An Energy-Efficient Method for Preventing Internal Sinkhole Attacks on INSENS based WSNs using Interactive Authentications

Kyu-hyun Song
College of Information
and Communication Engineering
Sungkyunkwan University
Suwon 440-746,
Republic of Korea

Tae-ho Cho
College of Software
Sungkyunkwan University
Suwon 440-746, Republic of Korea

ABSTRACT
Wireless Sensor Networks (WSNs) are composed of many sensor nodes and a base station for collecting event information from a wide local area. However, an attacker can easily intrude into a network through external nodes by exploiting characteristics of wireless communication and the limited hardware resources of sensor nodes. Specifically, an attacker can intrude into a sensor network and launch a sinkhole attack in order to capture and redirect event reports of WSNs. Intrusion-tolerant routing for wireless sensor networks (INSENS) has been proposed to prevent sinkhole attacks via intrusion of an external node. INSENS blocks the intrusion of an external node using three symmetric keys to prevent the sinkhole attack. However, even in the presence of INSENS, a sinkhole attack can again be launched by the compromised node because the network does not account for compromised nodes. In this paper, proposed method with three steps involving the interactive authentication method prevent sinkhole attacks by a compromised node. The proposed method detects fake route request messages and drops the compromised node. Proposed method improves the number of the delivered event reports to the base station (BS) by around 65.72% when compared to INSENS. Thus, it improves the reliability of the network and reduces the average energy consumption by around 22.32% because it prevents internal sinkhole attacks.

Keywords
Wireless sensor networks, Sinkhole attack, Intrusion tolerance, Secure routing, Interactive authentication.

1. INTRODUCTION
Wireless Sensor Networks (WSNs) were proposed to collect event information in a wide area. Sensor nodes collect event data and send it to the base station (BS) [1]. The BS shows the collected events to the user. Figure 1 shows the composition of WSNs. WSNs are used in various fields such as science, medicine, and the military, but an attacker can easily intrude using an external node into the WSNs due to the nature of wireless communication and the constrained hardware resources of the sensor nodes [1, 2]. An attacker can secretly intrude into the networks through external nodes and then try to launch a sinkhole attack on the network. The sinkhole attack changes the routing path of the networks to the intruding node [3, 4].

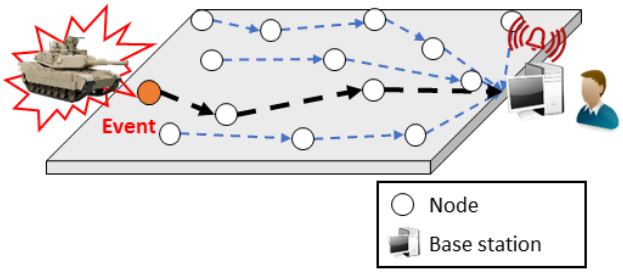

Fig 1 : Composition of WSNs

J. Deng et al. proposed Intrusion-tolerant routing for wireless sensor networks (INSENS) to prevent sinkhole attacks by intruding nodes; INSENS prevents the sinkhole attack by blocking the intrusion of the external node in advance using a Global Key (GK), Pair-wise Key (PK) and Cluster Key (CK) [4]. However, the attacker can easily compromise the sensor node due to the open environment and limited hardware resources. The attacker can again try a sinkhole attack with the compromised node because intrusion-tolerant routing for wireless sensor networks (INSENS) does not consider the compromised node [3, 5]. This paper proposes three-step algorithm involving the interactive authentication method. This paper aim is to prevent sinkhole attacks by compromised nodes in INSENS-based WSNs. The proposed method detects the fake route request (REQ) message, blocks the compromised node through the sending node verification step, the waiting step, and the interactive authentication step to prevent the sinkhole attack. Experimental results show that proposed method reduces unnecessary energy consumption and enhances reliability of event reports by preventing sinkhole attacks by compromised nodes.

This paper is structured as follows. Section 2 presents and discusses external sinkhole attacks, INSENS, internal sinkhole attacks and research motives related to proposed method. Section 3 presents a detailed description of the proposed method. Section 4 shows the results of comparing the proposed method with INSENS. Section 5 describes the conclusions and future research directions.

2. RELATED WORKS
This section describes the external sinkhole attack, INSENS, the internal sinkhole attack, and the motivations for this research.

2.1 External sinkhole attack
WSNs are composed of many sensor nodes in order to collect event data from a wide local area and the BS. The sensor nodes

collect the event data and send the event report to the BS using wireless communication [1]. The BS shows the circumstances of the field to the user with a collection of event reports. However, an external attacker easily intrudes to WSNs, exploiting hardware limitations of nodes and the wireless nature of communication. The attacker tries a sinkhole attack in order to intercept the event reports of WSNs through an external node. The attacker changes the routing path of the WSN to the attacker's own external node via the sinkhole attack. This attack is called an external sinkhole attack [6]. An illustration of an external sinkhole attack is shown in Figure. 2.

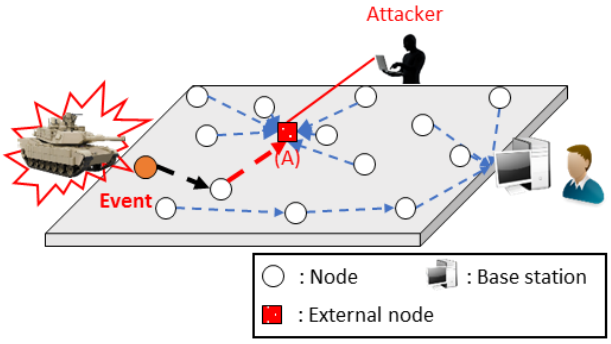

Fig 2: External sinkhole attack process

The attacker tries a sinkhole attack in order to intercept the event reports of WSNs using the external node (A). The attacker broadcasts a fake route request message to trick the BS and nodes in the best routing path in order to change the routing path of the neighboring nodes (A). The attacker can then cause chaos in the WSNs by using forged event reports [3].

Localized Encryption and Authentication Protocol (LEAP) and INSENS were proposed in order to prevent sinkhole attacks from an external node [4, 7]. LEAP prevents Intrusion of an external node using an individual key, pairwise key, cluster key, and group key. However, LEAP is difficult to implement in large-scale networks [7]. LEAP has limitations because it cannot be used to save large numbers of keys due to the limited hardware resources of sensor nodes, and it is difficult to join a new node or block nodes of the network [4]. This study is based on INSENS, which solved the aforementioned problems.

2.2 INSENS

J. Deng et al. proposed INSENS to prevent sinkhole attacks by blocking the intrusion of an external node. INSENS has two versions: one version is basic INSENS, and the other version is enhanced INSENS. This paper is based on enhanced INSENS, which resolved many vulnerabilities of basic INSENS. Enhanced INSENS blocks the intrusion of external nodes through verification of the neighbor node using a global key (GK), pair-wise key (PK) and cluster key (CK) [4].

2.2.1 Operation process

Enhanced INSENS (INSENS) is operated using echo, key exchange, route requests, and setup steps. Before deploying the nodes, INSENS injects a GK to each node in order to block an external intrusion. The echo step employs encrypted communication by using the injected GK in advance for blocking intrusion of an external node. Each node composes and broadcasts an echo message using the GK. The neighbor nodes receiving the echo message verify the message with their own GK and generate the PK with the node sending the messages. The PK is inserted into an echo-back message and is then broadcasted. The echo and echo-back message are as follows.

$$\boldsymbol{ECHO||E_{GK}\,(ID_x||nonce)} \quad (1)$$

$$\boldsymbol{ECHOBACK||E_{GK}\,(ID_y||\,nonce+1||K_{y,x})} \quad (2)$$

Here, (1) is the echo message and (2) is the echo-back message. ***ECHO*** and ***ECHOBACK*** are the message type, $\boldsymbol{E_{GK}}$ is encrypted using the GK, $\boldsymbol{ID_x}$ and $\boldsymbol{ID_y}$ are the IDs of nodes x and y, ***nonce*** is a random number, and $\boldsymbol{K_{y,x}}$ is the PK shared between node x and node y. After the echo step, each node generates a CK for confidentiality of the network, and forwards its own CK to each neighbor node using the PK of each of its neighbor nodes as well as its own PK. After the key exchange step, the BS generates an REQ message by using its own CK to establish a routing path setup. The structure of the REQ message is as follows.

$$\boldsymbol{REQ||ID_s||E_{CK\_s}(OHC||ID_{BS})} \quad (3)$$

Here, (3) is the REQ message, ***REQ*** is the message type, $\boldsymbol{ID_s}$ is the ID of the sender node, ***CK_s*** is the CK of sender node, ***OHC*** is the one-way hash chain (OHC) for preventing re-use and looping of the message, and $\boldsymbol{ID_{BS}}$ is the ID of the BS, i.e., the terminus of the desired routing path. The routing path setup process using the REQ message is shown in Figure 3.

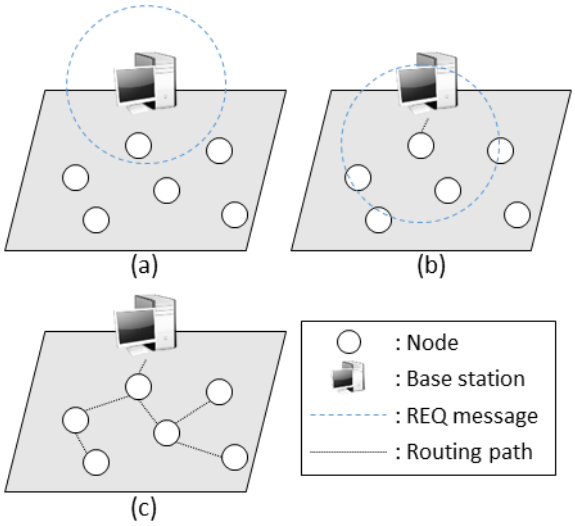

Fig 3: Routing path setup process of INSENS

After the key exchange step is completed, the BS broadcasts a REQ message in order to setup the routing path and it increments the OHC by 1. When it receives the REQ message, the neighbor nodes verify the message using the CK of the sender node and their own OHC (a). The nodes, which finished the authentication step of REQ messages, broadcast the encrypted REQ message using its own CK, and increments its own OHC by 1 (b). WSNs repeat this process in order to set the routing path [4].

2.2.2 Internal sinkhole attack

The sensor nodes of WSNs can be very vulnerable to compromise because they operate in the open environment. The sensor nodes are compromised by an attacker through these vulnerabilities. The attacker can gain the security information and network control with a compromised node [6]. The attacker tries the sinkhole attack using the compromised node internal to the WSN. This type of attack is an internal sinkhole attack. The attacker has limitations for attempting an internal sinkhole attack of the INSENS. The limitations are as follows.

- The attacker must compromise the neighbor nodes of the BS in order to obtain the security information of the BS

because each node only has the security information of its own neighbor nodes [8].

- The sinkhole attacks for spoofing the role of the BS are prevented by the BS even if the attacker obtains the security information and control of compromised neighbor nodes of the BS.

Therefore, the attacker resorts to sinkhole attacks for spoofing the shortest path. The sinkhole attack exploits characteristics of INSENS, which sets up the routing path only with the flooding of messages in INSENS. The attacker broadcasts the fake REQ message for spoofing of the shortest path. The structure of a fake REQ message is as follows.

$$\boldsymbol{REQ||ID_C||E_{CK\_C}(OHC||ID)} \quad (4)$$

Here, (4) is the fake REQ message, $\boldsymbol{ID_C}$ is the node ID of the compromised node, and $\boldsymbol{CK\_C}$ is the CK of the compromised node. The fake REQ message is the same as the normal REQ message..

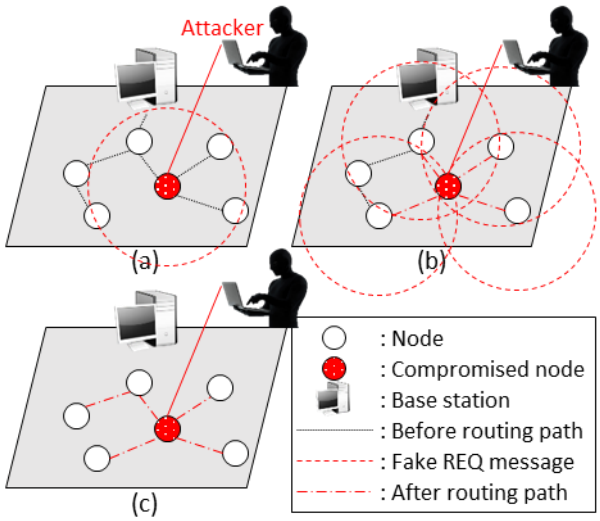

Fig 4: Sinkhole attack occurred into INSENS

However, the INSENS mistakes the fake REQ message for the normal REQ message because the network does not recognize the node as compromised The routing path of WSNs is changed to the compromised node due to the fake REQ message. This is an internal sinkhole attack. The internal sinkhole attack process of INSENS-based WSNs is shown in Figure. 4. The attacker broadcasts a fake REQ message for the purpose of an internal sinkhole attack (a). The neighbor nodes change the routing path to the compromised node because it mistakes the fake REQ message for the normal REQ message (b). The routing paths of WSNs are changed by repetition of this process (c).

2.3 Problem statement

The preceding researchers are conducting extensive studies on the defense and detection of internal sinkhole attacks, such as "Launching a Sinkhole Attack in Wireless Sensor Networks; the Intruder Side"[9], "Detection & Prevention of Sinkhole Attack on AODV Protocol in Mobile Adhoc Network"[10], etc. [11]. However, detection methods that are based on the number of reports have difficulty detecting an internal sinkhole attack because some of the reports can be delivered to the BS, according to the type of internal sinkhole attack. Also, detection methods based on routing tables are not appropriate for INSENS because it does not save routing tables. In addition, the preceding studies are slow to initially react because they detect an internal sinkhole attack only after the routing path is changed. Therefore, we needs an efficient method for preventing the sinkhole attack in advance.

A previously proposed secure routing method checks the status of the information for the neighbor node to resolve internal sinkhole attacks [12]. However, this method has difficulty detecting internal sinkhole attacks when the node density is low. This paper propose a method to securely prevent the sinkhole attack without being influenced by the density of sensor nodes.

3. Proposed method

This section will describe the assumptions, the attack model, and the operation process of the proposed method, and then analyze the results. This study supposes the following.

- WSNs are constructed using the MICA2 mote [8].
- The BS performs two-way communication with all sensor nodes.
- When setting the first routing path, the nodes are not compromised.

Attack models include external and internal sinkhole attacks. External sinkhole attacks attempt to attack without security information via external node intrusion. An external sinkhole attack occurs when the external nodes intrude in a random location and attempt to attack without security information. The internal sinkhole attack is compromised by selecting one of the normal sensor nodes, and uses the security information to perform a sinkhole attack [12].

3.1 Operation process

The proposed method detects a false REQ message through the BS and each node, which has interactive authentication, and blocks it to prevent the sinkhole attack. BSs generate each pair-wise key (BS_PK) and share it with each node for interactive authentication before the network is configured. The REQ message checks to determine whether it is real or fake using a three-step verification process. The verification steps are as follows.

1) Sender node verification (SNV)
2) Wait for REQ messages from some of the Neighbor nodes (WRN)
3) Perform Interactive Authentication (PIA)

The algorithm flowchart of the proposed method is shown in Figure 5.

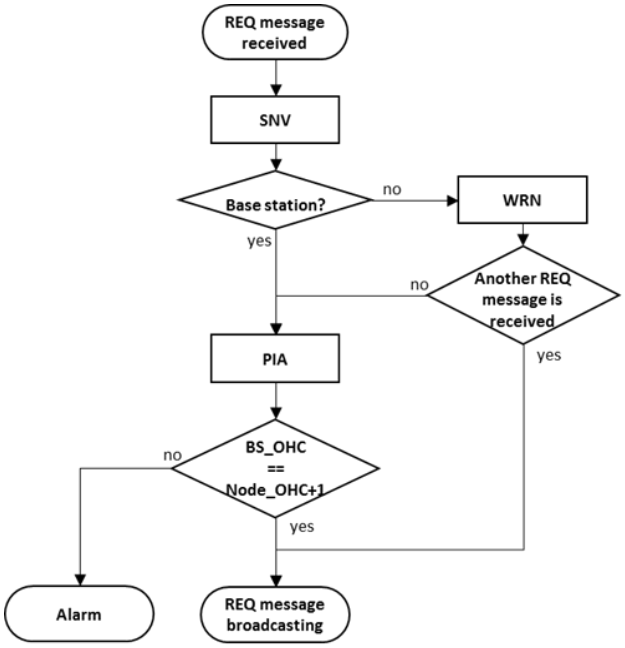

Fig 5: proposed algorithm flowchart

The node receiving the REQ message executes the SNV step to check the sender node of the REQ message. If the sender node of the message is the BS, it verifies the message through the PIA step otherwise it executes the WRN step. The WPN step uses the message to flood using the characteristics of wireless communication. Nodes receive the REQ message from another neighbor node due to the flooding of the messages. If the receiving node receives the REQ messages from the other nodes, the receiving node sets the routing path and then broadcasts the REQ message, otherwise it executes the PIA step. The PIA step verifies the REQ message via the interactive authentication between the BS and each node. The interactive authentication is selected by one node of the network in the standby step. This node attempts to initiate interactive authentication with the BS. Selection priorities are as follows.

1) The parent node on an existing routing path.

2) The distance to the nearest node from the node requesting the route

Selected nodes generate the authentication REQ (REQ_A) message using their own OHC + 1 for interactive authentication with the BS, and then broadcast to the BS. The BS compares the OHC + 1 values of the received messages and their own OHC value. The structure of the REQ_A and ACK message is as follows.

$$REQ\_A||ID_S||E_{BS,S}(OHC+1||nonce) \quad (5)$$

$$ACK||ID_{BS}||E_{BS,S}(nonce+1) \quad (6)$$

Here, (5) is the REQ_A message and (6) is the ACK message. ***REQ_A*** and ***ACK*** are the type of message, ID_S and ID_{BS} are the IDs of the node requesting verification and the ID of the BS, respectively, $E_{BS,S}$ is encrypted by the PK of the node S and the BS, ***OHC + 1*** is the OHC value increased by one for message verification, and ***nonce*** is used to prevent re-use of the message.

3.2 Example of the proposed method

The proposed method uses the flooding characteristics of broadcasting messages to save energy. When the normal REQ message is broadcast, the operation of the proposed method is shown in Figure 6. The BS broadcasts the REQ message and its own OHC is increased by one.

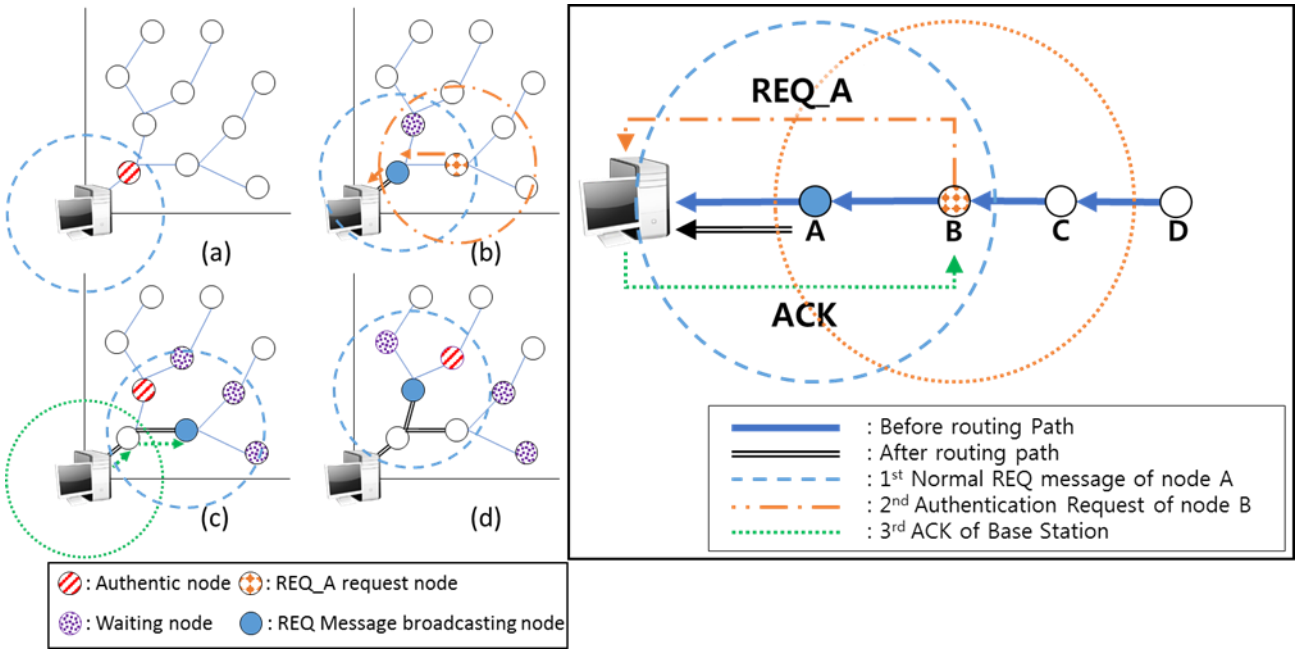

Figure 6: Operation of the proposed method by a normal REQ message

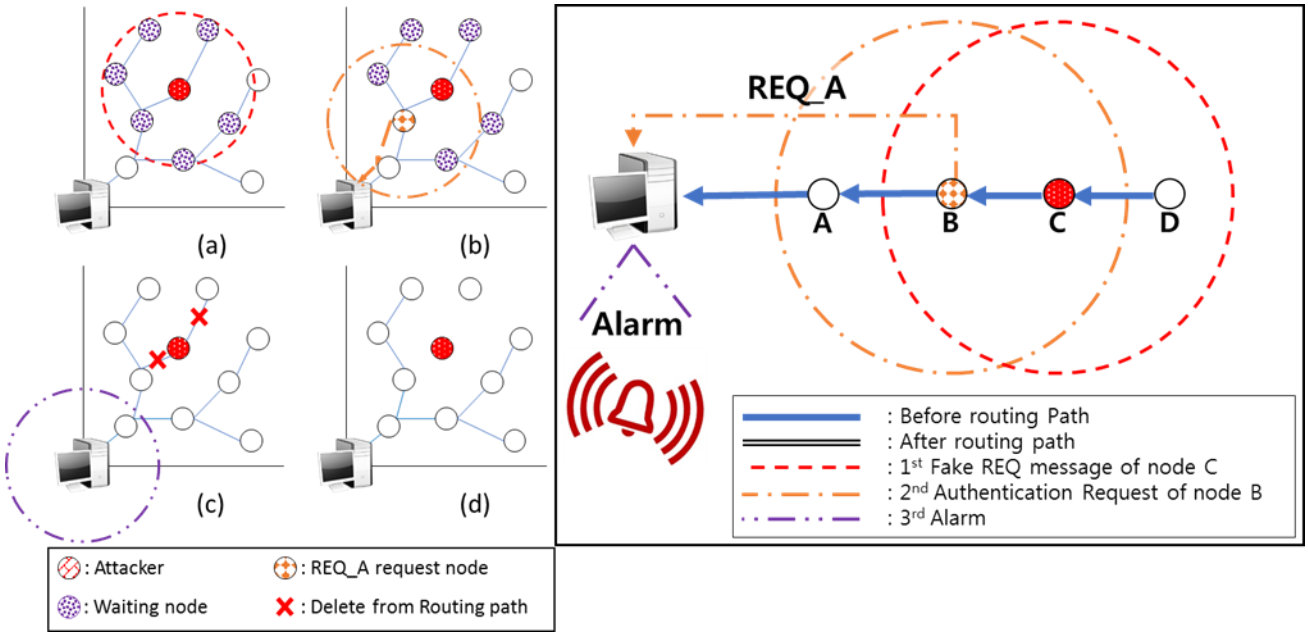

Fig 7: Operation of the proposed method by a fake REQ message

The node A, which received the message, checks the sender node and sets up the routing path via the PIA step because the sender node is the BS (a). Afterward, node A broadcasts an REQ message. The neighbor nodes of node A check the sender node via the CSN step and execute the WRN step to verify the message. Node B, which first received the REQ message, executes the PIA step because the neighbor nodes of node A have not received the other REQ message. Node B inserts its own OHC +1 and nonce value to the REQ_A message and it broadcasts (b). The BS compares its own OHC value and the OHC value of the received REQ_A message. The BS inserts the nonce + 1 into the ACK message because the two OHC values are the same, and then broadcasts it. Node B receives the ACK message and sets the routing path. Then, node B broadcasts the REQ message. The nodes of the waiting step trust the REQ message because they receive the REQ message of node B. Therefore, the nodes set the routing path to transmitting node A of the first received message REQ (c). The nodes, which are finishing the routing path setup, broadcast the REQ message. Afterward, the nodes set the routing path without the PIA step because they receive a number of REQ messages (d). When a fake REQ message is broadcast, the operation of the proposed method is shown in Figure 7. The attacker broadcasts the fake REQ message for the internal sinkhole attack using the compromised node C. Neighbor nodes receive the fake REQ message and execute the CSN step. The neighbor nodes execute the WRN step because the originating node is not the BS (a). Parent node B of node C on the existing routing path performs the PIA step because the different REQ message is not received during the predetermined time. The BS receives the REQ_A message and compares the OHC of the message to its own OHC, and it sounds the alarm because the two OHC keys are different. The nodes of the WRN step and node B receive the signal to sound the alarm of the BS and they delete node C in their neighbor node list.

The proposed method has the expected effect, with the following features.

- It securely prevents internal sinkhole attacks without being influenced by the density of nodes.
- It increases the number of messages which safely arrive at the BS.
- It reduces unnecessary energy consumption of the WSNs through the prevention of sinkhole attacks.

3.3 Experiment results

This study was implemented in a simulation environment using C ++. The simulation environment has INSENS-based WSNs and WSNs using the proposed method. Attack models include external and internal sinkhole attacks. The field size is 1000 X 1000m^2, the number of BSs are two. The number of nodes are 900, and the communication range of the node and the BS is 75m [8]. The energy consumed to send a message is 16.25μJ per byte and to receive a message is 12.5μJ per byte. The energy consumed to encrypt and decrypt the message is 9μJ [6]. The waiting time for the WRN step is 0.00046 seconds after the message reception [8]. Figure 8 shows the energy consumption of the proposed method and INSENS.

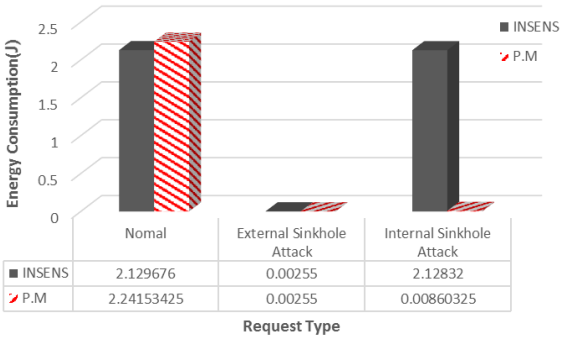

Fig 8: Energy consumption according to each message

The proposed method results in about 5.25% greater energy consumption than INSENS for the normal route request. However, it is possible to reduce the energy consumption due to prevention of both external and internal sinkhole attacks. On the other hand, INSENS increases the energy consumption because internal sinkhole attacks are not prevented. Figure 9 shows the total energy consumption of the INSENS-based WSNs and the

WSNs based on the proposed method versus the Fake Request Ratio.

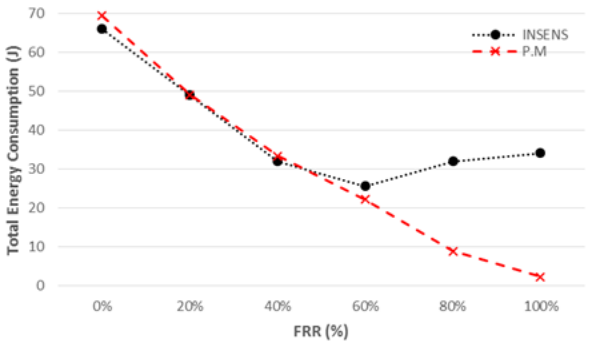

Fig 9: Energy consumption according to FRR

The OHC value of the nodes in INSENS are increased because it does not prevent internal sinkhole attacks. So, the energy consumption is reduced because they do not use the normal routing path configuration. Therefore, When the FRR is more than 50%, the energy consumption of INSENS is increased again. On the other hand, When the FRR is more than 0%, the proposed method consumes about 5.11% more energy than INSENS due to interactive authentication. However, when the FRR increases, the energy consumption of the proposed method is reduced by blocking internal sinkhole attacks. As a result, the proposed method reduces the average energy consumption up to about 22.32% verses INSENS. Figure 10 shows the compromised node blocking rate according to the FRR.

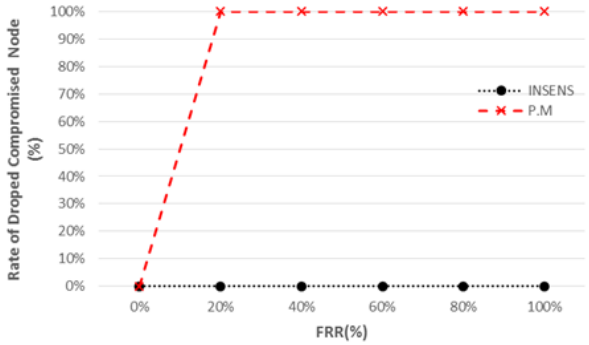

Fig 10: blocking compromised node rate according to FRR

INSENS can't block the compromised node because it does not detect the internal sinkhole attack. However, the proposed method blocks the compromised nodes due to the detection and prevention of an internal sinkhole attack. Therefore, it is possible to build a secure network via the proposed method. Figure 11 shows the number of event reports that arrived at the BS. The number of delivered reports to the BS tapers off with an increasing FRR using INSENS. However, the proposed method delivered the event reports to the BS, except for the dropped event reports, because the compromised node was blocked by the proposed method. According to the experimental results, proposed method improves the reliability of WSNs because it improves the number of delivered event reports to the BS by about 65.72% relative to INSENS, and it extends the lifetime of WSNs by reducing the average energy consumption by up to around 22.32%, due to the prevention of internal sinkhole attacks.

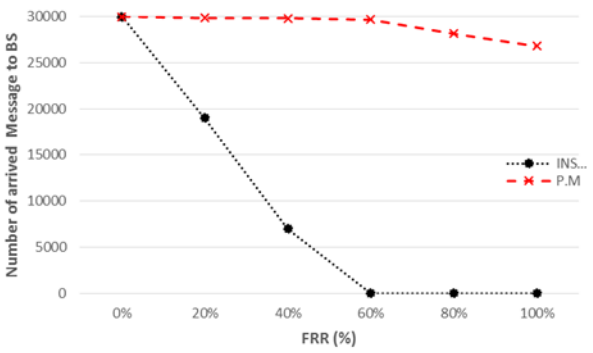

Fig 11: Number of arrived event reports according to FRR

4. CONCLUSIONS AND FUTURE WORK

In this paper, the proposed method, which uses the three-step interactive authentication algorithm, prevents internal sinkhole attacks. The proposed method shares the BS_PK in advance between each node and the BS, and the key is used in the proposed algorithm to prevent internal sinkhole attacks. The proposed method prevents the internal sinkhole attack regardless of the node density. Proposed method reduces the average energy consumption by up to about 22.32% and it improved the number of delivered event reports to the BS by up to about 65.72%, relative to INSENS. In this study, we were able to improve the reliability of wireless sensor networks and reduce the energy consumption by using the proposed method to prevent sinkhole attacks regardless of the node density. Our Future research will be to study prevention of a cooperative routing attack using multiple compromised nodes.

5. ACKNOWLEDGMENTS

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (No. NRF-2015R1D1A1A01059484)

6. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A survey on sensor networks," Communications Magazine, IEEE, vol. 40, pp. 102-114, 2002.

[2] K. Akkaya and M. Younis, "A survey on routing protocols for wireless sensor networks," Ad Hoc Networks, vol. 3, pp. 325-349, 2005.

[3] X. Du and H. Chen, "Security in wireless sensor networks," Wireless Communications, IEEE, vol. 15, pp. 60-66, 2008.

[4] J. Deng, R. Han and S. Mishra, "INSENS: Intrusion-tolerant routing for wireless sensor networks," Comput. Commun., vol. 29, pp. 216-230, 2006.

[5] E. Shi and A. Perrig, "Designing secure sensor networks," Wireless Communications, IEEE, vol. 11, pp. 38-43, 2004.

[6] F. Ye, H. Luo, S. Lu and L. Zhang, "Statistical en-route filtering of injected false data in sensor networks," Selected Areas in Communications, IEEE Journal On, vol. 23, pp. 839-850, 2005.

[7] S. Zhu, S. Setia and S. Jajodia, "LEAP : Efficient security mechanisms for large-scale distributed sensor networks," ACM Transactions on Sensor Networks (TOSN), vol. 2, pp. 500-528, 2006.

[8] (Accessed: Jan. 11, 2016). MICAz: wireless measurement system. Available: http://trl.iba.edu.pk/Memsic-set-2.pdf.

[9] I. Krontiris, T. Giannetsos and T. Dimitriou, "Launching a sinkhole attack in wireless sensor networks; the intruder side," in 2008 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, 2008, pp. 526-531.

[10] N. Gandhewar and R. Patel, "Detection and prevention of sinkhole attack on AODV protocol in mobile adhoc network," in Computational Intelligence and Communication Networks (CICN), 2012 Fourth International Conference On, 2012, pp. 714-718.

[11] E. C. Ngai, J. Liu and M. R. Lyu, "An efficient intruder detection algorithm against sinkhole attacks in wireless sensor networks," Comput. Commun., vol. 30, pp. 2353-2364, 2007.

[12] Kyu-Hyun Song and Tae-Ho Cho. Secure route determination method to prevent sinkhole attacks in INSENS based wireless sensor networks. Journal of Korean Institute of Intelligent Systems 26(4), pp. 267-272. 2016.