Computer Networks 108 (2016) 210–222

Contents lists available at ScienceDirect

Computer Networks

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

Imposter detection for replication attacks in mobile sensor networks

Tassos Dimitriou[a], Ebrahim A. Alrashed[b,*], Mehmet Hakan Karaata[b], Ali Hamdan[b]

[a] Computer Technology Institute, Greece
[b] Department of Computer Engineering, Kuwait University, Kuwait

ARTICLE INFO

Article history:
Received 17 December 2015
Revised 13 July 2016
Accepted 22 August 2016
Available online 30 August 2016

Keywords:
Mobile sensor networks
Imposter detection
Node replication attack
Wireless network security
Node revocation
Soundness & completeness

ABSTRACT

In a node replication attack, an adversary creates replicas of captured sensor nodes in an attempt to control information that is reaching the base station or, more generally, compromise the functionality of the network. In this work, we develop fully distributed and completely decentralized schemes to detect and evict multiple imposters in mobile wireless sensor networks (MWSNs). The proposed schemes not only quarantines these malicious nodes but also withstand collusion against collaborating imposters trying to blacklist legitimate nodes of the network. Hence the completeness and soundness of the protocols is guaranteed. Our protocols are coupled with extensive mathematical and experimental results, proving the viability of our proposals, thus making them fit for realistic mobile sensor network deployments.

1. Introduction

A *Wireless Sensor Network* (WSN) is a wireless network of small sensors deployed in a specific area to sense various aspects of the environment. A *Mobile Wireless Sensor Network* (MWSN) is a special type of WSN in which sensors are mobile. MWSNs convey the sensed data to base stations or sink nodes, which can be either static or mobile, thus trying to cope with rapid topology changes that make sensing problematic in ordinary sensor networks. As a result, they extend the number of applications for which static (WSNs) are used [1]. Sensors can be attached to people for health and physiological monitoring, to animals in order to track their movements and their feeding habits, or to unmanned aerial vehicles (UAVs) for surveillance, environmental mapping and control [2,3].

In a typical WSN, where the sensor nodes are stationary, the sink or other nodes can ascertain the authenticity of a sensor node by tying its identity to its *claimed* geographic location [4]; through the help of witness nodes, location claims coming from conflicting areas in the network indicate the existence of a replication attack.

In a MWSN, however, the constant movement of nodes makes location-based detection a nearly impossible task. As a result, an adversary can assume the identity of a legitimate node and use it to communicate with the rest of the network. As sensor nodes are not tamper-resistant devices [5], the adversary can create *replicas* of nodes after compromising a node and replicating its cryptographic or other material. We refer to such replicas as *imposters* if they use the identity of existing sensor nodes to communicate with the sink or other nodes of the network.

Since the credentials of replicated nodes do not differ from those of legitimate ones, there is no easy way to distinguish between the two, thus making imposter detection a very difficult process. This type of attack, which is known as *node replication attack* in the literature, has important repercussions in wireless sensor networks security: by assuming a false identity, an imposter can send misleading information, replay old packets which could bias aggregation results or enable other types of attacks in the network, like selective forwarding, sinkhole attacks, etc. [6–8].

Contributions. In this work, we address the problem of node replication attacks by proposing a number of lightweight, *decentralized* protocols to detect imposters in MWSNs. Contrary to prior work that focuses only on imposters that can replicate only a *single* node ID, our schemes work even in those cases where imposters have assumed the identities of *different* nodes. This case is more challenging as it poses another problem: imposters can *frame* legitimate nodes, thus resulting in their dismissal from their network.

In this work, we show not only how to detect these powerful imposters but also maintain the number of false-positives (evictions of legitimate nodes) to a bare minimum. Eventually, when a sensor node is identified to be an imposter, it is prevented from communicating with other nodes in the network by means of an effective quarantining mechanism. Hence our protocols are both sound and complete. Finally, through extensive simulations, we

* Corresponding author.
E-mail address: dr_ebrahim@mac.com (E.A. Alrashed).

http://dx.doi.org/10.1016/j.comnet.2016.08.019
1389-1286/

demonstrate the practicality and viability of our approach in detecting and mitigating the node replication attack.

Organization. The rest of the paper is organized as follows. In Section 2, we review related work on imposter identification in wireless sensor networks. In Section 3, the threat model and assumptions are discussed, while in Section 4, a number of schemes are presented and analyzed. Experimental results are discussed and evaluated in Section 5. Finally, Section 6 concludes the paper.

2. Related work

In this section we review prior work on imposter identification which also comes under the name of *node replication* detection. 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.

Network-based detection typically relies on the use of a *claimer-reporter-witness* framework, originally proposed by Parno et al. in [4]. These techniques, which mostly work for *stationary* networks, store information about the location of a sensor node in one or more witnesses in the network which can then detect and report replicas once they receive more than one location claim from nodes interacting with a particular sensor node. A more detailed review of works addressing the problem in stationary sensor networks can be found in [12].

For mobile sensor networks, one line of research involves the study of properties possessed by the network as a whole in order to trigger the existence of imposters [13,14]. In [13], a centralized scheme is proposed where a base station is used to calculate the speed of nodes based on location information received by neighbors of that node. If the speed exceeds a predefined threshold, an alarm is raised and the replica is detected. In a similar manner, the basic idea in the work of [14] is to differentiate between the time a node u encounters another node v when there are no replicas in the network (during initial deployment) as opposed to the case when replicas exist. The authors come up with a scheme based on the difference of the distributions of these two cases, hence replica identification is possible with certain probability. These approaches, however, rely on the existence of an all-powerful base station that maintains a complete picture of the network, thus requiring heavy localization and synchronization primitives by the nodes.

A different line of research involves the use of *tokens*, to authenticate the genuineness of a mobile node [15–17]. Once two sensor nodes encounter each other, they exchange random, unpredictable numbers. If the two nodes meet again, both of them request the other for the random number they exchanged at earlier time. If the other cannot reply or replies with a wrong number, the node is treated as an imposter and an alarm is raised. In this work we build upon this technique as it uses lighter cryptography and leads to simpler protocols. Our work, however, differs from these past results in three important aspects.

- First, our scheme can effectively neutralize *multiple* imposters that are copying *different* legitimate IDs. In contrast, past works ([13–17]) only consider imposters that are copies of a *single* node which makes detection easier; once the replicated ID is found, all imposters can be evicted from the network.
- Second, we develop protocols that are completely *decentralized* and nodes themselves, without the need of a powerful base station ([13–16]) or mobile sinks [17], succeed in quarantining these imposters.
- Finally, as in this more challenging case imposters can collaborate to blacklist legitimate nodes, we show how to avoid false positives by coming up with appropriate mitigation strategies.

3. Threat model and assumptions

We consider a mobile wireless sensor network (MWSN) consisting of N mobile sensor nodes deployed in a certain area of interest. Sensor nodes route their sensed data to a stationary base station or to a mobile sink that acts as a gateway to some external network using appropriate routing protocols ([25–27]). We assume all network nodes have limited resources and they are similar in terms of energy, memory and computational capabilities. In particular, sensor nodes have limited wireless communication radius and only the base station can broadcast messages to all nodes, if necessary. Thus, typically, nodes have a small number of neighbors which can utilize in forwarding data or exchange tokens that can be used in detecting imposters. They also move randomly within the specified coverage area but not necessarily with the same speed. As a result, the time and the location of node encounters, as well as the IDs of the meeting nodes are generally unpredictable.

We define an *imposter* to be a malicious node which uses the identity of a legitimate node to communicate with other nodes in the network. In our model, the imposter has obtained the cryptographic credentials of a genuine node u after compromising that node. It then uses these keys to communicate with the sink or other nodes, using u's identity and claiming to be node u. Messages received by either u or its imposter are indistinguishable, so it is not possible to differentiate between the two by virtue of messages sent. The only way that the presence of an imposter can be detected is if a third node encounters both u and its replica, one after the other, and one of them replies with the wrong nonce.

We assume the base station is well protected, hence the adversary cannot generate new IDs by obtaining the corresponding base station credentials. Following [4,16], this is possible by assuming the existence of an ID-based cryptography scheme. Thus a node u is deployed with a private key K_u^{-1} and any other node can derive u's public key K_u by applying an appropriate function F to u's ID, i.e. $K_u = F(u)$. Such dynamic generation of public keys is a more preferable solution over a traditional public key infrastructure (PKI) system which would require every node to prove the validity of its public key by transmitting an appropriate certificate signed by the base station; the other alternative which requires every node to be preloaded with *all* nodes'public keys is clearly an impractical task for large scale sensor networks.

While key management schemes in WSNs are mainly based on symmetric cryptography, recent works [18–21] have demonstrated the feasibility of public key cryptography on resource-constrained sensor nodes. TinyPK [18] utilizes the RSA cryptosystem to provide authentication and key exchange between an external party and a sensor network. The use of Elliptic Curve Cryptography (ECC) [19] constitutes a much better alternative to traditional public key (PK) cryptography algorithms as it is possible to generate short 160-bit keys in resource-constrained devices. Identity-based solutions based on pairings have also been implemented [20,21] for sensor nodes based on 8-bit microprocessors (e.g., MICA 2 and MICAz motes or the Tmote Sky sensors), showing that pairing-based cryptography is indeed a practical alternative for sensor networks.

In the protocols of the next section, we follow the ID-based approach to *authentication* that can be achieved by tying the identity of a node to its public key so that any other node can verify the authenticity of a signed message by deriving the public key of the node from its unique ID. Since the only requirement in our protocols is the ability to generate and verify signatures, Shamir's original Identity-based signature scheme [22] can also be used as we don't need the full set of capabilities provided by pairings. This approach can lead to even lighter implementations when combined with ECC as discussed above. In Section 4.1.1, however, we also suggest a symmetric cryptography alternative to signing that requires less computation but more communication overhead.