Discrete Event Modeling and Simulation of Probabilistic Voting-based Filtering to Find Proper Security Parameters in Wireless Sensor Networks

Su Man Nam
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

In wireless sensor networks (WSNs), sensor nodes are vulnerable to false vote and false report injection attacks since they are widely deployed without infrastructure. Although some en-route filtering schemes can effectively detect the two attacks, these schemes need to set up various security factors before deploying the sensors in a sensor field. In this paper, we use a simulation model and find the proper security factors for a security scheme in a real-world simulation environment. We demonstrate that the scheme achieves better energy savings and detection power when the number of required message authentication codes (MACs) in a report is five and the number of detected MACs is two.

Keywords

WSN, Voting-based Filtering, Discrete Event Modeling

1. INTRODUCTION

Recently, wireless sensor networks (WSNs) have been used after technological advances in wireless communication [1-3]. A WSN organizes a large number of sensors and a base station (BS) in a sensor field [3]. The sensor nodes conduct event sensing, event reporting, and data verification, and the BS collects the event data from the nodes, analyzes the data, and provides information to users. However, the sensor nodes are easily stricken due to their vulnerability in an open-collaborative and large-scale environment without infrastructure. Adversaries use various attack patterns to inject false data into the network.

There are two types of attacks that can be made toward an application layer of a sensor network. False MAC injection attacks exploit fabricated message authentication codes (MACs) in a report, resulting in information loss, while false report injection attacks attach fabricated event data and MACs regarding non-existent events to a report, resulting in energy drain and false alarms in the sensor network. A probabilistic voting-based filtering scheme (PVFS) detects both of these attacks through the number of detected fabricated MACs (threshold) in a report. Although this scheme maintains effective energy savings and security strength, PVFS-based WSNs in the real world should approximately select a PVFS's security factors (threshold, MAC length) without precise measurements of the environment. Since discrete event system specification (DEVS) has hierarchical and modular structure characteristics, the sensor network is suitable for implementing the DEVS model because it has the same characteristics.

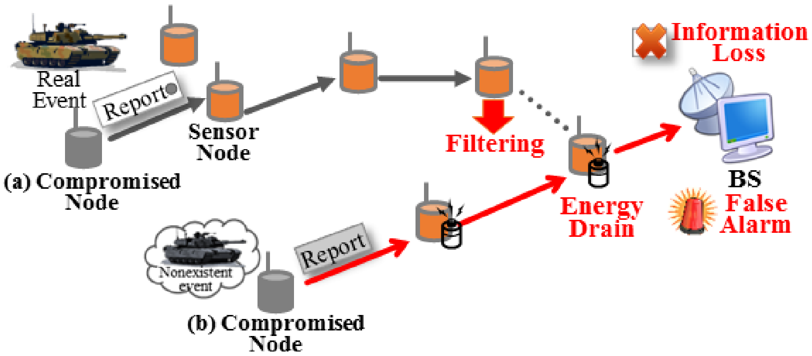

Fig 1: False MAC injection and false report injection attacks

In this paper, we use a simulation model, which is described in [4, 5], for a PVFS-based WSN and find appropriate security factors. The simulation model is implemented using DEVS. Since DEVS has hierarchical and modular structure characteristics, the WSN model is suitable for implementing the DEVS model because it has the same characteristics [4].

The rest of this paper is organized as follows: Section 2 introduces PVFS and DEVS. Section 3 shows a simulation model for a PVFS-based WSN. In Section 4, we present a performance evaluation of the simulation model. We draw conclusions at the end of this paper.

2. BACKGROUND

In this section, we briefly describe the operation of PVFS and the DEVS.

2.1 PVFS

In a PVFS [4, 6, 7], a cluster consists of a cluster head (CH) and multiple member sensor nodes (SNs) within a hop using a cluster-based model. The BS produces a global key pool (n partitions × m keys) and assigns the keys of a partition to each cluster. A source CH selects its verification nodes based on its distance and the verification nodes receive keys from the source.

A source CH broadcasts the event data to its SNs as an event occurs. The SNs check the data and generate a MAC, and transmit the MAC to its CH. The source CH collects all the MACs and generates a report including the collected MAC, and forwards it toward the BS. While the report is transmitted, the verification nodes verify the MACs in the report through their keys. If the number of detected MACs is below a threshold, the report is transmitted to the next node; if the number is over the threshold, the report is dropped. When the report arrives at the BS, it is verified again. In the PVFS, it is important to appropriately decide MAC length of a report and threshold for detecting an attack type.

2.2 DEVS

The DEVS formalism developed by Zeigler consists of hierarchical and modular discrete event models to analyze systems [8, 9]. DEVS's strength comes from its model reusability, expandability, and availability. The DEVS formalism defines two types of models: atomic models and coupled models.

$$\mathrm{M} = \langle X, S, Y, \delta_{int}, \delta_{ext}, \lambda, t_a \rangle,$$

where X is an external input set, S is a sequential state set, Y is an external output set, δ_{int} is an internal transition function, δ_{ext} is an external transition function, λ is an output function, and t_a is a time advance function.

Several atomic models can be coupled to build a more complex model, called a coupled model, and a coupled model can itself be used as a component in a larger coupled model. A coupled model is defined by the following structure:

$$\mathrm{N} = \langle X, Y, D\ \{M_i\}, C_{xx}, C_{yx}, C_{yy}, select \rangle,$$

where X is the set of input events, Y is the set of output events, D is a set of component names, M_i is a component of the basic model, C_{xx} is the set of external input couplings, C_{xy} is the set of internal couplings, C_{yy} is the set of external output couplings, and *select* is a tie-breaking function.

3. SIMULATION MODEL

We use a simulation model described in [4] and simulate the model for finding appropriate security factors to improve the performances of the sensor network. In [4], simulation results are individually evaluated according to the factors. In this paper, the simulation model removes controlled model within Clusters model in described in [4] so that the model is presented as the real world. In [5], the simulation model is modified. In this paper, we use the simulation model. In this section, we show the simulation in detail.

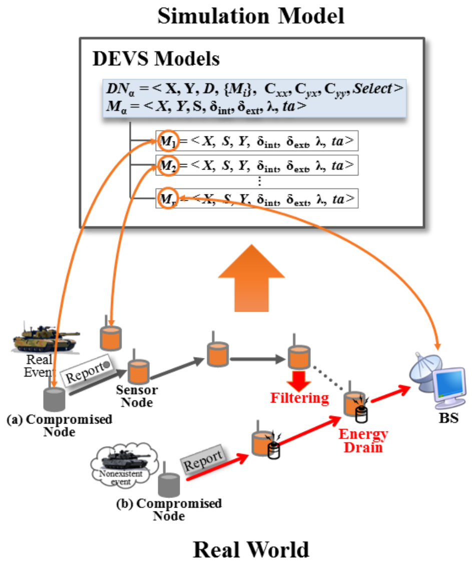
Simulation Model
DEVS Models
DN_α = < X, Y, D, {M_i}, C_{xx}, C_{yx}, C_{yy}, $Select$>
M_α = < X, Y, S, δ_{int}, δ_{ext}, λ, ta>
M_1 = < X, S, Y, δ_{int}, δ_{ext}, λ, ta>
M_2 = < X, S, Y, δ_{int}, δ_{ext}, λ, ta>
M_n = < X, S, Y, δ_{int}, δ_{ext}, λ, ta>
Real Event
Report
Sensor Node
(a) Compromised Node
Filtering
BS
Energy Drain
Nonexistent event
Report
(b) Compromised Node
Real World

Fig 2: Relationship between the real world and the DEVS model

Figure 2 shows the relationship between real world objects and the DEVS model. The sensor nodes (CH, SN) and one BS correspond to the atomic model.

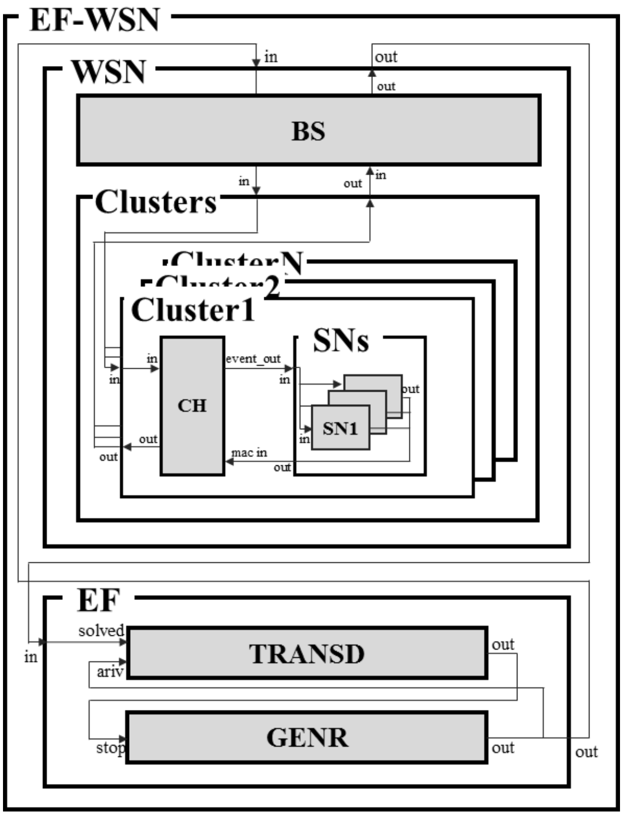

Fig 3: Simulation model

CH performs the following five behaviors:

- Event data broadcast
- MAC collection
- Report production
- Report verification

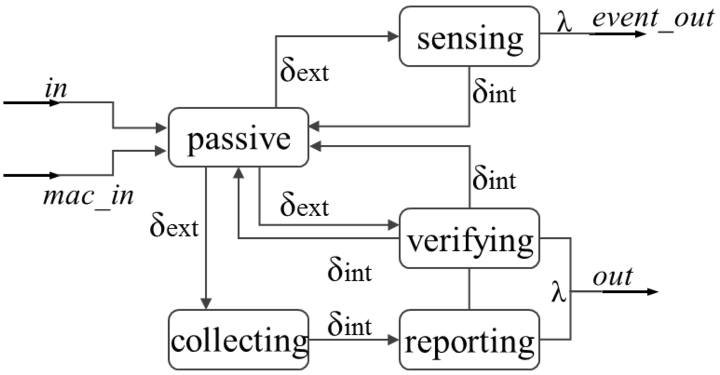

Fig 4: CH’s state transition diagram

Figure 4 shows the CH’s state transition diagram. The CH model distinguishes between the in-port and out-port, depending on the data type. The in-port of the model uses *in* and *mac_in* for reports and MACs, respectively, and the out-port uses *event_out* and *out* for event data and reports, respectively. The behaviors of the CH transfer the following state transitions:

- Event data broadcast: passive → sensing
- MAC collection and report production: passive → collecting → reporting
- Report verification: passive → verifying

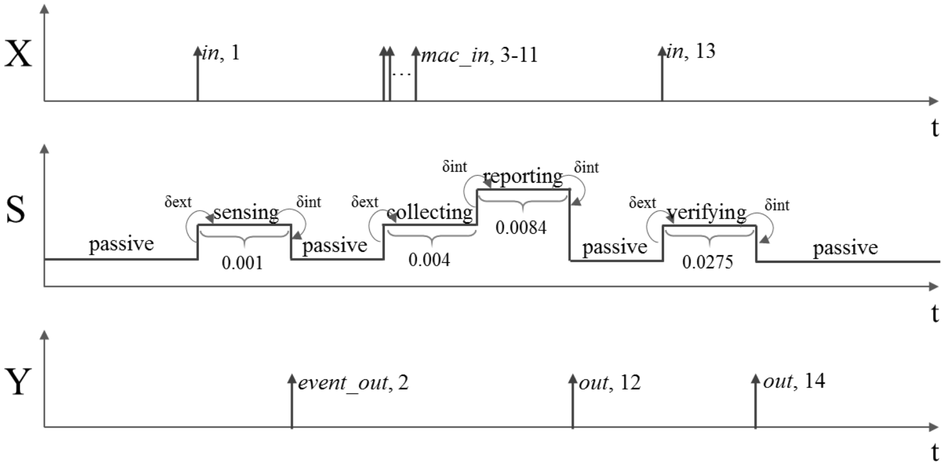

Fig 5: CH’s timing diagram

Figure 5 shows the CH’s timing diagram. In the timing diagram, X is the input, S is the state transition, and Y is the output. The CH model forwards an event (ID: 1) through the in-port *in*, transfers the passive phase to the phase sensing, and broadcasts the event data (ID: 2) through the out-port *event_out*. This model then collects MACs (ID: 3-11) during state collection through the port *mac_in*, transfers the passive phase to the phase reporting, and forwards a report (ID: 12) to the next CH model. In the next model, after receiving a report (ID: 13), the report is verified during phase verification.

The SN performs the following behavior:

- MAC production

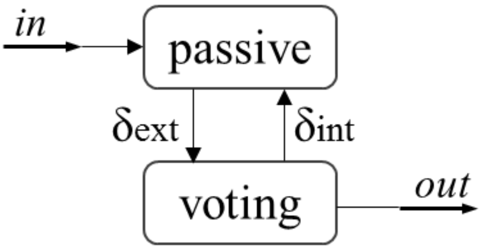

Fig 6: SN model

Figure 6 shows the SN’s state transition diagram. The SN model has the in-port *in* and the out-port *out*. The SN’s state is transferred as follows:

- MAC production: passive → voting

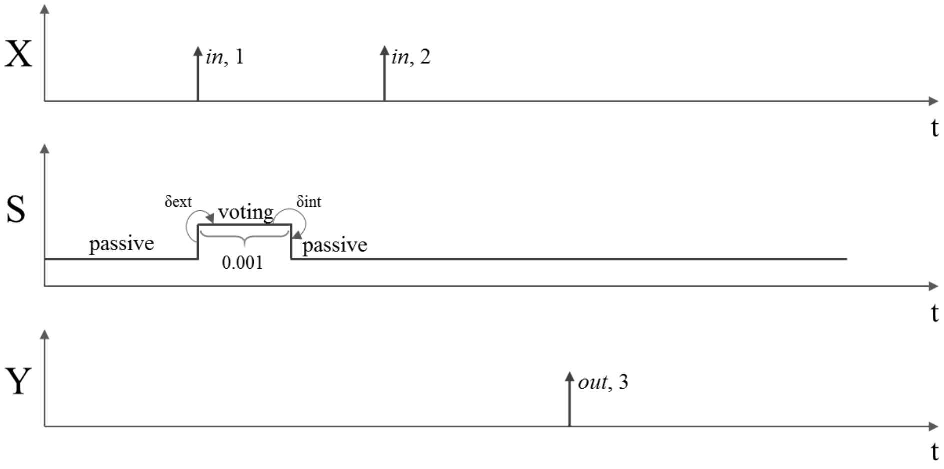

Fig 7: SN's state transition

Figure 7 shows the SN's timing diagram. The SN model receives event data (ID: 1) through the port *in* and produces MACs in the phase voting. The MAC is transmitted through the port *out* to its CH model.

The BS performs the following behaviors:

- Report verification

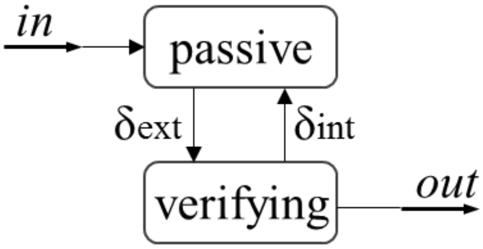

Fig 8: BS's state transition model

Figure 8 shows the BS's state transition diagram. The BS model's ports are *in* and *out*. This model's state transfers are as follows:

- Report verification: passive → verifying

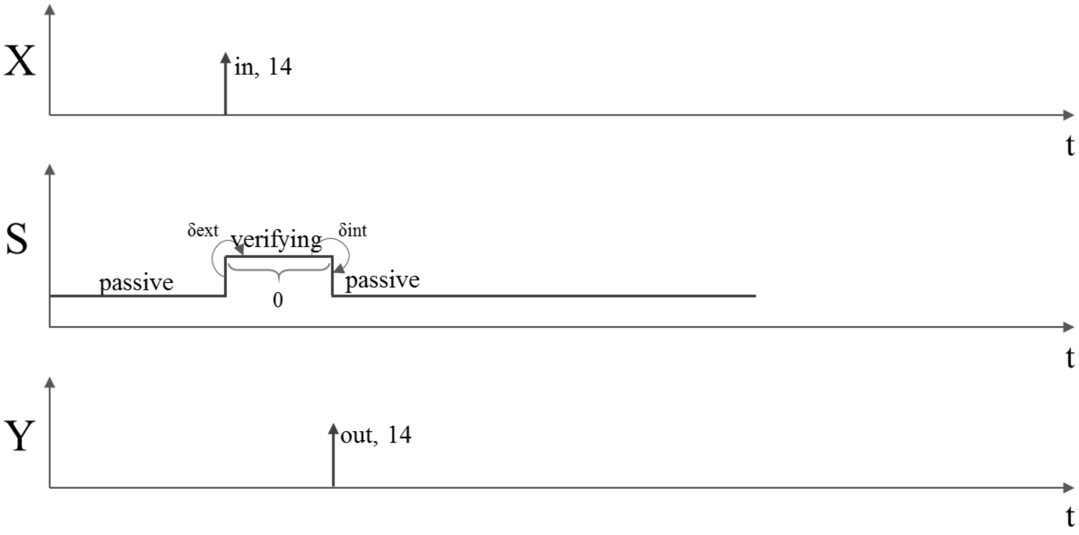

Fig 9: BS's timing diagram

Figure 9 shows the timing diagram of the BS model. The BS model receives a report (ID: 14) through the in-port *in* and verifies the report during phase verifying.

4. SIMULATION RESULTS

A simulation was executed to analyze the security protocol of the PVFS with the MAC length of a report and the threshold of a detected false MAC using DEVS. A sensor field was 1,000 x 1,000 m^2 and included 1,000 nodes (100 CHs and 900 SNs). The field organized 100 clusters in which a cluster consists of a CH and nine nodes. The initial energies of the CH and SN were 2 J and 1 J, respectively. Their transmission ranges were 150 m and 60 m, respectively. The size of each report is 36 bytes and the size of the MAC is 1 byte. Each node uses 16.25 µJ per byte to transmit, 12.5 µJ per byte to receive, and 15 µJ per byte to generate. To verify a MAC at the verification nodes, each node consumes 75 µJ [10]. The simulation sets 10 percent of the false traffic ratio (FTR), which probabilistically generates false data (e.g., fabricated MACs and false reports) among legitimate reports. In the simulation experiments, we randomly generated 500 events. There was no packet loss in the experiment.

In this paper, we evaluate the effectiveness of the threshold (2, 3) according to MAC length (s = 4, 5).

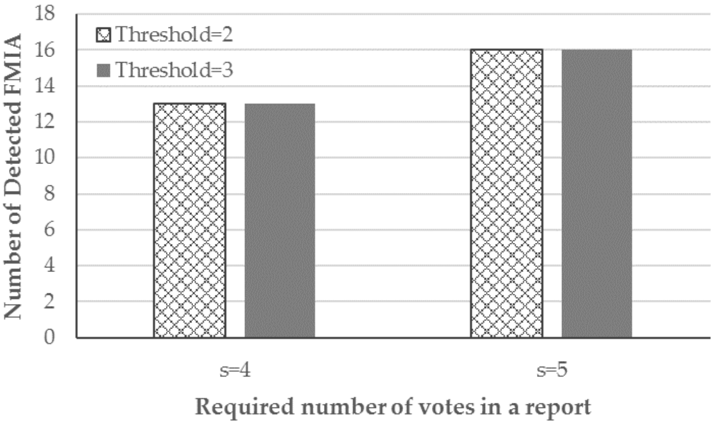

Fig 10: Number of detected FMIAs

Figure 10 shows the number of detected false MAC injection attacks (FMIAs) versus the required number of votes in a report. In the detection of the FMIA, the simulation model is not affected by changes in the MAC length and threshold.

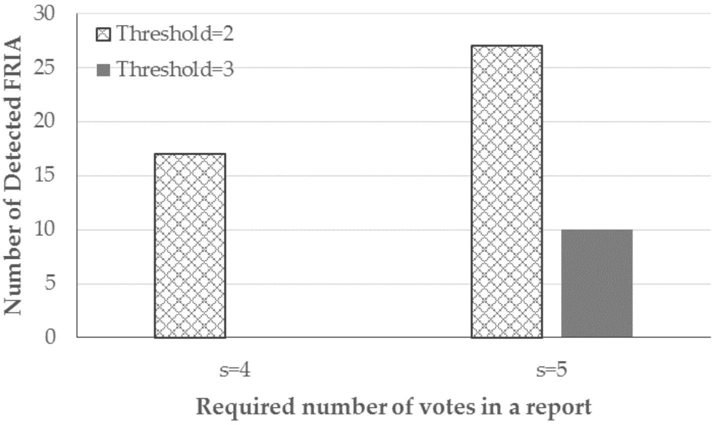

Fig 11: Number of detected FRIAs

Figure 11 presents the number of detected false report injection attacks (FRIAs) versus the number of detected FRIA. For s = 4 and threshold = 3, false reports are not detected due to the use of their unaffected factors. Therefore, when the MAC length of a report is five, the number of detected FRIAs increased for a threshold of 2 and 3.

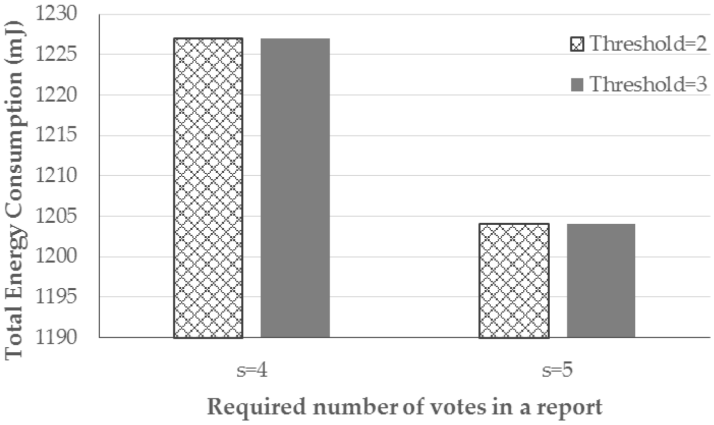

Fig 12: Total energy consumption

Figure 12 shows the total energy consumption versus the required number of votes in a report. As shown in Figure 11, s = 5 for energy consumption is less than s = 4 because of the high detection of false data.

5. CONCLUSION

In WSNs, various attacks are easily generated since they are widely deployed without infrastructure. It is important to evaluate the performance of the sensor network in virtual environments with these attacks through the simulation model. The simulation model of the PVFS analyzes the network performances, which are the security strength and energy efficiency, according to the MAC length and threshold of the detected MACs. Consequently, for s = 5 and threshold = 2, the simulation model demonstrates a better performance than with other factors.

6. ACKNOWLEDGEMENT

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).

7. REFERENCES

[1] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam and E. Cayirci, "Wireless sensor networks: a survey," *Computer Networks,* vol. 38, *(4),* pp. 393-422, 3/15, 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] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: a survey," *Wireless Communications, IEEE,* vol. 11, *(6),* pp. 6-28, 2004, 2004.

[4] S. M. Nam and T. H. Cho, "Modeling and Simulation of Threshold Analysis for PVFS in Wireless Sensor Networks," *International Journal of Research - GRANTHAALAYAH (IJRG),* vol. 4, *(8),* pp. 1-10, Aug. 2016.

[5] S. M. Nam and T. H. Cho, "Modeling and simulation of vote length analysis for probabilistic voting-based filtering in wireless sensor networks: Against false report and vote injection attacks," in *2017 International Conference on "International Conference on Research and Innovations in Science" (ICRISET 2017),* Anand, India, June 2017, pp. 196-204.

[6] F. Li, A. Srinivasan and J. Wu, "PVFS: A Probabilistic Voting-based Filtering Scheme in Wireless Sensor Networks," *International Journal of Security and Network,* vol. 3, *(3),* pp. 173-182, 2008.

[7] M. Akram and T. H. Cho, "Energy efficient fuzzy adaptive selection of verification nodes in wireless sensor networks," *Ad Hoc Networks,* vol. 47, pp. 16-25, 9/1, 2016.

[8] B. P. Zeigler and H. S. Sarjoughian, "Introduction to devs modeling and simulation with java: Developing component-based simulation models," *Technical Document, University of Arizona,* 2003.

[9] B. P. Zeigler, *Object-Oriented Simulation with Hierarchical, Modular Models: Intelligent Agents and Endomorphic Systems.* Academic press, 1990.

[10] 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, *(4),* pp. 839-850, 2005.