sensors

MDPI

Article

Localization-Free Detection of Replica Node Attacks in Wireless Sensor Networks Using Similarity Estimation with Group Deployment Knowledge

Chao Ding [1], Lijun Yang [2,*] and Meng Wu [3,*]

1 College of Computer Science, Nanjing University of Posts and Telecommunications; Nanjing 210003, China; dingchao_129@163.com
2 College of Internet of Things, Nanjing University of Posts and Telecommunications; Nanjing 210003, China
3 Key Lab of "Broadband Wireless Communication and Sensor Network Technology" of Ministry of Education, Nanjing University of Posts and Telecommunications; Nanjing 210003, China
* Correspondence: yanglijun@njupt.edu.cn (L.Y.); wum@njupt.edu.cn (M.W.); Tel.: +86-25-8588-2439 (M.W.)

Academic Editor: Kemal Akkaya
Received: 25 November 2016; Accepted: 9 January 2017; Published: 15 January 2017

Abstract: Due to the unattended nature and poor security guarantee of the wireless sensor networks (WSNs), adversaries can easily make replicas of compromised nodes, and place them throughout the network to launch various types of attacks. Such an attack is dangerous because it enables the adversaries to control large numbers of nodes and extend the damage of attacks to most of the network with quite limited cost. To stop the node replica attack, we propose a location similarity-based detection scheme using deployment knowledge. Compared with prior solutions, our scheme provides extra functionalities that prevent replicas from generating false location claims without deploying resource-consuming localization techniques on the resource-constraint sensor nodes. We evaluate the security performance of our proposal under different attack strategies through heuristic analysis, and show that our scheme achieves secure and robust replica detection by increasing the cost of node replication. Additionally, we evaluate the impact of network environment on the proposed scheme through theoretic analysis and simulation experiments, and indicate that our scheme achieves effectiveness and efficiency with substantially lower communication, computational, and storage overhead than prior works under different situations and attack strategies.

Keywords: security in wireless sensor networks; replica node detection; location similarity; deployment knowledge

1. Introduction

Low-power wireless sensor networks (WSNs) are known to be capable of rapid deployment in large geographical area in a self-organized manner, which makes them particularly suitable for real-time large-scale data collection and event monitoring for mission-critical applications, such as border monitoring, target tracing, and in-network aggregation. In such applications, the sensors are deployed in a hostile environment with potential security threats. However due to the constraints of network scale and fabrication cost, the sensor nodes are usually exploited by adversaries with poor security guarantees. Meanwhile, since in most cases WSNs are remotely administrated by the network operator, the sensor nodes are often deployed in an unattended manner. Thus, compared with traditional wired and wireless networks, WSNs are much more vulnerable to a variety of attacks from the inside and outside of the network, such as eavesdrop, forge, and node compromise. In recent years a large amount of research efforts [1–5] focus on the security issues of WSNs and their corresponding fields.

Among such attacks, the node replica attacks [6] may be particularly dangerous to WSNs because it is difficult for the security mechanisms to identify the replica nodes with a reasonable time and resource consumption. However, once an adversary possesses a small number of compromised nodes, he can easily generate a large number of replicas which share the keying materials and IDs with the original ones, spreading the replicas throughout the network. The sensor nodes, which pass the verification of the network security protocols, are able to create pairwise shared keys with other nodes and the basestation (BS) with legal keying materials and IDs and, thus, capable of encrypting, decrypting, and authenticating their own communications on demand, as if they were the original compromised ones. To our best knowledge, the cost of generating replicas for the adversary is much lower than that of compromising equal quantities of sensor nodes, which makes it extremely economical to launch node replica attacks. By injecting a large number of replicas into the target network, the adversary manages to continuously undermine the network without being detected. For example, he can overhear the traffic pass through his deployed replicas, and inject false data to disturb the data collection. Alternatively, he could adopt more aggressive strategies which undermine the network protocols such as clustering and in-network aggregation, thereby incurs continuous harm to the network operations. To some extents, node replica attack are far more dangerous than node compromise attacks, as the time and effort spent on the node replication are much less. However, compared with other security threats, like eavesdropping, forgery, denial of service, and node compromise, the node replica attack receives much less attention. We, thus, believe that it is necessary to develop distributed lightweight countermeasures to address the threat of node replicas in an early stage of network.

A straightforward solution to the node replica problem is to equip the tamper-proof hardware on each node in the network against illegal loading of security materials and malicious program rewriting. However such a solution is much too expensive for most sensor network applications. Additionally, although tamper-proof hardware has the adversaries spending more time and effort on node compromise, it may still be possible to bypass tamper-resistance for a small number of nodes in reasonable amounts of time. Another class of solutions [6–11] identifies the replica nodes based on the location claims reported by the sensor nodes themselves. These solutions deduce the location anomalies based on the conflicts existing in the location claims. However, these location-claim-based schemes are vulnerable to the falsified location claims generated by the replicas. The replica nodes manage to elude the detection by reporting the same location as the original compromised nodes to BS.

To address the limitation of the prior works, we propose a location-free scheme to detect node replica attacks in sensor networks using group deployment knowledge. Our scheme adapts the location claim idea presented in [6]. The basic idea behind our proposal is that it is reasonable to treat a node as a replica when its claimed location is far away from its true location. However, the exact physical positions of sensor nodes are difficult to obtain since accurate localization is not practical in sensor networks due to high cost and various types of environmental uncertainties. Thus, instead of deploying resource-consuming localization techniques on resource-constrained sensor nodes, we design a novel metric named location similarity to quantify the deviation between true and claimed location using locality sensitive hashing (LSH) based similarity estimation techniques [12]. Such a metric only requires the sensor nodes to collect their neighbors' IDs as well as receiver signal strength indicator (RSSI) [13] of the top four nearest deployment points. In addition, our scheme works on the basis of the assumption that sensor nodes are deployed in groups and the nodes in each group are placed around the predefined location named deployment point. In such a case, our work allows most nodes within a group to communicate without generating any location claims. Due to the aforementioned advantages, our scheme achieves an effective and efficient replica detection with low communication, computation, and storage overhead.

Furthermore, we validate the security performance of the proposed scheme through heuristic analysis under different attack strategies and demonstrate that our scheme provides robust replica detection even under the condition that there are considerable nodes compromised by adversaries. In addition, we also evaluate the effectiveness and efficiency of the proposed scheme through both

theoretic analysis and simulation experiments. The results show that our approach achieves effective and efficient replica detection while incurring significantly low overhead.

The rest of paper is organized as follows: in Section 2, we propose the preliminaries of this paper, including the network assumptions, attacker models and the group based random deployment strategy. In Section 3, we present the mathematical definition of the node replica attack and the location similarity, In Section 4, we describe the details of the proposed LR2ND scheme. In Sections 5 and 6, we present the security analysis and the performance evaluation, respectively. Finally in Section 7, we summarize our work.

2. Related Works

The discussion about the replica node attack in WSNs was firstly found in Parrno et al's. work [6], in which randomized multicast and line-selected multicast schemes are proposed to address such problems. In the Randomized Multicast scheme, signed location claims are sent to randomly chosen witness nodes for the validation of consistency. A node will be considered to be replicated if two conflicting location claims about this node are found. The improved line-selected scheme effectively reduces the communication overhead incurred by location claim transmission of the randomized multicast scheme by having every claim-relaying node participate in the replica detection and revocation process. However, these multicast-based schemes [6] and their variants [7], have to periodically multicast the location claims over the whole lifetime of the network, resulting in very large communication and computation overhead. In our scheme, replica detection works on the basis of group deployment knowledge. Only the nodes placed outside its home group are required to send location claims, which achieves significant higher resource efficiency than [6] does.

Based on the line-selected multicast scheme of [6], Conti et al. [8] proposed a randomized improved scheme RED to enhance the performance in terms of replica probability, storage and computation overheads. However, compared with [6], the communication resource efficiency of RED scheme has no significant improvement. Furthermore, the protocols require repeated claims over time, which means that the communication overhead of such scheme needs to be multiplied by the number of runs during the entire network lifetime. In contrast, our proposed scheme achieves higher communication resource efficiency than RED by only requiring location claims when new arrivals are placed in the network.

Abinaya et al. [9] proposed the improved scheme X-RED on the basis of RED [8]. The main design principle of X-RED is similar to RED, but the witness is selected dynamically using a randomized hash function. The approach of randomized witness selection can evenly distribute overhead among nodes, which effectively prevent single point of failure. However, the drawback of very large overhead caused by periodic claim examination is still not improved in X-RED.

Zhu et al. [7] proposed a replica detection scheme based on grid cell topology, which detects replicas by multicasting location claim to single cell or multiple cells. The chief advantage is that it enhances the detection accuracy of schemes proposed in [6]. However, its communication overhead has no significant improvement compared with [8]. Our scheme can achieve similar detection accuracy with much lower communication overhead.

Choi et al. [10] proposed a localized replica detection scheme for sensor networks based on regionalized deployments. In this work, the network is viewed as a subsets of non-overlapping subregions, each of which has an exclusive subset. If the intersection of these subsets is not empty, it is reasonable to imply that replicas are included in such subsets. However, the adversary can bypass the detection of Choi's et al. work by some specific replica placing methods. Our scheme can effectively address this problem.

Ho et al. [11] proposed replica detection schemes based on group deployment knowledge which adapts the location claim idea from [6]. In these schemes, the sensor nodes inside their home zone can transmit their ordinary messages without any extra validation of security protocols, whereas the sensor nodes outside their home zone are not allowed to transmit messages unless they are

authenticated by location claims. Compared with prior works, this scheme eliminates most of the communication, computational, and storage overheads, since it only requires part of nodes to generate and send their location claims. However, the scheme is built on the assumption that every node knows its own position by some kind of localization protocols, and its detection performance depends on the accuracy of localization. Actually, it is very expensive to deploy localization schemes on the resource-limited sensor nodes, and accurate localization is hard to get since there are various uncertainties in WSNs. In addition, the usage of localization may introduce more potential threats due to the security vulnerabilities of existing localization schemes [14]. Our scheme achieves secure, effective, efficient replica detection with similar low communication, computational, and storage overheads in a localization-free manner.

Khedim et al. [15] propose a mobile assistant clone detection (MCD) protocol aiming at mitigating the dependence on the GPS and beacon nodes. MCD is a hybrid protocol which uses patrol robots and honeypots for the node replication detection in static sensor networks to enhance the detection performance. However this scheme requires extra expensive hardware which substantially increases the deployment costs. Meanwhile the scheme requires periodic examination on all of the nodes in the network over the entire network lifetime. In contrast, the proposed scheme only starts the node replication detection on demand and the detection is limited in a local region. Chen et al. [16] propose an intrusion detection algorithm to address the problem of replication attacks in the clustered wireless sensor networks based on a novel clustering protocol NI-LEACH. The main advantage is that the scheme is configurable according to the performance requirements by choosing appropriate encoder functions. However, this scheme requires the witness nodes to be randomly selected from network in order to undertake large amount of computation intensive and energy consuming tasks. The sensor nodes which act as witnesses will rapidly run out of energy.

Ho et al. [17] propose a node replication detection scheme which is composed of quorum-based multicast (QBM) and star-shape line-selected multicast (SLSM), which can deterministically detect the replicas. However this scheme still requires repeated claim checking, which results in large amount of communication.

Additionally, a Sybil attack [18] can be regarded as an extended form of node replication attacks, there are also some typical schemes. Pecori [19] proposes a security protocol which resists Sybil attacks through the use of a combined trust-based algorithm exploiting reputation techniques. Compared to similar methods, such a trust-based algorithm shows promising results in thwarting a Sybil attack in a Kademlia network.

3. Preliminaries

In this section, we first present the underlying assumptions and sensor deployment strategy, and then describe the detailed attack model of our scheme.

3.1. Network Assumptions

We assume that the proposed scheme works in a typical two-dimension static sensor networks in which every node holds their own position immediately after deployment. All direct communications links in the network are bidirectional. This assumption is common in the current generation of sensor networks. We assume that the sensor nodes in the network can be divided into two categories from the duty perceptive: the ordinary nodes and BS, where the ordinary nodes generate ordinary data and location claim related messages, and send them to BS via single- or multi-hop transmission, whereas BS collects location claims for further analysis of similarity estimation and the final decision on the suspicious replicas. We also assume that the sensor network are deployed in an open space which allows us to perform distance measurement using the Receiver signal strength indication (RSSI) extracted from media access control layer protocols. This assumption is common in current generation of WSNs. Additionally, BS may take further security measurements such as software attestation and node revocation if necessary. We also assume that BS is a trusted entity. This is a common and

reasonable assumption since if BS is compromised, the sensor networks suffers a risk of a single-point of failure, which means that the entire mission of the sensor network can be easily undermined.

Furthermore, we assume that each node in the network has and only has a unique ID so that BS is able to correctly parse the source of location claims. Moreover, we assume that the message authentication code (MAC) is adopted to filter the unauthorized modification on the network traffic and verify the message source. In this work we adopt the MAC algorithm proposed in [20], whose main advantage is that the authentication tag of such a MAC algorithm can be aggregated, resulting in a substantial reduction of the location claim size.

3.2. Sensor Deployment Strategies

We adopt a group-based random deployment strategy in our scheme. In this work, we assume that the whole network is divided into grids as shown in Figure 1, and define the grid intersection points as predefined deployment points. Before deployment, we firstly place sensor nodes exactly at these deployment points as beacons. The rest of sensor nodes are allocated into groups and programed with the corresponding group information, such as Group ID. We assume that the number of nodes in each group is even. Then, during the deployment, the nodes in the same group are randomly placed around corresponding deployment point. We assume that the coordination of the sensor nodes within one single group follows the two-dimensional Gaussian distribution. This is reasonable and practical deployment strategy since in most sensor network applications sensor nodes are spread over the target region in a randomly scatter manner such as dropped from airplane or spread by hand. This assumption is supported by the fact that the group deployment strategy has been used for various applications in sensor networks, such as key distribution [21,22] and public key authentication [23].

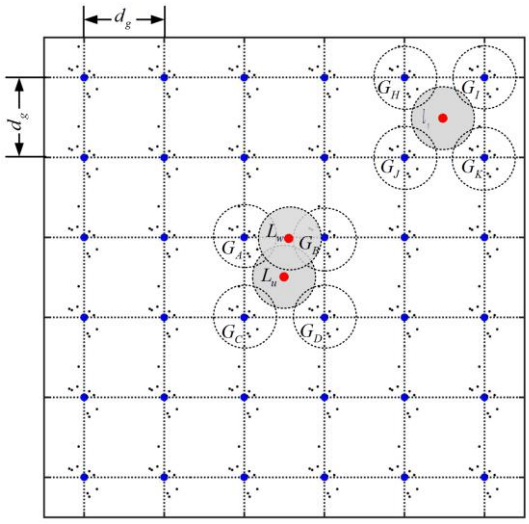

Figure 1. The group-based random deployment strategy for WSNs.

The detailed deployment rules and basic assumptions are described in the form as follows: (1) assuming that there are m predefined deployment points that are placed at the grid intersection points, denoted by $g_1, g_2, \ldots, g_m$; (2) assuming that there are a total of M nodes in the network and these nodes are divided into m groups, denoted by $G_1, G_2, \ldots, G_m$, namely M/m nodes in each group; and (3) the nodes within the same group are *i.i.d.*, following a two-dimensional Gaussian joint distribution. For example, for the node k ($k = 1, 2, \ldots, M/m$) in group G_i ($i = 1, 2, \ldots, m$), the probability density function of node k's coordination (x_k, y_k) is represented as:

$$f(x, y|k \in G_i) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{2\sigma^2}\right), \quad (1)$$

where (x_i, y_i) is the coordination of predefined deployment point g_i, whereas the coordination of the sensor nodes from different groups are independent from each other. Figure 2 illustrates the distribution of the probability density distribution over the entire deployment region.

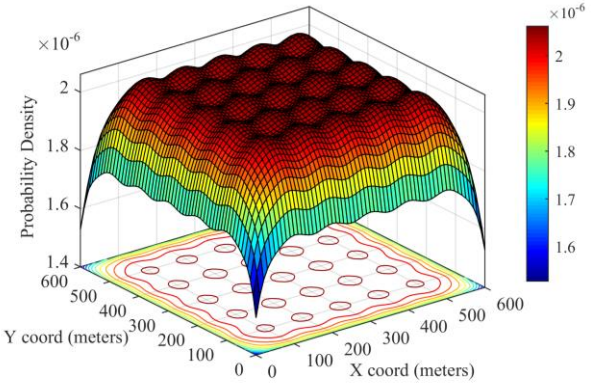

Figure 2. The overall probability distribution over the entire deployment region.

Furthermore, let $l = \sqrt{(x - x_i)^2 + (y - y_i)^2}$ be the Euclidian distance between node k and the predefined position g_i. Then (1) can be rewritten as:

$$f_{eu}(l|k \in G_i) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{l^2}{2\sigma^2}\right). \tag{2}$$

3.3. Attack Model

In this work, we assume that the adversary can launch a node replication attack by compromising a subset of nodes, generating a large amount of their replicas, and spreading the replicas throughout the networks. Upon compromising a node u, the adversary is able to produce a group of replicas $u' = \{u'_1, u'_2, \ldots, u'_r\}$ of which the IDs and secret materials are the same as the original compromised node u. The replicas can easily bypass the authenticity and integrity validation of the existing cryptographic security mechanism since they can sign, encrypt, and decrypt the message to play the role just like their original compromised node. Once the replicas are recognized as a legal part of the network, they can launch a variety of attacks, such as false data injection, protocol disruption, and traffic jamming. Moreover, replicas can also assist the original compromised nodes to extend the attack range, as well as reduce the attack cost.

However, we impose several constraints on the adversary's behaviors. We assume that the adversary is not able to generate new legal IDs since all of the nodes' IDs are determined before deployment following the deployment strategy. Additionally, the adversary also cannot extract the data in the nodes' memory before they are compromised. We also assume that the adversary can only compromise a minority of sensor nodes since if he can compromise a major fraction of the network, he will not benefit much from the node replica attack. Furthermore, the adversary would make every effort to extend the deployment range of the replicas; in other words, the replica should be placed at a distance from its origins. Although the replica nodes are hard to detect when they are place close to their original compromised nodes, this will not bring any benefits to the adversary.

The adversary can undermine the location claim-based protocols by deploying large amounts of compromised nodes to report fake locations and participate in local control protocols. However, such an attack strategy requires the adversary to place one compromised node to accompany each replica in the network, resulting in a very high cost for launching node replica attacks. We suppose that the adversary does not adopt this attack strategy. This assumption goes unstated but is implied by the use of signed location claims in other replica detection schemes [6,8]. In addition, it is worthwhile to note that deploying multiple replicas of a single compromised node into the same region does not

bring the adversary more benefits. This is because the output of multiple replicas would be treated as redundant and discarded. Multiple replicas with the same ID would not have more influence in a region than a single replica. Furthermore, due to page limitation, in this work we only discuss the case in which no collisions exist between the compromised nodes and their replicas.

4. Localization-Free Replica Detection Based on Similarity Estimation

In this section, we first present a formal statement of the node replica problem, and propose a neighborhood relationship knowledge-based metric that allows us to detect replicas without assistance of nodes' positions using local sensitivity hashing (LSH)-based location similarity estimation techniques. We then describe the details of proposed protocol that stop node replica attacks without resource-consuming localization mechanisms. Finally we present a simulation model that estimate the detection threshold of the proposed scheme. Table 1 lists the most frequently used notation in this paper.

Table 1. Frequently-used notations in this paper.

R_z	the communication radius of sensor nodes and beacons	$k_{\langle i,j \rangle}$	secret key shared between node i and j
d_{th}	the trust threshold	$k_{\langle i,BS \rangle}$	secret key shared between node i and BS
τ_{CD}	the threshold of confliction detection	$C_{k_{\text{prv}}}$	the certification signed by BS
τ_{RD}	the threshold of replica detection	LAQ/LAR	the location authentication request/reply
S_{nei-D}	the derived neighboring vector	NAN	the node authentication needed request
S_{ob}	the observed neighboring vector	LCQ/LCD	the location claim request/decision
$k_{\text{prv}}/k_{\text{pub}}$	private/public key of BS		

4.1. Problem Statement

Once the adversary succeeds in compromising a node, he can create replica nodes as follows: he extracts ID and all secret materials from the compromised nodes and loads these key information into the replicas so that the replica has the same ID and secret materials as its origin node. The adversary can compromise multiple nodes and generate multiple replicas of a single compromised node. For a specific compromised node v, there may be n replica nodes in the network, denoted by $v_1, v_2, \ldots, v_n$. We assume that compromised node v is placed at the location $L_v\ (x_v, y_v)$ while the replicas are placed at locations $L_{v_k}(x_{v_k}, y_{v_k}), k = 1, 2, \ldots, n$. To bypass the location claim conflict detection, the replicas may falsify the location claims denoted by $L'_{v_k}\left(x'_{v_k}, y'_{v_k}\right)$, which equals to L_v. A straightforward solution for this problem is to build a threshold-based detection mechanism on the distance between the claimed location L'_{v_k} and the true location L_{v_k}. However, the true location L_{v_k} is not visible for BS without accurate localization by multiple witness nodes. Actually, in most sensor network applications, accurate localization is difficult to achieve due to strict resource constraints. The inaccuracy existing in localization may incur large amounts of uncertainties and false alarms in the replica detection schemes. How to handle these uncertainties and enhance the effectiveness with a substantially low communication, computation, and storage overheads is an important issue explored in this paper.

4.2. Neighborhood-Based Detection Metric

To address the limitation of the localization based replica detection schemes, we propose a novel detection metric named location similarity based on neighborhood relationship knowledge. The basic

idea behind location similarity is that the neighborhood situation of two nodes should be very different when they are far away from each other. Taken the deployment situation shown in Figure 1 as example, the nodes u, v, and w are placed at locations L_u, L_v, and L_w, respectively. The location L_u has quite a long distance from L_v, while L_u are near L_v. From Figure 1, we can see that the neighboring nodes of u mostly reside in the groups G_A, G_B, G_C, and G_D, which is quite different from the neighboring nodes' distribution of node v (in G_H, G_I, G_J, G_K), but similar to the neighboring distribution of node w. Hence, we believe that it is reasonable to judge a node as a replica as long as the deviation of the neighboring node distribution between its claimed location and true location exceeds a predefined threshold.

We first introduce a notation *neighboring vector* to indicate the distribution of one node's neighborhood. We further define *derived neighboring vector* as the vector that indicates the neighboring distribution derived from one node's claimed location. Then we have:

Definition 1 (Neighboring Vector (NV))**.** *For any node u, the normalized vector $S_{nei}(u)$ is NV if and only if*

$$S_{nei}(u) = \frac{\left(S_{L_u}^{(G_1)}, S_{L_u}^{(G_2)}, \ldots, S_{L_u}^{(G_m)}\right)}{\sum_{i=1}^{m} S_{L_u}^{(G_i)}}, \tag{3}$$

where m is the number of groups while $S_{L_u}^{(G_i)}$ accounts for the number of u's neighboring nodes which belongs to group G_i when u is at its true location L_u.

According to Definition 1, we try to find out the mathematical mapping relationship between NV and the physical location based on the group deployment knowledge so that we can derive the neighboring distribution from the claimed location of sensor nodes. Theorem 1 and Inference 1 show such a relationship.

Theorem 1. *Let R_z be the communication radius of a sensor node. Let $g(z|k \in G_i)$ be the probability that node k from group G_i resides within the neighborhood of a node which is z distance from the predefined deployment point g_i. Then the probability $g(z|k \in G_i)$ is:*

$$g(z|k \in G_i) = I\{z < R_z\}\left[1 - e^{-\frac{(R_z - z)^2}{2\sigma^2}}\right] + \int_{|z-R_z|}^{z+R_z} f_{eu}(l|k \in G_i) \cdot 2l \arccos\left(\frac{l^2 + z^2 - {R_z}^2}{2lz}\right) dl, \tag{4}$$

where function $f_{eu}(l|n_i \in G_i)$ is the node distribution function illustrated in Equation (2), constant value R_z represents the communication radius of each node, and $I\{\cdot\}$ is the set indicator function. The value of $I\{\cdot\}$ is 1 when $z < R_z$ holds, and 0 otherwise.

Proof. For group G_i, the sensor nodes that are l-distance from deployment point g_i should reside in a circle which is centered at g_i with the radius l. If these nodes also reside in the communication range of the node u, they should reside in a circle which is centered at u with the radius R_z. In other words, as illustrated in Figure 3a,b, the nodes that satisfy the conditions under the lemma hypothesis should resides on the g_i's arc within the u's circle. Let $L_{arc}(l, z, R_z)$ denote the length of such arc. Based on the node distribution presented in Section 3.2, the probability that the nodes reside in u's communication range when they are l-distance from g_i, denoted by $g(z|\text{dist}(k, g_i) = l)$, can be derived as the probability that the node k falls on an infinitesimal ring area (the bold area in Figure 3) $L_{arc}(l, z, R_z) \cdot dl$. Then we have:

$$g(z|\text{dist}(k, g_i) = l) = f_{eu}(l|k \in G_i) \cdot L_{arc}(l, z, R_z) \cdot dl. \tag{5}$$

Based on the basic geometry knowledge, we can derive the length of arc L_{arc} as:

$$L_{arc}(l,\, z,\, R_z) = 2l \cdot \arccos\left(\frac{l^2 + z^2 - R_z^2}{2zl}\right). \tag{6}$$

When the condition $z \geq R_z$ holds, the line segments l, z, R_z form a triangle in which z is longer than R_z. Using the triangle axiom, l ranges from $z - R_z$ to $z + R_z$. Then $g(z|k \in G_i)$ can be derived as:

$$\begin{aligned} g(z|k \in G_i) &= \int_{z-R_z}^{z+R_z} f_{eu}(l|k \in G_i) \cdot L_{arc}(l,\, z,\, R_z) \cdot dl \\ &= \int_{z-R_z}^{z+R_z} f_{eu}(l|k \in G_i) \cdot 2l \cdot \arccos\left(\frac{l^2+z^2-R_z^2}{2zl}\right) \cdot dl. \end{aligned} \tag{7}$$

On the other hand, when the condition $z < R_z$ holds, we consider two different cases. In the first case, l ranges from $R_z - z$ to $R_z + z$, the value of L_{arc} is the same as Equation (6). In the second case, l ranges from 0 to $R_z - z$, the whole circle centered at g_i with the radius l resides inside the circle centered at α with the radius R_z. Then we have:

$$g(z|k \in G_i) = \int_0^{R_z-z} f_{eu}(l|k \in G_i) \cdot 2\pi l \cdot dl + \int_{R_z-z}^{R_z+z} f_{eu}(l|k \in G_i) \cdot 2l \cdot \arccos\left(\frac{l^2 + z^2 - R_z^2}{2zl}\right) \cdot dl \tag{8}$$

We can then merge Equations (7) and (8) with the assistant of *indicator function*, as follows:

$$g(z|k \in G_i) = I\{z < R_z\}\left[1 - e^{-\frac{(R_z-z)^2}{2\sigma^2}}\right] + \int_{|z-R_z|}^{z+R_z} f_{eu}(l|k \in G_i) \cdot 2l \cdot \arccos\left(\frac{l^2 + z^2 - {R_z}^2}{2lz}\right) \cdot dl. \tag{9}$$

Hence, Lemma 1 has been proved. Similar derivation can also be found in [24]. □

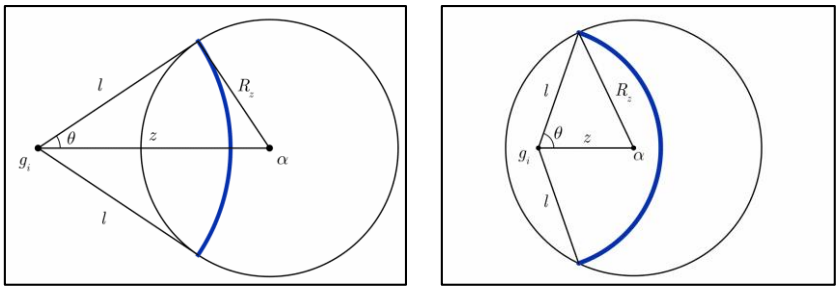

Figure 3. Probability of the nodes which are l-distance from g_i residing in α's communication range: (**a**) $z \geq R_z$; (**b**) $z < R_z$.

Theorem 1 indicates that for a specific node u, we can obtain the probability that the nodes from any group G_i reside in its neighborhood given node u's radio radius R_z, and its distance z from the corresponding deployment point g_i. Then we can derive the average number of nodes that reside in node u's communication range for every group in the deployment region given the physical location of u. We use a derived neighboring vector (DNV) to describe such average neighboring distribution of a node. Inference 1 shows how we get DNV, based on Theorem 1.

Inference 1. *Given the location L_u of node u and deployment points $\{g_1, g_2, \ldots g_m\}$ over the entire network, the normalized DNV derived by location L_u can be obtained as*

$$S_{nei-D} = \frac{\left(\mu_{L_u}^{(G_1)}, \mu_{L_u}^{(G_2)}, \ldots, \mu_{L_u}^{(G_m)}\right)}{\sum_{i=1}^{m} \mu_{L_u}^{(G_i)}}, \quad \mu_{L_u}^{(G_i)} = \frac{M}{m} g\left(\text{dist}\left(L_u, L_{g_i}\right) | k \in G_i\right), \tag{10}$$

where M is the total number of nodes in the network, m is the number of deployment groups, whereas $g\left(\text{dist}\left(L_u, L_{g_i}\right) | k \in G_i\right)$ accounts for the probability that the nodes from group G_i reside in the communication range of node u.

Proof. Consider group G_i, according to the deployment strategy described in Section 3.2, the number of nodes in such group is M/m. Since every node in group G_i resides in the communication range of u with a success probability $g\left(\text{dist}\left(L_u, L_{g_i}\right) | k \in G_i\right)$. The number of nodes from G_i residing in u's range is a random value which follows Bernoulli distribution. Hence, the mean of such random values comes to $(M/m) g\left(\text{dist}\left(L_u, L_{g_i}\right) | k \in G_i\right)$. □

Based on the bijective relationship between the physical location and neighboring distribution shown in Theorem 1 and Inference 1, we use the deviation of the neighboring vector instead of the Euclidian distance to characterize the difference between two physical locations. The former is much easier to obtain and compute in a practical sensor network deployment. To further reduce the complexity of computation of inter-vector distance, we adopt the locality sensitive hashing (LSH) coding algorithms, such as MinHash and SimHash [12], to encode the neighboring vector, so that the computation of the inter-vector distance can be simplified as a computation of the Hamming distance between two binary sequence. We define the NV-based location similarity as follows.

Definition 2. *Neighboring vector-based location similarity (NV-LS). Let $S_{nei-\text{I}}, S_{nei-\text{II}} \in \mathbb{R}^m$ denote neighboring vectors corresponding to two different locations. Let $\text{LSH}(\cdot)$ be LSH encoding function whose output is b-bit binary code. According to LSH sequence similarity defined in* [12]*, the NV-LS of S_{nei-I} and S_{nei-II} is*

$$\text{sim}(S_{nei-\text{I}}, S_{nei-\text{II}}) = 1 - \frac{D_h(\text{LSH}(S_{nei-\text{I}}), \text{LSH}(S_{nei-\text{II}}))}{b}, \tag{11}$$

where $D_h(\text{LSH}(S_{nei-\text{I}}), \text{LSH}(S_{nei-\text{II}}))$ is the Hamming distance between the binary sequences $\text{LSH}(S_{nei-\text{I}})$, $\text{LSH}(S_{nei-\text{II}})$. The location similarity is a real-value that ranges from 0 to 1. The two locations are very close to each other when the value of location similarity tends to 1.

4.3. Protocol Description

We assume that M sensor nodes in the network are divided into m groups and each group has M/m nodes. As illustrated in Figure 1, the beacon nodes are placed accurately at the deployment points which are distributed in grids with grid spacing d_g. Every node has a unique identity (ID) which includes two parts: the node ID (NID) and group ID (GID). We also assume that key materials are pre-loaded to sensor nodes for pairwise key establishment and other security mechanisms. In addition, we adopt an aggregated MAC approach proposed in [20] for all of the MAC generation and verification in our work. The aggregated MAC is a lightweight data integrity verification technique. In our previous works [25], we demonstrate that the aggregated MAC is affordable for sensor networks by evaluating the computation overhead on the MICA2 motes. Additionally, the trust-based scheme [19] is also an alternative solution for the integrity verification. In this approach, the authentication tags can be aggregated by performing an XOR operation, namely the aggregated tag can be obtained by:

$$\text{Tag} = \bigoplus_{i=1}^{n} \text{tag}_i = \bigoplus_{i=1}^{n} \text{MAC}_{k_i}(\text{data}_i), \quad \text{tag}_i = \text{MAC}_{k_i}(\text{data}_i). \tag{12}$$

The size of the aggregated tag equals to any single tag_i. Whereas it can be used to verify the integrity of $\text{data}_1 \| \text{data}_2 \| \ldots \| \text{data}_n$. The proposed protocol includes three phases as follows:

Phase 1: Initiation. We first define a system parameter *trust threshold*, denoted by d_{th}. Consider a particular node u from group G_i; we define G_i as u's home group. Node u accepts and forwards the messages from the nodes in group G_j if the Euclidian distance between g_i and g_j is smaller than d_{th}. This means that if nodes from the group whose deployment point is far enough from the deployment point of u's home group, the probability that they become u's neighbors is small enough to be ignored. Thus, prior to deployment, for each group, the network deployer draws a circle centered at its deployment point with radius d_{th}, records the deployment points that falls in such circle region on a *trust list*, and pre-loads the trust list into all the members of this group. Moreover, the network deployer uses a non-interactive public key establishment algorithm named SOK [26] to build pairwise keys shared between BS and sensor nodes. Let k_{prv} be the private key held by BS, and k_{pub} be the public key preloaded into sensor nodes. Using the private key k_{prv}, the network deployer generates a certification $C_{k_{\text{prv}}}(\text{NID}_u \| \text{GID}_u)$ on node u's ID and group ID, and preloads the certification together with node u's unique identity. In addition, BS shares a unique secret key with each node in the network, denoted by $k_{\langle i, \text{BS} \rangle}$, $i = 1, 2, \ldots, M$. This shared secret key is also preloaded into the nodes before deployment.

Immediately after deployment, the beacon nodes located at the deployment points start to broadcast beaconing messages. The sensor nodes over the entire network keep on listening to the beaconing messages and recording the RSSI of the beaconing messages using the techniques from the Media Access Control layer protocols until the initiation phase ends. During the initiation phase, the nodes pick the largest three RSSI values and the corresponding deployment points. Meanwhile, the nodes start a neighbor discovery process and try to establish a unique pairwise key with each one of their immediate neighbors using some secret sharing techniques such as key pre-distribution [21,22]. Let $k_{\langle u, v \rangle}$ be the pairwise key shared between node u and v. During this process, the nodes also authenticate the integrity of their neighbors' identities and belonging group by verifying the certification $C_{k_{\text{prv}}}$ signed by BS. The nodes will rejects the forwarding requests by their neighbors which do not pass such authentication.

Phase 2: Location Claim Generation and Probabilistic Forwarding. Suppose that a node u in group G_i receives a request from node v to forward a message. Node u first checks the group part of node v's identity $(\text{NID}_v \| \text{GID}_v)$ to make sure that node v's identity is authenticated and its group ID is on the trust list. If so, node u accepts node v as a benign node and forwards the message as requested. Otherwise node u rejects to forward node v's message and sends back a *Node Authentication Needed* (NAN) request to ask for authentication that proves node v's integrity.

Upon receiving the NAN request, node u broadcasts a location authentication request (LAQ) around its neighborhood. The LAQ message consists of node u's ID and corresponding timestamp, denoted by:

$$\text{LAQ} = \langle \text{ID}_u \| ts_{\text{id}} \rangle. \tag{13}$$

After sending LAQ, node u start a timer $t_{\text{wait}-\text{LAR}}$ to wait for its neighbors' reply.

Once the neighbors $\{v_1, v_2, \ldots v_r\}$ of node u receive the LAQ request, they generate a message authentication code of their own IDs and corresponding timestamps using the key shared with BS. Then they encrypt all these data with the key shared with node u and send back the encrypted message to node u. We define this encrypted message as location authentication reply (LAR), represented as:

$$\text{LAR} = \left\langle \text{enc}_{k_{\langle v_i, u \rangle}} \left(\text{ID}_{v_i} \| \text{MAC}_{k_{<v_i, BS>}}(\text{ID}_{v_i}) \right) \right\rangle, \; i = 1, 2, 3, \ldots, r, \tag{14}$$

where ID_{v_i} represents the ID of the neighboring node v_i, $ts_{id_{v_i}}$ is a timestamp representing the time that LAR is sent back, $k_{\langle v_i, BS \rangle}$ and $k_{\langle v_i, u \rangle}$ represent the key shared between v_i and BS, as well as between v_i and u, respectively. The $\text{enc}(\cdot)$ and $\text{MAC}(\cdot)$ represent the encryption and MAC generation functions, respectively.

When the timer $t_{\text{wait}-\text{LAR}}$ runs out, node u collects the received LAR messages, decrypts them, and extracts neighbors' IDs and corresponding authentication tags from decrypted messages. Then node u obtains IDs $= \{\text{ID}_{v_1}, \text{ID}_{v_2}, \ldots, \text{ID}_{v_r}\}$ and Tags $= \left\{\text{Tag}_{v_1}, \text{Tag}_{v_{21}}, \ldots \text{Tag}_{v_r}\right\}$, where $\text{Tag}_{v_i} = \text{MAC}(\text{ID}_{v_i})$. Using the aggregated MAC approach, node u generates location claim request (LCQ), as LCQ $= \langle L_{\text{RSS}} \| \text{IDs} \| \text{Tag} \rangle$. The L_{RSS} field contains the collected three largest RSSI and corresponding IDs of beacons. The IDs field contains the IDs of node u's neighbors which reply u's LAQ, as well as the ID of node u itself. The Tags field contains the authentication tags of IDs of node u's neighbors, as well as the authentication tag of L_{RSSI} and node u's ID. The detailed description of node u's LCQ is as follows:

$$\begin{aligned} &L_{\text{RSS}} = \langle \text{RSSI}_x \| \text{RSSI}_y \| \text{RSSI}_z \| g_x \| g_y \| g_z \rangle, \\ &\text{IDs} = \langle \text{ID}_u \| \text{ID}_{v_1} \| \text{ID}_{v_2} \| \ldots \| \text{ID}_{v_r} \rangle, \\ &\text{Tag} = \left\langle \text{MAC}_{k_{\langle u, \text{BS} \rangle}} (L_{\text{RSSI}} \| \text{ID}_u) \oplus \left(\bigoplus_{i=1}^{r} \text{MAC}_{k_{\langle v_i, \text{BS} \rangle}} (\text{ID}_{v_i}) \right) \right\rangle, \\ &\text{LCQ} = \langle L_{\text{RSS}} \| \text{IDs} \| \text{Tags} \rangle. \end{aligned} \tag{15}$$

Note that the reason we add node u's ID and corresponding authentication tag is to prevent the replay and forgery attacks.

After the task of LCQ generation, node u selects several nodes from its neighbors in IDs fields to forward the LCQ message. When the selected neighbors receive the LCQ from node u, they check whether its ID is on the IDs filed. If so, they forward the LCQ message to BS with the probability P_f, otherwise directly discard such message.

Phase 3: Replica Detection and Revocation. Upon receiving LCQ message from node u, BS computes $\text{Tag}' = \text{MAC}_{k_{\langle u, \text{BS} \rangle}} (L_{\text{RSSI}} \| \text{ID}_{\text{u}}) \oplus \left(\bigoplus_{i=1}^{r} \text{MAC}_{k_{\langle v_i, \text{BS} \rangle}} (\text{ID}_{v_i}) \right)$ using the key shared with node u and its neighbors $v_1, v_2, \ldots, v_r$ to find out whether Tag$'$ equals to Tag, so as to verify the integrity of $L_{\text{RSS}} \| \text{IDs}$. If LCQ does not pass integrity verification, BS directly discards the LCQ message. Otherwise, BS parses the data payload $L_{\text{RSS}} \| \text{IDs}$ and accordingly decides whether node u is replica in the following steps.

Step 1: DNV extraction. Using the RSSI distance measurement model proposed in [27], BS can derive node u's distance to the corresponding beacon nodes. Let constant A be the signal strength (in dBm) received at the point that 1 meter away from the signal source, and r be multipath fading factor. Both A and r are regarded as the prior network environment knowledge. Given the RSSI P_R (in dBm) about the signal source, the distance d between receiver and source is obtained by:

$$\lg d = -\frac{P_R - A}{10r}. \tag{16}$$

According to Equation (16), we can derive the distance from node u to deployment point g_x (respectively, g_y, g_z), denoted by $d(g_x, u)$ (respectively, $d(g_y, u)$, $d(g_z, u)$). The deployment points g_x, g_y, g_z are the nearest to node u. Based on the triangle axiom, it is easy to infer that node u is located in a triangular region of which the vertexes are g_x, g_y, g_z, as illustrated in Figure 4a.

Given the distances $d(g_x, u)$, $d(g_y, u)$, $d(g_z, u)$, and angle $\alpha = \pi/4$, the angle β can be obtained by:

$$\angle\beta = \arccos\left(\frac{2d_g^2 + d^2(g_x, u) - d^2(g_z, u)}{2d(g_x, u)d(g_z, u)} \right). \tag{17}$$

Based on the primary geometric theory, we can derive the distance between node u and deployment point g_w:

$$d(g_w, u) = \sqrt{d^2(g_x, u) + d_g^2 - 2d_g d(g_x, u) \cos\left(\beta + \frac{\pi}{4}\right)}, \tag{18}$$

and the radian value of angle θ:

$$\measuredangle\theta = \pi - \arccos\left(\frac{d_g^2 - d(g_x, u)d_g\cos\left(\beta + \frac{\pi}{4}\right)}{d(g_w, u)d_g}\right). \tag{19}$$

Consider that the deployment point g_{w1}, g_{w2}, …, g_{wk} that in the three o'clock direction of g_w, as shown in Figure 4b, the distance between node g_{wi} and g_w is $i \cdot d_g$, $i = 1, 2, \ldots, k$. Given the distance $d(g_w, u)$, $d(g_{wi}, g_w)$, and the angle θ, we can obtain the distance from node u to the nodes $\{g_i | i = 1, 2, \ldots, k\}$ as:

$$d(g_{wi}, u) = i^2 d_g^2 + d^2(g_w, u) - 2i \cdot d_g \cdot d(g_w, u)\cos\theta,\ i = 1, 2, \ldots, k \tag{20}$$

By using similar methodology, BS can obtain the distance between node u and other deployment points in the target region. Let $Z = \{z_{g_i} = d(g_i, u) | i = 1, 2, \ldots m\}$ denote such distances to the deployment points over the entire network. According to Inference 1, BS can derive the DNV with respect to location information L_{RSS} based on set Z, denoted by:

$$S_{nei-D}(u) = \frac{\left(\mu_{L_u}^{(G_1)}, \mu_{L_u}^{(G_2)}, \ldots, \mu_{L_u}^{(G_m)}\right)}{\sum_{i=1}^{m} \mu_{L_u}^{(G_i)}},\ \mu_{L_u}^{(G_i)} = \frac{M}{m} g\left(z_{g_i} | k \in G_i\right),\ i = 1, 2, \ldots, m. \tag{21}$$

Step 2: Conflict detection. Using the method in Step 1, BS derives the distance between node u and deployment point g_u. If the value of distance $d(u, g_u)$ is larger than a threshold τ_{CD}, BS will determine node u as replica. Furthermore, BS searches the cached location claims that pass the detection to find out whether it has received a former version of node u's location claim. If so, BS compares the L_{RSS} fields of the two versions to check whether a conflict exists. In the case that there exists a conflict, BS removes the conflicted location claim from cache and start the replica revocation operations in Step 3.

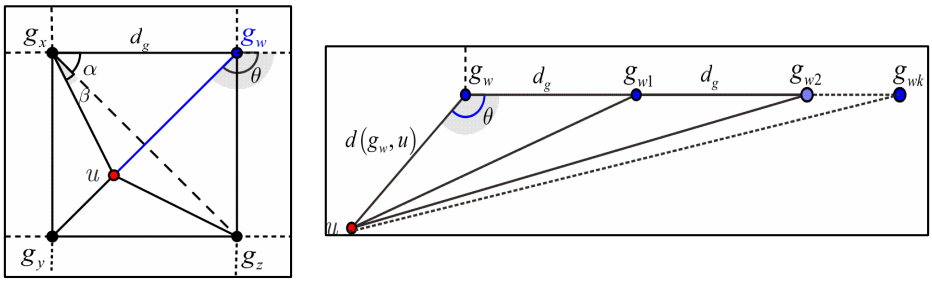

Figure 4. Derivation of DNV by the RSSI data from the L_{RSS} field of LCQ. (**a**) Deciding node u's position in a single grid using RSSI of the nearest three deployment points; and (**b**) derivation of the distances between node u and the deployment points using the group deployment knowledge.

Step 3: Decision and revocation. BS parses the IDs field, and counts the IDs of node u's neighbors by groups to which they belong. Then BS obtain a observed neighboring vector (ONV), which indicates the u's neighboring distribution observed by BS. Note that the ONV is generated by node u's neighbors and its integrity is verified by BS, which means that it has high reliability. Let $S_{ob}(u) = \left\{S_{ob}^{(G_1)}(u), S_{ob}^{(G_2)}(u), \ldots, S_{ob}^{(G_m)}(u)\right\} / \sum_{i=1}^{m} S_{ob}^{(G_i)}(u)$ denote the node u's ONV. BS computes the NV-LS between the $S_{ob}(u)$ and $S_{nei-D}(u)$. BS firstly encodes $S_{ob}(u)$ and $S_{nei-D}(u)$ using LSH

coding algorithms like MinHash [12], of which the output are b-bit binary sequence. BS then computes the Hamming distance D_h of the two output sequences, and finally obtain the NV-LS as:

$$\mathrm{sim}(S_{nei-D}(u),\, S_{ob}(u)) = 1 - \frac{D_h(\mathrm{LSH}(S_{nei-D}(u)),\, \mathrm{LSH}(S_{ob}(u)))}{b}. \tag{22}$$

If the NV-LS is larger than a threshold τ_{RD}, BS determines node u as benign and add the L_{RSS} field of node u's location claim to cache of BS. Otherwise, it determines node u as replica and raises an alarm. After the decision is reached, BS scans $S_{ob}(u)$ again to find the non-zero elements which indicates the groups whose members reside in node u's neighborhood, and directed broadcasts the location claim decision (LCD) messages to such groups. The LCD can be represented as:

$$\mathrm{LCD} = \left\langle \mathrm{ID}_u \| \mathrm{RES} \| \mathrm{Sig}_{k_{\mathrm{priv}}} \right\rangle, \tag{23}$$

where the field RES is the detection result about node u, and $\mathrm{Sig}_{k_{\mathrm{priv}}}$ is the digital signature generated by BS with its private key. BS send several copies of LCD message to ensure that the LCD message reaches the node u's neighboring groups. Once such message reaches one member of the target group, it will be flooded throughout the entire group. The cost of such local flooding is limited because it happens in a very small region. This ensures that every member in the target groups receives the LCD message. The nodes in the target groups add node u to their conditional trust lists and start to forward the messages from node u when the LCD message indicates that node u is benign.

4.4. Obtaining the Replica Detection Threshold

In this work, we adopt a training approach to estimate the replica detection threshold τ_{RD}, and obtain the training data from network simulations. According to the deployment strategy presented in Section 3.2, we generate a network simulation scenario, deploy our proposed protocol and repeat the simulation for k rounds. In each round of simulation, we obtain the training data as follows:

Step 1. We obtain the actual physical position $L_a(i) = (x_{ai},\, y_{ai})$, where $i = 1,\, 2,\, \ldots,\, N$, and derive the actual DNV for the selected nodes, represented as:

$$S^a_{nei-D} = \left\{ S^a_{nei-D}(i) \,\middle|\, \frac{\left(\mu^{(G_1)}_{L_a(i)},\, \mu^{(G_2)}_{L_a(i)}, \ldots, \mu^{(G_m)}_{L_a(i)}\right)}{\sum_{i=1}^{m} \mu^{(G_i)}_{L_a(i)}},\, i = 1,\, 2,\, \ldots,\, N \right\} \tag{24}$$

Step 2. We collect the L_{RSS} field of LCQ message from the N selected nodes, and compute the corresponding estimated position $L_e(i) = (x_{ei},\, y_{ei})$ where $i = 1,\, 2,\, \ldots,\, N$. Then we derive the estimated DNV for the selected nodes, represented as:

$$S^e_{nei-D} = \left\{ S^e_{nei-D}(i) \,\middle|\, \frac{\left(\mu^{(G_1)}_{L_{ei}},\, \mu^{(G_2)}_{L_{ei}}, \ldots, \mu^{(G_m)}_{L_{ei}}\right)}{\sum_{i=1}^{m} \mu^{(G_i)}_{L_{ai}}},\, i = 1,\, 2,\, \ldots,\, N \right\} \tag{25}$$

Step 3. We collect the IDs field of the LCQ message from the selected nodes, and derive the ONV for them: $S_{ob} = \left\{ S_{ob}(i) \middle| \left(S^{(G_1)}_{ob}(i),\, \ldots,\, S^{(G_m)}_{ob}(i) \right),\, i = 1,\, 2,\, \ldots,\, N \right\}$.

Step 4. We derive the NV-LS of L_a and L_e, denoted by sim_a and sim_e, respectively, where $\mathrm{sim}_a = \left\{ \mathrm{sim}_a(i) \middle| \mathrm{sim}\left(S^a_{nei-D}(i),\, S_{ob}(i)\right),\, i = 1 \ldots N \right\}$, and $\mathrm{sim}_e = \left\{ \mathrm{sim}_e(i) \middle| \mathrm{sim}\left(S^e_{nei-D}(i),\, S_{ob}(i)\right),\, i = 1 \ldots N \right\}$. Then we can compute the deviation χ_{sim} between sim_a and sim_e caused by the network uncertainties and measurement errors, represented as:

$$\chi_{\mathrm{sim}} = \{ \chi_{\mathrm{sim}}(i) \,||\mathrm{sim}_a(i) - \mathrm{sim}_e(i)|,\, i = 1,\, 2,\, \ldots,\, N \} \tag{26}$$

After k round repeated simulations, we obtain k sample sequence χ_{sim}, i.e., $k \times N$ samples of the NV-LS deviation as the training data, which form a sample distribution. Then we use the ξ-percentile to decide the threshold from these training data, which means that $1 - \xi$ percent of the samples falls within the range $(\tau_{\text{RD}}, 1)$ if τ_{RD} equal to the value of the ξ-percentile.

5. Security Analysis

Unlike other security threats against WSNs, such as node compromise and forgery, thorough elimination of replica nodes in a sensor network is infeasible and uneconomic since there are several choices for the adversary to hide the existence of replicas from the detection schemes at the cost of minimizing the functionality of replicas. For instance, the adversary can place the replica nodes very close to its original compromised ones, or strengthen the replicas' transmission power towards the original ones' neighbors, to make the replicas act exactly the same as its original compromised nodes. In this way, it is extremely difficult to find out which node is a replica, although the adversary benefits very little from such a kind of node replica attack. Consequently, in the security analysis of this section, we concentrate on the investigation of the effectiveness of our scheme that suppresses the damage caused by the replicas of a given compromised node.

To evaluate the security performance of our proposal, we adopt an 80-bit security level (RSA-1024 equivalent) elliptic curve Diffie-Hellman (ECDH) scheme [28] to provide a security guarantee for the key establishment during the process of neighbor discovery. We also adopt an 80-bit security level Data Encryption Standard (DES) algorithm [29] for the LAR message transmission.

5.1. Limitation of the Impact Range of Node Replica Attack

We first define the number of ordinary nodes that accept the replicas and forward their messages as the metric to quantify the damage of the node replica attack. Suppose the adversary compromises a node v and scatters several replicas of node v in the target deployment region. Under the protection of our scheme, we can observe that only the nodes which have node v's group ID in their trust lists will accept the replicas as trusted neighbors. Recall that the length of the trust list depends on the value of d_{th}. We can accordingly derive the upper bound of the impact of node replica attack with respect to d_{th}. We can infer that the nodes are impacted by node v's replicas if and only if the deployment points of their home groups falls in the circle region centered at node v with the radius d_{th}. Then the upper bound $\varsigma_{\text{imp}}^{(\text{upper})}$ of the number of nodes impacted by node v's replicas can be obtained by:

$$\varsigma_{\text{imp}}^{(\text{upper})} = \frac{M}{m} \left\| \left\{ g_i \middle| \text{dist}(g_u, g_i) = d_g \sqrt{k^2 + r^2} \leq d_{\text{th}}, 0 < k, r \leq \left\lfloor \frac{\sqrt{m}}{2} \right\rfloor \right\} \right\|_0 . \tag{27}$$

To further investigate the impact of node replica attack under the limitation d_{th} of our scheme, we set an ideal scenario in which $M = 2560$ nodes are deployed in 700×700 m^2 area with $m = 64$ groups, the distance between deployment points $d_g = 100$ m, and the communication radius of sensor nodes and beacon nodes is 150 m. The numeric result about Equation (27) are shown in Figure 5. We observe that the upper bound of replica impact step up with the increase of d_{th}. That's what we expect since more groups are involved in the replicas' range when the trust threshold rises. When d_{th} falls in the range $(100, 150)$, about 6.25% of nodes are affected by the replicas. When d_{th} falls in the range $(150, 200)$, the fraction of the affected nodes rise to 12.5%. This is still a small fraction considering the very large impact range of replica attack. Thus, we should set d_{th} to a small value so as to suppress the impact of such a replication attack. However we would like to point out that the network connectivity and communication efficiency will be severely degraded when the value of d_{th} is very small since the node will keep discarding the messages from the nodes whose group IDs are not on the trust list, as shown in the analysis of next section.

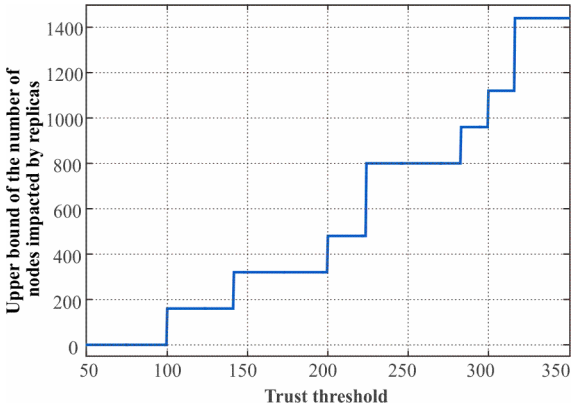

Figure 5. The trust threshold d_{th} vs. the upper bound of replica impact ς_{upper} given M = 2560, m = 64, $g_d = 100$.

5.2. Defense Capacity Analysis on Location Claim-Based Detection

To compensate for the degradation of the network connection caused by the trust-based selective forward, our scheme gives the nodes whose messages are rejected by their neighbors a second chance to prove their integrity by a neighboring vector-based location claim. However, we infer that the adversary can still take at least three potentially effective attack strategies against our scheme. We analyze the defense capacity under the three attack strategies respectively in this subsection.

Strategy I: GID forgery. In this strategy, the adversary modifies the GID of the replicas so that the replicas' neighbors will accept them as nodes from trusted groups, and forward their messages. However recall that the integrity of GID is protected by the certification $C_{k_{\text{prv}}}(\text{NID}\|\text{GID})$ signed by BS, as well as the BS being assumed to be a trusted entity, which means that its private key cannot be compromised. Thus, to achieve the modification of GID, the adversary has to compromise the public key certification algorithm. In this case, the defense capacity of this work depends on the security strength of the adopted certification algorithm itself, which is beyond our discussion. In other words, if the adopted certification algorithm provides enough security strength, our scheme can defeat such an attack strategy.

Strategy II: L_{RSS} forgery. In this strategy, the adversary generates falsified L_{RSS} field of LCQ according to the neighboring distribution derived from the IDs field so as to bypass the neighboring vector-based replica detection at BS. This means that the value of L_{RSS} should be kept consistent with the true position of replicas. The replicas, thereby, should be placed at the position less than τ_{CD} away from the deployment point of their home group, otherwise they will be caught by the conflict detection of BS. This constrains the adversary's benefits when the threshold τ_{CD} is small. For instance, when $\tau_{\text{CD}} = R_z$, the replicas and their origin compromised node should be deployed in the same group under the limitation of our scheme, which gains little benefits than that of simply performing node compromise attack. However according to the deployment model used in this work, the node deployment is not accurate but follows two-dimension Gaussian distribution. Several nodes locate at the positions outside the communication range of deployment points with probability, resulting in the false positives in our work. Let P_a be the false positive rate, i.e., the probability that the benign nodes are determined as replicas given they are located outside their home group. Then we have:

$$\begin{aligned} P_a &= 1 - \int_0^{2\pi}\int_0^{\tau_{\text{CD}}} f(\rho\cos\theta, \rho\sin\theta)\cdot \mathrm{d}\rho\mathrm{d}\theta \\ &= 1 - \int_0^{2\pi}\int_0^{\tau_{\text{CD}}} \frac{1}{2\pi\sigma^2}\exp\left(-\frac{\rho^2}{2\sigma^2}\right)\cdot \mathrm{d}\rho\mathrm{d}\theta \\ &= \exp\left(-\frac{\tau_{\text{CD}}^2}{2\sigma^2}\right), \end{aligned} \tag{28}$$

where σ is the standard deviation of the two-dimension Gaussian distribution, and the indicator about deployment accuracy in our context.

Using the same simulation scenario in Section 5.1, we consider three different cases in which $\sigma = 50, 100, 150$, respectively. In each case, we vary τ_{CD} from 50 to 300, the numeric results about the false positive rate of our scheme are shown in Figure 6. We observe that the false positive rate P_a decreases significantly with the increase of the conflict detection threshold τ_{CD}. This is reasonable since more suspected nodes pass the conflict detection when the value of conflict detection threshold is larger. We also observe that P_a increases with the rise of σ when τ_{CD} is fixed, which indicates that the security performance of our scheme enhances when the deployment accuracy improves. When $\sigma = 50$, $\tau_{\mathrm{CD}} = R_z = 150$, the value of false positive rate is less than 1%, which means that the proposed scheme achieves a promising performance when the parameter are properly configured.

Strategy III: IDs forgery. We assume that the adopted aggregated MAC and encryption algorithms provide enough security strength for our scheme to protect the integrity and non-repudiation of LAR message. This prevents the adversary from misleading the replica decision of BS by generating falsified LAR messages. However the adversary still has a chance to drive the compromised nodes to modify the IDs field. The replica nodes can deliberately remove several neighbors' identities in the IDs field of LCQ message. For a particular replica node v, it can remove the identities of neighbors belonging to node v's nearest groups, denoted by $G_{v1}, \ldots G_{vk}$, so as to reduce the corresponding elements $S_{\mathrm{ob}}^{(G_{v1})}, \ldots, S_{\mathrm{ob}}^{(G_{vk})}$ of node v's ONV $S_{ob}(v)$. Note that since node v cannot add falsified IDs into the IDs field, the value of $S_{\mathrm{ob}}^{(G_{v1})}, \ldots, S_{\mathrm{ob}}^{(G_{vk})}$ should be reduced to a very low level if node v is far away from its origin. Recall that because node v has to select neighbors in the IDs field for message forwarding, the behavior of removing a neighbors' identity from the IDs field results in a significant decrease in the success probability that node v's LCQ message arrives at BS. Hence, the negative ID forgery is limited by the probabilistic LCQ forwarding mechanism.

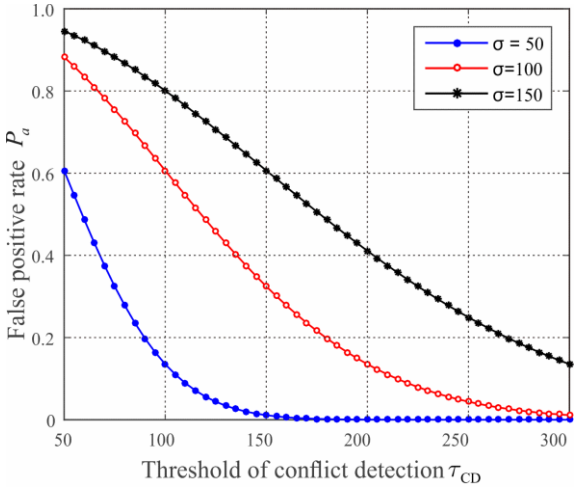

Figure 6. The conflict detection τ_{CD} vs. false positive rate P_a given $\sigma = 50, 100, 150$, respectively.

6. Performance Evaluation

In this section, we evaluate the performance of our proposed scheme from both an effectiveness and efficiency perspective. Firstly, we theoretically analyze the efficiency of our proposed scheme in terms of the communication, computation, and storage overheads, and then provide further quantitative evaluation on the effectiveness of our scheme through simulation experiments. Furthermore, we compare the performance of our scheme with the previous replica detection approaches [6,11].

6.1. Communication, Computation, and Storage Overhead

Communication overhead. In our context, the communication overhead is defined as the extra traffic brought by our scheme. We consider a worst case in which the trust threshold d_{th} is set to a small value so that the nodes will be asked for location claims if they are placed outside the range of its home group. According to Equation (28), the number to nodes that are required to forward location claims is $M \cdot P_a = M \cdot \exp\left(-\frac{\tau_{\text{CD}}}{\sigma^2}\right)$. For a particular node v, the average number of its neighbors is $N_v = \frac{M}{m}\sum_{i=1}^{m} g\left(\text{dist}\left(L_v, L_{g_i}\right) | k \in G_i\right)$. During the process of node v's location claim, r out of node v's neighbors ask node v for node authentication, node v sends LAQ to all its N_v neighbors, receives, at most, the same amount of LAR from those neighbors, and then forwards the LCQ to BS. Finally BS forwards LCD to the target groups. According to [6], the average hops between two randomly-selected nodes is approximately $O\left(\sqrt{N}\right)$. Thus, we can derive the communication overhead for a particular node as:

$$c_{\text{comm}} = r \cdot l_{\text{NAN}} + N_v \cdot (l_{\text{LAQ}} + L_{\text{LAR}}) + O\left(\sqrt{M}\right) \cdot l_{\text{LCQ}} + k \cdot O\left(\sqrt{M}\right) \cdot l_{\text{LCD}}, \tag{29}$$

where l_{NAN}, l_{LAQ}, l_{LAR}, l_{LCQ}, l_{LCD} are the message length of NAN, LAQ, LAR, LCQ, and LCD, respectively, whereas k is the number of nonzero elements of S_{ob}.

Then we have the upper bound of total communication overhead, as follows:

$$\begin{aligned} C_{\text{comm}} &= M \cdot P_a \cdot \left(r \cdot l_{\text{NAN}} + \overline{N}_{\text{nei}} \cdot (l_{\text{LAQ}} + L_{\text{LAR}}) + O\left(\sqrt{M}\right) \cdot l_{\text{LCQ}} + k \cdot O\left(\sqrt{M}\right) \cdot l_{\text{LCD}}\right) \\ &= O\left(P_a \cdot \left(M\overline{N}_{\text{nei}} + M\sqrt{M}\right)\right), \end{aligned} \tag{30}$$

where $\overline{N}_{\text{nei}}$ is the average number of neighbors for a node. From Equation (30), we can see that the upper bound of communication overhead in the worst case depends on the total number of nodes in the entire network and average number of neighbors.

Computation overhead. Since, in our scheme, the cryptographic operations consume the overwhelming majority of the computation resources, we use the average number of cryptographic operations as a metric to measure the computation overhead for our scheme. We assume that BS is a trusted entity with strong computation and storage capacity, so we only focus on the computation overhead that incurs at the ordinary sensor nodes. Let Q_{cert} denote the computation overhead for the single-time operation of certification verification, let Q_{MAC} denote the overhead for the single-time operation of MAC generation, and let Q_{ENC} denote the overhead for the single-time operation of message encryption. Besides, we assume that f_c is the fraction of the nodes are replicas, and they are placed randomly following a two-dimensional uniformly distribution. For a particular node u, it has to verify the certifications of their neighbors' IDs N_u times during neighbor discovery, where N_u is the number of node u's neighbors. Node u also has to generate the MAC and encrypt the LAR message for the nodes needed to be location authenticated. Note that. in the worst case, there are $N_u \cdot f_c$ replicas and $P_a(1 - f_c)N_u$ benign nodes are asked for location claims. Finally, in the case that node u is asked for location claims (the case incurs with the probability $f_c + (1 - f_c)P_a$), node u generates an extra MAC of its own ID for LCQ message. Then we obtain the computation for a particular node by:

$$c_{\text{cmp}} = \overline{N}_{\text{nei}} Q_{\text{cert}} + \overline{N}_{\text{nei}}(f_c + (1 - f_c)P_a)(Q_{\text{MAC}} + Q_{\text{enc}}) + (f_c + (1 - f_c)P_a)Q_{\text{MAC}}. \tag{31}$$

The total computation overhead for our scheme is accordingly:

$$\begin{aligned} C_{\text{cmp}} &= M\overline{N}_{\text{nei}} Q_{\text{cert}} + M\left(\overline{N}_{\text{nei}} + 1\right)(f_c + (1 - f_c)P_a)Q_{\text{MAC}} + M(f_c + (1 - f_c)P_a)Q_{\text{enc}} \\ &= O\left(\overline{N}_{\text{nei}}(f_c + (1 - f_c)P_a) + (f_c + (1 - f_c)P_a)\right). \end{aligned} \tag{32}$$

Storage overhead. As aforementioned, we focus on the storage occupation of our scheme on the resource-constrained sensor nodes. Recall that the conflict detection and replica detection are

performed at BS, thus, the sensor nodes in the network do not need to cache other nodes' location claims. In this work, the nodes only keep the secret key and certification signed by BS in memory, whose size is negligible compared to that of location claims.

Comparison of the resource efficiency. We compare the resource efficiency with the prior works [6,14] in terms of communication, computation, and storage overhead. The corresponding comparison results are shown in Tables 2–4. We present the results on the additional traffic incurred by the schemes all over the network in Table 2, the results on the average computation overhead on each node in the network in Table 3, and the results on the storage occupation for the location claims in Table 4.

Table 2. Communication overhead comparison.

Scheme	Communication Overhead
Randomized Multicast [6]	$O(k^* M^2)$
Line-selected Multicast [6]	$O\left(k^* \cdot M\sqrt{M}\right)$
Location Claim Scheme I [11]	Negligible
Location Claim Scheme II [11]	$O\left(P_a \cdot M\sqrt{M}\right)$
Location Claim Scheme III [11]	$O\left(P_a \cdot M\sqrt{M} \cdot \log_2(m)\right)$
Our proposed scheme	$O\left(P_a \cdot \left(M\overline{N}_{\text{nei}} + M\sqrt{M}\right)\right)$

* k is the rounds of location claim executed in [1].

Table 3. Computation overhead comparison.

Scheme	Computation Overhead
Randomized Multicast [6]	$O\left(k^*\sqrt{M}\right)$
Line-selected Multicast [6]	$O\left(k^*\sqrt{M}\right)$
Location Claim Scheme I [11]	Negligible
Location Claim Scheme II [11]	$O\left(\overline{N}_{\text{nei}} + P_a \cdot \left(\frac{M}{m}P_s^{**} + \sqrt{M}\right)\right)$
Location Claim Scheme III [11]	$O\left(\overline{N}_{\text{nei}} + P_a \cdot \left(\frac{M}{m}P_s^{**} + \log_2(m)\sqrt{M}\right)\right)$
Our proposed scheme	$O(\overline{N}_{\text{nei}}(f_c + (1-f_c)P_a) + (f_c + (1-f_c)P_a))$

** Ps is the probability that the nodes in the replica's home group caches the replica's location claim.

Table 4. Storage overhead comparison.

Scheme	Claim Storage Overhead
Randomized Multicast [6]	$O\left(k^*\sqrt{M}\right)$
Line-selected Multicast [6]	$O\left(k^*\sqrt{M}\right)$
Location Claim Scheme I [11]	Negligible
Location Claim Scheme II [11]	$O\left(\overline{N}_{\text{nei}} + P_a \cdot \frac{M}{m} \cdot p_s\right)$
Location Claim Scheme III [11]	$O\left(\overline{N}_{\text{nei}} + P_a \cdot \frac{M}{m} \cdot \log_2(m) \cdot p_s\right)$
Our proposed scheme	Negligible

As illustrated in Tables 2–4, the overhead of the line-selected and randomized multicast scheme in [6] linearly increases with the rounds of location claim over the network lifetime, which means that

the overhead of these schemes will exceed the other reference schemes as time increases. The scheme I in [11] is clearly the most resource efficient since it hardly brings any additional tasks to network routines for replica detection. However, this scheme only provides primary security services and will be defeated if the replica nodes modify their group ID to bypass the trust threshold-based detection of scheme I. Compared with scheme I, the scheme II and III in the same paper provide much better security performance at the cost of extra overhead incurred by location claims. However those two schemes still cannot resist the attack strategy in which the replicas provide the falsified location claims that exactly the same as their original compromised nodes. In contrast, our scheme can effectively resist the falsified group ID and falsified location claim attack strategies. The communication overhead of our scheme is slightly higher than that of Scheme II, and much lower than that of Scheme III, given that the average number of neighbors $\overline{N}_{\text{nei}}$ is much less than the total number of nodes M in most cases, whereas the computation and storage overhead of our scheme is much lower than similar schemes including Scheme II and III. This is because, unlike Scheme II and III, our scheme is able to protect integrity of the location claims without the en-route digital signature used in the two schemes. Additionally, the conflict check and replica detection in our scheme is performed at BS, which significantly reduces the computation overhead and storage occupation on the resource-constraint sensor nodes.

From the analysis above, we believe that compared with the prior works [6,11], our proposed scheme enhances the security resilience to the various attack strategies at the cost of a reasonable increase of communication overhead. Additionally, our work achieves promising performance in terms of computation and storage overhead. This is because the major computation tasks are deployed on BS.

6.2. Experimental Setup and Methodology

To validate the previous theoretical analysis results, we conduct experiments on the TOSSIM [30] platform to evaluate effectiveness and efficiency in terms of the detection rate and communication overhead under various network configuration. Based on the experiment results, we compare the performance of the proposed scheme with the previous discussed works, namely the randomized multicast and line-selected multicast schemes in [6], as well as Schemes I–III in [11].

In this simulation, following the network deployment strategy in Section 3.2, we establish a simulation scenario by placing M sensor nodes, which is averaged to m = 64 groups, in a 700 $\times$ 700 m^2. The target area are divided into a 8 $\times$ 8 mesh grid, and the deployment points of each group are placed at the cross points of such grid. The group members are deployed following the two-dimension joint Gaussian distribution, where the mean is the coordination of corresponding deployment point while the standard deviation is σ. We set the maximum communication radius $R_z = 150$ m, and the distance between two neighboring deployment points $d_g = 100$ m. We assume that the target area is located in an open space with a low-level asymmetric radio channel, and choose the corresponding simulation parameters as shown in Table 5.

Table 5. Simulation parameters.

Parameter	Value
Power decay in reference distance (A)	55 dB
Maximum data rate	250 Kbps
Packet size	36 Bytes
Average radio noise floor	−110 dBm
Standard deviation for WGN	4.0 dB
Receiving Sensitivity	−105 dBm

To emulate the three aforementioned attack strategies, we take the following procedure:

Step 1. After deployment, we randomly pick k ($k = \lfloor 0.005 \times M \rfloor$) nodes from the network topology, and mark them as compromised nodes. Let L_{c1}, L_{c2}, ..., L_{ck} denote the locations of these k compromised nodes.

Step 2. For each compromised node, we generate r replica nodes and place them D meters away from their original compromised nodes. L_{p1}^{ci}, L_{p2}^{ci}, ... L_{pr}^{ci} denote the locations of compromised node i's (i = 1, 2, ... , k) replicas, where $\left| L_{pj}^{ci} - L_{ci} \right| = D$.

Step 3. In attack strategy I, the replica node modifies its group identity GID to the nearest group while keeping its NID the same with its origins. In attack strategy II, the replica nodes keep their L_{RSS} field consistent with the IDs field in their LCQ message. In attack strategy III, the replica nodes make their L_{RSS} field the same with their original compromised nodes, and keep their IDs field consistent with the L_{RSS} field in the LCQ message.

Moreover, in order to investigate the performance of the proposed scheme under different environments, we adopt a variety of network configurations by varying the deployment standard deviation σ from 50 to 150, the number of replicas for each original compromised node r from one to five, as well as the distance between replicas and origins D from 150 to 300. We use two metrics to evaluate the proposed scheme: (1) detection rate: suppose there are m replica nodes in the network, and n of them are detected, the detection rate is measured as n/m; and (2) average communication overhead: during the entire lifetime of the simulation, the average packets of the proposed scheme sent per node. For each network configuration, we conduct our simulation for 6000 s, and repeat the simulation 100 times, using the average value of the above metrics as the experiment results.

6.3. Results and Discussion

Detection Rates under Attack Strategy I. To investigate the effectiveness under various attack strategies, we present the detection rate of the proposed and prior works under different D, and σ. We set the total number of sensor nodes $M = 2560$, the number of compromised nodes $k = 12$, and the number of replicas for each compromised node $r = 3$. The detection rates of the proposed and previous works under attack strategy I are shown in Figures 7 and 8. We observe that the Scheme I–III have almost no resilience to attack strategy I, since the adversary can make sensor nodes accept the replicas as trusted neighbors by falsifying the GID field of the replicas' message. In this case, Schemes I–III will not trigger the location claim detection mechanism. On the other hand, our proposed scheme and the randomized and line-selected multicast can effectively resist such attack strategies since all of these schemes do not rely on the trusted neighbor detection mechanism. Furthermore, our proposed scheme achieves a promising effectiveness and robustness under different D and σ, the detection rate comes to around 97% in all of the network configurations. This is because, in our scheme, the integrity of GID is guaranteed by the certification signed by BS, which can hardly be compromised by the adversary.

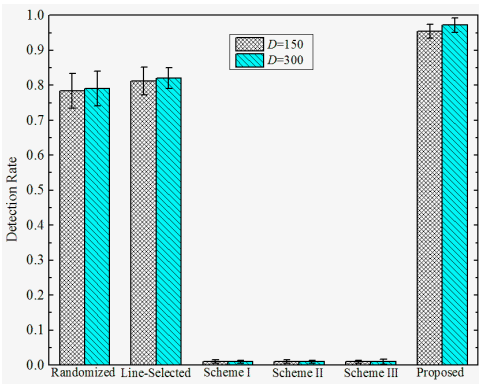

Figure 7. Detection rate of the proposed versus prior works under attack strategy I when $\sigma = 100$.

Detection Rate under Attack Strategy II. The detection rates under attack strategy II are shown in Figures 9 and 10. We observe that both the proposed and previous works can achieve relatively high detection rates against attack strategy II. This is because they adopt similar location claim confliction detection mechanisms. Furthermore, the detection rates of all the schemes rise when the distance D between replicas and their origins becomes higher. We infer that is because the deviation between the replicas and their origins becomes larger, resulting in higher detection rates, whereas the detection rates of all of the schemes decrease as the standard deviation σ increases. This means that the detection rates are hindered by the deployed accuracy.

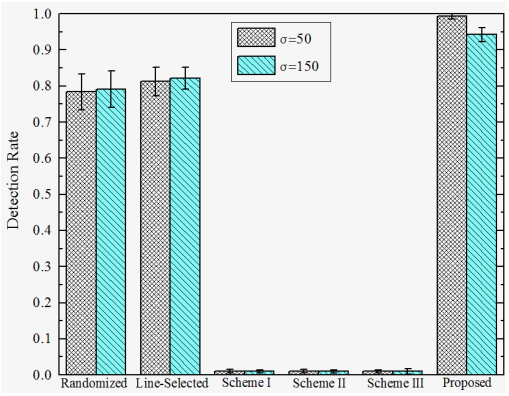

Figure 8. Detection rate of proposed versus prior works under attack strategy I when D = 200.

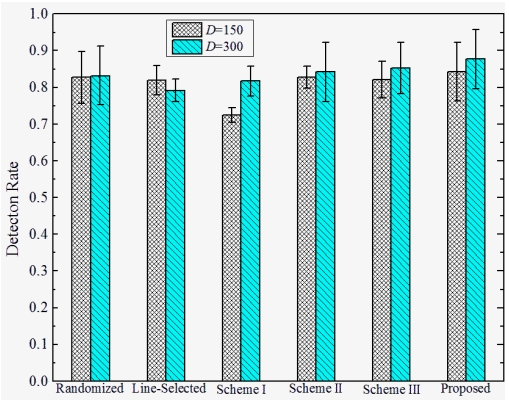

Figure 9. Detection rate of the proposed versus prior works under attack strategy II when $\sigma = 100$.

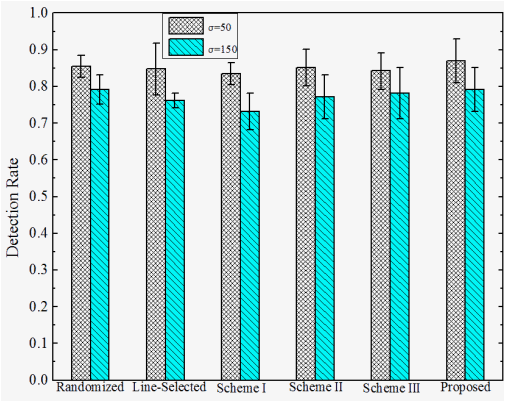

Figure 10. Detection rate of the proposed versus prior works under attack strategy II when D = 200.

Detection Rate under Attack Strategy III. The detection rates under attack strategy III are shown in Figures 11 and 12. We observe that the Randomized Multicast, Line-selected Multicast, and Scheme II and III have almost no resilience to attack strategy II. This is because all of these schemes do not take the case that replica nodes generate falsified location claims into consideration. Once the adversary generates the falsified location claim that is identical to their origin compromised nodes, the multicast-family schemes will accept the replicas as benign nodes since the falsified location claim will bypass the confliction detection whereas Schemes II and III will detect the location anomaly of the falsified location claim and send them to their home group, the confliction detection will still accept these falsified claims since they are exactly the same as their original compromised nodes. Nevertheless, Scheme I can detect the replicas, which limits the damage of attack strategy III to some extents. This is because the trusted neighbor detection will block the suspicious replicas' communication. However, note that the detection of Scheme I will still be disabled when the adversary combines attack strategies I and III. In contrast, our proposed scheme reveals the significant advantage of resilience to attack strategy III since we use a neighboring-based location similarity estimation. For all of the network configurations, our scheme has at least 85% probability to detect the replicas which adopt attack strategy III. Furthermore, the detection rate becomes higher with the increase of distance D, as well as the decrease of the deployment standard deviation σ. The reason behind such a tendency is quite similar to the tendency of the detection rate under attack strategy II.

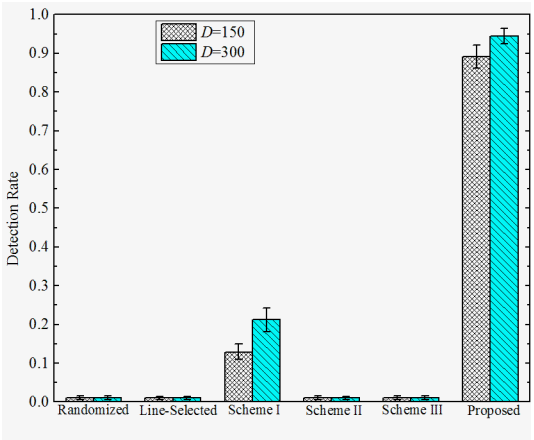

Figure 11. Detection rate of the proposed versus prior works under attack strategy III when $\sigma = 100$.

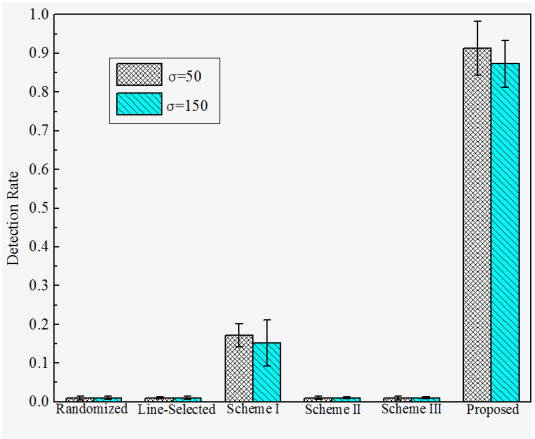

Figure 12. Detection rate of proposed versus prior works under attack strategy III when D = 200.

Comparison of Communication Overhead. We investigate the communication overhead of the proposed scheme with various number of total nodes in the network, and compare it with the previous

schemes. We set $P_a = 0.3$ and $P_s = 0.8$ while varying M from 2560 to 10,240. The corresponding simulation results are shown in Figure 13. As illustrated in Figure 13, the simulation results on the communication overhead are closely match our theoretical analysis in Section 6.1. The communication overhead of our scheme only grows at $O\left(\sqrt{M}\right)$ with the number of nodes M in the network. The value of communication overhead of our proposal falls in between that of Scheme II and Scheme III. When the number of nodes ranges from 2560 to 10,240, our scheme requires each node to transmit up to an average of 76.3628 packets. This is because, in our scheme, the nodes are required to reply the LAQ in addition to sending location claims. However it is worthwhile to point out that compared with the aforementioned Scheme II and Scheme III, in our scheme fewer sensor nodes are involved in the location claim transmission, which results in a lower value of total communication overhead for the entire network.

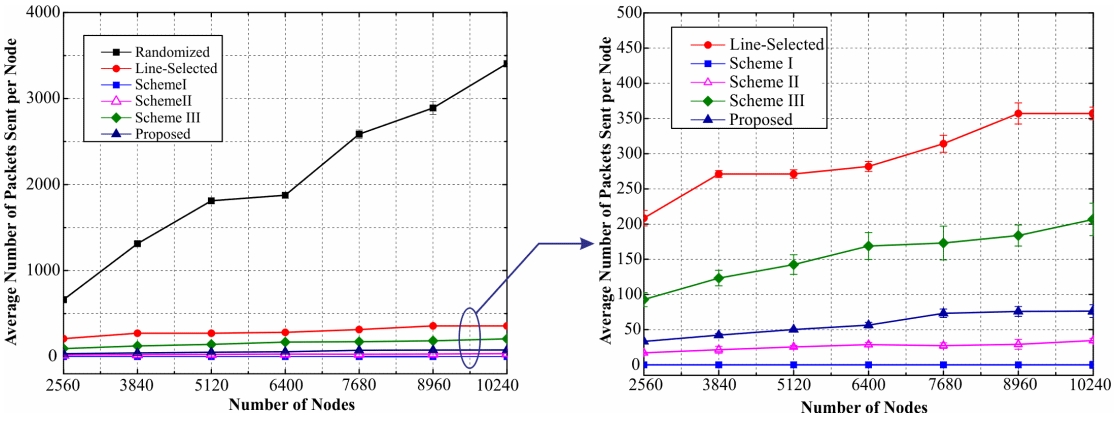

Figure 13. Communication overhead of the proposed and previous works.

7. Conclusions

We have proposed a collaborative replica detection scheme for sensor networks that takes advantages of location similarity estimation techniques to provide high resiliency to a variety of dangerous attack strategies. Our scheme detects the replicas by verifying the authenticity and consistency of the nodes' location claims using the neighborhood relationship derived from the group deployment knowledge. Specifically, we introduce the metric neighboring vector based location similarity (NV-LS) in this work to quantify the difference between the true and claimed location of each node in the network, and perform a threshold decision to find the replicas. Compared with the previous works, our scheme provides additional security services to prevent the replica nodes from falsifying their location claims without deploying resource-consuming localization algorithms on the resource-constraint sensor nodes.

Additionally we present heuristic analyses to evaluate the security strength against potentially effective attack strategies and show that the adversary's benefits are significantly limited by the constraint of the proposed scheme for all the presented attack strategies. Furthermore, we evaluate the effectiveness and efficiency of our approach through theoretical analysis and simulation experiments, and compare them with that of prior works. Both of the results demonstrate that our approach provides better security resilience against the presented types of attack strategies in terms of a higher detection rate with a reasonable increase of communication overhead, as well as lower costs of computation and storage overhead.

Acknowledgments: This work is supported by the National Natural Science Foundation of China for Youth (Grant No. 61602263), the Natural Science Foundation of Jiangsu Province for Youth (Grant No. BK20160916), the Natural Science Foundation of Jiangsu Province (Grant No. BK20151507), National Basic Research Program of China (973 Program) under Grants 2011CB302903, the Key Program of Natural Science for Universities of Jiangsu Province (Grant No. 10KJA510035), Sponsored by NUPTSF (Grant No. NY216020), NUIST, PAPD and CICAEET.

Author Contributions: All authors have contributed to the proposed work in various degrees. Chao Ding proposed the scheme, and presented part of the security and performance analysis. He also wrote the experiment code, conducted the experiments and evaluated the empirical performance of the proposed scheme. Lijun Yang presented part of the security analysis. Meng Wu amended the manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Fu, Z.; Wu, X.; Guan, C.; Sun, X.; Ren, K. Toward Efficient Multi-Keyword Fuzzy Search Over Encrypted Outsourced Data With Accuracy Improvement. *IEEE Trans. Inf. Forensics Secur.* **2016**, *11*, 2706–2716. [CrossRef]
2. Fu, Z.; Ren, K.; Shu, J.; Sun, X.; Huang, F. Enabling Personalized Search over Encrypted Outsourced Data with Efficiency Improvement. *IEEE Trans. Parallel Distrib. Syst.* **2016**, *27*, 2546–2559. [CrossRef]
3. Xie, S.; Wang, Y. Construction of Tree Network with Limited Delivery Latency in Homogeneous Wireless Sensor Networks. *Wirel. Pers. Commun.* **2014**, *78*, 231–246. [CrossRef]
4. Xia, Z.; Wang, X.; Sun, X.; Wang, Q. A Secure and Dynamic Multi-keyword Ranked Search Scheme over Encrypted Cloud Data. *IEEE Trans. Parallel Distrib. Syst.* **2015**, *27*, 340–352. [CrossRef]
5. Fu, Z.; Sun, X.; Liu, Q.; Zhou, L.; Shu, J. Achieving Efficient Cloud Search Services: Multi-keyword Ranked Search over Encrypted Cloud Data Supporting Parallel Computing. *IEICE Trans. Commun.* **2015**, *98*, 190–200. [CrossRef]
6. Parrno, B.; Perrig, A.; Gligor, V.D. Distributed detection of node replication attacks in sensor networks. In Proceedings of the IEEE Symposium on Security and Privacy, Oakland, CA, USA, 8–11 May 2005; pp. 49–63.
7. Zhu, B.; Addada, V.G.K.; Setia, S.; Jajodia, S.; Roy, S. Efficient Distributed Detection of Node Replication Attacks in Sensor Networks. In Proceedings of the Computer Security Applications Conference, Miami Beach, FL, USA, 10–14 December 2007; pp. 257–267.
8. Conti, M.; Pietro, R.D.; Mancini, L.V. A randomized, efficient and distributed protocol for the detection of nodes replication attacks in wireless sensor networks. In Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing, ACM MobiHoc, Montreal, QC, Canada, 9–14 September 2007; pp. 80–89.
9. Abinaya, P.; Geetha, C. Dynamic detection of node replication attacks using X-RED in wireless sensor networks. In Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES2014), Chennai, India, 27–28 February 2014; pp. 1–4.
10. Choi, H.; Zhu, S.; Potra, T.F.L. SET: Detecting node clones in sensor networks. In Proceedings of the Third International Conference on Security and Privacy in Communication Networks and the Workshops (SecureComm 2007), Nice, France, 17–21 September 2007; pp. 341–350.
11. Ho, J.-W.; Liu, D.; Wright, M.; Das, S. K. Distributed detection of replica node attacks with group deployment knowledge in wireless sensor networks. *Ad Hoc Netw.* **2009**, *7*, 1476–1488. [CrossRef]
12. Charikar, M. Similarity estimation techniques from rounding algorithms. In Proceedings of the Thirty-Fourth ACM Symposium on Theory of Computing, Montreal, QC, Canada, 19–21 May 2002; pp. 380–388.
13. Zanca, G.; Zorzi, F.; Zanella, A.; Zorzi, M. Experimental comparison of RSSI-based localization algorithms for indoor wireless sensor networks. In Proceedings of the Workshop on Real-World Wireless Sensor Networks, Glasgow, UK, 1 April 2008.
14. Sastry, N.; Shankar, U.; Wagner, D. Secure verification of location claims. In Proceedings of the ACM Workshop on Wireless Security, San Diego, CA, USA, 19 September 2003.
15. Farah, K.; Nabila, L. The MCD Protocol for Securing Wireless Sensor Networks against Nodes Replication Attacks. In Proceedings of the Proceedings of International Conference on Advanced Networking Distributed Systems and Applications, Bejaia, Algeria, 17–19 June 2014; pp. 58–63.
16. Guo, C.; Guo, S.; Yang, Y.; Fei, W. Replication attack detection with monitor nodes in clustered wireless sensor networks. In Proceedings of the 2015 IEEE 34th International Performance Computing and Communications Conference (IPCCC), Nanjing, China, 14–16 December 2015; pp. 1–8.
17. Ho, Y.S.; Ma, R.L.; Sung, C.E.; Tsai, I.C.; Kang, L.W.; Yu, C.M. Deterministic detection of node replication attacks in sensor networks. In Proceedings of the 2015 IEEE International Conference on Consumer Electronics, Taipei, Taiwan, 6–8 June 2015; pp. 468–469.

18. Douceur, J.R. The Sybil Attack. In Presented at the Revised Papers from the First International Workshop on Peer-to-Peer Systems, Cambridge, MA, USA, 7–8 March 2002.
19. Pecori, R. S-Kademlia: A trust and reputation method to mitigate a Sybil attack in Kademlia. *Comput. Netw.* **2016**, *94*, 205–218. [CrossRef]
20. Katz, J.; Lindell, A. Aggregate message authentication codes. In Proceedings of the Cryptographers' Track at the RSA Conference, San Francisco, CA, USA, 8–11 April 2008; pp. 155–169.
21. Chan, H.; Perrig, A.; Song, D. Random key predistribution schemes for sensor networks. In Proceedings of the IEEE Symposium on Research in Security and Privacy, Berkeley, CA, USA, 11–14 May 2003; pp. 197–213.
22. Eschenauer, L.; Gligor, V.D. A key-management scheme for distributed sensor networks. In Proceedings of the 9th ACM Conference on Computer and Communication Security, Washington, DC, USA, 18–22 November 2002; pp. 41–47.
23. Perrig, A.; Szewczyk, R.; Tygar, J.D.; Wen, V.; Culler, D.E. SPINS: Security protocols for sensor networks. *Wirel. Netw.* **2002**, *8*, 521–534. [CrossRef]
24. Lei, F.; Wenliang, D.; Peng, N. A beacon-less location discovery scheme for wireless sensor networks. In Proceedings of the IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies, Miami, FL, USA, 13–17 March 2005; pp. 161–171.
25. Yang, L.; Ding, C.; Wu, M. RPIDA: Recoverable Privacy-preserving Integrity-assured Data Aggregation Scheme for Wireless Sensor Networks. *KSII Trans. Internet Inf. Syst.* **2015**, *37*, 2808–2814.
26. Oliveira, L.B.; Aranha, D.F.; Gouvêa, C.P.L.; Scott, M.; Câmara, D.F.; López, J. TinyPBC: Pairings for authenticated identity-based non-interactive key distribution in sensor networks. *Comput. Commun.* **2011**, *34*, 485–493. [CrossRef]
27. Kumar, P.; Reddy, L.; Varma, S. Distance measurement and error estimation scheme for RSSI based localization in Wireless Sensor Networks. In Proceedings of the Fifth IEEE Conference on Wireless Communication and Sensor Networks (WCSN), Allahabad, India, 15–19 December 2009; pp. 1–4.
28. NIST. Special Publication 800-57: Recommendation for Key Management. Available online: http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf (accessed on 15 December 2015).
29. Diffie, W.; Hellman, M.E. *Exhaustive Cryptanalysis of the NBS Data Encryption Standard*; The Institute of Electrical and Electronics Engineering: Long Beach, CA, USA, 1977.
30. Levis, P.; Lee, N.; Welsh, M.; Culler, D. TOSSIM: Accurate and scalable simulation of entire TinyOS applications. In Proceedings of the 1st International Conference on Embedded Networked Sensor Systems, Los Angeles, CA, USA, 5–7 November 2003.