Automatic Threshold Reset Scheme using a Double Fuzzy System for Improvement of Detection Rate in a Probabilistic Voting-based Filtering Scheme of WSNs

Sang-hyeok Lim
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) consist of several sensor nodes and base stations that collect information through sensors located in a large area. However, WSNs have disadvantages in that they are easily damaged by an attacker because of their random and unattended deployment in an open environment, where individual management is difficult. An attacker can execute a false report injection attack or a false vote injection attack through compromised nodes. The probabilistic voting-based filtering scheme (PVFS) is a scheme to prevent these two kinds of attacks. Before sending the report, the proposed method selects the validation node, judges the validity of the report, and filters the set threshold values. Threshold settings determine the security and lifetime of the network, so setting the appropriate security values is important. In this paper, we propose a fuzzy-based PVFS method that detects the aggressiveness of the attacker and sets the appropriate threshold values. This paper confirms that the proposed method improves the energy efficiency and detection ability of the network.

Keywords

Wireless sensor networks, False report injection attack, False vote injection attack, Secure routing, Fuzzy system, Interactive authentication.

1. INTRODUCTION

WSNs are composed of many sensor nodes and a base station (BS). When an event occurs, the sensor node detects the event and reports it over multiple hops of the sensor nodes to the BS [1]. These WSNs are used for data collection and event detection in various fields such as military systems, home networks, and forest fire monitoring [2]. However, many applications have limited computational power and low energy, and they are easily compromised by attackers because they are randomly distributed in an open environment that operates independently and is difficult to individually manage [3, 4]. Attackers exploit these vulnerabilities to attack WSNs by injecting reports containing false information or false votes. Fig.1 shows a schematic of these attacks. A false report injection attack is one that injects a report about a non-existent event through the compromised sensor node. The goal of this attack is to exhaust the energy resources of the nodes on the propagation path and generate a false alarm at the BS. The false vote injection attack injects false votes into legitimate reports, thereby preventing the legitimate report from reaching the BS. Li and Wu proposed a probabilistic voting-based filtering scheme (PVFS) [5] to prevent such attacks. In PVFS, all nodes constitute a network that exploits cluster-based organization. When a cluster head (CH) recognizes an event, it generates a report on that event. Then, the member nodes judge the authenticity of the report and generate their own message authentication codes (MACs), alternatively referred to as votes in PVFS. CH randomly selects votes and inserts them into the report. Verification nodes on the path use MAC and threshold values to defend against attacks. An attacker can attempt a false report injection attack and false vote injection attack through the compromised member node. In the existing PVFS, the user does not know the method of attack and cannot set an appropriate threshold value. Incorrect threshold settings result in detection performance degradation for attacks and waste energy. Therefore, setting the appropriate threshold value is important, and the security protocol should reflect the opinions of network users. This paper proposes technique that uses a fuzzy system to determine the method of attack and automatically reset the appropriate threshold value to reflect the expected method of attack. Using the proposed method, it can be confirmed that the detection rate of the false vote injection attack increased and energy waste decreased. Related works are described in Section 2. The problems with existing schemes are described in detail in Section 3, and the proposed method is presented. Section 4 describes the experimental environment and results. Finally, Section 5 provides a conclusion and a discussion of future work.

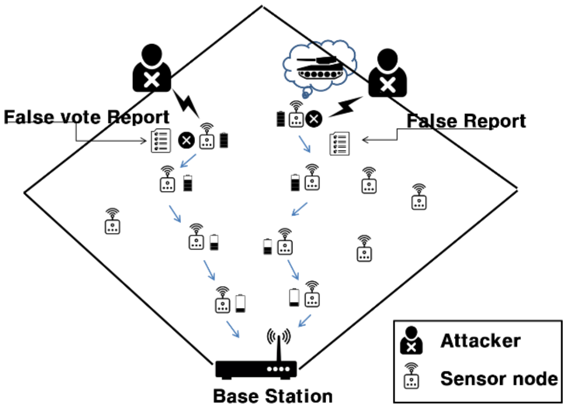

Fig.1 False report and false vote injection attacks.

2. RELATED WORK

This section describes the false report attack and the false vote injection attack on WSNs, a fuzzy logic system, and PVFS, a defense mechanism against such attacks.

2.1 False report attack

A false report attack is one that injects a false report on an event that does not exist in the network [6, 7]. These are mainly false positive attacks (FPAs). In the absence of a defense system, false reports can arrive at the BS, trigger false alarms, and cause unnecessary energy consumption. The attacker attaches a false MAC to the report, generated using the MAC of the node itself and the s-1 false key. Because of this feature, false reports contain a large number of false votes. DEF, SEF, CCEF, IHA, and BECAN are examples of defense schemes against such false report attacks [8 -12].

2.2 False vote injection attack

A false vote injection attack is one that prevents a legitimate report from reaching the BS. This is mainly called a false negative attack (FNA). The compromised member node makes a false vote on the legitimate report, so that the wrong information is written in the report. Because of this feature, the false MAC-injected report contains a small number of false votes. A false vote in a legitimate report will cause the verification node to regard the report as a false report and drop the report during path verification. This attack causes the BS to lose important information.

2.3 Fuzzy logic system

One of the advantages of fuzzy rule systems is that they can be used for pseudo-reasoning, which is very useful if there is uncertainty in the inference process or if the data are ambiguous [13-15]. There are uncertainties in attacks that occur at the application layer of the actual WSN because of their types and frequency. It is also difficult to determine accurate attack figures when using a limited range of thresholds. Therefore, similar inferences are needed to deal with this fuzzy information. The inference of the fuzzy logic system method uses the min-max synthesis method of the Mamdani model, and the reverse fuzzy method for the output uses the gravity center method [16]. The Mamdani type fuzzy inference process proceeds in four steps. The first stage is the fuzzing of input variables. Here, it determines how many input values belong to each of the appropriate fuzzy sets. The rule evaluation stage proceeds as follows. It takes fuzzy input and obtains the number representing the evaluation result of the previous case. Fuzzy system applies this number to the belonging function of the latter case. Step 3 integrates the rules as an output and combines the membership functions of all rules after the rule set in the previous step into one fuzzy set. Finally, reverse fuzzy logic is performed in Step 4. For the output value to be a number, the input in the deserialization process is a combined output fuzzy set, and the output is represented by a single number. At this stage, the center of gravity method is used.

2.4 PVFS

To cope with false report injection and false vote injection attacks in WSNs, the proposed PVFS uses a true threshold value (Tt) and a false threshold value (Tf) to detect and filter false reports and false vote injection reports at validation nodes. Fig.2 shows the report generation and verification node selection process. In the initial network configuration, the nodes are divided into cluster units, and the CHs responsible for report generation are selected for each cluster.

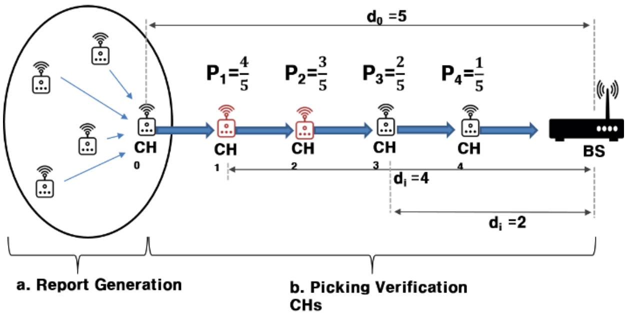

Fig.2 Report generation and verification node choosing processes.

The verification nodes among the CHs are probabilistically selected to verify the report. The probability p uses the distance d_0 between the BS and the event cluster and the distance d_i between BS and CH_i. The verification node selection process is shown in Fig.2-b. Fig.3 shows the key allocation step in which the BS divides the key pool into N partitions and delivers them to each CH. Each partition contains L keys that are the size of the cluster. The CH uses one of the keys in the partition as its own key and distributes the remaining L-1 keys to the member nodes. A key is allocated to the member nodes according to the partition of the key pool.

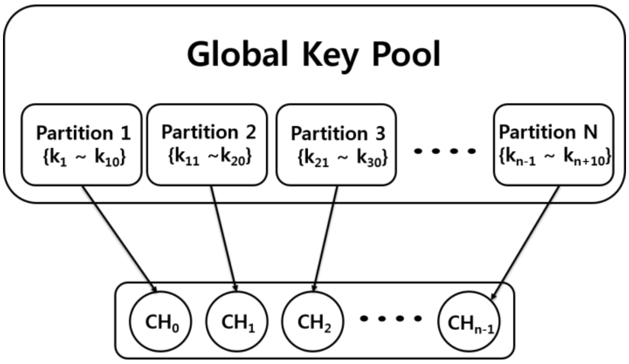
Global Key Pool
Partition 1 $\{k_1 \sim k_{10}\}$
Partition 2 $\{k_{11} \sim k_{20}\}$
Partition 3 $\{k_{21} \sim k_{30}\}$
Partition N $\{k_{n-1} \sim k_{n+10}\}$
CH_0
CH_1
CH_2
CH_{n-1}

Fig.3 Key distribution process.

The node selected as the verification node stores the keys of the member nodes of the event occurrence cluster one by one. In the report generation step, CH generates a report on an event and broadcasts it to member nodes. The member nodes confirm this, and if the report is judged as a normal report, the MAC created by its own key is transmitted to the CH. CH extracts a predetermined number of MACs received from member nodes and adds them to the report. In the report verification process, the verifying nodes compare their own keys with the keys in the report. If they have the same key, they determine the MAC of the report. If the MAC value generated by the same key is different, the vote is regarded as false, and Tf is increased. In the filtering process, if the false count reaches the threshold value, the report is judged to be a false and is immediately dropped. If the true count value reaches the threshold value, the report is considered legitimate and is sent to the BS without further validation.

3. PROPOSED METHOD

This section presents the problems associated with the existing PVFS and describe our proposed method for solving the problem.

3.1 Problem statement

In WSNs, sensor nodes are placed in open areas and are easily damaged by attackers. An attacker can perform two types of attacks through the compromised node: FNA and FPA. Li and Wu proposed PVFS to cope with these two attacks. PVFS is a defense against FNA and FPA, and it is used as the most effective defense scheme on the assumption that an attacker will attack using both types of attacks. However, users do not know what kinds of attacks are happening, and they therefore have difficulties in setting thresholds of PVFS. The network has low security and poor energy efficiency when the user misidentifies the method of attack and sets the wrong Tt and Tf values. Setting Tt high and Tf low is efficient for FPA defense, but the FNA detection rate is low. Conversely, when the thresholds are reversed, the detection rate of the FPA decreases, while the detection rate of the FNA increases. This trade-off is a disadvantage of PVFS that must be tolerated to defend against two kinds of attacks at the same time. Therefore, the user must set an appropriate threshold value to prevent attacks.

3.2 Assumption

In an open area, the sensor field is cluster-based, and no node is attacked in the node initial deployment phase. The attacker can conduct both FNA and FPA. The verification nodes are not corrupted.

3.3 System overview

Fig.4 shows the operation of the proposed method. Each validation node examines the vote of the report with its own key, and if it is judged to be a false vote, the node increases the Tf of the report by 1 and increases the false vote count (FVC) of the node by 1. These operations are repeated for a fixed cycle. During the cycle, the network filters the attack via PVFS, and verification nodes store the FVC value. When the cycle ends, the BS collects the FVC values stored by each verification node. The number of attacks occurring during one cycle is the sum of the number of reports whose Tf value is not 0 and the FVC collected from the node. Three input values enter the fuzzy system and are used to identify the type of attack. In the second fuzzy system, the importance of energy, defensive posture, and the type of attack (which is the output of the first fuzzy system) are provided as inputs. The second fuzzy system derives the appropriate threshold set as an output based on the three input values. The new threshold set obtained through this process is applied to the next cycle, and this process is repeated.

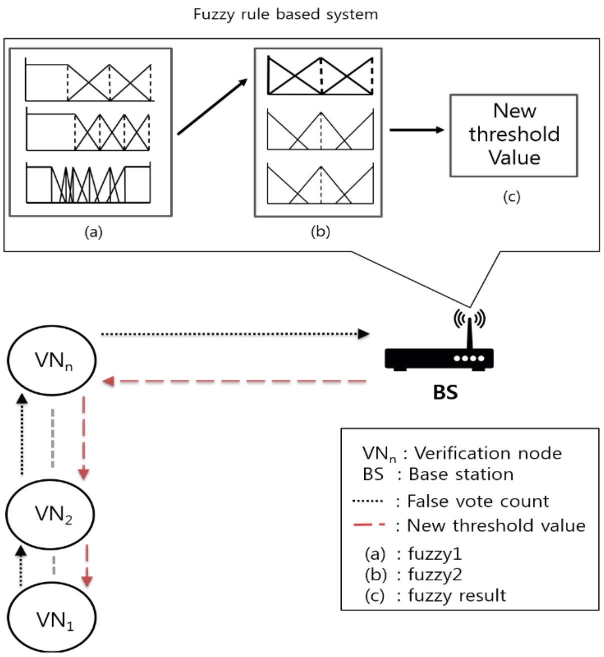

Fig.4 Process of the proposed scheme.

3.4 System input and output

This section describes the input and output values of the fuzzy system and describes the reason for and importance of each input. Fig.5 shows an overview of the fuzzy rule-based system.

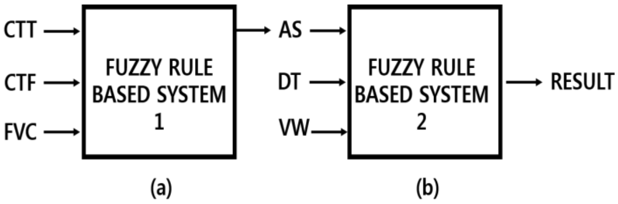

Fig.5 Fuzzy rule-based system overview.

The following variables are used as inputs for the proposed method, and the reasons for selecting the variables are described in detail.

- Current True Threshold value (CTT): In PVFS, the Tt threshold value is used to reduce energy consumption for verification by transmitting information to the BS through the nodes in the path, without unnecessary verification of the normal report. Higher Tt values require more hops, so the report moves to increase the Tf value.
- Current False Threshold Value (CTF): In PVFS, the Tf threshold value is used to detect false votes contained in a report. If the count reaches a threshold, the report is judged as false and is dropped. As the Tf value is increased, more false votes are counted and applied to fuzzy logic.
- False Vote Count (FVC): The biggest difference between a false report and a false vote-injected report is the difference in the number of false votes included in the report. Whenever a false vote is detected at the verification node, the value is accumulated and applied to the fuzzy logic as the most important indicator of the attack type.
- Attack style (AS): The AS is an output value from the primary fuzzy system. The style of attack is defined as the ratio of false report injection attacks and false vote injection attacks and is the most important factor in setting new thresholds. If the percentage of false report attacks is high, Tt goes up and Tf goes down. If the rate of false vote injection attacks is high, the thresholds respond in opposite ways.
- Detection tendency (DT): Detection tendency is a value that determines which attacks are primarily detected by network users based on the detection performance of false report attacks and the detection performance of false vote-injected report attacks. This input value affects the settings of the new Tt and Tf.
- Vibration width (VW): Vibration width indicates the variation of the attack style of the attacker. For example, if the rate of false report attacks per cycle changes from 100% to 0% to 100% to 0%, the VW will be set to a high value. Also, if the attack rate changes by a small amount from 100% to 70% to 50%, the VW is set to a low value. The threshold value behaves such that the tradeoff cost increases as the variation width increases. So, the threshold values are included in the fuzzy input value because they affects the new threshold value.

3.5 System membership function

This section describes fuzzy membership functions and rules. Two fuzzy systems were used in the proposed scheme. The output of the first fuzzy system is the input of the second fuzzy system. Fig.6 shows the membership function of the first fuzzy system.

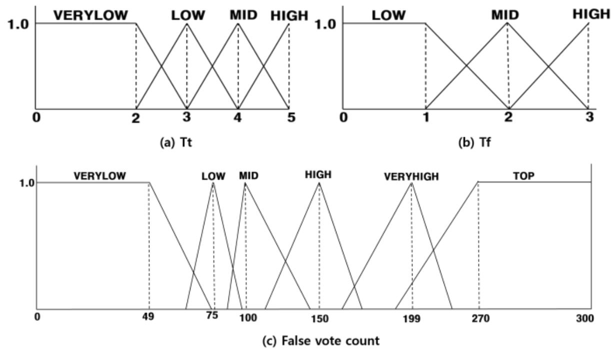

Fig.6 Fuzzy membership function1.

- (a) True vote threshold = {VL (LERYLOW), L (LOW), M (MID), H (HIGH)}
- (b) False vote threshold = {L (LOW), M (MID), H (HIGH)}
- (c) False vote count = {VL (LERYLOW), L (LOW), M (MID),HIGH, VH (VERYHIGH), T (TOP)}

Table 1 shows some of the fuzzy rules used in the system. There are 72 rules in total. If Tt is LOW, Tf is LOW, and the number of false votes is LOW, the system will determine the attack propensity within the LOW range. This means that the attack will likely be a false vote injection. On the other hand, if Tt is VERY HIGH, Tf is HIGH, and the number of false votes is TOP, the system will determine the attack propensity within the HIGH range. This means that the attack will likely be a false report.

Table 1 : Fuzzy rules

No	Input			output
	CTT	CTF	FVC	OUT
1	LOW	LOW	LOW	MID
10	LOW	MID	VERYHIGH	HIGH
23	MID	LOW	TOP	HIGH
42	HIGH	MID	VERYLOW	LOW
71	VERYHIGH	HIGH	TOP	HIGH

Figure 7 shows the membership function of the second fuzzy system, and the output of fuzzy system 1 becomes the input of fuzzy system 2. As shown, 2-a is the output value of the first fuzzy system and indicates the attack type.

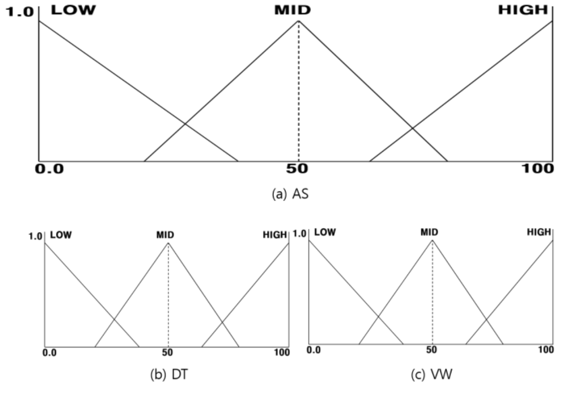

Fig.7 Fuzzy membership function2.

- Attack style = { L (LOW), M (MID), H (HIGH)}
- Detection tendency = {L (LOW), M (MID), H (HIGH)}
- Vibration width = { L (LOW), M (MID),HIGH}

Table 2: Fuzzy rule 2

No	Input			output
	AS	DT	VW	OUT
1	LOW	LOW	LOW	VERY LOW
5	LOW	MID	HIGH	LOW
10	MID	LOW	MID	MID
19	HIGH	LOW	MID	VERY HIGH
26	HIGH	HIGH	HIGH	LOW

Table 2 shows some of the fuzzy rules used by the system to set new thresholds. A total of 27 rules are applied, and the inputs are AS, DT, and VW. AS provides the output of the first fuzzy system to inform the user of what types of attacks are most likely to occur, and DT is applied to reflect the network user's tendency to protect against attacks such as FNA and FPA. Finally, VW is the variation range of the attack type. Larger values result in more stable output settings.

4. EXPERIMENTAL RESULTS

This section describes the simulation environment and its results. Experimental results are provided separately for the output of the first fuzzy system and the output of the second fuzzy system.

4.1 Experimental environment

This section shows the simulation results of energy efficiency and security for the proposed method and compared them with those of PVFS. To show the efficiency of the proposed method, The proposed scheme assumes the following experimental environment [7]; these values are also listed in Table 3. Each time 200 attacks occurred, the style changed randomly, and a threshold reset occurred after every 100 attacks.

Table 3: Parameters for the Experiments

Parameter	Value
Field size	1000 x 1000(m^2)
Number of nodes	100-4000
Number of experiments	1000
L	10
S	5
Tt	3-5
Tf	1-3

4.2 Experimental result

The simulation results for the 9 commonly used threshold sets are shown in the following graphs. Fig.8 shows the detection success rate according to the ratio of false reports per cycle. The fuzzy system was used to confirm the attack type with 97% accuracy. Figure 9 shows the detection rate according to Tf value. Since Tf = 1 is rarely used, it is excluded from the simulation. The attack type is the ratio of FPA to FNA corresponding to the attacks in this experiment. The attack style was based on FPA such that a value of 100% means that only FPA occurred.

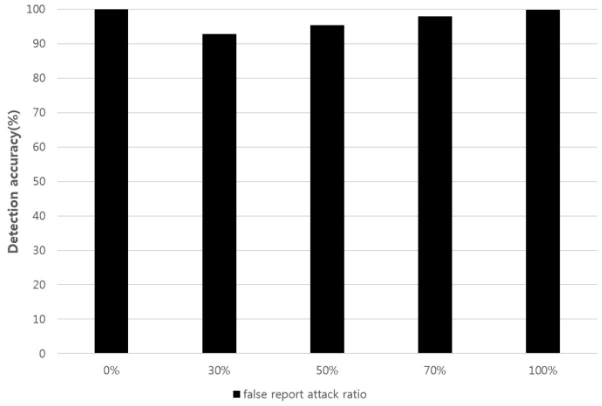

Fig.8 Success rate of attack style tendency detection

Fig.8 shows the detection accuracy of the attack style. The styles were divided into 5 types: 0%, 30%, 50%, 70%, and 100%. The advantage of the fuzzy system is that it can also determine the median of these discrete values. The overall accuracy is 97%, indicating that the overall reliability is high.

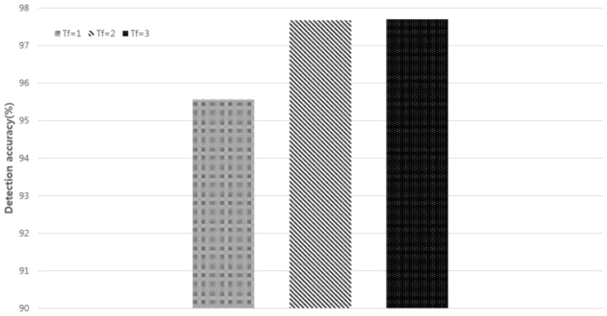

Fig.9 Accuracy according to Tt value.

Fig.9 shows the detection accuracy of the fuzzy system with respect to the Tf value. As the Tf value decreases, the accuracy decreases. The reason for this is that a larger number of false votes results in a more accurate fuzzy membership function. But, if the Tf is low, false report injection attacks and false negative voting attacks are quickly filtered out, and an insufficient number of false votes are collected.

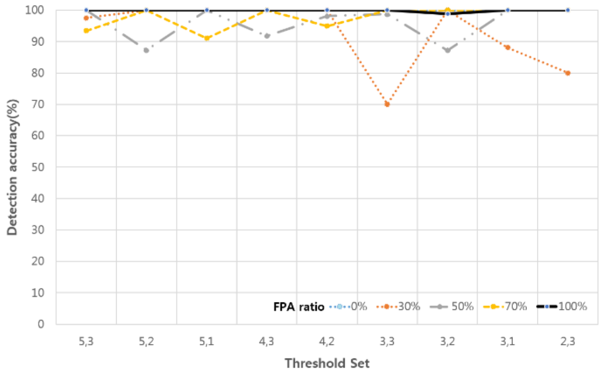

Fig.10 Overall detection accuracy.

Fig.10 shows the accuracy of the fuzzy system for the nine threshold sets. The highest accuracy is observed for attack styles of 100% or 0%. The proposed scheme applies only five threshold sets, which are the most commonly used in real situations. The values are (4,1), (4,2), (3,1), (3,2), and (3,3).

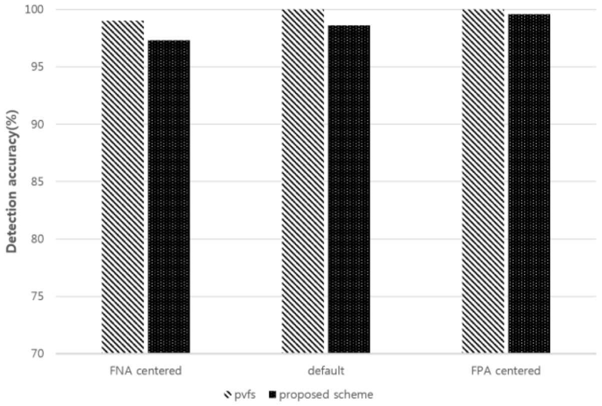

Fig.11 False report attack detection performance.

Fig.11 shows the filtering performance against a false report injection attack on the existing PVFS and that of the proposed method in a situation where the attack style changes randomly. The existing PVFS followed the experimental environment in [5], fixed at Tt = 5, and proceeded by changing Tf from 1 to 3. In the case of the proposed scheme, the threshold is set to three cases: the normal case, the false negative attack central defense case, and a false positive attack central defense case.

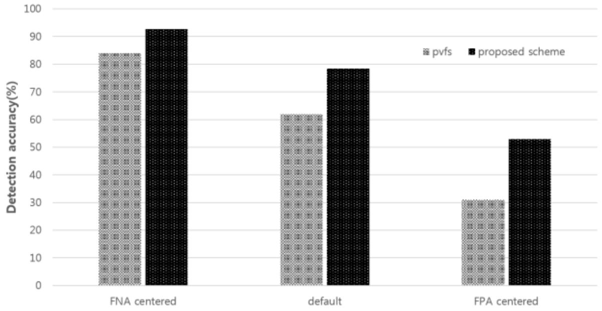

Fig.12 False vote injection attack detection performance.

Fig.12 shows the filtering performance against the false vote injection attack of the existing PVFS and that of the proposed method in a situation where the attack style changes randomly. False vote injection attack filtering performance of the existing PVFS differs greatly from that of the proposed method, and the weaknesses of PVFS are observed when Tf is set to 1. If the proposed scheme is set as a false attack centric report, the detection rate for false injection attacks is remarkably lower than that of false vote attacks. This allows us to identify tradeoffs associated with the values of the threshold settings.

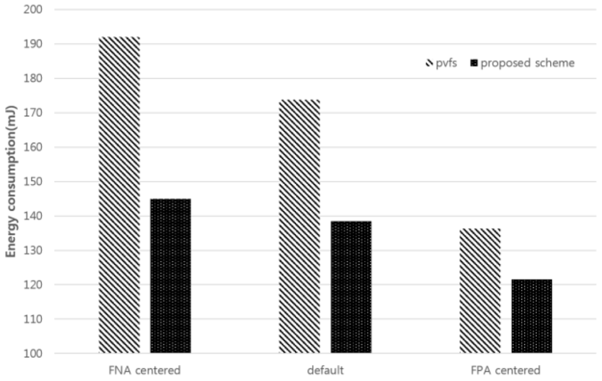

Fig.13 Amount of energy consumption.

Fig.13 shows the energy wasted in the proposed PVFS and the proposed method, where the cycles are repeated and accumulated. The existing PVFS was simulated and tested against the threshold set of (5, 2), and the proposed method was also tested for the normal case, i.e., neither false positive nor false negative centric. Energy waste is the cost incurred by filtering failure, which occurs when a false vote-injected report is dropped during filtering. Experiments on energy waste due to false report detection failures can be ignored in the proposed technique because the difference between the two techniques is negligible. Because of the superior detection performance, energy consumption is much lower than the amount of energy wasted in a false vote injection attack.

5. CONCLUSIONS

This paper proposes a method to identify the type of attack in a wireless sensor network using a fuzzy system. It also automatically resets the appropriate thresholds to those of PVFS. It maintains the detection rate for false positive attacks, improves the detection rate for false negative attacks, and saves energy. Future experiments will be conducted to further improve the accuracy of the first fuzzy system and to find the optimal threshold reset period.

6. ACKNOWLEDGMENTS

This research was supported by the MISP (Ministry of Science, ICT & Future Planning), Korea, under the National Program for Excellence in SW) (2015-0-00914) supervised by the IITP (Institute for Information & communications Technology Promotion)" (2015-0-00914)

7. REFERENCES

[1] Perrig, Adrian, John Stankovic, and David Wagner. "Security in wireless sensor networks." Communications of the ACM 47.6 (2004): 53-57

[2] Zhang, Wensheng, and Guohong Cao. "Group rekeying for filtering false data in sensor networks: A predistribution and local collaboration-based approach." INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE. Vol. 1. IEEE, 2005.

[3] Al-Karaki, Jamal N., and Ahmed E. Kamal. "Routing techniques in wireless sensor networks: a survey." IEEE wireless communications 11.6 (2004): 6-28.

[4] Perrig, Adrian, John Stankovic, and David Wagner. "Security in wireless sensor networks." Communications of the ACM 47.6.

[5] Li, Feng, and Jie Wu. "A probabilistic voting-based filtering scheme in wireless sensor networks." Proceedings of the 2006 international conference on Wireless communications and mobile computing.ACM, 2006.

[6] Jeba, S. A., and B. Paramasivan. "False data injection attack and its countermeasures in wireless sensor networks." European Journal of Scientific Research 82.2 (2012): 248-257.

[7] Jeba, S. A., and B. Paramasivan. "An evaluation of en-route filtering schemes on wireless sensor networks." International Journal of Computer Engineering & Technology (IJCET) 3 (2012): 62-73.

[8] Yu, Zhen, and Yong Guan. "A dynamic en-route scheme for filtering false data injection in wireless sensor networks." SenSys. Vol. 5. 2005.

[9] Zhu, Sencun, et al. "An interleaved hop-by-hop authentication scheme for filtering of injected false data in sensor networks." Security and privacy, 2004. Proceedings. 2004 IEEE symposium on. IEEE, 2004.

[10] Lu, Rongxing, et al. "BECAN: a bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks." IEEE transactions on parallel and distributed systems 23.1 (2012): 32-43.

[11] Yang, Hao, and Songwu Lu. "Commutative cipher based en-route filtering in wireless sensor networks." Vehicular Technology Conference, 2004. VTC2004-Fall. 2004 IEEE 60th. Vol. 2. IEEE, 2004.

[12] Ye, Fan, et al. "Statistical en-route filtering of injected false data in sensor networks." IEEE Journal on Selected Areas in Communications 23.4 (2005): 839-850.

[13] Zadeh, Lotfi A. "Fuzzy sets." Information and control 8.3 (1965): 338-353.

[14] J. Yen and R. Langari, Fuzzy Logic: Intelligence, Control, and Information. Prentice-Hall, Inc., 1998.

[15] G. Klir and B. Yuan, Fuzzy Sets and Fuzzy Logic. Prentice hall New Jersey, 1995.

[16] R. Babuška, "Fuzzy Systems, Modeling and Identification," Delft University of Technology, Department of Electrical Engineering Control Laboratory, Mekelweg, vol. 4, 1996.