DEVELOPMENT OF A SIMULATION MODEL OF PROTECTED WIRELESS SENSOR NETWORK

E. Basan, O. Makarevich and N. Tsopcalo
Institute of Computer Technologies and Information Security of the Engineering and Technology, Academy of the Southern Federal University, Russia
E-Mail: ele-barannik@mail.ru

ABSTRACT

This research is devoted to protection of wireless sensor networks (WSN) that is capable of counteracting a wide spectrum of active intrusions. The importance of this problem is caused by the recent increase of popularity of wireless sensor network in the industry and in the military and the lack of protection means that take particular features of WSN into account and can effectively repel active attacks. The aim of this research is to develop a WSN simulation, which would allow to evaluate state of nodes using several trust evaluation approaches and to compare the result with the proposed method. Trust level computation uses three main parameters for node state evaluation, which allows expanding the scope of detected attacks.

Keywords: wireless sensor network, trust, probability, anomalous behavior, condition graph, attack, intruder, beta-distribution, normal distribution, confidence interval.

1. INTRODUCTION

Nowadays, the problems of increasing protection and ensuring the security of information systems (IS) are really important. At the same time, an information system can combine various technologies, such as cloud computing, terminal access, and virtualization. Recently, wireless sensor networks have become a part of information systems. Examples of such systems include “smart home”, “smart city”, etc. Thus, while ensuring the protection of an information system that includes a wireless sensor network, care must be taken not only for the security of standard automated network devices, but also for the security of sensor nodes. At the same time, because WSN have features that are not typical for wired and wireless computer networks, such as physically unprotected nodes, lack of infrastructure, dynamically changing topology, limited resources of the node-sensor, changes in the quantitative and qualitative network configuration, methods and means of protection must take these factors into account.

As an intruder model in this study, we consider an internal intruder, who actively destabilizes the work and causes damage to both the network itself and the facility that is under its control. This study examines trust-based security systems that not only support trusted relationships between sites, but also detect abnormal activity caused by an attacker.

In this research we use a probabilistic metric for trust evaluation. The majority of approaches [7], [8], [9], [10], that use beta distribution metric for representing successful si and unsuccessful fi network events. This approach allows estimating the probability that the node is untrusted with high number of discarded packets; this attribute is used as unsuccessful network events. Delivered or redirected packets are counted as successful events. RFSN system was the one the first systems that used Bayes theorem and beta-distribution for trust evaluation [10]. Several modifications were made to that system. For example, BTMS system calculates the trust value from both direct and indirect information [9]. Principal drawbacks of the existing systems can be summarized as follows:

- usually, the evaluation of successful / unsuccessful network host events is used as attributes for evaluating trust, which allows to detect an attacker in case of a small attacks series;
- sensor nodes make all computations themselves while constantly exchanging trust values, which decreases the total bandwidth and can decrease;
- the time needed for the attack detection should be decreased in order to raise the detection efficiency.

2. DESIGN OF WSN SIMULATION WITH EMBEDDED TRUST EVALUATION

The designed WSN is based on three conditions that are relevant to both individual nodes and the entire network. Condition S1 means that the node (or the network) is in trusted mode, i.e. its trust value exceeds 0.5. Condition S2 implies that the network node trust is around 0.5 (within 0.45-0.55). Condition S3 Implies that the node is untrusted, i.e. its trust value is much lower than 0.5. The following limitations of state changes apply: a system can change state S1 to state S2 and state S3; a system can change from S2 to S1 and S3; a system cannot change state S3. Reaching the latter state is equivalent of ceasing to exist: all interactions with the node are stopped.

Change of state occurs whenever a node Pi reaches a boundary condition caused by a change of one attribute Ai or a combination of attributes.

The following three parameters can influence the state: P1= fi is the number of dropped packets; P2= L is the node’s load; P3=e is the remaining charge level. Particular factors influence these parameters; we denote them as attributes Aij. At the same time, it is not always the actions of the attacker that contribute to the occurrence of these changes, but also current events that occur while the network is running. That is why we define the threshold value of 0.5 that determines particular node states.

In this research we are offering to consider the influence of parameter P1 on network nodes' conditions.

Consider the attributes that can influence the rate of dropped packets P1: A11 = malicious implies that an intruder deliberately dropped the packet; A12 = rate is the data transfer rate; A13 = interval is the data transmission interval; A14 = packet Size is the size of a transferred packet.

In order to determine the probabilities of a node's transition to a particular state, it is necessary to determine the type of distribution law for parameters P1. We carried out a simulation of WSN with NS – 2.35 simulator [12] obtained data sets that are related to the defined parameters.

The network consists of 25 sensor nodes that exchange data among themselves and send data to the base station N0. The packets are sent according to a predefined scenario that corresponds to a chosen data transmission model by timeout. In this model we use one of the synchronization mechanisms offered by IEEE 802.15.4 namely beacon-enabled network operation mode. Based on the simulation results, the system outputs a trace file. This file allows you to analyze the data collected over the simulation for transmitted packets, moving nodes and residual energy. The simulated WSN's topology is shown in Figure-1.

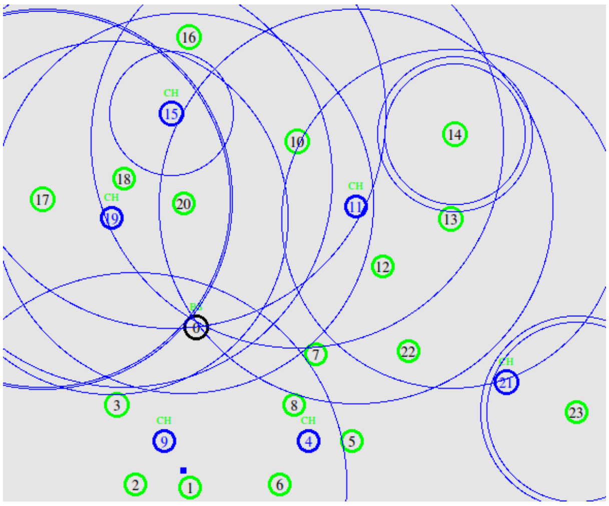

Figure-1. Simulated WSN's topology.

2.1 development of the network node's state calculation method

As described earlier, a network node can receive three states, which are affected by changing the parameters of the network node. After simulating the WSN, and having received data sets for each parameter, we can determine a distribution type for each obtained data set. With regard to parameter P1 (dropped packets), the choice of appropriate distribution is based on [8], [9], and [10]. The authors of these publications choose beta-distribution as long as parameters α and β correspond to successful and unsuccessful node events.

The method for determining the state of a node is based on calculating the probability of whether the node is trusted. At the same time, if we talk about trust in the case of using the beta distribution, then the probability that the node is untrusted is based on the correlation of successful and unsuccessful network events. It is important that, when the successful events of a network node grow, the unsuccessfully transmitted packets are not lost. For this, in some approaches [9], a "penalty factor" is used, which increases the significance of unsuccessful events. In addition, it is proposed to use additional weighting factors in this approach. Direct trust value Tdir is calculate by a node for its neighbor by observing it. The following two characteristics are considered for trust evaluation: successful operations Gi and unsuccessful operations Fi of node i. Suppose node A is measuring trust level to node B, then node A is logging all the events D splitting them into successful events Gi for each of three packet groups and unsuccessful events Fi. The set of all packets is divided into a subset of data packets, management packs and routing packets. In this case, successful / unsuccessful node events can be represented as:

Gi = sarp + scbr + saodv + rarp + rcbr + raodv,
Fi = daodv + darp + dcbr.

Weights are calculated using the information about the probability that a particular group of packets is involved in active attacks [11]. Weighting factors allow us to increase the weight / importance of the particular category of packets to which the attack is most likely to be directed. If the weight coefficients are correctly calculated, the results of the calculations can increase attack detection level. Then, each node i calculates trust values for each type of packets:

$$T_{i,AODV} = w_1 \frac{G_{A,B}(\Delta t)_{AODV} + 1}{G_{A,B}(\Delta t)_{AODV} + F_{A,B}(\Delta t)_{AODV} + 2} * (1 - \frac{F_{AODV}}{D_{AODV}}),$$

$$T_{i,CBR} = w_2 \frac{G_{A,B}(\Delta t)_{CBR} + 1}{G_{A,B}(\Delta t)_{CBR} + F_{A,B}(\Delta t)_{CBR} + 2} * (1 - \frac{F_{CBR}}{D_{CBR}}),$$

$$T_{i,ARP} = w_3 \frac{G_{A,B}(\Delta t)_{ARP} + 1}{G_{A,B}(\Delta t)_{ARP} + F_{A,B}(\Delta t)_{ARP} + 2} * (1 - \frac{F_{ARP}}{D_{ARP}}).$$

The calculation of the total direct value of trust, adding weight factors and the penalty factor, is made by the formulas:

$$T_{i,dir} = \sum (T_{i,AODV}, T_{i,CBR}, T_{i,ARP}) .$$

3. INTRODUCING TRUST COMPUTATION MECHANISMS TO THE WSN SIMULATION

Simulator ns-2.35 implements the following packet transmission scheme. From LL level, the packets are sent to PriQueue queue. This class is derived from Drop Tail, i.e. packet accounting is implemented for this type of the queue at the moment. For queue REDQueue, trust calculation is not yet supported. Routing packets (AODV) are prioritized and are placed in the head of the queue, i.e. they can hardly be dropped. ARP-packets are placed into the common queue. Three types of packets are distinguished during calculation: AODV, ARP and all the others (usually, these packets contain data from our scripts, their type is CBR). For each type, the number of incoming queues, the number of those who left the queue for the MAC level and the number of discarded ones are counted. Those that left the queue contribute to successful events G_i, while the number of dropped packets contribute to unsuccessful events F_i. WSN component scheme is presented in Figure-2. One can see where Trust estimation model is placed in the common structure. The trust calculation module takes the data from the MAC layer and then performs calculations and outputs the data to a trace file.

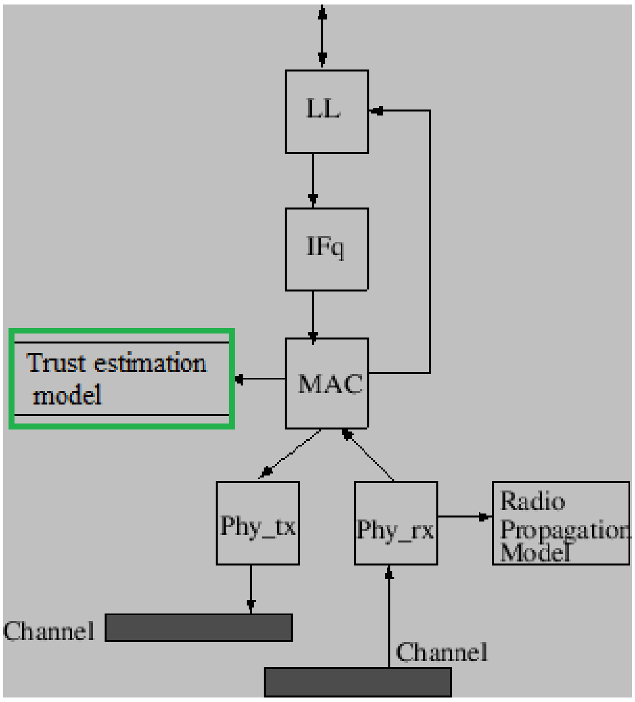

Figure-2. WSN structure with trust evaluation model.

The trust level message is added to the trace file in the following form:

T -t 0.100116 -n 15 -v1 1.000000 -v2 1.000000 -v3 1.000000 -l 3 -m 3 -d 0 -lh 2 -mh 2

The values of the fields are as follows: T is a string feature with trust value output; t is the timestamp of event; n is the number of the node; v1 is the trust value calculated with RFSN scheme [9]; v2 is the trust value calculated with the penalty according to BTMS system [10]; v3 is the trust value evaluated according to the proposed approach Ti,dir; l is the quantity of packets that were sent from LLC level to the output queue; m is the number of packets that were sent to MAC level from the queue; d is the number of dropped packets; lh is the total number of AODV and ARP packets that were sent down from LLC level; mh is the total number of AODV and ARP packets, that reached MAC level.

So it becomes possible to evaluate the state of any node in the network. In addition, this model allows you to compare the accuracy of an untrusted node evaluation and determine the best method for calculating the trust level.

4. CONCLUSIONS

Direct trust evaluation is based on matching successful and unsuccessful network events with the beta-function. Successful events take the role of $\alpha = Gi$ variable, while the value of $\beta = Fi$ is reserved for unsuccessful events. In order to increase the significance of unsuccessful events, the penalty factor is used and weights are applied. To find out the effectiveness of using the penalty factor and weighting factors, we perform calculations for various situations where the ratio of the number of successful and unsuccessful network events changes. Such calculations were carried out for the developed method and analogs. The results of the calculations are presented in Table-1. From Table-1 we can conclude that the system RFSN = *v1* does not react to the increase of unsuccessful events if the number of successful events grows as well, however the node is not always considered as trusted. The system BTMS = *v2* detects an intruder, when there are twice as many unsuccessful events than successful events. The designed method = *v3* detects an intruder in all the cases even if the difference between Gi and Fi is less than a half. From Table-1 it is clear that when an intruder is identified using the developed method, 40% of dropped packets are sufficient for successful intruder identification, which is not true for the other two approaches.

Table-1. Direct trust value evaluation under Black-hole attack.

Gi	Fi	v1	v2	v3
4	2	0,616	0,625	0,46
34	20	0,54	0,625	0,39
79	47	0,42	0,625	0,34
89	53	0,40	0,625	0,32
149	61	0,5	0,7	0,48

In the case of analogs of the developed method, the ratio of successful / unsuccessful events should exceed 50%. Thus, the method being developed can reduce the detection time of a node that is in the untrusted state. Suppose there are two intruder nodes *N4* and *N11* that drop packets, with *N4* starts dropping packets from the 1st second and *N11* starts dropping from the 80th second. Below is a fragment of trace file, from which one can conclude that the developed method *v3* demanded 15 second to start consider that node as undefined and 33 seconds to mark it as untrusted. Other methods v1 and *v2* could not detect any anomalous activity even in 98 seconds. In Figure-3 one can see the comparison of intrusion detection level for the proposed approach and its analogs.

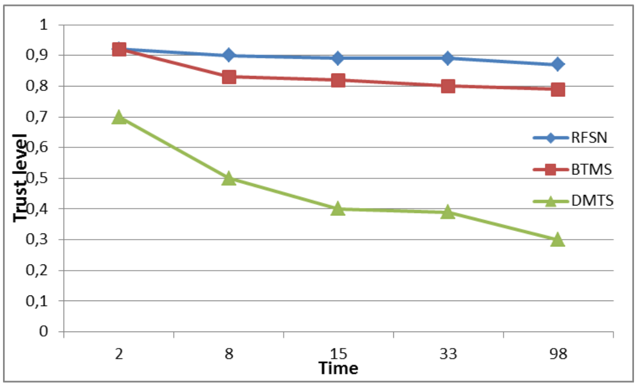

Figure-3. Trust level change for the intruder node.

Given that an attacker can distract the network in a relatively short period of time or can conduct time-based attacks, the fact of its detection, even with little impact, is very important.

Wormhole attack. This attack is implemented as shown in Figure-4.

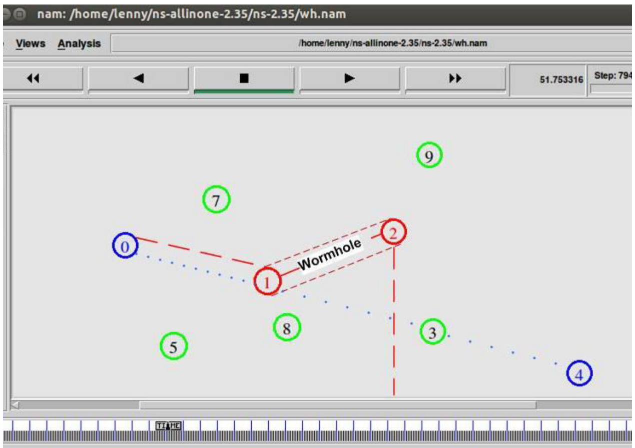
nam: /home/lenny/ns-allinone-2.35/ns-2.35/wh.nam
Views
Analysis
/home/lenny/ns-allinone-2.35/ns-2.35/wh.nam
51.753316
Step: 794
0
1
2
3
4
5
7
8
9
Wormhole

Figure-4. Worm Hole attack implementation scheme in NS-2.35 simulator.

Node 0 sends packets to node 4 via UDP; node 1 receives the packet, because it is the most profitable node for transmission, and redirects the packet to node 2 with which it has a tunnel (the red path), and finally node 2 drops it. If the tunnel was not established, the packet should have been sent to node 3 (blue trajectory). To detect this attack, the developed method took 6 seconds. Gray-hole attack. This attack was implemented in such a way that the attacker does not constantly drop packets, but only at certain points in time. In order to return the attacking attacker to normal behavior, a trusted type of behavior was added when the node does not discard packets:
$ns at 10.0 "[$node (7) set ragent_] trusted".

During this attack, the node 7 is positioned between nodes N0 and N9 and starts dropping the packets according to that scenario. The result of trust evaluation for the malicious node is shown in Table-2.

Table-2. Direct trust evaluation under Gray-hole attack.

Time	Trust level		
	RFSN	BTMS	DMTS
1	1	1	1
5	0,95	0,95	0,91
10	0,8	0,7	0,5
15	0,6	0,4	0,3
20	0,6	0,48	0,4
25	0,7	0,5	0,4
30	0,6	0,39	0,35
35	0,6	0,37	0,32
40	0,6	0,42	0,4
45	0,5	0,39	0,31
50	0,6	0,37	0,34
55	0,5	0,29	0,2
60	0,4	0,19	0,1

As we can see from the table, the developed method reveals the attacker most quickly and all the time after that the level of his trust remains below the threshold value. The time intervals, when an attacker conducts an attack, are marked gray. Thus, the method demonstrates stability when detecting an attacker.

According to the results of the experiments, it can be seen that the developed method effectively detects malicious nodes that implement attacks related to dropping packets and blocking the network operation. In this case, the method demonstrates a higher response rate than analogs. Thus, the proposed model of WSN with built-in trust evaluation mechanisms demonstrates lower temporal expenses for detecting active attacks when an attacker is dropping packets. Built-in confidence mechanisms, as well as analysis of the modeling system and features of the development of the WSN model, allows researchers to implement their own methods of calculating trust and compare them with existing solutions.

ACKNOWLEDGEMENTS

This work was supported by Ministry of Education and Science Initiative research projects № 2.6244.2017/8.9

REFERENCES

[1] T. Beth, M. Borcherding and B. Klein. 1994. Valuation of trust in open networks, Volume 875 Lecture notes in computer science. pp. 3-18.

[2] S.D. Kamvar, M.T. Schlosser and H. Garcia-Molina. 2003. Eigenrep: Reputation management in P2P networks, in World-Wide Web Conference.

[3] R. Sherwood, S. Lee and B. Bhattacharjee. 2006. Cooperative peer groups in NICE, Comput. Netw. 50(4): 523-544.

[4] Y. Renand, A. Boukerche. 2008. Modeling and managing the trust for wireless and mobile ad-hoc networks, in IEEE International conference on communications, ICC2008, pp. 2129-2133.

[5] R. Falcone, G. Pezzulo, and C. Castelfranchi. 2003. A fuzzy approach to a belief-based trust computation. in Lecture Notes on Artificial Intelligence. pp. 73-86.

[6] G. Theodorakopoulos and J.S. Baras. 2004. Trust evaluation in ad-hoc networks, in The 3rd ACM workshop on Wireless security, WiSe'04, pp. 1-10.

[7] A. Josang. 1999. An algebra for assessing trust in certification chains, in The Network and Distributed Systems Security Symposium NDSS99.

[8] G. Lenzini, M.S. Barghand, B. Hulsebosch. 2008. Trust-enhanced security in location-based adaptive authentication. Electronic Notes in Theoretical Computer Science. (197): 105-119.

[9] Renjian Feng, Xiaona Han, Qiang Liu, and Ning Yu. A Credible Bayesian-Based Trust Management Scheme for Wireless Sensor Networks // Hindawi Publishing Corporation International Journal of Distributed Sensor Networks. C. 1-9 DOI: http://dx.doi.org/10.1155/2015/678926

[10] Ganeriwal, L. K. Balzano and M. B. Srivastava. 2008. Reputation based framework for high integrity sensor networks // ACM Trans. Sen. Netw. 4(3), C. 1-37.

[11] E.S. Abramov, E.S. Basan. 2013. Analysis of attack scenarios against wireless sensor networks // Proc. XIII Intl. research conf. «IS-2013». Part 1. - Taganrog: SFedU, pp. 60-72, in Russian.

[12] Elmar Schoch, Michael Feiri, Frank Kargl, Michael Weber. 2008. Simulation of Ad Hoc Networks: ns-2 compared to JiST/SWANS // SIMUTools. Marseille, France.

[13] A.M. Grzhibovsky. 2008. Data types, distribution checks and descriptive statistics. Ekologia cheloveka. pp. 52-58, in Russian.

[14] Alexander Basan, Elena Basan, Oleg Makarevich. 2016. Methodology of Countering Attacks for Wireless Sensor Networks Based on Trust. International Conference on Cyber-enabled distributed computing and knowledge discovery CyberC 2016. Publication Year: pp. 409-412.