Adaptive Detection of Hello Flood Attack in Wireless Sensor Networks

H. Khosravi, R. Azmi, and M. Sharghi

***Abstract*—Hello Flood attack is a type of attack in wireless sensor networks. In this attack, the malicious node is able to disturb the security of network by sending periodic hello packets with high signal strength. In this research, an intrusion detection system based on neighborhood is proposed. It is based on a principle that those sensor nodes which are close to each other spatially are tended to have similar behavior. If a node shows a significant difference in its behavior in comparison with other nodes in neighborhood, it is considered as a malicious node. For optimizing, the adaptive method filtering based on Alpha-Beta is used. For our simulation, we have used TOSSIM-the sensor network simulator in TinyOS systems. The simulating result shows the detection method based on neighborhood has a high detection accuracy and low false positive rate. The proposed method also detects collusion attack of malicious nodes.**

***Index Terms*—Wireless sensor network, hello flood attack, adaptive intrusion detection, collusion attack.**

I. Introduction

Wireless sensor networks (WSNs) have become one of the most promising research and development areas over the past few years. WSNs have been used in many applications such as battlefield surveillance, traffic monitoring, healthcare, environment monitoring, and etc. Such networks usually consist of hundreds or even thousands of small-sized sensor nodes with limited computational capability and broadcast power. WSNs typically deployed in open, unprotected and unattended environments for long term operation to monitor and collect data. A WSN is vulnerable to several types of attacks; such as Hello Flood, Selective Forwarding, Sinkhole, and Worm Hole Attacks. Therefore security is an important factor in the design of WSNs [1]. In this paper, we focus on Hello Flood Attack.

Intrusion detection systems (IDSs) are proper mechanisms to defend against both insider and outsider attack which are widely used in wired networks. However, an IDS scheme designed for wired network cannot be directly applied to WSNs mainly due to limitations of processing power, memory, and energy. In WSNs, some attacks can be observed only by the neighbors of a malicious node. Hence, we assume that each sensor node runs an IDS agent and monitors its neighbors. The collected data, we believe, should be analyzed locally by a sensor node itself without any collaboration with other nodes since communication is highly energy consuming each bit transmitted in WSNs consumes about as much power as executing 800-1000 instructions. Consequently, we can minimize the resources used on detection modules because communication is more costly than computation in WSNs [2].

WSNs routing protocol can be classified into three categories based on network structure: flat-based routing, and hierarchical-based routing, and location-based [3]. In flat-based routing protocols, each node plays the same roles in routing procedure. In this paper, we use flat topology and SBA routing protocol which Peng and Lu was presented [4]. In this scheme no need to consider special node as a cluster head in each group. Grouping are formed based on distance between nodes, so the sensor nodes which are in the same group and close to each other cannot have different observation between their sensed data.

In this work, we consider the neighbor-based anomaly detection technique. The basic idea is that sensor nodes situated spatially close to each other should be dealing with similar behavior. If a node's behavior significantly differs from its neighbors, the node is considered malicious. We use alpha-beta filtering to adapting algorithm to changing network dynamic, so can detect colluding attacks of malicious nodes. Although these properties are welcome in WSNs, the technique was not researched yet.

The rest of this paper is organized as follows. In Section Π, we define Hello Flood attack and collusion attack. In Section III, related works of hello flood attack detection and prevention are presented. In Section IV, detection algorithm and routing protocol are described. In Section V, we present simulation parameters used in simulating the proposed approach. In Section VI, the results and analysis are described. Finally, we conclude the work in Section VII.

II. Hello Flood Attack

As shown in Fig. 1, some routing protocols in WSN require nodes to broadcast hello messages to announce themselves to their neighbors. A node which receives such a message may assume that it is within a radio range of the sender [5].

However in some cases this assumption may be false; sometimes a malicious node broadcasting routing or other information with more powerful transceiver than a general sensor node does. Nodes receiving such hello packets may falsely assume that they are within the radio range of the sender and try to forward their packets through this malicious node. These packets will be lost since they will not even reach the malicious node. Hence the network is left in a state of confusion. Protocols which depend on localized information exchange between neighboring nodes for topology maintenance or flow control are mainly affected by this type of attack [6].

Manuscript received July 24, 2015; revised March 2, 2016.

The authors are with Alzahra University, Iran (e-mail: hasti_khosravi67@yahoo.com, azmi@alzahra.ac.ir, msharghi@alzahra.ac.ir).

doi: 10.18178/ijfcc.2016.5.2.452

Received Signal Strength Indication (RSSI): In wireless sensor networks, received signal strength indicator compares the signal level with the threshold value which is defined previously.

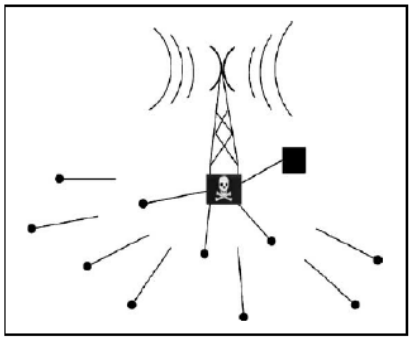

Fig. 1. Hello flood attack.

We also consider the scenario that a set of colluding nodes cheat many sensor nodes in a subgroup. This attack is difficult to detect by using the conventional methods. All of the malicious nodes send hello packet with high signal strength to sensor networks in a subgroup. So, sensor nodes in that subgroup are used to high signal strength and cannot detect colluding sensors.

III. Related Work

Researchers also suggested detecting malicious node using signal strength [6]-[8].

In [7] a solution proposed based on signal strength and geographical information for detecting Hello Flood attack. Every sensor node monitors its surrounding and whenever a transmission signal is detected by a sensor node, it would check if the signal strength of the transmitting node is compatible with the originator node's geographical position. Although, this solution was one of the first solutions in the domain, it is not efficient in many ways. The large overhead needed for transmitting data is a problem both for sending and processing. Also it is not energy efficient since all nodes are monitoring and processing data all the time. In addition, sometimes there are other reasons rather than attacks that may cause a change in the signal strength which make this solution impractical.

In [8], a collaborative intrusion detection architecture based on neighbors monitoring was proposed. The neighbor nodes communicate with each other in order to detect Selective Forwarding, Hello Flood and Jamming attacks. Their mechanism was implemented for Collaboration Tree Protocol (CTP) on the TinyOS environment.

Although, the collaboration among nodes makes this scheme strong, the communication overhead is a problem. Another drawback of this study is that it did not consider the power consumption rate related to the performance which is a very critical issue in WSNs.

In [6], proposed a countermeasure for the Hello flooding attack based on signal strength measurements and client puzzles. In this approach, the nodes are classified into "friend" and "stranger" groups, according to the signal power measurements. Requests with very abnormal powers are rejected. The strangers are then asked to solve puzzles. Aside from the puzzles which incur computational cost and are only useful if the number of requests is high, the received power level is not a good index to rely on for protection purposes. The main drawback of this approach is that it cannot detect the attack that is launched by set of colluding nodes.

IV. Proposed Approach

We assume that packet headers are not encrypted and every sensor node monitors its direct neighbors in an informal mode. Also, every sensor nodes and malicious nodes are set in the beginning of the network. Next, routing protocol and detection algorithm are explained.

A. Scalable Broadcast Algorithm

Two-hops neighbor knowledge is mostly used in broadcasting protocols to reduce the number of transmission, such as the Source-based Protocol, Dominant Pruning, Multipoint Relaying, Ad hoc Broadcast Protocol, and Lightweight and Efficient Network-wide Broadcast Protocol [9], [10]. As we mentioned in the related work's section, Peng and Lu proposed two-hop neighbor knowledge-based scalable Broadcast Algorithm (SBA) to reduce redundant forwarded packets. We also apply this algorithm as a part of our detection technique. We require each sensor equipped a detection module, which stores two-hop neighbor list. When the sensor nodes are first deployed in the sensing environment, each node exchange periodic 'hello' packets; each "hello" packets contains the node's identifier and the list of direct neighbors so that each node knows of all its 2-hop neighbors. Upon receiving a new broadcast packet from a neighboring node, node x should initiate a random back off timer and keep on receiving packets from other neighboring nodes. After the random back off timer expires, node x should determine if it has any two-hop neighbors that are not covered by the one-hop neighbors having sent the packet. If so, x has to rebroadcast the packet.

B. Detection Algorithm

In this paper we have proposed a solution for detection of hello flood attack which is based on Received Signal Strength Indication (RSSI).

The node b_{ij} is considered malicious by the node a_i if the Euclidean distance from RSSI(b_{ij}) to the center of the set {RSSI(b_{i1}), ..., RSSI(b_{ij})} is greater than the adaptive threshold δ_{RSSI} which is calculated in alpha-beta filtering in the next section. The "center" of the set is defined as the arithmetic average of RSSI of *ai*'s neighbors. Equation (1) shows related calculations.

$$\text{RSSI}(b_{ij}) - \text{AVG}(\text{RSSI}(b_{i1}), ..., \text{RSSI}(b_{imi})) > \delta_{RSSI} \quad (1)$$

C. Alpha-Beta Filtering

An Alpha-Beta filter is a steady state version of Kalman filter [11], under the condition of stationary noise processes and a fixed measurement rate. The simplicity and computational efficiency of the Alpha-Beta filter justifies its use in resource constrained WSNs. An Alpha-Beta filter has two internal states, where the first state is obtained by integrating the value of the second state over time. The two states can be called position x and velocity v. The Alpha-Beta Filtering is shown in algorithm 1. Assuming that velocity

remains approximately constant over the small time interval ΔT between measurements, the position state is projected forward to predict its value at the next sampling time refer to (2).

$$xk = xk1 + (vk1) \times dt \quad (2)$$

Since velocity variable v is presumed constant, so its projected value at the next sampling time equals the current value

Then, the prediction of error or noise r is calculated using. In the following the alpha value is multiplied by error probability (*rk*) and added to *xk*. *vk* is calculated by the following formula:

$$vk += (b \times rk)/dt \quad (3)$$

Next, the new value of *xk*-1 and *vk*-1 are set and the value of xk-1 is the output of algorithm.

Algorithm 1: Alpha-Beta Filtering

```
Require: dt, xk-1, vk-1, a, and b
Define xm, xk, vk
While(True)
1: xm = rand() % 100;// input signal
2: xk = xk-1 + (vk-1 * dt)
3: vk = vk-1
4: rk = xm – xk
5: xk += a * rk;
6: vk += ( b * rk ) / dt;
7: xk-1 = xk;
8: vk-1 = vk;
9: return xk-1
```

As we mentioned, Alpha-Beta filtering calculate Threshold of RSSI based on difference between received signal strength of two neighbor every 500_{ms}. Therefore, on the basis of main algorithm malicious or reality nodes identified. If a node is defined as malicious node, it is removed from neighbors list and added to the black list.

V. IDS Design and Implementation

A network of 100 sensor nodes was simulated in the TOSSIM – a simulator for TinyOS applications [12]. The size of network is 100 m × 100 m. Initially, the nodes are randomly placed in fixed position. Each node has at most 10 neighbors. Every node runs an IDS agent which monitors the information flowing in its neighborhood. IDS run periodic every 5 ms until every neighbor are set. The simulation parameters are given in Table I.

TABLE I: Simulation Parameters

Parameter	Value
Network size	100*100
Number of nodes	100
Number of malicious nodes	16
MAC	CSMA
Operating application	TinyOs
Channel bandwidth	1 Mbps

In collusion scenario, 10 colluding malicious nodes are set in a group with 10 sensor nodes. Maximum neighbor degree of a node is 10.

VI. Evaluation Metrics

In order to evaluate anomaly detection techniques, several metrics are defined as follow.

A. Receiver Operating Characteristic

The Receiver Operating Characteristic (ROC) curve [13] is used to evaluate the performance of the IDS. A ROC curve is a classical method for determining possible optimal models. The ROC analysis is based on the true positive rate (TPR) and false positive rate (FPR). The true positive rate (TPR) is the proportion of anomalous instances classified correctly over the total number of anomalous instances present in the test data. TPR is also known as sensitivity. The false positive rate (FPR) is the proportion of normal instances incorrectly classified as anomalous over the total number of normal instances contained in the test data.

There is a trade-off between the TPR and the FPR where adjusting a parameter, such as a threshold, to increase the TPR will result in an increase to the FPR. To examine this tradeoff, a receiver operating characteristic (ROC) curve is used. A ROC curve, Fig. 2, is generated by varying a parameter, such as the signal strength of malicious nodes. The resulting FPR and TPR form the ROC curve. Perfect performance is achieved when there is a TPR of 1 and an FPR of 0. The larger the area under the ROC curve, the better the performance of the anomaly detection algorithm.

In addition to examining the trade-off between the FPR and TPR, it is also necessary to compare the sensitivity of an anomaly detection technique to parameter selection. The area under ROC curve (AUC) [14] is used as a measurement of the performance of the scheme and is computed for a given ROC by calculating the area under the ROC curve. An AUC value of 1 indicates that the scheme has achieved 100% accuracy and an AUC value of less than 0.5 indicates that the performance is worse than the random assignment of the labels. By varying a parameter in the anomaly detection scheme, a plot of parameter versus AUC value provides a method to analyze sensitivity to parameter selection.

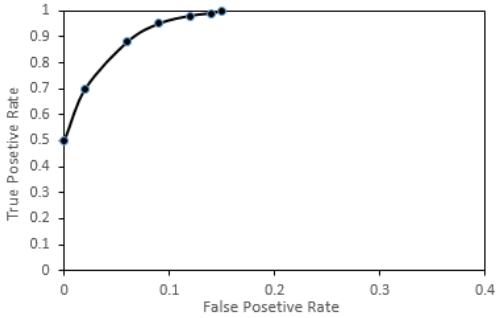

Fig. 2. ROC curve of hello flood attack detection.

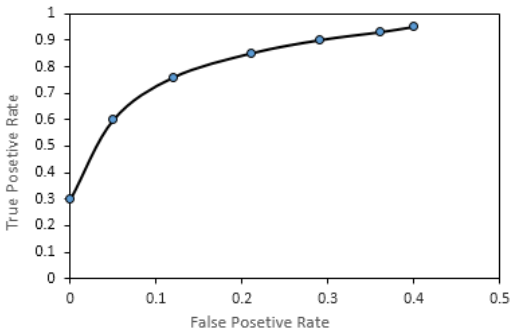

Fig. 3. ROC curve of collusion attack detection.

Fig. 2 illustrates ROC curve, corresponding to the accuracy of our approach for Hello Flood attack. The result of this measurement could be influenced by signal strength of malicious nodes and alpha-beta filtering. Under a particular monitored network, ROC curves have an optimal operating point for given IDS. It shows that the detection rate is 98 % with 0.12 false positive.

An important issue for the use of this anomaly detection approach in practice is how to set the signal strength and alpha-beta values. As can be seen from Fig. 2 a reasonable tradeoff between the detection rate and false positive rate is achieved at a false positive rate of approximately 0.1 on normal traffic.

We can characterize the performance of our approach using two measures based on the ROC curves in Fig. 2. The first measure is the detection rate for a false positive rate of 0.1. The second measure is the area under the ROC curve. The area under the ROC curve for a perfect IDS would be 100%, whereas the curve for an IDS that picks at random would be a diagonal line, where the area under this curve would be 50%. This area, for our scheme is almost 100%.

Fig. 3 shows the ROC curve of collusion attack in proposed structure. This result proves that our scheme can obtain good True Positive Rate with only 0.4 False Positive Rate.

B. Packet Delivery Ratio

We record number of packet successfully received at the destination node to analyze average delivery ratio.

Experimental result is illustrated in Fig. 4. It shows the comparison of the average number of received packets in two cases; with and without IDS under hello flood attack. In the case that we have no IDS, when the number of the malicious nodes increase, fewer sensor nodes receive the packet. Also, the number of lost packets obviously increases with the number of attacker. In the case that we have IDS in WSN, the averaged delivery packets mostly better. A small amount of packets are lost and the number of lost packet increases slightly with the number of attacker in our IDS method.

C. Average End-to-End Delay

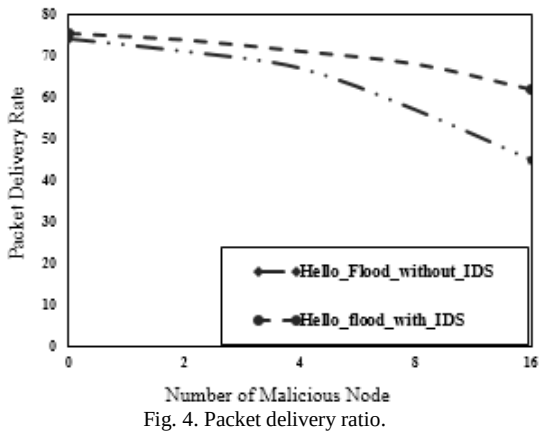

Fig. 4. Packet delivery ratio.

For multi-hop networks, latency is a key performance metric. We measure the time between the moment that the source node send out a packet and the moment the sensor node receive it. Average end-to-end delay is the average amount of time for all packets to reach destination. As we show in Fig. 5, our scheme can obtain good average delay under Hello flood attacks in comparison with when we have no IDS. When, the number of malicious nodes is 16 the average delay of our approach is further because when we have no IDS, a large number of the packets lost in order to Hello Flood attack, so those packets do not count in the calculation of end-to-end delay. As a result fewer packets lead to lower delay.

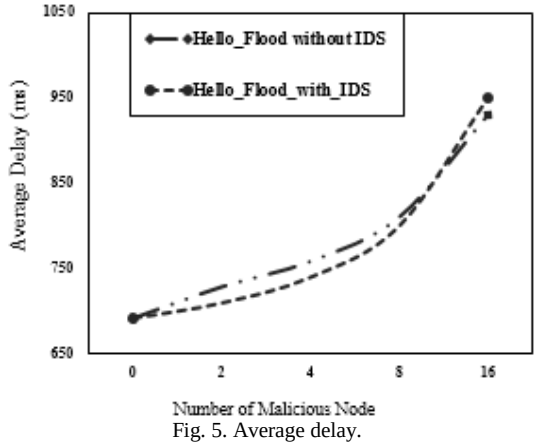

Fig. 5. Average delay.

VII. Conclusion

In this paper, we explored hello flood and collusion attacks in order to find out whether they can be detected by the adaptive detection technique. The alpha-Beta filtering was proposed for adapting algorithm with dynamic changes in network. We evaluated the implemented IDS in the TOSSIM simulator. We found out that our IDS is capable of detecting Hello Flood attack with reasonably low occurrence of false positives and high true positive. Also, the simulation results show that our IDS model have high packet delivery ratio and low delay. This solution can also detect collusion attack with an acceptable detection rate.

In the near future, we will explore our scheme to decrease false positive ratio. Also we evaluate the scheme to detect various attacks in WSN. Specially, evaluating it under selective forwarding attack would be the most priority.

References

[1] P. Wei and X. C. Lu, "On the reduction of broadcast redundancy in mobile ad hoc networks," presented at the 1st ACM International Symposium on Mobile Ad Hoc Networking & Computing, IEEE Press, 2000.

[2] T. H. Hai and E. N. Huh, "Detecting selective forwarding attacks in wireless sensor networks using two-hops neighbor knowledge," in *Proc. Seventh IEEE International Symposium on Network Computing and Applications*, pp. 325-331, IEEE, July, 2008.

[3] J. N. Al-Karaki and A. E. Kamal, "Routing techniques in wireless sensor networks: A survey," *Wireless Communications, IEEE*, vol. 11, no. 6, pp. 6-28, 2004.

[4] H. Eui-Nam and T. H. Hai, *Lightweight Intrusion Detection for Wireless Sensor Networks*, INTECH Open Access Publisher, 2011.

[5] A. Hamid, M. Mamun-Or-Rashid, and C. S. Hong, "Defense against lap-top class attacker in wireless sensor network, in Advanced Communication Technology," in *Proc. the 8th International Conference*, vol. 1, p. 5, IEEE, February 2006.

[6] V. P. Singh, S. Jain, and J. Singhai, "Hello flood attack and its countermeasures in wireless sensor networks," *International Journal of Computer Science*, vol. 7, no. 3, p. 23, 2010.

[7] W. R. Pires, T. H. Figueiredo, H. C. Wonga, and A. Loureiro, "Malicious node detection in wireless sensor networks," in *Proc. Parallel and Distributed Processing Symposium*, p. 24, April 2004.

[8] A. Stetsko, L. Folkman, and V. Matyáš, "Neighbor-based intrusion detection for wireless sensor networks," in *Proc. 6th International*

Conference on Wireless and Mobile Communications (ICWMC), pp. 420-425, IEEE. September 2010.
[9] A. Durresi, V. K. Paruchuri, S. S. Iyengar, and R. Kannan, "Optimized broadcast protocol for sensor networks," *Transactions on Computers, IEEE*, vol. 54, no. 8, pp. 1013-1024, 2005.
[10] B. Williams and T. Camp, "Comparison of broadcasting techniques for mobile ad hoc networks," in *Proc. the 3rd ACM International Symposium on Mobile Ad Hoc Networking & Computing*, pp. 194-205, ACM.
[11] R. Penoyer, "The alpha-beta filter," *C User's Journal*, vol. 11, no. 7, pp. 73-86, 1993.
[12] P. Levis, N. Lee, M. Welsh, and D. Culler, "TOSSIM: Accurate and scalable simulation of entire TinyOS applications," in *Proc. the 1st ACM Conference on Embedded Networked Sensor Systems*, pp. 126-137, New York, NY, USA, ACM, 2003.
[13] C. G. Li, "A framework for signal strength based intrusion detection system for link layer attacks in wireless network," Doctoral dissertation, Carleton University, Ottawa, 2007.
[14] Y. Jie, Q. Yang, and J. J. Pan, "Sensor-based abnormal human-activity detection," *IEEE Transactions on Knowledge and Data Engineering*, pp. 1082-1090, 2008.

Hasti Khosravi is a master student of information technology engineering at Alzahra University, Iran. She got his bachelor of engineering in information technology in 2013 from Islamic Azad University, Sanadaj Branch. Her thesis is about designing a hybrid intrusion detection system in a wireless sensor network.

Reza Azmi received his BS degree in electrical engineering from Amirkabir University of Technology, Tehran, Iran in 1990 and his MS and PhD degrees in electrical engineering from Tarbiat Modares University, Tehran, Iran in 1993 and 1999 respectively. Since 2001, he has joined Alzahra University, Tehran, Iran. He was an expert member of image processing and multi-media working groups in ITRC (From 2003 to 2004), optical character recognition working group in supreme council of information and communication technology (From 2006 to 2007) and security information technology and systems working groups in ITRC (From 2006 to 2008). He was project manager and technical member of many industrial projects. Dr Azmi is founder of Operating System Security Lab (OSSL), Medical Image Processing Lab (MIPL), Face and Facial Expression Recognition Lab (FFERL), Web-based Anomaly Detection Lab (WADL) and Optical Character Recognition Lab (OCRL) in Alzahra University. He is currently an assistant professor of Computer Engineering at Alzahra University.

Mehran Sharghi received his PhD degree in computer science from University of Dundee, Dundee, UK in 2003. His main research was focused on medical imaging. He is currently a lecturer within Computing Department of Alzahra University, Tehran, Iran. His current research interest includes image processing and computer networks.