Adaptive Security Modules in Incrementally Deployed Sensor Networks

Meng-Yen Hsieh[1] and Yueh-Min Huang[2]
[1]Department of Information Science, Hsing Kuo University of Management, Taiwan, R.O.C.
[2]Department of Engineering Science, National Cheng-Kung University, Taiwan, R.O.C.
tab.hsieh@mail.hku.edu.tw, huang@mail.ncku.edu.tw

Abstract

Distributed wireless sensor networks often suffer problems on detecting malicious nodes, which always bring destructive threats. Thus, sensor networks have to supply authentication services for sensor identity and data communication. As matter of fact, intrusion detection and prevention schemes are always integrated in sensor security appliances so that they can enhance network security by discovering malicious or compromised nodes. This paper provides adaptive security modules to improve secure communication in distributed sensor networks. The primary security module provides online identity authentication services to new incoming sensor nodes which being distributed after initial deployment. The advanced security module addresses compromised node detection issues to exclude internal compromised nodes. The proposed schemes can accomplish secure communications in the sensor networks when the network lifetime is divided into multiple time intervals. The network security and network performance are evaluated with the adaptive security modules, which shows efficient protection and sensible overheads to sensor nodes can be achieved.

Keywords: sensor network, authentication, security, incremental deployment, hierarchical architecture

1 Introduction

Distributed sensor networks consisting of many low-energy sensors are used to monitor oceans and wildlife, manufacturing machinery performance, building safety, earthquakes, and many military applications. Homogeneous sensor nodes are often deployed in open and unattended environments without physical protection. Wireless sensor networks have a number of characteristics, such as centralization, cooperative transmission, vulnerability, limited transmission range, and resource constrains. Wireless sensor networks are generally more prone to physical security threats than other wireless networks. Hence, security design is vital for sensor network applications because of vulnerability to active and passive attacks due

to the wireless nature of link connections among sensor nodes. The possibility of different attacks from internal and external malicious nodes should be considered.

Any two sensor nodes connecting directly require security properties including confidentiality, authentication, integrity, and freshness [1, 2]. The sensor networks may be deployed in un-trusted locations or sensor nodes communicate highly sensitive data, thus two sensor nodes need to communicate with a secure link. The standard approach for keeping secure communication is to encrypt the data with a secret key that only intended receivers possess, hence achieving confidentiality. Identity and data authentication are significant for sensor network applications. During the period of the establishment of sensor networks, authentication is necessary. Malicious nodes can imitate a normal node or intrude the network to inject fake or copied messages. Therefore the receivers have to make sure that the data used in any decision-making process originates from the correct source. In the two-node communication case, authentication can be performed through a purely symmetric key cryptography. A pairwise secret key shared between the sender and the receiver can compute a message authentication code to achieve identity and message authentication. In communication, data integrity ensures the receiver that the received data is not altered in transit by a malicious node. Besides, data freshness is important, since all sensor networks steam some forms of time varying measurement. In general, data freshness implies that the data is latest, and it ensures that no malicious nodes replayed old data.

Since the sensor nodes always have limited resource, the overheads must be considered. In generally, the communication overhead of sensor nodes is much more expensive than the storage and computation overheads. Sensor nodes constitute a hierarchical architecture and work with self-organization management methods to reduce the communication overhead. In hierarchical sensor networks, the network is typically organized into clusters, with ordinary member nodes (MNs) and the cluster heads (CHs) playing different roles [3, 4]. The CHs are responsible for additional tasks such as gathering and processing the sensing data from their localized MNs, and relaying aggregated results towards the base station (BS), while localized member nodes are responsible for sensing. Figure 1 illustrates a two-level network where CHs are located in the top level to communicate directly with the BS/Sink, and the localized sensor nodes in the lower level. However, a CH in this architecture needs to use a strong transmission with long-distance radio to communicate directly with the BS.

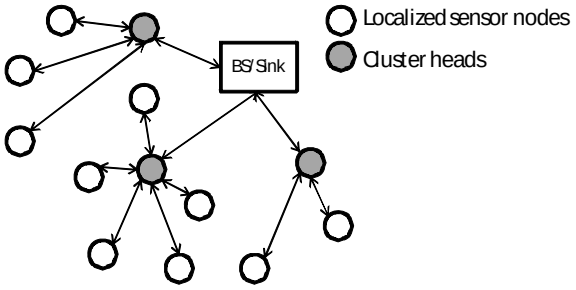

Figure 1: Illustration of a hierarchical sensor network

This paper proposes secure communication with two security modules in a distributed sensor network. Neighboring sensor nodes in the network can establish secure links and broadcast authentication. Malicious or compromised nodes can be detected and eliminated from the network. The proposed network is designed for incrementally deployed networks, and employs a key chain to validate sensor node certificates at each increment. The rest of the paper is organized as follows. Section 2 describes related work on security issues of attack categories. Section 3 introduces the primary security module to design an authentication service for incrementally deployed sensor nodes. Section 4 provides malicious node detection schemes to discover and exclude malicious and compromised nodes. Section 5 provides security analyses and evaluates the performance cost of a sensor network with dynamic authentication. Finally, conclusions and future work are made in Section 6.

2 Security Issues

In open environments, sensor nodes are susceptible to attacks by malicious nodes. The networks only use cryptography instead of the firewall to protect information against malicious attacks. To differentiate attacks from malicious nodes, sensor network security can easily be breached either by passive attack, such as eavesdropping, or active attacks, such as denial of service (DOS) attacks. However, to separate attacks from malicious nodes where they are from, inside and outside attacks are considered in sensor network.

Passive Attacks: The attacks do not have a direct effect on network communications and only eavesdrop or monitor the transferring message. Due to the nature of wireless channels, passive attacks are performed easily. For confidential transmission, sensing data exchanged among nodes should be encrypted. Traffic analysis is another threat to the sensor networks.

Active attacks: The attacks not only eavesdrop data transmission, but also affect network communication, for instances, manipulating transferring data, obstructing transmissions, and injects faulty data into the network. An active attacker could masquerade as a legitimate

member of the network and broadcast its unused information or replay old data, which might cause DoS attacks inside the network. Furthermore, since the sensor nodes are not tamper-proof, the attacker might physically compromise captured nodes. After capturing normal nodes, the attacker is likely to gain cryptographic key information from the captured nodes.

Outside Attacks: An outside attacker node is a malicious node which is not authorized in the network. If the attacker is passive, it can attempt to steal private or sensitive information. The attacker can also modify or spoof packets to compromise the authenticity of communication or inject interfering wireless signals to jam the network. An outside adversary can inject ineffectual data to deprive the received node's battery, which can capture or physically destroy the node. To prevent against outside attacks, transmission and identity authentication, which is performed by shared key cryptographic schemes, is necessary.

Inside Attacks: Using compromised nodes to attack from the inside is the main threat to sensor networks. The compromised nodes can easily destroy or disrupt network operations. A compromised node has the following characteristics: (1) the compromised node is running some malicious code that is different from the code running on a legitimate node and seeks to steal information from the sensor network or disrupt its normal function; (2) the compromised node uses the same radio frequency as the other normal sensor nodes so that it can communicate with them; (3) the compromised node is authenticated and participates in the sensor network. Since secure communication in sensor networks is encrypted and authenticated using cryptographic keys, compromised nodes with the secret keys of a legitimate node can participate in the secret and authenticated communication of the network.

3 Primary Security Design

3.1 Key Pre-distribution schemes

The proposed primary security module provides a dynamic authentication service for incrementally deployed sensor nodes even in the presence of outside malicious nodes. The dynamic authentication is derived from the TESLA Certificate [11]. Outside malicious nodes can be deployed, when a system is designed for incrementally deployed sensor nodes. To deal with this problem, sensor nodes deployed already in the network need to authenticate new incoming sensor nodes. This authentication service influences that two neighbour nodes establish a secure link for their data transmission afterwards. In the authentication service, each node has to gain a certificate, called *TCert*, from the base station (BS). The BS plays the role of a certificate authority and issues valid certificates to sensor nodes. Suppose that the BS

has divided the network lifetime into multiple time intervals, assigning different system keys to different time intervals. The BS adopts the key chain ($\{TK_i\}$: $TK_0 \leftarrow TK_1 \leftarrow \ldots \leftarrow TK_N$) as system keys. Each system key is disclosed at the beginning of the corresponding time interval.

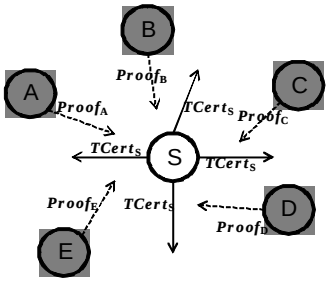

Figure 1: A new incoming node broadcasts a *Tcert* and receives *proof* messages from neighboring nodes.

Fig. 1 depicts the details of the dynamic authentication of a node, S, when it is deployed. Node *S* broadcasts its certificate, $TCert_S$, with one key element of its key chain, called tK_1^S. Each of the existing sensor nodes neighboring *S* can receive $TCert_S$ and tK_1^S. The neighboring nodes validate the $TCert_S$ to confirm that it has not expired. The valid $TCert_S$ is stored; otherwise the certificate is dropped. When cluster round *i* starts, the system key TK_i is disclosed by the BS. All nodes in the network receive the system key, TK_i, since the BS has unlimited power to transmit data with long-distance radio, which covers the entire network. Node *V*, as one of the neighboring nodes, uses TK_i to verify the MAC code of the received $TCert_S$, and to decrypt the key information, (K_S, tK_0^S). Node *V* takes the hash function *F* to hash the key tK_1^S to check the correctness of the key tK_0^S of Node *S*. After hashing with $F(tK_1^S) = tK_0^S$, if the key commitment is right, K_S of Node *S* can be trusted. Node *V* establishes a pairwise key, $K_{V,S}$, using its private function f_V and K_S, which can be represented as $K_{V,S} = f_v(K_S)$. A reception message, $Proof_V$, must be constructed by Node *V*, and returned to *S*. The message *proof* transmitted from Node *V* to Node *S* carries a key derivation ($f_V(ID_S)$) and the key commitment of Node *V* (tK_0^V) encrypted by $K_{V,S}$, besides responding to the corresponding $TCert_S$. After accepting the *proof* message, Node *S* derives a pairwise key, $K'_{V,S}$, using the key derivation and its private function. Since the private functions of Node *S* and Node *V* are commutative, the pairwise key created by Node *V* is same as the pairwise key created by Node *S*. Node *S* uses $K_{S,V}$ to decrypt the key commitment of Node *V*, tK_0^V. Key $K_{V,S}$ also verifies the correctness of the *proof* message and checks the message integrity. The fresh

nonce, called $Nonce_V$, generated from Node *V*, and attached to the *proof* message, prevents replay attacks. Consequently, when TK_i is disclosed, *V* can authenticate *S*, establish a pairwise key, and share key commitments.

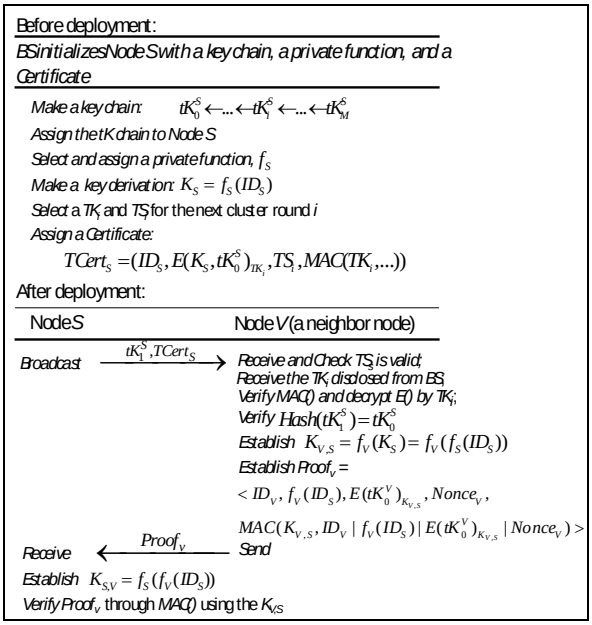

Figure 2: The dynamic authentication with *TCert*

Therefore, any malicious node without a correct certificate cannot enter the network. However, Compromised nodes that have always existed in the network can still establish pairwise keys with new nodes if they are neighbors.

3.2 Basic Secure Communications

Due to dynamic authentication, two neighbouring nodes can share a pairwise key and their key commitments each other. When a sensor node detects something, the sensed data is routed from the node to the BS using hop-to-hop authentication or encryption based on the sensitivity and criticality of data. However, sensor nodes are constrained in energy supply and bandwidth so that routing in sensor networks is very challenging. Routing protocols in sensor networks are divided into three categories, data-centric, hierarchical and location-based [9]. Assume that no malicious node can participate in the routing path from the source node to the BS.

For example, node S can deliver sensed data to the BS through a routing path (S→X→BS) using hop-to-hop authentication in a sensor network. The detailed procedure is as follows:

S→X: <Data, (ID_S), Nonce, H_S, MAC($K_{S,X}$, H_S | Nonce | Data)>; $H_{S,X}$ = hash (ID_S);

X→BS: <Data, (ID_S, ID_X), Nonce', $H_{S,X}$, MAC($K_{X,BS}$, $H_{S,X}$ | Nonce' | Data)>; $H_{S,X}$ = hash (hash (ID_S), ID_X)

When using hop-to-hop authentication, node S sends the sensed data to the next hop. The encryption procedure applied to the same route is as follows:
S→X: E($K_{S,X}$, Data, | Nonce); X→BS: E($K_{X,BS}$, Data | Nonce);

3.3 Multiple Time Intervals

For the dynamic authentication incrementally deployed sensor networks, the network lifetime is divided into multiple time intervals. The time intervals are corresponding to the periodically disclosed keys of the system key chain, $\{TK_i\}$. Since one key of the key chain is disclosed at the beginning time of its corresponding time interval, one new node with a *TCert* encrypted by the next disclosure key should be deployed at the end of the current interval.
In the cluster-based network, the LEACH architecture [5] is appropriate to the proposed dynamic authentication, when multiple time intervals of the network lifetime are corresponding to multiple cluster rounds. Figure 3 indicates that the system key, TK_i, is disclosed in the i^{th} cluster round in LEACH. New incoming nodes with new *TCert* certificates deployed in the $(i\text{-}1)^{th}$ cluster round will be authenticated by their neighboring nodes at the beginning of the i^{th} cluster round.

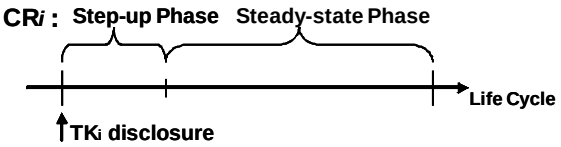

Figure 3: The system key disclosure in LEACH

4 Malicious Node Detection

This section introduces detection and prevention methods of malicious compromised nodes which are advanced security issues including monitor mechanisms, alarm return protocols, and trust-value evaluation methods.

4.1 Monitor Mechanisms

To generate alarm packets for abnormal activity events, normal nodes as monitors can discover abnormal activities in suspect nodes. The mechanisms provided in [6, 7, 8] can monitor node availability and message traffic among neighbor nodes. Four possible monitoring methods are described below:

Neighboring node monitoring: A node monitors the traffic going in and out of its neighbors. Moreover, a node can limit the traffic from its neighbors. In Fig. 4, nodes X and Z are the monitor nodes for the link from X to Y. Information from each packet in the link is saved in a buffer at each monitor. Y is expected to forward the packet to the next hop as Node D. According the monitor scheme [8], the probability of issuing an alarm at one monitor node is given by:

$$P_{g|w} = \sum_{i=g}^{w} \binom{w}{i} (1-P_a)^i (P_a)^{w-i} \qquad (1)$$

, where P_a is the probability of missed detection; w packet fabrications occur within a certain time window, T. At least g fabrications are detected by the monitor node;

Node availability monitoring using Hello/Wakeup beacons: A sensor node typically runs according to an active/sleep schedule. Assume that each scheduled node must actively advertise a Hello/Wakeup beacon when waking up from sleep mode to active mode. Monitor nodes close to awakening nodes can monitor the availability of these nodes when the active/sleep schedules are known. Although collision and interference cause beacon advertisement to fail, monitor nodes can reply to an awakening node with acknowledge packets after listening to a Hello/Wakeup beacon. A threshold waiting time is set when one monitor node knows that another node is awakening. The monitor node issues an alarm if it does not receive any Hello/Wakeup beacon from the awakening node. If t is the round trip time of a request and the response beacons in one-hop distance, then the threshold time is set as $T_{listen} \geq N_{beacon} \times t$, where N_{beacon} is the maximum number of Hello/Wakeup beacons issued from one awakening node. In general, neighboring nodes should know each other's active/sleep schedules, and a CH should know the schedules of its members in a cluster.

Node availability monitoring using Hello/Measure beacons: A node can actively measure another node in one hop through Hello/Measure beacons. A measured node is abnormal when a monitor node does not gain enough replies after continuing to measure up to m times within a time limit. The normal availability of a node i, defined as A_i, is computed from the number of Hello/Measure beacons m as follows:

$$A_i = (\sum_{j=1}^{m} MBeacon_j) / m \qquad (2)$$

, where $MBeacon_j$ denotes the j^{th} Hello/Measure beacon; If the monitor node gains a response matching to the measure beacon, then $MBeacon_j = 1$; otherwise, $MBeacon_j = 0$.

Node availability monitoring combined with packet forwarding: To reduce the overhead of control packets, a node can measure the availability of another node when forwarding packets between them. Acknowledge packets are required to correspond to forwarding packets.

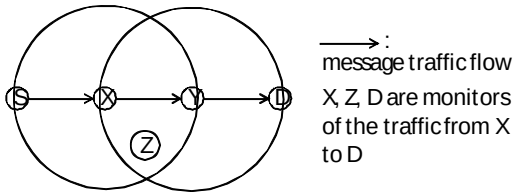

Figure 4: Illustration of neighboring node monitoring

4.2 Alarm Return Protocols

In the study, the network enables sensor nodes to be monitor nodes with the ability of accusing suspect nodes using alarm packets. An alarm packet must be returned to the BS when a monitor node detects abnormal activities in the network. The malicious nodes should be announced by the BS, but not their neighboring nodes. First, the monitor node broadcasts an alarm packet to neighbor nodes. Second, any neighbor node got the packet forwards the packet through one of known routes from it to the BS. According to the first step, the monitor node can notify its neighboring nodes that some malicious node is located in the neighborhood. The second step reduces the broadcast storm. The packet formats of alarm return protocols are described blow:

X→Neighbor: AccuseEvent, (ID_X) tK_i, MAC(tK_i, AccuseEvent), H_X

Y→Z: AccuseEvent, (ID_X, ID_Y), MAC($K_{Y,Z}$, AccuseEvent|(ID_X, ID_Y)), H_X

A hash value related to an alarm packet is generated. For example, Node X can generate a hash value, denoted as H_X, generated by a one-way hash function with its individual key, an *AccuseEvent*, and one key of its key chain.

$H_X = \text{Hash}(IK_X, AccuseEvent \mid tK_i^X)$, where IK_X is the private key only shared between X and BS; tK_i^X is one key of the key chain of X, authenticated by $\text{Hash}^i(tK_i) = hash(\ldots hash(tK_i)\ldots) = tK_0$.

To prevent unlimited or irregularly accusations produced, an alarm packet issued from one monitor node should be attached with one key of its' key chain. Therefore, the number of times that an alarm packet is issued by a monitor node is limited to the key number of the key chain of the node. The neighboring nodes of the monitor node can also authenticate the alarm

packet. Furthermore, the BS must check the validity of an alarm packet and the accuser using the private key between it and the monitor node.

In a flat sensor network, the alarm return paths are routing paths from monitor nodes to the BS. The monitor node must establish a route to the BS according the flat routing protocols. In a cluster-based network, monitor nodes can deliver alarm packets to the BS via CHs. Figure 5 indicates four possible routing paths in the three return paths when one monitor node returns an alarm packet. The first return path in Fig. 5(a) presents that a monitor node as a CH accuses a suspect MN of abnormal activities, and return an alarm packet to the BS. The second return path in Fig. 5(b) presents that a monitor node as a MN belonging to some cluster accuses a suspect node which is not the CH of the accuser. The alarm packet issued by the MN will be relayed to the BS via the CH of the MN. The third return path in Fig. 5(c) is to identify a monitor node as a MN if a cluster accuses its CH of abnormal activities. Since the suspect CH is not trusted by the monitor node, an alarm packet issued by the MN is returned to the BS via other valid nodes belonging to other clusters.

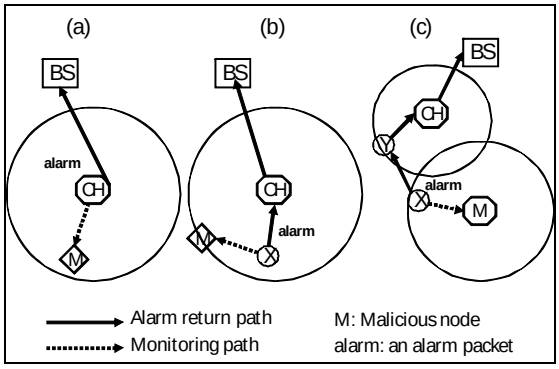

Figure 5: Illustration of the three alarm return paths in the cluster-based network.

When BS gets an alarm packet, BS needs to decide the accused node whether or not it should be insulated from the network, based on the trust value evaluation.

4.3 Trust Value Evaluation

Different abnormal activities have different seriousness to the network. To accuse a node of being compromised, the study applies a trust value (TV) approach to evaluate the reliability of sensor nodes. Public weight-based and threshold schemes to calculate the trust value of a node are available. Generally, each deployed node has an initial trust value, denoted as TV=1. A node is not compromised if its TV is greater than a threshold trust value. To evaluate trust values of sensor nodes, the seriousness value of an alarm packet should accuse a node and reduce its TV to “platitudes”.

In the network, five abnormal activity events are illustrated in a malicious or compromised node, and these are divided into two sets, E1 and E2, in which E1 = {e1, e2} and E2 = {e3, e4, e5}: (1) e1: the node executes a false active/sleep schedule; (2) e2: the node does not react to incoming hello messages within a short period of time; (3) e3: the node drops sensed packets; (4) e4: the node delivers bogus or duplicate packets, and (5) e5: the node interferes with normal forwarding packets. E1 is one type of abnormal activity that includes sensor unavailability events, and E2 is the other type of abnormal activity that includes the misbehavior events of packet forwarding. For normal security, the unavailability of a sensor node is more serious than its packet forwarding misbehavior, since the nature of wireless channels implies that packet forwarding is unstable. Therefore, events in E1 are more serious than those in E2.

In a flat network, the BS only gives different seriousness values to different sets of abnormal activity. However, in a cluster-based network, BS can consider that different sensor roles have different powers to issue an alarm packet. Considering a hierarchical architecture, a monitor node plays a cluster head, a member node, or an ordinary node which is trusted differently in the sensor network. A suggested model includes three possible monitor roles. The CHs, denoted as R1, can monitor or detect member nodes, besides aggregating sensed data. The MNs, denoted as R2, can monitor cluster heads, as well as reporting sensed data. A node, not belonging to any cluster, can be monitored by its one-hop neighboring nodes. The monitor node plays a third role, denoted as R3. Alarm packets generated from monitor nodes influence the trust value of an accused node. In the model, one alarm packet has a seriousness value based on different accusation events from different roles. Figure 6 depicts the model with possible seriousness values (*P0, P1, P2, P3, P4, P5*), according to abnormal activity events and monitor roles.

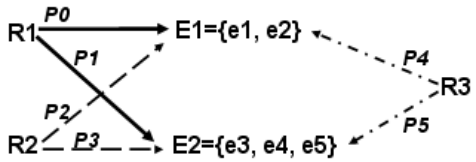

Figure 6: A seriousness model in a cluster-based network.

5 Analysis and Performance

This study focuses on authenticity, integrity, and freshness for secure communication when the network is operated with the proposed security modules.

5.1 Security Analysis

In the primary security module, the network employs a key chain to validate sensor node certificates at each increment, which is called dynamic authentication. The BS periodically discloses one key of the chain, {*TK*}, according to the time order of network time intervals, so that certificate authentication is achieved in sensor networks using symmetric cryptography. This authentication scheme prevents outside malicious nodes without valid certificates from participating in the network. However, each valid *TCert* must expire in a short period of time to reduce the probability of malicious nodes reusing the certificate. To shorten the period of the expiry time of a certificate, the deployment of new nodes is suggested to be at the end of each time interval. The validity period should be adjusted according to network application needs. In general, cooperation of neighboring nodes can detect replay attacks through overhearing. The dynamic authentication from a new incoming node to existing neighbor nodes can complete efficiently with one broadcast time and one time of receiving a response, which reduces the probability of malicious node attacks. Moreover, malicious nodes without the key chain cannot imitate the BS to deploy new nodes into the network, since they do not have the right key to disclose.

In the advance security module, the network must detect and prevent the damages from compromised node. The dynamic authentication scheme triggers the establishment of pairwise keys ($K_{i,j}$) between neighboring nodes, when the pairwise key is constructed by their key derivations and their private functions following the commutative law. Using pairwise keys, hop-to-hop data transmission in sensor networks can achieves authenticity, integrity, and confidentiality. The dynamic authentication scheme also triggers the exchange of key commitments (tK_0) between neighboring nodes. A key commitment is used to broadcast authentication when a monitor node broadcasts an alarm packet to the network.

This study adopts one-way hash key chain for one-hop broadcast authentication to broadcast an alarm packet from a monitor node to its neighborhood. A malicious node could locate itself in the neighborhood of the monitor node to catch alarm packets with disclosed keys and make replay attacks. However, it will not be successful because of the Triangle Inequality technique. From the triangle Inequality theorem, when a node floods a packet containing a message and a one-way hash key, its neighbor nodes will accept the packet before it accepts a re-forwarded copy from a malicious node. The malicious node cannot reuse disclosed one-way hash keys to attack other nodes since these keys are effective for only a one-hop distance.

An individual key, IK_i allows that alarm return to take place at legal monitor nodes, since the key is only shared between one valid node and the BS. Malicious nodes without individual keys cannot produce and return any alarm packet to the BS. Although a malicious node could compromise normal nodes to catch individual keys, fake alarm packets issued by malicious nodes are limited to the number of the key chains corresponding to those compromised individual keys. The BS plays a third party to judge whether suspect nodes accused by monitor nodes are malicious using the trust-value evaluation method. The BS can record the information of malicious nodes to a black list. When the network lifetime is divided into multiple time intervals, the black list with added malicious node judged in the current interval must be distributed to the network at the beginning of the next interval. The black list can be authenticated using the disclosed system key.

5.2 Overhead Analysis

Besides the MAC scheme, the network uses symmetric cryptography, and one-way hash generation to achieve dynamic authentication. Since encryption and MAC operations were performed quickly using symmetric keys, the computation overhead was very low for sensor nodes. When a subset of RC5 was applied for the block cipher, the time cost of encrypting a message with 8 bytes of data and generating a MAC code with one key was about 2.38 to 3.32 milliseconds, and the energy cost of those operations was about $30\,\mu$J. The time costs of encryption and decryption were 1.64 milliseconds and 1.78 milliseconds, respectively. The time cost of one-way key chain verification was about 4.1 milliseconds. Furthermore, the time cost of a *TCert* authentication was about 10.64 milliseconds. This analysis employed the following consumption rates [10]: 16.25 μJ/byte and 12.25 μJ/byte for transmission and reception, respectively, and assumed 36-byte packets for each transmission. Using the TinyOS framework, this study evaluates the performance of the security modules equipped in the sensor network, including the storage, computation, and communication overhead, since we do not consider the complex routing protocols and alarm return protocols.

Storage Overhead: A deployed node (Node$_i$) holds pairwise keys ($K_{i,j}$), key commitments ($tK_0^j, j \neq i$) of neighboring nodes, one system key commitments (TK_0), an individual key (IK_i), an individual key chain (tK). Furthermore, Node*i* must have memory space to record several certificates (*TCert*) from new incoming nodes deployed at the same time in its neighborhood. Each node has storage space for a neighbour list to record the neighbour nodes with shared pairwise keys. Suppose that the ID of a node is 2 bytes long, and one record in a neighbour

list is 7 bytes long. Suppose that each key is 8 bytes long, each nonce is 2 bytes long, a timestamp was 4 bytes long, and a *TCert* is 36 bytes long. Suppose that p is the max number of keys of one individual key chain for one node, and t is the number of *TCert* certificates from new incoming nodes deployed at the same time. If Node$i$ has n neighboring nodes, then the node needs $8\times(2n+p+5)$ bytes to store all keys. Each node needs $36t$ bytes to store valid certificates. Current sensor nodes such as Berkeley or MICA Motes provide at least 128 KB data and 128 KB program space, which are enough the proposed design.

Computation Overhead: This overhead was much lower than the communication overhead. A deplyed node processes packets from neighbor nodes. The complexity of the computation is O(n) if n is the number of neighboring nodes of a sensor node. For example, if one node has 8 new incoming nodes at the same time, the delay time of authenticating certificates of those nodes is about 85 milliseconds. One sensor node does not have the computation overhead of evaluating trust values for other nodes, since alarm packets must be returned to the BS. The results suggest that the computation overhead of encryption/decryption, multiple hashing, MAC verification, and the *TCert* verification is reasonable in sensor networks.

Communication Overhead: The overhead was evaluated in terms of dynamic authentication, hop-to-hop secure communication, and alarm return protocols. When the routing overheads consisting of hop-to-hop communication and alarm returns are not analyzed in this study, dynamic authentication incurred the following communication cost. When one node was deployed, it advertised the certificate information (36 bytes) consisting of a key and a *TCert*. One *TCert* incurred one encryption and one MAC operation. Each *proof* message (26 bytes) comprised a key derivation, a nonce, a key, and a MAC code. The *certificate* and *proof* messages can be implemented in TinyOS since their packet sizes were smaller than the maximum size. The communication cost of broadcasting and receiving a *TCert* message was defined as $585\,\mu$J and $441\,\mu$J, respectively. The communication cost of broadcasting and receiving a *proof* message was defined as $422.5\,\mu$J and $318.5\,\mu$J, respectively. Suppose that the node had n neighbor nodes, and n was (a) plus (b). Before the node was deployed, a neighbor nodes always existed. b neighbor nodes would come after the node. Therefore, the communication cost in the node for dynamic authentication was $(585+318a+863.5b)\,\mu$J.

5.3 Performance evaluation

Considering the limited resource in sensor nodes, the cryptographic schemes applied to the proposed modules must be chosen carefully in terms of their code size and computation overhead. Table 1 indicates the characteristics of sensor nodes in the network.

Table 1: Characteristics of sensor nodes

TYPE	VALUE
CPU	8-bit, 4MHz
Memory	8K Bytes intrusion flash 512 Bytes RAM 512 Bytes EEPROM
Bandwidth	10 k bps
Communication	916MHz radio
Operating System	TinyOS
OS Code Space	3500 bytes

Figure 7 shows a comparison of system life using a cluster-based sensor network equipped with only the primary security module versus a conventional static clustering algorithm and the LEACH approach. The static clustering algorithm has cluster heads and associated clusters chosen initially and they remain fixed. Data fusion is performed at the cluster heads. For this experiment, each sensor node was initially given 0.5 J of energy. Figure 7 shows that the proposed network supporting dynamic authentication is better for the useful system lifetime compared to the static clustering algorithm. The LEACH approach is better than our system because sensor nodes deployed incrementally with dynamic authentication have to consume some overhead for neighbor node communication. The sensor nodes were dead after the 800^{th} round.

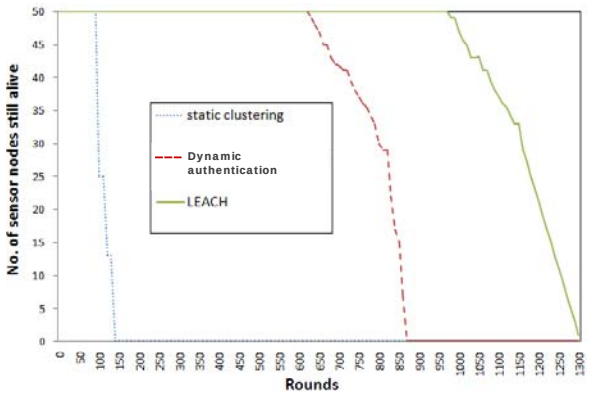

Figure 7: System lifetime using Static clustering, Dynamic authentication, and LEACH with 0.5 J/node

6 Conclusions and Future Work

This study presented an adaptive security design including two security modules to secure communication in sensor networks. The basic network security depends on the primary security module, where new incoming nodes can be authenticated by their neighboring nodes. The network can reinforce insecure sensing regions by deploying new sensor nodes. The primary security design triggers the establishment of secure links and broadcast authentication between neighbour nodes. Based on the primary security design, monitor mechanisms, alarm return protocols, and trust evaluation methods in the malicious node detection module enhance the security in sensor network. The proposed alarm return protocols efficiently look for suspect sensor nodes. Trust value evaluation performed by the BS is required to judge malicious nodes. The advance security design can achieve malicious node detection and prevention. Consequently, the network outperforms other secure architectures of sensor networks in detecting and eliminating external and internal malicious nodes. This paper also analyzes the performance of the proposed security modules in terms of their storage, computation, and communication overhead. When the dynamic authentication is performed in the sensor networks, the performance results are acceptable.

The main achievements of this study include: (1) A distributed sensor network with adaptive security policies using dynamic authentication and malicious node detection, (2) Using the proposed security modules, only legitimate nodes can carry out secure transmission in the network. Our future research plan will consist of continuation of the proposed network with several security tasks: (1) to establish a path key between any two nodes using key agreement techniques when they need to exchange information, and (2) to develop another module with different organizations to increase network connectivity.

7 References

[1] A. Perrig, R. Szewczyk, D. Tygar, V. Wen, and D. Culler, “SPINS: Security Protocols for Sensor Networks”, *Wireless Networks*, vol. 8, no. 5, pp. 521-534 (2002).

[2] B. Doyle, S. Bell, A. F. Smeaton, K. McCusker, and N. O'Connor., “Security Considerations and Key Negotiation Techniques for Power Constrained Sensor Networks”, *The Computer Journal*, vol. 49, no. 4, pp. 443-453 (2006).

[3] L.B. Oliveira, H.C. Wong, Antonio A. F. Loureiro, and Ricardo Dahab, “On the Design of Secure Protocols for Hierarchical Sensor Networks”, *International Journal of Security and Networks*, vol. 2, no.3/4, pp. 216-227 (2007).

[4] Y. P. Chen, A. L. Liestman, and J. Liu, “Clustering algorithms for ad hoc wireless networks”, *Ad Hoc and Sensor Networks 2004*.

[5] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan, “Energy-Efficient Communication Protocol for Wireless Microsensor Networks”, *Proc. of the 33rd Annual Hawaii International Conference on System Sciences*, pp. 3005-3014, Jan. (2000).

[6] C.C. Su, K.M. Chang, M.F. Horng, and Y.H. Kuo, “The New Intrusion Prevention and Detection Approaches for Clustering-based Sensor Networks”, *IEEE Wireless Communications and Networking Conference*, Mar. 2005.

[7] S. Sanyal el etc, “Security Scheme for Distributed DoS in Mobile Ad Hoc Networks”, *Lecture Notes in Computer Science*, vol. 3326, pp.541 (2004).

[8] I. Khalil, S. Bagchi, and N.B. Shroff, “LITEWORP: A Lightweight Countermeasure for the Wormhole Attack in Multihop Wireless Network”, *International Conference on Dependable Systems and Networks* (2005).

[9] K. Akkaya and M. Younis, “A Survey of Routing Protocols in Wireless Sensor Networks”, *Elsevier Ad Hoc Network Journal*, vol. 3/3 pp. 325-349 (2005).

[10] Q. Xue, and A. Ganz, "Runtime Security Composition for Sensor Networks", *Proc. of IEEE Vehicular Technology Conference*, Oct. (2003).

[11] M. Boghe, and W. Trappe, “An Authentication Framework for Hierarchical Ad Hoc Sensor Networks”, *Proc. of the 2003 ACM Workshop on Wireless Security*, pp. 79-87 (2003).