The CSI Journal on Computer Science & Engineering

The CSI Journal on
Computer Science and Engineering
Vol. 8, No. 2 & 4 (b), 2010
Pages 35-41
Regular Paper

A Distributed Method for Preventing Wormhole Attacks in Wireless Sensor Networks

Seyed Morteza Mirbostani Mehregan Mahdavi

Department of Computer Science and Engineering, University of Guilan, Rasht, Iran

Abstract

As sensor networks become wide-spread in various applications, security issues emerge as the main concern. Wormhole attack as a severe security threat is capable of disturbing and even disabling the functionality of wireless sensor networks (WSNs) because merely authenticity and confidentiality approaches are unable to prevent it. In this paper, we propose a distributed method for detecting three kinds of wormhole attacks in WSNs using information provided by each legitimate network node, about its possible neighbors. Our mechanism is comprised of a routing and a detection algorithm which is capable of handling both static and dynamic wormholes. According to the simulation results, detecting wormholes with low false alarm rate (FAR), low failure detection rate (FDR), and estimating their locations are the main advantages of our method compared to the existing approaches.

Keywords: Sensor Networks, Security, Wormhole Attacks, Attack Detection and Prevention, Secure Routing.

1. Introduction

With the recent advances in wireless communications, wireless sensor networks (WSNs) are emerging as one of the most promising technologies. These networks consist of small nodes with limited resources of memory, computation, communication, and power with the potential of providing solutions to the wide range of applications such as military, industry, surveillance and healthcare. The sensor nodes in these networks are capable of sensing, processing, and transmitting natural phenomena interactions to other network systems for the purposes of monitoring temperature, humidity, pressure, etc. [1]. Due to the type of application and environment in which a WSN is deployed, a lot of challenging problems may be encountered. Many previous researches based on WSN assume the deploying environment as a trusted one. However, there are many mission-critical tasks with data confidentiality as a great concern. Therefore, the environment becomes distrusted which forces the developers to take security measures into account at design time. Various types of attacks such as jamming, tampering, Black holes [2], Sybil attacks [3], Wormhole attacks [4], etc., of different network layers can affect packet transmission in WSNs.

In this paper, we focus on one of the disruptive forms of attacks, called wormhole attack [4, 5], where the adversary records wireless transmission of legitimate nodes in one part of the network, tunnels them through a low-latency link, called wormhole link, to another distant part of the network, and replays the recorded messages. Wormhole nodes located at the endpoints of the link may be a sensor node compatible with other nodes of the network or a more robust one such as a laptop. Moreover, these malicious nodes' working frequency can be equal or different from others. Connecting two distant areas in the network can cause severe damage to the routing mechanisms used by network protocols. When the wormhole link is lengthy, the influence of the attack on the network performance will be intensified. However, a short wormhole link which connects two parts of the network with overlap cannot totally be harmful. Therefore, here we consider these two parts distant.

Wormhole attack is assumed to be destructive due to the fact that the adversary can perform it without compromising

cryptographic system of the network. Creating a false short route by the attacker causes network traffics to be attracted to the wormhole exceedingly; thus, the adversary can drop all passing packets, drop them selectively, or send them to another distant part of the network without changing their content.

In this paper we propose a distributed method for detecting three kinds of wormhole attacks in WSNs using information provided by each legitimate network node, about its possible neighbors which are in direct communication range. The chief advantages of this method are the ability of estimating location of wormholes in order to defend against the attack, and distributing the process between intermediate nodes in order to reduce the overhead. We also propose a potential routing mechanism, compatible with our detection method. Simulation results show minimum false alarms in our detection algorithm which makes it effective for most scenarios.

This paper is organized as follows: Section 2 discusses related work. Section 3 gives a detailed overview on wormhole attacks. Section 4 introduces a routing mechanism along with discussing the details of our wormhole detection algorithm. Section 5 evaluates the performance of the detection mechanism. Finally, Section 6 concludes the paper and discusses future work.

2. Related Work

The wormholes have destructive effects on all kinds of wireless networks. Employing static routes in networks' routing protocols is an effective defense against the attack; however, WSNs with limited resources of power and computation cannot utilize this feature. Dynamic topology with no infrastructure make WSNs susceptible to wormhole attacks. Many approaches have been proposed in order to detect and prevent such an attack in ad-hoc and sensor networks which we will discuss them briefly.

One of the most common methods for preventing wormhole attacks is 'Packet Leashes' by Hu et al. [4, 5]. They proposed embedding a secure leash of timing and/or positioning to each data packets on every single hop it travels in order to restrict their transmission distance. They proposed two kinds of packet leashes: geographical leashes which require each node to be equipped with Global Positioning System (GPS) and loosely synchronized clock (in the order of ms) and temporal leashes which rely on tightly synchronized clock (in the order of ns) on each node. By sending or receiving a packet, a network node can embed its timestamp and location to the packet. While the network nodes are synchronized, each of them has the ability to detect wormhole attack by calculating the mismatch between timestamp and location differences during the existence of an attack.

Another similar method is an end-to-end mechanism proposed in [5] assumes the knowledge of location information and loosely synchronized clocks. All these methods use lightweight hash chain to authenticate the nodes [7] because the Message Authentication Code is capable of being handled in the real time. Our method is similar to geographical leashes because of using GPS and synchronized clock, except it can detect three kinds of wormholes. The distributed scheme for reducing overhead on each node makes our method different from end-to-end mechanism [6].

Another detection method is proposed in [8] by Capkun et al. In their approach, no clock synchronization is used and every node is equipped with a special hardware which is capable of responding to one-bit challenge without any delay. It can measure signal's traveling time with an accurate clock and use the information to calculate the distance between two nodes. Using customized special hardware makes this method hard for implementation in standard WSNs and is the main difference of it from our method.

An anchor-based approach is proposed in [9, 10] which is resistant to many attacks such as wormhole attacks. They use a hop-counting technique to estimate the distance between regular nodes and anchor nodes of the network. If a wormhole link exists in the network, the distance between some regular nodes from anchor nodes will change. This method uses a threshold value to determine whether this anomaly in measured distance is because of a wormhole link or localization error. The problem is that this method is based on anchor nodes.

These nodes must be set up manually beforehand. A graph theoretic framework is proposed in [11] to defend against wormhole attacks. This method is based on special-purposed nodes, called 'guard nodes' that have unique abilities such as high transmission range with different antenna characteristics, and aware of their true position which makes the method impractical for real-world scenarios. The main advantage of our method here is that it does not need any special-purpose nodes. In [12], an algorithm is proposed that uses connectivity information to determine forbidden structures in the network connectivity graph in order to detect wormhole attacks.

Although it is designed for wireless networks, it is not yet modified to be suitable for WSNs with limited resources. In [13], a technique based on multi-dimensional scaling (MDS) is proposed which uses distance between nodes to create a layout of the network for visualization. Normally, the layout is almost flat without any distortion; however, with the existence of wormhole, network layout can be distorted. The need for a centralized computation center and not being able to be applied to the irregular shaped networks are the main differences of it from our mechanism.

Equipping mobile devices with directional antennas can increase their security levels. In [14], Hu and Evans proposed a method in which they utilize directional antennas to secure network against wormhole attacks. They assume every network node is equipped with a directional antenna that has the same orientation. Whilst receiving the packets, each node verifies the direction of receiving signals with the sending node. In case of any mismatch, communication will not be established.

A slightly different approach in designing a secure localization scheme, called SERLOC has been applied in [15]. In SERLOC, several location-aware anchor nodes, each equipped with a directional antenna, called 'beacon nodes' send out localization beacons in order to help other regular nodes of the network to relatively determine their location. These methods are assumed to be good solution for networks which rely on directional antennas; however, it cannot directly be applied to other networks.

3. The Wormhole Attacks

A typical wormhole attack includes two strategically placed adversary nodes at different ends of a network and connected with a low-latency wired or wireless link compared to default links used in the network. The link can tunnel data flow in both ways (i.e. the wormhole link is assumed to be bidirectional). At a point in the network, one of the adversary nodes records packets it overhears and forwards them through a wormhole link. At the moment when the second colluding node which is placed at another part of the network receives the packets, it replays them for its regular neighbor nodes.

In this attack, the strategy of replaying legitimate network messages of one area at another far apart area of the network makes these two origin and destination area find themselves close to each other and the nodes located in them falsely become immediate neighbors. This puts adversary in a powerful position capable of performing the followings:

- Establishing a low-latency link by the attacker makes network traffics to be attracted to the wormhole link and routing protocols choose this link as the shortest one.
- Wormhole attack can disrupt the flow of data by modifying packets if it has access to the network keys or selectively dropping them. It can record packets for later analysis as well or forward the packets at a later time.
- By periodically turning wormhole link on and off, an attacker can generate unnecessary routing activities to deplete energy of network nodes and disrupt network functions.
- The attack can still be performed even if confidentiality and authenticity approaches are provided in the network. Moreover, there is no need for adversary to have access to the cryptographic keys of the network in order to initiate the attack.

In Figure 1, a network under a typical wormhole attack is shown. Intruders A and B can be invisible to other regular nodes of the network. When node X wants to send its message to node Y, it finds the shortest route through wormhole link. Intruder A, merely records and tunnels overheard wireless data of node X to the other intruder. Intruder B replays the message for node Y. In this case, if both A and B are invisible, we have closed wormhole. There are other cases in which one or both of the endpoints are visible to other regular nodes; thus, wormhole is called half open wormhole and open wormhole [6] respectively. If an endpoint is visible, it assumes to be a hop count from regular network nodes.

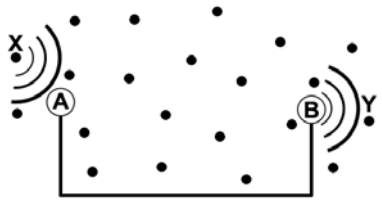

Figure 1. A network under a typical wormhole attack. Two wormhole endpoints, A and B, are connected with a low latency wire link. Node Y assumes node X to be its immediate neighbor

4. Distributed Wormhole Attack Detection Algorithm

The wormhole as an anomaly influences the network route discovery protocols. An attacker cannot modify the encrypted data transmitted in the network if authenticity is applied in the communication protocols; thus, the main impact of a wormhole is on its neighboring nodes which are determining their direct neighbors. In this section, we describe how our detection algorithm works.

4.1. Network Assumptions

In our proposed method, we consider the following assumptions to be applied to the network for our algorithm to work properly.

The adversary is considered neither to have access to the data encryption keys used in the communication protocols of the network nor to be able to crack them in any ways. The entire network's data, received or transmitted by nodes, are authenticated and only legitimate nodes with keys have granted access to use them, or to include additional data to the packets. Therefore, no intruder has the ability to modify the content of the packets in order to counterfeit them to its benefit. In other words, to achieve data integrity and authentication, a message authentication code should be used. The digital signature algorithms can be a solution to provide authenticity for the network. Although digital signature may cause more overhead on mobile nodes, using powerful processors and the feature of distribution that our algorithm possesses can obviate this issue.

The wormhole link is considered to be bidirectional. This assumption can be seen in many protocols used by real-world wireless networks. Therefore, wormhole nodes may have more features and are more powerful than regular network nodes in many aspects such as transmission range, power, and computation. Moreover, during the algorithm procedure, for the purpose of clarity, we assume wormhole nodes to be static which means the wormhole link between two specific wormhole nodes is fixed (i.e. wormhole link doesn't change its origin and destination endpoint randomly). However, later, we will explain that our proposed method is cable of handling dynamic wormholes as well.

We assume every node of the network is location-aware. Considering military applications and mission critical tasks in which these nodes are used, most of the attack scenarios take place in a hostile environment. Thus, location-awareness of nodes is essential in these scenarios. Devices such as Global Positioning System (GPS) can be used to provide accurate positioning information for a node in the network; since, GPS devices determine their position with accuracy of 5 meters or less. The accuracy of GPS directly influences the precision of wormhole attacks detection algorithm; however, state-of-the-art technologies in GPS devices have increased their accuracy. These systems also include a clock with enough accuracy to compensate the needs of our mechanism.

4.2. Algorithm Description

Wireless sensor networks are considered as constrained devices in many aspects. Their limited processing power and

energy as well as availability of memory make them incapable of inheriting current wireless network protocols without necessary modifications. Therefore, our goal is to design a wormhole attack detection mechanism, suitable for WSNs.

According to standard routing protocols, every node has a list of its available direct neighbors with one hop distance to it. If we assume every node has the same radio coverage range (R), the maximum distance of a single hop cannot exceed R. Node a generates its one hop distance neighbors set, $N_1(a)$. Suppose b as a one hop neighbor of node a ($b \in N_1(a)$). Node b holds information such as time of update, t_b, its position, P_b, and $N_1(b)$; thus, node a can generate its two hop distance neighbors set, $N_2(a)$, and three hop distance neighbors set, $N_3(a)$, using these information. The wormholes have influences on these neighbor sets. On one hand, if both origin and destination endpoints are invisible to regular nodes, a closed wormhole is placed which directly effects $N_1(a)$. On the other hand, if one of the endpoints is visible to other regular nodes, it will be counted as a hop and the effects can be seen in $N_2(a)$.

Furthermore, if both of the wormhole link endpoints are visible to the network, two extra hops will be counted which directly influences the number of N_3 (a) members. Wormhole link as an anomaly shows itself in one of these sets by adding extra fake members to them depending on the type of wormhole attack. The calculated distance of a fake neighbor $c \in N_1(a)$ to node a violates the maximum real distance that a true member $b \in N_1(a)$ and node a can have. The existence of an attack can be evaluated using $||P_a - P_c|| - \delta > R$. If the equation is satisfied, a closed wormhole attack is detected. The value δ is the maximum error of distance between two nodes. For half open wormholes, R should be changed to $R'=2R$ and for open wormholes, R should be change to $R''=3R$. There are situations in which attacker records network message and replays them in another area in the network at a later time.

Hence we consider an expiry time for the packets, T_{TL}. In order to detect wormholes which perform this act, $||t_s - t_c|| + \Delta > T_{TL}$ should be evaluated. When this equation is satisfied, a wormhole is detected. Here Δ is clock synchronization error and t_s is the time in which data packet departs source node s and t_c is the time in which data packet arrives to node c. Another situation is where we have dynamic nodes in the network. Here we define P_{per} as maximum permissive distance a network node can move ($P_{per} < R$). If $||P_{Carr} - P_{Cdep}|| - \delta > P_{per}$ is satisfied for node c, a wormhole is detected. In this equation, P_{Carr} is position of node c at the time in which data packet arrives, t_{Carr}, and P_{Cdep} is position of node c at the time in which data packet departs, t_{Cdep}.

Due to the constrained feature of sensor nodes in the network, all the processing operations such as calculations and verifications are distributed among the nodes which are located in the chosen route. In this way, total computational costs will decrease significantly. Another advantage of this method is that the algorithm will stop at the point that wormhole attack is detected in order to find another route to transmit the packets from source to destination.

4.3. Algorithm Procedure

In the network, when a message is transmitted from a source in order to get to purposed destination, it may travel through the areas in which wormhole links exist. Hence, it falls into the link assuming it is the shortest route to the destination point. Here, we don't have any wormhole attack unless a legitimate network message falls into one. In order to detect whether there is a wormhole link along the route, we designed a procedure consisting of two major sections. These sections include a routing algorithm secured against wormhole attacks along with our wormhole detection algorithm which can be utilized in networks with standard routing protocols as well.

1) Sending WSRP: When a source node intends to send a message to a destination node, it floods the neighbors with Wormhole secured routing packets (WSRP). The duty of these packets is to find a route to the destination. A WSRP holds the information of previous nodes that it passes through them. Information such as node's id, WSRP time of arrival, t_{arr}, and departure, t_{dep}, position of the node at the time WSRP arrives, P_{arr}, and departs, P_{dep}, and Packet's life time, T_{TL}, are held in a routing table for each previous nodes, WSRP passes through. these information can be gathered while WSRP is finding the route to the destination or while it is appended with the source node data packet.

a) Finding Neighbors: At the time of sending WSRP, each node n determines its 1-hop neighbor list, $N_1(n)$, and sends arrived WSRP to each member of the list. All the information of node n in addition to $N_1(n)$ are added to the routing table of WSRP.

b) Filtering Previous Nodes: If node $m \in N_1(n)$, it receives a WSRP from node n, in the next step. Like the previous step, node m detemines its $N_1(m)$. For WSRP to not go back through its path, to the previous nodes it passed before, $N_1(m) = N_1(m) - (N_1(m) \cap \text{WSRP}(m))$ should be applied to filter previous nodes. The set WSRP(m) includes information of all the previous nodes by id, that WSRP passes untill it gets to node m. This filtering makes the detemination of the shortest route possible.

After WSRP floods the network to find the destination nodes, both previous (a) and (b) steps are performed on each node in the path until one of the WSRP packets finds the destination node sooner than the others. The packet travels back the exact route direction to determine the shortest route suitable for source node's request.

2) Sending data packets: when the shortest route is specified by the source node, considering wormhole attack to be static, we are not sure there is any wormhole in between. An adversary can apply the attack dynamically, resulting in trapping data packets; therefore, the route is assumed to be unsecure by the source node. In order to determine the reliability of the route, source node appends its received WSRP to the data packet. Each node through the route is evaluated by its id, included in WSRP routing table to prevent wormhole attack to completely change the location of wormhole link endpoints and makes WSRP unable to go through that chosen route. Furthermore, our proposed algorithm in table I should be evaluated on each node through the route in order to detect any anomalies if available. In the first step of the algorithm procedure, the nodes information can be gathered by WSRP during shortest route determination or after appending WSRP to data packets. Here we consider the second order.

Table 1. Wormehole attack detection algorithm

INPUT: for node m on the route
Level 0: Checking expiry time of source node s packet
If $||t_m - t_s|| + \Delta > T_{TL}$ **then**
Return "wormhole is detected in L0"
Else
Go to Level 1
End if
Level 1: nodes with 1-hop distance
For node $n \in N_1(m)$ **do**
If $||P_n - P_m|| - \delta > R$ **or** $||P_{narr} - P_{ndep}|| - \delta > P_{per}$ **then**
Return "wormhole is detected in neighbor area of n"
Else
Go to Level 2
End if
End for
Level 2: nodes with 2-hop distance
For node $k \in N_2(m)$ **do**
If $||P_k - P_m|| - \delta > R'$ **or** $||P_{karr} - P_{kdep}|| - \delta > P_{per}$ **then**
Return "wormhole is detected in neighbor area of k"
Else
Go to Level 3
End if
End for
Level 3: nodes with 3-hop distance
For node $u \in N_3(m)$ **do**
If $||P_u - P_m|| - \delta > R''$ **or** $||P_{uarr} - P_{udep}|| - \delta > P_{per}$ **then**
Return "wormhole is detected in neighbor area of u"
End if
End for

4.4. Prominence of the Proposed Method

Another advantage of our method is the capability of detecting dynamic wormholes. In a network when the locations of wormhole links are variable, a wormhole can have more than two endpoints. It means that origin point of wormhole link records legitimate messages and tunnels them to destination point for a specific period of time. After that period of time is finished, it tunnels that very message to another destination point. Utilizing our algorithm, it detects these variations easily because all nodes information is gathered by WSRP to be evaluated. In each level of wormhole attack detection algorithm, if an attack is detected, the node's id will be saved in WSRP and by the level of algorithm in which attack is detected; distance of wormhole to that node can be estimated. The WSRP finishes its path to source node and delivers its anomaly list to be enlisted in source node's block list. Afterwards, source node will ignore that route and wait for other WSRPs to declare a route.

5. Simulation Results

In this section, we study the practical impact of distributed wormhole attack detection algorithm by implementing our mechanism in NS2 [16]. In our experiments, we used n = 400 nodes, with their positions randomized on grids. Here we assume w as grid width. Different values of grid width, w = 2, 4, 6, 8, 10, 12, 14 are used in a simulation area with 840 * 840 meters dimension. Total of 50 wormholes with various hop distances have been placed in the network randomly. These wormholes have dedicated communication channel and smaller latency than other regular network nodes. The transmission range of nodes was set to 80 meters. IEEE 802.11 MAC layer protocol and AODV routing protocol were used. UDP/CBR for network traffics was set to 2Mbps. Using grid width with the mentioned values enables simulation area to have parts with high and low density. When w = 2, the density of nodes in the area can increase significantly in comparison with w = 14.

In order to evaluate the accuracy of our mechanism as a security enhancement, we studied its false alarms ratio and error rate. False Alarm Rate, FAR, is used to determine when there is no attack but the algorithm detects a normal localization error as a wormhole. Failure Detection Rate, FDR, is used to determine when there is a wormhole but the algorithm is unable to detect it. Our mechanism can estimate the location of wormhole endpoints; therefore, in addition to those concepts, we studied estimation error rate, EER, in each simulation trial. The amount of each concept is divided by the number of trials. Our simulation results in terms of FAR, FDR, and EER are shown in Figure 2.

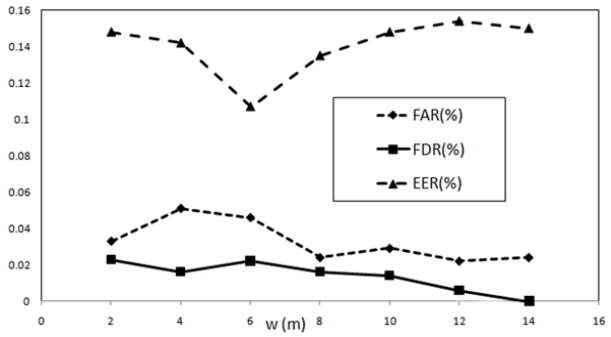

Figure 2. FAR, FDR, and EER for different grid width values

Our mechanism has a low FAR and FDR. When the value of w is less than 2 meters, the position of nodes in the area is compressed. By increasing w, fewer nodes share a certain area. When w > 12, the number of failure detections decreases and inclines to zero. The FDR curve shows that when w > 12, the algorithm detection rate increases significantly. On the other hand, while w > 8, the number of False Alarms won't increase. The FAR curve shows that for w > 8, the curve is almost horizontal. It also shows that in networks with nodes scattered densely (w < 8) in some areas, our algorithm's mistakes in detecting normal localization errors as wormhole links intensifies. The block with w < 8 is where our mechanism encounters more detection errors.

After an attack is detected, the link will be ignored for future use by the source node; however, the location of wormhole endpoint will be estimated by simply referring to the neighbor area of previous node which its position is being evaluated by the current node. In Figure 3, EER curve shows the error rate for false estimation of the wormhole location. This error occurs when consecutive wormhole links exist. In accordance with EER curve, our mechanism can estimate the origin point of wormhole links 85% of the time. Therefore the route will be ignored but in some mentioned cases, origin point of the link cannot be estimated accurately. The curve has a peak and a bottom which makes the location estimation successful 85% to 90% of the times.

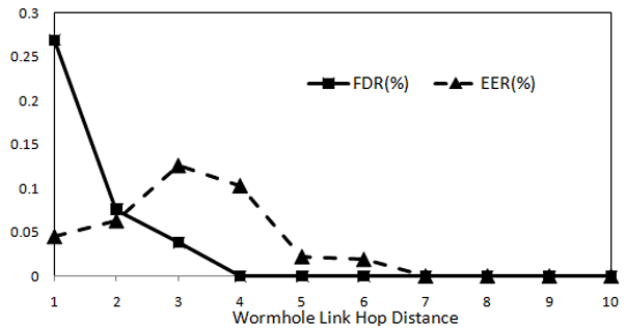

Figure 3. FDR and EER for various hop distances of Wormhole links

In Figure 3, it is shown that our mechanism is able to detect wormhole links with hop distance greater than 4. According to FDR curve in Figure 2 in oppose to Figure 3, we can tell that almost all attacks that cannot be detected by our algorithm are wormhole links with less than 4 hop distance. Moreover, EER curve descends while wormhole links with hop distance greater than 4 exist.

6. Conclusions and Future Work

It is important to notice that short wormhole links have less or no harmful effects on routing protocols of the whole network. Therefore, the extensive results of simulation show that our mechanism has the ability to detect all wormhole links with hop distance ≥ 4 which makes the whole system secured against various types of wormholes since the short wormhole links are less harmful.

The distributed feature of our method as well as having the exact location of regular network nodes makes the whole system robust and suitable for critical applications in which no node needs to perform heavy computations solely.

Currently, the prevention method in our mechanism is blocking the route with wormhole by reporting it to source node as well as estimating the neighboring area of the attack initiation. For future work, the prevention method can be improved in a way that EER decreases to less than 15%, all nodes in the wormhole neighbors can be checked, and a reporting system can be designed to forbid network nodes to communicate with malicious ones. We can improve our routing algorithm to adopt more security measures against other types of attacks as well.

References

[1] X. Chen, K. Makki, K. Yen, and N. Pissinou, "Sensor Network Security: A Survey," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 2, pp. 53-72, 2009.

[2] A. D. Wood, and J. A. Stankovic, "Denial of Service in Sensor Networks," *IEEE Trans. Computer*, vol. 35, no. 10, pp. 54–62, 2002.

[3] J. Newsome, E. Shi, D. Song, and A. Perrig, "The Sybil Attack in Sensor Networks: Analysis & Defenses," *Proc, IEEE Int'l Symp. Information Processing in Sensor Networks*, pp. 259-268, 2004.

[4] Y. Hu, A. Perrig, and D. Johnson, "Packet leashes: A Defense Against Wormhole Attacks in Wireless Ad Hoc Networks," *Proc, IEEE Int'l Conf. Computer and Communications (INFOCOM)*, pp. 1976-1986, 2003.

[5] Y. Hu, A. Perrig, and D. Johnson, "Wormhole Attacks in Wireless Networks," *IEEE Trans. Communications*, vol. 24, no. 2, pp. 370-380, 2006.

[6] W. Weichao, B. Bharat, Y. Lu, and X. Wu, "Defending Against Wormhole Attacks in Mobile Ad Hoc Networks," *Journal of Wireless Communication and Mobile Computing*, vol. 6, no. 4, pp. 483-503, 2006.

[7] A. Perrig, R. Canetti, D. Song, and D. Tygar, "Efficient and Secure Source Authentication for Multicast," *Proc, IEEE Int'l Symp. Network and Distributed System Security (NDSS),* pp. 35-46, 2001.

[8] S. Capkun, L. Buttyan, and J. Hubaux, "SECTOR: Secure Tracking of Node Encounters in Multi-hop Wireless Networks," *Proc, ACM Workshop on Security of Ad Hoc and Sensor Networks*, pp. 21-32, 2003.

[9] W. Du, L. Fang, and N. Peng, "LAD: Localization Anomaly Detection for Wireless Sensor Networks," *Journal of Parallel and Distributed Computing*, vol. 66, no. 7, pp.874-886, 2006.

[10] D. Liu, P. Ning, and W. Du, "Attack-resistant Location Estimation in Sensor Networks," *Proc, IEEE Int'l Symp. Information Processing in Sensor Networks (IPSN)*, pp. 99-106, 2005.

[11] R. Poovendran, and L. Lazos, "A Graph Theoretic Framework for Preventing the Wormhole Attack in Wireless Ad Hoc Networks," *ACM Journal of Wireless Networks (WINET)*, pp. 27-59, 2005.

[12] R. Maheshwari, J. Gao, and S. R. Das, "Detecting Wormhole Attacks in Wireless Networks using Connectivity Information," *Proc, IEEE Int'l Conf. Computer Communications (INFOCOM),* pp. 107-115, 2007.

[13] W. Wang, and B Bhargava, "Visualization of Wormholes in Sensor Networks," *Proc, ACM Workshop on Wireless Security*, pp.51-60, 2004.

[14] L. Hu, and D. Evans, "Using Directional Antennas to Prevent Wormhole Attacks," *Proc, IEEE Int'l Symp. Network and Distributed System Security (NDSS)*, pp. 1-11, 2004.

[15] L. Lazos, and R. Poovendran, "SERLOC: Secure Range-independent Localization for Wireless Sensor Networks," *Proc, ACM Workshop on Wireless Security*, pp. 21-30, 2004.

[16] "Network Simulator (NS2)," http://www.isi.edu/nsnam/ns.

Seyed Morteza Mirbostani was born in Rasht in 1985. He started his undergraduate study in "Electronic Engineering" in 2003 at the University of Guilan, Rasht and graduated in 2008. Immediately, he started his postgraduate study in "electronic Engineering, Electronics" at the University of Guilan in the area of "Security in Computer Networks" and graduated in 2011.
E-mail: m.mirbostani@gmail.com

Mehregan Mahdavi started his undergraduate study in "Software Engineering" in 1988 at Ferdowsi University of Mashhad and graduated as the first distinguished student in 1992. In 1994, he started his postgraduate study in "Software Engineering" at Amirkabir University of technology and graduated in 1997. He started his Ph. D in the area of "Web Applications" in 2001 at the University of New South Wales. After finishing his Ph. D, he worked at Macquarie University for about a year as a research fellow (academic level B). He has been working as an assistant professor in the Department of Computer Science and Engineering, University of Guilan since 2005.
E-mail: mehregan.mahdavi@gmail.com

Paper Handling Data:

Submitted: 22.06.2011
Received in revised form: 30.05.2013
Accepted: 20.06.2013
Corresponding author: Seyed Morteza Mirbostani, Department of Computer Science and Engineering, University of Guilan, Rasht, Iran.