VOL 3 (2019) NO 1

e-ISSN : 2549-9904
ISSN : 2549-9610

INTERNATIONAL JOURNAL ON INFORMATICS VISUALIZATION

A Dynamic ID Assignment Mechanism to Defend Against Node Replication Attack in Static Wireless Sensor Networks

Mojtaba Jamshidi[#], Abdusalam Abdulla Shaltooki[#], Zahra Dagal Zadeh[*], Aso Mohammad Darwesh[#]

[#] *Department of Information Technology, University of Human Development, Sulaimani, Iraq*

[*] *Department of Computer Engineering, Kermanshah Branch, Islamic Azad University, Kermanshah, Iran*

E-mail: jamshidi.mojtaba@gmail.com, salam.abdulla@gmail.com, *dagalzadeh_67@yahoo.com,* aso.darwesh@uhd.edu.iq

***Abstract*— One of the known dangerous attacks against wireless sensor networks (WSNs) is node replica. In this attack, adversary captures one or more normal nodes of the network, generates copies of them (replicas) and deploy them in the network. These copied nodes are controlled by the adversary which can establish a shared key with other nodes of the network easily and exchange information. In this paper, a novel algorithm is proposed to defend against this attack in static sensor networks. The proposed algorithm employs a multi-tree architecture to assign ID to the nodes dynamically and prevent attachment of the injected replica nodes to the network by the adversary. The efficiency of the proposed algorithm is evaluated in terms of memory, communication, and computation overheads and the results are compared with other existing algorithms. Comparison results indicate the superiority of the proposed algorithm in terms of mentioned measures. In addition, the proposed algorithm is simulated and its efficiency is evaluated in terms of probability of detecting replica nodes. Experiment results show that the proposed algorithm has favorable performance in detection of replica nodes.**

***Keywords*— Static Wireless Sensor Networks, Replica Node Attack, dynamic ID Assignment, Multi-tree Structure**

I. INTRODUCTION

Today, WSNs are widely used in many applications including environment, military, and explorations. Since sensor nodes (SNs) have low computation, memory and radio capacity and they are applied in critical conditions especially in the military, security establishment in these networks is very important and has attracted the attention of many researchers [1][2].

One of the dangerous attacks in WSNs is node replication attack or replica node. An adversary might capture one or more nodes of the network and extract important information including its keying material. Replica nodes are able to establish a key with legal nodes. An adversary can inject these replica nodes into the network and implement various attacks. Replica nodes are controlled by the adversary but they have locking information which allows them to seem like legal nodes of the network. Protocols which are used for secure communication in SNs allow replica nodes to establish pairwise keys with other nodes and the base station. Therefore, these replica nodes are able to encrypt, decrypt and verify all communications.

An adversary can exploit this inter-network position in different ways. For instance, the adversary can monitor a major part of the network traffic passing through replica nodes, destruct monitoring operation of the sensors by injecting distorted data and disrupt common WSN protocols including clustering and data aggregation [3][4][5].

Till now, algorithms like [6-14] have been proposed to defend against replica node attack in static sensor networks which are mainly based on the transmission of location claim messages towards witness nodes or locations in the network. Such algorithms have high memory and communication overhead. In [15-23], algorithms have been proposed to defend against replica node attack in mobile sensor networks which cannot be employed in static sensor networks.

In this paper, a novel algorithm based on a dynamic ID assignment mechanism is proposed to defend against replica node attack in static WSNs.

The rest of this paper is organized as follows. Section II presents previous work, system assumption, attack model, and the proposed algorithm. Section III discusses the performance evaluation and simulation results. The paper is concluded in Section IV.

II. MATERIAL AND METHOD

In this section, we first present some existing algorithms which are proposed to defend against node replication attack in static wireless sensor networks. Then, we present the

assumptions and the attack model. Finally, the proposed algorithm is presented.

Related Work

In [6], four probability distributed algorithms called NNB, DM, RM, and LSM have been proposed to detect replica nodes in static sensor networks which employ public key encryption and transmission of location claim messages to witness nodes for detecting replica nodes.

In [7], another protocol called SET has been presented for detecting replica nodes. SET employs set operations (union and intersection) on subsets of the network to detect replica nodes. In [8], two other algorithms called SDC and P-MPC based on Localized Multicast or LM have been used to detect replica nodes. These algorithms operate in sensor networks with grid topology. In the SDC algorithm, a geographical hash function [24] is used for unique and random mapping of node ID L to a cell in the grid. The difference of P-MPC with SDC is in the selection of the destination cell for transmission of location claim messages. In SDC, each location claim is transmitted to a singular cell but in P-MPC, location claim is transmitted and mapped to several cells with different probabilities.

In [9], a centralized algorithm called RED has been proposed which its main idea is to transmit location claims to locations of the network selected based on a random value broadcast by a central point (periodically). In [10], the RED algorithm has been investigated in detail. In [11], a distributed, deterministic and flexible algorithm called DDR has been proposed which lies on a node-witness-based strategy. In DDR, when a location claim message is transmitted from a node to a verified destination, compatibility of the messages in the intermediate nodes existing along the path towards the final destination is investigated.

In [12], two other algorithms called RAWL and TRAWL have been proposed. In RAWL, for each node u, several hops are taken randomly in the network and nodes which have been passed are selected as witnesses of node u. Analyses on TRAWL are based on RAWL and a trace table is added to each node to reduce memory cost. In [13], an algorithm based on compressive sensing called CSI has been proposed for detection of replica nodes. The main idea of CSI is that each node broadcasts a constant value a to its single-hop neighbors. Constant value a can be considered as data sensed by each sensor node. Sensor nodes aggregate or transmit numbers received from descendant nodes along aggregation tree using data aggregation techniques. The base station as the root of the aggregation tree receives aggregated data and stores network's sensed data. The base station detects replica nodes considering the stored data. In [14], four algorithms have been proposed to detect replica node attach which employs Bloom filters [25] to compress information stored in sensors and two cell forwarding and cross forwarding techniques to increase detection rate.

Algorithm [15] is based on the generation and exchange of random numbers among nodes and algorithm [16] is based on the movement speed of nodes in the environment. In [17], another algorithm called EDD has been proposed which its main idea is inspired by the issue that a network without replica node, in a specific period of length T, number of times that node u faces a specific node v should be very limited. For a network with two replica nodes v, the number of times that node u faces node v in a period of length T should be larger than a threshold. In [18], an algorithm based on network segmentation been proposed which divides the network environment to separate sectors where each sector has a central node which can operate both as environment sensor and detect replica node attacks. Each central node in each sector keeps ID list and location of the existing nodes.

In [19], pairwise key and Bloom filter have been used to present a centralized algorithm for detecting replica nodes in mobile sensor networks which does not require location information of the nodes. In [20], the routing algorithm has been developed using mobility to present a penetration detection algorithm for mobile sensor networks. In general, detection procedure in SHD is based on the transmission of <ID, neighbor-list> message to nodes in their radio range when the protocol begins and then employing query methods. In [22], another algorithm has been proposed for the detection of replica nodes in mobile sensor networks which employs sign based ID authentication to detect replica nodes. In [23], another algorithm has been proposed which only employs single-hop communications and node mobility to detect replica nodes in mobile sensor networks.

System Assumptions and Attack Model

In this study, it is assumed that the network contains n sensor nodes which are distributed randomly in a 2D area. In addition, the network contains S sink nodes which deploy along the operational environment. Each node has a unique primary ID, PID and remains static after deployment in the network environment. Nodes are not aware of their local position. Nodes communicate with each other through a wireless radio channel and employ omnidirectional broadcast. All nodes, except sink nodes, have the same software and hardware facilities (in terms of radio range, memory, and energy).

It is also assumed that the sensor network is deployed in a hostile environment, therefore, this network is insecure and the adversary can capture some of the nodes and generate replications of them and inject them into the network. Indeed, it is assumed that network nodes are secure from attack for at least Test after deployment in the operational environment [26].

The Proposed Algorithm

The main idea of the proposed algorithm is to use a multi-tree architecture based on multi-sink [27, 28] for dynamic assignment of ID to sensor nodes after deployment in the operational environment. If this mechanism is used, replica nodes generated by the adversary cannot be easily attached to the network. In the following, multi-sink architecture and the proposed algorithm are described in detail.

As can be seen in Fig. 1, in multi-sink architecture, there are several sink nodes which settle at one side of the operational area. Sink nodes can communicate with each other and communicate with the base station (location of the network manager) directly. Sensor nodes sense environment data and transmit required data to the sink or sinks in multiple hops. In this architecture, data received by at least one of the sinks is the sufficient condition for data delivery. In other words, it is not important that the packet generated by a node is delivered to which sink node, but it is sufficient

that the packet is delivered to one of the sinks. This architecture has two main advantages:

Increasing the lifetime of the network: if there is only one sink in the network, sensor nodes around this sink transmit a lot of the traffic; thus, their energy is deprived quickly. But if multi-sink architecture is used, this problem is resolved.

High data delivery rate: it is obvious that if there are several sinks in the network, packet delivery probability is increased. Because the probability that there exists a path from source node to the destination node (sink), especially in low-density networks, is increased.

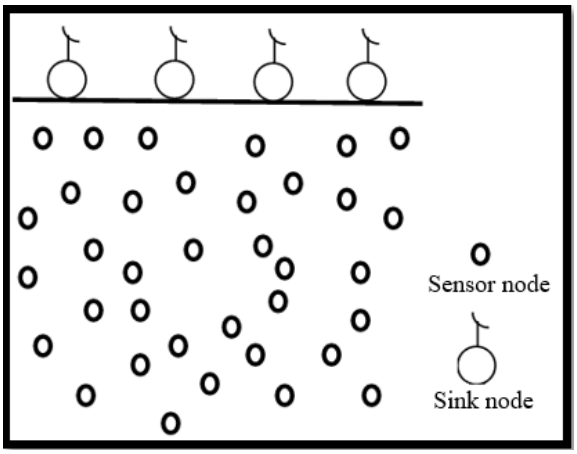

Fig. 1 Multi-sink architecture in WSNs

After deployment of the nodes in the environment, all sink nodes create and broadcast a Route Generate Packet (RGP) simultaneously. Structure of the route generates packets is given in Fig. 2. Field SinkID is the ID of the sink node, field SenderID is ID of the packet transmitter node and field Level is level of the packet transmitter node in the tree.

SinkID	*SenderID*	*Level*

Fig. 2 Structure of the RGPs

For instance, sink SK1 generates and broadcasts an RGP with content $<<SinkID=SK_1,\ SenderID=SK_1,\ Level=0>>$. Each sensor node with in the neighborhood of sink SK1 receives this packet. Sensor node u opens the RGP and since the value of its Level is 0 and its SinkID is SK1, it considers itself as level 1 in the routing tree of SK1. Thus, it should first register its final ID, FID and then broadcast the RGP for the lower level nodes. The final ID of a sensor node u is obtained using Eq. (1):

$$F_{ID} = P_{ID} \| L \| SK_j \qquad (1)$$

Where L is level of node u in the routing tree and SK_j is the root of that. In fact, the final ID of a node is obtained by attaching primary ID, its level in the routing tree and ID of the corresponding sink. In general, if node v is in the ith level of the routing tree of sink SK_j, its final ID would by $v\|i\|j$.

Therefore, in the above example, node u changes its ID to u11 and changes the RGP to $<<SinkID=SK_1,\ SenderID=u11,\ Level=1>>$ and broadcasts it. Indeed, by receiving an RGP p by each node, the node discards the packets received previously. This trend is continued until all route generate packets are delivered to total accessible nodes of the network. When this procedure is finished, a virtual cell structure is created as shown in Fig. 3. In fact, each level of the sink tree corresponds to a cell and sensor nodes located in this level of the tree would be static nodes of this cell.

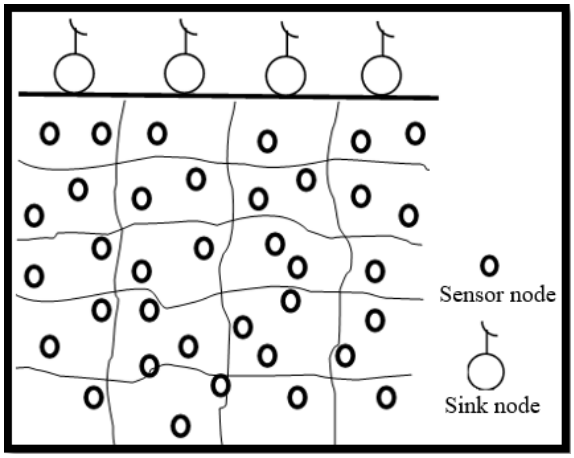

Fig. 3 virtual cell of the network after applying the proposed network

This procedure is executed in T_{est} from the beginning of the network life and it should be noted that time is so short that an adversary cannot interfere with. After this stage, network nodes perform their mission which is sending and receiving data. Now, if the adversary enters the network environment and captures a sensor node like *uij* (node *u* at level *i* of routing tree of sink SK_j) and decodes it then he/she creates several replications of that node and injects them in the network. replication nodes, after deployment in the network, broadcast a Hello message to join the network.

Each legal node *vkl* (node *v* at level *k* of SK_l tree) operates as follows upon receiving the Hello message from replica nodes *uij*:

- If $|i-k|>1$ *or* $|j-l|>1$ is satisfied, node *vkl* identifies *uij* as a malicious node and does not communicate with it. the hello message transmitted from a node is delivered to the nodes existing in its adjacent cells. Therefore, if a node receives a Hello message from a node which does not belong to the adjacent cells, it should be considered as a malicious node.
- Otherwise, node *vkl* identifies *uij* as a valid node and communicate with it. Therefore, if replica nodes are settled in the cell from which the adversary has captured a node, they can cheat the legal nodes and communicate with them. It is obvious that replica nodes cannot perform effectively in this situation.

When a node detects a malicious replica node can inform other nodes of its cell and adjacent cells by issuing an alarm.

III. Results And Discussion

In this section, we first evaluate the overhead of the proposed algorithm in terms of memory, communication, and computation. Then, simulation results of the proposed algorithm are presented in terms of probability of detecting replica nodes.

A. *Overhead Evaluation*

Memory overhead: the memory overhead of the proposed algorithm is zero. Because when the algorithm is executed, nodes do not require to store a specific data. While other algorithms impose a large overhead to sensor nodes. Table 1 compares the memory overhead of the proposed algorithm and other algorithms. In Table 1, n is the total number of nodes in the network.

Communication overhead: considering the energy constraints of sensor nodes, energy consumed by the proposed algorithms is very important for sensor networks. Since packet transmission consumes more energy compared to packet processing and packet reception, the calculating number of transmitted packets which is imposed to the network due to using a specific algorithm (known as communication overhead), is an important measure for evaluating the efficiency of the algorithm proposed for sensor nodes. In the proposed algorithm, each node only transmits one RGP. Therefore, the communication overhead of this algorithm is $O(1)$. Table 1 also compared the communication overhead of the proposed algorithm with other algorithms and the results indicate the favorable efficiency of the proposed algorithm.

Computational overhead: no computation overhead is imposed on the sensor nodes for executing the proposed algorithm except calculating the final ID by equation (1).

TABLE 1
COMPARING MEMORY AND COMMUNICATION OVERHEAD OF THE PROPOSED ALGORITHM WITH OTHER ALGORITHMS

Algorithm	Memory overhead	Communication overhead
LSM[6]	$O(\sqrt{n})$	$O(n\sqrt{n})$
SET[7]	$O(\frac{n}{T})$	$O(n)$
P-MPC, SDC [8]	$O(w)$	$O(r\sqrt{n})+O(s)$
RED[9]	$O(d)$	$O(n\sqrt{n})$
RAWL[12]	$O(\log n\times\sqrt{n})$	$O(\log n\times\sqrt{n})$
XED[15]	$O(4\times d\times E[X])$	$O(1)$
SPRT[16]	$O(n\sqrt{n})$	$O(n)$
EDD, SEDD [17]	$O(1)$ / $O(n)$	$O(n)$
Algorithm [18]	$O(d)$	$O(n\times\log n)$
The proposed algorithm	0	$O(1)$

B. *Simulation Results*

In order to evaluate the efficiency of the proposed algorithm, a number of experiments have been performed, the obtained results are compared with other algorithms. The evaluated measure is detection probability.

In order to simulate the network environment, JSIM simulator [29] is used. In the simulations, it is assumed that the network contains n sensor nodes which are distributed randomly in a 250m*250m area. adversary captures a node and creates R copies of that and injects them in the network randomly. The number of sink nodes is selected such that one side of the operational area is covered. Transmission range of nodes is considered to be r. In order to assure the validity of the results, each simulation is repeated 20 times and the final result is obtained by averaging results of these 20 repetitions.

Experiment 1:

In this experiment, the transmission range of each node is r=10m and number of replica nodes from the captured node is varied from R=2~6 and the results are evaluated for different values of n. Results of this experiment are shown in Fig. 4.

Results of this experiment show that a number of replica nodes increases, detection probability of the proposed algorithm increases because the probability that at least one of the replica nodes is located in a cell farther from its particular cell (cell corresponding to the captured nodes) increases, thus it is detected easily.

In addition, results of this experiment show that as network density increases, n, the probability of detecting replica nodes also increases. When network density is low, some nodes of the network might be isolated and do not join any tree. Moreover, replica nodes might settle in a location in which none of the neighbors are legal nodes. Thus, detection probability is decreased.

Experiment 2:

Purpose of this experiment is to compare the efficiency of the proposed algorithm with other algorithms in terms of detection probability. In this experiment, the total number of nodes is n=1000. In addition, in the most difficult case of applying replica node attack, R=2. Radio range of nodes is adjusted such that each node has almost d=20 neighbors. Table 2 shows the list of evaluated algorithms along with adjusted parameters and the obtained results. As can be seen from the results, the proposed algorithm with the detection probability of 0.98 outperforms LSM, B-MEM, BC-MEM, C-MEM and CC-MEM with detection probabilities of 0.89, 0.86, 0.93 and 0.95.

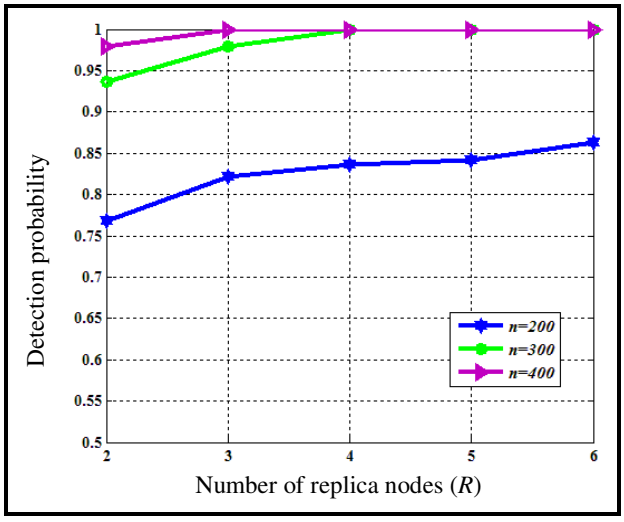

Fig. 4 Detection probability of the proposed algorithm for different values of n and R

TABLE 2
COMPARING THE EFFICIENCY OF THE PROPOSED ALGORITHM AND 5 OTHER ALGORITHMS IN TERMS OF DETECTION PROBABILITY OF REPLICA NODES

Algorithm	Parameters	Detection probability
LSM [6]	# line segment=6	0.89
B-MEM [14]	# line segment=6	0.86
BC-MEM [14]	# line segment=5	0.93
C-MEM [14]	-	0.95
The proposed algorithm		0.98

IV. CONCLUSIONS

In this paper, a novel algorithm is proposed to defend against replica node attack in sensor networks. the proposed algorithm employs a multi-tree multi-sink architecture for dynamic detection of nodes. The efficiency of the proposed algorithm is evaluated in terms of memory, communication and computation overhead and the results are compared with results of other algorithms. Comparison results show that the proposed algorithm outperforms other algorithms. In addition, the proposed algorithm is implemented in JSIM simulator environment and several experiments are performed to evaluate the efficiency of the proposed algorithm in terms of detection probability and the results indicate the favorable efficiency of the proposed algorithm.

REFERENCES

[1] Akyildiz, I.F. and Kasimoglu, I.H., 2004. Wireless sensor and actor☆ networks: research challenges. *Ad hoc networks*, *2*(4), pp.351-367.

[2] Jamshidi, M., Zangeneh, E., Esnaashari, M. and Meybodi, M.R., 2017. A lightweight algorithm for detecting mobile Sybil nodes in mobile wireless sensor networks. *Computers & Electrical Engineering*, *64*, pp.220-232.

[3] Jamshidi, M., Ranjbari, M., Esnaashari, M., Qader, N.N. and Meybodi, M.R., 2018. Sybil Node Detection in Mobile Wireless Sensor Networks Using Observer Nodes. JOIV: *International Journal on Informatics Visualization*, 2(3), pp.159-165.

[4] Dhakne, A.R. and Chatur, P.N., 2017. Detailed Survey on attacks in wireless sensor network. In *Proceedings of the International Conference on Data Engineering and Communication Technology* on (pp. 319-331). Springer, Singapore.

[5] Padmavathi, D.G. and Shanmugapriya, M., 2009. A survey of attacks, security mechanisms and challenges in wireless sensor networks. *arXiv preprint arXiv:0909.0576*.

[6] Parno, B., Perrig, A. and Gligor, V., 2005, May. Distributed detection of node replication attacks in sensor networks. In *Security and Privacy, 2005 IEEE Symposium on* (pp. 49-63). IEEE.

[7] Choi, H., Zhu, S. and La Porta, T.F., 2007, SET: Detecting node clones in sensor networks. In *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on* (pp. 341-350). IEEE.

[8] Zhu, B., Addada, V.G.K., Setia, S., Jajodia, S. and Roy, S., 2007. Efficient distributed detection of node replication attacks in sensor networks. In *acsac* (pp. 257-267). IEEE.

[9] Conti, M., Di Pietro, R., Mancini, L.V. and Mei, A., 2007. A randomized, efficient, and distributed protocol for the detection of node replication attacks in wireless sensor networks. In *Proceedings of the 8th ACM international symposium on Mobile ad hoc networking and computing* (pp. 80-89). ACM.

[10] Conti, M., Di Pietro, R., Mancini, L. and Mei, A., 2011. Distributed detection of clone attacks in wireless sensor networks. *IEEE transactions on dependable and secure computing*, *8*(5), pp.685-698.

[11] Kim, C., Park, C., Hur, J., Lee, H. and Yoon, H., 2009. A distributed deterministic and resilient replication attack detection protocol in wireless sensor networks. In *Communication and Networking* (pp. 405-412). Springer, Berlin, Heidelberg.

[12] Zeng, Y., Cao, J., Zhang, S., Guo, S. and Xie, L., 2010. Random-walk based approach to detect clone attacks in wireless sensor networks. *IEEE Journal on selected areas in communications*, *28*(5).

[13] Yu, C.M., Lu, C.S. and Kuo, S.Y., 2012. CSI: compressed sensing-based clone identification in sensor networks. In *Pervasive Computing and Communications Workshops (PERCOM Workshops), 2012 IEEE International Conference on* (pp. 290-295). IEEE.

[14] Zhang, M., Khanapure, V., Chen, S. and Xiao, X., 2009. Memory efficient protocols for detecting node replication attacks in wireless sensor networks. In *Network Protocols, 2009. ICNP 2009. 17th IEEE International Conference on* (pp. 284-293). IEEE.

[15] Yu, C.M., Lu, C.S. and Kuo, S.Y., 2008. Mobile sensor network resilient against node replication attacks. In *Sensor, Mesh and Ad Hoc Communications and Networks, 2008. SECON'08. 5th Annual IEEE Communications Society Conference on* (pp. 597-599). IEEE.

[16] Ho, J.W., Wright, M. and Das, S.K., 2009. Fast detection of replica node attacks in mobile sensor networks using sequential analysis. In *INFOCOM 2009, IEEE* (pp. 1773-1781). IEEE.

[17] Yu, C.M., Lu, C.S. and Kuo, S.Y., 2009, September. Efficient and distributed detection of node replication attacks in mobile sensor networks. In *Vehicular Technology Conference Fall (VTC 2009-Fall), 2009 IEEE 70th* (pp. 1-5). IEEE.

[18] Gowtham, B. and Sharmila, S., 2012. Location Traced Hybrid Detection of Node Replication Attack in Mobile Wireless Sensor Network. *Proc. of the Special Issue of International Journal of Computer Applications (0975–8887) on Information Processing and Remote Computing–IPRC*.

[19] Deng, X.M. and Xiong, Y., 2011. A new protocol for the detection of node replication attacks in mobile wireless sensor networks. *Journal of Computer Science and Technology*, *26*(4), pp.732-743.

[20] Lou, Y., Zhang, Y. and Liu, S., 2012. Single hop detection of node clone attacks in mobile wireless sensor networks. *Procedia Engineering*, *29*, pp.2798-2803.

[21] Ho, J.W., Wright, M. and Das, S.K., 2011. Fast detection of mobile replica node attacks in wireless sensor networks using sequential hypothesis testing. *IEEE transactions on mobile computing*, *10*(6), pp.767-782.

[22] Zhu, W.T., Zhou, J., Deng, R.H. and Bao, F., 2012. Detecting node replication attacks in mobile sensor networks: theory and approaches. *Security and Communication Networks*, *5*(5), pp.496-507.

[23] Conti, M., Di Pietro, R. and Spognardi, A., 2012. Wireless sensor replica detection in mobile environments. In *International Conference on Distributed Computing and Networking* (pp. 249-264). Springer, Berlin, Heidelberg.

[24] Ratnasamy, S., Karp, B., Yin, L., Yu, F., Estrin, D., Govindan, R. and Shenker, S., 2002. GHT: a geographic hash table for data-centric storage. In *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications* (pp. 78-87). ACM.

[25] Bloom, B.H., 1970. Space/time trade-offs in hash coding with allowable errors. *Communications of the ACM*, *13*(7), pp.422-426.

[26] Bekara, C. and Laurent-Maknavicius, M., 2007. A new protocol for securing wireless sensor networks against nodes replication attacks. In *Wireless and Mobile Computing, Networking and Communications, 2007. WiMOB 2007. Third IEEE International Conference on* (pp. 59-59). IEEE.

[27] Seah, W.K., Tan, H.P. and Lee, P.W., 2010. Multipath virtual sink architecture for underwater sensor networks. In *Underwater Acoustic Sensor Networks* (pp. 78-113). Auerbach Publications.

[28] Jamshidi, M., Andalib, A. and Naseri, L., 2016. A Three-level Propagation Method of Routing Packets Specialized for Underwater Wireless Sensor Networks. *International Journal of Computer Applications*, *147*(7).

[29] J-SIM Simulator, https://sites.google.com/site/jsimofficial/, December 25, 2017.