Wireless Pers Commun (2017) 96:2585–2602
DOI 10.1007/s11277-017-4313-1

Securing Underwater Sensor Networks Against Routing Attacks

Tooska Dargahi[1] · Hamid H. S. Javadi[2] · Hosein Shafiei[3]

Published online: 25 May 2017

Abstract With the advances in technology, there has been an increasing interest both from research and industrial communities in the use of Underwater Wireless Sensor Networks (UWSNs). These networks are vulnerable to a wide class of security attacks. In fact, UWSNs are particularly more susceptible to attacks than their ground-base counterparts due to the challenges imposed by their deployment environment. This paper proposes a distributed approach to detect and mitigate routing attacks in such networks. An analytical model is provided to capture the interactions between various contributing parameters. Our simulation experiments validate the correctness and efficiency of the proposed approach.

Keywords Wireless sensor networks · Underwater wireless sensor networks · Security · Routing attacks

1 Introduction

Wireless sensor networks (WSNs) have drawn great amount of attention both from research and industrial communities during the last decade. Various aspects of such networks have been already studied to the point that these type of networks are now well-established for wide range of applications [2, 11]. These networks provide numerous advantages over their traditional wired counterparts such as self-organization, reduced deployment time and cost, adaptability, communication and processing capabilities, wireless connectivity and low energy consumption. Many industrial systems are directly

✉ Hamid H. S. Javadi
h.s.javadi@shahed.ac.ir

[1] Department of Computer Engineering, West Tehran Branch, Islamic Azad University, Tehran, Iran

[2] Department of Mathematics and Computer Science, Shahed University, Tehran, Iran

[3] Electrical and Computer Engineering Department, University of Tehran, Tehran, Iran

reliant on the underwater environments. The aforementioned key features of WSNs have made these networks attractive tools for underwater missions. As a result, Underwater WSNs (UWSNs) have been fielded in today's applications of different areas such as monitoring [1, 10], alerting and observation systems [19]. Many industry activists ranging from oil companies to environmental organizations widely deploy UWSNs to perform various operations such as off-shore oilfields monitoring or marine pollution alert systems [16]. For example, the integrated ocean observing system (IOOS) which is a partnership between various industrial companies, federal government and academia, gathers underwater data using UWSNs on oceans, coasts, and lakes [19]. The gathered data provides a comprehensive understating of the aqueous environment that can be used in many economical and health-related applications.

UWSNs usually utilize acoustic links between underwater nodes because of high energy absorption of water that decreases the propagation rate of radio waves. The acoustic links impose unique challenges to the field. First, propagation speed is much lower in water. Second, the bandwidth is very restricted and the effect of fading and the refractive properties of the sound channel are relatively high, which results in higher bit error rate compared with WSNs [7].

Security is a vital concern in UWSNs. Detecting intrusion activity and finding efficient methods to combat various kinds of attacks are of particular importance to almost every application of such networks [15, 30, 37]. Without availability, data confidentiality and integrity many real-world applications of these networks are in vain. However, UWSNs are particularly more susceptible to attacks in every level of the protocol stack than their ground-base counterparts. The low bandwidth of underwater channels, propagation delays with large variation along with high bit error rate amplify the vulnerability of UWSNs [10, 13, 42]. Outside attackers or malicious insiders may conduct various types of attacks to interfere with the normal operation of the network.

There is a wide class of attacks that can be conducted to sabotage UWSNs. However, as there is no general solution for all security threats, we focus on routing attacks [15] in these networks that can be mitigated using a local monitoring approach. An attacker may establish a *wormhole* attack in which a malicious node captures packets from one location in the network and tunnels them to another colluded node at a distant point. The second malicious node then relays the captured packets locally to the destination. The main deception of this type of attack is that it creates the illusion that the two end points of the tunnel are very close to each other, which convinces other nodes to use this route more frequently, leading to revelation of many critical security measures, e.g., they can launch a variety of attacks against the data traffic flowing on the wormhole, such as selectively dropping the data packets. Furthermore, the attacker may conduct a *sinkhole* attack where a malicious node attracts the traffic of its neighbors by pretending that it has the shortest path to the base-station. The sinkhole may also launch a variety of attacks against the data traffic, such as tampering data aggregation algorithms or interfering with clustering protocols.

In this paper we propose a distributed detection and mitigation approach to combat routing attacks in UWSNs. We show that characteristics of underwater environment pose unique challenges which make the previously proposed approaches for terrestrial sensor networks inefficient. We utilize a sliding window at each node of the network to store the ongoing traffic of their neighbors and to monitor their behavior. An analytical model is provided to capture the interactions between various contributing parameters. We provide a theoretical model for the density of node deployment to detect malicious activities. Furthermore, an upper bound for the probability of malicious nodes isolation is obtained. Finally, we provide extensive simulations to verify the obtained results.

The rest of the paper is organized as follows. We first review the most related work in Sect. 2. In Sect. 3 we present our proposed approach for detection and mitigation of routing attacks in UWSNs. Section 4 provides assumptions and describes the analytical model. Section 5 presents extensive simulation results. Finally, Sect. 6 provides some concluding remarks and outlines directions of future research.

2 Background and Related Work

Due to the unique characteristics of UWSNs, compared to terrestrial sensor networks, such networks require dedicated research in different layers of the protocol stack, such as physical and network layer [4, 10]. In what follows we highlight the main challenges in UWSNs, with particular concentration on routing attacks.

2.1 Unique Challenges of UWSNs

Acoustic links: There are various parameters such as path loss, noise, Doppler spread, multipath, and high and variable propagation delay that affect the acoustic links which restrict the available bandwidth of the acoustic channel and make it highly reliant on both range and frequency. The bandwidth of such links for short range communication is between 20 and 50 kHz with the PSK modulation and the available data transmission rate usually does not exceed 20kbps for ranges up to tens of meters [32, 39].

Node Deployment: Node placement in UWSN is a challenging issue due to transmission loss in such networks, which have led to several research studies in this context [14]. Unlike ground-base WSNs in which the topology can be optimized [9], there are limited options for the deployment of UWSN which can be categorized as follows:

- Sensor nodes are anchored to the bottom of the ocean such that they form a 2-dimensional network similar to WSNs,
- Using surface buoys so that nodes are attached by wires of various length, in order to provide the ability to observe a specific depth of the water,
- Using mobile underwater robots to carry sensor nodes,
- A floating buoy that can be inflated by a pump assists each sensor node deployed at the bottom of the ocean to reach the desired depth.

In this paper we consider the latter case for the deployment of sensor nodes.

Other Issues: Due to the lack of solar energy in deep water, nodes cannot be charged. Moreover, the battery replacement is not possible especially when nodes are deployed at the bottom of the oceans. Although, there are studies investigating energy harvesting solutions for underwater nodes using piezoelectric and microbial fuel cells [38], those approaches impose significant amount of production costs which jeopardizes one of the main goal of the sensor networks i.e., affordability. Thus, the energy in such networks is very restricted compared to land-based sensor networks. Also in such environments, fouling and corrosion can damage the nodes. These issues make the design of sensor nodes and protocols even more challenging.

2.2 Routing in UWSNs

Routing has been always a challenging issue in UWSNs. Considering the aforementioned unique challenges, researchers proposed several routing protocols for such networks. These protocols can be classified into different categories based on their main concentration in routing the data packets inside the network [4, 10]. Each of the routing protocols has taken a specific metric into account, such as energy efficiency, mobility, or reliability. However, security issues that could degrade the performance of the network have been less investigated in routing algorithms [15, 26]. Such routing attacks include wormhole, sinkhole, Sybil and hello flooding attacks [15]. While several cryptographic approaches [3, 12] have been proposed in the literature to detect external attacks such as Sybil and flooding attacks, only a few work focused on internal attacks, such as sinkhole and wormhole attacks [6]. In this paper, we concentrate on internal malicious attacks against routing protocols, i.e., wormhole and sinkhole attacks, and provide a background on the existing attacks in the following:

2.2.1 Routing Attacks Against UWSNs

Wormhole attack: In a wormhole attack a malicious node first captures a routing packet from one of its neighbors and uses a secret tunnel to send the packet to another colluded node which eventually delivers the packet to the destination. In this way, a tunnel is formed between the two colluded nodes. Even though the two ends of the tunnel may be at a longer distance compared with other routes, it can prevent the source from discovering other legitimate routes greater than two hops away from the destination and thus disrupts network functionality. The tunnel can be established using two well-known methods [27]: encapsulated channel (in-band channel) and out-of-band channel as shown in Fig. 1a, b, respectively.

Wormhole combating and mitigation strategies in ground-base sensor networks have attracted many research studies during the past years. These studies can be roughly categorized as follows:

1. Modification of a well-known routing protocol to avoid wormhole nodes during path discovery [18],
2. Deployment of an intrusion detection system (IDS) or taking advantage of extra special hardware which have been extensively studied in [17, 41, 44],

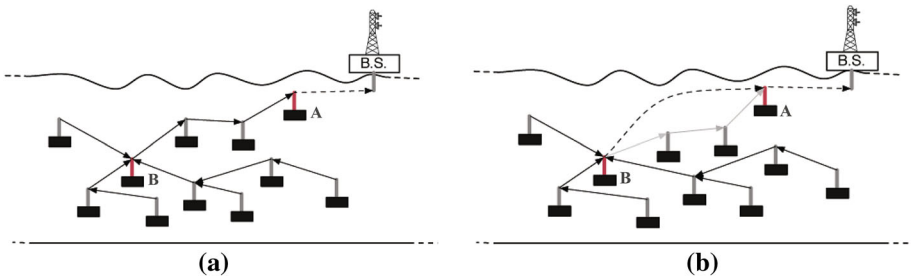

Fig. 1 Two types of wormhole attack. **a** An example to show the encapsulated attack channel. Nodes *A* and *B* establish a wormhole, **b** an example to show the out-of-band attack channel. The attackers with *red antennas* (nodes *A* and *B*) are able to communicate with each other even when they are far away from each other. (Color figure online)

3. Adopting a local monitoring strategy to reconnaissance of every neighborhood in order to detect malicious behavior which is done by each node in its own neighborhood. This method has been introduced in [21, 22].

Few research studies have addressed approaches to detect and mitigate wormhole attack in UWSNs. Wang et al. [43] propose a distributed visualization technique against wormhole attacks in UWSNs. Every node collects the distance estimations from its neighbors and reconstructs the local network topology within two hops using multi-dimensional scaling (MDS). It then uses the distortions in edge lengths and angles among neighboring nodes in the reconstructed network to locate the fake neighbor connections. The proposed approach depends on secure distance estimation for which there is no existing solution. Moreover, it conducts resource-consuming procedures of network reconstruction that deplete the priceless energy of each node. In [46] a set of wormhole-resilient secure neighbor discovery protocols has been proposed. These protocols utilize the acoustic signals directions of arrival technique to enable each node to discover its true neighbors. However, in the cases in which the straight signal is lost, due to collision, interference or shadow zone, and the signals bouncing on the surface or the sea bottom are received, there is an error in determining the correct direction of the arrival. Moreover, the proposed approaches are restricted to the neighbor discovery protocols hence they do not address dynamic routing attacks. It suffices for an attacker to compromise one or two nodes to establish such attacks. Thus, after the neighbor discovery phase and during the normal operation of the network, various attacks could be conducted using possibly few colluding compromised nodes.

Sinkhole attack In a sinkhole attack, a malicious node advertises itself as a best possible route to the base-station which deceives its neighbors to use the route more frequently. Thus, the malicious node has the opportunity to tamper with the data, damage the regular operation or even conduct many further challenges to the security of the network.

Two types of attacker may establish sinkhole attacks; a malicious insider or a resourceful outsider. In the former case, an adversary utilizes a compromised node to launch the attack. In the latter, a laptop-class adversary equipped with high performance computation and communication capabilities conducts a single-hop route from the surrounding region to the base-station which convinces the neighbors to send all the traffic through such route. Furthermore, the high quality route not only attracts the neighbors of sinkhole but also it attracts almost all the nodes that are closer to the sinkhole than the base-station (may be from several hops away) which amplifies the threat. Figure 2 depicts a sinkhole attack.

Various research studies have been focused on detection and mitigation of the sinkhole attack in terrestrial sensor networks [24, 25, 33, 35, 36]. Ngai et al. [33] propose a light-weighted algorithm to detect sinkhole attacks. In their approach the base-station collects the

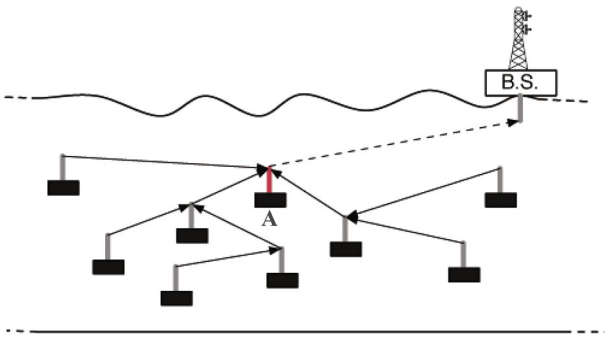

Fig. 2 An example to show the sinkhole attack. The attacker (illustrated with a *red antenna*) sends each received packet straight to the base-station. (Color figure online)

network flow information using a distributed approach, and then an efficient identification algorithm analyzes the collected data to locate the sinkhole. Their work also considers a case in which there exist multiple colluded attackers in the network. In [35], the authors utilize a dynamic trust management system to counter such attacks. In another interesting approach Krontiris et al. [24], propose an IDS system to detect such attacks. The study elaborates on a realistic scenario that uses the MintRoute protocol of TinyOS. As a result of such scenario the authors embed the appropriate rules in the proposed IDS system to successfully detect the intruder node. Shafiei et al. [36] propose two sinkhole detection approaches, centralized and distributed, considering an energy hole that forms around each sinkhole. The sinkhole attack from the perspective of the intruder also has been studied in [25]. The paper describes various methods to launch the attack. It reveals the weaknesses of the well-known routing protocols and demonstrates them in practice. The authors propose detection rules to be included in IDS designs in order to combat the attack. To the best of our knowledge, there are no research studies that either detect or mitigate such attacks in UWSNs [15].

2.3 Local Monitoring Approaches

In a local monitoring strategy, every node actively monitors the behavior of its neighbors. It detects misbehavior through inspection of the traffic going in and out of their neighbors and attempts to either diagnose or quarantine its malicious neighbors. Local monitoring has been adopted in WSN-related research studies to combat some of the serious attacks. Particularly, the authors of [20, 22, 23] have presented techniques for detection and mitigation of wormholes in order to prevent colluded nodes from selectively dropping or modifying data packets. According to their proposed scheme, each node monitors its neighbors' traffic and checks whether each of them forwards others' data packets to legitimate destinations. Upon discovery of misbehavior, monitor nodes send alert messages to their neighbors. The message divulges the dangerous nature of the monitored node to other neighbors leading to elimination of that node from future routings. Although recently researchers proposed new routing protocols having security considerations in mind, such as [6], our concentration in this work is local-monitoring-based approach.

3 Proposed Detection and Mitigation Approach

In [22] a local monitoring approach has been utilized to detect wormholes in WSN. In the proposed approach, every node keeps track of its neighbors and further its neighbors of neighbors. The activity of each neighbor is inspected to detect malicious activities. Upon detection of the malicious activity, the neighbors of suspicious node isolate it from future routings. Although the proposed approach performs well in WSNs, this approach can not be applied in UWSNs due to the challenges of such networks. In fact, the high bit error rate of UWSNs along with its propagation delay make the proposed approaches for WSNs impractical for UWSNs.

In local monitoring approaches every node monitors its neighbors and checks if they forward every received packet according to the routing before a specific time threshold i.e., a threshold beyond which the monitor node considers the packet as a dropped packet. So it is trivial that the threshold plays a key role in the detection process. Small threshold results in false-positive detection whereas large threshold fails to detect malicious activities. The main issue is that a feasible threshold can not be extracted in an underwater environment due to the large variation of the propagation delay.

We propose a collaborative detection strategy that detects and mitigates routing attacks in UWSNs. We suppose that right after deployment every node discovers its neighbors through a secure and wormhole-resilient neighbor discovery protocol using geometric relationships of pair of true neighboring transceivers calculated by signals' directions of arrival [46]. Furthermore, we consider that each node synchronizes its local clock with each of its neighbors using a pairwise synchronization approach.

Pairwise time synchronization is concerned with relative time offsets between pairs of nodes. There are two well-known approaches for pairwise clock synchronization: (1) receiver-receiver synchronization where a pair of receivers identify their clock differences using a broadcasted packet sent by a reference node, and (2) sender-receiver synchronization in which the two ends communicate to estimate their clock differences and the receiver changes its time accordingly. Another approach is global time synchronization which aims at providing a network-wide time reference. However, global and receiver-receiver synchronizations are challenging in underwater environments due to the variable propagation delays. Thus, in this paper we consider sender-receiver synchronization protocols for underwater acoustic wireless networks introduced in [8, 31]. It has been shown that the proposed synchronization schemes are feasible in an underwater environment [29].

We require a contention-free medium access control (MAC) at each of the sensor nodes. CDMA is a promising physical and MAC layer technique in underwater environments. It is robust to fading, it successfully handles multipath effect and using this technique the receivers can distinguish among signals simultaneously transmitted by multiple devices. So, we consider CDMA-based MAC layer introduced in [34].

In our approach each node maintains two sliding windows of size t i.e., W and W' for each of its neighbors at their local time. For example suppose that nodes i and j are neighbors and assume that the time window is equal to t seconds. Node i maintains W_{ij} and j maintains W_{ji}. Each node overhears the in-going traffic of its neighbors and stores the received messages in the regarding sliding window (W) for five consecutive seconds. Furthermore, each node stores the out-going traffic of its neighbors in W'. Thus, W_{ij} contains the messages addressed to node j and overheard by node i, whereas W'_{ij} contains the messages sent by node j and overheard by node i. Figure 3 shows an example of such scheme at node i at time t_0. Each shaded box shows that node i received a packet from the corresponding neighbor at that time duration. Moreover, every node such as i conducts an indicator M_{ij} (called maliciousness indicator) for each of its neighbors such as j which is incremented for each malicious activity of j that is detected by i.

Our approach is comprised of three different phases as follows:

- *Discovery phase* Each node broadcasts a *neighbor discovery* packet. Upon receiving this packet, each node replies with *neighbor pulse* packet.

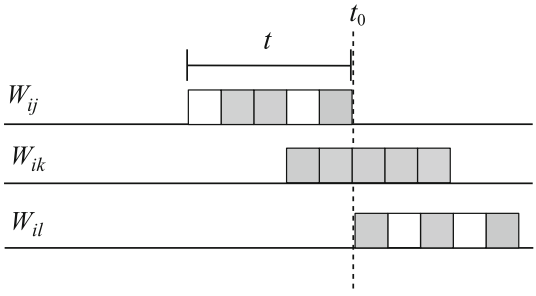

Fig. 3 An example of sliding window of node i at time t_0. It conducts a window for each of its neighbors such as j and k. Windows are not aligned due to pairwise time offsets

- *Silent monitoring phase* Each node overhears the channel and stores all of the relayed packets from its neighbors except for packets which are generated by the sender itself. Since we consider contention-free MAC layer, each node can indeed receive all of the traffic in its radio range without interference. At specific times the receiver extracts a hash-based signature from the aggregated raw data of the content of the sliding window using lightweight approaches proposed in [45], incorporates the time stamp and broadcasts a *monitoring report* packet. This packet involves the ID of the monitored neighbor, the value and the time stamp of the signature.
- *Detection phase* Periodically, each node extracts the signature of the outgoing traffic of each of its neighbors stored in W' and compares it with the corresponding monitoring report packet. If the two signatures do not match, the node increments M by one unit. The aforementioned procedures are summarized in Algorithms 1 and 2.

Algorithm 1: Discovery and Monitoring Phase

```
broadcast neighbor discovery packet;
eliminate attackers based on signals' directions of arrival;
while true do
    for all neighbors do
        while windowsize ≤ t do
            store packets of neighbor j in W_j;
        end
        if |W_j| > t then
            extracts a signature from W_j;
            add time-stamp to the signature;
            broadcast a monitoring report;
        end
    end
end
```

Thus to sum up, our proposed technique searches for discrepancies between the incoming and the outgoing traffic of a given node. A symptom that a node misbehaves is that its incoming and outgoing traffic are not equal. However, there are some remarks that must be pointed out:

- *Sink nodes*: If a node is the destination of a packet, the incoming and the outgoing traffic of that node become unequal. However, its neighbors ignore such messages from extracting signature as they can realize from the header of the packet that the node is the destination.
- *Variable fields*: Some of the fields in each packet change during the transmission toward the destination such as hop-count or other routing related fields such as TTL. In this way the incoming and the outgoing traffic are different (from the signature stand point) whereas the node is behaving correctly. To remedy this issue, each node does not consider those fields in the extraction of the signature. Moreover, the protocol only considers data packets rather than control packets since data packets are the main targets of routing attack (to selectively drop, manipulate and etc.)
- *Overhead*: The overhead of the proposed protocol is low in terms of transmitted control packets. Few local packets are transmitted for the neighbor discovery. Reports are generated only when a node detects possible malicious activity in its neighborhood. It

is worth mentioning that the complexity of our proposed algorithm depends on the number of sent/received packet by neighboring nodes of each node.

- *Hidden nodes*: Consider the scenario depicted in Fig. 4 where node *C* connects to nodes *A* and *B*. Both nodes are able to overhear the outgoing traffic from *C*. However, none of them are able to overhear the incoming traffic of node *C* from the other. Thus, these nodes raise false-positive monitoring reports. The effect of this issue on the probability of malicious activity detection is thoroughly discussed in the next section.

Algorithm 2: Detection Phase

```
while true do
    for all neighbors do
        extract signature from W';
        check the signature with the received monitoring reports;
        if signatures do not match then
            M_ij ← M_ij + 1;
        end
        if M_ij ≥ threshold then
            broadcast an alert about the malicious node;
        end
    end
    if more than α alerts received about a node then
        isolate the suspicious node;
    end
end
```

3.1 Detection of Routing Attacks

The above scheme can detect different routing-related attacks. In what follows we describe various scenarios in which our proposed model can successfully detect the attack.

3.1.1 Detection of Sinkhole Attack

As discussed, one of the serious threats of sinkhole is that it can drop or tamper the received packets. The proposed scheme can detect this type of attack by comparing the extracted out-going and in-going signature of neighbors' traffic at each node. If a malicious neighbor drops a packet the content of the regarding out-going sliding window changes so the two signatures do not match. Moreover, if the attacker tampers each of the packets the regarding signature changes significantly which makes all of the neighbors realize the

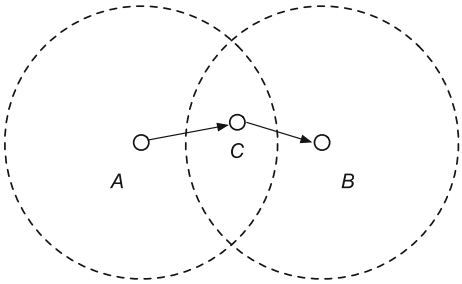

Fig. 4 Node *B* overhears the out-going traffic of node *C*. It cannot overhear the incoming traffic of node *C* from node *A*

malicious activity of the compromised neighbor. It is worthy to mention that our approach only detects those sinkholes aimed at (selective) dropping or tampering packets. Those sinkhole attacks in which the attacker simply attracts packets for traffic analysis and/or eavesdropping cannot be detected using this approach. Detection of these type of attacks is a good direction for future research studies.

3.1.2 Detection of Out-of-Band Wormhole Attack

The proposed method can also detect out-of-band wormhole attacks. The out-of-band channel connects two colluded nodes such that the neighbors of the two ends can not overhear the traffic. If one end of the channel wants to re-transmit a received packet, since its neighbors do not consider the packet in the in-going sliding window, the two signatures do not match. For example consider the scenario in Fig. 5a. *A* and *B* are two malicious nodes that try to conduct an out-of-band wormhole attack. *D* stores all of the in-going traffic of *B* except the traffic flowing in the out-of-band channel. So it does not consider those packets in the signatures. It then broadcasts the signature to its neighbors such as *E*. *E* overhears *B*'s out-going traffic and extracts the regarding signature. Node *E* realizes that the two signatures do not match which reveals the malicious activity.

3.1.3 Detection of Encapsulated Wormhole Attack

The colluded node at the end of the encapsulated wormhole path manipulates the header of each packet to convince other nodes that the route through the wormhole is feasible. Using our proposed mechanism the neighbors of the attacker realize the manipulation by checking the signatures. Figure 5b shows an example of such scheme. Nodes *A* and *B* form an encapsulated wormhole. Node *B* manipulates the header of incoming packets from node *X* to deceive other nodes that *A* and *B* are neighbors. However, the attack can be caught by *E* using the received in-going signature from *D*.

3.2 Isolation Scheme

As described above, each node increments the corresponding maliciousness indicator (M) upon detection of malicious activity from one of its neighbors. If the indicator for a neighbor reaches a system-wide predefined threshold (γ), the node generates and broadcasts an alert indicating the suspected malicious node. Each node isolates the suspicious neighbor if it receives more than α alerts in order to prevent false accusations where α is also a predefined threshold. The isolation is a lightweight process since it is conducted

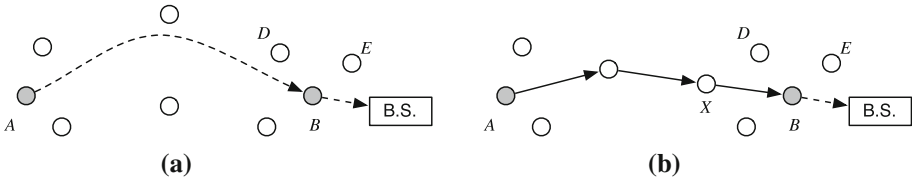

Fig. 5 Detection of two types of attacks. **a** Nodes *A* and *B* establish an out-of-band channel. Node *B* delivers the received packet to the base-station, **b** nodes *A* and *B* conduct an encapsulated channel. Node *B* delivers the received packet to the base-station

locally in the sense that the neighbors do not accept or send any packet from/to the malicious node which isolates it from the entire network.

4 Analytical Model

Assume that nodes are randomly deployed at the surface of the ocean and reach to the desired depth using an inflatable floating buoy to form a 3-dimensional uniformly distributed sensing volume.

In what follows we find the density of node deployment which guarantees that the there exists at least one node to monitor every link in the network. Figure 6 demonstrates an example to explain the problem. Figure 6a shows two neighboring nodes where spheres around each node represent their communication range. Figure 6b is a 2-dimensional projection of spheres. Circles around nodes A and B depict the communication range of each node. It can be observed that the nodes which reside in the shaded area between A and B reside in the communication range of both nodes, thus they can monitor the link between them. For example D can overhear the link between A and B whereas since C is placed far from B it cannot perform the monitoring task. Thus, the volume of the intersection of communication range around a pair of nodes must be determined in order to obtain the aforementioned density.

Assume that nodes A and B are placed within each other's radius, δ is the Euclidean distance between the two nodes, r is the transmission range of each node and $V(\delta)$ is the volume of the intersection of the spheres around the two nodes. Using elementary geometry the volume of intersection can be determined as follows:

$$V(\delta) = \frac{1}{12}\pi(4r + \delta)(2r - \delta)^2 \tag{1}$$

The right-hand side of the above equation is minimized when $r = \delta$, thus $V_{\min}(\delta) = \frac{5\pi}{12}r^3$. The expected value of $V(\delta)$ can be obtained as follows:

$$\mathrm{E}[V(\delta)] = \int_0^r \left(\frac{1}{12}\pi(4r + \delta)(2r - \delta)^2\right)\frac{3\delta^2}{r^3}d\delta = \frac{5\pi}{8}r^3 \tag{2}$$

Thus there are $\frac{5\pi/8r^3}{8\pi/3r^3 - 5\pi/8r^3}\rho$ nodes in the intersection volume where ρ is equal to the node deployment density. Thus, if $\rho > 5$ there exists, on average, one node in the volume of

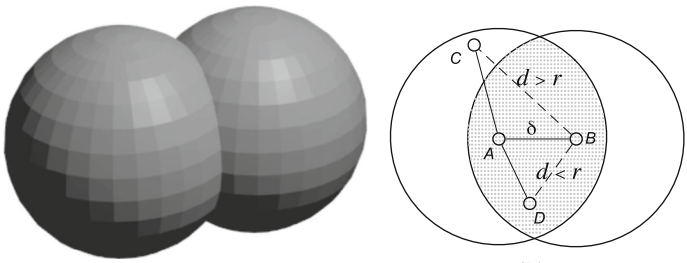

Fig. 6 The two nodes A and B reside in each other's transmission range **a** A 3D view, **b** A 2D projection

intersection of each pair of neighboring nodes. Moreover, if $\rho > \frac{8\pi/3r^3 - 5\pi/12r^3}{5\pi/12r^3} + 2 \approx 7.5$ then there exists at least one node in the aforementioned volume.

Due to the packet losses which may occur in both in-going or out-going sliding windows, signature mismatch may take place and hence false-positive detection and isolation of unmalicious nodes occur. In what follows we obtain an upper bound for the probability of false-positive isolation of a node using our proposed scheme. Assume that P_l equals to the probability of packet loss, λ be the packet generation rate according to the Poisson process and t denotes the size of the sliding window.

Let P_{FPD} be the probability of false-positive detection and X be the random variable which denotes the number of signature mismatches of node x in one of its neighbors. According to the described method, the neighbors of a node consider it as a malicious node if the number of mismatches reaches γ. Thus,

$$\Pr(X > \gamma) = 1 - \sum_{i=0}^{\gamma} \binom{\lambda t}{i} P_l{}^i (1 - P_l)^{\lambda t - i} \tag{3}$$

Applying Hoeffding's bound yields,

$$P_{\text{FPD}}(x) \leq 1 - \exp\left(-2\frac{(\lambda t P_l - \gamma)^2}{\lambda t}\right) \tag{4}$$

Let Y be the number of alerts received by node y indicating the malicious activity of one of its neighbors such as x and P_{FPI} be the probability of false-positive isolation of node x. Thus,

$$\Pr(Y(x) > \alpha) = 1 - \sum_{i=0}^{\alpha} \binom{n}{i} (P_{\text{FPD}}(x))^i (1 - P_{\text{FPD}}(x))^{n-i} \tag{5}$$

where $n = \frac{4}{3}\pi r^3 \rho$ denotes the number of neighbors. It follows that,

$$P_{\text{FPI}}(x) \leq 1 - \exp\left(-2\frac{(\frac{4}{3}\pi r^3 \rho P_{\text{FPD}}(x) - \alpha)^2}{\frac{4}{3}\pi r^3 \rho}\right) \tag{6}$$

thus a bound on the probability of false-positive isolation can be obtained using the above inequality.

5 Simulation

Our proposed approach has been implemented in Castalia simulator [5]. Castalia is a discrete event simulator for sensor networks based-on OMENT++ [40]. Numerous validation experiments have been established. However, for the sake of specific illustration, validation results are presented for a limited number of scenarios. We adopted 95 percent confidence level to make sure that, on average, the confidence interval which is calculated using t-student distribution and standard error contains the true values around 95 percent of the time.

In our simulation, we consider the following network configuration. We assumed contention-free packet transmission where nodes are scattered in a 100 m^3 volume of shallow water. We assume acoustic bandwidth of 10 kHz and intersymbol interference (ISI) of 100 symbols where system operates at 10 kilosymbols per second. We also

assumed different network sizes i.e., number of nodes are equal to $n = 100$, $n = 200$ and $n = 300$ in different scenarios. In our analysis, we considered HydroCast [28] as the routing protocol due to its efficiency compared to other routing methods [4]. It should be noted that, since our proposed method is a passive local monitoring method, its efficiency is independent from the choice of the routing algorithm.

Figures 7a–d show the effect of the sliding window size (t) on four different measures. Figure 7a depicts the probability of malicious activity detection versus the size of the sliding window. As shown in the figure the probability grows as the size increases. However, after a certain size the probability starts to decrease which is mainly because the probability of packet loss increases which in turn raises the probability of signature mismatch. This effect also can be observed in Fig. 7b where the probability of false-positive detection is depicted. As the size increases the probability grows due to the lossy channel. Figure 7c, d show the effect of the size of the sliding window on the probability of isolation and false-positive isolation, respectively.

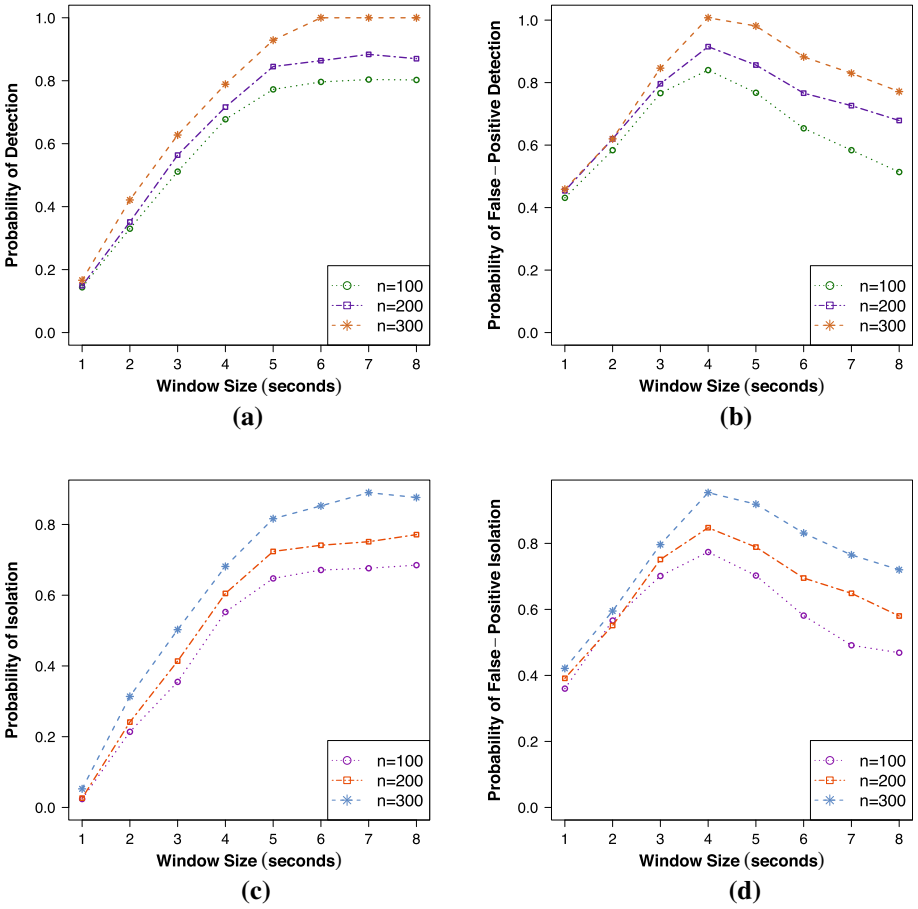

Fig. 7 The effect of window size on various measures. **a** The effect of t on the probability of malicious activity detection, **b** the effect of t on the probability of false-positive malicious activity detection, **c** the effect of t on the probability of malicious activity isolation, **d** the effect of t on the probability of false-positive malicious activity isolation

The effect of the detection threshold (γ) on the above measures is depicted in Fig. 8a, b. Figure 8a represents the probability of malicious activity detection versus the detection threshold. As shown in the figure, the probability decreases as the threshold increases due to the need for greater amount of suspicious activity in the presence of packet loss. Figure 8b reveals that by increasing the threshold the probability of false-positive detection decreases since each node requires more signature mismatches to consider its neighbor as a malicious node.

Figure 9a, b explore the relation between the probability of isolation, the probability of false-positive isolation and the isolation threshold (α). Figure 9a illustrates the probability of isolation when the threshold varies. It can be realized that the probability decreases as the threshold increases since the isolation process requires further alerts to isolate the suspicious nodes. Figure 9b shows the probability of false-positive isolation versus α. By increasing α the probability decreases because each node decides based on more received alerts which diminishes the effect of false accusations.

Figure 10a shows a snapshot of the simulation at time equal to 2000 seconds versus the total number of packets that are routed through a malicious route. We assume that there are 4 compromised nodes in the network that form encapsulated wormholes among each other. The attack started within 100 seconds after the start of the simulation. As shown in the figure, the cumulative number of packets that are routed through wormholes in the absence of our proposed method continues to increase steadily with time. However, by utilizing the proposed method the cumulative number decreases. Also note that, there exists a time interval between detection and complete isolation of the malicious nodes due to the cached routes in the intermediate nodes.

The impact of the number of compromised nodes on the the total number of packets that are routed through the malicious routes is depicted in Fig. 10b. As shown in the figure, the cumulative number of packets increases significantly as the number of compromised nodes grows due to the formation of alternative malicious routes in every neighborhood of the network. It is worth mentioning that in our experimental analysis, we did not consider energy consumption and delay. This is because the proposed method is a passive monitoring method and it does not impose delay in normal functioning of the network. The delay in detecting the malicious nodes depends on the pre-defined signature analysis time,

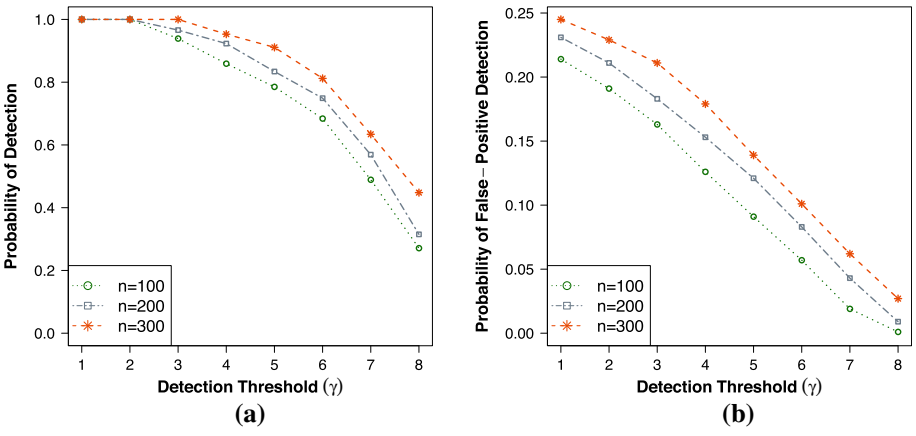

Fig. 8 The effect of the detection threshold (γ) on various measures. **a** The effect of γ on the probability of detection, **b** the effect of γ on the probability of false-positive detection

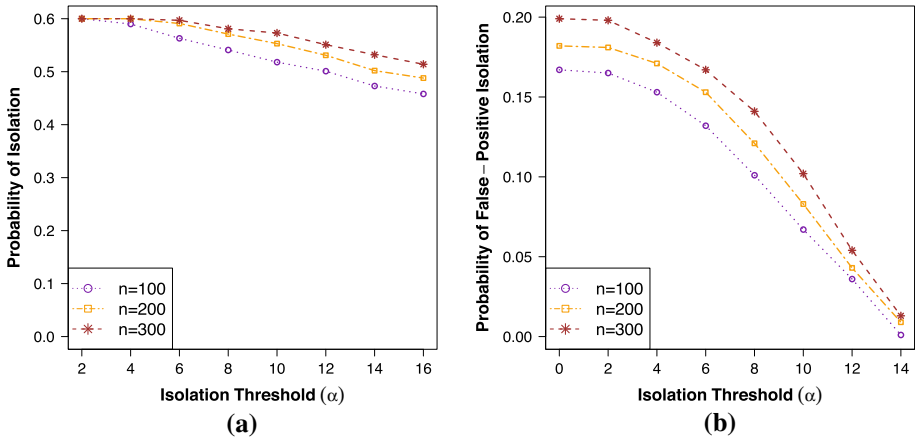

Fig. 9 The effect of the isolation threshold (α) on various measures. **a** The effect of α on the probability of isolation, **b** the effect of α on the probability of false-positive isolation

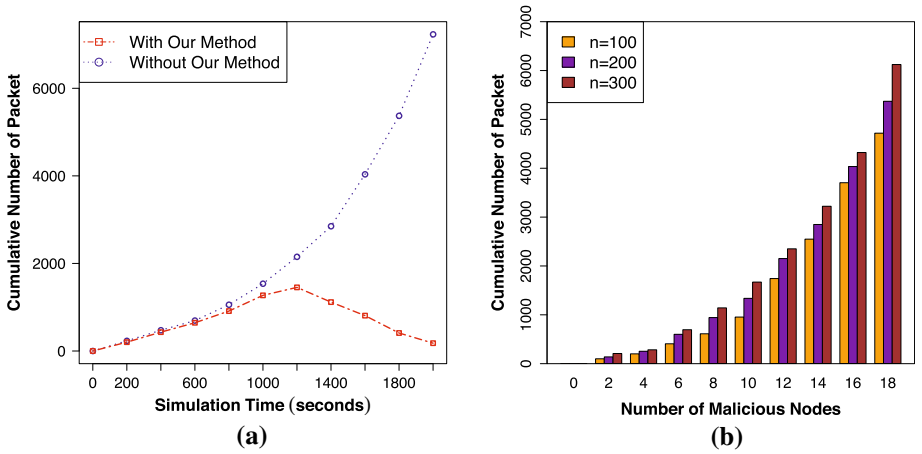

Fig. 10 Simulation results for parameters, **a** the cumulative number of packets flowing through malicious routes in 2000 s of simulation, **b** the cumulative number of packets flowing through malicious routes versus the number malicious routes

which can be modified to gain the best efficiency based on the environment condition. Moreover, the amount of consumed energy to run the proposed algorithm is negligible, since each node needs only to monitor the neighboring nodes' traffic, extract a signature and perform a comparison.

6 Conclusion

Underwater wireless sensor networks have been utilized in many industry-related applications. Security is an indispensable concern in such networks. In this paper, a distributed detection and mitigation approach to combat routing attacks in UWSNs is presented. An

analytical model is provided to capture the interactions between various contributing parameters. We carried out extensive simulations to validate our proposed method. Our next steps target to elaborate on the detection and mitigation of other security threats against UWSNs.

References

1. Akyildiz, I., Pompili, D., & Melodia, T. (2005). Underwater acoustic sensor networks: Research challenges. *Ad Hoc Networks*, *3*(3), 257–279.
2. Akyildiz, I., Su, W., Sankarasubramaniam, Y., & Cayirci, E. (2002). Wireless sensor networks: A survey. *Computer Networks*, *38*(4), 393–422.
3. Ateniese, G., Capossele, A., Gjanci, P., Petrioli, C., & Spaccini, D. (2015). Secfun: Security framework for underwater acoustic sensor networks. In: *OCEANS 2015-Genova* (pp. 1–9). IEEE
4. Ayaz, M., Baig, I., Abdullah, A., & Faye, I. (2011). A survey on routing techniques in underwater wireless sensor networks. *Journal of Network and Computer Applications*, *34*(6), 1908–1927.
5. Boulis, A. (2012). Castalia: *A simulator for wireless sensor networks and body area networks*. http://castalia.npc.nicta.com.au/. Accessed October 29, 2012.
6. Capossele, A., De Cicco, G., & Petrioli, C. (2015). R-carp: A reputation based channel aware routing protocol for underwater acoustic sensor networks. In *Proceedings of the 10th ACM international conference on underwater networks and systems (WUWNET'15)* (pp. 492–499). ACM.
7. Casari, P., & Zorzi, M. (2011). Protocol design issues in underwater acoustic networks. *Computer Communications*, *34*(17), 2013–2025.
8. Chirdchoo, N., Soh, W., & Chua, K. (2008). MU-Sync: A time synchronization protocol for underwater mobile networks. In *Proceedings of the international workshop on underwater networks (WUWNet'08)* (pp. 35–42). ACM.
9. Chiwewe, T., & Hancke, G. (2012). A distributed topology control technique for low interference and energy efficiency in wireless sensor networks. *IEEE Transactions on Industrial Informatics*, *8*(1), 11–19.
10. Climent, S., Sanchez, A., Capella, J. V., Meratnia, N., & Serrano, J. J. (2014). Underwater acoustic wireless sensor networks: Advances and future trends in physical, mac and routing layers. *Sensors*, *14*(1), 795–833.
11. Conti, M. (2015). *Secure wireless sensor networks*. Berlin: Springer.
12. Dini, G., & Lo Duca, A. (2012). A secure communication suite for underwater acoustic sensor networks. *Sensors*, *12*(11), 15133–15158.
13. Domingo, M. (2011). Securing underwater wireless communication networks. *Wireless Communications*, *18*(1), 22–28.
14. Felamban, M., Shihada, B., & Jamshaid, K. (2013). Optimal node placement in underwater wireless sensor networks. In *Proceedings of the 27th international conference on advanced information networking and applications (AINA'13)* (pp. 492–499). IEEE.
15. Han, G., Jiang, J., Sun, N., & Shu, L. (2015). Secure communication for underwater acoustic sensor networks. *IEEE Communications Magazine*, *53*(8), 54–60.
16. Heidemann, J., Ye, W., Wills, J., Syed, A., & Li, Y. (2006). Research challenges and applications for underwater sensor networking. In *Proceedings of the wireless communications and networking conference (WCNC'06)* (Vol. 1, pp. 228–235). IEEE.
17. Hu, L., & Evans, D. (2004). Using directional antennas to prevent wormhole attacks. In *Proceedings of the network and distributed system security symposium (NDSS'04)*. IEEE.
18. Hu, Y., Perrig, A., & Johnson, D. (2003). Packet leashes: A defense against wormhole attacks in wireless networks. In *Proceedings of the international conference on computer communications (INFOCOM'03)* (pp. 1976–1986). IEEE.
19. Integrated Ocean Observing System. http://www.ioos.gov. Accessed November 30, 2012.
20. Khalil, I. (2010). ELMO: Energy aware local monitoring in sensor networks. *IEEE Transactions on Dependable and Secure Computing*, *99*, 1–10.
21. Khalil, I., Awad, M., & Khreishah, A. (2012). CTAC: Control traffic tunneling attacks countermeasures in mobile wireless networks. *Computer Networks*, *56*(14), 3300–3317.
22. Khalil, I., Bagchi, S., & Shroff, N. (2007). LITEWORP: Detection and isolation of the wormhole attack in static multihop wireless networks. *Computer Networks*, *51*(13), 3750–3772.

23. Khalil, I., Bagchi, S., & Shroff, N. (2007). SLAM: Sleep-wake aware local monitoring in sensor networks. In *Proceedings of the annual IEEE/IFIP international conference on dependable systems and networks (DSN'07)* (pp. 565–574). IEEE.
24. Krontiris, I., Dimitriou, T., Giannetsos, T., & Mpasoukos, M. (2008). Intrusion detection of sinkhole attacks in wireless sensor networks. In *Algorithmic aspects of wireless sensor networks* (pp. 150–161).
25. Krontiris, I., Giannetsos, T., & Dimitriou, T. (2008). Launching a sinkhole attack in wireless sensor networks: The intruder side. In *Proceedings of IEEE international conference on wireless and mobile computing (WiMob'08)* (pp. 526–531).
26. Lal, C., Petroccia, R., Conti, M., & Alves, J. (2016). Secure underwater acoustic networks: Current and future research directions. In *2016 IEEE third underwater communications and networking conference (UComms)* (pp. 1–5). IEEE.
27. Lazos, L., Poovendran, R., Meadows, C., Syverson, P., & Chang, L. (2005). Preventing wormhole attacks on wireless ad hoc networks: A graph theoretic approach. In *Proceedings of the wireless communications and networking conference (WCNC'05)* (Vol. 2, pp. 1193–1199). IEEE.
28. Lee, U., Wang, P., Noh, Y., Vieira, L. F. M., Gerla, M., & Cui, J. H. (2010). Pressure routing for underwater sensor networks. In *INFOCOM* (pp. 1676–1684).
29. Li, Z., Guo, Z., Hong, F., & Hong, L. (2013). E2DTS: An energy efficiency distributed time synchronization algorithm for underwater acoustic mobile sensor networks. *Ad Hoc Networks*, *11*(4), 1372–1380.
30. Liu, C. X., Liu, Y., Zhang, Z. J., & Cheng, Z. Y. (2012). High energy-efficient and privacy-preserving secure data aggregation for wireless sensor networks. *International Journal of Communication Systems*, *26*(3), 380–394.
31. Liu, J., Zhou, Z., Peng, Z., & Cui, J. (2010). Mobi-sync: Efficient time synchronization for mobile underwater sensor networks. In *Proceedings of the global telecommunications conference (GLOBECOM'10)* (pp. 1–5). IEEE.
32. Melodia, T., Kulhandjian, H., Kuo, L. C., & Demirors, E. (2013). Advances in underwater acoustic networking. *Mobile Ad Hoc Networking: Cutting Edge Directions* 804–852.
33. Ngai, E., Liu, J., & Lyu, M. (2007). An efficient intruder detection algorithm against sinkhole attacks in wireless sensor networks. *Computer Communications*, *30*(11), 2353–2364.
34. Pompili, D., Melodia, T., & Akyildiz, I. (2009). A CDMA-based medium access control for underwater acoustic sensor networks. *IEEE Transactions on Wireless Communications*, *8*(4), 1899–1909.
35. Roy, S., Singh, S., Choudhury, S., & Debnath, N. (2008). Countering sinkhole and black hole attacks on sensor networks using dynamic trust management. In *Proceedings of the IEEE symposium on computers and communications (ISCC'08)* (pp. 537–542).
36. Shafiei, H., Khonsari, A., Derakhshi, H., & Mousavi, P. (2014). Detection and mitigation of sinkhole attacks in wireless sensor networks. *Journal of Computer and System Sciences*, *80*(3), 644–653.
37. Shin, S., Kwon, T., Jo, G., Park, Y., & Rhy, H. (2010). An experimental study of hierarchical intrusion detection for wireless industrial sensor networks. *IEEE Transactions on Industrial Informatics*, *6*(4), 744–757.
38. Srujana, B. S., Mathews, P., Harigovindan, V., et al. (2015). Multi-source energy harvesting system for underwater wireless sensor networks. *Procedia Computer Science*, *46*, 1041–1048.
39. Stojanovic, M. (2007). On the relationship between capacity and distance in an underwater acoustic communication channel. *ACM SIGMOBILE Mobile Computing and Communications Review*, *11*(4), 34–43.
40. Varga, A. et al. (2001). The OMNeT++ discrete event simulation system. In *Proceedings of the European simulation multiconference* (Vol. 9).
41. Venkatesan, N., Agarwal, T., Lalitha, V., & Vijay Kumar, P. (2011). Distributed intrusion detection in the presence of correlated sensor readings: Signal-space and communication-complexity view-point. *Ad Hoc Networks*, *9*(6), 1015–1027.
42. Wahid, A., Lee, S., & Kim, D. (2012). A reliable and energy-efficient routing protocol for underwater wireless sensor networks. *International Journal of Communication Systems*, *29*(1), 1–10.
43. Wang, W., Kong, J., Bhargava, B., & Gerla, M. (2008). Visualisation of wormholes in underwater sensor networks: A distributed approach. *International Journal of Security and Networks*, *3*(1), 10–23.
44. Xenakis, C., Panos, C., & Stavrakakis, I. (2011). A comparative evaluation of intrusion detection architectures for mobile ad hoc networks. *Computers & Security*, *30*(1), 63–80.
45. Yuksel, K., Kaps, J., & Sunar, B. (2004). Universal hash functions for emerging ultra-low-power networks. In *Proceedings of the communications networks and distributed systems modeling and simulation conference (CNDS'04)*.

46. Zhang, R., & Zhang, Y. (2010). Wormhole-resilient secure neighbor discovery in underwater acoustic networks. In *Proceedings of the IEEE international conference on computer communications (INFOCOM'10)* (pp. 1–9)

Tooska Dargahi Received her B.Sc. in Computer Engineering from Iran University of Science and Technology, Iran, in 2004, and her M.Sc. and Ph.D. in Computer Engineering from Islamic Azad University, Science and Research Branch, Iran, in 2008 and 2014, respectively. She is currently a lecturer in the Department of Computer Engineering, Islamic Azad University, West Tehran Branch, Tehran, Iran. Her research interests are security and privacy in wireless networks.

Hamid H. S. Javadi received his B.Sc. in Computer Science and Mathematics, in 1993, his M.Sc. in Computer Algebra, in 1995, and his Ph.D. in Computional Algebra in 2002, from Amirkabir University of Technology, Tehran, Iran. He was ranked as the top student during his education. He is currently an Associate Professor in the Department of Mathematics and Computer Science, Shahed University, Tehran, Iran. His research interests are Algorithms, Algebra, and Combinatorial designs and their applications in security and Cryptography.

Hosein Shafiei received his B.Sc. in Computer Science from Shahid-Beheshti University, Tehran, Iran, in 2006, and M.Sc. in Computer Engineering from Amirkabir University of Technology, Tehran, Iran, in 2008, and his Ph.D. in Computer Engineering from University of Tehran, Tehran, Iran, in 2015. He is currently a lecturer in the Computer Engineering Department of the University of Tehran, Tehran, Iran. His research interests are graph theory and wired and wireless networks.