International Journal of Computer Applications Technology and Research
Volume 5– Issue 7, 433 - 438, 2016, ISSN:- 2319–8656

A Lightweight Algorithm for Detecting Sybil Attack in Mobile Wireless Sensor Networks using Sink Nodes

Abdolreza Andalib
Department Of Computer Software, Qeshm International Branch, Islamic Azad University, Iran

Mojtaba Jamshidi
Department Of Computer Software, Qazvin Branch, Islamic Azad University, Iran

Farahnaz Andalib
Department Of Computer Software, Kermanshah Branch, Islamic Azad University, Iran

Davod Momeni
Department Of Computer Software, Kermanshah Branch, Islamic Azad University, Iran

Abstract: Considering the application of wireless sensor networks in critical area, such as battlefields, establishing security in these networks is of utmost importance. One of the most serious and dangerous attack against these networks is Sybil attack. In this attack, a malicious hostile node creates multiple fake identities simultaneously. This misleads legitimate nodes and, by mistake, they assume each of these identifiers as real separate nodes. In this attack, malicious hostile node attracts so heavy traffic that can dramatically disrupt routing protocols which has devastating effects on the network functions such as data integration, voting, and resource allocation. The current research proposes a new lightweight algorithm for detecting Sybil attack in Mobile Wireless Sensor Networks using sink nodes. The proposed algorithm is implemented to be assessed in terms of detection and error rates efficiency in a series of experiments. Comparison of the experiment results with the results of other available algorithms revealed optimal performance of the proposed algorithm.

Keywords: Wireless Sensor Networks, Sybil attack, Lightweight algorithm, Sink nodes

1. INTRODUCTION

A wireless sensor network consists of hundreds to thousands small and inexpensive sensor nodes that work together to provide the possibility of monitoring the environment and collecting information. In such networks, usually, there are one to several sink nodes that collect all network data and send commands to one, several or all network nodes. In other words, after capturing (or sensing) information from the environment, sensor nodes send them, step by step, to sink node(s). Being inexpensive and small, sensor nodes have limitations in terms of energy, memory, and computing capability. Because of these limitations, complex encryption and security algorithms of other networks (such as local networks) cannot be applied and set up on the "resource limited" sensor nodes. However, sink nodes do not have such limitations and are usually informed about general network data including encryption keys, number of nodes, nodes identities, etc. [1].

Sybil attack is one of the major attacks that affect the routing layer. In this attack, as indicated in figure 1, a malicious hostile node, after being distributed in the operating network environment, creates multiple fake identities simultaneously, called Sybil nodes (herein after called "Sybil nodes"). In figure 1, each normal node has only one identity while each malicious node creates 6 identities (IDs 5-10). This misleads neighboring legitimate nodes and they assume each of these Sybil nodes is a real separate node, Whereas, all the Sybil nodes are just and only a real (malicious hostile) node. Therefore, the hostile node attracts so heavy traffic and disrupts routing protocols to a large extent which has devastating effects on the network functions such as data integration, voting, and resource allocation [2][3][4].

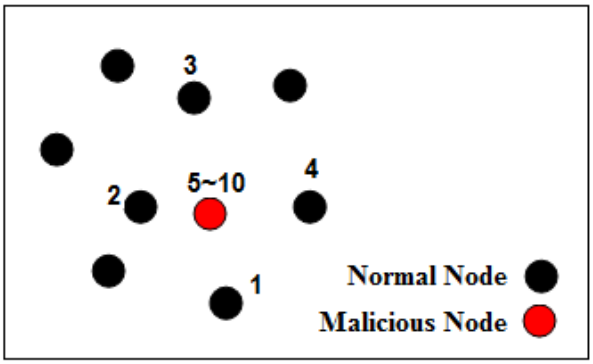

Figure 1 – An example of Sybil attack set-up

So far, various strategies are developed to counteract with Sybil attack in fixed sensor networks. In [5], for instance, an algorithm is proposed, based on radio resource testing, to detect Sybil nodes in which each node assigns a different channel to each of its neighbors to broadcast some message on. However, this method is not efficient considering the limitations of sensor nodes (assigning a separate channel to each neighbor). Also, strategies based on identity verification, such as those proposed in [5] and [6], firstly require a huge memory space and secondly get involved in processing complex checking algorithms in order to store essential identity verification data (including shared encryption keys, identity certificates, etc.). In addition, strategies based on Received Signal Strength Indicator (RSSI) [7], as the one developed in [8], cannot be proper solutions, as well, since the radio signal is susceptible to be interfered by the environment, on one hand, and the malicious node can fail the algorithm by adjusting its sending power, on the other hand. So, using these

strategies for detecting Sybil nodes of mobile sensor networks will not be effective. Because, in the first place, these strategies impose heavy costs (computing, communication and memory) on resource-limited sensor nodes. Second, due to nodes mobility in mobile sensor networks, the above mentioned algorithms either have errors or fail in the process of detecting Sybil nodes.

In this paper, a new lightweight algorithm for detecting Sybil attack in Mobile Wireless Sensor Networks is proposed. The main underlying idea of the proposed algorithm is exchanging a random number between sink and sensor nodes.

The rest of this paper is organized as follows: section 2 reviews the literature. Section 3 explains system hypotheses and the attack model. Section 4 elaborates on the proposed algorithm and section 5 gives the performance evaluation and simulation results. Finally, section 6 concludes the paper.

2. Literature Review

Sybil attack was introduced in [4], for the first time, for peer-to-peer networks. Researchers in [5] analyzed the attack in wireless sensor networks, for the first time, and developed several defense mechanisms including radio resource testing, key validation for random key predistribution, position verification, identifier registration, and remote code verification or code attestation. In radio resource testing, each node assigns a separate channel to its neighbors to broadcast on. In identifier registration approach, a trusted central authority poll the network to identify Sybil nodes. In [7] an algorithm is proposed based on Received Signal Strength Indicator (RSSI) to estimate the location of nodes in the network. In [8] the locating mechanism proposed in [7] is used for detecting Sybil nodes. The algorithm uses four location-aware (routing) nodes that are capable of hearing the packages from all network areas to detect Sybil nodes. When a node sends a package, routing nodes cooperate to estimate its location. It is enough to identify the target nodes because all Sybil nodes are located in a same area. The method which is developed in [9] for detecting Sybil nodes needs no hardware or information about signal strength yet it solely uses data regarding the number of neighbors to identify malicious node and fake (Sybil) identities. The algorithm functions in a distributed manner based on no central point such as base stations or specific nodes (location-aware). Also, [10] proposed an algorithm based on Received Signal Strength Indicator (RSSI) mechanism to detect Sybil attacks in sensor networks that use Low Energy Adaptive Clustering Hierarchy (LEACH) protocol for clustering. [11] Proposed another algorithm based on RSSI technique to detect Sybil nodes when the nodes are adjusting the broadcasting power. In [12] a new algorithm is proposed based on identifying Angle of Arrival (AOA) mechanism called Trust Evaluation Based on AOA (TEBA). Considering that a Sybil nodes can create multiple identities with only one real location, anchor nodes detects Sybil identities that their signal phase difference is less that trust threshold (calculated by assessing trust angle of neighboring sensor nodes). In [13] a method is developed to counteract with Sybil attack which collects route information by collective intelligence algorithm during network activity and detects Sybil node by its energy changes in the meantime. Also, in [14], another RSSI-based algorithm is proposed for detecting Sybil nodes in LEACH routing protocol. IN [16], a new algorithm is proposed based upon customer puzzles and learning automata to deal with Sybil attacks in wireless sensor networks. Added to these, in [17] and [18], other algorithms are proposed that uses guard nodes in detecting Sybil nodes in mobile sensor networks.

3. System Hypotheses and the Attack Model

A sensor network contains n sensor nodes and m sink nodes that are randomly distributed in a two-dimensional area. All sensor nodes are mobile and, according to mobile models (such as Random waypoint), move in the operating environment during the network lifetime. Sink nodes may be mobile or fixed, as well. Each node has one identity and is unaware of its own location. Nodes communicate via wireless radio channels and use Omni-directional distribution approach. Radio range of all nodes (sensor and sink) are the same and equal to r. It is also assumed that sensor nodes cannot resist against interference and enemies can access their confidential information in case of capturing them so as to reprogram them. But, sink nodes are equipped with tamper-resistant hardware and enemies cannot decode and reprogram them.

Here, an attack model is considered, based upon classifications given in [5], i.e. “direct, simultaneous, and stolen identities” Sybil attack. That is, the enemy captures multiple valid identities (e.g. S1~S10) in the network, first. Then, program a malicious node so that it creates S1 to S10 (Sybil nodes) identities simultaneously after being deployed in the network operating environment. In addition, legitimate nodes communicate with Sybil nodes “directly” (not through another node). It should be noted that enemy creates malicious either by itself or by capturing and reprogramming legitimate nodes in the network. Like normal sensor nodes, malicious nodes can move in the operation environment. Considering the attack model, it is assumed that our network has an identity assignment mechanism [19]. In the identity assignment mechanism, the enemy is not capable of creating fake identities; therefore, it has to capture legitimate nodes of the network to set up a Sybil attack. It is also assumed that each node has to send a “Hello” or “route request” message, by arriving at a new location in the network. In fact, this is a requirement for mobile sensor networks so each node can identify its neighbors instantly, set up a security key with them (if necessary), communicate, create its own routing table, etc. [20]. It is obvious that, in this case, each malicious node must send a “hello”, “route request”, etc. message per each of its Sybil identities, after arriving at a new location in the network (simultaneous Sybil attack [5]). Our proposed algorithm detects Sybil nodes by such kinds of communicated messages.

4. The Proposed Algorithm

The main underlying idea of the proposed algorithm is to produce random numbers and exchanging them between sink and sensor nodes so as to detect the Sybil nodes. Generally, the proposed algorithm contains two simple phases that are explained below.

4.1 Configuration

This phase runs before distribution of nodes in the operating environment of the network. According to mechanism given in [19], a single identity is assigned to each sensor node, in the first place. Then, a table, as the one indicated in figure 2a, is loaded on each sensor node and another table, as the one

indicates in figure 2b, is loaded on each sink node; the tables are called history. For every sink nodes and sensor nodes, an identity is registered in SinkID and NodeID column of the tables. The number column belongs to random numbers; the initial value of this field for all sensor and sink nodes is null. In the next place, all nodes have to be distributed in the network environment, randomly.

(b)

NodeID	*number*
N1	
N2	
. . .	
Nn	

(a)

SinkID	*number*
SK1	
SK2	
...	
SKm	

Figure 2 – The structure of history tables in sensor nodes (a) and sink nodes (b) memory

4.2 Test

After being distributed in the environment, th**e** sensor nodes began sending "Hello" (to identify the neighbors), routing, data, and … messages. After a period (t), the nodes start moving in the environment and when they arrive at a new point, they start sending the messages (i.e. 'Hello, routing, data messages), again. Thus, each node can detect its existing neighbors, during different periods of network lifetime, by considering these types of broadcasted messages. Testing phase of the proposed algorithm also runs during alternative periods (t) of network lifetime. The testing phase runs as follows: whenever there is a sensor node (N_i) in the vicinity of a sink node (SK_j), the sink node generates a random number (p) and stores it in the number column of its history table if it learns that there is an empty field in the number column of N_i. Then, it sends the random number (p) to N_i, along with a message containing $<< SK_j, p >>$. N_i updates its history as it receives the message; in other words, it registers p in the number field of SK_j. But, if the number field of N_i in SK_j history has already been filled with a random number, like p', (i.e. the sink node (SK_j) has sent a random number (p') to N_i before) the sink node (SK_j) requests the random number from N_i. Accordingly, N_i sends the random number that exists in its history, like p", to the sink node. If p" be equal to p', the sink node considers N_i as a legitimate node; otherwise, it regards N_i as a malicious Sybil node.

Now, consider a scenario in which, SK_j sends p' to a sensor node (N_i) during t_1. Also, assume that an enemy steals legitimate identities ($N_{i\sim k}$) from the network and a malicious node is programmed itself to distribute the stolen Sybil identities, i.e. $N_{i\sim k}$, after being scattered in the network environment. If during $t_l > t_1$ periods, there is a Sybil node (N_i) in the vicinity of a sink node (SK_j), it request the Sybil node (N_i) to return the random number because there is no empty member field for N_i in history of the sink node (SK_j) (it is filled with p'). But if the Sybil node (N_i) generates a random number, since it has no random number available in history for the sink node, it can register the number in its history and return it to the sink node. Hope that the random number be the same number that the sink node (SK_j) has requested (i.e. p'). If so, the Sybil node (N_i) cannot be detected. If not, the sink node marks N_i as a malicious node. In this case, Sybil node's success depends on complexity of the random number. To avoid suspicion to these random numbers by the malicious nodes, a 32- or 64-bit number field can be used.

In another scenario, assume that a sink node (SK_j) sends a random number (p') to a Sybil node (N_i) during t_1 period (i.e. the first time that the sink node (SK_j) identifies a N_i node in its vicinity). Now, if during $t_l > t_1$ periods, the sink node (SK_j) identifies a N_i node in its vicinity, it request the Sybil node (N_i) to return the random number because it has no empty member field for N_i in its history. Since, the N_i node, that is now (during t_l) in the vicinity of SK_j, is a legitimate node (not a Sybil node (N_i)) and has no random number in its history for the sink node (SK_j), it sends a warning message to the sink node. Upon receiving the alarm, the sink node understands that N_i is captured by enemy. Therefore, it considers N_i as a Sybil node. It should be noted that each sink node generates and sends the random number to N_i just and only once; however, it may request the N_i (when the node is in its vicinity) to return the random number several times during the network lifetime.

5. Performance Evaluation and Simulation Results

In this section, we first evaluate the proposed algorithm memory, communication, and processing overheads, then present the simulation results.

5.1 Performance Evaluation

Memory Overheads: in the proposed algorithm, each sensor node requires a memory space with O (m) function and each sink node requires a memory space with O (n) function to store data in history. Since sink nodes are not resource-limited, memory overheads with O (n) function can be imposed on these nodes. Yet, given that the number of sink nodes in most of the sensor networks are generally are too small (less than 10), sensor nodes of the memory overheads tolerate very little memory.

Communication overheads: Assuming that each node has d neighbors, on average, each sink node "sends" or "requests" d random numbers during each time period. Also, each sensor node, sends at least 0 and at most m message for the sink nodes in each time period, during the test phase. Because it may have no sink node or many sink nodes in its vicinity. Therefore, connection overheads of sensor nodes and sink nodes are of O (m) and O (d) type, respectively.

Processing Overheads: in each time period, during the test phase, each sink node has to update its history (which is of O (d x n) time order) per all its existing neighbors and each sensor node requires at most O (m^2) processing time to update its history (by assuming a linear search), as well. Because in any of the time periods during the test phase there may appear m sink nodes in the vicinity.

5.2 Simulation Results

After running the proposed algorithm, its performance is evaluated through a series of tests. The main criterion in the evaluation is detection rate; i.e. the amount of Sybil nodes

detected by a security algorithm. What is more, the proposed algorithm performance is compared with those given in [8-10,13,14,16], in terms of average detection rate and average rate of error.

In simulation, it is assumed that the network includes n sensor nodes and m sink nodes which are randomly distributed in 100 x 100 m^2 area. The function area has W=5 malicious hostile nodes (Sybil attack runners) and each malicious node creates S Sybil identity. So, (W x S) Sybil nodes are available in the network. All the nodes (normal and malicious) has equal radio range (r=10). In addition, the mobility model of [21] is employed for the nodes movement in the function area. In order to be sure of the results validity, each test is repeated 100 times and the final result is the average of these 100 tests.

Experiment 1: parameters considered in this experiment are n=30, S=20, and m=1, 2, 3. Detection rate of the proposed algorithm is evaluated in time periods 25 to 200 during the test phase. Figure 3 indicates the experiment results. The results indicate that increasing time periods of the test phase increases the Sybil nodes detection rate. For example, when the number of sink node equals to m=1, detection rates of time period 25, 100, and 200 are about 27%, 88%, and 100%, respectively. The reason is clear; assume that enemy has captured N_i and its malicious node creates a Sybil node with N_i identity, after being distributed in the network. In this case, a sink node can detect Sybil N_i only if it faces with both legitimate N_i and malicious hostile node (in its vicinity) during the test phase (not necessarily in a certain time period). Accordingly, by increasing the number of phase test periods, it is more probable that the sink node faces with both N_i and the malicious node.

Also, since, in the proposed algorithm, sink nodes detects Sybil nodes in a fully independent way, its detection rate increases by increasing the number sink nodes (m). Figure 3 clearly indicates the test results.

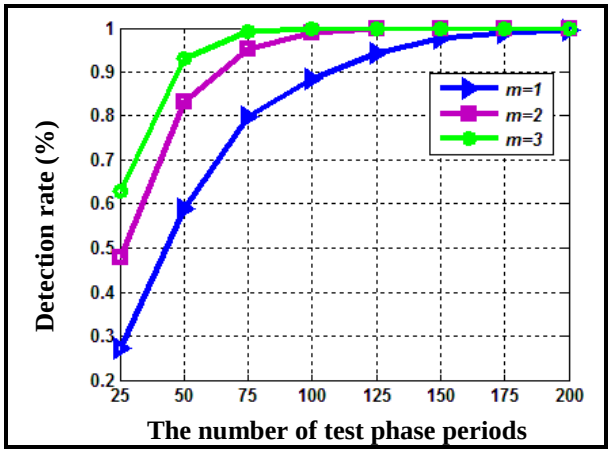

Figure 3 – Detection rate of the proposed algorithm per different parameter (m) values and different test phase periods

Experiment 2: This experiment is to evaluate the effect of Sybil identities number, distributed by malicious nodes (S), on the proposed algorithm performance. Parameters considered in this experiment are n=300 and m=1. Detection rate of the proposed algorithm is evaluated for Ss 4 to 20 (by increasing 4 units per step). Figure 4 indicates the experiment results for time periods 25 to 200 during the test phase. The experiment results prove that changes of S have no significant effect on detection rate of the proposed algorithm because what matters the most in this algorithm is the presence of malicious and legitimate nodes, the identities of which is captured by the enemy, in the vicinity of sink nodes during the test phase. But, considering the Sybil attack model assumed here (i.e. "simultaneous" model), whenever the malicious node moves to a new location in the network, it sends "Hello", routing, etc. messages, per each of its Sybil identities. So, the number of distributed Sybil identities of the malicious node does not matter; however, it is enough to be in the neighboring environment of the both malicious and legitimate nodes, with captures identity, only once, then the sink node can detect the Sybil nodes.

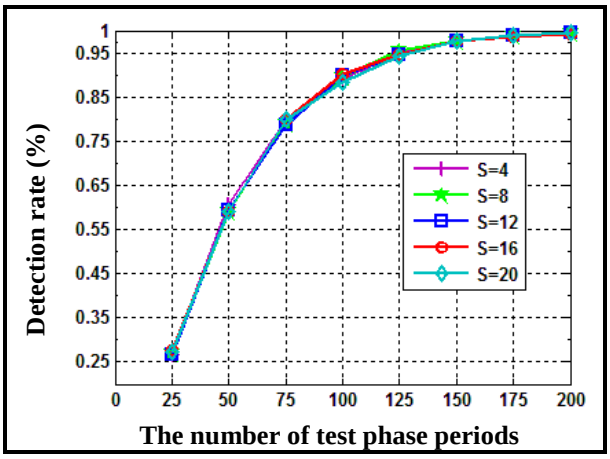

Figure 4 – the effect of parameter (S) on detection rate of the proposed algorithm

Experiment 3: This experiment is to evaluate the effect of network nodes number (n) on the proposed algorithm performance. Parameters considered in this experiment are S=10 and m=1. Detection rate of the proposed algorithm is evaluated for ns 100 to 500. Figure 5 indicates the experiment results for time periods 25 to 200 during the test phase. The experiment results prove that network nodes number (n) has no significant effect on detection rate of the proposed algorithm because, unlike other algorithms, such the one developed in [9], the proposed algorithm is not based on network density but on confrontation of malicious and legitimate nodes, the identities of which is captured by the enemy, in the vicinity of sink nodes. Therefore, increase or decrease of the number of nodes in the network leaves no effect on detection rate of the proposed algorithm.

Experiment 4: in this experiment**,** the proposed algorithm performance is compared with others in terms of **average detection rate and average rate of error.** Figure 6 and figure 7 indicate the average detection rate and average rate of error. The average detection rate of the proposed algorithm, [8], and [16] are almost 100% which is better than others. Also, it is worth noting that, unlike other algorithms, the proposed algorithm makes no error in detecting Sybil nodes. Whereas, error rates of other algorithms in [8] and [16] are 6% and 5%, respectively. It is because the other algorithms are based on information collected from the neighborhood or on RSSI. So, they may have error in detecting Sybil nodes. But, the underlying idea of the proposed algorithm is confrontation of malicious nodes with sink and legitimate nodes, the identities of which is captured by the enemy.

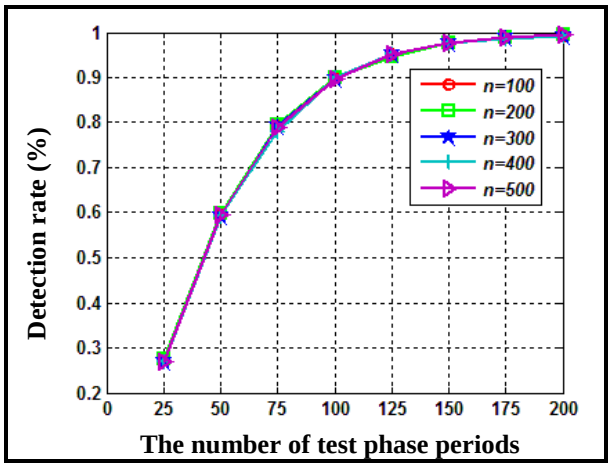

Figure 5 – The effect of parameter (S) on wrong detection of the proposed algorithm

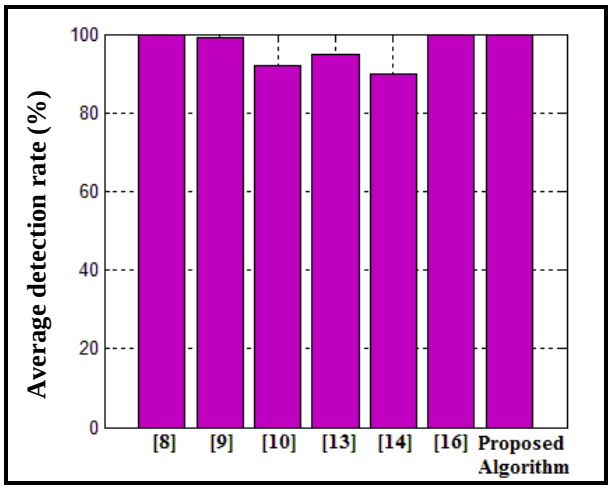

Figure 6 – Comparison of the average detection rate of the proposed algorithm with other algorithms

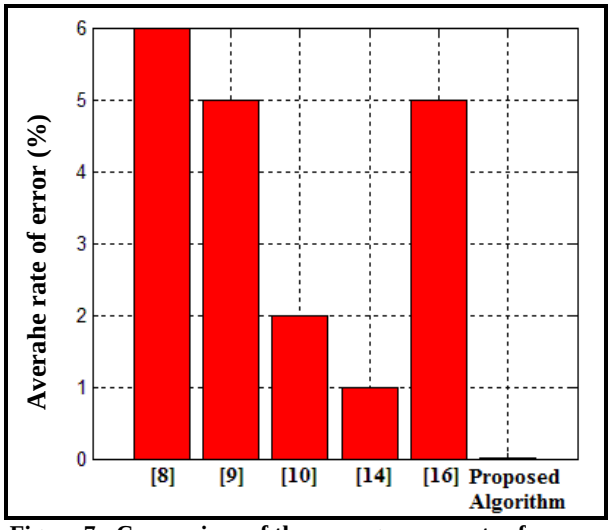

Figure 7 - Comparison of the average error rate of the proposed algorithm with other algorithms

6. Conclusion

The current research proposes a new lightweight algorithm for detecting Sybil attack in Mobile Wireless Sensor Networks using sink nodes. The main underlying idea of the proposed algorithm is generate random numbers by sink nodes and exchange them between sink and sensor nodes to detect Sybil nodes. The proposed algorithm is implemented to be assessed in terms of detection and error rates efficiency in a series of experiments. Comparison of the experiment results with the results of other available algorithms revealed optimal performance of the proposed algorithm.

7. REFERENCES

[1] Akyildiz Ian F. and Kasimoglu Ismail H., "Wireless sensor and actor networks: research challenges", in: Proceedings of the Ad Hoc Networks 2, pp. 351–367, 2004.

[2] Karlof C. And Wagner D, “Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures”, in: Proceedings of the AdHoc Networks, pp. 299-302, year 2003.

[3] Padmavathi G. and shanmugapriya D, "A survey of attacks, security mechanisms and Challenges in Wireless sensor networks", in: Proceedings of the International Journal of Computer Science And Information Security (IJCSIS), Vol. 4, No. 1 & 2, August 2009.

[4] Douceur J. R., “The Sybil attack”, in: Proceedings of the Douceur J. R., “The Sybil attack”, in: Proceedings of the First International Workshop on Peer-to-Peer Systems (IPTPS '02), 2002.

[5] Newsome J., Shi E., Song D. and Perrig A., “The Sybil attack in sensor networks: analysis and defenses”, in: Proceedings of the International Symposium on Information Processing in Sensor Networks, pp. 259–268, April 2004.

[6] D. Liu, P. Ning, “Establishing pairwise keys in distributed sensor networks”, in: Proceedings of the ACM Conference on Computer and Communications Security, pp. 52–61, October 2003.

[7] Zhong S., Li L., Liu Y. G. and Yang Y. R., "Privacy-preserving location based services for mobile users in Wireless Networks", In: Proceedings of the Technical Report YALEU/DCS/TR-1297, Yale Computer Science, 2004.

[8] Demirbas M. and Song Y., "An RSSI-based scheme for Sybil attack detection in wireless sensor networks", In: Proceedings of the IEEE Computer Society International Symposium on World of Wireless, Mobile and Multimedia Networks, pp. 570–574, 2006

[9] Ssu K. F, Wang W. T. and Chang W. C., “Detecting Sybil attacks in wireless Sensor Networks using neighboring information”, in: Proceedings of the Computer Networks 53, pp. 3042–3056, 2009.

[10] Chen S., Yang G. and Chen S., “A Security Routing Mechanism against Sybil Attack for Wireless Sensor Networks”, in: Proceedings of the International Conference on Communications and Mobile Computing, 2010.

[11] Misra S. and Myneni S., “On Identifying Power Control Performing Sybil Nodes in Wireless Sensor Networks Using RSSI”, in: Proceedings of the IEEE Communications Society, 2010. http://www.csee.usf.edu/~labrador/Share/Globecom/DATA/03-384-04.PDF

[12] ZHANG Y., FAN K.-.F., ZHANG S.-B. and MO W., ”AOA based trust evaluation sheme for Sybil attack detection in WSN”, in: Proceedings of the journal on Application Research of Computers, 2010.

[13] Muraleedharan R., Ye X. and Osadciw L.A., “Prediction of Sybil Attack on WSN Using Bayesian Network and Swarm Intelligence”, in: Proceedings of the Wireless Sensing and Processing, Orlando, FL, USA, March 2008.

[14] Jangra A., Swati, Priyanka, “Securing LEACH Protocol from Sybil Attack using Jakes Channel Scheme (JCS)”, in: Proceedings of the International Conferences on Advances in ICT for Emerging Regions(ICTer2011), 2011.

[15] Jiangtao W., Geng Y., Yuan S. and Shengeshou C., “Defending Against Sybil Attacks Based on Received Signal Strength in Wireless Sensor Networks”, in: Proceedings of the journal of elctronics, Vol. 17,No. 4, Oct. 2008.

[16] Jamshidi, M., Esnaashari, M. and Meybodi, M. R., "An Algorithm for Defending Sybil Attacks based on Client Puzzles and Learning Automata for Wireless Sensor Networks", in: Proceeding of 18th National Conference of Computer Society of Iran , Sharif University, Tehran, Iran, March 14-16, 2013.

[17] Jamshidi, M., Esnaashari, Nasri A., Hanani A. and Meybodi, M. R., "Detecting Sybil Nodes in Mobile Wireless Sensor Networks using Observer Nodes”, in: Proceeding of 10th International ISC Conference On Information Security &Cryptologhy, Computer Society of Iran , yazd University, yazd, Iran, August 29-30, 2013.

[18] Rezai A., Jamshidi M. and AkbariTorkestani J., "A lightweight and robust algorithms to detect Mobile Sybil Nodes in Moblie Wireless Sensor Networks using Information about the mobility of nodes”, in: Proceeding of 10th International ISC Conference On Information Security &Cryptologhy, Computer Society of Iran , yazd University, yazd, Iran, August 29-30, 2013.

[19] Butler K. and et al., “Leveraging Identity-Based Cryptography for Node ID Assignment in Structured P2P Systems”, in: Proceedings of the IEEE transaction on parallel and distributed systems, Vol. 20, 2009.

[20] Piro C., Shields C. and Levine B. N. , ”Detecting the Sybil Attack in Mobile Ad hoc Networks”, in: Proceedings of the Securecomm and Workshops, pp 1-11, 2006.

[21] Yu C. M., Lu C. S. , and Kuo S. Y., “Mobile Sensor Network Resilient Against Node Replication Attacks” In: Proceedings of the IEEE Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON), June 2008.