Ad Hoc Networks 65 (2017) 1–12

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

An adaptive stabilizing imposter detection scheme for distributed mobile wireless sensor networks[☆]

Ebrahim A. Alrashed[*], Mehmet H. Karaata, Ali Hamdan, Badour Albahar

Department of Computer Engineering, Kuwait University, Kuwait

ARTICLE INFO

Article history:
Received 19 May 2016
Revised 11 July 2017
Accepted 17 July 2017
Available online 18 July 2017

Keywords:
Mobile wireless sensor network
Imposter detection
Stabilization
Transient faults

ABSTRACT

Mobile wireless sensor networks (MWSNs) are wireless networks of small sensors moving around a certain coverage area relaying information among themselves and conveying their readings and data to base stations. *Imposters* are malicious nodes actively engaging other legitimate nodes in the network to read or inject wrong data. MWSN are susceptible to imposter attack and therefore, the protection of MWSN from imposter nodes placed by an adversary to contaminate the sensed data is essential for the reliability of its operation. Imposters detection algorithms need to be distributed in nature and therefore they are susceptible to variety of faults that can perturb the variables for algorithm and cause a major malfunction in the operation of the algorithm and subsequently the entire network if proper recovery mechanisms are not employed. The distributed nature of imposter detection schemes for WSN and the physical environment where the sensors are deployed require some approaches such as *stabilization* to deal with faults. A stabilizing distributed algorithm can withstand *transient faults* and start in an arbitrary initial configuration by eventually entering a legitimate system configuration regardless of the current system configuration. We view a fault as a transient fault if it affects the states of the system processes but not their program. In this paper, we propose an imposter detection scheme that can effectively deal with transient faults and arbitrary initialization. In addition, the proposed algorithm effectively adapts to the introduction and the removal of sensor nodes to/from the WSN which makes the proposed algorithm appropriate for practical sensor network applications. Other faults that can occur in the network and in the nodes are beyond the scope of this work.

1. Introduction

A mobile wireless sensor network (MWSN) is a network of small sensor nodes usually deployed in an area of special interest to sense critical data like humidity, radiation, animal movements and feeding habits, etc. [1]. The sensor nodes are mobile in a confined geographic area with random direction and fixed speed. Sensor nodes can either relay information among themselves in a distributed fashion and sensed data can reach the base station through the use of appropriate WSN routing algorithms, or the use of a mobile central sink that moves around the network and collects the sensed data from the nodes. Sensor nodes are limited in both their battery power and their processor computation power, which lower their cost and make them more attractive to deploy in large number over a vast geographical area. However, these limitations restrict the amount of data that the sensor node can generate, process, store, and relay.

To further reduce the cost, weight, and volume of sensor nodes, they are typically not equipped with any physical tamper proof casings. Therefore, an adversary can physically capture a node and extract its id, authentication keys, state and program. The adversary can install the extracted information into one or more sensor nodes called *replicas* (or *imposters*) which the adversary can then deploy in the network. Once deployed, legitimate nodes consider the replicas as legitimate and continue to communicate with them since they possess the required security credentials. These malicious replica nodes then are be able to eavesdrop confidential communications, disconnect the network by ceasing to relay data from other legitimate nodes, or send falsified information leading to different attacks such as sinkhole attacks, HELLO flood attacks, etc. [2], which subsequently can hinder or disable the network operation. This form of attack is known in the literature as the *imposter* or *node replication attack* Unlike a typical WSN, where sensor nodes are stationary and the geographic location of a sensor node can be used to authenticate its identity, in a MWSN, nodes continuously

[☆] This work is supported by Kuwait University Research Administration, Grant No [EO 03/12]

[*] Corresponding author:
E-mail addresses: ibrahim.esmael@ku.edu.kw (E.A. Alrashed), mehmet.karaata@ku.edu.kw (M.H. Karaata), ali.hmdn@gmail.com (A. Hamdan), bedouralbahar@gmail.com (B. Albahar).

http://dx.doi.org/10.1016/j.adhoc.2017.07.003
1570-8705/© 2017 Elsevier B.V. All rights reserved.

move and hence using the geographic location to authenticate the identity of the sensor node is unfeasible, and hence making MWSN susceptible to *imposter attacks*. Therefore, it is vital for the proper operation of a MWSN to detect imposters through which variety of security threats can be prevented.

Various types of faults can occur in wireless sensor networks. In particular, WSNs are highly vulnerable to *transient faults* since they are deployed in unprotected natural environments without much protection against rays and other environment factors due to weight restrictions. In [3], Koushanfar et al. define transient faults as “The consequence of temporary environmental impact on otherwise correct hardware. For example, often the impact of cosmic radiation may be transient”. In [4], Finocchi et al. stated that cosmic rays and radiation can cause soft faults on the circuit memory, “Indeed, soft memory errors due to power failures, radiation and cosmic rays tend to increase with memory size and speed [20,27,33], and can therefore have a harmful effect on memory reliability”. In [5], Ziegler et al.conducted studies on the effect of cosmic rays on different 16 Mb DRAM Memory chips, and concluded that there is a correlation between memory chip cell design and SER sensitivity to cosmic rays. In [6], the authors state that as the density of circuits increase transient (soft faults) become more frequent. Since mobile sensor nodes are made with VLSI memory chips and are exposed to the harsh environment, radiation, and cosmic rays, hence, these memory chips experience transient faults just as any electronic component does. Let system processes be the set of all processes in the system, state of a process be defined by the values of its variables, and system configuration be a Cartesian product of the states of all processes. We view a fault as a transient fault if it affects the states of the system processes but not their program. Programs can be stored in ROM or safe memory elements that are not vulnerable to transient faults, therefore such an assumption is realistic.

Moreover, during the operation of the system, a WSN can have nodes joining and leaving the network due to node failures, sabotage, battery power expiry and addition of new or replacement nodes. These changes in the network can also cause the system to enter an *illegitimate configuration*, a system configuration in which the system exhibits undesirable behavior, where the imposters are not detected and/or *false positives*, legitimate nodes detected as imposters, are produced. These faults can occur at the initial stage of network deployment, during the operation of the network and when nodes join the network due to the addition of new or replacement nodes, or leave the network due to node failures, sabotage, and battery power expiry. These cases cause the system to enter an illegitimate system configuration for the distributed imposter detection algorithm.

A system is said to be stabilizing if, regardless of the system’s initial state, it is guaranteed to reach a legitimate state in finite time (or a finite number of steps) and the system configuration remains legitimate, thereafter. Due to this behavior, a stabilizing system withstands transient faults in the system variables and revert the system back to a correct and legitimate configuration where the normal operation of the system resumes and continues. To tolerate transient faults and eventually converge to a legitimate configuration where all the imposters are detected and all those detected are imposters, a stabilizing imposter detection algorithm is essential for a WSN. A system’s ability to start in an arbitrary initial state is also highly desirable in sensor networks as such systems can avoid initialization overhead and can often readily deal with introduction and removal of sensor nodes. Therefore, it is vital for critical algorithms such as the imposter detection algorithm to be stabilizing. There exist several imposter detection schemes proposed in the literature [3–12]. However, none of these can handle transient faults in the system variables that can cause the scheme to function unpredictably and potentially render the scheme ineffective in detecting imposters in the network.

In our previous works [7,8], we developed a replica/imposter detection in mobile WSN. In this work however, we focus on developing a stabilizing replica/imposter detection algorithm for a mobile WSN with soft faults. The proposed algorithm detects and quarantines imposters without any need to initialize sensor nodes and tolerates transient faults without a time overhead to quarantine all imposters in the network. Although there are limited number of stabilizing algorithms for WSNs, our work in this paper is the first stabilizing replica/imposter detection for a mobile WSN. It is a novel approach in the area. Stabilization is a strong and desired property that brings resilience to transient faults and allow the system to start from arbitrary initial state. This property is especially important is mobile WSNs to accommodate the addition new nodes to the network and the removal of nodes due to destruction or battery charge depletion making the proposed scheme appropriate for practical sensor network applications.

This paper is organized as follows. In Section 2 we survey related work on imposter detection in MWSN. In Section 3, the model of computation and assumptions are discussed, while in Section 4, the proposed algorithm is presented. The correctness and stabilization of the proposed algorithm is proved in Section 5. In Section 6 we present our experimental results, and finally, Section 7 concludes the paper.

2. Background

There are various schemes proposed for imposter detection, or node *replication attack detection*, in literature for wireless sensor networks. Initial work [9–11] focused on the study of radio-based detection which attempts to authenticate nodes, and eventually detect imposters, based on signal strength or other physical characteristic of radio communication. Parno et al. [12] proposed a scheme, called *Randomized Multicasting*, which is devised mainly for stationary WSNs. In their scheme, a number of random witness nodes are chosen, then each node in the network sends location claims to these witness nodes. If a witness node receives more than one location claim for a node *id*, a node replication attack is detected.

Abirami in [13] proposed the *Range-based Detection Method* scheme for MWSN. This distributed scheme utilizes the fact that if a node is detected somewhere, it cannot appear somewhere else. Each node estimates the distance to its neighbors and categorizes these neighbors either as close neighbors or far ones. This information is stored in a neighbor-information table, which is periodically broadcasted. Upon comparing between those received neighbor-information tables, a node replication attack can be detected.

A *Challenge and Response Based Strategy* is proposed in [14]. In this scheme each node has two arrays for storing random numbers, array S for the sent random numbers, and array R for the received random numbers. Also each node has a *Blacklist* array to store the blacklisted nodes. When two nodes meet, they check if the other node is blacklisted or not. Otherwise, they check whether they have sent and received stored random numbers. If the nodes have not encountered aprior, each node generates a random number within a range, hashes this random number using a hash function and sends this hash value to the other node. The random number is then stored in the nodes S array, while the received hash value will be stored in the R array. When the two nodes meet again, they exchange the previously received hash value and each node hashes value the random number it previously stored in its S array. If the two values match, which means the nodes are genuine, new random values are generated and new hash values are exchanged. Otherwise, if the values do not match, which means the node is a replica, the node is added to the blacklist.