Wireless Sensor Network, 2012, 4, 40-44
http://dx.doi.org/10.4236/wsn.2012.42006 Published Online February 2012 (http://www.SciRP.org/journal/wsn)

Fuzzy Based Assignment Method of Filtering Nodes in Wireless Sensor Networks

Soo Young Moon, Tae Ho Cho
School of Information and Communication Engineering, Sungkyunkwan University
Suwon, South Korea
Email: moonmous@ece.skku.ac.kr

Received November 6, 2011; revised December 12, 2011; accepted January 29, 2012

ABSTRACT

Wireless sensor networks (WSNs) are networked systems that are able to sense various events and report the events to a user to enable appropriate responses. One of security threats to a WSN is false data injection attacks in which an attacker steals some sensor nodes in the network and injects forged event messages into the network through the captured nodes. As a result, the intermediate nodes on the forwarding paths of the false event messages waste their limited energy. Additionally, the network cannot provide the user with correct information. There have been many studies on en-route detection of false event messages for WSNs. Yang *et al*. proposed the commutative cipher-based en-route filtering scheme (CCEF) which establishes a secure session between a sink node and a cluster head (CH) based on the commutative cipher. In CCEF, each intermediate node on the path between the sink node and the CH receives an event message and verifies the authenticity of the session based on a probability. Due to the probabilistic approach, it is hard to adapt to the change of false traffic ratio in the network and energy inefficiency may occur. We propose a filtering scheme which applies a deterministic approach to assign filtering nodes to a given session. The proposed method consumes less energy than that of CCEF without sacrificing security.

Keywords: Wireless Sensor Network; WSN; False Data; Filtering Scheme

1. Introduction

Wireless sensor networks (WSNs) are networked systems that are able to sense various events and report the events to a user to enable appropriate responses. Many tiny sensor nodes and a sink node organize a WSN. Each sensor node contains components of sensor, microcontroller, and transmission modules [1,2]. WSNs can be deployed in broad range of applications such as military, environment and transportation [3]. A sensor node is given highly constrained resources of energy, communication bandwidth, computation capability and memory. Furthermore, a WSN operates autonomously in hostile environment such as a battle field. Various security threats are derived from the characteristics of WSNs [4]. One of the security threats to a WSN is false data injection attacks in which an attacker steals some sensor nodes in the network and injects forged event messages into the network through the captured nodes. As a result, the intermediate nodes on the forwarding paths of the false event messages waste their limited energy. Additionally, the network cannot provide the user with correct information [5]. **Figure 1** visualizes false data injection attacks in a WSN.

Many studies [5-10] have been made to enable early detection of false event messages en-route to save the sensor nodes' energy. Among them, Yang, *et al.* proposed the commutative cipher-based filtering (CCEF) [5] which establishes a secure session between the sink node and a cluster head (CH) in an interesting region based on the commutative cipher. Each intermediate node on the path verifies the authenticity of the session for received event messages based on a probability. In CCEF, sensor nodes can filter out false event messages in early phases. However, its probabilistic approach is hard to adapt to the change of false traffic ratio, and energy inefficiency may occur.

We propose a filtering scheme based on deterministic approach to assign filtering nodes (*i.e.* the nodes which will verify the authenticity of the session for received event messages) to each session. We use fuzzy logic [11, 12] to choose the filtering nodes among the nodes on the path between the sink node and the CH. The proposed method derives the fitness value of a node as a filtering node from three inputs; the node's energy, the size of message authentication codes (MACs) and the false traffic ratio (FTR) in the network. The sink node assigns a node as a filtering node if the fitness value of the node exceeds a threshold value. The proposed method is able

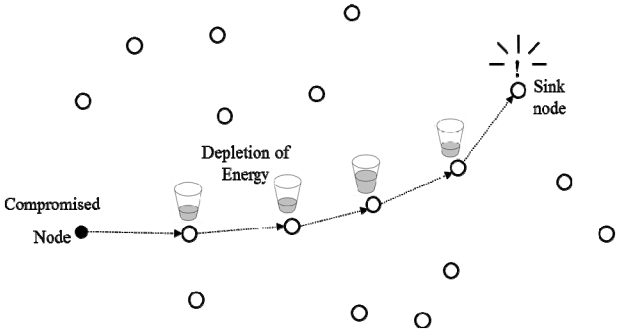


Figure 1. False Data Injection Attacks.

to detect false event messages with a few verification operations. Hence, the proposed method consumes less energy than that of CCEF.

2. Commutative Cipher-Based En-Route Filtering Scheme

CCEF [5] is a session-based filtering scheme in which event messages are sent through a secure session between the sink node and a CH. It is a non-symmetric key based filtering scheme that enables the intermediate nodes on the path to verify the authenticity of the session for received event messages without knowing the authentication key that was used to endorse the event messages. Commutative cipher is used both to authenticate and to verify an event message.

In CCEF, a secure session between the sink node and a CH must be established first. The sink node generates two keys for a session; a session key and a witness key. The sink node chooses a node in an interesting region as a CH node. Then the sink node sends a query message to the CH through the intermediate nodes. The query message contains the query ID, CH's ID, the session key encrypted by the unique key of the CH and the witness key in plain text.

After the session is established, the CH and remaining nodes in the interesting region collaboratively report events whenever the events occur. In each event message, there are two types of MACs; the session MAC and the node MAC. Each node on the path between the sink node and the CH receives event messages, verifies the session MAC based on the commutative cipher, and forward them only when the verification successes. The verification at each intermediate node is performed based on a probability for saving the node's energy [5]. **Figure 2** illustrates the operation of CCEF.

In **Figure 2(a)**, QID is the query ID, $\{K_S\}K_{CH}$ is the session key encrypted by the unique key of the CH, and K_W is the witness key in the query message. In **Figure 2(b)**, R is the event description and NMAC is the node MAC in the event message.

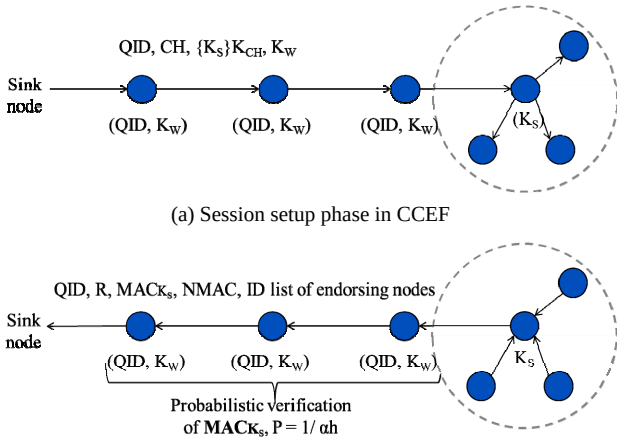


Figure 2. Operation of CCEF.

3. Proposed Method

3.1. Assumptions

The assumptions in the proposed method are as follows. The network is static, *i.e.* the sensor nodes do not move after initial deployment. Each node is assigned with its unique ID and preloaded with its unique key shared with the sink node. The sink node can estimate the energy level of every node and the FTR in the network.

3.2. Problem Definition

We focus on detecting and filtering false event messages injected by a non-CH compromised node in the proposed method. Those false event messages can be detected and dropped when they are verified at the first filtering node in the routing paths. We assume that a CH is not compromised for a given session and do not consider cooperation between multiple compromised nodes. The goal of our proposed method is to detect the false event messages with low computation overhead by assigning a few filtering nodes for a given session.

3.3. Motivation

In CCEF, each node verifies received event messages based on a probability. The probability of verification at each node does not change for various FTRs. Hence, energy inefficiency may occur due to the energy consumption for verifying correct event messages or the energy consumption for forwarding false event messages. The proposed method controls the number of verification operations for an event message by assigning filtering nodes for each session based on FTR of the network. The objective of the proposed method is to reduce the unnecessary energy consumption for verifying correct event messages and for forwarding false event messages and to increase the network lifetime.

3.4. Overview

The proposed method consists of four phases; the initialization phase, the session setup phase, the en-route filtering phase, and the sink verification phase. In the initialization phase, every node is preloaded with its own ID and own key shared with the sink node before node deployment. Sensor nodes are distributed randomly in their geographic location. After the deployment, all the nodes obtain their location information from some localization components and report the location information to the sink node.

In the session setup phase, sink node selects a node in the interesting region as a CH node, and prepares the session key and the witness key for the session. Then, the sink node examines every node on the path between the sink node and the CH to assign the session filtering nodes among the nodes. The sink node derives the fitness value of each node based on 1) the node's energy, 2) the size of message authentication codes (MACs) and 3) the false traffic ratio (FTR) in the network. Then the sink node assigns the filtering nodes to the session based on the fitness values. The sink node initializes the session by sending a query message to the CH through the intermediate nodes. The query message contains the query ID, the CH's ID, the session key encrypted by the CH's unique key and the witness key in plaintext. Each node receives the query message, stores the query id, and stores the witness key only if it is the filtering node for the session. The CH receives the query and decrypts the session key by using its key.

In the en-route filtering phase, the CH and the remaining nodes in the interesting region cooperate to derive an event description when an event occur. The CH computes the session MAC by encrypting the event description by the session key. Each of the remaining nodes computes its own MAC by using its own key and sends the MAC to the CH. The CH compresses the MACs received from the remaining nodes into the node MAC. The CH attaches the session MAC and the node MAC to the final event message. The CH sends the event message to the sink node through the intermediate nodes on the path.

Each node on the path receives the event message and first checks the query ID. If the query ID in its memory corresponds to that in the event message and it is the filtering node, the node verifies the session MAC in the event message by using the witness key based on the commutative cipher. If the verification fails, the node drops the event message. Otherwise, it forwards the event message to the next node on the path between the sink node and the CH.

In the sink verification phase, the sink node that receives the event message verifies the session MAC and the node MAC in the event message. If at least one MAC is incorrect, the event message is dropped.

Figure 3 shows the session setup phase and the en-route filtering phase of the proposed method.

In **Figure 3(a)**, {1,3} is the list of filtering node IDs. Each of Node 1 and Node 3 receives the query message and stores the (QID, K_W) pair for verifying event messages to be received. On the other hand, Node 2 just stores the query ID since it is not a filtering node for the session.

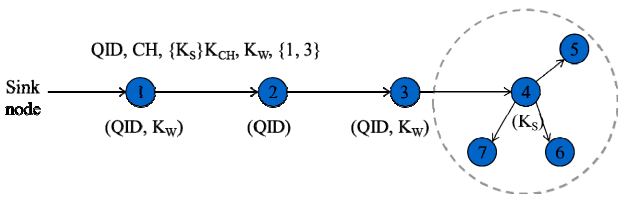
In **Figure 3(b)**, CH generates an event message and sends it to the sink node through the intermediate nodes on the path to the sink. A non-filtering node (*i.e.*, Node 2) just checks the QID in the event message and forwards it to the next node on the path. Filtering nodes (*i.e.*, Node 1 and Node 3) checks the QID and verifies the session MAC by the witness key.

3.5. Derivation of Fitness Values

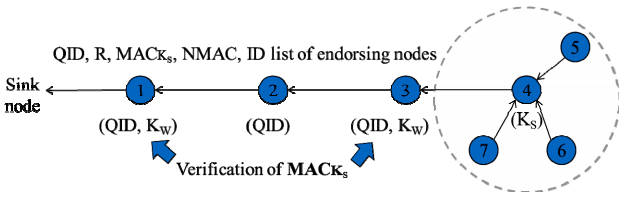
There are three input factors to derive the fitness value of each node on the path between the sink node and the CH as a filtering node for the session; 1) the node's energy, 2) the size of message authentication codes (MACs) and 3) the false traffic ratio (FTR) in the network. If the node's energy level is too low, the fitness value of the node is low. The size of MACs in each event message denotes the communication overhead for one message transmission. If its value is high, the energy consumption for forwarding false event messages is high. Hence, we set the fitness values of the nodes high. If the FTR is large, obviously we should set the fitness values of the nodes high to detect and drop false event messages early in their phase.

We adopt the fuzzy logic to derive the fitness value of sensor nodes. Hence, the input factors and the fitness value of each node are represented by the following fuzzy membership functions in **Figure 4**.

NODE_ENERGY corresponds to the energy level of each node and its value exists between 0 and 100 (%). It



(a) Session setup phase in proposed method



(b) En-route filtering phase in proposed method

Figure 3. Operation of Proposed Method.

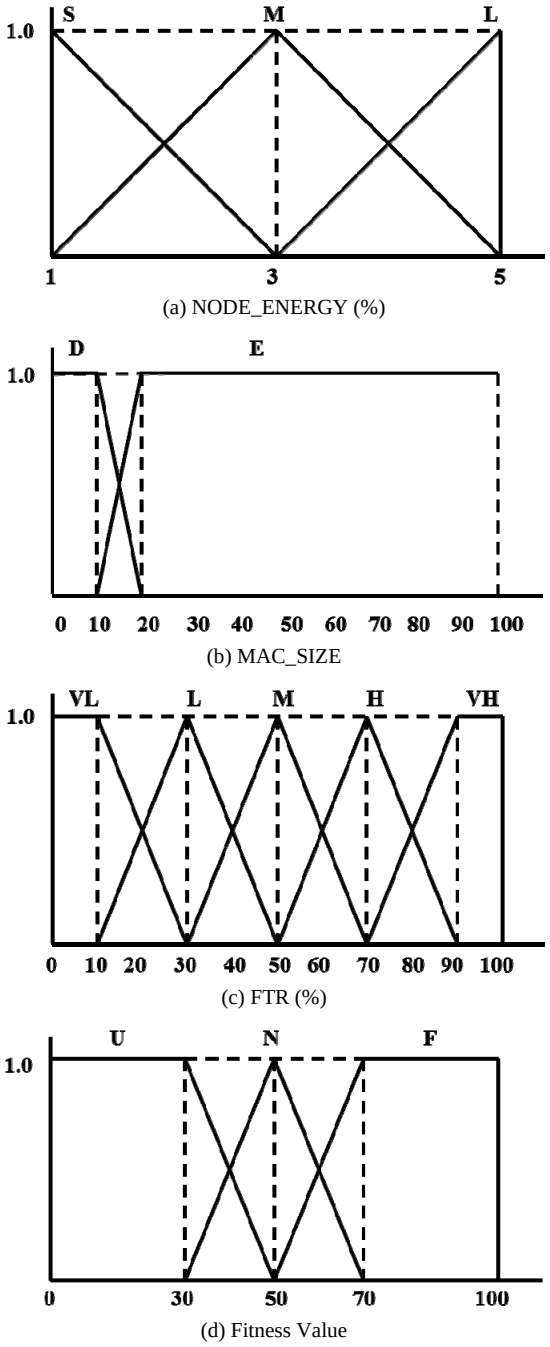


Figure 4. Fuzzy membership functions of inputs and output.

has only two fuzzy sets; Drained (D) and Enough (E). We assume that if a node's energy level is below 10%, its energy has been almost drained. MAC_SIZE is the size of each MAC in an event message. It is related to the communication overhead of nodes. MAC_SIZE has three fuzzy sets; Small (S), Medium (M), and Large (L). FTR is the fraction of forged event messages over the total number of event messages in the network. It is a dominant input factor to derive the fitness value of each node as a filtering node since it is directly related to the security. FTR has five fuzzy sets; Very Small (VS), Small (S), Medium (M), High (H), and Very High (VH). Fitness Value is the output of the fuzzy logic in the proposed method. If its value is larger than 50, the node becomes a filtering node for the session. Since there are two, three, and five fuzzy sets for the three input factors, respectively, obviously there are 30 combinations that we should consider. **Table 1** shows some of the fuzzy if then rules to derive the fitness value for the conditions.

4. Simulation Result

The simulation environment in our proposed method is as follows. 3750 sensor nodes are deployed in a 125 × 125 m^2 network. Each event message is 40 bytes long and requires 1.15 mJ to transmit and receive. The computation of commutative cipher consumes 9 mJ [5,13]. We simulated the energy consumption and detection ratios of CCEF and the proposed method for various FTR (0% ~ 100%) values. **Figure 5** represents the energy consumption of the CCEF and the proposed method.

As shown in the figure, the fuzzy logic-based proposed method achieves 5 to 40 % less energy consumption compared to CCEF with various parameter (*i.e.* α) values for the whole range of FTR. The reason is that the proposed method adapt to the change of FTR. That is, when FTR is low, the proposed method reduces the number of verification operations by selecting small number of filtering nodes, and vice versa. Hence, the proposed method is able to save the energy consumption for verifying correct event messages and the energy consumption for forwarding false event messages. **Figure 6** represents the false event message detection ratio of the CCEF and the proposed method.

CCEF achieves relatively constant detection ratios (60 to 100 %) of false event messages as FTR varies since the parameter values are static. On the other hand, the proposed method does not filter out false event messages until the FTR is less than 30 %. However, when FTR is larger than 40 %, the propose method achieves almost 100 % of detection ratios of false event messages. The reason is that the proposed method is able to adaptively increase the number of verification operations by selecting more filtering nodes for a session.

Table 1. Fuzzy If-Then Rules.

Rule#	NODE_ENERGY	MAC_SIZE	FTR	Fitness Value
0	VL	S	VL	U
14	VL	L	VH	U
17	E	S	M	F
21	E	M	L	N
29	E	L	VH	F

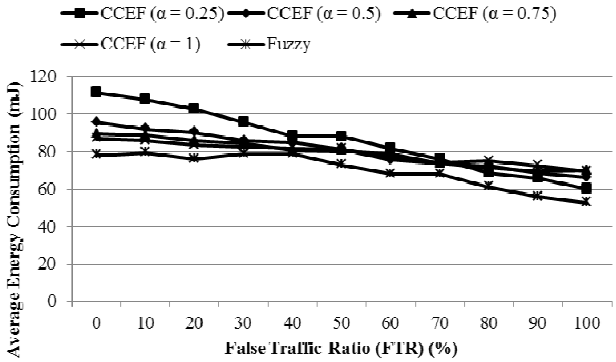


Figure 5. Comparison of Average Energy Consumption.

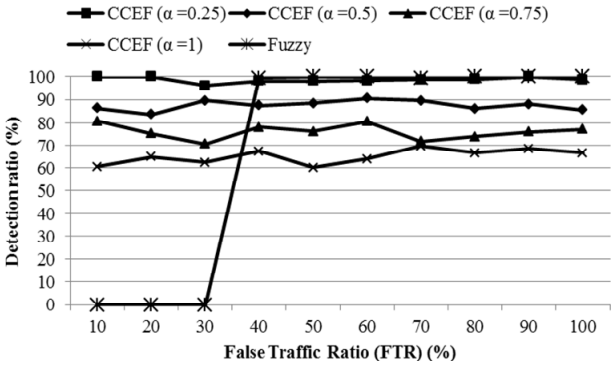


Figure 6. Comparison of detection ratios of false event messages.

5. Conclusions

In WSN, false data injection attacks are great threats because they deplete the limited energy of sensor nodes and compromise the accuracy of the collected data at the sink node.

Among the filtering schemes to detect false event message en-route, CCEF was proposed by Yang, *et al*. In CCEF, a CH sends event messages through a secure session with the sink node. CCEF allows the intermediate nodes on the path (between the sink node and the CH) to verify the authenticity of event messages without symmetric key-sharing and to detect false event messages early in their phase. However, sensor nodes verify the event messages based on a probability without consideration of FTR, leading to energy inefficiency. For these reasons, we proposed a filtering scheme which applies a deterministic approach to assign filtering nodes to a given session. The selection of filtering nodes is based on the fuzzy logic with three input factors; a node's energy, the size of MACs and the FTR in the network. We found that the proposed method consumes less energy than that of CCEF for various FTR values. We have also found that the proposed method achieves almost 100% of detection ratio for FTR values greater than 40%.

6. Acknowledgements

This work was supported by the Korean Research Foundation Grant funded by the Korean Government (KRF-2011-0004955).

REFERENCES

[1] I. F. Akyldiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "A Survey on Sensor Networks," *IEEE Wireless Communication Magazine*, Vol. 40, No. 8, 2002, pp. 102-116. doi:10.1109/MCOM.2002.1024422

[2] J. N. Al-Karaki and A. E. Kamal, "Routing Techniques in Wireless Sensor Networks: A Survey," *IEEE Wireless Communication Magazine*, Vol. 11, No. 6, 2004, pp. 6-28. doi:10.1109/MWC.2004.1368893

[3] N. Xu, "A Survey of Sensor Network Applications," University of Southern California, Technical Report, 2002.

[4] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," *Ad Hoc Networks Journal*, Vol. 1, No. 2-3, 2003, pp. 293-315.

[5] H. Yang and S. Lu., "Commutative Cipher Based En-Route Filtering in Wireless Sensor Networks," Vehicular Technology Conference, 26-29 September 2004, pp. 1223-1227.

[6] S. Zhu, S. Setia, S. Jajodia and P. Ning, "An Interleaved Hop-by-Hop Authentication Scheme for Filtering of Injected False Data in Sensor Networks," *Proceedings of Security and Privacy*, Oakland, 9-12 May 2004, pp. 259-271.

[7] F. Ye, H. Luo, S. Lu and L. Zhang, "Statistical En-Route Filtering of Injected False Data in Sensor Networks," *IEEE Journals on Selected Areas in Communications*, Vol. 23, No. 4, 2005, pp. 839-850. doi:10.1109/JSAC.2005.843561

[8] Z. Yu and Y. Guan, "A Dynamic En-Route Scheme for Filtering False Data Injection in Wireless Sensor Networks," 25*th IEEE International Conference on Computer Communications*, Barcelona, Spain, 23-29 April 2005, pp. 294-295.

[9] H. S. Seo, H. Y. Lee, S. J. Lee and D. G Lee, "Fuzzy-Based Filtering Solution Selection Method for Dynamic Sensor Networks," *Intelligent Automation and Soft Computing*, Vol. 16, No. 4, 2010, pp. 577-590.

[10] J. M. Kim, H. S. Seo and J. Kwak, "Routing Protocol for Heterogeneous Hierarchical Wireless Multimedia Sensor Networks," *Wireless Personal Communications*, Vol. 57, No. 3, 2011, pp. 577-590.

[11] John Yen and Reza Langari, "Fuzzy Logic," Upper Saddle River, 1999.

[12] Free Fuzzy Logic Library, 2011. http://ffll.sourceforge.net

[13] Moog Crossbow, 2011. http://www.xbow.com/

