Computer Science and Information Systems 13(1):115–130 DOI: 10.2298/CSIS141027040L

Preserving Source- and Sink-location Privacy in Sensor Networks

Sangho Lee[1], Jong Kim[1], and Yoonho Kim[2]

[1] Department of Computer Science and Engineering,
Pohang University of Science and Technology (POSTECH),
Pohang, Republic of Korea
{sangho2,jkim}@postech.ac.kr
[2] Department of Computer Science,
Sangmyung University,
Seoul, Republic of Korea
yhkim@smu.ac.kr

Abstract. Protecting the location privacy of source and sink nodes in a sensor network is an important problem. Source-location privacy is to prevent event source tracking by adversaries and sink-location privacy is to protect sink nodes from adversaries who try to disrupt the sensor network. In this paper, we propose a constant-rate broadcast scheme for ensuring their location privacy. This scheme (1) equalizes traffic patterns of the sensor network to deal with eavesdropping and (2) minimizes the routing information of each sensor node to deal with node compromising. We further reduce the overhead of the proposed scheme by proposing a forwarder-driven broadcast (FdB) scheme that allows efficient multiple broadcasts with smaller buffer usage. Analysis and evaluation results show that FdB can support multiple broadcasts with small message delivery time and buffer usage.

Keywords: sensor networks, location privacy, global eavesdropper, compromised node, information leak.

1. Introduction

Sensor networks are developed for efficient sensing and gathering of data in a large area such as a city, a habitat, or a battlefield. They consist of a large number of resource-constrained sensor nodes and a small number of sink nodes (or base stations). Each sensor node senses events and gathers data in its sensing coverage and it then sends these data to the sink node via multi-hop wireless communications through the other sensor nodes due to its limited transmission range and battery capacity.

However, sensor networks suffer from *location privacy problems*, because adversaries can estimate the locations of their targets (source and/or sink nodes) by exploiting the limited capability of sensor nodes and their wireless structure. Furthermore, several researchers show that message encryption is vulnerable to adversaries who estimate source and destination of messages by eavesdropping and analyzing where wireless traffic is initiated and terminated [2, 11]. Therefore, we demand a method to prevent the traffic analysis to ensure location privacy in sensor networks.

1.1. Related Work

Numerous researchers conduct various studies on location privacy problems of sensor networks, which can be classified either *source-location privacy* or *sink-location privacy* problems.

Source-location Privacy. The source-location privacy problem is how to conceal a source of messages from an eavesdropper, which is serious for applications where the location of an event is important. For example, we consider a sensor network monitoring our troops and enemy troops in battlefields. When enemy troops can assay the location of our troops from the wireless signals of the sensor network, they can locate and attack our troops.

We classify the source-location privacy problem as either source-location privacy against a *local eavesdropper* or source-location privacy against a *global eavesdropper*. The monitoring area of the local eavesdropper is assumed to be the transmission area of a sensor node whereas that of the global eavesdropper is assumed to be the entire sensor network.

Several researchers study how to achieve source-location privacy against a local eavesdropper [11, 8, 10, 14, 18, 20, 25, 26, 13] and a global eavesdropper [17, 16, 19, 24, 25, 27]. The studies on source-location privacy against a local eavesdropper utilize random delay injections [8, 10] or randomized routing [11, 14, 18, 20, 25, 26, 13] to disturb an adversary's traffic analysis in a local area. The studies on source-location privacy against a global eavesdropper utilize dummy source nodes that send dummy messages to a sink node at a constant rate [17, 16]. However, the dummy messages cause much energy consumption and slow message delivery. Therefore, researchers propose schemes for reducing the energy consumption, such as using a small number of dummy source nodes [17, 16], using proxy nodes to filter dummy messages [27], or re-using dummy messages from one-hop neighbor nodes [19]. Researchers also propose schemes for reducing the message delivery latency, such as sending messages to the sink at a probabilistic rate [24] or finding optimal paths to the sink nodes [25, 19].

Sink-location Privacy. The sink-location privacy problem is how to conceal the destination (a sink node in general) of messages from an eavesdropper, which is important for a sensor network where survival of the network is important. For example, a sensor network monitoring battlefields should resist physical attacks to the network. If enemies can estimate which node is a sink node, they try to destroy the sink node to break down the entire network.

We classify studies on sink-location privacy into sink-location privacy against a local eavesdropper [2–5, 9] and sink-location privacy against a global eavesdropper [17]. The studies on sink-location privacy against a local eavesdropper utilize dummy traffic [2–5, 9] and randomized routing to disturb the traffic analysis [3, 4, 9]. The study on sink-location privacy against a global eavesdropper [17] suggests schemes using dummy sink nodes and backbone flooding. In the scheme using dummy sink nodes, each source node redundantly sends messages to several dummy and real sink nodes. In the backbone flooding, each source node sends messages to one of the backbone nodes that flood received messages to all other backbone nodes. The sink node is a neighbor of a backbone node to overhears flooding. However, in both schemes, all traffic still meets at either dummy sink nodes or backbone nodes.

1.2. Motivation and Research Goal

The lack of studies that (1) solve both source- and sink-location privacy problems simultaneously and (2) consider violations of location privacy due to compromised nodes motivates us to conduct a new study. Although Metha et al. [17] consider both source- and sink-location problems in their study, they independently discuss the problems. Furthermore, dealing with compromised nodes is important to assure location privacy, because an adversary can use the routing information leaked by compromised nodes to estimate locations of other sensor nodes. We demand a scheme that can solve the problems simultaneously and effectively.

In this paper, we propose a scheme for source- and sink-location privacy against global eavesdroppers and compromised nodes in sensor networks. The proposed scheme makes every node to *constantly broadcast same-size messages* to its neighbors, which are either real or dummy messages. Only real messages are delivered to every sensor node whereas dummy messages are discarded in a one-hop area. This constant-rate broadcast can eliminate traffic correlations and traffic concentration points from sensor networks while minimizing routing information at each node, because no path from a source node to a sink node is maintained. Therefore, we can guarantee location privacy against global eavesdroppers and routing information leaked by compromised nodes.

We also propose a *forwarder-driven broadcast (FdB)* scheme for reducing the large transmission overhead of message flooding especially when several source nodes simultaneously broadcast real messages. FdB support efficient multiple broadcasts while minimizing buffer usage of each sensor node. We show the efficiency of FdB over the simple flooding and the multipoint relaying (MPR)-based broadcast [21] by mathematical analyses and simulation.

We summarize the main contributions of this paper as follows:

- We propose a scheme for assuring both source-location and sink-location privacy against global eavesdroppers and leaked information from compromised nodes in sensor networks for the first time. All previous work only considers a subset of these problems.
- We propose a forwarder-driven broadcast (FdB) scheme that can decrease the message delivery time and each node's buffer usage for simultaneous constant-rate broadcasts.
- We perform both mathematical analyses and simulations to verify the performance of the proposed scheme.

1.3. Paper Organization

This paper is organized as follows. Section 2 discusses models and problems. Section 3 introduces the proposed scheme. Section 4 evaluates the proposed scheme. Lastly, Section 5 concludes this paper.

2. Models and Problems

2.1. Network Model

We assume a *mission-critical* sensor network requiring higher security and minimum response time rather than a long lifetime (e.g., a sensor network to monitor battlefields).

This sensor network consists of several homogeneous sensor nodes and a single sink node and its sensing field consists of several square cells having randomly deployed sensor nodes, as described in previous work [13, 24, 27]. When two or more nodes locate in the same cell, one of them is activated and others are deactivated. Each cell has an identifier representing its location but only the sink node can match the identifier with its real location. Each sensor node in a cell can directly send messages to its eight one-hop neighboring cells.

To avoid revealing the entire network topology from compromised nodes, only the sink node maintains information about the entire network topology. Each sensor node only keeps information about its one-hop and two-hop neighboring cells by exchanging information about one-hop neighbors with its one-hop neighbors as in a multi-point relay (MPR)-based scheme [21].

Each sensor node has (1) an individual key shared with the sink node, (2) pair-wise shared keys shared with each of its one-hop neighbors, respectively, and (3) a one-hop cluster key shared with all of its one-hop neighbors. A cluster consists of a node and its one-hop neighbor nodes. When the cluster consists of N nodes, each node of the cluster has $N - 1$ pair-wise shared keys and a one-hop cluster key. We assume that previous key management schemes [7, 15, 29, 28, 1] are used to manage these keys.

2.2. Attack Model

The main objectives of an adversary are locating event sources or destroying a sensor network. We assume the adversary has two attack methods: *global eavesdropping* and *node compromising* to locate source or sink nodes. A global eavesdropper can eavesdrop on all traffic in the entire sensor network area using a powerful device that can capture all wireless signals in a large area or deploying several sensor nodes to the target area to monitor traffic [25, 30]. In general sensor networks, source nodes exhibit higher transmission rates than other nodes. Furthermore, almost all traffic goes to the sink. The global eavesdropper can easily locate source and sink nodes by analyzing traffic patterns.

We also assume that although an adversary can compromise sensor nodes, remote attestation schemes [23, 22, 12] can detect them at last. Therefore, an adversary can leak the information stored in the compromised nodes but cannot use the compromised nodes for further attacks such as packet injection and routing disruption.

2.3. Problem Definition

Problem 1. How can we ensure the location privacy of a sensor network against a global eavesdropper?

Definition 1. *Location privacy* is a security property that can be fully satisfied when an adversary cannot determine the location of a target node with a probability larger than $1/N$ where N is the number of nodes in the network.

Definition 2. A *global eavesdropper* is an adversary who can monitor all traffic in a sensor network. When a communication flow exists, the global eavesdropper can estimate the locations of the source and destination of the communication flow from traffic patterns and concentration points.

Researchers propose anti-traffic analysis schemes such as random forwarding [11, 20] and random delay injection [2, 8, 5] to prevent traffic pattern analyses. Some schemes also use dummy communication flows [6] to make traffic pattern analysis much harder.

Even if anti-traffic analysis schemes and dummy flows are used, the global adversary can still guess the source and destination from traffic concentration points. In general sensor networks, almost all the traffic goes to the sink node. Thus, nodes near the sink node send more messages than other nodes. Moreover, some sensor nodes tend to send more messages than other nodes due to unbalanced event occurrence in the sensor network area. Therefore, communication flows should be carefully scheduled to eliminate traffic concentration points.

Uniform scheduling is a simple method to eliminate traffic concentration points and take advantages of random forwarding, delay injection, and dummy flows.

Definition 3. A uniformly scheduled sensor network with encryption is a network that in a designated period, every node of the network transmits the same number of encrypted messages.

Remark 1. *A uniformly scheduled sensor network with encryption can preserve location privacy against global eavesdroppers.*

A global eavesdropper can estimate the locations of its targets from correlations between specific message deliveries and traffic concentration points. In a uniformly scheduled sensor network, however, an eavesdropper cannot capture specific traffic correlations, because the traffic patterns of the uniformly scheduled sensor network are always similar. Moreover, uniform scheduling can eliminate traffic concentration points. When a sensor network is uniformly scheduled, an adversary cannot obtain any advantage from global eavesdropping. As a result, Problem 1 can be solved if we suggest a uniformly scheduled sensor network with encryption.

Problem 2. How can we ensure location privacy of a sensor network against leaked information form compromised nodes?

Definition 4. *A compromised node* is a sensor node that is captured and analyzed by an adversary. This node can provide hints to the adversary about the location of other nodes.

In most routing protocols for sensor networks, every sensor node has routing information to efficiently forward messages to destinations. This routing information can be used as a hint to estimate the location of the destinations. Thus, to protect location privacy against leaked information from compromised nodes, we need to eliminate or reduce the routing information of each node.

Remark 2. *When every sensor node of a sensor network has no or limited routing information, an adversary cannot obtain sufficient location information from a compromised node.*

Having no or limited routing information implies that each node can only identify communications in its local area. Even if an adversary compromises a node, he cannot know the source and destination of communication flows if the node is not near to the source or destination. Therefore, random selection is the best strategy for the adversary. The probability that a randomly selected node nears to the target is less than or equal to

$8/N$ in our network model. Because N is large, the probability is negligible. An adversary cannot obtain sufficient location information from a compromised node. Consequently, Problem 2 can be solved when we use a routing protocol that does not require each node to maintain routing information.

We have to solve Problems 1 and 2 to provide location privacy in sensor networks. A possible solution for both problems is a constant-rate broadcast scheme. If every sensor node constantly broadcasts its real or dummy messages to the entire sensor network, (1) no traffic correlations and no traffic concentration points exist in the sensor network and (2) each sensor node is not required to maintain routing information. However, the constant-rate broadcast scheme may generate too much traffic. As a result, our goal is to suggest a scheme that is just as secure as the constant-rate broadcast scheme while generating less traffic.

3. Proposed Scheme

3.1. Constant-rate Broadcast

In this section, we propose a constant-rate broadcast (or flooding) scheme for ensuring location privacy against global eavesdroppers and compromised nodes. Constant-rate transmissions can eliminate traffic concentration points from sensor networks and broadcast-based routing can deliver messages from sensor nodes to a sink node without providing hints about the location of the sink node. We can preserve the location privacy of sensor networks when we can guarantee these two properties.

The basic idea of the proposed scheme is having each sensor node to periodically broadcast a *same-size* single message that is either *real*, *dummy*, or *control* messages. First, a sensor node sends a real message if it gathers some sensing data or receives real messages from its neighbors in the previous periods. Second, a node sends a dummy message for avoiding traffic analysis attacks if it gathers no data or receives no real message from its neighbors in the previous periods. Third, a node sends a control message including its neighbor information or an updated one-hop cluster key if its neighbors seem to be deactivated or when a one-hop cluster key needs to be changed. Each node appends padding to its messages for generating same-size messages and then encrypts them with a cluster key if they are real or dummy messages, or pair-wise shared keys if they are control messages.

Each node needs to handle messages from its neighbors during message sending intervals. If it receives a real message, it decrypts the message and then checks whether the message is new. It only stores new messages and its sensing data to the sending buffer. If it receives a control message, it updates neighbor information or the cluster key. Otherwise, it simply discards the message.

When a sensor node is scheduled to broadcast a message, it has to inspect its sending buffer to compose the message. If the buffer is not empty, it pops data from the buffer, generates a real message with the data, and then broadcasts the message to its neighbors. If the buffer is empty, it generates a dummy or control message and then broadcasts the message to its neighbors.

Transmission Schedule. The proposed scheme demand careful transmission scheduling to avoid signal collisions due to periodic message broadcast, so we assign different time

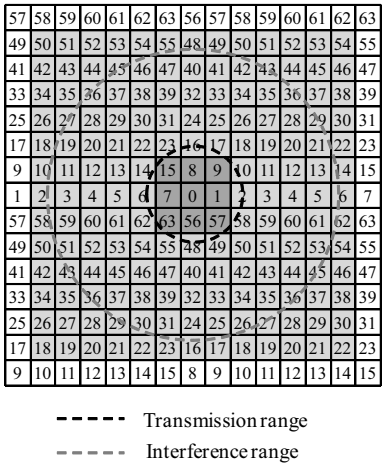


Fig. 1: Transmission schedule when transmission range is 1 cell and interference range is 6 cells centered at cell 0

slots to each cell to avoid collisions. When a node's transmission range is t cells from itself and its interference range is i cells from itself, we assign different time slots for each of $(t + i + 1)^2$ cells. For instance, if a node's transmission range is one cell and its interference ranges are six cells, then each of the 64 cells has a different time slot from 0 to 63 (Figure 1). In this case, when the sending rate is a second, each node has $1/64$ seconds to broadcast a message in every second.

Dead Node Revocation. The proposed scheme has to revoke dead nodes that do not broadcast messages for a long time period (e.g., broken or battery-exhausted nodes) to maintain the network structure. If a sensor node detects dead neighbor nodes, it simply removes those nodes from its neighbor list and then updates a cluster key to revoke the dead nodes. We also treat an attested compromised node as a dead node to exclude it from the network.

3.2. Forwarder-driven Broadcast (FdB)

We have to solve the main problem of the constant-rate broadcast: it generates many redundant messages which can overflow the entire sensor networks. A reputable solution to reduce the number of redundant messages is the multipoint relaying (MPR)-based broadcast scheme [21]. In MPR-based broadcast scheme, a sender node selects a small number of relay nodes that can cover all the two-hop neighbors of the sender nodes. When the sender node broadcasts messages, only the selected relay nodes re-broadcast those messages to their neighbor nodes, which can ensure two-hop delivery of every message with minimal redundancy.

Although MPR-based broadcast scheme works well with constant-rate broadcast when a small number of senders exist, its performance decreases substantially as the number of senders increases. This performance degradation is due to the bottleneck relay nodes that

Input: a set N_o of one-hop neighbors of the node which runs this algorithm;
sets N_is of one-hop neighbors of nodes n_i in N_o
Output: a selected source set S

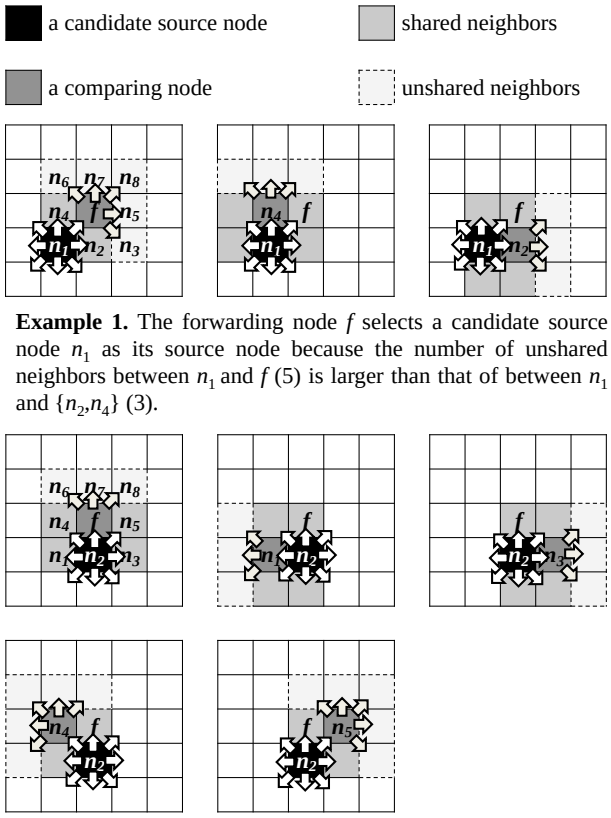
```
1  S ← ∅;
2  C ← ∅;
3  foreach n_i ∈ N_o do
4  |  I_{o,i} ← N_o ∩ N_i;
5  |  D_{o,i} ← N_o − N_i;
6  |  put n_i in S;
7  |  foreach n_j ∈ I_{o,i} do
8  |  |  D_{j,i} ← N_j − N_i;
9  |  |  if |D_{o,i}| ≤ |D_{j,i}| then
10 |  |  |  remove n_i from S;
11 |  |  |  put n_i in C;
12 |  |  |  break;
13 |  |  end
14 |  end
15 end
16 if |S| < |N_o|/2 and |C| ≠ 0 then
17 |  foreach n_i ∈ C do
18 |  |  if n_i is a diagonal neighbor then
19 |  |  |  remove n_i from C;
20 |  |  |  put n_i in S;
21 |  |  |  if |S| ≥ |N_o|/2 then break;
22 |  |  end
23 |  end
24 end
25 while |S| < |N_o|/2 and |C| ≠ 0 do
26 |  remove a random n_i from C;
27 |  put n_i in S;
28 end
29 return S
```

Fig. 2: Source selection algorithm

have many neighbors and, hence, are repeatedly selected by several senders. This bottleneck is a serious hindrance to our scheme, because even relay nodes can only send a single message for each period. This problem can be solved if senders arrive at a consensus when selecting relay nodes. But, this requires a more complex relay node selection algorithm and two-hop communications between senders.

We propose a forwarder-driven broadcast (FdB) scheme for solving the preceding bottleneck problem, which allows each forwarding (or relaying) node to *select* source nodes from its neighbors and *relay* messages only from the chosen nodes. Forwarding nodes have to carefully select source nodes to reduce communication costs while preserving network connectivity. For example, if a forwarding node selects all of its neighbors as sources, the communication cost is the same as that of the simple flooding. On the other hand, if a forwarding node selects a few source nodes, network connectivity may be broken.

We propose a source selection algorithm for forwarding nodes to preserve network connectivity with a small number of source nodes (Figure 2). Let us explain the proposed algorithm with the examples in Figure 3. In these examples, a forwarding node f tries to select some of its one-hop neighbors $n_1, n_2, \ldots, n_8$ as source nodes. For each candidate source node c in $\{n_1, \ldots, n_8\}$, f compares its advantage over the shared neighbors $\mathbf{h}$ between itself and c. The advantage comes up with the number of the unshared neighbors that are neighbors of f but not neighbors of c. A large number of unshared neighbors means that f has an opportunity to forward messages from c to a large number of the nodes that cannot directly receive messages from c. f decides whether it has to forward messages from c by comparing the number of the unshared neighbors between itself and c, and between $\mathbf{h}$ and c. In Example 1, for $c = n_1$ and $\mathbf{h} = \{n_2, n_4\}$, f chooses n_1 as its source node, because the number of unshared neighbors between n_1 and f (5) is larger than the number of unshared neighbors between n_1 and $\{n_2, n_4\}$ (both are 3). In Example 2, for $c = n_2$ and $\mathbf{h} = \{n_1, n_3, n_4, n_5\}$, f does not choose n_2 is its source nodes, because the



Example 1. The forwarding node f selects a candidate source node n_1 as its source node because the number of unshared neighbors between n_1 and f (5) is larger than that of between n_1 and $\{n_2,n_4\}$ (3).

Example 2. f will not selects a candidate source node n_2 as its source node because the number of unshared neighbors between n_1 and $\{n_4,n_5\}$ (5) is larger than that of between n_1 and f (3).

Fig. 3: Source selection examples

number of unshared neighbors between n_2 and a subset of $\mathbf{h}$, $\{n_4, n_5\}$ (both are 5) are larger than the number of unshared neighbors between n_2 and f (3).

3.3. Security Analysis

Security against Global Eavesdroppers. The proposed scheme can ensure the location privacy against global eavesdroppers, because all nodes perform almost the same operations. In wireless networks, a global eavesdropper can ascertain *which nodes* send messages, their *sending rate*, and *the number of messages*. An adversary can estimate the location of a source node from such information, as the source node may send more messages at a higher rate than other nodes. Furthermore, an adversary can estimate the location of a sink node, because the neighbors of the sink node tend to send more messages than other nodes to deliver messages to the sink node. In contrast, the proposed scheme makes every node to send the same number of encrypted real or dummy messages at a constant rate.

Therefore, the global eavesdropper cannot determine the source or sink nodes from the number and rate of messages.

Security against Compromised Nodes. The proposed scheme can assure the location privacy against compromised nodes, because each node maintains the minimum routing information. In the proposed scheme, a sensor node does not know the location of the sink node, so it has to broadcast a real message to an entire sensor network. As a result, even if adversaries compromise a sensor node, they can only obtain the compromised node's local information such as the information about the neighbor nodes and whether a receiving message is real or dummy. We also assume that other nodes can detect a compromised node using remote attestation [23, 22, 12] and update their cluster keys to revoke the compromised node. After the revocation, the adversaries cannot obtain any new information from the sensor networks. They have to use the old information or compromise other nodes to obtain new information for detecting source or sink nodes. Consequently, compromised nodes only give limited advantages to adversaries.

3.4. Event Rate and Buffer Size

In this section, we conduct mathematical analyses to obtain the upper bound of event rates and buffer size for determining the proper message sending rate and buffer size.

Message Delivery Time. We first estimate the worst-case message delivery time to obtain the upper bound of the message sending rate. When the maximum number of messages that has to be forwarded is F and the maximum number of hops for each message to be forwarded is h, the message delivery can be estimated as a h-stage pipeline with F instructions due to multi-hop store-and-forward communications of sensor networks. Thus, the message delivery interval is $h + (F - 1)$. In an $N \times N$ grid ($N \geq 3$), when some of the outermost $4(N - 1)$ nodes have real messages, they have to send their messages up to $N - 1$ hop away nodes, because they do not know where a sink node is. Some of next $4(N - 3)$ nodes that have real messages also need to send their messages up to $N - 2$ hop away nodes and so on (Figure 4). Therefore,

$$h = \begin{cases} \frac{1}{N^2}\left(\sum_{i=1}^{\frac{N-1}{2}} 4(N+1-2i)(N-i) + \frac{N-1}{2}\right), \\ \text{where } N \text{ is odd.} \\ \frac{1}{N^2}\sum_{i=1}^{\frac{N}{2}} 4(N+1-2i)(N-i), \\ \text{where } N \text{ is even.} \end{cases} \tag{1}$$

In the simple flooding, F is the same as the number of sensor nodes N^2. F is also N^2 in MPR-based broadcast, because some bottleneck nodes need to re-broadcast all messages. In FdB, in contrast, F is $\lceil N^2/2 \rceil$, since each forwarding nodes need to re-broadcast the half of all messages. For instance, when $N^2 = 81$ the worst case message delivery interval of the simple flooding and MPR-based broadcast is about 87.96 and FdB is about 47.96. As a result, when each node sends a message every second, it can send a real message every 48 s in FdB.

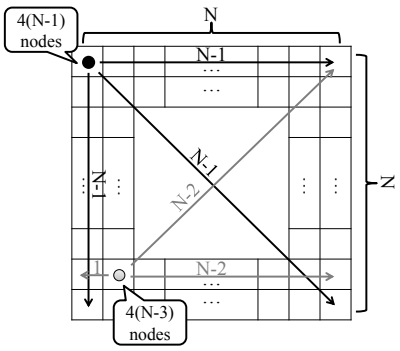


Fig. 4: Example of N × N grid topology to analyze message delivery time

Peak Buffer Usage. We estimate the peak buffer usage of the central node that requires a larger buffer than any other node. We also assume that every node generates a message to maximize buffer usage.

In the first period, the central node receives and store s messages generated by its one-hop source neighbors while broadcasting its own message (s may be eight in the simple flooding and MPR-based broadcast in worst case, and four in FdB.) Thus, the peak buffer usage is $s + 1$.

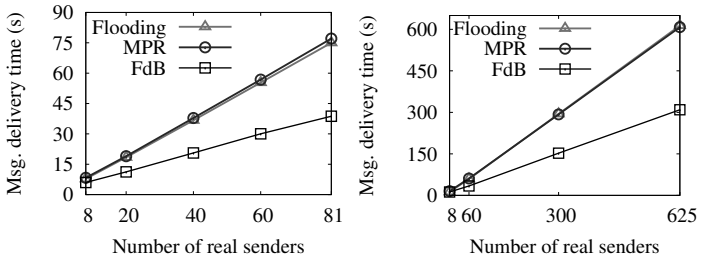
In the next s periods, the central node receives s^2 messages from its source neighbors. Among the s^2 messages, the central node only stores the $2s$ messages from the two-hop away nodes, because the other $s^2 - 2s$ messages are useless (redundant or dummy messages). The peak buffer usage depends on how the $2s$ messages are delivered to the central node. If they are delivered in advance of all the useless messages, the buffer usage instantly increases to $2s + s - 2$, because s is already in the buffer and at least two periods are required to receive $2s$ messages from s source neighbors. Thus, two messages can be forwarded out. If they are delivered after the useless messages, however, the buffer usage increases to $2s$, since s messages can be forwarded out.

In the next $2s$ periods, similarly, the central node receives $2s^2$ messages and store the $3s$ messages from the 3-hop away nodes while forwarding $2s$ messages. In the worst case, the buffer usage instantly increases to $3s + 2s - 3$. In the best case, the buffer usage increases to $3s$.

The buffer usage keeps increasing until the central node receives $s\lfloor N/2 \rfloor$ messages generated by $\lfloor N/2 \rfloor$-hop away source nodes. In that time, the buffer usage instantly increases to $s\lfloor N/2 \rfloor + s(\lfloor N/2 \rfloor - 1) - \lfloor N/2 \rfloor$ in the worst case and $s\lfloor N/2 \rfloor$ in the best case. On average, the peak buffer usage is

$$\frac{1}{2}\left((3s-1)\left\lfloor \frac{N}{2} \right\rfloor - s\right). \tag{2}$$

For instance, when $N^2 = 81$ the peak buffer usage of the simple flooding and MPR-based broadcast is 52 and that of FdB is 24 in the worst case, and 42 and 20 in the average case, respectively. The peak buffer usage linearly depends on the square root of the number of nodes. This implies that its scalability is higher than the message delivery time.



(a) 81 nodes (b) 625 nodes

Fig. 5: Comparison on the effect of the number of real senders on the average message delivery time

4. Evaluation

4.1. Simulation Environments

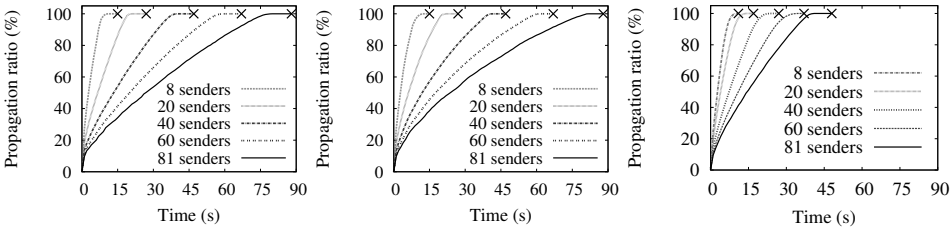
We simulate the proposed scheme using the NS-2 simulator version 2.33. The simulation areas are 90×90 m and 250×250 m. We randomly deploy 81 and 625 nodes to 10×10 m cells of the two simulation areas, respectively. Each node's transmission range is 29 m to ensure direct message broadcast to eight neighbor cells (diagonal cells need $20\sqrt{2} \approx 29$.) The interference range is 60 m; thus, message broadcasts can interfere with cells six-hops away. For each second, each node is scheduled to broadcast one message within $1/64$ seconds; their transmission schedule is as in Section 3.1.

4.2. Message Delivery Time

We estimate the message delivery time of the simple flooding, MPR-based broadcast, and FdB schemes. In the simulation, each of the randomly selected nodes broadcasts a real message to entire networks. First, we measure the average message delivery time. Simulation results show that the average message delivery time linearly depends on the number of real senders. Furthermore, the average message delivery time of FdB is almost half of that of the simple flooding and MPR-based broadcast schemes (Figure 5).

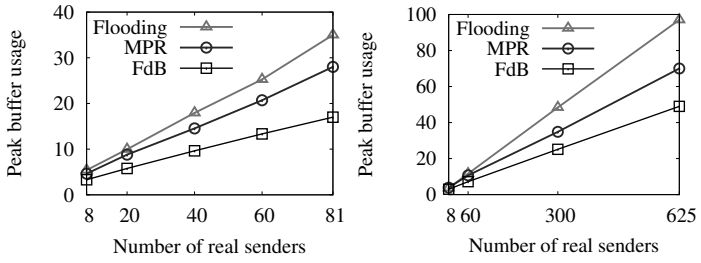
Next, we check the worst-case message deliver time that is the time when the message propagation ratio reaches to 100%. The results show that the worst-case message delivery time of FdB is about half of that of the simple flooding and MPR-based broadcast schemes. Moreover, the worst-case message delivery time does not exceed the mathematically analyzed upper bound of the message delivery time (Figure 6).

From the simulation results, we verify that the message delivery of FdB is twice faster than those of the simple flooding and MPR-based broadcast schemes, so FdB is more suitable to the environments where sensing events are time critical. We also confirm that the message delivery time of MPR-based broadcast is almost the same as the simple flooding in constant-rate broadcast due to the bottleneck relay nodes selected by several sender nodes (Figures 5 and 6).



(a) Flooding (b) MPR-based broadcast (c) Forwarder-driven broadcast

Fig. 6: Comparison on the effect of the number of real senders on the worst case message time that is the time when messages are delivered to all nodes (in 81 nodes). Each × mark represents the mathematically analyzed upper bound of the message delivery time



(a) 81 nodes (b) 625 nodes

Fig. 7: Comparison on the effect of the number of real senders on the peak buffer usage

4.3. Peak Buffer Usage

We check the peak buffer usage of the simple flooding, MPR-based broadcast, and FdB schemes. Simulation results show that the FdB needs a smaller buffer than those of the simple flooding and MPR-based broadcast schemes (Figure 7). Therefore, we verify that the storage cost of FdB is smaller than the simple flooding and MPR-based broadcast schemes. MPR-based broadcast requires a smaller buffer than the simple flooding but it requires a larger buffer than the FdB due to the bottleneck relay nodes. In Section 3.4, we mathematically analyze peak buffer usages and derive Equation 2. In 81 nodes, the analyzed upper bound is 42 for flooding and MPR-based broadcast, and 20 for FdB. In 625 nodes, the analyzed upper bound is 134 for flooding and MPR-based broadcast, and 64 for FdB. As a consequence, we confirm that the simulated peak buffer usage do not exceed the mathematically analyzed peak buffer usages.

4.4. Bottleneck Relay Node

We compare the CDF of the number of source neighbors of each sensor node in 81 nodes to examine the existence of bottleneck relay nodes (Figure 8). In simple flooding, 95.1% of sensor nodes have to relay real messages from more than four neighbors. In MPR-based broadcast, 29.6% of sensor nodes have to relay messages from more than five neighbors.

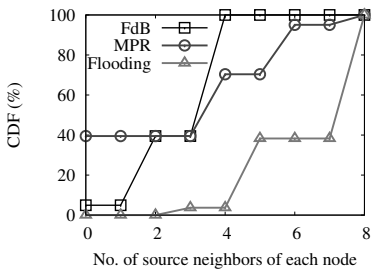


Fig. 8: Comparison on the CDF of the number of source neighbors of each sensor node (in 81 nodes)

This bottleneck is due to the 39.5% of sensor nodes that do not relay any real message. On the other hand, in FdB, all sensor nodes have to relay messages from less than or equal to four neighbors. FdB has no bottleneck, so its message delivery time and buffer usage are definitely better than the simple flooding and MPR-based broadcast.

5. Conclusion

In this paper, we proposed a constant-rate broadcast scheme for ensuring location privacy against global eavesdroppers and leaked information from compromised nodes in sensor networks. The proposed scheme makes each node to broadcast real or dummy messages at a constant rate, so a global eavesdropper cannot determine traffic correlations and traffic concentration points. Furthermore, an adversary cannot obtain location information on other nodes from compromised nodes, because each node does not know where other nodes are. We also proposed a forwarder-driven broadcast (FdB) scheme for efficient concurrent broadcasts. Mathematical analyses and simulation results showed that FdB was better than the simple flooding in terms of message delivery time and buffer requirements.

Acknowledgments. This work was supported by Samsung Research Funding Center of Samsung Electronics under Project Number SRFC-TB1403-04.

References

1. Chen, C., Tsai, Y., Castiglione, A., Palmieri, F.: Using bivariate polynomial to design a dynamic key management scheme for wireless sensor networks. Computer Science and Information Systems 10(2), 589–609 (2013)
2. Deng, J., Han, R., Mishra, S.: Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks. In: Proc. Int. Conf. Dependable Systems and Networks (DSN) (June-July 2004)
3. Deng, J., Han, R., Mishra, S.: Countermeasures against traffic analysis attacks in wireless sensor networks. In: Proc. 1st Int. Conf. Security and Privacy for Emerging Areas in Communications Networks (SecureComm) (September 2005)
4. Deng, J., Han, R., Mishra, S.: Decorrelating wireless sensor network traffic to inhibit traffic analysis attacks. Pervasive and Mobile Computing J. 2(2), 159–186 (April 2006)
5. Deng, J., Han, R., Mishra, S.: INSENS: Intrusion tolerant routing for wireless sensor networks. Computer Communications 29(2), 216–230 (January 2006)

6. Díaz, C., Preneel, B.: Taxonomy of mixes and dummy traffic. In: Proc. 3rd Working Conf. Privacy and Anonymity in Networked and Distributed Systems (I-NetSec) (2004)
7. Eschenauer, L., Gligor, V.D.: A key-management scheme for distributed sensor networks. In: Proc. 9th ACM Conf. Computer and Communications Security (CCS) (November 2002)
8. Hong, X., Wang, P., Kong, J., Zheng, Q., Liu, J.: Effective probabilistic approach protecting sensor traffic. In: Proc. IEEE Military Communications Conf. (MILCOM) (October 2005)
9. Jian, Y., Chen, S., Zhang, Z., Zhang, L.: A novel scheme for protecting receiver's location privacy in wireless sensor networks. IEEE Trans. Wireless Communications 7(10), 3769–3779 (October 2008)
10. Kamat, P., Xu, W., Trappe, W., Zhang, Y.: Temporal privacy in wireless sensor networks. In: Proc. 27th Int. Conf. Distributed Computing Systems (ICDCS) (June 2007)
11. Kamat, P., Zhang, Y., Trappe, W., Ozturk, C.: Enhancing source-location privacy in sensor network routing. In: Proc. 25th Int. Conf. Distributed Computing Systems (ICDCS) (June 2005)
12. Li, Y., McCune, J.M., Perrig, A.: SBAP: Software-based attestation for peripherals. In: Proc. 3rd Int. Conf. Trust and Trustworthy Computing (Trust) (2010)
13. Li, Y., Ren, J.: Preserving source-location privacy in wireless sensor networks. In: Proc. 6th Annual IEEE Communications Society Conf. Sensor, Mesh and Ad Hoc Communication and Networks (SECON) (June 2009)
14. Li, Y., Ren, J.: Source-location privacy through dynamic routing in wireless sensor networks. In: Proc. 29th IEEE Conf. Computer Communications (INFOCOM) (March 2010)
15. Liu, D., Ning, P., Li, R.: Establishing pairwise keys in distributed sensor networks. ACM Trans. Information and System Security 8(1), 41–77 (February 2005)
16. Mehta, K., Liu, D., Wright, M.: Location privacy in sensor networks against a global eavesdropper. In: Proc. IEEE Int. Conf. Network Protocols (ICNP) (October 2007)
17. Mehta, K., Liu, D., Wright, M.: Protecting location privacy in sensor networks against a global eavesdropper. IEEE Trans. Mobile Computing 11(2), 320–336 (2012)
18. Ouyang, Y., Le, Z., Chen, G., Ford, J., Makedon, F.: Entrapping adversaries for source protection in sensor networks. In: Proc. Int. Symp. World of Wireless, Mobile and Multimedia Network (WOWMOM) (2006)
19. Ouyang, Y., Le, Z., Liu, D., Ford, J., Makedon, F.: Source location privacy against laptop-class attacks in sensor networks. In: Proc. 4th Int. Conf. Security and Privacy for Emerging Areas in Communications Networks (SecureComm) (September 2008)
20. Ozturk, C., Zhang, Y., Trappe, W.: Source-location privacy in energy-constrained sensor network routing. In: Proc. 2nd ACM Workshop Security of Ad Hoc and Sensor Networks (SASN) (October 2004)
21. Qayyum, A., Vienno, L., Laouiti, A.: Multipoint relaying for flooding broadcast messages in mobile wireless networks. In: Proc. 35th Annual Hawaii Int. Conf. System Sciences (HICSS) (January 2002)
22. Seshadri, A., Luk, M., Perrig, A., van Doorn, L., Khosla, P.: SCUBA: Secure code update by attestation in sensor networks. In: Proc. ACM Workshop Wireless Security (WiSe) (2006)
23. Seshadri, A., Perrig, A., van Doorn, L., Khosla, P.: SWATT: Software-based attestation for embedded devices. In: Proc. IEEE Symp. Security and Privacy (S&P) (2004)
24. Shao, M., Yang, Y., Zhu, S., Cao, G.: Towards statistically strong source anonymity for sensor networks. In: Proc. 27th IEEE Conf. Computer Communications (INFOCOM) (April 2008)
25. Wang, H., Sheng, B., Li, Q.: Privacy-aware routing in sensor networks. Computer Networks 53(9), 1512–1529 (2009)
26. Xi, Y., Schwiebert, L., Shi, W.: Preserving source location privacy in monitoring-based wireless sensor networks. In: Proc. 20th Int. Parallel and Distributed Processing Symp. (IPDPS) (April 2006)
27. Yang, Y., Shao, M., Zhu, S., Urgaonkar, B., Cao, G.: Towards event source unobservability with minimum network traffic in sensor networks. In: Proc. 1st ACM Conf. Wireless Network Security (WiSec) (March-April 2008)

28. Yao, L., Liu, B., Xia, F., Wu, G.W., Lin, Q.: A group key management protocol based on weight-balanced 2-3 tree for wireless sensor networks. Information - An International Interdisciplinary Journal 14(10), 3261–3278 (2011)
29. Zhu, S., Setia, S., Jajodia, S.: LEAP+: Efficient security mechanisms for large-scale distributed sensor networks. ACM Trans. Sensor Networks 2(4), 500–528 (January 2006)
30. Zhu, Y., Bettati, R.: Compromising location privacy in wireless networks using sensors with limited information. In: Proc. 27th Int. Conf. Distributed Computing Systems (ICDCS) (2007)

Sangho Lee received the B.S. in computer engineering from Hongik University, Korea, in 2006, and the M.S. and Ph.D. degrees in computer science and engineering from Pohang University of Science and Technology (POSTECH), Korea, in 2008 and 2013. He is currently a post-doctoral research associate in the Department of Computer Science and Engineering, POSTECH. His research interests include Web and online social network security, system security, and privacy protection.

Jong Kim received the B.S. in electronic engineering from Hanyang University, Korea, in 1981, the M.S. degree in computer science from the Korean Advanced Institute of Science and Technology (KAIST), Korea, in 1983, and the Ph.D. degree in computer engineering from Pennsylvania State University in 1991. He is currently a professor in the Department of Computer Science and Engineering, Pohang University of Science and Technology (POSTECH), Korea. From 1991 to 1992, he was a research fellow in the Real-Time Computing Laboratory of the Department of Electrical Engineering and Computer Science, University of Michigan. His major areas of interest are fault-tolerant computing, parallel and distributed computing, and computer security. He is the corresponding author of this paper.

Yoonho Kim received his B.S., M.S., and Ph.D. degrees in Computer Science from Seoul National University, Korea. He is currently a Professor of Department of Computer Science at Sangmyung University. His research interests include distributed computing, mobile communication & security, and Web based technologies.

Received: October 27, 2014; Accepted: June 10, 2015.