WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. (2012)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/wcm.2300

RESEARCH ARTICLE

Designing robust routing protocols to protect base stations in wireless sensor networks

Juan Chen[1], Hongli Zhang[1], Xiaojiang Du[2*], Binxing Fang[1] and Liu Yan[3]

[1] School of Computer Science and Technology, Harbin Institute of Technology, Harbin, 150001, China
[2] Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, U.S.A
[3] School of Astronautics, Harbin Institute of Technology, Harbin, 150001, China

ABSTRACT

A base station is the controller and the data-receiving center of a wireless sensor network. Hence, a reliable and secure base station is critical to the network. Once an attacker locates the base station, he or she can do many damages to the network. In this paper, we examine the base station location privacy problem from both the attack and defense sides. First, we present a new attack on base station: parent-based attack scheme (PAS). PAS can locate a base station within one radio (wireless transmission) range of sensors in high-density sensor networks. Different from existing methods, PAS determines the base station location on the basis of parent–child relationship of sensor nodes. Existing base station protection schemes cannot defend against PAS. Second, on the basis of PAS, we propose a two-phase parent-based attack scheme (TP-PAS). Our simulation results demonstrate that TP-PAS is able to determine the base station successfully in both low-density and high-density sensor networks. Then, to defend against PAS and TP-PAS, we design a child-based routing protocol and a parent-free routing protocol for sensor networks. Our theory analysis and experiment results show that the parent-free routing protocol has more communication cost and less end-to-end latency compared with the child-based routing protocol.

KEYWORDS
wireless sensor network; routing protocol; base station; security

***Correspondence**
Xiaojiang Du, Department of Computer and Information Sciences, Temple University, Philadelphia, PA 19122, U.S.A.
E-mail: dux@temple.edu

1. INTRODUCTION AND RELATED WORK

As an important part of the Internet of Things, wireless sensor networks (WSNs) are becoming increasingly popular with applications ranging from habitat monitoring to battle field. In sensor networks, sensor placement is often driven by the need to sense certain phenomena. Low-density sensor networks are suitable in circumstances with easy node replacement, while applications such as structural health monitoring require high-density deployments [1]. A sensor network with 40 or more neighbors per node is generally considered as a high-density sensor network [2].

Location privacy is an important security issues in WSNs. An effective location privacy preservation protocol for WSN can prevent attackers from identifying (and then capturing) important nodes (such as source and base station) by hiding their locations. Local passive attackers can locate a node by using localization techniques such as triangulation, angle of arrival, and signal strength [3]. Moreover, if an attacker knows the location of each node, he or she will be able to selectively compromise more important nodes, which will allow him or her to obtain much more information and/or cause more damages to the network.

Because of the important role of base station in WSNs, the location of base station is of critical importance. Existing base station location attacks include packet tracing attack [4], rate monitoring attack [4], and zeroing-in attack [5]. In [6], Deng *et al.* presented a few techniques to safeguard the base station against packet rate monitoring and time correlation attacks. A protection method called Differential Enforced Fractal Propagation (DEFP) with techniques of multi-path routing and fake message injection was proposed. However, these measures would take a long time to find the base station, as attackers concentrate on the traffic rates on different locations. In [7], Conner *et al.* proposed fake base station protocol for protecting the base station. The work creates a dummy base station away from the real base station. All data are first forwarded to the dummy base station. Then, the aggregated data are re-routed to the real base station. This scheme implicitly assumes that the fake base station is with powerful computation and storage ability. However, this may not be true in a homogenous network. Once an adversary destroys the fake base station and acquires its private information, he or she can track the real base station easily. Jian *et al.* [8] went further to design a new location privacy routing protocol with fake packet injection to provide path diversity and minimize the information that an adversary can deduce from the overheard packets about the direction towards the receiver. After that, Acharya and Younis [4] extended popular metrics for measuring anonymity to suit the unique characteristic of WSNs and presented two approaches for boosting the anonymity of the base station through packet re-transmission Base-station Anonymity increase through selective packet Re-transmission (BAR) and by repositioning the base station Relocation for Increased Anonymity (RIA). In BAR, a base station selectively transmits data packets that become forwarded through the network in order to confuse the adversary. Meanwhile, the RIA approach introduces the concept of dynamically relocating the base station in order to safeguard it. However, how to sense and measure the threat to the base station was not presented, which is very important and hard to address. In [5], Liu and Xu have investigated zeroing-in attacks that utilize hop counts and the packet time of arrival. A few adversaries observe the network metrics by eavesdropping the local communication and collectively determine the sink location by solving the least squares problem over the observations. Zeroing-in attack cannot be launched to routing protocols that do not use hop count information.

Existing routing protocols in WSNs can be mainly divided into location-based routing protocols [9,10] and parent-based routing protocols [11,12]. The former is not a secure protocol for base station protection, as each node knows the base station location. On the other hand, the latter is more secure as for base station protection because each node transmits messages according to its shortest hop count (the shortest hop between this node and the base station) instead of the base station location. In parent-based routing, each node has a parent set and transmits its message to one of its parent with a probability higher than 0.5. Hence, existing base station attack and protection schemes are mainly presented for the parent-based routing [4,6–8].

Different from prior work, we propose two base station location attack schemes: parent-based attack scheme (PAS) and two-phase parent-based attack scheme (TP-PAS). PAS and TP-PAS determine a base station by parent sets of some nodes at a few different spots, which to the best of our knowledge, cannot be defended by existing base station protection schemes. To defend against PAS and TP-PAS, two routing protocols, named child-based (CB) routing protocol and parent-free (PF) routing protocol, are then presented. Specifically, our contributions are mainly threefold:

- We introduce the base station attack scheme PAS. PAS determines a base station by parent sets of some nodes, which is different from a prior work. Theory analysis and experiment results show that PAS can locate a base station with the accuracy of one radio range in high-density sensor networks, sufficient to find the base station. However, simulation results demonstrate that PAS does not work well in low-density sensor networks.
- On the basis of PAS, we propose TP-PAS. Our experiment results show that TP-PAS is able to determine a base station successfully in both low-density and high-density sensor networks. Furthermore, TP-PAS outperforms PAS as for attack accuracy.
- To cope with PAS and TP-PAS, two routing protocols, CB and PF, are introduced. Our performance analysis shows that PF and CB can defend against PAS and TP-PAS, and have small communication and computation costs. Furthermore, CB and PF can defend against zeroing-in attack [5] because under CB and PF, nodes do not have hop count information. They can also be combined with some existing base station location protection schemes [6,8] to defend against packet tracing [4] and rate monitoring attacks [4]. Theory analysis and experiment results show that CB has less communication cost and more end-to-end latency compared with PF.

The rest of this paper is organized as follows. We give the network and attack model in Section 2. We discuss PAS in Section 3. Section 4 proposes another base station attack scheme TP-PAS. Sections 5 and 6 introduce CB and PF, respectively, to defend against PAS and TP-PAS. We evaluate the performance of CB and PF in Section 7. Finally, we draw our conclusion in Section 8.

2. NETWORK AND ATTACK MODEL

Our network model is the same as that in existing base station location protection routing protocols (e.g., [6,8]). The entire network consists of one base station and a large number of sensor nodes. Without loss of generality, we assume that sensor nodes are distributed uniformly throughout the network. The base station can be placed anywhere. A sensor has limited computation, power, and storage resources. The base station is not constrained in power, communication, and computation capabilities. We do not assume a specific medium access control protocol. Each sensor node has a transmission range R. If the distance between two sensor nodes is no more than R, the two nodes are neighbors, and they can communicate with each other directly. Each node has a parent set and transmits its message to one of its parents with a certain probability.

Next, we discuss the attack model. There may be multiple colluding adversaries in the network. An adversary may have more powerful hardware than a sensor. Specifically, an adversary may have the following capabilities:

- *Eavesdropping*. An adversary is able to receive messages sent by sensors within his or her monitoring range.
- *Active attacks*. An adversary can capture a sensor, compromise it, and then obtain all information stored in the sensor.
- *Node localization*. An adversary is able to estimate the location of a node, by using existing localization schemes, such as the angle of arrival and/or the signal strength [3].
- *Colluding*. Several adversaries may collude with each other to infer the base station location.

3. PARENT-BASED ATTACK SCHEME

In this section, we discuss PAS in details.

3.1. Overview of parent-based attack scheme

PAS determines the location of a base station by parent sets of some nodes. Let $R_{\text{opt}}(n_i)$ be the line passing through node n_i and the base station. For any two nodes, say, n_i and n_j, if $R_{\text{opt}}(n_i)$ and $R_{\text{opt}}(n_j)$ intersect, then the intersection is the location of the base station. Hence, by obtaining $R_{\text{opt}}(n_i)$ and $R_{\text{opt}}(n_j)$, an adversary can locate the base station. An adversary may find several locations close to $R_{\text{opt}}(n_i)$ and generate a fitted line that approximates $R_{\text{opt}}(n_i)$. More generally, if there are m ($m \geq 2$) adversaries, they can generate m fitted lines, compute the intersections, and then estimate the location of the base station from these intersections. Specifically, PAS consists of three steps:

(1) Location sampling. The ith ($1 \leq i \leq m$) adversary, say, $A_i \in \tilde{A}$, stays at a location close to node n_i. A_i tries to find h ($h \geq 1$) locations around $R_{\text{opt}}(n_i)$ via passive eavesdropping or active attacks (e.g., compromising the node) on some nodes.
(2) Line fitting. A_i performs a least squares linear regression and generates a best fit line for $h+1$ locations including the location of n_i and the h sampled locations obtained by step (1).
(3) Base station location estimation. The m adversaries place themselves at different spots. They each perform steps (1) and (2). After that, they generate m fitted lines and calculate the estimated location of the base station, referred to as the estimated base station location (EBSL).

3.2. Location sampling

The location sampling process is to find h locations close to $R_{\text{opt}}(n_i)$. Denote U as a set of node locations and denote (x_j, y_j) as the jth element (location) in U. Denote P_i as the set of n_i's parent nodes. First, we present a few definitions, lemmas, and theorems.

Definition 1. *Let* $\text{CM}(U) = (x, y)$*, where* x *and* y *are computed by Equations* (1) *and* (2)*, respectively.*

$$x = \left(\frac{1}{|U|}\right) \sum_{j=1}^{|U|} x_j \tag{1}$$

$$y = \left(\frac{1}{|U|}\right) \sum_{j=1}^{|U|} y_j \tag{2}$$

Definition 2. $\text{Node}(f)$ *is a node placed at location* f.

Definition 3. $\text{NodeSet}(U)$ *is a node set where each node is placed at a distinct location in* U*, and* U *is the location set.*

Definition 4. *Define* $f_{\text{key}}(n_i, h)$ *as the* h*th* ($h \leq h_i$)*-order critical location of node* n_i*, where* h_i *denotes the shortest hop count between* n_i *and the base station. Denote* $L^{(i)}_{\text{parent}}$ *as the set of locations of* n_i*'s parent nodes.*

(1) If $h = 1$, $f_{\text{key}}(n_i, h)$ is the location in $L^{(i)}_{\text{parent}}$, which is closest to CM $\left(L^{(i)}_{\text{parent}}\right)$.
(2) If $h \geq 2$, $f_{\text{key}}(n_i, h)$ is the first-order critical location of $\text{Node}(f_{\text{key}}(n_i, h-1))$.

Definition 5. *Let* $f_{\text{cm}}(n_i, h)$ *be the* h*th-order barycenter (center of mass) location of node* n_i.

(1) If $h = 1$, $f_{\text{cm}}(n_i, h)$ is CM $\left(L^{(i)}_{\text{parent}}\right)$.
(2) If $h \geq 2$, $f_{\text{cm}}(n_i, h)$ is the first-order barycenter location of node $\text{Node}(f_{\text{key}}(n_i, h-1))$.

Definition 6. *Define set* $F_{\text{cm}}(n_i, h) = \{f_{\text{cm}}(n_i, j) | 1 \leq j \leq h\}$.

Definition 7. *Define set* $F_{\text{key}}(n_i, h) = \{f_{\text{key}}(n_i, j) | 1 \leq j \leq h\}$.

Theorem 1. *In a sensor network where nodes are uniformly distributed,* $f_{\text{cm}}(n_i, 1)$ *is close to* $R_{\text{opt}}(n_i)$*; as the node density increases,* $f_{\text{cm}}(n_i, 1)$ *becomes closer to* $R_{\text{opt}}(n_i)$.

Proof. As shown in Figure 1, several circles with different radii, say, R, $2R$, and $3R$, are centered at the base station. The qth annulus is the area between the $(q-1)$th and qth circles. We have that nodes in the qth annulus are q hops away from the base station, where $q = 2, 3, 4, \ldots$. Let node n_i be in the $(q+1)$th annulus. Thus, P_i is in the qth annulus and is within the transmission range of n_i. P_i is in the dotted area in Figure 1. Because nodes are

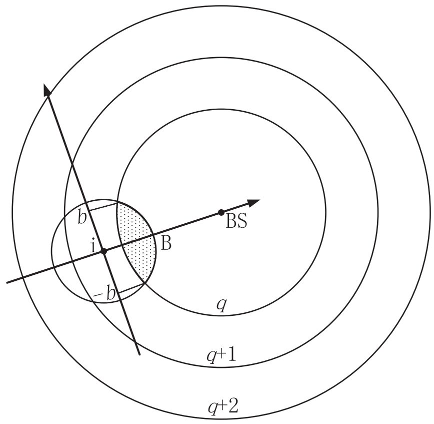


Figure 1. The area of n_i's parent nodes. BS, base station.

placed uniformly in the entire network, n_i's parents are also uniformly distributed on both sides of $R_{\text{opt}}(n_i)$. By Definition 5, we have that the y-coordinate of $f_{\text{cm}}(n_i,1)$ is $\bar{y} = (1/w)\sum_{j=1}^{w} y_j$, where w is the number of n_i's parents and y_j is the y-coordinate of the jth parent. As shown in Figure 1, we set up a Cartesian coordinate plane with the origin at node n_i, and the two axis lines are $R_{\text{opt}}(n_i)$ and a line perpendicular to $R_{\text{opt}}(n_i)$. Let the y-coordinates of nodes in the parent area range from $-b$ to b. Then, $y_1, y_2, \ldots, y_w$ are independent random variables following the uniform distribution in $[-b, b]$. Hence, we have the expectation of $y_j - \mathrm{E}(y_j) = 0$, for $1 \le j \le w$. According to the law of large numbers, for any $\varepsilon > 0$, we have

$$\lim_{w \to +\infty} p\left\{\left|\frac{1}{w}\sum_{j=1}^{w} y_j\right| < \varepsilon\right\} = 1 \tag{3}$$

When w becomes large, the average of y_j converges to the expected value 0 with probability 1. This means that $f_{\text{cm}}(n_i,1)$ is close to the line $R_{\text{opt}}(n_i)$. Furthermore, we have $w \propto \rho$, where ρ denotes the node density. Hence, as the node density increases, w also increases, and $f_{\text{cm}}(n_i,1)$ becomes closer to the line $R_{\text{opt}}(n_i)$.

□

Lemma 1. *In sensor networks with nodes uniformly distributed, locations in $F_{\text{cm}}(n_i, h)$ are close to $R_{\text{opt}}(n_i)$, and they become closer to $R_{\text{opt}}(n_i)$ as ρ increases.*

Proof.

(1) When $h = 1$, according to Theorem 1, $f_{\text{cm}}(n_i,1)$ is close to $R_{\text{opt}}(n_i)$, and $f_{\text{cm}}(n_i, 1)$ becomes closer to $R_{\text{opt}}(n_i)$ as ρ increases. Hence, Lemma 1 is true when $h = 1$.

(2) Assume when $h = j$ $(1 \le j \le h_i)$, where h_i denotes the shortest hop count between n_i and the base station, Lemma 1 is true. We have the following: $f_{\text{cm}}(n_i, j)$ is closer to $R_{\text{opt}}(n_i)$ as ρ increases. By Definition 4, we have that $f_{\text{key}}(n_i, j)$ is the location of the node, which is $\text{Node}(f_{\text{key}}(n_i, j-1))$'s parent and is closest to $f_{\text{cm}}(n_i, j)$. Hence, $f_{\text{key}}(n_i, j)$ is closer to $R_{\text{opt}}(n_i)$ as ρ increases. Let l be the line passing through $f_{\text{key}}(n_i, j)$ and the base station. Then, l approximates $R_{\text{opt}}(n_i)$ as ρ increases. Because $f_{\text{cm}}(n_i, j+1)$ is the first-order barycenter location of $\text{Node}(f_{\text{key}}(n_i, j))$, according to Theorem 1, we have that $f_{\text{cm}}(n_i, j+1)$ is close to l and $f_{\text{cm}}(n_i, j+1)$ becomes closer to l with increasing ρ. Thus, $f_{\text{cm}}(n_i, j+1)$ becomes closer to $R_{\text{opt}}(n_i)$ as ρ increases. Lemma 1 is true when $h = j+1$. Hence, the locations in $F_{\text{cm}}(n_i, j+1)$ becomes closer to $R_{\text{opt}}(n_i)$ as ρ increases.

□

Theorem 2. *By passively monitoring (and/or actively compromising) node n_i, an adversary can find $f_{\text{key}}(n_i, 1)$ and $f_{\text{cm}}(n_i, 1)$.*

Proof. By passively monitoring node n_i for enough time, an adversary can capture messages from both n_i and its neighbors, and infer their relationships and find out P_i. Then, he or she can locate nodes in P_i by some existing localization techniques, such as the angle of arrival technique in [6]. If a routing protocol is combined with security schemes such as fake-message injection [12], it is infeasible for a passive adversary to find out P_i as he or she cannot distinguish real messages from fake ones. In that case, an adversary may launch active attacks on node n_i and then obtain its secret information including P_i and the keys. After that, he or she can locate n_i's parents by angle of arrival [6]. With locations of n_i's parents, the adversary can obtain $f_{\text{key}}(n_i,1)$ and $f_{\text{cm}}(n_i,1)$. □

Lemma 2. *By monitoring or compromising node n_i and* $\text{NodeSet}(F_{key}(n_i, h-1))$*, an adversary can find* $F_{\text{cm}}(n_i, h)$.

Proof.

(1) When $h = 1$, according to Theorem 2, an adversary can find $f_{\text{cm}}(n_i, 1)$ by monitoring or compromising node n_i.

(2) When $h \ge 2$, by Definitions 4 and 5, we have that $f_{\text{cm}}(n_i, h)$ is the first-order barycenter location of $\text{Node}(f_{\text{key}}(n_i, h-1))$. Therefore, an adversary can find $f_{\text{cm}}(n_i, h)$ by monitoring or compromising node $\text{Node}(f_{\text{key}}(n_i, h-1))$ by Theorem 2.

According to Lemma 1, we have that locations in $F_{\text{cm}}(n_i, h)$ $(1 \le h \le h_i h_i)$ are close to $R_{\text{opt}}(n_i)$, where h_i denotes the hop count of node n_i. The location sampling process is completed if an adversary obtains $F_{\text{cm}}(n_i, h)$. By Lemma 2, we have that an adversary can find $F_{\text{cm}}(n_i, h)$ by monitoring or compromising n_i and $\text{NodeSet}(F_{\text{key}}(n_i, h-1))$. □

3.3. Line fitting

By the aforementioned location sampling process, the adversary A_i obtains U_i that includes h sampled locations and the location of n_i. After that, A_i performs a least squares linear regression and generates a best fit line, say, $l_i : y = ax + b$, for locations in U_i, where a and b are computed by Equations (4) and (5), respectively. $(x_{i,j},\ y_{i,j})$ denotes the jth element in U_i. By Lemma 1, locations in U_i are close to $R_{\text{opt}}(n_i)$; hence, l_i is close to $R_{\text{opt}}(n_i)$.

$$a = \frac{\left(\sum_{j=1}^{h+1} x_{i,j} \sum_{j=1}^{h+1} y_{i,j} - (h+1) \sum_{j=1}^{h+1} x_{i,j} y_{i,j}\right)}{\left(\sum_{j=1}^{h+1} x_{i,j} \sum_{j=1}^{h+1} x_{i,j} - (h+1) \sum_{j=1}^{h+1} x_{i,j}^2\right)} \tag{4}$$

$$b = \frac{\left(\sum_{j=1}^{h+1} x_{i,j} y_{i,j} \sum_{j=1}^{h+1} x_{i,j} - \sum_{j=1}^{h+1} y_{i,j} \sum_{j=1}^{h+1} x_{i,j}^2\right)}{\left(\sum_{j=1}^{h+1} x_{i,j} \sum_{j=1}^{h+1} x_i - (h+1) \sum_{j=1}^{h+1} x_{i,j}^2\right)} \tag{5}$$

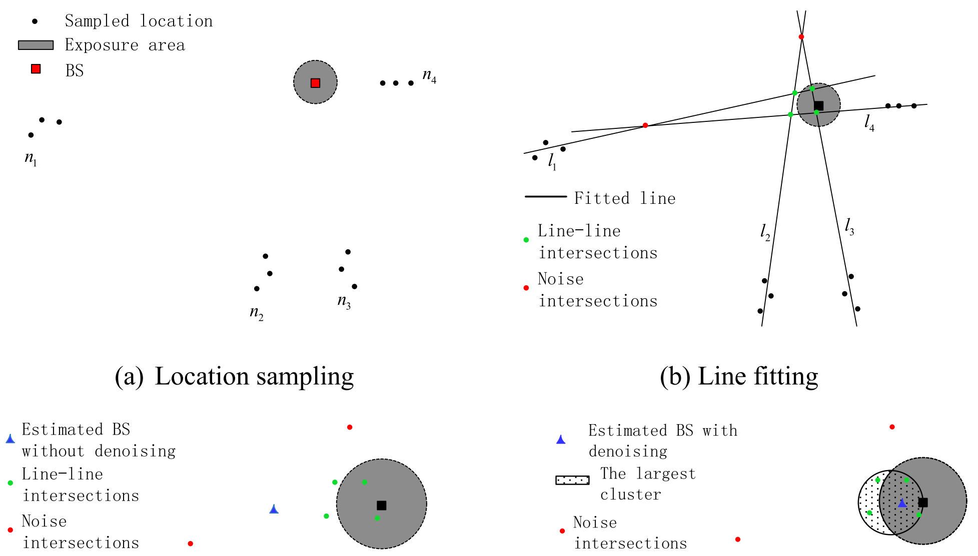
3.4. Estimation of base station location

If there are m adversaries and each of them performs the location sampling and line fitting processes, then they can obtain m lines: $L = \{l_i | 1 \le i \le m\}$. Let an estimation point be the intersection of two lines in L. Suppose we have k $\left(k \le c_m^2\right)$ estimation points from L, where c_m^2 denotes the number of two combinations from m elements. It is possible that some estimation points (called noise points) are far away from the base station. There are two reasons for having noise points. (1) If the node density ρ is very low, for an adversary A_i, one or two of his or her sampled locations might be away from $R_{\text{opt}}(n_i)$; thus, l_i is also away from the base station, which causes some intersections of l_i being far away from the base station. (2) Two or more lines in L are nearly parallel. For example, if $R_{\text{opt}}(n_i)$ and $R_{\text{opt}}(j)$ are nearly parallel to each other, then l_i and l_j are nearly parallel, and they will have no intersections or their intersections are far away from the base station. Let S be the set of the k estimation points. PAS can reduce the number of noise points in S by clustering and can then obtain a more accurate location of the base station [13]. The de-noising process is as follows:

(1) Apply hierarchical clustering [13] on S and generate k' clusters with a given threshold.
(2) Find the maximum cluster, say, $c_{\max}$, which includes the largest number of estimation points.
(3) The EBSL is CM($c_{\max}$).

3.5. An example

Figure 2 presents an example of PAS. We assume that if an adversary is in the exposure area (shaded region in Figure 2(a)), he or she can find the base station [5]. We can see from Figure 2(a) that four adversaries lie close to nodes n_1, n_2, n_3, and n_4, respectively. They obtain $f_{\text{key}}(n_i, j)$ and $f_{\text{cm}}(n_i, j)$ $(1 \le i \le 4,\ 1 \le j \le 2)$ by the location



(a) Location sampling

(b) Line fitting

(c) base station estimation without denoising

(d) base station estimation with denoising

Figure 2. Parent-based attack scheme procedure. BS, base station.

sampling process introduced in Section 3.2. Then, each adversary performs a least squares linear regression and generates a fitted line as is shown in Figure 2(b). After that, they compute the EBSL by intersections from these lines. However, the EBSL in Figure 2(c) is far away from the real base station location. This is due to some noise points discussed in Section 3.4. So if some noise points are removed by clustering, the EBSL is well estimated as we can see in Figure 2(d).

3.6. The effectiveness of parent-based attack scheme

We use the mean error Δd and the mean square error $\Delta\delta$ to evaluate the performance of PAS. Δd and $\Delta\delta$ are computed by Equations (6) and (7), and they are used to measure the attack accuracy. In Equations (6) and (7), e is the number of attacks and d_i denotes the difference between the EBSL and the actual base station location during the ith attack. Δd and $\Delta\delta$ are divided by the communication range R as in most existing localization works (e.g., [4,14]).

$$\Delta d = \frac{1}{(e * R) \sum_{i=1}^{e} |d_i|} \tag{6}$$

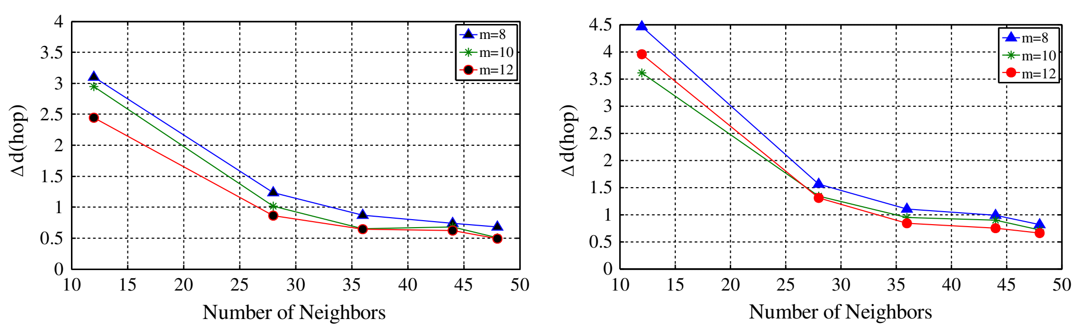
$$\Delta\delta = \frac{1}{\left(R * \sqrt{e * \sum_{i=1}^{e} (d_i - \Delta d)^2}\right)} \tag{7}$$

The effectiveness of PAS is validated by an event-driven sensor network simulator written in C++. For uniform sensor deployment, we divide the monitored area into small grids and place one node in a grid. To be more realistic, each node is not placed exactly in the center of a grid. For example, if (x, y) is the center of a grid, a sensor node is placed at $(x+\varepsilon, y+\varepsilon')$, where ε and ε' are two uniform random variables on $(-0.5, 0.5)$. The base station is randomly placed in the network. The following results are averaged over 100 runs.

PAS is evaluated for both parent-based routing and tree-based broadcast routing. Tree-based broadcast routing is a special kind of parent-based routing, and each node has only one parent instead of a parent set. For tree-based broadcast routing, a base station broadcasts a message, and each node, say, n_i, determines its parent to be one of its neighbors, which is the first to transmit the broadcasting message with a hop count less than n_i [4]. Note that in tree-based broadcast routing, both $f_{\mathrm{cm}}(n_i, h)$ and $f_{\mathrm{key}}(n_i, h)$ are n_i's parent locations.

Our simulation uses a sensor network of 1024 nodes with $h = 1$, and the clustering threshold η is chosen as $2.5R$. The mean errors for parent-based routing and tree-based broadcast routing of PAS are shown in Figure 3(a,b), respectively, where the x-axis is the average number of neighbors of each node and m is the number of adversaries in the network. Figure 3(a,b) shows that as the number of adversary increases, the mean error decreases. Also, the mean error decreases when the number of neighbors increases. This is consistent with Lemma 1. As can be seen, the mean error is reduced when the average number of neighbors increases, which also follows Lemma 1. When the average number of neighbors is over 36, adversaries can locate the base station with an accuracy of one radio range by passively monitoring or actively compromising eight nodes for parent-based routing and 10 nodes for tree-based broadcast routing. However, the situation is different in low-density networks. When the average number of neighbors is as low as 12, the attack accuracy is still not good even by passively monitoring or compromising 12 nodes.

Figure 4 shows the mean error for varying the network size (number of sensors) with $n = 36$, $h = 1$, $\eta = 2.5R$, and $m = 12$, where n denotes the average number of neighbors. As the network size grows, we notice that the mean error increases in general. It is also observed that the mean error increases significantly when the network size is more than 1024. Furthermore, when the network size is more than 1444, the mean error is tending towards stability.



(a) (b)

Figure 3. Mean error versus number of neighbors and adversaries: (a) parent-based routing and (b) tree-based broadcast routing.

DOI: 10.1002/wcm

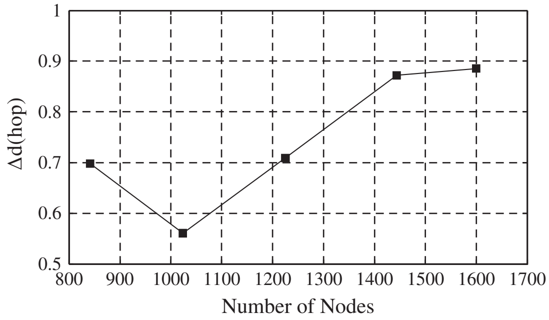


Figure 4. Mean error versus network size.

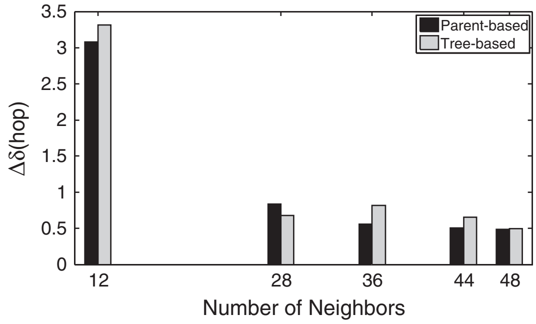


Figure 5. Mean square error versus number of neighbors.

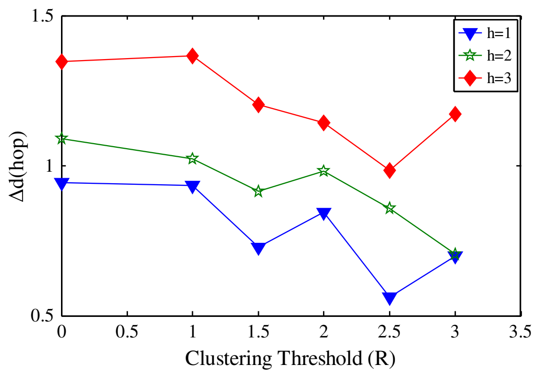


Figure 6. Mean error versus clustering threshold.

Figure 5 shows the mean square error in parent-based routing protocols and tree-based broadcast routing protocol for varying number of neighbors with $N = 1024$, $h = 1$, $\eta = 2.5R$, and $m = 12$. It is observed that the larger the number of neighbors, the less the mean square error, which indicates that PAS is more robust when the number of neighbors is large. Furthermore, parent-based routing also shows a lower $\Delta\delta$ when compared with tree-based broadcast routing. Thus, PAS is more robust for parent-based routing than for tree-based broadcast routing.

Figure 6 shows the mean error for varying η and h in parent-based routing. In this simulation, the parameters are set as follows: $N = 1024$, $n = 36$, and $g = 12$, where g denotes the total number of nodes that has been attacked, and $g = m^*h$. Figure 6 shows that Δd decreases when h becomes smaller, which indicates that given a fixed total number of nodes that have been attacked (i.e., given g), the attack accuracy is high even if each adversary only attacks a small number of nodes. In addition, the results also show that Δd has the lowest value. Note that $\eta = 0$ means PAS without clustering.

To sum up, the aforementioned simulation results show that PAS can locate the base station with high accuracy (e.g., within one radio range) by attacking only a small number of nodes (e.g., eight nodes) in high-density networks. However, PAS does not work well in low-density networks.

4. TWO-PHASE PARENT-BASED ATTACK SCHEME

In this section, we discuss TP-PAS in details.

4.1. two-phase parent-based attack scheme

Experiment results in Section 3.6 illustrate that PAS does not work well in low-density networks. Therefore, on the basis of PAS, we propose TP-PAS. Our simulations show that TP-PAS can locate a base station successfully (within one radio range) by passively monitoring or actively compromising 10 nodes in both low-density and high-density networks. Furthermore, TP-PAS outperforms PAS in attack accuracy. Specifically, the main idea of TP-PAS is as follows:

(1) **First phase**: Each of m adversaries obtains $h_1 + 1$ locations and then generates a fit line by the location sampling and line fitting processes. Hence, m adversaries obtains m lines and calculates k line–line intersections.
(2) **Second phase**: Adversaries find the most closest m' intersections from k intersections, where $m' \leq m$. We use N_i to denote the sensor node that is close to the ith of m' intersections. Then, m' adversaries move close to m' sensor nodes, say, $\{N_i | 1 \leq i \leq m'\}$, and launch PAS attack again.

4.2. Experiment comparison

Experiment environment here is the same as that introduced in Section 3.6. Let h_1 and h_2 be the number of sampled locations discussed in two phases, respectively. Both Δd and $\Delta\delta$ show the same change of trends with different parameters for both parent-based routing and tree-based broadcast routing in Section 3.6. We therefore only consider the attack ability of TP-PAS for parent-based routing in this section.

The mean error and mean square error over different numbers of neighbors for PAS and TP-PAS are shown in Figure 7(a,b), respectively. In this simulation, the

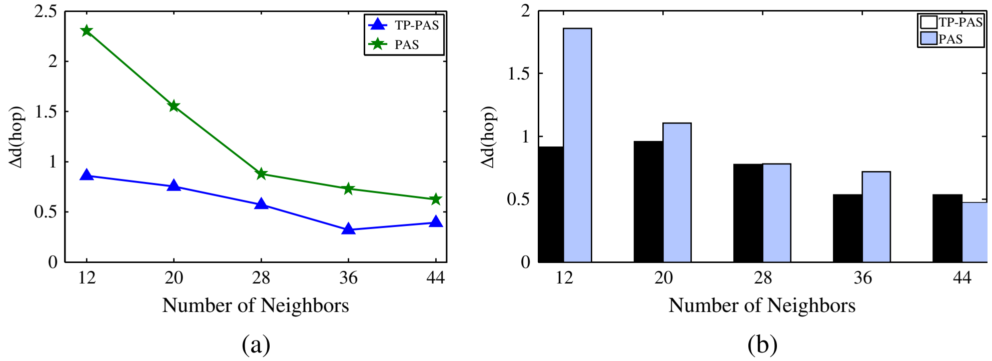


Figure 7. Attack ability over different number of neighbors: (a) mean error versus number of neighbors and (b) mean square error versus number of neighbors. TP-PAS, two-phase parent-based attack scheme; PAS, parent-based attack scheme.

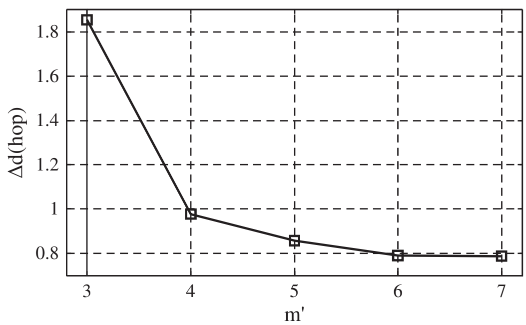


Figure 8. Mean error versus number of compromised nodes.

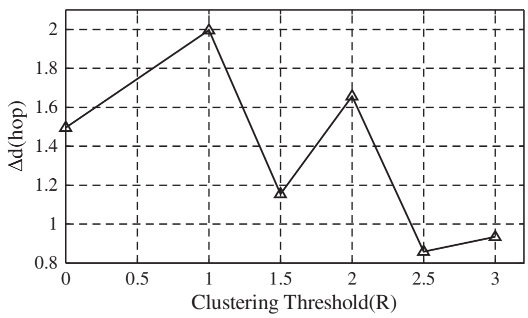


Figure 9. Mean error versus clustering thresholds.

parameters are set as follows: $N = 841$, $n = 12$, $h_1 = h_2 = 1$, $\eta = 2.5R$, $m = 5$, and $m' = 5$. Figure 7(a) illustrates that TP-PAS can locate a base station within one radio range in both low-density and high-density networks. And results from Figure 7(a) also show that the mean error decreases as node density increases in both PAS and TP-PAS. Figure 7(b) shows that TP-PAS becomes more robust as node density increases. More importantly, we can see from these two figures and conclude that TP-PAS outperforms PAS in attack ability.

We also study the attack accuracy over different number of nodes being attacked with $N = 841$, $n = 12$, $h_1 = h_2 = 1$, $\eta = 2.5R$, and $m = 5$. It can be observed from Figure 8 that the mean error decreases as the number of nodes being attacked in the second phase increases. It is true that the attack ability improves with more adversaries. Figure 9 shows the mean error for varying η with $N = 841$, $n = 12$, $h_1 = h_2 = 1$, $m = 5$, and $m' = 5$. It can be observed that Δd reaches the lowest value when $\eta = 2.5R$.

We can conclude from Figures 7–9 that TP-PAS works well in both high-density and low-density networks. And we also note that TP-PAS outperforms PAS in attack ability.

5. CHILD-BASED ROUTING PROTOCOL

We propose the CB, which is robust to both PAS and TP-PAS attacks for two reasons: first, each node stores a child set instead of a parent set, and it only transmits messages coming from its children; and second, whenever a node transmits or receives a message, it updates its or its neighbors' broadcast keys. Therefore, adversaries cannot infer the parent set of each node as they cannot differentiate transmission relationship between two nodes. CB can also defend against zeroing-in attack [3], and it is easy to combine CB with existing base station location protection schemes [6,8] to protect a base station from typical packet tracing and rate monitoring attacks. Specifically, CB consists of two stages: network initialization and message sending.

5.1. Network initialization

We assume the network is secure (e.g., no attacks) for a short time after sensor nodes are deployed. During this period, the communications among sensor nodes are secure. Each sensor node, say, n_i, is preloaded with a hashing function H and two pairwise keys k_i and $k_{i,\mathrm{BS}}$. k_i is n_i's broadcast key, which is shared between n_i and its neighbors. And $k_{i,\mathrm{BS}}$ is shared between n_i and the base

Algorithm 1 Neighboring Parents Generation
Input: n_i, P_i
Output: $P_i^{'}$

1. $P_i^{'} = \Phi$;
2. n_j is chosen from P_i randomly;
3. **do**
4. $P_i = P_i - \{n_j\}$;
5. $P_i^{'} = P_i^{'} \cup n_j$;
6. $\check{N} = N_{nei}^{a_1} \frown N_{nei}^{a_2} \ldots \frown N_{nei}^{a_u}$; // $P_i^{'} = \{a_1, a_2, \ldots a_u\}$ and $u = |P_i^{'}|$

Figure 10. Neighboring parents generation algorithm.

station. Node n_i is also preloaded with four random numbers: θ $(0 < \theta < 1)$, h_{fake}, node ID $- i$, and γ_i, where θ denotes the probability for fake message generation, h_{fake} denotes the transmission times of a fake message, and γ_i is a random number used for broadcast key updating.

The purpose of the initialization stage is to find a child set $\acute{L}_i$ for sensor node n_i, such that if $\acute{L}_i \cap \acute{L}_j \neq \Phi$, then n_i and n_j are neighbors. Thus, only neighboring nodes might share one or more child nodes. Child set is used to avoid duplicate message transmission. For example, if n_i and n_j share a common child n_r, n_i and n_j must be neighbors. Once n_j transmits a message from n_r, it will be heard by n_i; thus, n_i will not retransmit this message again. During the initialization stage, the base station first broadcasts a message. After the broadcast process, n_i obtains the following information: the hop count h_i, the parent set P_i, and the random numbers of its neighbors for broadcast key updating. Then, n_i finds a subset $P_i' \subset P_i$ such that nodes in P_i' are neighbors. After that, n_i requests to be a child of the nodes in P_i' by sending a request message M_i^{REQ} with the form $REQ||P_i'||i$. Lastly, each node in P_i' adds n_i to its child set.

As stated earlier, $\acute{L}_i$ is obtained by P_i'. Thus, the key point is how to find P_i'. We use a neighboring parents generation algorithm to generate P_i' as shown in Figure 10. Node n_i firstly chooses a neighbor, say, n_j, from P_i randomly, adds n_j to P_i', and sets $P_i = P_i - \{n_j\}$. Then, n_i tries to find the next node, say, n_r, from P_i such that compared with the other nodes in P_i, N_{nei}^r has the most common nodes with $\check{N}$, where $\check{N} = N_{\text{nei}}^{a_1} \cap N_{\text{nei}}^{a_2} \cap \cdots \cap N_{\text{nei}}^{a_u}$, $P_i' = \{a_1, a_2, \ldots a_u\}$, and N_{nei}^r denotes n_r's neighbors. Node n_i repeats the node selection process until it cannot find a node in P_i such that the node is a neighboring node of each node in P_i'.

5.2. Message sending

If source node n_i wants to send a message M_i to the base station, it broadcasts M_i to its neighbors with the form $i||E_{k_i}(\text{TRUE}||E_{k_{i,\text{BS}}}(data))$, where TRUE denotes that M_i is a real message. For $\forall n_j \in N_{\text{nei}}^i$, if n_j receives M_i and $n_i \in \acute{L}_j$, n_j decrypts M_i by k_i. Then, n_j keeps M_i for a random delay t $(t \leq \delta)$ before it transmits M_i with the form $M_j = j||E_{k_j}(\text{TRUE}||E_{k_{i,\text{BS}}}(data))$. However, if some other node has transmitted M_i within t, n_j discards M_i. If n_j receives M_i and $n_i \notin \acute{L}_j$, n_j generates a fake message $M_j^{\text{fake}} = j||E_{k_j}(\text{FAKE}||h_{\text{fake}}||n_r||PA)$ with probability θ and sends it to one node, say, n_r, in $N_{\text{nei}}^j - \acute{L}_j$, where FAKE denotes that M_j^{fake} is a fake message, h_{fake} is the max transmission times, and PA is the padding part. If n_r receives M_j^{fake}, n_r updates h_{fake} with $h_{\text{fake}} - 1$. If $h_{\text{fake}} \neq 0$, n_r transmits M_j^{fake} to one node in $N_{\text{nei}}^r - \acute{L}_r$. In order to conceal the real messages by fake messages whenever a node, say, n_i, transmit a real or fake message, both n_i and its neighbors update k_i by Equation (8). By doing this, an adversary cannot differentiate the real messages from the fake ones as he cannot decrypt messages without previous pairwise keys.

$$k_i = H(k_i \oplus \gamma_i) \tag{8}$$

5.3. Performance analysis

In this section, we analyze the performance of CB. We discuss the communication cost, computation cost, transmission latency, and security performance in Sections 5.3.1–5.3.3, respectively.

5.3.1. Communication cost.

The communication cost is the total number of transmissions of a process. The communication cost of CB includes the message transmissions during the network initialization phase and the message sending phase. Note that we do not include the communication cost of the initial broadcasting because it is the same as of other existing routing protocols (e.g., [15,16]). After the broadcast process, each node, say, n_i, sends a child request message to P_i', and the communication cost for this is N.

In the message sending stage, the communication cost is caused by real and fake message transmissions. For each source node n_i, it sends a real message to the base station through the shortest path routing; hence, the communication cost is h_i. In addition, some fake messages are generated, and the communication cost is no more than $n^* h_i^* \theta^* h_{\text{fake}}$. This is because for any node n_j, if n_j receives a real message from node n_i and $n_i \notin \acute{L}_j$, then n_j generates a fake message with probability θ. Thus, the upper bound on communication cost in the message sending stage is $h_i + n^* h_i^* \theta^* h_{\text{fake}}$.

5.3.2. Computation cost versus transmission latency.

CB is a lightweight routing protocol because it only uses hashing functions and symmetric cryptography. For a real or fake message reception, each node needs one hashing operation for broadcast key updating and one decryption operation. In addition, each node also needs one hashing operation for broadcast key updating and one encryption operation for real or fake message forwarding.

The transmission latency is evaluated by the transmission times of a message before it reaches the base station. The upper bound on transmission latency of a message from source node n_i to the base station is $\delta^* h_i$.

5.3.3. Security performance.

CB is robust to both PAS and TP-PAS, as adversaries cannot obtain the parent set of any node. An adversary may lie close to node n_i and monitor and obtain messages that come from n_i and its neighbors. However, he or she cannot infer and obtain P_i. This is because whenever n_i sends a real message, each node in $\overline{P_i'}$ generates a fake message with probability θ, where $\overline{P_i'} = N_{\text{nei}}^i - P_i'$. And he or she cannot differentiate fake messages from real ones. Then, he or she may compromise node n_i and obtains n_i's private information. However, he or she still cannot differentiate fake messages from real ones because he or she is not able to decrypt messages without previous broadcast keys. This is because whenever a message is transmitted by a node, say, n_j, k_j is updated by both n_j and its neighbors immediately. Therefore, the adversary cannot obtain P_i.

6. PARENT-FREE ROUTING PROTOCOL

As PAS is based on parents' locations, it will be infeasible for an attacker to find out the base station location if no sensor stores its parents' information. From the aforementioned principle, we propose a parent free (PF) routing protocol to defend against PAS attack. The main idea of PF is as follows. Each node, say, n_i, has u onion packets, each of which denotes a route from n_i to the base station. Node n_i sends messages to the base station by onion packets. As node n_i has no information about its parents, an adversary cannot find out n_i's parents by compromising n_i. Furthermore, in PF, two successive nodes in a route may not be parent–child, that is, the next forwarding node may not be the parent of the previous one in a route. Therefore, even if an adversary finds out that a message has been transmitted from one node to another, he is not sure whether the latter is the parent of the former. Hence, PF can defend against PAS attack. PF consists of two phases: network initialization and message sending. We present the details of PF as follows.

6.1. Network initialization

Assume the network is secure (e.g., no attacks) for a short time after sensor nodes are deployed. This is a common assumption used by several literatures (e.g., [3]). During this period, the communications among sensor nodes are secure. Before deployment, each node n_i is preloaded with several parameters: node ID $- i$, keys k_i, and $k_{i,\text{BS}}$. k_i is n_i's broadcast key, which is shared between n_i and its neighbors. And $k_{i,\text{BS}}$ is shared between n_i and the base station. After deployment, the base station generates and then sends u onion packets to n_i by the following two steps:

(1) Topology discovery. The base station first sends out a broadcast message to all nodes in the network. When each node receives the broadcast message, it updates the hop count and also includes the following in the message: its broadcast key, parent set P_i, and non-parent set $\overline{P_i}$ $\left(\overline{P_i} = N_{\text{nei}}^i - P_i\right)$..., where N_{nei}^i denotes the neighboring nodes of n_i. After the broadcast, each node (say, n_i) obtains the aforementioned information from its neighbors. Then, n_i sends P_i and $\overline{P_i}$ to the base station. Thereafter, each node deletes P_i and $\overline{P_i}$.

(2) Onion packets generation. For each node, say, n_i, the base station generates u onion packets $R_i = \left\{r_i^{(1)}, r_i^{(2)}, \ldots, r_i^{(u)}\right\}$ and sends R_i to n_i. For a route $a \to b \to \cdots \to \text{BS}$, $r_i^{(v)}$ $(1 \le v \le u)$ has the following form: $E_{k_{a,\text{BS}}}(a||E_{k_{b,\text{BS}}}(b||\ldots)||PA)$. Specifically, $r_i^{(v)}$ is computed as follows:

- **Route discovery**. First, n_i is chosen as the current node. Then, the base station selects the first node in route $r_i^{(v)}$, say n_j, from P_i and $\overline{P_i}$ with probability p and $1-p$, respectively. Next, n_j is chosen as the current node, and the base station repeats the aforementioned node selection process. The node selection process is repeated until the base station is reached.
- **Duplicate route deletion.** If $r_i^{(v)}$ is the same as some previously discovered route, the base station runs the route discovery process again and tries to find a new route.
- $r_i^{(v)}$ **Generation.** $r_i^{(v)}$ is an onion packet with multi-layer encryptions. For example,

if $r_i^{(v)}$ goes through node n_i, a, and b to reach the base station, then $r_i^{(v)}$ has the form $E_{k_{a,\mathrm{BS}}}$ $(a||E_{k_{b,\mathrm{BS}}}(b)||PA)$, where PA is a padding, which makes all onion packets of n_i have the same size.

6.2. Message relay

Suppose node n_i is a source node and wants to send a message M_i to the base station, n_i chooses an onion packet $r_i^{(v)}$ randomly from R_i and broadcasts M_i with the form $i \parallel E_{k_i}\left(r_i^{(v)}||E_{k_{i,\mathrm{BS}}}(data)\right)$. For $\forall n_j \in N_{\mathrm{nei}}^i$, if n_j receives M_i, n_j decrypts M_i and obtains $r_i^{(v)}$. Next, n_j tries to decrypt $r_i^{(v)}$ by $k_{j,\mathrm{BS}}$. If n_j cannot decrypt $r_i^{(v)}$ successfully, n_j discards M_i. Otherwise, n_j transmits the message to its neighbors with the form $M_j = j||E_{k_j}\left(\left(r_i^{(v)}\right)'||E_{k_{i,\mathrm{BS}}}(data)\right)$, where $\left(r_i^{(v)}\right)'$ has the same length as $r_i^{(v)}$. $\left(r_i^{(v)}\right)'$ is firstly decrypted from $r_i^{(v)}$ and then padded by random bits. For example, if n_j receives an onion packet $r_i^{(v)} = E_{k_{j,\mathrm{BS}}}(j||E_{k_{s,\mathrm{BS}}}(s||E(\ldots))||PA)$ from n_i, n_j decrypts the packet and obtains $E_{k_{s,\mathrm{BS}}}(s||E(\ldots))$, then n_j adds a new padding—PA'.

6.3. Performance evaluation

In this section, we evaluate the performance of our PF routing protocol, including the communication cost, computation cost, and security.

6.3.1. Communication cost.

The communication cost is the total number of transmissions of a process. The communication cost of PF includes the message transmissions during the network initialization phase and the message sending phase. Note that we do not include the communication cost of the initial broadcasting because it is the same as other existing routing protocols (e.g., [15,16]). After the broadcast, each node, say, n_i, sends P_i and $\overline{P_i}$ to the base station through the shortest path routing. The communication cost for this is as follows:

$$Q = \frac{\sum_{q=1}^{h_{\max}} \tilde{N}_q q = \sum_{q=1}^{h_{\max}} (2q-1)nq = 2n\sum_{q=1}^{h_{\max}} q^2 - n\sum_{q=1}^{h_{\max}} q = nh_{\max}(h_{\max}+1)(4h_{\max}-1)}{6} \tag{9}$$

where $\tilde{N}_q$ is the number of nodes with hop count q, n denotes the average number of neighbors, and $h_{\max}$ denotes the max hop count. If N nodes are uniformly distributed in the network, we thus have $h_{\max} = \sqrt{N/n}$ and $Q = n\sqrt{n/N}\left(\sqrt{n/N}+1\right)\left(4\sqrt{n/N}-1\right)/6$.

Thereafter, the base station sends u onion packets to each node, and the communication cost is also Q. All in all, we have the total communication cost $2Q$ in the initialization phase.

In the message sending phase, if a source node n_i sends a message to the base station, the communication cost is $h_i + 2h_i(1-p)$.

6.3.2. Computation cost versus transmission latency.

The computation cost for PF is low because PF only uses symmetric encryption. The computation during the network initialization phase is a one-time operation, and it is carried out by the base station where power and computational resource are abundant. During the message sending phase, two encryption operations are needed if a source wants to send a message to the base station. In addition, whenever a node transmits a message, it needs three decryption/encryption operations with two for message verification and one for message transmission.

The transmission latency is evaluated by the transmission times of a message before it reaches the base station. Therefore, the average transmission latency of a message from source node i to the base station is $h_i + 2h_i$ $(1-p)$.

6.3.3. Security analysis.

PF is robust to PAS attack, as adversaries cannot find out the parents of any node. An adversary could stay close to node n_i and monitor and obtain messages exchanged between n_i and its neighbors. Also, the adversary could compromise n_i and obtain all its secret information. However, he or she still cannot find P_i, even though he or she is able to infer the transmission relationship between node n_i and its neighbors. This is because the next forwarding node of n_i may not be n_i's parent (according to the route discovery process). Furthermore, PF can defend against the zeroing-in attack [5] because in PF, nodes do not have hop count information. It is also easy to combine PF with existing base station location protection schemes [3,15] to defend against the packet tracing [4] and rate monitoring attacks [4].

7. EXPERIMENT COMPARISON

We conducted experiments to evaluate our routing protocols. The simulation setup is the same as that in

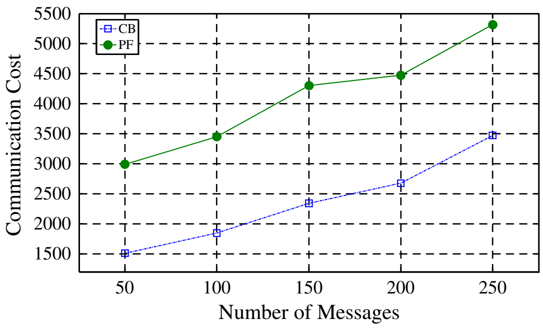


Figure 11. Number of messages versus average communication cost. CB, child-based routing protocol; PF, parent-free routing protocol.

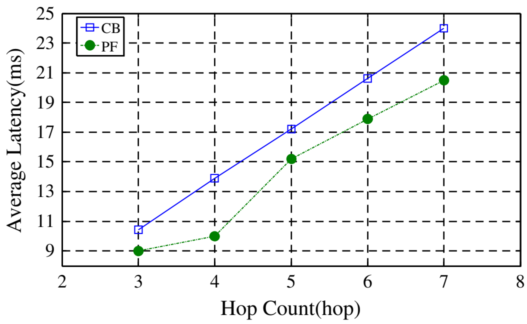


Figure 12. Hop count versus latency. CB, child-based routing protocol; PF, parent-free routing protocol.

PAS (Section 3.6). The base station is in the center of the network. The communication cost and transmission latency comparison between CB and PF are presented in Figures 10 and 11, respectively. In this simulation, the parameters are set as follows: $N = 841$, $n = 12$, $u = 3$, $\theta = 0.1$, $p = 0.8$, $h_{\text{fake}} = 1/4h_{\max}$, and $\delta = 2$ ms.

We selected 11 sources randomly and studied the communication cost with different numbers of messages sent by all sources. Figure 11 illustrates that the communication cost increases for both CB and PF with a growth in the number of messages sent by all sources. It is also observed that PF has a higher communication cost compared with CB. This is because in the initialization stage, each sensor sends its neighboring information to the base station, and the base station also sends u onion packets to each node, which results in extra communication cost in PF. An interesting observation is that the increased communication cost is almost the same despite the increase in the total number of messages. This is because the extra communication cost in PF in the initialization stage is a one-time cost, and it accounts for the most part of the increased communication cost.

Figure 12 illustrates the average end-to-end latency for different hop counts. Five sources with different hop counts are randomly selected. The results of each experiment are averaged over 20 runs. The results show that as the hop count increases, both CB and PF take more time to transmit a message to the base station. It is also observed that CB has a higher average latency compared with PF, because each message is kept for a random time before it is transmitted in CB. In PF, a node transmits a message to one of its parents with probability θ instead of 1, which also results in a slight increase in latency. However, the latency increase is small, because messages can be delivered to the base station successfully with a high probability only with a high transmission probability (more than 0.5).

We can see from both Figures 11 and 12 that compared with CB, PF has less transmission latency and more communication cost. Thus, we can choose PF and CB according to different applications. It is better to use PF in event-driven applications that are sensitive to latency. On the other hand, if energy cost is more important, CB is a better choice.

8. CONCLUSIONS

In this paper, we studied the base station location privacy problem from both the attack and defense sides. First, we presented a new base station attack scheme: PAS. Our theoretical analysis and experiments show that PAS can locate a base station within one sensor radio range in high-density sensor networks. Then, on the basis of PAS, we proposed TP-PAS. Our simulation results demonstrate that TP-PAS is able to determine a base station successfully in both low-density and high-density sensor networks. To protect a base station from PAS and TP-PAS, we designed the PF and CB routing protocols for sensor networks. Theory analysis and experiment results show that CB has less communication cost and more end-to-end latency compared with PF. So, we can choose them according to different situations.

ACKNOWLEDGEMENTS

This research was supported in part by the China National Basic Research Program (973 Program) under grant 2011CB302605; the China National High Technology Research and Development Program (863 Program) under grants 2010AA012504 and 2011AA010705; the Natural Science Foundation of China under grants 61073194 and 61173144; and by the US National Science Foundation under grants CNS-0963578, CNS-1002974, CNS-1022552, and CNS-1065444, as well as the US Army Research Office under grant W911NF-08-1-0334.

REFERENCES

1. McHenry T, Heidemann J. MAC stability in sensor networks at high network densities. *Technical Report ISI-TR-2007-626*, USC/Information Sciences Institute, 2007.

2. Intanagonwiwat C, Estrin D, Govindan R, Heidemann J. Impact of network density on data aggregation in wireless sensor networks. In *Proceedings of the IEEE 22nd International Conference on Distributed Computing Systems*, Vienna, Austria, 2002.
3. Patwari N, Ash JN, Kyperountas S, Hero AO, Moses RL, Correal NS. Locating the nodes: cooperative localization in wireless sensor networks. *IEEE Signal Processing Magazine* 2005; **22**(4): 54–69.
4. Acharya U, Younis M. Increasing base-station anonymity in wireless sensor networks. *Ad Hoc Networks* 2010; **8**(8): 791–809.
5. Liu Z, Xu W. Zeroing-in on network metric minima for sink location determination. In *Proceedings of the Third ACM Conference on Wireless Network Security (WiSec'10)*, Hoboken, NJ, USA, 2010.
6. Deng J, Han R, Mishra S. Countermeasures against traffic analysis attacks in wireless sensor networks. In *Proceedings of the 1st International Conference on Security and Privacy For Emerging Areas in Communications Networks*, Athens, Greece, 2005.
7. Conner W, Abdelzaher T, Nahrstedt K. Using data aggregation to prevent traffic analysis in wireless sensor networks. In *Proceedings of the International Conference on Distributed Computing in Sensor Systems (DCOSS'06)*, San Francisco, 2006.
8. Jian Y, Chen S, Zhang Z, Zhang L. Protecting receiver-location privacy in wireless sensor networks. In *Proceedings of the 26th IEEE International Conference on Computer Communications (INFOCOM'07)*, Alaska, USA, 2007.
9. Karp B, Kung HT. GPSR: greedy perimeter stateless routing for wireless networks. In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM'00)*, Boston, Massachusetts, USA, August 2000.
10. Bhuiyan M, Gondal I, Kamruzzaman J. LACAR: location aided congestion aware routing in wireless sensor networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC'10)*, Sydney, Australia, 2010.
11. Zhao J, Govindan R. Understanding packet delivery performance in dense wireless sensor networks. In *Proceedings of the 1st International Conference on Embedded Networked Sensor Systems (SenSys'03)*, Los Angeles, California, USA, 2003.
12. Intanagonwiwat C, Govindan R, Estrin D. Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM'00)*, Boston, Massachusetts, USA, 2000.
13. Johnson SC. Hierarchical clustering schemes. *Psychometrika, Springer* 1967; **32**(3): 241–254.
14. Erol-Kantarci M, Oktug SF, Vieira LFM, Gerla M. Performance evaluation of distributed localization techniques for mobile underwater acoustic sensor networks. *Ad Hoc Networks* 2011; **9**(1): 61–72.
15. Kamat P, Zhang Y, Trappe W, Ozturk C. Enhancing source-location privacy in sensor network routing. In *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS'05)*, Columbus, USA, 2005.
16. Pongaliur K, Xiao L. Maintaining source privacy under eavesdropping and node compromise attacks. In *Proceedings of the 30th IEEE International Conference on Computer Communications (INFOCOM'11)*, Shanghai, China, 2011.

AUTHORS' BIOGRAPHIES

Juan Chen is a PhD candidate in the School of Computer Science and Technology at the Harbin Institute of Technology, China. She received her BE degree in 2005 and her MS degree in 2008, in Computer Science, from South West University and Harbin Institute of Technology, respectively. Her research interests include wireless network security and privacy.

Hongli Zhang received her BS degree in Computer Software from the Sichuan University on July 1994, and received her MS and PhD degrees in Computer Architecture from the Harbin Institute of Technology on July 1996 and December 1999, respectively. Her research interests are focused in the area of network security, Internet measurement, and network computing. She was awarded three Ministry Science and Technology Progress awards and published over 100 papers in journals and international conferences.

Xiaojiang (James) Du is currently an associate professor in the Department of Computer and Information Sciences at Temple University. Du received his BS degree in Electrical Engineering from Tsinghua University, Beijing, China, in 1996 and his MS and PhD degrees in Electrical Engineering from the University of Maryland College Park in 2002 and 2003, respectively. Du was an assistant professor in the Department of Computer Science at North Dakota State University between August 2004 and July 2009, where he received the Excellence in Research Award in May 2009. His research interests are wireless networks, security, and computer networks and systems. He has published over 80 journal and conference papers in these areas and has been awarded more than $2M research grants from the US National Science Foundation (NSF) and Army Research Office. He serves on the editorial boards of four international journals. Du is the Chair of the Computer and Network Security Symposium of the IEEE/ACM International Wireless Communication and Mobile Computing conference 2006–2010. He is a technical program committee member of several premier ACM/IEEE conferences such as INFOCOM (2007–2012), IM, NOMS, ICC, GLOBECOM, WCNC, BroadNet, and IPCCC. Du is a senior member of IEEE and a life member of ACM.

Binxing Fang received his BS degree in Computer Science from the Harbin Institute of Technology of China in 1981. He received his MS and PhD degrees in Computer Science from Tsinghua University and Harbin Institute of Technology of China in 1984 and 1989, respectively. He is a member of the Chinese Academy of Engineering. His research interests include information security, information retrieval, and distributed systems. Fang is the director of the National Computer Network Emergency Response technical team in China, an expert of the National 863 High-Tech project information security technology field. Fang is the principal investigator of over 30 projects from the state and ministry/province in China. He has published over 200 papers.

Yan Liu is a PhD candidate in the School of Astronautics at the Harbin Institute of Technology of China. She received her BE degree in 2007 and her MS degree in 2009, in Detection Technology and Automatic Equipment Science, from the Northeast University of China. Her research interests include self-calibration of cameras and objects tracking with wireless sensor networks.