Cross-layer Enhanced Source Location Privacy in Sensor Networks

Min Shao*, Wenhui Hu*, Sencun Zhu*, Guohong Cao*, Srikanth Krishnamurthy† and Tom La Porta*
*Department of Computer Science and Engineering, The Pennsylvania State University
†Department of Computer Science and Engineering, University of California, Riverside
* {mshao,wxh180,szhu,gcao,tlp}@cse.psu.edu, † krish@cs.ucr.edu

***Abstract*—Source location privacy is an important issue in sensor network monitoring applications. It is difficult to be addressed by traditional security mechanisms, because an external attacker may perform simple traffic analysis to trace back to the event source. Solutions such as flooding or using dummy messages have the drawback of introducing a large amount of message overhead. In this paper, we avoid using network-wide dummy messages by utilizing beacons at the MAC layer. Beacons are sent out regularly, which essentially forms a constant-rate of dummy messages. Using beacons to replace the dummy messages may increase the delivery delay of event information because beacons are only sent out at the predefined beacon interval, but this latency can be controlled. To do this, we propose a cross-layer solution in which the event information is first propagated several hops through a MAC-layer beacon. Then, it is propagated at the routing layer to the destination to avoid further beacon delays. Simulation results show that our cross-layer solutions can maintain low message overhead and high privacy, while controlling delay.**

I. Introduction

In many sensor applications, such as habitat and environmental monitoring, sensors are deployed in natural habitats to monitor events (animals appearing, gun shots, presence of people, etc.) and report these events to a base station. If the sensor network is deployed in a benign and friendly environment, all monitored events can arrive at the base station safely. However, sensor networks normally operate in unattended, harsh or hostile environments, which expose them to various kinds of security threats such as eavesdropping, node compromising, jamming, and physical disruption.

Although traditional security mechanisms such as encryption, authentication and secure routing can address some of the above security threats, they cannot address the following threats in sensor networks. Consider a sensor monitoring application. When a sensor detects an event, it sends a message including event related information to the base station. With traditional security mechanisms, an attacker may not be able to determine the content of the message, but he may find the traffic flow by simple traffic analysis. Furthermore, the attacker may trace back to the location of the event source, and find sensitive information such as whether, when and where an event of concern has happened. For example, knowledge of the appearance of an endangered animal in a sensor field [1] may enable an attacker to take some action to capture the animal. Therefore, it is important to provide privacy besides security in sensor networks.

However, providing source location privacy in sensor networks is complicated by the network scale, the highly constrained system resources, and the fact that sensor networks are often deployed in unattended and hostile environments. As a result, attackers have various ways to monitor the communications in the network, which allow them to infer the temporal/spatial information of certain events and determine where the events originate. Further, many techniques employed in the Internet are not appropriate for sensor networks [2], [3] due to their unique requirements. Although researchers in [1] have made progress in protecting source location privacy in sensor networks, the privacy level becomes low when the attacker has a larger hearing range than the sensor nodes or when it is randomly located in a field.

In this paper, we aim to provide better source location privacy for sensor networks. We consider applications, such as wildlife and environmental monitoring which may demand privacy but can tolerate latencies on the order of a few seconds. For these applications, network-wide dummy messages can achieve a high privacy level, but at the cost of high network traffic and low delivery ratio. To improve the network performance, we avoid the network-wide dummy messages by utilizing beacons at the MAC layer. Beacons are sent out regularly, which essentially forms a constant-rate of dummy messages. By adding event information to the beacon frame, we can deliver the information to the base station (BS) without adding extra traffic. However, using beacons to replace the dummy messages may increase the delay because beacons have to be sent out at the predefined beacon interval. The beacon interval is usually long to allow sensor sleep and generate less traffic. Therefore, it is a challenge to control the delivery delay without changing the beacon interval.

We propose two cross-layer solutions in which the event information is first propagated several hops through the MAC-layer beacon. In the first solution, after information is propagated via beacon broadcasts to a random node, it is routed directly to the BS. In the second, we provide a second level of indirection. After the first round of beacon broadcasting the event information is routed to a random node instead of the BS from where it undergoes a second round of beacon broadcasting. After this, it is routed to the BS.

Compared to existing work [1] which provides privacy against attackers sitting near the BS, our solution can provide much higher privacy. Further, our solution can provide much better privacy against attackers located randomly, perhaps even close to events, within the sensor field. Simulation results demonstrate that our cross-layer solutions can maintain low message overhead and high privacy, while controlling the

delay.

The rest of the paper is organized as follows. We first introduce some background knowledge in Section II. Then, we present the assumptions and design goals in Section III. After that, Section IV,V,VI present three privacy protection schemes. The performance is evaluated in Section VII. Finally, we describe the related work in Section IX and conclude this paper in Section X.

II. Background

In this section, we present some background information on IEEE 802.15.4, which is used in wireless sensor network platforms (e.g., ZigBee [4]).

A. Beacons

In beacon-enabled IEEE 802.15.4, beacons are sent out periodically to announce node presence and certain system parameters, as shown in Figure 1. The MAC payload contains the superframe specification, the pending address specification, address list, and beacon payload fields. The MAC payload is prefixed with a MAC header and appended with MAC footer. The MAC header contains the MAC frame control fields, beacon sequence number and addressing information fields. The MAC footer contains a 16 bit frame check sequence. The MAC header, MAC payload, and MAC footer together form the MAC beacon frame. The beacon payload field is an optional sequence of up to $MaxBeaconPayloadLength$ octets specified to be transmitted in the beacon frame by a higher layer.

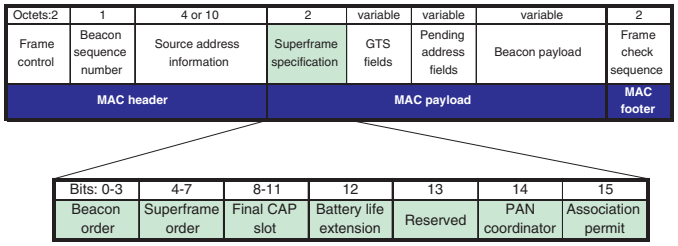


Fig. 1. 802.15.4 Beacon Frame Format

Nodes sleep between beacons to extend their battery life. Beacon interval ranges from 15.36 milliseconds to 786.432 seconds as defined in IEEE 802.15.4.

In IEEE 802.15.4, beacon payload is reserved for different applications. In ZigBee [4], the size of beacon payload is 15 bytes. The content of beacon payload enables new devices to perform network discovery and allow select a network and a neighbor more efficiently. Another example [5] uses 17 bytes beacon payload for an adapted backoff exponent algorithm. In our approach, we can use the beacon payload to transfer data in MAC layer.

B. MAC Layer Encryption

IEEE 802.15.4 provides link layer security which includes four basic security services: access control, frame integrity, data encryption, and sequential freshness. Data encryption uses a stream cipher to protect data from being read by parties without the cryptographic key.

The symmetric encryption algorithm of 802.15.4 uses a key K and an initial vector (IV) as the seed and stretches it into a large pseudo-random keystream $G_K(IV)$. The keystream is then XOR against the plain text: $C = (IV, G_K(IV) \bigoplus P)$. The security mechanisms in 802.15.4 are symmetric-key based using keys provided by the upper layer. Given the rich literature in key management, we do not address key management issues in this paper. We assume two nodes can establish a pairwise key based on existing solutions [6]–[8].

III. Problem Definition

In this section we begin by introducing the network and attacker model, and then discuss the simulation framework.

A. Network Model

As in other sensor networks [9], we assume that a sensor network is divided into cells (or grids) where each pair of nodes in neighboring cells can communicate directly with each other. A cell is the minimum unit for detecting events; for example, a cell header coordinates all the actions inside a cell. For the network to be connected, we assume the nodes in a cell rotate their roles as cell leader and at least one node in the cell is awake. Each cell has a unique id and every sensor node knows in which cell it is located through its GPS or an attack-resilient localization scheme [10], [11].

We assume that a base station (BS) works as the network controller to collect the event data. The BS is interested in the source of an event. Every event has an event id; for example, we may assign a unique id to each type of animal. When a cell detects an event, it will send a triplet (cell id, event id, timestamp), which provides the BS with the source location of the event as well as the time it was detected.

B. Attacker Model

According to the classification in [12], we assume that the attacker is *external*, *local* and *passive*. By external, we assume that the attacker does not compromise or control any sensors. The attacker may launch active attacks by channel jamming or other denial-of-service attacks. However, since these attacks are not related to source anonymity, we do not address them in this paper.

A local attacker can only observe and launch attacks in a limited range. Suppose a sensor network is used to monitor the appearance of pandas. Once a panda appears in some place, the sensors in that place will send a message to the BS. A hunter will be the attacker, and it tries to trace back to the event source to locate the panda. Similar to [1], we assume that the attacker starts from the BS, where it is guaranteed that all packets will arrive eventually. The hunter (attacker) is constantly listening/receiving. Once the attacker hears the first message, he knows which node among the neighborhood sent that message, and will move to the transmitting node. If the attacker does not hear any message for a certain time, he goes back one step and keeps listening. The attacker repeats this process until he reaches the source.

We also consider attackers that are located close to the event or in the sensor field between the event and the BS. These are important to analyze as well because attackers may be

present in the sensor field, although this attack model was not considered in [1].

We assume that the attacker has sufficient resources (e.g., in storage, computation and communication). The attacker can launch a rate monitoring attack and a time correlation attack. In a rate monitoring attack, the attacker pays more attention to the nodes with different (especially higher) transmission rates. In a time correlation attack, the attacker may observe the correlation in transmission times between a node and its neighbor, attempting to deduce a forwarding path.

C. Design Goals

Encryption and authentication can be used to prevent content-based analysis. However, these techniques cannot prevent the traffic analysis attack described above. To defend against this attack, we need to hide the event related traffic. There are different ways to hide this traffic. For example, we can initiate network-wide dummy traffic; this provides the best privacy but the worst performance (high collision rate, low delivery ratio, high network traffic, low network lifetime, etc.). On the other hand, if we reduce the amount of dummy traffic, the performance improves but at the cost of privacy.

In this paper, we try to achieve source anonymity, but also consider the traffic overhead. It is always desirable for the network to have low traffic overhead; however, the privacy level will be low if the source node sends the event notification directly to the BS. For example, the attacker can determine the forwarding path and trace it back to the source. Decreasing traffic overhead without losing privacy is a challenge.

D. Privacy Level Simulation Model

We built a simulator based on CSIM to study the privacy level of different solutions. The network has 10,000 nodes and the BS is located at the center of the network. An event is created at a random location and stays there until it is captured or a certain amount of time expires. Once the attacker gets close to the event (e.g., within certain hops of the panda), the event is considered captured. As soon as the event appears at a location, the closest sensor node, which becomes the source, will start sending packets to the BS to report its observations.

The source generates a new packet every 50 clock ticks until the simulation ends, which occurs either when the attacker catches the event source or when the attacker cannot catch the event source within a certain amount of time. In the trace back simulations, the attacker starts from the BS and moves toward the transmitting node. If the attacker fails to hear any traffic within 200 clock ticks, he goes back one step.

IV. The Naive Solution

Observation: Beacons perform as a "heart-beat" in a beacon-supported network even when there are no events detected. Therefore, if we add event information to the MAC layer beacons, the event information can be spread to the BS without incurring extra routing layer traffic.

A. Privacy Protection

Based on the above observation, we propose a naive privacy protection scheme. After a source node detects a certain event, instead of passing the event information to the routing layer, the node encodes its node id, event id and timestamp in a beacon frame constructing a *modified* beacon frame as shown in Figure 4(a). The source node sends out the *modified* beacon frame to its neighbors in MAC layer encryption mode. After a neighbor node receives the *modified* beacon frame, it decrypts it to extract the event information and adds the event information into its own beacon frame, which will be sent out at the next beacon interval. Every node in the sensor network repeats this process and the event information will eventually arrive at the BS.

Figure 2 shows the process of the naive solution. The solid square node detects an event and broadcasts this event to its neighbors. This process repeats and it eventually reaches the BS. In this small network, 3 hops are needed for the event to reach the BS.

Beacon frames are flooded to all the neighbor nodes. Therefore, in order to stop the event information from circulating in the network, every node maintains a record of the event information they receive from the beacon frame. Each time a node receives a beacon frame, it checks if it already has the event information. If so, it refills the beacon frame with dummy information. If not, it saves the event information and sends it out through its own beacon. Old event information entries are removed after a certain number of beacon intervals to save memory. For example, if the maximum hop from any node to BS is 10, the old entries can be safely removed after 10 beacon intervals.

B. Privacy

The naive solution is simple and achieves perfect privacy, because all the *modified* beacon frames perform exactly the same as the regular beacons. From the attacker point of view, every node sends beacons as defined in the protocol. Therefore, the probability to identify the source node is $1/N$, where N is the total number of sensor nodes in the network.

C. Latency

The event information is only sent out on beacon intervals. The expected waiting time on every node is $t_b/2$, where t_b is the beacon interval. Thus, the latency for an event is $T = t_w + (t_b/2 + (s_b + s_e)/R) \times h$, where t_w is the time from when an event happens to the end of the current beacon interval ($t_w < t_b$), s_b is the size of a regular beacon frame, s_e is the size of the extra event information, R is the transmission rate and h is the number of hops from the source to the BS. Because $t_b/2 \gg (s_b + s_e)/R$ and t_w is expected to be $t_b/2$, $T = (t_b/2) \times (h + 1)$.

The latency is decided by t_b and h. As introduced in Section II-A, the beacon interval (t_b) could be very large. Thus, the latency may be too long. Generally, this naive solution only works in a small-scale network.

In this naive solution, it is impossible for the attacker to find the source location or the occurrence of an event, because the network performs the same before and after an event occurs. However, the latency is long and uncontrollable. In the following sections, we propose several solutions to reduce the latency.

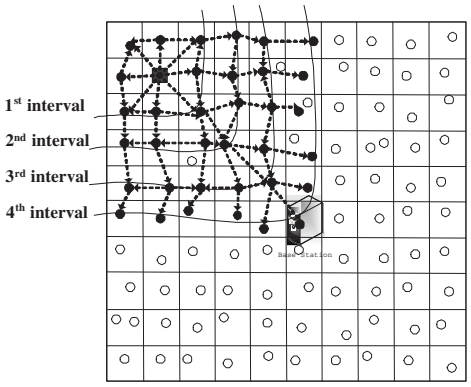


Fig. 2. The Naive Solution

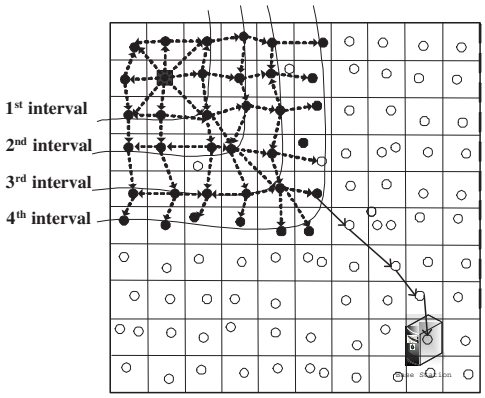
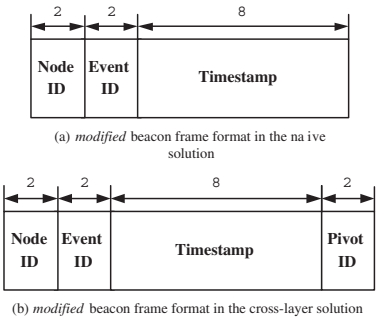


Fig. 3. A Cross-Layer Solution



(a) *modified* beacon frame format in the na ive solution

(b) *modified* beacon frame format in the cross-layer solution

Fig. 4. *modified* beacon frame format

V. A Cross-Layer Solution

Observation: Compared to sending event notification within beacon frames at beacon intervals, sending packets through the routing layer is much faster, but at the cost of lower privacy. Therefore, we want to combine routing and MAC layer solutions.

A. Privacy Protection

The cross-layer solution has two phases: MAC-layer broadcast and routing. In the first phase (MAC-layer broadcast), nodes perform in the same way as the naive solution. After a sensor node detects some event, it broadcasts the event information within MAC layer beacon frames for several hops (a system parameter H). Then, it switches to the second phase (routing). One node, referred to as the *pivot node*, passes the event information to the routing layer and sends it to the BS via routing.

As shown in Figure 3, after the solid square node detects the event, it broadcasts the event information inside beacons for 4 hops (first phase). One node on the 4th hop is selected to send the event information to the BS directly through using conventional routing (second phase).

In Figure 3, if the same pivot node is used for routing all event information, the attacker will be able to easily trace back to the pivot node by observing routing layer traffic. Therefore, different pivot nodes are used to send different event information. This forms different traffic flows to the BS.

The source node is responsible for selecting the pivot node. The source node knows which nodes are H hops away based on the cell information. It randomly picks one of these nodes as the pivot node for each event occurrence and adds that node id to the beacon frame as shown in Figure 4(b).

B. Privacy

To analyze the privacy of the cross-layer system, let's consider Figure 5. The dark circle represents the sensor that detects an event. A_i represents the location at which an attacker may be located. The circle around the event corresponds to the range over which the event information is carried in beacon broadcasts. We assume attackers are located within the range of the sensor detecting the event (A_1), an attacker within the transmission range of the pivot node p_1 (A_2), and an attacker near the BS (A_3).

We define F as the area of the sensor field, N as the number of nodes in the network, D as the degree of a node (the average number of neighbors each node has), r as the transmission range of a node, and H as the number of hops over which the MAC layer broadcast of the event information propagates.

Consider attacker A_1. This attacker only hears periodic beacons transmitted from within its reception range, r. These beacons provide no information as to the location of an event; in fact, they provide no information as to whether an event has occurred at all. Therefore, this attacker, if it knows an event has occurred, either must search the entire sensor field area, F, to find the event, or guess from the pool of all sensors in the network, N, which sensor generated the event. In other words, against A_1, the solution provides perfect privacy. In fact, this is true for all attackers within the beacon propagation range of the event, as long as they are not within range of the pivot node or on the path from the pivot node to the BS.

Consider attacker A_2. This attacker can overhear that p_1 has not received a routing layer packet, but that it does perform a routing layer transmission. Because this transmission is not a beacon, the attacker can conclude an event has occurred within H hops of this pivot node. The attacker cannot trace back to the source because between the pivot node and the source the event information is hidden within the beacon messages which provide no information as described above. Thus, in this case, the attacker can search the area of the circle, $(\pi \cdot (Hr)^2)$, or it can guess which node within the circle initiated the event. Due to broadcasting, the number of such nodes is D^H. Note that because the pivot node changes with each event, A_2 will not likely be within range of the chosen pivot node for all events, so pivot node discovery itself does not have a high likelihood.

Now consider attacker A_3. This attacker can overhear the transmission to the BS. Because this transmission is not a beacon, the attacker can conclude it is the result of an event. In this case the attacker must trace back to find the source of this message. This is difficult because the pivot node p_1 changes with each transmission, causing the path of the flow to the BS to change. Therefore the attacker may take many search iterations to find this pivot node, if at all. Once the pivot node is found, the ability to find the event is the same as for attacker A_2. In fact, any attacker place between the pivot node and the BS will follow the same process for traceback. The difference is that if the attacker is located at the BS, they

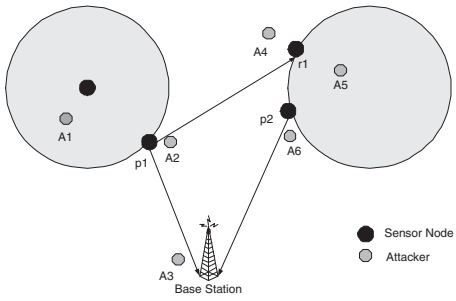
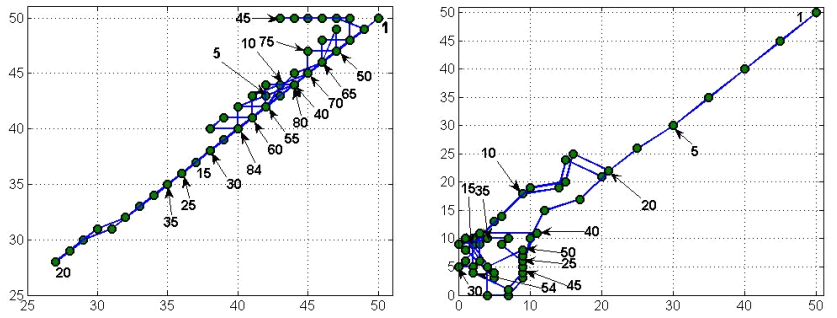


Fig. 5. Privacy Analysis



(a) 84 hops trace, the attacker fails with τ=2 (b) 54 hops trace, the attacker succeeds with τ=5

Fig. 6. The attacker's trace with MAXHOP=6 in a 100*100 network when 66 event messages are sent. The number in the figure shows the attacker hop count.

will always hear an event transmission to the BS. If they are located on the path to the BS, because the selection of the pivot node changes, they may miss many event transmissions.

		$\tau = \frac{attacker's\ hearing\ range}{sensor\ node's\ hearing\ range}$				
		1	2	3	4	5
MAXHOP	4	-	204	65	32	23
	6	-	-	137	97	38
	8	-	-	389	219	139
	10	-	-	401	235	150

TABLE I
AVERAGE HOP NUMBER TO CAPTURE THE SOURCE NODE WHEN SOURCE AND BS ARE 47 HOPS AWAY

It is possible for the attacker to go through the event source before reaching the pivot node, which is hard to analyze. Thus, we use simulations to show the privacy level of our solution. In Table I, $MAXHOP$ specifies the delay tolerance of the network. When $MAXHOP = 4$, it means the event information is broadcast for 4 hops before entering the second phase. As shown in Table I, when the attacker's hearing range is the same as a normal node, the attacker cannot locate the pivot node. When the attacker's hearing range increases, it takes less time to locate the event source.

Figure 6 shows the detailed movement of an attacker in a 100*100 network[1] when $MAXHOP = 6$ during the time that 66 event messages are generated by the source. The attacker starts from (50,50) where the BS is located and the source stays at (3,3) (not shown in the Figure). When the attacker's hearing range is two times that of a normal node, it fails to find the pivot node after 84 hops of trials; the closest point it can reach is (27,28). However, when the attacker's hearing range increases to 5 times that of a normal node, it captures a pivot node within 54 hops.

C. Latency

The latency of this solution is fully determined by phase one which completes when the event reaches the pivot node. Meanwhile, the distance to the pivot node can be selected based on the application latency limitation or other conditions defined by the source node. To achieve lower latency, the pivot node will be closer to the source node, i.e., H will be reduced, and thus the attacker has a smaller area to search and find the pivot node.

In the next section, we propose a second solution to improve the privacy, especially against attackers near the BS, while controlling latency.

VI. A Double Cross-Layer Solution

A. Privacy Protection

The cross-layer solution provides a way to control the latency, but some privacy is sacrificed because attackers near the BS have a possibility of tracing back to the pivot node. To address this issue, we propose a double cross-layer solution which still controls latency while improving privacy.

In the double cross-layer solution, the MAC-layer broadcast phase is divided into two parts. Similar to the cross-layer solution, after the first MAC-layer broadcast, a pivot node is chosen. However, the pivot node does not send the event information to the BS directly. Instead, it sends the event information to a randomly chosen node in the network through the routing layer. Then, this random node will enter the second MAC-layer broadcast mode. At the end of the second MAC-layer broadcast, a pivot node is chosen which routes the event information to the BS directly.

As shown in Figure 7, after the solid square node detects the event, it broadcasts the event information for two hops (the first MAC-layer broadcast). One node on the second hop is selected to send the event information to a random node (p) through routing. This node enters into the second MAC-layer broadcast mode and broadcasts the event information for another two hops. One node is selected again on the second hop relative to the new node as the pivot node. This node sends the event information to the BS directly through routing. Compared with Figure 3, both solutions spend the same number of beacon intervals (4).

B. Privacy

To analyze the privacy of the double cross-layer solution, we add attackers $A_4 - A_6$ and a second pivot node, p_2 to Figure 5. The privacy for attacker A_1 remains the same. Pivot node p_1 now transmits to a reception node r_1. The choice of p_1 changes randomly so the privacy with respect to A_2 also remains the same. Now consider attacker A_4. This attacker can easily traceback to p_1 if it happens to overhear the reception

[1]We use similar parameters as that in [1]. Although the real sensor network may not be at such a large scale, it is used to demonstrate the privacy level of different solutions.

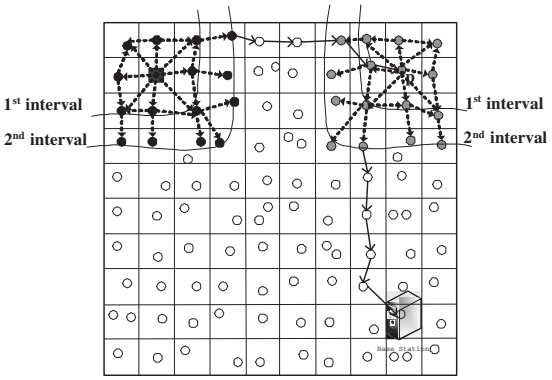
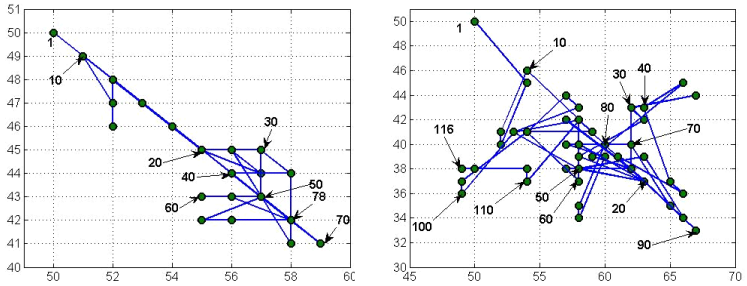


Fig. 7. A Double Cross-Layer Solution



(a) 78 hops trace, the attacker fails with τ=2 (b) 116 hops trace, the attacker fails with τ=5

Fig. 8. The attacker's trace in a 100*100 network with MAXHOP=6 when 133 event messages are sent. The number in the figure shows the attacker hop count.

and transmission of node r_1, so it has the same ability to find the event as does A_2.

Attacker A_5 hears information similar to A_1. Therefore, it does not know an event has even occurred. If it is told an event has occurred, it needs to guess among all nodes within the radius of its circle, D^H, to find the reception node, r_1 from which it could trace back to p_1 and narrow the event area. Because both the pivot node p_1 and r_1 are chosen randomly and independently, this is equivalent to the attacker searching the entire network of N nodes. In other words, no information is gained by A_5.

Attacker A_6 will overhear the transmission from p_2 and realize and event has occurred. However, it must trace back to r_1, which can be any node in the network for each transmission, from where it can trace back to p_1. This makes its ability to discovery the event the same as A_5.

Attacker A_3 will still hear the event arriving at the BS. In the double cross-layer solution, A_3 can trace back to the second pivot node, p_2. At this point, it is in the same position as A_6, so its ability to find the event is the same as guessing among the N nodes in the network. The ability of A_3 to find the pivot node p_1 is not helped by an increase in reception range.

Figure 8 shows the detailed movement of an attacker in the role of A_3 tries to traceback to the source. In this example $MAXHOP = 6$ during the time that 133 event messages are sent out. The attacker starts from (50,50) where the BS is located and the source stays at (3,3). When the attacker's hearing range is two times that of a normal node, it fails to find the pivot node after 78 hops of trials. From Figure 8(a), we can see that the attacker is checking a totally unrelated area. When the attacker's hearing range increases to 5 times that of a normal node, it still falls into the wrong area.

C. Latency

Since the routing delay is much shorter compared to the Beacon interval, the latency in this solution is fully decided by the MAC-layer broadcast. By carefully choosing MAXHOP, the latency of the double cross-layer solution may be made similar to the cross-layer solution.

This analysis shows that the double cross-layer solution provides much stronger privacy against nodes near the BS while still being able to control latency.

VII. Performance Evaluations

In this section, we evaluate the performance and privacy of the three solutions in our paper (the naive solution, the cross-layer solution and the double cross-layer solution) and compare them with the phantom routing scheme proposed in [1]. In phantom routing, the source node that detects an event propagates the event information using a random walk. After a certain number of hops (h_{walk}), a node that receives the event information broadcasts it with a certain probability (P_{fwd}). The information is then broadcast under the control of P_{fwd} by all receiving nodes from this point on. Eventually, the BS will receive the information.

Performance Simulation Setup: In our simulation, each beacon message adds an extra 12 or 14 bytes for event information depending on the solution. In the cross-layer and the double cross-layer solutions, MAC flooding takes two hops. We follow the parameters defined in IEEE 802.15.4, with the data rate at 40 Kb/s, the preamble lasting 800μs for each packet, and the network is considered to be synchronized.

In the simulation, we set up a network with 100*100 cells which is similar to [1]. The base station is located at the center of the network by default. An alternate location (5, 5) is also evaluated for comparison. Source node location is static during each simulation. The distance between the source and the BS ranges from 10 to 40 and is 30 by default. The impact of different beacon intervals, ranging from 0.1 to 1 second, is evaluated and the beacon interval is set as 1 second by default. Event detection interval is 1 seconds. The probabilistic flooding rates (P_{fwd}) of phantom routing are 0.7 and 1.0. Each simulation runs for 100,000 seconds.

Two metrics are used to evaluate the performance of the proposed schemes: latency and traffic overhead.

- *Latency:* The time for an event message traveling from the source to the BS.
- *Traffic Overhead:* The average number of bytes transmitted in each cell in every second.

Privacy Simulation Setup: For the privacy evaluation, we mainly compare the privacy between double cross-layer solution and phantom routing ($P_{fwd} = 0.7$) scheme. *Capture likelihood* and *attacker's hop count* are used to measure the privacy level. Following the simulation model defined in Section III-D, we run the simulation for a total of 200,000 clock ticks and repeat each simulation 10,000 times starting

with different random seeds and record the total number ($n_{capture}$) that the attacker succeeds in capturing the event. The *capture likelihood* is defined as $\frac{n_{capture}}{10{,}000}$. Moreover, within all the successful captures, we define *attacker's hop count* as the average hop number for a attacker to capture an event.

As mentioned in the previous sections, there are other kinds of attacker models. For example, the attackers can be located randomly, perhaps even close to events, within the sensor field. In such cases, our cross-layer solution should have much better privacy compared to phantom routing due to the use of beacons at the MAC layer. In other words, even the attacker hears the communication from the pivot node, it still has a hard time to locate the event source. Since the phantom routing has much worse privacy compared to ours under this attacker model, we will not further compare them.

A. The Impact of Source-Destination Distance

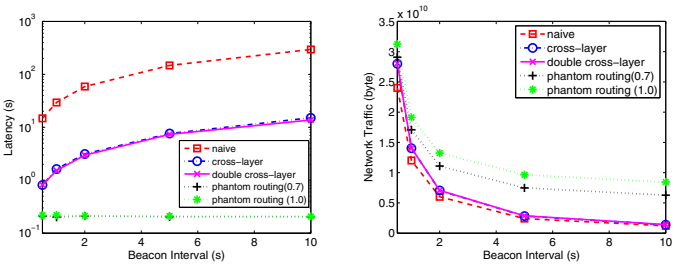
In this section, we investigate the impact of source-destination distance (s-d distance) on performance.

Figure 9(a) shows that the latency of the naive scheme which linearly increases as the s-d distance increases. For every 10 hops increase in s-d distance, the latency of the naive solution increases 5s which is caused by the beacon interval. In the mean time, we notice that the cross-layer and double cross-layer solutions have a slight delay increase compared to phantom routing. The latency of the cross-layer and the double cross-layer schemes almost remain stable when the s-d distance changes. The latency of the cross-layer and the double cross-layer solutions consist of two parts, MAC flooding latency and routing latency. The first part relates to the beacon interval and the second part relates to the routing delay. Since the beacon interval is much greater than the routing delay, the latency of the two schemes is mainly decided by the MAC flooding latency which is fixed.

Figure 9(b) shows the network traffic for all the schemes which are independent of the s-d distance. Network traffic is caused by beacon traffic and routing traffic. Beacon traffic always exists in the network and is not affected by the s-d distance. However, routing traffic is decided by the number of nodes involved in the routing process. In all of the schemes, random nodes are introduced, so the number of nodes involved in the routing process is variable. Phantom routing incurs a high traffic overhead. The network traffic of phantom routing (0.7) is about 25 times higher than ours.

The capture likelihood is compared in Figure 9(c) under 2, 20 and 47 hops of s-d distance. When the s-d distance increases, the capture likelihood decreases. This is because the attacker has to travel further to discover where the source is and the attacker may be diverted on the way. Under the same s-d distance, the capture likelihood is lower in the double cross-layer solution than in the phantom routing solution. Furthermore, when the attacker has the same hearing range as the nodes ($\tau = 1$), the capture likelihood is low except in phantom routing with s-d=2. When the hearing range increases, the capture likelihood of phantom routing increases to 1 immediately. However, the capture likelihood of the double cross-layer changes from .5962 to .9540 and finally to 1.0000 with s-d=20. Obviously, the higher the capture likelihood is, the lower the privacy is. But the capture likelihood might change when the simulation time changes, therefore it cannot be used to measure the privacy by itself. In a later section (Section VII-D), we will discuss it in more detail.

B. The Impact of Beacon Interval



(a) Latency (b) Network Traffic

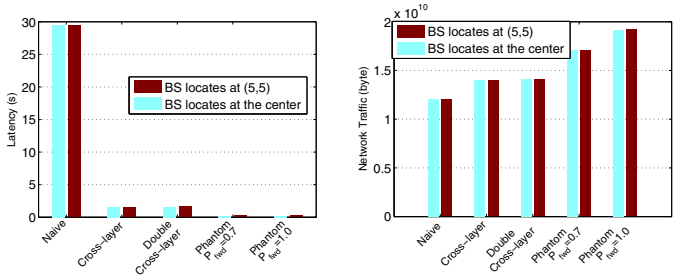
Fig. 10. The Impact of Beacon Interval

In this section, we investigate the impact of beacon interval on performance.

In Figure 10(a), we can see that the latency of phantom routing is independent to the beacon interval because it's determined by the routing delay. However, our three solutions all show the pattern that the latency increase as the beacon interval increases because these schemes use beacons to transfer data and their latency is mostly decided by the beacon interval.

The network traffic of phantom routing remains unchanged when the beacon interval changes, because it does not rely on beacons. However, our solutions depend on beacons to send out messages. When the beacon interval increases, fewer beacons are sent out, which causes the network traffic to decrease. As shown in Figure 10(b), the network traffic of phantom routing is twice as high as the traffic of all our solutions when beacon interval is 0.1s, which is due to the routing layer flooding in the phantom routing scheme. And this difference further increases to about 25 times when beacon interval increases to 1s.

C. The Impact of Base Station Location

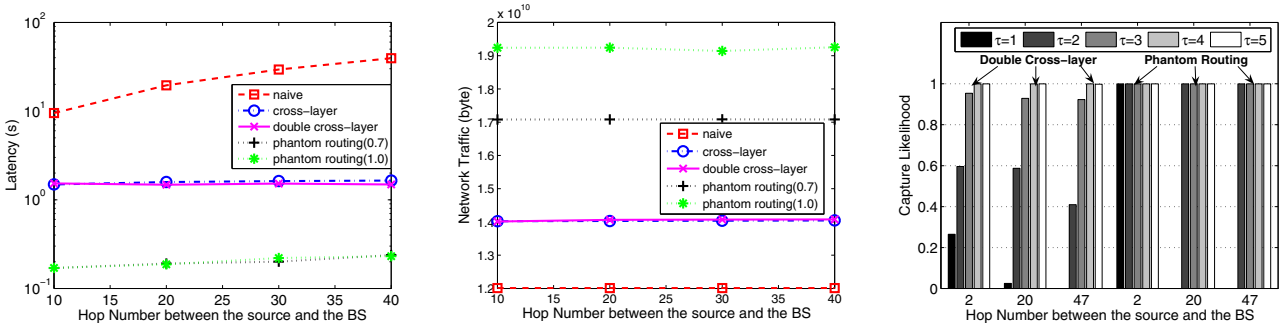


(a) Latency (b) Network Traffic

Fig. 11. The Impact of Base Station Location when s-d distance is 30

In this section, an alternative BS location is considered and compared.

From Figure 11, we can see that the BS location only has a slight influence over the network performance in our solutions. As we discussed above, network traffic doesn't change when s-d distance changes. Similarly, no matter where the BS is, the network traffic depends on the beacon traffic and the routing



(a) Latency (b) Network Traffic (c) Capture Likelihood

Fig. 9. The Impact of Source-Destination Distance

traffic, and neither of them relies on the BS location. For the latency, when the s-d distance is fixed, the latency of the naive solution is fixed no matter where the BS is. The latency of the cross-layer or the double cross-layer solutions mainly depends on MAC flooding which is not related to the BS location.

D. The Impact of the Attacker's Hearing Range

As discussed above, capture likelihood cannot be used to measure the privacy level by itself; that is, the same capture likelihood does not mean the same privacy level. So, in this section, we measure how many hops it actually takes the attacker to capture the event source.

In Figure 12(a), we see that when $\tau = 2$, it takes the attacker more than 2000 hops to capture an event which is only 20 hops away with the double cross-layer solution. This indicates that if the event ends anytime earlier before sending 2000 messages, it will not be captured. When the hearing range increases, the attacker's hop count decreases, but it still takes more than 100 hops, which is 5 times of the s-d distance.

On the other hand, with phantom routing, the attacker could use only 23 hops to capture an event when $h_{walk} = 0$ and $\tau = 5$, which is not safe at all. When phantom routing adopts a higher h_{walk}, the attacker takes longer time to find the event, but is still much shorter than that of the double cross-layer solution. Similar observations can be seen from Figure 12(b), which indicates that the double cross-layer solution has much higher privacy than phantom routing.

VIII. Discussion

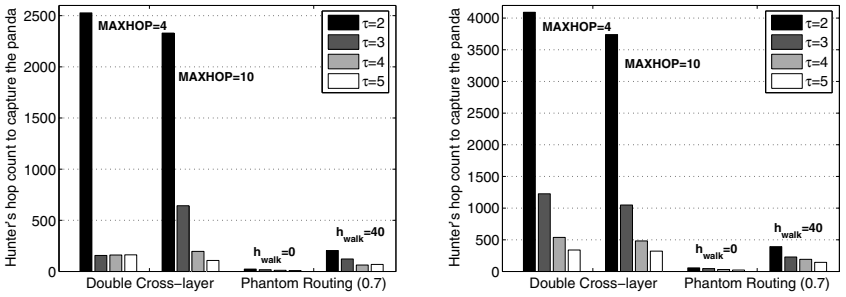
Our cross-layer solutions share the same limitation as in other dummy traffic based schemes [13]–[17] in that beacon rate (or dummy traffic rate in the other schemes) is fixed. Thus, when the real data rate increases and become higher than the beacon rate, real messages will need to be buffered in the source nodes. In the extreme cases the buffer could be overflowed, resulting in message losses. In our default application scenario where the event information is brief, including information such as event source, timestamp and event type, we would expect that buffer overflow will seldom occur. Indeed, if there are multiple messages in the buffer for the same event, in reality the source node could discard those with smaller timestamps. Nevertheless, for applications known to have a high data traffic rate, we could either increase beacon rate or/and increase beacon size to piggyback multiple event messages into one beacon. The price we have to pay to achieve source location privacy is network overhead. The challenge is to cope with burst traffic without losing packets. In this case, a simple solution is only to provide the best-effort service – messages might have to be dropped occasionally, which also occurs during network congestion. Another option is that a source node first estimates the local traffic pattern, and then sends a control message to the BS secretly using our cross-layer schemes. The control message contains information such as burst traffic rate and its duration. The BS can broadcast a message to the network (or an appropriate area around the source node) to temporarily increase the beacon rate for some time. After that, the beacon rate drops to the default one. Note that although not mentioned explicitly, our schemes work the same for the case of multiple data sources.

IX. Related Work

Since Chaum's seminal work [18], so far hundreds of papers [19] have been proposed on building, analyzing, and attacking anonymous communication systems. Due to space limit, we will only discuss those most relevant ones in wireless networks.

We first mention a few works based on the local attacker model. In [1], [20], the Panda-Hunter model was first formalized. A random walk based phantom flooding scheme is proposed to defend against an external local attacker/hunter from tracing back to the source node in an asset monitoring sensor network. [21] proposes a cyclic entrapment method to protect source locations by leading adversaries into traffic loops.

Recently, several location privacy schemes [13]–[17] have been proposed under the global attacker model. All these schemes introduce some dummy traffic to hide the real locations. [13] proposes several techniques for hiding the base station from an external global attacker. [14] theoretically formulates the definition of *statistically strong source anonymity* and proposes a realization scheme to provide high privacy level while minimizing the event message transmission delay. In [15], several filtering schemes are proposed to drop the dummy messages on their ways to the BS to minimize the network traffic while preserving source anonymity. In [16], [17], dummy traffic is introduced so that every sensor node transmits at a constant rate. Compromised sensors may be a problem because they know which packets are from real



(a) Source is 20 hops away from the BS (b) Source is 47 hops away from the BS

Fig. 12. The Impact of Attacker's Hearing Range

sources, our previous work [22] have addressed this issue. Privacy is defined based on entropy [17] and several optimizations are proposed to reduce the forwarding latency of real event messages [16].

Our work differs from the previous work in several ways. First, the existing work either introduces local flooding based dummy traffic [1], [20] or network-wide constant dummy traffic [14]–[17]. The dummy traffic greatly increases the network transmission overhead. Based on the cross-layer principle, our schemes leverage the MAC layer beacon messages to relay the real event information to save communication costs. Second, as our goal is to address local attacks, its privacy level is only comparable to those under the same model; the schemes for the global attacker model provide stronger privacy guarantee, but with higher message overhead.

Several on-demand protocols have been proposed for anonymous routing in mobile ad hoc networks, including ANODR [23], ASR [24], and MASK [25]. They use techniques such as pseudonym and broadcast, and most of them are based on public key cryptography. Moreover, they are not designed for the global, passive, and external adversary model and neither provides unobservability. In our schemes, no on-demand routing is necessary because every sensor knows where to forward its messages.

X. Conclusion

In this paper, we proposed several solutions on source location privacy protection for sensor networks. The proposed solutions offer different levels of location privacy, network traffic and latency by exploring the cross-layer features. Simulation results verified that the double cross-layer solution can reduce the traffic overhead with a reasonable latency. More importantly, it can achieve these benefits without losing source location privacy.

To the best of our knowledge, this is the first paper to use cross-layer approaches to address the source location privacy issues in sensor networks. As the initial work, we do not expect to solve all the problems. In the future, we will investigate smarter attacker models and look into other issues such as how to quantify privacy.

ACKNOWLEDGMENTS

This work was supported in part by MURI/ARO W911NF-07-1-0318.

References

[1] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing source-location privacy in sensor network routing," in *ICDCS '05*.
[2] D. Chaum, "The dining cryptographers problem: Unconditional sender and recipient untraceability," *Journal of Cryptology*, vol. 1, no. 1, pp. 65–75, 1988.
[3] B. Waters, E. Felten, and A. Sahai, "Receiver anonymity via incomparable public keys," *ACM CCS'03*, October 2003.
[4] ANON., "ZigBee Specification," ZigBee Alliance, Inc., 2007.
[5] V. P. Rao and D. Marandin, "Adaptive Backoff Exponent Algorithm for Zigbee (IEEE 802.15.4)," in *NEW2AN*, 2006, pp. 501–516.
[6] W. Du, J. Deng, Y. Han, and P. Varshney, "A pairwise key predistribution scheme for wireless sensor networks," in *CCS '03*.
[7] D. Liu and P. Ning, "Establishing pairwise keys in distributed sensor networks," in *CCS '03*.
[8] S. Zhu, S. Setia, and S. Jajodia, "Leap: Efficient security mechanisms for large-scale distributed sensor networks," in *CCS '03*.
[9] S. RatNasamy, B. karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: A Geographic Hash Table for Data-Centric Storage," *ACM International Workshop on Wireless Sensor Networks and Applications*, September 2002.
[10] D. Liu, P. Ning, and W. Du, "Attack-resistant location estimation in sensor networks," in *Proceedings of The 4^{th} International Conference on Information Processing in Sensor Networks (IPSN)*, 2005.
[11] Y. F. D. W. Y. Zhang, W. Liu, "Secure localization and authentication in ultra-wideband sensor networks," *IEEE Journal on Selected Areas in Communications*, April 2006.
[12] A. Back, U. Möller, and A. Stiglic, "Traffic analysis attacks and trade-offs in anonymity providing systems," in *IHW '01: Proceedings of the 4th International Workshop on Information Hiding*. London, UK: Springer-Verlag, 2001, pp. 245–257.
[13] J. Deng, R. Han, and S. Mishra, "Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks," *DSN '04*.
[14] M. Shao, Y. Yang, S. Zhu, and G. Cao, "Towards statistically strong source anonymity for sensor networks," *INFOCOM 2008*.
[15] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao, "Towards event source unobservability with minimum network traffic in sensor networks," in *WiSec '08*.
[16] Y. Ouyang, Z. Le, D. Liu, J. Ford, and F. Makedon, "Source location privacy against laptop-class attacks in sensor networks," *SecureComm '08*.
[17] K. Mehta, D. Liu, and M. Wright, "Location privacy in sensor networks against a global eavesdropper," in *ICNP 2007*.
[18] D. Chaum, "Untraceable electronic mail, return address, and digital pseudonyms," *Communications of the ACM*, vol. 24, no. 2, pp. 84–88, 1981.
[19] "Anonymity bibliography," http://freehaven.net/anonbib/date.html 2005.
[20] C. Ozturk, Y. Zhang, and W. Trappe, "Source-location privacy in energy-constrained sensor networks routing," *SASN '04*.
[21] Y. Ouyang, X. Le, G. Chen, J. Ford, and F. Makedon, "Entrapping adversaries for source protection in sensor networks," *WoWMoM 2006*.
[22] M. Shao, S. Zhu, W. Zhang, G. Cao, and Y. Yang, "pdcs: Security and privacy support for data-centric sensor networks," *INFOCOM 2007*.
[23] J. Kong and X. Hong, "Anodr: anonymous on demand routing with untraceable routes for mobile ad-hoc networks," in *MobiHoc '03*.
[24] B. Zhu, Z. Wan, M. S. Kankanhalli, F. Bao, and R. H. Deng, "Anonymous secure routing in mobile ad-hoc networks." in *LCN*, 2004.
[25] Y. Zhang, W. Liu, , and W. Lou, "Anonymous communications in mobile ad hoc networks," in *IEEE INFOCOM*, 2005.