Enhancing Source-Location Privacy in Sensor Network Routing

Pandurang Kamat, Yanyong Zhang, Wade Trappe, Celal Ozturk
Wireless Information Network Laboratory (WINLAB)
Rutgers University, 73 Brett Rd., Piscataway, NJ 08854.
Email: {pkamat,yyzhang,trappe,celal}@winlab.rutgers.edu

Abstract— **One of the most notable challenges threatening the successful deployment of sensor systems is privacy. Although many privacy-related issues can be addressed by security mechanisms, one sensor network privacy issue that cannot be adequately addressed by network security is source-location privacy. Adversaries may use RF localization techniques to perform hop-by-hop traceback to the source sensor's location. This paper provides a formal model for the source-location privacy problem in sensor networks and examines the privacy characteristics of different sensor routing protocols. We examine two popular classes of routing protocols: the class of flooding protocols, and the class of routing protocols involving only a single path from the source to the sink. While investigating the privacy performance of routing protocols, we considered the tradeoffs between location-privacy and energy consumption. We found that most of the current protocols cannot provide efficient source-location privacy while maintaining desirable system performance. In order to provide efficient and private sensor communications, we devised new techniques to enhance source-location privacy that augment these routing protocols. One of our strategies, a technique we have called phantom routing, has proven flexible and capable of protecting the source's location, while not incurring a noticeable increase in energy overhead. Further, we examined the effect of source mobility on location privacy. We showed that, even with the natural privacy amplification resulting from source mobility, our phantom routing techniques yield improved source-location privacy relative to other routing methods.**

I. Introduction

Sensor networks promise to have a significant commercial impact by providing strategic and timely data to new classes of realtime monitoring applications. One of the most notable challenges looming on the horizon that threatens successful deployment of sensor networks is privacy. Providing privacy in sensor networks is complicated by the fact that sensor networks consist of low-cost radio devices that employ readily-available, standardized wireless communication technologies. As an example, Berkeley Motes employ a tunable radio technology that is easily observable by spectrum analyzers, while other examples exist of sensor devices employing low-power versions of 802.11 wireless technologies. As a result of the open-architecture of the underlying sensor technology, adversaries will be able to easily gain access to communications between sensor nodes either by purchasing their own low-cost sensor device and running it in a monitor mode, or by employing slightly more sophisticated software radios capable of monitoring a broad array of radio technologies.

Privacy may be defined as the guarantee that information, in its general sense, is observable or decipherable by only those who are intentionally meant to observe or decipher it. The phrase "in its general sense" is meant to imply that there may be types of information besides the message content that are associated with a message transmission. Consequently, the privacy threats that exist for sensor networks may be categorized into two broad classes: content-oriented security/privacy threats, and contextual privacy threats. Content-oriented security and privacy threats are issues that arise due to the ability of the adversary to observe and manipulate the exact content of packets being sent over the sensor network, whether these packets correspond to actual sensed-data or sensitive lower-layer control information. Although issues related to sensor security are important, we believe many of the core problems associated with sensor security are on the road to eventual resolution due to an abundance of recent research by the technical community, c.f. [1–3].

Contextual privacy issues associated with sensor communication, however, have not been as thoroughly addressed. In contrast to content-oriented security, the issue of contextual privacy is concerned with protecting the *context* associated with the measurement and transmission of sensed data. For many scenarios, general contextual information surrounding the sensor application, especially the location of the message originator, are sensitive and must be protected. This is particularly true when the sensor network monitors valuable assets since protecting the asset's location becomes critical.

Many of the privacy techniques employed in general network scenarios are not appropriate for protecting the source location in a sensor network [4–7]. This is partially due to the fact that the problems are different, and partially due to the fact that many of the methods introduce overhead which is too burdensome for sensor networks. One notable challenge that arises in sensor networks is that the shared wireless medium makes it feasible for an adversary to locate the origin of a radio transmission, thereby facilitating hop-by-hop traceback to the origin of a multi-hop communication.

To address source-location privacy for sensor networks, this paper provides a formal model for the source-location privacy problem and examines the privacy characteristics of different sensor routing protocols. We introduce two metrics for quantifying source-location privacy in sensor networks, the safety period and capture likelihood. In our examination of popular routing techniques used in today's sensor networks, we also considered important systems issues, like energy consumption, and found that most protocols cannot provide efficient source-location privacy. We propose new techniques to enhance source-location privacy that augment these routing protocols. It is important that this privacy enhancement does not come at a cost of a significant increase in resource consumption. We have devised a strategy, called phantom routing, that has proven flexible and capable of preventing the adversary from tracking the source location with minimal increase in energy overhead.

II. Asset Monitoring Sensor Networks

One important class of future sensor-driven applications will be applications that monitor a valuable asset. For example, sensors will be deployed in natural habitats to monitor endangered animals, or may be used in tactical military deployments to provide information to networked operations. In these asset monitoring applications, it is important to provide confidentiality to the source sensor's location.

In order to facilitate the discussion and analysis of source-location privacy in sensor networks, we need to select an exemplary scenario that captures most of the relevant features of both sensor networks and potential adversaries in asset monitoring applications. Throughout this paper, we use a generic asset monitoring application, which we have called the *Panda-Hunter Game*, as well as refer to a formal model for asset monitoring applications that can benefit from source-location privacy protection. In this section we begin by introducing the Panda-Hunter Game and the formal model, and then discuss how to model the Panda-Hunter Game using a discrete, event-driven simulation framework.

A. The Panda-Hunter Game

In the Panda-Hunter Game, a large array of panda-detection sensor nodes have been deployed by the Save-The-Panda Organization to monitor a vast habitat for pandas [8]. As soon as a panda is observed, the corresponding *source* node will make observations, and report data periodically to the *sink* via multi-hop routing techniques. The game also features a hunter in the role of the adversary, who tries to capture the panda by back-tracing the routing path until it reaches the source. As a result, a privacy-cautious routing technique should prevent the hunter from locating the source, while delivering the data to the sink.

In the Panda-Hunter Game, we assume there is only a single panda, thus *a single source*, and this source can be either stationary or mobile. During the lifetime of the network, the sensor nodes will continually send data, and the hunter may use this to his advantage to track and hunt the panda. We assume that the source includes its ID in the encrypted messages, but only the sink can tell a node's location from its ID. As a result, even if the hunter is able to break the encryption in a reasonably short time frame, it cannot tell the source's location. In addition, the hunter has the following characteristics:

- **Non-malicious:** The adversary does not interfere with the proper functioning of the network, otherwise intrusion detection measures might flag the hunter's presence. For example, the hunter does not modify packets in transit, alter the routing path, or destroy sensor devices.
- **Device-rich:** The hunter is equipped with devices, such as antenna and spectrum analyzers, so that it can measure the angle of arrival of a message and the received signal strength. From these two measurements, after it hears a message, it is able to identify the immediate sender and move to that node. We emphasize, though, that the hunter cannot learn the origin of a message packet by merely observing a relayed version of a packet. In addition, the hunter can detect the panda when it is near.
- **Resource-rich:** The hunter can move at any rate and has an unlimited amount of power. In addition, it also has a large amount of memory to keep track of information such as messages that have been heard and nodes that have been visited.
- **Informed:** To appropriately study privacy, we must apply Kerckhoff's Principle from security to the privacy setting [9]. In particular, Kerckhoff's Principle states that, in assessing the privacy of a system, one should always assume that the enemy knows the methods being used by the system. Therefore, we assume that the hunter knows the location of the sink node and knows various methods being used by the sensor network to protect the panda.

B. A Formal Model

In order to understand the issue of location privacy in sensor communication, we now provide a formal model for the privacy problem. Our formal model involves the definition of a general asset monitoring network game, which contains the features of the Panda-Hunter game analyzed in this paper.

Definition 1: An asset monitoring network game is a six-tuple $(\mathcal{N}, S, A, \mathcal{R}, \mathcal{H}, \mathcal{M})$, where

1) $\mathcal{N} = \{n_i\}_{i \in I}$ is the network of sensor nodes n_i, which are indexed using an index set I.
2) S is the network sink, to which all communication in the sensor network must ultimately be routed to.
3) A is an asset that the sensor network monitors. Assets are characterized by the mobility pattern that they follow.
4) $\mathcal{R}$ is the routing policy employed by the sensors to protect the asset from being acquired or tracked by the hunter $\mathcal{H}$.
5) $\mathcal{H}$ is the hunter, or adversary, who seeks to acquire or capture the asset A through a set of movement rules $\mathcal{M}$.

The game progresses in time with the sensor node that is monitoring the asset periodically sending out messages.

The purpose of the network is to monitor the asset, while the purpose of the routing strategy is two-fold, to deliver messages to the sink and to enhance the location-privacy of the asset in the presence of an adversarial hunter following a movement strategy. We are therefore interested in privacy measures and network efficiency metrics.

Definition 2: The privacy associated with a sensor network's routing strategy $\mathcal{R}$ can be quantified through two differing performance metrics:

1) The safety period Φ of a routing protocol $\mathcal{R}$ for a given adversarial movement strategy $\mathcal{M}$ is the number of new messages initiated by the source node that is monitoring an asset, before the adversary locates the asset.
2) The capture likelihood L of a routing protocol $\mathcal{R}$ for a given adversarial movement strategy $\mathcal{M}$ is the probability that the hunter can capture the asset within a specified time period.

On the other hand, the network's performance may be quantified in terms of its energy consumption, and the delivery quality. A sensor node consumes energy when it is sending messages, receiving messages, idling, computing, or sensing the physical world. Among all the operations, sending and receiving messages consume the most energy [10, 11]. We measure the energy consumed in a sensor network by the total number of messages that are sent by all the nodes within the entire network until the asset is captured. We assume that messages are all the same length, each sensor transmits with the same transmission power, and hence each transmission by each sensor requires an equal amount of energy. Consequently, the greater the amount of messages required by a strategy, the more energy that strategy consumes. We use two metrics to measure the delivery quality. One is the average message latency, and the other is the event delivery ratio.

In order to illustrate the formal model of the asset monitoring game, we examine a special case of the Panda-Hunter Game. Suppose that we have a sensor network $\mathcal{N} = \{n_i\}$, where nodes n_i are located on a two-dimensional integer grid and that one of these nodes is designated as the network sink. Network devices might monitor a stationary panda, i.e. the asset A, located at a particular sensing node n_A. This node will periodically transmit sensor messages to the sink S following a routing policy $\mathcal{R}$. One possible routing policy $\mathcal{R}$ might be to employ shortest-path routing in which a single route is formed between the source and sink S according to a gradient-based approach. A hunter $\mathcal{H}$, might start at the network sink S, and might follow a movement strategy $\mathcal{M}$. One possible movement strategy could involve $\mathcal{H}$ repeatedly determining the position of the node that relayed the sensor message and moving to that relay node. Another movement strategy might involve $\mathcal{H}$ initially moving two hops, in order to get a head start, and then continue by moving one hop at a time. The safety period Φ corresponds to the amount of messages transmitted by the source which, in the case of the first movement strategy, corresponds directly to the amount of time it takes the hunter to reach the panda. On the other hand, there is a possibility, in the second movement strategy, that the hunter might skip past the panda (when the panda is one hop from the sink), in which case the hunter will miss the panda entirely and thus $L \neq 1$. Clearly, both the safety period Φ and the capture likelihood L depend on the location of the panda, the mobility of the panda, the routing strategy $\mathcal{R}$ and the movement rules $\mathcal{M}$ for the hunter.

C. Simulation Model

We have built a discrete event-based simulator to study the privacy protection of several routing techniques. We are particularly interested in large-scale sensor networks where there is a reasonably large separation between the source and the sink. In order to support a large number of nodes in our simulations, we have made a few approximations. Unless otherwise noted, for the simulation results provided in this paper, we have a network $\mathcal{N}$ of 10,000 randomly located nodes, and the hunter had a hearing radius equal to the sensor transmission radius.

In reality, wireless communication within one hop involves channel sensing (including backoffs) and MAC-layer retransmissions due to collisions. Our simulator ignores the collisions. We emphasize that this should not have a noticeable effect on our accuracy for the following reasons. First, when more reliable MAC protocols are employed, the probability of collision decreases considerably, and channel sensing time may go up correspondingly. Second, sensor networks usually involve light traffic loads with small packets, which result in a lower likelihood of collisions. As a result, our simulator focuses on the channel sensing part. We employ a simple channel sensing model: if a node has m neighbors that may send packets concurrently, the gap before its transmission is a uniformly distributed random number between 1 and m clock ticks. Further, we argue that, although the absolute numbers we report in this paper may not directly calibrate to a real network, the observed performance trends should hold.

Next, let us look at how we implement the Panda-Hunter game in our simulator. In the game, the panda pops up at a random location. Section III considers the scenario where the panda stays at the source until it is caught, while Section IV investigates how the routing techniques perform for a moving panda. Once the hunter gets close to the panda (i.e., within Δ hops from the panda), the panda is considered captured and the game is over. As soon as the panda appears at a location, the closest sensor node, which becomes the source, will start sending packets to the sink reporting its observations. The simulator uses a global clock and a global event queue to schedule all the activities within the network, including message sends, receives and data collections. The source generates a new packet every T clock ticks until the simulation ends, which occurs either when the hunter catches the panda or when the hunter cannot catch the panda within a threshold amount of time (e.g. the panda has returned to its cave).

III. Privacy Protection for a Stationary Source

Rather than build a completely new layer for privacy, we take the viewpoint that existing technologies can be suitably modified to achieve desirable levels of privacy. We will therefore examine several existing routing schemes $\mathcal{R}$ to protect the source's location, while simultaneously exploring how much energy they consume. Specifically, we explore two popular classes of routing mechanisms for sensor networks: flooding and single-path routing. For each of these techniques, we propose modifications that allow for enhanced preservation of the source's location or allow us to achieve improved energy conservation. After exploring each of these two classes, we combine our observations to propose a new technique, which we call *phantom routing*, which has both a flooding and single-path variation. Phantom routing is a powerful and effective privacy enhancing strategy that carefully balances the tradeoffs between privacy and energy consumption.

A. Baseline Routing Techniques

In sensor networks, flooding-based routing and single-path routing are the two most popular classes of routing techniques. In this study, we first examine baseline routing strategies $\mathcal{R}$ from these two classes, and examine their capabilities in protecting the source-location privacy as well as in conserving energy in great depth.

1) Flooding-based Routing: Many sensor networks employ flooding to disseminate data and control messages [12–15]. In flooding, a message originator transmits its message to each of its neighbors, who in turn retransmit the message to each of their neighbors. Although flooding is known to have performance drawbacks, it nonetheless remains a popular technique for relaying information due to its ease of implementation, and the fact that minor modifications allow it to perform relatively well [16, 17].

In our baseline implementation of flooding, we have ensured that every node in the network only forwards a message once, and no node retransmits a message that it has previously transmitted. When a message reaches an intermediate node, the node first checks whether it has received that message before. If this is its first time, the node will broadcast the message to all its neighbors. Otherwise, it just discards the message. Realistically, this would require a cache at each sensor node. However, the cache size can be easily kept very small because we only need to store the sequence number of each message. We assume that each intermediate sensor node can successfully decrypt just the portion of the message corresponding to the sequence number to obtain the sequence number. Such an operation can easily be done using the CTR-mode of encryption. It is thus reasonable to expect that each sensor device will have enough cache to keep track of enough messages to determine whether it has seen a message before.

Probabilistic flooding [16, 17] was first proposed as an optimization of the baseline flooding technique to cut down energy consumption. In probabilistic flooding, only a subset of nodes within the entire network participate in data forwarding, while the others simply discard the messages they receive. The probability that a node forwards a message is referred to as the *forwarding probability* ($P_{forward}$), and plain flooding can be viewed as probabilistic flooding with $P_{forward} = 1$.

In our simulation, we implement probabilistic flooding as follows. Every time a node receives a new message (it discards the message that it has received before no matter whether it has forwarded it or not), it generates a random number q that is uniformly distributed between 0 and 1. If $q < P_{forward}$, the node will forward/broadcast this message to its neighbors. Otherwise, it will just discard that message. The parameter, $P_{forward}$, is important to the overall performance of this approach. A small value can help reduce the energy consumption though at the expense of lower network coverage and connectivity, while a large value can ensure a higher network coverage and connectivity but will have a correspondingly higher energy consumption.

2) Single-Path Routing: Unlike flooding, a large number of energy-efficient routing techniques allow a node to forward packets only to one of (or a small subset of) its neighbors. This family of routing techniques is referred to as *single-path routing* in this paper (e.g., GPSR [18], trajectory-based routing [19], directed diffusion [14], etc). Single-path routing techniques usually require either extra hardware support or a preconfiguration phase. For example, in [18], Karp and Kung propose to use the location information of a node, its neighbors and the destination to calculate a greedy single routing path. In [19], Niculescu and Nath propose trajectory-based routing, which uses the location information associated with a node and its neighbors to create a routing path along a specified trajectory. Such location information can be obtained by either using GPS or other means. In Directed Diffusion [14], an initial phase sets up the "gradients" from each sensor node towards the sink. Later in the routing phase, each intermediate forwarding node can use its neighbors' gradients to implement single-path routing. Whenever the source or the sink changes, a re-configuration stage is required in order to reset the routes.

In this study, we try not to assume extra hardware for a normal sensor node. Instead, we use an initial configuration phase to set up the gradients, i.e. hop count between each node and the sink. In the configuration phase, the sink initiates a flood, setting the initial hop count to 0. Any intermediate node will receive the packet many times. It makes sure that it only processes the packet from all of its neighbors once, discarding duplicates. Every time it receives the message, it increments the hop in the message, records it in its local memory, and then broadcasts to its neighbors. After the initial phase, among all the hop counts it has recorded, a sensor node chooses the minimum value as the number of hops from the sink, and updates its neighbors with that number. Then, every sensor node maintains a neighbor list, which is rank-sorted in ascending order according to each neighbor's hop count to the sink. The head of the list, which has the shortest distance to the sink, is said to have the maximum gradient towards the sink. In the baseline single-path routing protocol, as soon as the source generates a new packet, it forwards the packet to the neighbor with the maximum gradient. Every node along the routing path will repeat this process until the packet reaches the sink. Our version of single-path routing thus corresponds to shortest-path routing, and we use these two terms interchangeably.

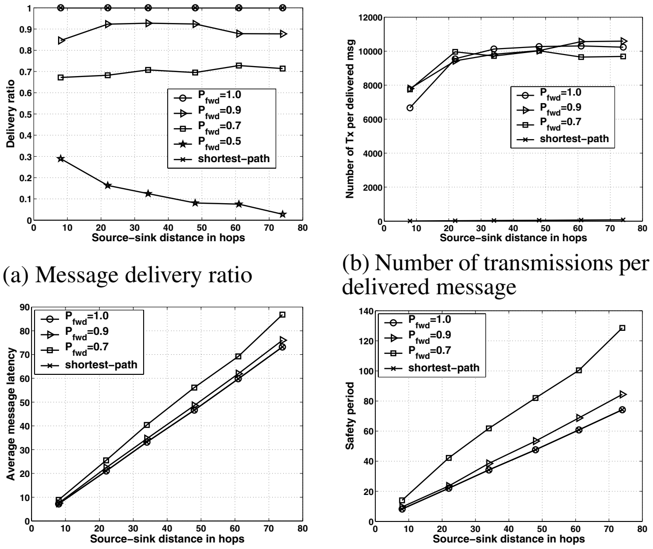
```
Algorithm: Adversary Strategy I: Patient Adversary $\mathcal{H}$

next_location = sink;
while (next_location != source) do
    Listen(next_location);
    msg = ReceiveMessage();
    if (IsNewMessage(msg)) then
        next_location = CalculateImmediateSender(msg);
        MoveTo(next_location);
    end
end
```

Algorithm 1: *The adversary waits at a location until it receives a new message.*

3) Adversary Model and Performance Comparison: Before we delve into the location-privacy protection capability of routing techniques, we define one class of hunter $\mathcal{H}$. In Algorithm 1, the hunter follows a simple but natural adversary model, where the adversary starts from the sink, waits at a location until it hears a new message, and then moves to the immediate sender of that message. It repeats this sequence until it reaches the source location. In this model, the adversary assumes that as long as he is patient enough, he will obtain some information that can direct him to the source. We thus refer to this $\mathcal{H}$ model as a *patient adversary*.

Figures 1(a)-(d) provide the performance of these baseline routing techniques for a patient adversary for different source-sink distances. In this set of results, we have 10,000 nodes uniformly randomly distributed over a 6000×6000 (m^2) network field. The average number of neighbors is 8.5. Among 10,000 nodes, less than 1% are weakly connected with less than 3 neighbors.

a) Delivery Quality: As expected, baseline flooding and shortest-path routing both give good delivery quality, namely, 100% delivery ratio (Figure 1(a)) and lowest message latency (Figure 1(c)). On the other hand, probabilistic flooding may have a poorer delivery quality. In particular, we find that probabilistic flooding techniques with $P_{forward} < 0.7$ result in a low message delivery ratio, especially when the source and the



(a) Message delivery ratio

(b) Number of transmissions per delivered message

(c) Message latency

(d) Safety period

Fig. 1. Performance of baseline routing techniques.

sink are far apart. Figure 1(a) shows that for $P_{forward} = 0.5$, the message delivery ratio can drop below 5%. As a result, we focus our attention on probabilistic flooding techniques with $P_{forward} \geq 0.7$ in the discussion below.

b) Energy Consumption: We use the number of transmissions to measure energy consumption, and instead of using the total energy consumed, we report energy consumption per successfully delivered message since some of the messages may not reach the sink (for probabilistic flooding) and this metric captures the wasted energy. For baseline flooding, every message can successfully reach the sink, and each message incurs n transmissions, where n is the number of sensor nodes in the network. Similarly, single-path routing can deliver all the messages, while each message incurs h transmissions where h is the number of hops in the shortest source-sink path. The number of transmissions per successfully delivered message is more complicated for probabilistic flooding schemes. Each successfully delivered message incurs $nP_{forward}$ transmissions, yet there is no guarantee that each message reaches the sink. This behavior has been studied thoroughly by the community [16, 17].

The effective energy usage is reported in Figure 1(b). Shortest-path routing incurs a much lower energy consumption (h as we discussed above). Three flooding-based techniques have similar energy consumption figures for each successfully delivered message (n as we discussed above). We would like to point out that those data points below $n = 10,000$ for nearby source-sink configurations are because we stopped the simulation as soon as the panda was caught and the flooding of messages had not yet finished.

c) Privacy Protection: Although single-path protocols have desirable energy consumption since they reduce the number of messages sent/received, they are rather poor at protecting the source location privacy (Figure 1(d)). Since only the nodes that are on the routing path forward messages, the adversary can track the path easily, and can locate the source within h moves. The safety period Φ of baseline single-path routing protocols is the same as the length of the shortest routing path because the adversary can observe every single message the source transmits.

At first glance, one may think that flooding can provide strong privacy protection since almost every node in the network will participate in data forwarding, and that the adversary may be led to the wrong source. Further inspection, however, reveals the contrary. We would like to emphasize that *flooding provides the least possible privacy protection as it allows the adversary to track and reach the source location within the minimum safety period.* Figure 1(d) shows that flooding and shortest-path routing lead to the same minimal privacy level. Specifically, the safety period is the same as the hop count on the shortest path.

The poor privacy performance of flooding can be explained by considering the set of all paths produced by the flooding of a single message. This set consists of a mixture of different paths. In particular, this set contains the shortest source-sink path. The shortest path is more likely to reach the hunter first, and thus the hunter will always select the shortest path out of all paths produced by flooding.

In addition to its energy efficiency, probabilistic flooding can improve the privacy protection as well. Imagine there exists a path $\{1, 2, 3, 4, sink\}$, and the adversary is waiting for a new message at node 4. In flooding, the subsequent message will certainly arrive at node 4. However, in probabilistic flooding, the subsequent message may not arrive at node 4 because neighboring nodes may not forward, or take longer to arrive. As a result, the source will likely have to transmit more messages in order for the adversary to work his way back to the source. The more messages the adversary misses, the larger the safety period for the panda, and hence source location protection is provided.

The primary observation is that it is hard for probabilistic flooding techniques to strike a good balance between privacy protection and delivery ratio. For instance, in our study, probabilistic flooding with $P_{forward} = 0.7$ can improve the safety period of baseline flooding roughly by a factor of 2. At the same time, however, it has a message delivery ratio of 70%, which may not be enough for some applications. On the other hand, $P_{forward} = 0.9$ can give a good delivery ratio, but its privacy level is only marginally improved compared to baseline flooding.

B. Routing with Fake Sources

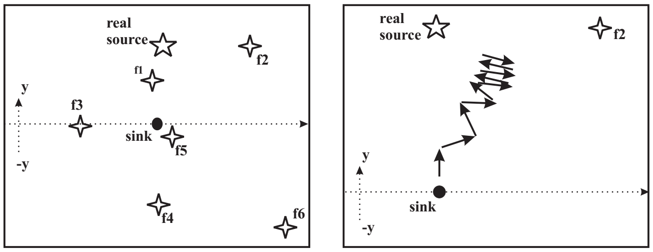
Baseline flooding and single-path routing cannot provide privacy protection because the adversary can easily identify the shortest path between the source and the sink. This behavior may be considered a result of the fact that there is a single source in the network, and that messaging naturally pulls the hunter to the source. This suggests that one approach we can take to alleviate the risk of a source-location privacy breach is to devise new routing protocols $\mathcal{R}$ that introduce more sources that inject fake messages into the network.

In order to demonstrate the effectiveness of fake messaging, we assume that these messages are of the same length as the real messages, and that they are encrypted as well. Therefore, the adversary cannot tell the difference between a fake message and a real one. As a result, when a fake message reaches the hunter, he will think that it is a legitimate new message, and will be guided towards the fake source.

One challenge with this approach is how to inject fake messages. We need to first decide how to create the fake sources, and when and how often these fake sources should inject false messages. Specifically, we want these fake sources to start only after the event is observed, otherwise the use of fake sources would consume precious sensor energy although there is no panda present to protect.

First, let us look at one naive injection strategy that does not require any additional overhead, which we refer to as the *Short-lived Fake Source* routing strategy. This strategy uses the constant P_{fake} to govern the fake message rate, and choose $P_{fake} \propto \frac{1}{n}$. For any node within the network, after it receives a real message, it generates a random number q that is uniformly distributed between 0 and 1. If $q < P_{fake}$, then this node will produce a fake packet and flood it to the network. In this strategy, the fake source changes from one fake message to another. Although this strategy is easy to implement, it does not improve the privacy level of baseline flooding because the fake sources are short-lived. Even if the hunter is guided by one fake message towards a wrong location, there are no subsequent fake messages around that location to draw him even further away, so he can catch the next real message. As a result, we need a persistent fake source to mislead the hunter.

Thus, we introduce a *Persistent Fake Source* routing strategy. The basic idea of this method is that once a node decides to become a fake source, it will keep generating fake messages regularly so that the hunter can be misled. It is intuitive that a fake source close to the real source, or on the way from the sink to the source, can only help lead the adversary towards



(a) Different locations of fake sources

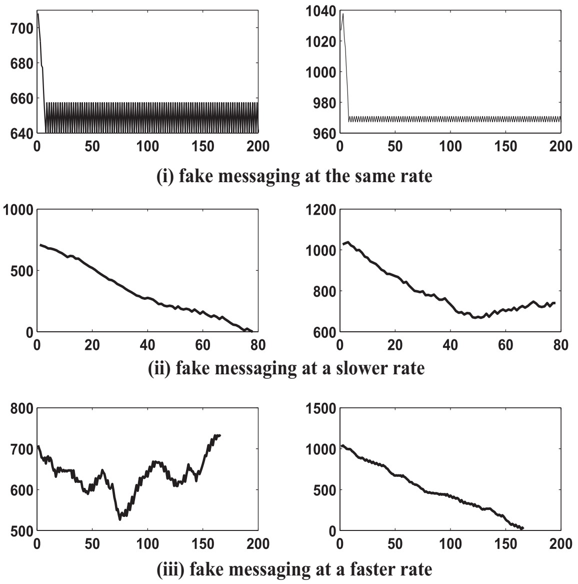
(b) Pulls from both real source and fake source

Fig. 2. Routing with fake sources.

the real source, thus providing a poor privacy protection (such as $f1$ in Figure 2(a)). As a result, locations $f2, f3, f4, f5, f6$ are better alternatives in terms of protecting privacy. Among these locations, we would like to point out that the distances of the fake sources to the sink should be considered as well when choosing a fake source. For example, if a fake source is too far away from the sink compared to the real source, such as $f6$ in our example, then it would not be as effective in pulling the adversary. On the other hand, if a fake source is too close to the sink, it can draw the hunter quickly towards its location, and as we mention below, a hunter can easily detect the fake source in such cases. As a result, we conclude that the fake sources should be comparable to the real source with respect to their distances to the sink. Hence, $f2, f3$, and $f4$ are good candidates.

The above discussion assumes that we have the global picture of the network deployment. There are many ways of implementing this in a distributed manner, and in this study, we discuss a simple way where we assume that each node knows the hop count between itself and the sink, and that the sink has a sectional antenna. The first assumption can be achieved by a simple flood from the sink, as described in Section III-A. The second assumption is valid because sinks usually are much more powerful than normal sensor nodes. Suppose the source is h hops away from the sink and seeks to create a fake source on the opposite side of the sink with a similar distance to the sink. Then the source can embed that information into the data packets. As soon as the sink receives the hop count from the source, it will send a message to one of its neighbors that are in the direction of $-y$ (using the sectional antenna). This node will further pass the message to one of its neighbors whose hop count is larger than its own. If the current node that has the message does not have any neighbors with a larger hop count then we backtrace one step. We repeat this procedure until the message reaches a node whose hop count is comparable to h, and it becomes a fake source. This simple method also allows us to control the number of fake sources.

After a fake source is chosen, the rate of fake messaging can have a significant impact. Figure 3 presents the time series of the hunter's distance from the real source and the fake source for different fake messaging rates corresponding to $f2$ in the scenario in Figure 2(b). If the fake messages are injected into the network at the same rate as the real messages (as shown in Figure 3(i)), then the hunter oscillates between the real source and the fake source, and cannot make progress towards either of them. If the fake messages are injected at a slower rate, as shown in (ii), then the hunter will be drawn towards the real source easily. On the other hand, if the fake messaging rate is higher than the real messaging rate, then the hunter will be kept at the fake source (Figure 3(iii)).

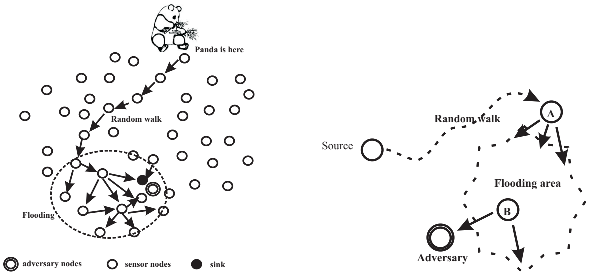


(a) Time series of the hunter's distance from the real source

(b) Time series of the hunter's distance from the fake source

Fig. 3. Fake messaging rates.

The Perceptive Hunter Adversary Model: From the discussion above, one can quickly conclude that, if we have a large energy budget, we can always let fake sources inject messages at a comparable or faster speed than the real messages to protect privacy. However, this scheme cannot work for a more sophisticated hunter. By using the fact that the hunter knows that fake sources are used (Kerckhoff's Principle), the hunter may detect that he has arrived at a fake source because he cannot detect the panda. As a result, if the fake source is too close to the sink, or injects fake messages too fast, then it will be identified as a fake source quickly. Hence, it may appear appealing for the fake source to inject messages at the same rate as the real source. For the scenario in Figure Figure 2(b), we present the results in Figure 3(i), where it is seen that the hunter cannot reach either source, but just oscillate between the two. In the figure, the arrows depict the heard messages that can pull the adversary towards both the real source and the fake source. The hope is that the hunter is trapped by the two conflicting pulls into a "zigzag" movement and will not reach the real source. However, the adversary can detect the zigzag movement rather easily, with the help of its cache that stores the history of locations it has recently visited. At this point, the hunter can conclude that he might be receiving fake messages. As a response, the hunter can choose a random direction and only follow messages from that direction. In our example, let us assume that the adversary chooses to follow the messages from its right, and it can reach the fake source. As soon as it reaches the fake source, it stops because the subsequent messages it receives are from the location it is at, and it can conclude it is sitting at a message source. On the other hand, the hunter is assumed to be able to detect the panda if it is at the real source. As a result, it can conclude that it has reached a fake source. Thus, it *learns* that it should only follow messages coming from its left, and can attempt to trace back to the real source. The lessons learned from the study of fake sources is that, though at an enormous energy cost, fake messaging is nonetheless not effective in protecting the privacy of source locations.



(a) Phantom flooding protocol (b) Example scenario

Fig. 4. Illustration of Phantom Flooding.

C. Phantom Routing Techniques

In the previous sections, we examined the privacy protection capabilities of baseline routing techniques and fake messaging techniques. Both approaches are not very effective in protecting privacy. In both approaches, the sources (either the real one or the fake ones) provide a fixed route for every message so that the adversary can easily back trace the route. Based on this observation, we introduce a new family of flooding and single-path routing protocols for sensor networks, called *phantom routing techniques*. The goal behind phantom techniques is to entice the hunter away from the source towards a phantom source.

In phantom routing, the delivery of every message experiences two phases: (1) the random walk phase, which may be a pure random walk or a directed walk, meant to direct the message to a phantom source, and (2) a subsequent flooding/single-path routing stage meant to deliver the message to the sink. When the source sends out a message, the message is unicasted in a random fashion for a total of h_{walk} hops. After the h_{walk} hops, in phantom flooding the message is flooded using baseline (probabilistic) flooding. In phantom single-path routing, after the h_{walk} hops the message transmission switches to single-path routing. A depiction of the phantom flooding protocol is illustrated in Figure 4(a).

We now discuss the random walk phase in more detail. The ability of a phantom technique to enhance privacy is based upon the ability of the random walk to place the phantom source (after h_{walk} hops) at a location far from the real source. The purpose of the random walk is to send a message to a random location away from the real source. However, if the network is more or less uniformly deployed, and we let those nodes randomly choose one of their neighbors with equal probability, then there is a large chance that the message path will loop around the source spot, and branch to a random location not far from the source.

To further quantify this notion, suppose the network of sensors $\mathcal{N}$ is arrayed on a two-dimensional integer grid with the source and asset A located at $(0,0)$. Suppose the random walk chooses randomly from moving north, south, east, or west, i.e. from $\{(1,0),(-1,0),(0,1),(0,-1)\}$, with equal probability and that the random walk may visit a node more than once. We now estimate the probability that, after h_{walk} hops, the phantom source is within a distance $d < h_{walk}$ of the true source. The movement consists of h_{walk} steps, where each step is an independent random variable X_j with vector values $\{(1,0),(-1,0),(0,1),(0,-1)\}$. The location of the random walk, after h_{walk} steps, is given by

$$D_{h_{walk}} = X_1 + X_2 + \cdots + X_{h_{walk}}.$$

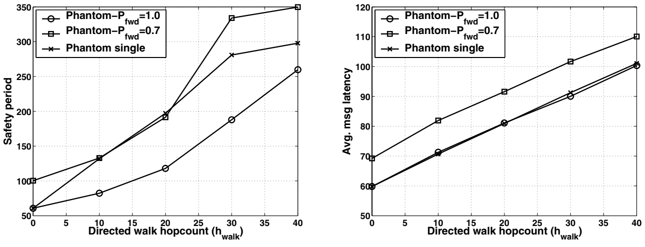
Then, by the central limit theorem, $D_{h_{walk}}/\sqrt{h_{walk}}$ converges in distribution to a bivariate Gaussian with mean $\mathbf{0} = (0,0)$, and covariance matrix $(1/2)\mathbf{I}$. Consequently, $D_{h_{walk}} \sim \mathcal{N}(\mathbf{0}, \frac{h_{walk}}{2}\mathbf{I})$. Let $B = B(\mathbf{0}, d)$ be a ball of radius d centered at $(0,0)$. The asymptotic probability of the phantom source's location $D_{h_{walk}}$ being within a distance d of the real source, after h random walk steps, is given by

$$\begin{aligned} P\left(D \in B\right) &= \frac{1}{h\pi} \int_B e^{-\frac{(x^2+y^2)}{h_{walk}}} dx\, dy \\ &= \frac{1}{h\pi} \int_0^d \int_0^{2\pi} e^{-r^2/h_{walk}} r \, d\theta \, dr \\ &= 1 - e^{-d^2/h_{walk}}. \end{aligned} \quad (1)$$

From this formula, we may examine the likelihood of the phantom's source being within 20% of h_{walk} from the true source after h_{walk} steps, i.e. $d = h_{walk}/5$. The probability is $p = 1 - e^{-h_{walk}/25}$. As we increase h_{walk}, the probability tends to 1, indicating that relative to the amount of energy spent moving a message around, we remain clustered around the true source's location. That is, purely random walk is inefficient at making the phantom source far from the real source, and therefore for reasonable h_{walk} values the location-privacy is not significantly enhanced. These results have been corroborated by simulations involving more general network arrangements, but are not presented due to space considerations.

In order to avoid random walks cancelling each other, we need to introduce bias into the walking process, and therefore we propose the use of a *directed walk* to provide location-privacy. There are two simple approaches to achieving directed walk (without equipping sensor nodes with any extra hardware) that we propose:

- *A sector-based directed random walk.* This approach requires each sensor node to be able to partition the the 2-dimensional plane into two half planes. This can be achieved without using a sectional antenna. Instead, we assume that the network field has some landmark nodes. For example, after the network is deployed, we can mark the west-most node. Then we let that node initiate a flood throughout the network. For a random node i in the network, if it forwards a packet to its neighbor j before it receives the same packet from j, then it can conclude that j is to the east; otherwise, j is to the west. Using this simple method, every node can partition its neighbors into two sets, S_0 and S_1. Before the source starts the directed random walk, it flips a coin and determines whether it is going to use S_0 or S_1. After that, within the first h_{walk} hops, every node that receives the packet randomly chooses a neighbor node from the chosen set for that packet.
- *A hop-based directed random walk.* This approach requires each node to know the hop count between itself and the sink. This can be achieved by the sink initiating a flood throughout the network. After a node first receives the packet, it increments the hop count, and passes the packet on to its neighbors. After the flood phase, neighbors update each other with their own hop counts. As a result, node i can partition its neighbors into two sets, S_0 and S_1, where S_0 includes all the neighbors whose hop counts are smaller than or equal to i's hop count and S_1 includes all the neighbors with a larger hop count. Just as in the sector-based directed random walk, once the two sets are formed, each new message can choose a random set, and every node in the walk can choose a random neighbor from its corresponding set.



(a) Safety Period

(b) Average message latency

Fig. 5. Performance of different phantom routing techniques (source-sink separation is 60 hops).

We now discuss the ability of phantom techniques to increase the safety period, and hence the location-privacy of sensor communications. Phantom flooding can significantly improve the safety period because every message may take a different (shortest) path to reach any node within the network. As a result, after the adversary hears message i, it may take a long time before it receives $i + 1$. When it finally receives message $i+1$, the immediate sender of that message may lead the adversary farther away from the source. In the illustration shown in Figure 4(b), the adversary is already pretty close to the source before it receives the next new message. This new message goes through the random walk phase and reaches node A, and then goes through the flooding phase. The adversary receives this message from node B, and according to its strategy, it will be duped to move to node B, which is actually farther away from the source compared to the current location of the source.

Both phantom flooding and phantom single-path routing exhibit increased privacy protection because of the path diversity between different messages. We conducted a simulation to examine the privacy enhancement for both types of phantom routing. In this simulation, the source-sink separation was fixed at 60 hops, and we used a sector-based directed walk with different walk lengths h_{walk}. The results are presented in Figure 5. A value of $h_{walk} = 0$ corresponds to baseline cases. Phantom techniques clearly demonstrate a much better safety period compared to their baseline counterparts. More importantly, the improvement of phantom schemes keeps increasing with a larger h_{walk}. This is due to the fact that a larger h_{walk} creates a more divergent family of locations for the phantom source, and the probability of sending messages over precisely the same path decreases dramatically.

It is interesting to note that the safety period for phantom shortest-path is larger than for phantom flooding ($p = 1.0$). This behavior is due to the fact that, when we perform routing after the random walk, there is a high likelihood that the resulting single-paths from subsequent phantom sources will not significantly intersect and hence the hunter may miss messages. On the other hand, the resulting floods from subsequent phantom sources will still result in packets arriving at the hunter, allowing him to make progress.

The energy consumed by the phantom techniques is governed by two factors: (1) the walk distance h_{walk}, and (2) the type of flooding/single-path routing stage used. The random walk stage automatically introduces h_{walk} transmissions that were not present in the baseline cases. Typically, however, the predominant energy usage for flooding-based techniques comes from the flooding phase, and usually $h_{walk} \ll n$. As a result, the increased energy consumption is negligible (in fact, it does not even change the energy consumption of baseline flooding). Further, for single-path routing techniques, it introduces at most $2h_{walk}$ extra transmissions to the shortest path between the source and the sink, and the total energy consumption of this approach is still minimal.

Phantom techniques also introduce additional latency because every message is directed to a random location first. We conducted simulations to examine the increase in latency for phantom flooding and phantom single-path routing, as presented in Figure 5(b). Examining this plot we see that the additional latency increases roughly linearly with h_{walk} for each phantom technique. Combining the latency results and the safety period results, it is interesting to note that for a minor increase in latency, the safety period increases dramatically. For example, for $h_{walk} = 20$, the latency increased roughly 30% while the privacy almost quadrupled!

The Cautious Hunter Adversary Model: We now introduce a new model for the hunter $\mathcal{H}$, which we call the *cautious adversary* model. Since phantom techniques might leave the hunter stranded far from the true source location, the cautious adversary seeks to cope by limiting his listening time at a location. If he has not received any new message within a specified interval, he concludes that he might have been misled to the current location, and he goes back one step and resumes listening from there. We illustrate the cautious adversary model in Algorithm 2. We conducted an experiment with different source-sink separations using phantom single-path routing with $h_{walk} = 10$ hops. In our study, the cautious adversary waited at a location for a period of time corresponding to 4 source messages before deciding to retreat one step. The results are presented in Table I. The cautious adversary model does not provide any benefit over the patient adversary model, as the safety period is higher and the capture likelihood is less. This is because the hunter does not make significant forward progress. Consequently, it is better for the hunter to stay where he is and be patient for message to arrive.

IV. Privacy Protection for a Mobile Source

In this section we study routing and the location privacy of a mobile asset A. Particularly, in the context of the Panda-Hunter Game, the panda is now mobile. The observations regarding privacy for stationary assets do not directly apply to a mobile

```
Algorithm: Adversary Strategy II: Cautious Adversary $\mathcal{H}$

prev_location = sink;
next_location = sink;
while (next_location != source) do
    reason = TimedListen(next_location, interval);
    if (reason == MSG_ARRIVAL) then
        msg = ReceiveMessage();
        if (IsNewMessage(msg)) then
            next_location = CalculateImmediateSender(msg);
            MoveTo(next_location);
        end
    else
        next_location = prev_location;
        prev_location = LookUpPrevLocation(prev_location);
        MoveTo(next_location);
    end
end
```

Algorithm 2: *The adversary waits at a location for a period of time and returns to its previous location if no message arrives within that period of time.*

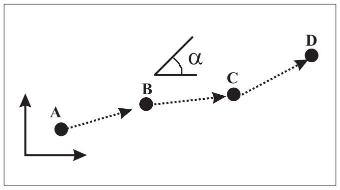


Fig. 6. A simple movement pattern.

asset scenario. Instead, a set of new questions arise. For example, since a mobile panda corresponds to a mobile source, there is a dynamically changing shortest routing path, and therefore it is natural to ask whether the moving panda alone is sufficient to protect its location privacy? Is a faster panda more safe or vice versa? How do flooding-based techniques fare for a mobile panda compared to a static one? How about single-path routing techniques?

The panda's mobility is defined by its movement pattern and its velocity. The purpose of this paper is not to define a sophisticated movement pattern, nor to study a comprehensive set of movement patterns. Rather, we employ a rather simple movement model, illustrated in Figure 6, to study privacy. In this model, the panda knows the coordinates and knows which direction it is moving along. The parameter α governs the direction of movement. Specifically, if u is its current location, and v is its next location, then the angle of $\overrightarrow{uv}$ should be be within the range $[0, \alpha]$. For instance, in Figure 6, the Panda traverses A, B, C, and D, and the direction of any link is within $[0, \alpha]$. Since our simulator has a finite network field, after the panda reaches the boundary of the network, it cannot find any sensor node in the specified direction, retreats a few steps, and resumes its normal pattern. In addition to its direction, it has the other parameters which describe its velocity: δ is the stay time at each location, and d denotes the distance for each of its movements. In the simulation, the sensor node that is closest to the Panda will become the new source, and will send $\lfloor \frac{\delta}{T} \rfloor$ (where T is the reporting interval) new messages before the Panda moves on.

The Impact of Velocity: We first conducted simulations to evaluate the effect of the panda's velocity on source-location privacy. In this experiment, the source-sink hop count is 48, the source sends out a message every 15 clock ticks and $P_{forward}$ is 1.0 in the flooding techniques studied. The results are presented in Table II. Here, the first observation is that, for all routing techniques, a fast moving panda (lower δ values) is safer than a slow panda. The second observation is that, among different techniques, the velocity of the panda has a more noticeable impact on single-path routing techniques than it does on flooding-based routing techniques. For single-path routing, the capture likelihood L is closely related to the velocity of the panda. In particular, a faster moving panda makes it unlikely that the adversary can track the panda. On the other hand, flooding for the same mobility allows the panda to be caught, though with an increased safety period Φ. This observation can be explained as follows. In single-path routing, subsequent shortest paths might not have significant overlap due to the panda's movement, and hence the hunter may not even see a subsequent message. On the other hand, flooding guarantees that the hunter will see the message, though not from the shortest source-sink path, and he may still follow the panda's movement. That is, a reasonably fast moving panda alone is sufficient to protect its location when using single-path routing. The third observation is that panda mobility can improve the privacy protection of phantom techniques more than it does to other schemes. These observations are due to the fact that the source mobility serves to further decorrelate the source's location from the phantom source's location, resulting in enhanced location privacy.

The Impact of the Hunter's Hearing Range: So far, we have assumed that the hunter's hearing range (r_H) is the same as any normal sensor node (r). Next, let us look at the impact of different hearing ranges on the privacy level of a network. For this purpose, we conducted a set of simulation studies for phantom single-path routing with a source-sink separation of 48 hops. The resulting capture likelihoods for different δ/T and r_H/r combinations are presented in Table III. In general, we find that a larger hearing range helps the hunter since this translates into the hunter hearing messages sooner and allows him to make larger moves, effectively allowing him to move faster. We also see that ability for the hunter to capture pandas improves with larger hearing ranges, and that the relative improvement is more pronounced for faster pandas. It should be realized, however, that this corresponds to introducing a powerful adversary. We also measured the impact of hearing range for single-path routing, and observed that phantom single-path routing has improved privacy for larger hearing radii compared to baseline single-path routing.

V. Related Literature

Contextual privacy issues have been examined in the context of general networks, particularly through the methods of anonymous communications. Chaum proposed a model to provide anonymity against an adversary doing traffic analysis [4]. His solution employs a series of intermediate systems called mixes. Each mix accepts fixed length messages from multiple sources and performs one or more transformations on them, before forwarding them in a random order. In the IP routing space, onion routing [5] uses this model to provide anonymous connections. Similarly, the Mixmaster remailer [6] is an email

Source-Sink Separation 8 hops		
	Capture Likelihood (L)	Safety Period (Φ)
Patient hunter	1	32
Cautious hunter	0.90	54

Source-Sink Separation 34 hops		
	Capture Likelihood (L)	Safety Period (Φ)
Patient hunter	1	90
Cautious hunter	0.60	301

Waiting time is 60 clock ticks and $h_{walk} = 10$ hops.

TABLE I

Comparison of phantom single-path routing for two adversarial models.

Routing techniques	$\delta/T = 2$		$\delta/T = 6$		$\delta/T = 18$	
	L	Φ	L	Φ	L	Φ
flooding	1.0	54	1.0	50	1.0	47
phantom-flood	1.0	92	1.0	75	1.0	78
single-path	0.43	51	0.80	50	1.0	51
phantom-single	0.40	134	0.67	169	1.0	107

In this experiment the hop count between the source and the sink is 48. The source emits a new message every 15 clock ticks.

TABLE II

The impact of moving velocity on different routing techniques.

implementation of Chaum mixes. Chaum mixes provide destination privacy when an attacker knows the source. An alternative strategy to anonymity was proposed by Reiter in [20], where users are gathered into geographically diverse groups, called Crowds, to make it difficult for identifying which user makes a Web request.

In [7], a distributed anonymity algorithm was introduced that removes fine levels of detail that could compromise the privacy associated with user locations in location-oriented services. For example, a location-based service might choose to reveal that a group of users is at a specific location, or an individual is located in a vague location, but would not reveal that a specific individual is located at a specific location. Duri examined the protection of telematics data by applying privacy and security techniques [21].

Preserving privacy is an important and challenging task in data mining and databases [22–24]. A common technique is to perturb the data and to reconstruct distributions at an aggregate level. A distribution reconstruction algorithm utilizing the Expectation Maximization (EM) algorithm is discussed in [25], and the authors showed that this algorithm converges to the maximum likelihood estimate of the original distribution based on the perturbed data.

Many of these methods are not appropriate for sensor networks, particularly sensor networks that are deployed for monitoring valuable assets. In particular, location-privacy techniques built using network security mechanisms, such as the anonymity provided by mixes, incur additional communication, memory, and computational overhead that are prohibitive for use in resource-constrained environments. Consequently, full-fledged privacy solutions are not appropriate, and lightweight, resource-efficient alternatives should be explored.

VI. Concluding Remarks

Sensor networks will be deployed to monitor valuable assets. In many scenarios, an adversary may be able to backtrace message routing paths to the event source, which can be a serious privacy breach for many monitoring and remote-sensing application scenarios. In this paper, we have studied the ability of different routing protocols to obfuscate the location of a source sensor. We examined several variations of flooding-based and single-path routing techniques, and found that none of these protocols are capable of providing source location privacy. To achieve improved location privacy, we proposed a new family of routing techniques, called phantom routing, for both the flooding and single-path classes that enhance privacy protection. Phantom routing techniques are desirable since they only marginally increase communication overhead, while achieving significant privacy amplification. Going forward we plan to investigate stronger adversarial models, as well as multiple asset tracking scenarios and their impact on location privacy in sensor networks.

Phantom Single-Path Routing

δ/T	$r_H/r = 1$	$r_H/r = 2$	$r_H/r = 3$
1	0.23	0.43	0.60
2	0.40	0.77	0.93
6	0.67	0.90	0.97
8	0.80	0.97	0.97

Single-path Routing

δ/T	$r_H/r = 1$	$r_H/r = 2$	$r_H/r = 3$
1	0.23	0.50	0.73
2	0.43	0.77	0.90
6	0.80	0.97	0.97
8	0.87	0.97	1.0

TABLE III

The impact of the hunter's hearing range on capture likelihood.

References

[1] A. Perrig, R. Szewczyk, D. Tygar, V. Wen, and D. Culler, "SPINS: security protocols for sensor networks," *Wireless Networks*, vol. 8, no. 5, pp. 521–534, 2002.

[2] L. Eschenaur and V. Gligor, "A key-management scheme for distributed sensor networks," in *Proceedings of the 9th ACM conference on Computer and communications security*, 2002, pp. 41–47.

[3] M. Bohge and W. Trappe, "An authentication framework for hierarchical ad hoc sensor networks," in *Proc. of the 2003 ACM Workshop on Wireless Security*, 2003, pp. 79–87.

[4] D. Chaum, "Untraceable electronic mail, return addresses, and digital pseudonyms," *Communications of the ACM*, vol. 24, pp. 84–88, 1981.

[5] M.Reed, P. Syverson, and D. Goldschlag, "Anonymous connections and onion routing," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 482–494, May 1998.

[6] "Mixmaster remailer," http://mixmaster.sourceforge.net/.

[7] M. Gruteser and D. Grunwald, "Anonymous Usage of Location-based Services through Spatial and Temporal Cloaking," in *Proceedings of the nternational Conference on Mobile Systems, Applications, and Services (MobiSys)*, 2003.

[8] "WWWF - the conservation organization," http://www.panda.org/.

[9] W. Trappe and L.C. Washington, *Introduction to Cryptography with Coding Theory*, Prentice Hall, 2002.

[10] A. Cerpa and D. Estrin, "ASCENT: Adaptive Self-Configuring Sensor Networks Topologies," in *Proceedings of IEEE INFOCOM'02*, June 2002.

[11] W. Ye, J. Heidemann, and D. Estrin, "An Energy-Efficient MAC Protocol for Wireless Sensor Networks," in *Proceedings of IEEE INFOCOM'02*, June 2002.

[12] C. L. Barrett, S. J. Eidenbenz, L. Kroc, M. Marathe, and J. P. Smit, "Parametric probabilistic sensor network routing," in *Proceedings of the 2nd ACM international conference on Wireless sensor networks and applications*, 2003.

[13] Z. Cheng and W. Heinzelman, "Flooding Strategy for Target Discovery in Wireless Networks," in *proceedings of the Sixth ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM 2003)*, 2003.

[14] C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks," in *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM)*, August 2000.

[15] H. Lim and C. Kim, "Flooding in Wireless Ad-hoc Networks," in *IEEE computer communications*, 2000.

[16] D. Braginsky and D. Estrin, "Rumor routing algorthim for sensor networks ," in *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, 2002.

[17] P. Th. Eugster, R. Guerraoui, S. B. Handurukande, P. Kouznetsov, and A.-M. Kermarrec, " Lightweight probabilistic broadcast," *ACM Transactions on Computer Systems (TOCS)*, vol. 21, no. 4, pp. 341 – 374, November 2003.

[18] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks," in *Proceedings of the Sixth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM)*, August 2000.

[19] D. Niculescu and B. Nath, "Trajectory Based Forwarding and its Applications," in *Proceedings of the Ninth Annual ACM/IEEE International Conference on Mobile Computing and Networks (MobiCOM)*, September 2003, pp. 260–272.

[20] M. Reiter and A. Rubin, "Crowds: anonymity for web transactions," *ACM Transactions on Information and System Security*, vol. 1, pp. 66–92, 1998.

[21] S. Duri, M. Gruteser, X. Liu, P. Moskowitz, R. Perez, M. Singh, and J. Tang, "Context and Location: Framework for security and privacy in automotive telematics," in *Proceedings of the 2nd international workshop on Mobile commerce*, 2002.

[22] R. Agrawal and R. Srikant, "Privacy preserving data mining," in *Proceedings of the ACM International Conference on Management of Data*, 2000, pp. 439–450.

[23] C. K. Liew, U. J. Choi, and C. J. Liew, "A data distortion by probabality distribution," *ACM Transactions on Database Systems*, vol. 10, no. 3, pp. 395–411, 1985.

[24] N. Minsky, "Intentional resolution of privacy protection in database systems," *Communications of the ACM*, vol. 19, no. 3, pp. 148–159, 1976.

[25] D. Agrawal and C. C. Aggarwal, "On the design and quantification of privacy preserving data mining algorithms," in *Proceedings of the 20th ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, 2001, pp. 247 – 255.