proposed scheme has a wider range than HPEQ. HPEQ only considers setting the hop count for each node. However, we have also predeployment as initial configuration. In the predeployment, each node has embedded keys and the unique *Credential* shared with the sink, we will examine in Section 4.

In the secure clustering, our scheme has two characteristics. One is that a node selected with a probability threshold based on LEACH [9] will be designated as the inspector monitoring behaviors of cluster. And the other is that all members should prove their validity to the sink.

Key management which determines the level of security uses three kinds of keys: a global key shared with all nodes in the network, a unique key for each node used to authenticate the node itself and shared with only the sink, and a cluster key shared with cluster members including the aggregator and the inspector.

In secure reporting, if a cluster is made securely, providing confidentiality and authentication of the sender is naturally feasible, due to the cluster key which shared only between the sink and cluster member nodes. Guaranteeing freshness of messages and delivery success is only needed. By the way, the original HPEQ can guarantee enough delivery ratio, even when jamming attack occurs in a way mentioned in [3]. And a nonce and addition operation can guarantee freshness of messages.

Therefore, we will mainly focus on secure clustering and key management.

Table 3.1: Notation

REQ_EN	Request of the remaining amount of energy to received node
REP_EN	Reply message to the sender node
SET_AGR	Message designating a node as the aggregator
AGR_NTF	Message encouraging nodes to join the aggregator
ID_X	ID of node X
CN, A, I, C, S	Aggregator candidate, aggregator, inspector, all cluster member nodes, the sink, respectively
$Nonce$	Randomly generated bit
CK, K_S	Cluster key and unique key of the sink, respectively
$E_{K_G}(M)$	Encrypted message M with the global key
$E_{K_X}(M)$	Encrypting message M with unique key of the node X
$Credential_X$	Pseudonym of node X, $E_{K_S}(ID_X\|\|Nonce)$
TR	The number of Transmitting and Receiving a message
$AUTH\_REQ_X$	Authentication token of node X, $Credential_X\|\|E_{K_X}(ID_X\|\|TR\|\|Nonce)$
$MAC_{K_X}(M)$	Message Authentication Code of M keyed with global or unique key of node X

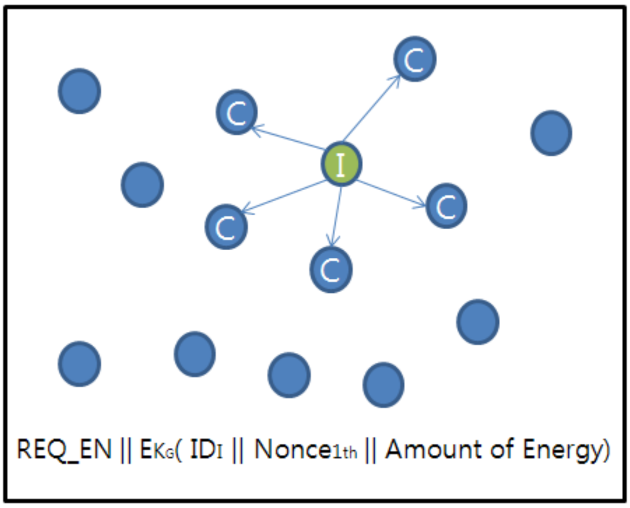

C
C
I
C
C
C
REQ_EN || E_{K_G}(ID_I || $Nonce_{1th}$ || Amount of Energy)

Figure 3.1: Inspector node broadcasting

3.2 Assumption and Notation

In this section, we will make some assumptions for the proper operation of the proposed scheme. All nodes initially have the same amount of energy resources. However, the sink has no constraint of energy resources and the computational power and is secure against an impersonation attack of an adversary and a compromising attack. Each node has two embedded keys: a global key and a unique key. The global key is shared with all the nodes deployed in the field and the sink and is used to prevent an adversarial node outside a network from joining the network. The unique key is used to authenticate the own node and to guarantee confidentiality of the encrypted message with the unique key. We also have another intrinsic assumption that cryptographic primitives such as the hash function, the encryption algorithms, *etc.* are reliable. And the last assumption is that, in the aggregator selection, probabilistically chosen inspector node has lower probability of compromising itself than nodes which are candidates of the aggregator. Finally, Table 3.1 summarizes the notations used in this paper.

3.3 Aggregator Selection

As same as the Aggregation Selection scheme of HPEQ, a node chosen with the probability threshold, which is called as the inspector node, broadcasts a message to neighbors called

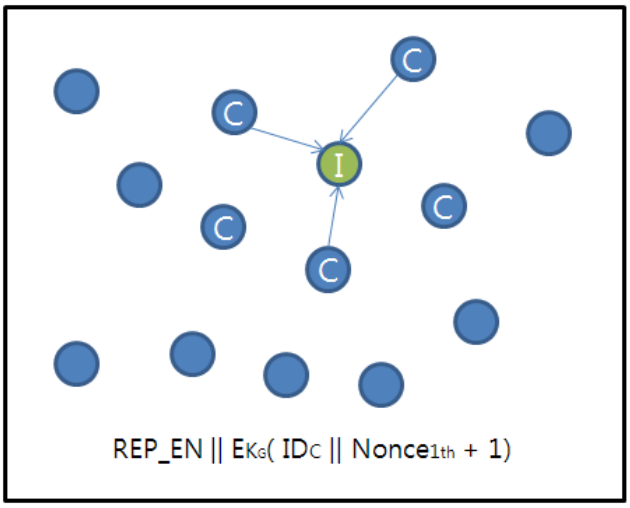


Figure 3.2: Reply of aggregator candidates

as candidates in this step. The encrypted message with global key enables only valid nodes to decrypt the message (Figure 3.1).:

(1) $REQ\_EN, E_{K_G}(ID_I||Nonce||\text{Amount of Energy})$

Candidates only which have more amount of energy answer a message (Figure 3.2).:

(2) $REP\_EN, E_{K_G}(ID_{CN}||Nonce + 1)$

$Nonce$ is added by 1 from the original nonce for freshness of sent message.

Then, the inspector node sends SET_AGR and an encrypted message with the global key including ID of the inspector node and $Nonce$ adding 2 from the original one for freshness of this message to a candidate which replies the largest amount of energy among candidates. Sent message format is (3):

(3) $SET\_AGR, E_{K_G}(ID_I||Nonce + 2)$

However, although the inspector node selects a candidate as the aggregator, we assume that the inspector node do not believe the selected aggregator yet, since an adversary can compromise a normal node and exaggeratedly inform the remaining amount of energy resources of an adversary. Thus, inspector will ask the sink to authenticate the selected aggregator, in the Cluster Configuration.

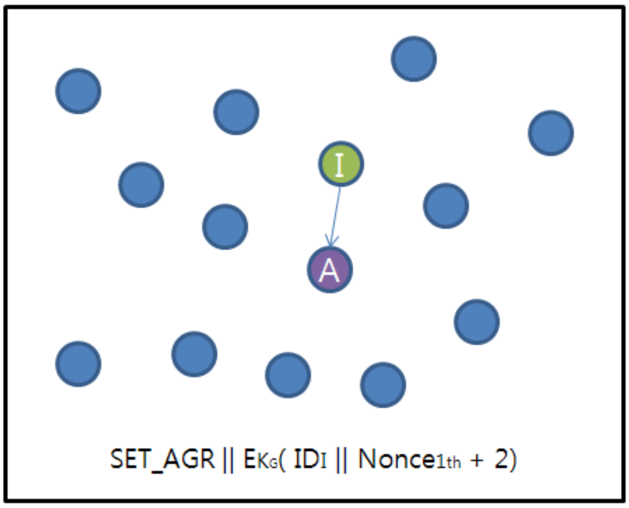


Figure 3.3: Designating aggregator

3.4 Cluster Configuration

After assigned as the aggregator, the aggregator floods a notification message and encrypted message including new $Nonce$ to guarantee freshness. Neighbors also floods the received message, recursively, until hop count becomes 0, according to HPEQ (Figure 3.4.:

(4) $AGR\_NTF, E_{K_G}(ID_A||newNonce)$

Children node receiving 0 of hop count answer with authentication token of children node (Figure 3.5). the authentication token ($AUTH\ \_REQ$) includes two essential factors: $Credential$ and TR. $Credential$ is to prevent exposure of the cluster topology from eavesdropper by encrypting ID of a node and a $Nonce$ with unique key of the sink. TR is a remaining energy metric for observation by the sink. This metric consists of 16 bit. Half bits are for transmission and the other is for receipt. If the number of communication is over 8 bit, it will be set into 0, but the sink can calculate properly.

(5) Child: $AUTH\_REQ_{child}$

Parents, receiving reply from their children, attach their authentication token, transmitting a message recursively to higher parents which sent notification message to them before (Figure 3.5).

(5) Parent: $AUTH\_REQ_P||AUTH\_REQ_{child_1}||\ldots$

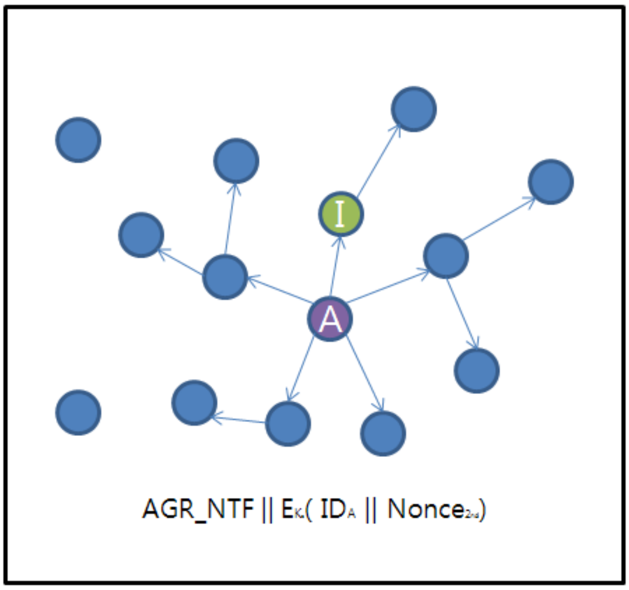


Figure 3.4: Notification of aggregator

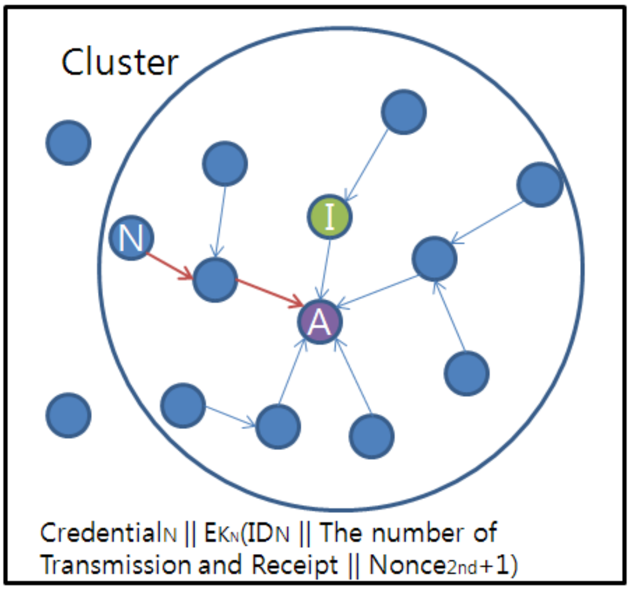

Cluster
N
I
A
$Credential_{N}$ || E_{K_N}(ID_{N} || The number of Transmission and Receipt || $Nonce_{2nd}$+1)

Figure 3.5: Joining cluster

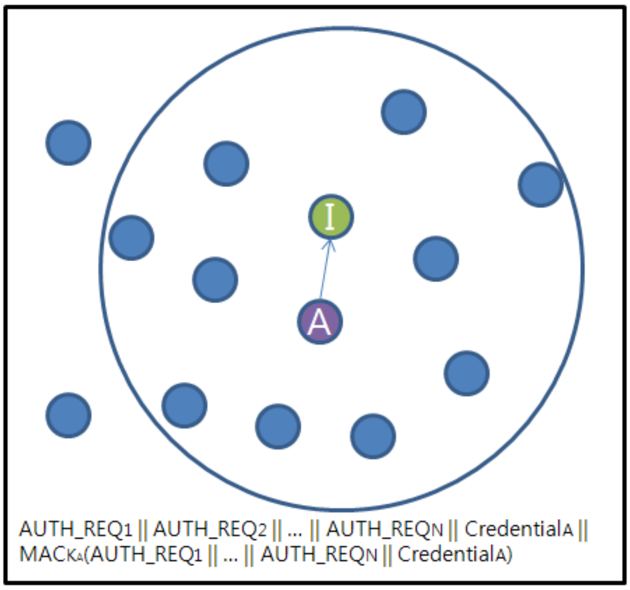

I
A
AUTH_REQ1 || AUTH_REQ2 || ... || AUTH_REQN || CredentialA ||
MACKA(AUTH_REQ1 || ... || AUTH_REQN || CredentialA)

Figure 3.6: Report to inspector node

Finally, the aggregator gathers authentication tokens.

(6) $AUTH\_REQ_1||\ldots||AUTH\_REQ_n||$
$Credential_A||MAC_{K_A}(M)$

And the aggregator reports the aggregated message, adding *Credential* of the aggregator and MAC of the message, to the inspector node (Figure 3.6). The inspector node add *REP_EN* sent by the aggregator and MAC keyed with unique key of the inspector node to the aggregated message, sending it to sink through multi-hop set in the initial configuration (Figure 3.7).

(7) $AUTH\_REQ_C||REP\_EN||$
$Credential_I||MAC_{K_I}(M)$

And then, the sink authenticate the message and if the received message is valid, the sink generates the cluster key, new nonces, and new *Credentials*, encrypts them with unique keys of nodes, and transmits them (Figure 3.8).

(8) $AUTH\_REQ_C||REP\_EN||$
$Credential_I||MAC_{K_I}(M)$

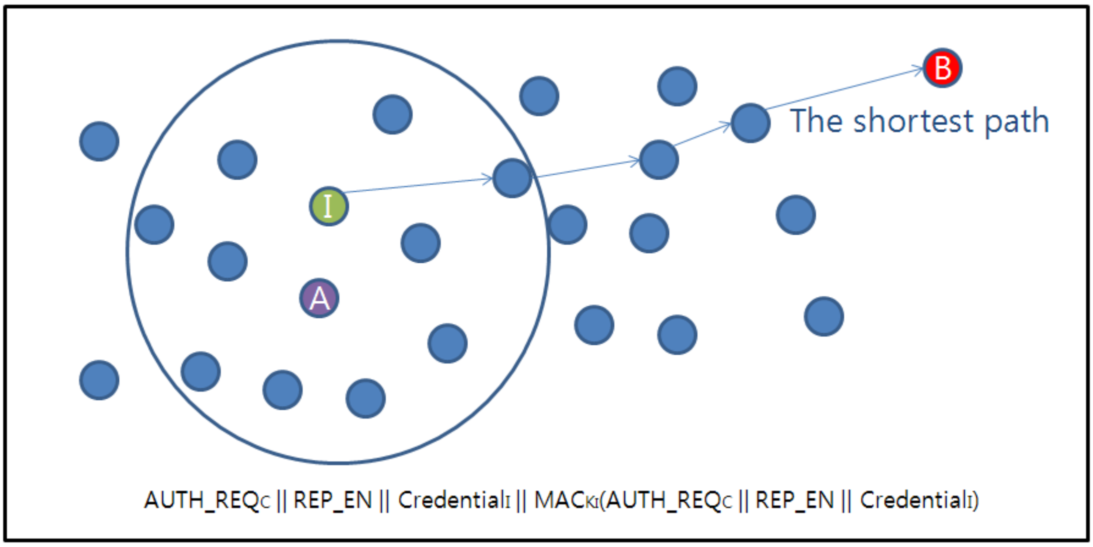

B
The shortest path
I
A
AUTH_REQ$_C$ || REP_EN || Credential$_I$ || MAC$_{K_I}$(AUTH_REQ$_C$ || REP_EN || Credential$_I$)

Figure 3.7: Report to sink node

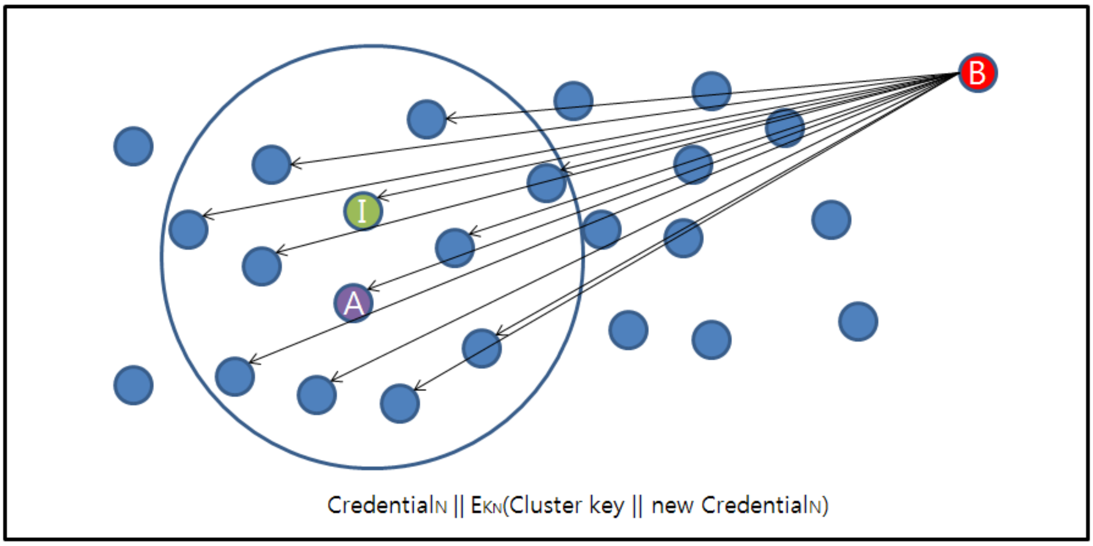

B
I
A
Credential$_N$ || E$_{K_N}$(Cluster key || new Credential$_N$)

Figure 3.8: Distribution of cluster key

4. Security Analysis

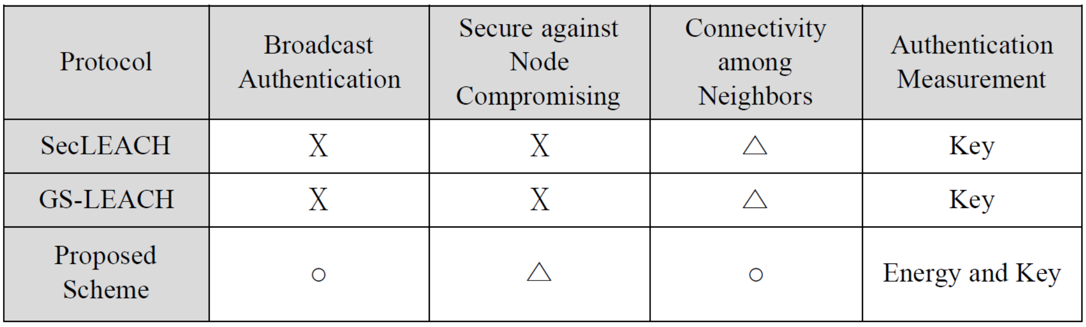
SecLEACH and GS-LEACH have three disadvantages. Fist, when an aggregator broadcasts a message, these protocols do not provide broadcast authentication. Next, SecLEACH and GS-LEACH are vulnerable to some attacks caused by node compromising. The last disadvantage is incomplete connectivity among sensor nodes.

In two protocols, firstly, an aggregator sends ID of an aggregator when broadcasting. Then, normal node chooses a source node of the strongest signal and finds a shared key with an aggregator by using a pseudorandom function. At this moment, if attackers have a device with high power level like a laptop computer and attack network [12], normal nodes misjudge that attackers are their aggregators. All sensor nodes will not work properly.

Since both SecLEACH and GS-LEACH use random key pre-distribution scheme for key management, all sensor nodes can be compromised from only a few number of compromised nodes [18].

Also, due to random key pre-distribution, both protocols do not provide complete connectivity among neighboring nodes. Indeed, when an aggregator is selected, normal nodes, which do not have the shared key with an aggregator, must take a sleep mode or directly communicate with BS with high energy consumption.

By the way, the proposed scheme provides confidentiality, freshness, integrity and almost full connectivity during clustering. The proposed scheme achieves the level of security that can defend, against exaggeratedly advertised amount of energy resources by



Protocol	Broadcast Authentication	Secure against Node Compromising	Connectivity among Neighbors	Authentication Measurement
SecLEACH	X	X	△	Key
GS-LEACH	X	X	△	Key
Proposed Scheme	○	△	○	Energy and Key

Table 4.1: Security Comparison

adversarial nodes outside a network, by utilizing the global key distributed to all nodes in the network. However, using only the global key, it cannot mitigate effects of compromised nodes which attempt to be the aggregator. Thus, we apply statistical calculation from the sink with reports of the number of transmissions and receipts. From statistical calculation of remaining amount of energy of nodes, the sink will not let an adversarial node to be an aggregator by sending the cluster key to only valid members of the cluster and will designate the inspector as the aggregator by sending additionally routing information based on reported members of nodes. By using multi-hop clustering, the proposed scheme solves LEACH families' orphan problem.

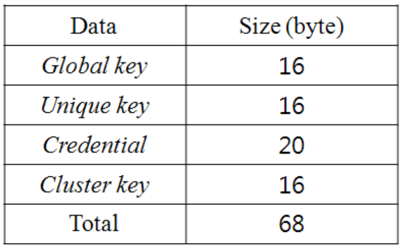
The proposed scheme can also mitigate jamming attacks, when reporting and aggregating events, due to the path repair mechanism of PEQ [3]. The communication in HPEQ [4] is $hop-by-hop$ communication. The sender on the path to the aggregator or the sink sends the message. However, the destined node will not notify the sender with ACK of destined node because of the jamming attack. The sender, then, floods $SEARCH$ message to find another path, but if any nodes do not answer $SEARCH$ message, the send will spend more energy on transmitting widely as described in [3].

However, if jamming attacks and other attacks (*e.g.* sybil attacks, sinkhole attack, and selective forwarding attacks which are performed form inside with compromised nodes), the proposed scheme cannot mitigate. Firstly, an adversary perform jamming attacks. The sender, then, will broadcasts $SEARCH$ message as following the path repair mechanism. At that time, a compromised neighbor nodes of an adversary will lure the sender into setting destination node. And then an adversary will drop or selectively send the reporting messages.

5. Overhead Evaluation

We address additional computation and communication overhead from original HPEQ caused by applying cryptographic primitives and additional messages for secure communicate. We assume use of 128-bit AES cryptographic algorithm for encrypting/decrypting messages, SHA-1 as a message digest algorithm. Thus, each node has to store following amounts of keys and *Credential* as shown on Table 5.1. 68 byte of stored data are very small capacity, even if a sensor node has the extremely memory-limited capacity.

Table 5.1: Size of Embedded Message Elements

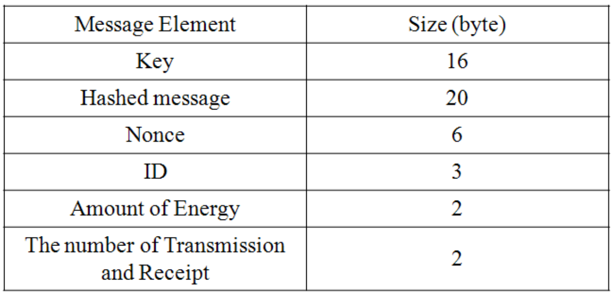


Data	Size (byte)
Global key	16
Unique key	16
Credential	20
Cluster key	16
Total	68

We also consider the communication overhead, which occupies the main part of energy dissipation. According to the radio model of [9], the amount of energy consumption on communication is affected by a transmission range and size of sent message. We, then, try to clear size of each element in messages on each step in our proposed scheme as represented Table 5.2.

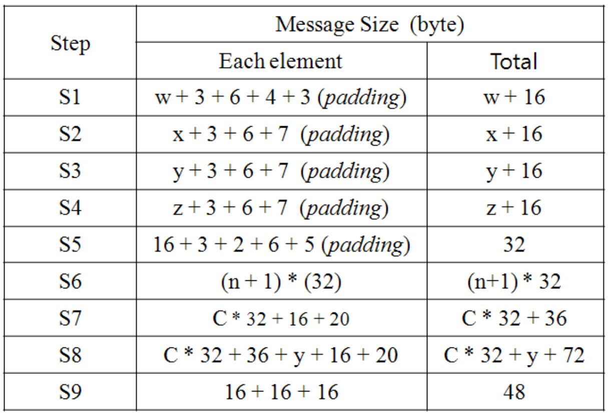
Table 5.3 presents length of messages used for each step in our scheme. w, x, y, and z denote REQ_EN, REP_EN, SET_AGR, and AGR_NTF, respectively. According to [5], from 8 byte to 24 byte length of messages, sending messages has very small gap of energy dissipation. Thus, we can ensure that our scheme shows almost the same amount of energy consumption of communication, until step 4. At step 5, we have a fewer communication overhead, since [5] presents that 32 byte length of messages increases overheads by approximately 33.55% from 24 byte length. Steps 6 and 7 cause more overheads if a sender node do not perform segmentation. Therefore, we need to divide message length into 24 byte, because, as above mentioned, 24 byte of message length consume almost same as 8 byte for transmission. Without sending all authentication tokens to the inspector node, our proposed scheme requires the same number of transmissions as original HPEQ to minimize additional transmission overheads.

Table 5.2: Size of Message Elements



Message Element	Size (byte)
Key	16
Hashed message	20
Nonce	6
ID	3
Amount of Energy	2
The number of Transmission and Receipt	2

Table 5.3: Size of Messages on each step



Step	Message Size (byte)	
	Each element	Total
S1	w + 3 + 6 + 4 + 3 (*padding*)	w + 16
S2	x + 3 + 6 + 7 (*padding*)	x + 16
S3	y + 3 + 6 + 7 (*padding*)	y + 16
S4	z + 3 + 6 + 7 (*padding*)	z + 16
S5	16 + 3 + 2 + 6 + 5 (*padding*)	32
S6	(n + 1) * (32)	(n+1) * 32
S7	C * 32 + 16 + 20	C * 32 + 36
S8	C * 32 + 36 + y + 16 + 20	C * 32 + y + 72
S9	16 + 16 + 16	48

We also try to compare with other key establishment scheme HIKES [11], which makes WSN secure against node compromising attacks. HIKES uses partial key escrow table for key generation. However, if the number of nodes in a cluster becomes big, an aggregator has to transmit a large message size. In the result, life time of network will be short. To compare with HIKES, we have some assumptions that each cryptographic primitive has same size with our proposed scheme, a cluster is composed of 2 hops, and each node located in 1 hop from an aggregator has 2 leaf nodes. In Figure 5.1, 2 hops and 1 hop are meaning a message size sent from a sensor node located 2 hops (or 1 hop) away from an aggregator.

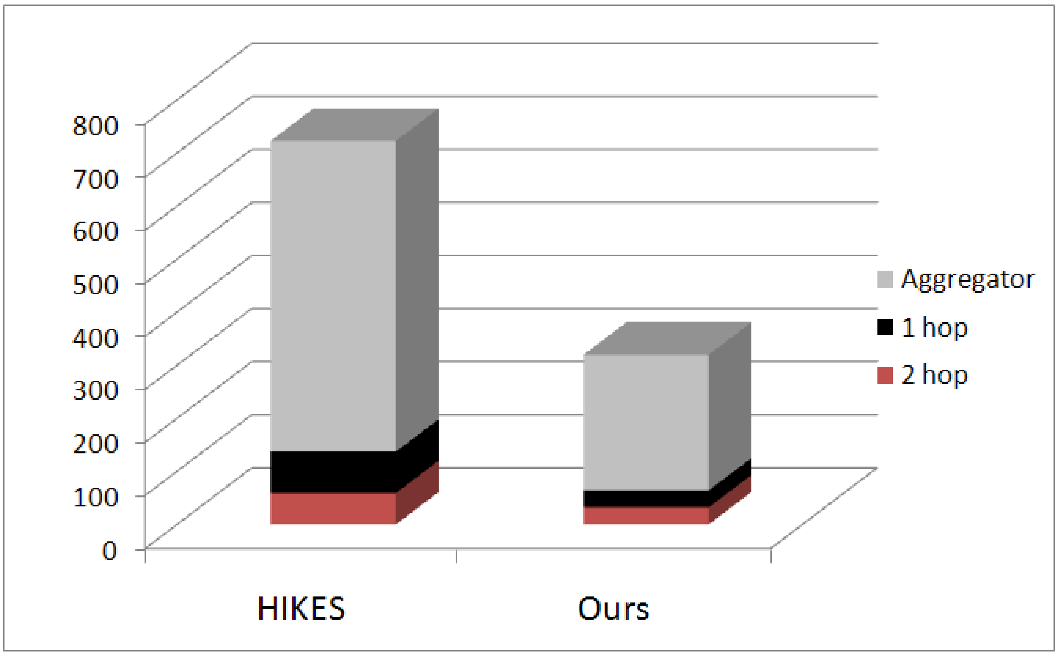


Figure 5.1: Overhead Comparison

6. Conclusion and Future Work

We examined requirements of applications monitoring critical conditions in WSN. Then, as related work, we introduced several routing protocols, which are suitable for applications monitoring critical conditions. Among introduced routing protocols, we chose HPEQ since HPEQ provides high reliability of data reporting, good load balance in WSN, high energy efficiency, and low latency. Also, we introduced clustering schemes for energy efficiency in WSN, such as LEACH, SecLEACH, and GS-LEACH. As described in before, LEACH is not concerning about security. On the other hand, SecLEACH and GS-LEACH are composed of LEACH and random key pre-distribution scheme. By the way, SecLEACH and GS-LEACH still have security vulnerabilities caused by random key pre-distribution scheme and nature characteristics of LEACH. Thus, we found that SecLEACH and GS-LEACH are vulnerable to key collision attacks and do not provide full connectivity.

Our proposed scheme provides authentication of not only an aggregator, but also all cluster members. Simultaneously, the proposed scheme provides broadcast authentication when selecting an aggregator among sensor nodes and full connectivity among sensor nodes and overcomes vulnerabilities of SecLEACH and GS-LEACH caused by random key pre-distribution scheme. Our proposed scheme uses energy level of each node as well as key for node authentication while other schemes usually uses only the key for nodes authentication in WSN. This way can be auxiliary of detecting a compromised node of an adversary, since the sink can continuously be reported from the inspector node and observe condition of the whole network. Also, the proposed scheme satisfies general security requirements, such as confidentiality with encryption, message integrity with MAC, node authentication as mentioned before, and message freshness with nonce. Despite provided security, our scheme has high energy efficiency.

While clustering among sensor nodes, an inspector node are designated for monitoring misbehavior of a cluster. A selected inspector node as IDS in WSN reports misbehavior of cluster to a sink. Then, a sink revokes suspected an aggregator of a cluster and promotes an inspector node a new aggregator.

By the way, we did not mention how the inspector node can be used for detecting abnormal behavior of a cluster. We believe that defining how to use inspector node as IDS in WSN is related to another research area. Thus, in this thesis, we mention what kinds of factors are required to research on how to use inspector node as IDS. we have to firstly define attacking models and survey other works related to IDS for comparison. Next, we should define attacking models on our proposed scheme and develop detection models. Then, the in-depth simulation and implementation also need to be verified.

요약문

무선 센서 네트워크에서 안전한 센서 노드의 클러스터 형성 기법 연구

무선 센서 네트워크는 유비쿼터스 컴퓨팅 환경에서의 필수적인 기술 중의 하나이다. 하지만 네트워크가, 연산, 저장, 전력 등의 자원이 제한된 센서노드로 구성돼있기 때문에, 보안상으로 더욱 취약하다. 중요 환경 감시목적의 라우팅 프로토콜은 그 목적에 걸맞게 알려진 공격에 대해서 강건해야 함에도 불구하고 대부분의 관련 제안된 방식들은 보안 요구사항들을 만족시키지 않는다. 그 라우팅 프로토콜들 중에서 HPEQ [4]는 긴급을 요구하는 사건의 감시에 목적을 두고 있는데, LEACH [9]의 클러스터링 방식을 변형시킨 다음에 PEQ [3]에 적용하여 에너지 효율성을 높였다. 하지만 여전히 기밀성, 메시지의 무결성, 각 센서노드의 인증 등 보안 요구사항이 고려되지 않아 매우 취약할 뿐만 아니라, 변형시킴에 따라 새로운 취약점도 내포하고 있다. 이러한 변형으로 인하여 사실상 보안을 고려한 LEACH 계열의 클러스터링 방식인 SecLEACH[17], GS-LEACH [1] 등을 적용하기 곤란하다. 이 뿐만 아니라, SecLEACH 와 GS-LEACH 역시 보안을 고려했지만, 그 근간이 되는 LEACH에서의 미아 노드 문제, 임의 키 사전 분배 방식에 따른 보안적 취약점을 그대로 가지고 있다.

이에, 본 논문은 기밀성, 데이터의 무결성 검사 그리고 각 노드의 인증과 같은 보안 요구사항을 만족시키는 에너지 보유량을 고려한 라우팅 프로토콜에서의 안전한 클러스터 형성 기법을 제안한다. 이는 기존의 방식인 SecLEACH 와 GS-LEACH가 가진 미아 노드 문제 뿐만 아니라 임의 키 사전 분배 방식이 가진 보안적 취약점을 다른 접근 방식을 통하여 해결 하였다. 또한, 제안되는 방식은 추후의 침입탐지시스템 운용을 고려하여 클러스터 내부에, 다른 추가비용 발생 없이 클러스터의 행동을 감시할 수 있는 노드를 구축하는 방식을 품고 있다.

References

[1] P. Banerjee, D. Jacobson, And S. N. Lahiri, "Security and Performance Analysis of a Secure Clustering Protocol for Sensor Networks", *Proceedings of 6th IEEE International Symposium on Network Computing and Applications (NCA 2007)*, 2007.

[2] A. Boukerche, I. Chatzigiannakis, and S. Nikoletseas, "A New Energy Efficient and Fault-tolerant Protocol for Data Propagation in Smart Dust Networks using Varying Transmission Range", *In 37th ACM/IEEE Annual Simulation Symposium - ANSS*, 2004.

[3] A. Boukerche, R. W. N. Pazzi, And R. B. Araujo, "A fast and reliable protocol for wireless sensor networks in critical conditions monitoring applications", *International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWIM'04)*, pp. 157-164, October 04-06, 2004, Venice, Italy.

[4] A. Boukerche, R. W. N. Pazzi, And R. B. Araujo, "HPEQ - A Hierarchical Periodic, Event-driven and Query-based Wireless Sensor Network Protocol", *Proceedings of the IEEE Conference on Local Computer Networks 30th Anniversary (LCN'05)*, pp. 560-567, 2005.

[5] C. Chang, D. J. Nagel, and S. Muftic, "Measurement of Energy Costs of Security in Wireless Sensor Nodes", *Proceedings of ICCCN 2007*, Honolulu, Hawaii, USA, August 13 - 16, 2007.

[6] I. Chatzigiannakis, S. Nikoletseas, and P. Spirakis, "A Comparative Study of Protocols for Efficient Data Propagation in Smart Dust Networks", *In Proc. 2nd ACM.POMC´2002*, 2002.

[7] L. Eschenauer and V. D. Gligor, "A key management scheme for distributed sensor networks", *In 9th ACM conference on Computer and communications security*, pp. 41-47, 2002.

[8] D. Estrin, R. Govindan, J. Heidemann, "Embedding the Internet", *Communication ACM 43*, 2000

[9] W. Heinzelman, A. Chandrakasan, And H. Balakrishnan, "Energy-efficient communication protocol for wireless sensor networks", *Proceedings of the 33rd Hawaii International Conference on System Sciences (HICSS'00)*, Hawaii,January 2000.

[10] J. Ibriq and I. Mahgoub, "A Secure Hierarchical Routing Protocol for Wireless Sensor Networks", *Communication systems (ICCS 2006). 10th IEEE Singapore International Conference on*, pp. 1-6, 2006.

[11] J. Ibriq and I. Mahgoub, "A Hierarchical Key Establishment Scheme for Wireless Sensor Networks", *Advanced Information Networking and Applications, (AINA '07). 21st International Conference on*, pp. 210-219, 2007.

[12] C. Karlof and D. Wagner, "Secure routing in wireless sensor networks: Attacks and countermeasures", *Elsevier's Ad-Hoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 2003.

[13] G. Khanna, S. Bagchi, and Y. Wu, "Fault Tolerant Energy Aware Data Dissemination Protocol in Sensor Networks", *IEEE DSN*, Florence, Italy, 2004.

[14] S. Lindsey and C. S. Raghavendra, "PEGASIS: Power Efficient GAthering in Sensor Information Systems", *in the Proceedings of the IEEE Aerospace Conference*, Big Sky, Montana, 2002.

[15] A. Manjeshwar, and D. P. Agrawal, "APTEEN: A Hybrid Protocol for Efficient Routing and Comprehensive Information Retrieval in Wireless Sensor Networks", *in the Proceedings of the 2nd IPDPS´02*, Ft. Lauderdale, FL, 2002.

[16] S. Nikoletseas, I.Chatzigiannakis, A. Antoniou, H. Euthimiou, A. Kinalis, And G. Mylonas, "Energy Effcient Protocols for Sensing Multiple Events in Smart Dust Networks.", *Proc. 37th Annual ACM/IEEE ANSS'04, IEEE Computer Society Press*, pp. 15-24, 2004.

[17] L. Oliveria, H. Wong, M.Bern, R. Dahab, And A.A.F. Lourerio, "SecLEACH: A Random Key Distribution Solution for Securing Clustered Sensor Networks", *In the Fifth IEEE International Symposium on Network Computing and Applications (NCA'06)*, 2006.

[18] T. Moore, "A Collusion Attack on Pairwise Key Predistribution Schemes for Distributed Sensor Networks", *Proceedings of the Fourth Annual IEEE International Conference on Pervasive Computing and Communications Workshops(PERCOMW '06)*, 13-17 March 2006.

[19] M. Younis, M. Youssef, and K. Arisha, "Energy-Aware Routing in Cluster-Based Sensor Networks", *in the Proceedings of the 10th IEEE/ACM MASCOTS' 02*, Fort Worth, TX, 2002.

감사의 글

이 논문을 완성하기까지 주위의 모든 분들로부터 수많은 도움을 받았습니다. 김광조 교수님께서는 틈틈히 연구 상황을 확인해 주셔서 체계적인 연구방향을 세울 수 있었습니다. 게다가 장성형, 진형에게 연구실 생활 및 연구 주제 등 많은 부분에서 격려 및 조언을 받았습니다. 또한, 바쁘신 와중에도 학위논문심사를 위해 참석하셔서 진심어린 조언을 주신 김명철 교수님, 이병천 교수님께 감사드립니다.

끝으로 오늘의 제가 있을 수 있도록, 항상 저를 믿어주시고 사랑으로 키워 주신 아버지, 어머니에게 감사드립니다. 저의 이 결실이 그분들께 조금이나마 보답이 되기를 바랍니다.

Curriculum Vitae

Name : Myunghan Yoo

Date of Birth : January 27, 1985

Birthplace : 1524-11, Bongcheon-dong, Gwanak-gu, Seoul, 151-050 KOREA

Domicile : 264-131, Sangdo-dong, Dongjak-gu, Seoul, 156-030 KOREA

Address : 102-1405, Daeju fiore, 367-7 Songjeong-dong, Gwangju-si, Gyeonggi-do, 464-903 KOREA

E-mail : bishnu@kaist.ac.kr

Educations

2000. 3. – 2003. 2. 한솔 고등학교

2003. 3. – 2008. 2. Information Security, Korea University of Technology and Education (B.S.)

2008. 2. – 2010. 2. Information and Communications Engineering, KAIST (MS course)

Career

2008. 1. – 2008. 12 EPCglobal 차세대 표준 연구

2008. 3. – 2009. 2. 디지털 계측제어계통 침투시험을 통한 사이버보안 취약성 분석 및 침투 테스트

2009. 2. – 2009. 9. WSN 분산 Authentication 기술 개발

2009. 6. – 2009. 8. (주)위너다임 인턴쉽 프로그램 참여

2009. 3. – 2010. 2. 디지털시스템 사이버보안 평가기술 개발

Publications

1. **Myunghan Yoo**, Jangseong Kim, and Kwangjo Kim, *A Secure Clustering Scheme over an Energy-aware Routing Protocol for Monitoring Critical Conditions*, Symposium on Cryptography and Information Security 2009 (SCIS 2009), Jan. 20-23, Otsu, Japan.

2. **유명한**, 김장성, 김광조, 중요 *환경 감시목적의 에너지 보유량을 고려한 라우팅 프로토콜에서의 안전한 클러스터 형성기법* 한국정보보호학회 하계학술대회 2009, pp.135-139, 2009년 6월 19일, 강원대학교, 삼척.

3. 김장성, **유명한**, 김광조, *u-City에서 센서네트워크에 기반을 둔 자녀 안심서비스의 보안요구사항 분석 및 시스템 제안* 한국정보보호학회 하계학술대회 2009, pp.135-139, 2009년 6월 19일, 강원대학교, 삼척.

4. **유명한**, 김장성, 김광조, *(Im)Possibility of Denial-of-Service Attacks on Network Layer in Wireless Mesh Networks* 한국정보보호학회 동계학술대회 2009, pp.77-81, 2009년 12월 5일, 연세대학교, 서울.

5. Jangseong Kim, **Myunghan Yoo**, and Kwangjo Kim, *A Privacy-Preserving Kid's Safety Care Service Based on Sensor Network in u-City* Symposium on Cryptography and Information Security 2010(SCIS 2010), Jan. 19-22, Kagawa, Japan. (to appear)