Location Privacy in Sensor Networks Against a Global Eavesdropper

Kiran Mehta Donggang Liu Matthew Wright
iSec Laboratory
Dept. of Computer Science and Engineering
The University of Texas at Arlington
Arlington, TX 76019
{kkm7320,dliu,mwright}@uta.edu

***Abstract*— While many protocols for sensor network security provide confidentiality for the content of messages, contextual information usually remains exposed. Such information can be critical to the mission of the sensor network, such as the location of a target object in a monitoring application, and it is often important to protect this information as well as message content. There have been several recent studies on providing location privacy in sensor networks. However, these existing approaches assume a weak adversary model where the adversary sees only local network traffic. We first argue that a strong adversary model, the *global eavesdropper*, is often realistic in practice and can defeat existing techniques. We then formalize the location privacy issues under this strong adversary model and show how much communication overhead is needed for achieving a given level of privacy. We also propose two techniques that prevent the leakage of location information: *periodic collection* and *source simulation*. Periodic collection provides a high level of location privacy, while source simulation provides trade-offs between privacy, communication cost, and latency. Through analysis and simulation, we demonstrate that the proposed techniques are efficient and effective in protecting location information from the attacker.**

I. Introduction

A wireless sensor network (WSN) typically comprises a large number of cheap, small, and resource-constrained sensors that are self-organized as an ad-hoc network to interact with and study the physical world. Sensor networks can be used in applications where it is difficult or infeasible to set up wired networks. Examples include target tracking, habitat monitoring, and military surveillance. These applications are subject to a variety of security issues in hostile environments.

Most of the efforts to date in sensor network security have focused on providing classic security services such as confidentiality, authentication, integrity, and availability. While these are critical requirements in many applications, they are not sufficient. The communication patterns of sensors can, by themselves, expose a great deal of contextual information. For example, delivering sensor data to the base station may disclose the locations of some critical events in the field, revealing valuable intelligence.

In hostile environments, it is particularly important to guarantee location privacy; failure to protect location-based information can completely undermine network applications. For example, in military applications, disclosure of the locations of soldiers due to nearby sensors communicating with the base station may allow an opposing force to launch accurate attacks against them.

Providing location privacy in a sensor network is extremely challenging. On the one hand, an adversary can easily intercept the network traffic due to the use of a broadcast medium for routing packets. He can then perform traffic analysis and identify the source node that initiates the communication with the base station. This can reveal the locations of critical and high-value objects (e.g., soldiers) being monitored by the sensor network. On the other hand, the resource constraints on sensor nodes make it very expensive to apply traditional anonymous communication techniques for hiding the communication from a sensor node to the base station.

A number of privacy-preserving routing techniques have been developed recently for sensor networks. However, these existing solutions can only be used to deal with adversaries who have only a local view of network traffic. A highly motivated adversary can easily eavesdrop on the entire network and defeat all these solutions. For example, the adversary may decide to deploy his own set of sensor nodes to monitor the communication in the target network. This is particularly true in a military or industrial spying context where there are strong incentives to gain as much information as possible from observing the traffic in the target network. Given a global view of the network traffic, the adversary can easily infer the locations of monitored objects. For example, the sensor node that initiates the communication with the base station should be close to the location of the object.

In this paper, we focus on privacy-preserving communication methods in the presence of a *global eavesdropper* who has a complete view of the network traffic. The contributions in this paper are two-fold.

- We point out that the assumption of a global eavesdropper who can monitor the entire network traffic is realistic for some applications. We also formalize the location privacy issues under this assumption and provide bounds on how much communication overhead is needed to achieve a given level privacy.
- We propose two techniques that prevent the leakage

1-4244-1588-8/07/$25.00 ©2007 IEEE

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

of location information: *periodic collection* and *source simulation*. These two schemes are both very effective at hiding the source sensors that initiate communication with the base station. We analyze their effectiveness and evaluate their communication overhead in both analysis and simulation.

Our two schemes for protecting location privacy have distinct properties that make them suitable for different applications. The periodic collection method ensures a high level of location privacy by making every sensor node periodically generate cover traffic. The source simulation method provides trade-offs between privacy, communication overhead, and latency by simulating the behavior of real objects at multiple places in the field to confuse adversaries. We also show how these two schemes can be integrated together to meet the requirements of multi-application networks.

The rest of the paper is organized as follows. The next section reviews existing algorithms for providing location privacy in sensor networks. Section III presents the network and adversary models. Section IV formalizes the privacy issues and gives the privacy evaluation model. Section V discusses the proposed techniques for location privacy. Section VI evaluates the proposed techniques via simulation study. Section VII concludes this paper and points out some future directions.

II. Existing Approaches

In this section, we describe previously-proposed algorithms for source location privacy in wireless sensor networks. These algorithms were designed to protect real objects in the field from a local eavesdropper by increasing the *safety period*, which is defined as the number of messages initiated by the current source sensor before the monitored object is traced [6].

The *flooding technique* [11] has the source node send out each packet through numerous paths to the base station to make it difficult for an adversary to trace the source. However, the problem is that the base station will still receive packets from the shortest path first. The adversary can thus quickly trace the source node. This method consumes a significant amount of energy without providing much privacy in return.

Kamat et al. describe two techniques for location privacy. First, they propose *fake packet generation* [6], which has the base station create fake sources whenever a sender notifies the base station that it has real data to send. These fake senders are away from the real source and approximately at the same distance from the base station as the real sender. Both real and fake senders start generating packets at the same instance. This scheme provides decent privacy against a local eavesdropper. Their other technique is *phantom single-path routing*, which achieves location privacy by making every packet generated by a source walk a random path before being delivered to the base station. As a result, packets will reach the base station following different paths. This algorithm is quite effective in dealing with a local eavesdropper.

Cyclic entrapment [10] creates looping paths at various places in the sensor network. This will cause a local adversary to follow these loops repeatedly and thereby increase the safety period. Energy consumption and privacy provided by this method will increase as the length of the loops increase.

However, all these existing methods assume that the adversary is a local eavesdropper. If an adversary has the global knowledge of the network traffic, it can easily defeat these schemes. For example, the adversary only needs to identify the sensor node who makes the first move during the communication with the base station. Intuitively, this sensor node should be close to the location of adversaries' interest. In this paper, we will focus on privacy-preserving techniques against a global eavesdropper.

III. Network and Adversary Model

Although prior research has attempted to solve location privacy problems for sensor networks, prior attacker models are not strong enough when we consider a well-funded, motivated adversary. In this section, we describe the network and adversary models that we study in this paper.

A. Network Model

Sensor networks are a relatively recent innovation. There are a number of different types of sensor nodes that have been and continue to be developed [5]. These range from very small, inexpensive, and resource-poor sensors such as SmartDust up to PDA-equivalent sensors with ample power and processing capabilities such as Stargate. Applications for networks of these devices include many forms of monitoring, such as environmental and structural monitoring or military and security surveillance.

In this paper, we consider a *homogeneous network model*. In the homogeneous network model, all of the sensors have roughly the same capabilities, power sources, and expected lifetimes. This is a common network architecture for many applications today and will likely continue to be popular moving forward. It has been well-studied and provides for relatively straightforward analyses in research as well as simple deployment and maintenance in the field.

B. Adversary Model

For the kinds of wireless sensor networks that we envision, we expect highly-motivated and well-funded attackers whose objective is to learn sensitive location-based information. This information can include the location of the events detected by the target sensor network such as the presence of a panda.

The Panda-Hunter example application was introduced in [6], and we will also use it to help describe and motivate our techniques. In this application, a sensor network is deployed to track endangered giant pandas in a bamboo forest. Each panda has an electronic tag that emits a signal that can be detected by the sensors in the network. A clever and motivated poacher could use the communication in the network to help him discover the locations of pandas in the forest more quickly and easily than by traditional tracking techniques. Since a single piece of panda fur sold in Chongqing, China for $66,500 in 2003 [8], poachers should be thought of as highly-motivated attackers who could be willing to invest the technical time and

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

money to create efficient ways to track their prey. Similarly, attackers in a military or industrial spying context would have strong, potentially life-or-death, incentives to gain as much information as possible from observing the traffic in system.

In this paper, we consider *global eavesdroppers*. For a motivated attacker, faster and more effective location identification can be done through eavesdropping on the entire network. While an array of targeted antennae may be possible, a simple way for the attacker to do this would be to deploy his own sensor network to monitor the target network. Note that, at the current price for a BlueRadios SMT Module at $25, the attacker needs only $25,000 to build a network of 1000 nodes [1]. Further, the number of nodes can typically be smaller than the the number of nodes in the target network as they monitor wireless radio signals instead of directly sensing the environment. Thus, for even moderately valuable location information, this can be worth the cost and trouble.

Although such an eavesdropping sensor network would face some system issues in being able to report the precise timing and location of each target network event, we do not believe that these would keep the attacker from learning more approximate data values. This kind of attacker would be able to query his own sensor network to determine the locations of observed communications. He could have appropriate sensors send beacon signals that could then be physically located. He could equip his sensors with GPS to get precise location information, or use localization algorithms that avoid the costs of GPS [9], [13].

In any case, it should be feasible to monitor the communication patterns and locations of events in a sensor network via global eavesdropping. An attacker with this capability poses a significant threat to location privacy in these networks, and we therefore focus our attention to this type of attacker.

IV. Privacy Evaluation Model

In this section, we formalize the location privacy issues under the global eavesdropper model. In this model, the adversary deploys an *attacking network* to monitor the sensor activities in the target network. We consider a powerful adversary who can eavesdrop the communication of every sensor node in the target network. Every sensor node i in the target network is an *observation point*, which produces an observation (i, t, d) whenever it transmits a packet d in the target network at time t. In this paper, we assume that the attacker only monitors the wireless channel and the contents of any data packet will appear random to him. This means that the meaningful information in each observation is only the node ID and the observation time. We thus simplify the observation to (i, t). We also assume that the sensor network starts operation at time 0.

Note that in this paper, we assume that an adversary cannot compromise any sensor node. While this is true for some applications, there are also scenarios where the adversary is able to compromise a few sensor nodes in the field. Compromising sensor nodes certainly allows the adversary to identify the locations of the objects more effectively. However, dealing with compromised sensor nodes is beyond the scope of this paper, and we will seek solutions to this issue in the future.

A. The Attacker

Let $O_{i,T}$ be the set of all observations made about node i by time T. Clearly, at time T, the knowledge that an attacker can obtain from eavesdropping the target network is

$$O_T = \bigcup_{i \in I} O_{i,T}$$

where I is the set of node IDs in the target network. The goal of the adversary is to *identify a set $S_T \subset I$ of nodes that represent the set of possible locations for the objects sensed by the target network.* This set indicates that the adversary believes that the objects being observed are close to some of the nodes in S_T at time T. Informally, the attacker will not believe that a lone observation (i, t) indicates the presence of an object. The object should generate a *trace*, which is a set of observations over the lifetime of the network up to time T. More precisely, for each $i \in S_T$, there must exist a set $A_i \subset O_T$ that can be exactly generated by an object whose position at time T is in the signal range of node i. Let us call a set of observations a *candidate trace* if the set could have been generated by the detection of a real object, according to the adversary's perspective.

For the attacker to determine whether a set of observations is a candidate trace, we define a *pattern analysis* function

$$f : 2^{\ddot{O}_T} \rightarrow I \cup \{\perp\}$$

where $\ddot{O}_T$ is the set of all possible observations, i.e., $\ddot{O}_T = \{(i, t)\}_{i \in I, 0 \leq t \leq T}$. This function returns the identity of the location of the object at time T, if the set of observations is a candidate trace, and returns $\perp$ otherwise. For simplicity, we assume that the pattern analysis does not return fractional values, e.g. a probabilistic measure of the chance that a trace is a candidate trace or not. We say that a pattern analysis function is *perfect* if it can identify all candidate traces without error, i.e. without false positives or false negatives. In this paper, we consider a strong adversary who uses a perfect pattern analysis function. Let f_p be such perfect pattern analysis function. We have

$$S_T = \{i \mid \exists A_i \subseteq O_T, (i = f_p(A_i)) \neq \perp\}.$$

B. Measuring Privacy

Privacy can be measured by the size of S_T. We assume that the nodes in S_T are equally likely to be the real objects. Let C be the number of real objects. The probability of any node in S_T being a real object can be estimated by $\frac{C}{|S_T|}$. Hence, we formally define the privacy of our system as the entropy

$$b = \sum_{S_T} -\frac{1}{|S_T|} \log_2 \frac{C}{|S_T|} = \log_2 \frac{|S_T|}{C}$$

We can use this notion to define optimal privacy. Let $\ddot{S}_T$ represent the set of all possible locations for the real object at time T based on the set of all possible observations $\ddot{O}_T$, i.e.,

$$\ddot{S}_T = \{i \mid \exists B_i \subseteq \ddot{O}_T, (i = f_p(A_i)) \neq \perp\}.$$

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

For simplicity, we always assume that the real object can be anywhere in the deployment field at time T. We thus have $\ddot{S}_T = I$. We then define set $S_T^* = I$ for optimal privacy, representing the case that $\forall i \in I$, $\exists A_i \subseteq O_T$ $s.t.$ $(i = f_p(A_i)) \neq \perp$. In other words, there is a subset of observations in O_T (the set of observations made by the adversary by time T) that support the case for a real object in the range of any sensor i. Let N be the network size ($N = |I|$); we have

$$b \leq \log_2 \frac{|S_T^*|}{C} = \log_2 \frac{N}{C}.$$

The level of location privacy is measured in terms of the number of bits. Depending on the users and applications, this can be easily modified to support different kinds of privacy measurement models. For example, we can define high, medium and low privacy levels using appropriate values of b.

We note that the level of privacy can change over time. Typically, the privacy would go lower if the attacker determines that a particular trace is no longer a candidate trace (decreasing S_T without changing C). Somewhat surprisingly, however, the level of privacy can increase. For example, if one candidate trace splits into two candidate traces, then the level of privacy goes up because S_T grows. The meaning of this depends on the application and the attacker model. For example, if the attacker seeks to physically destroy the object being observed with a missile (instant attack), then the privacy should be taken as the minimum at any time before T. In cases where the attacker must spend time to investigate candidate locations, then the average privacy over time is adequate. We provide a snapshot of the privacy at a given time, which can be used for either purpose.

C. Privacy and Communication Costs

In the following, we explore the relationship between the level of privacy and the amount of communication overhead. Let X_T be a random value that represents the number of observations generated for a real object by time T and let $E(X_T) = \epsilon_T$. This represents the number of packets generated for a real object by time T. For a given sensor node $j \in S_T$, with corresponding candidate trace A_j, we will have $E(|A_j|) = E(X_T) = \epsilon_T$.

Theorem 1: To achieve b bits of privacy, the average communication cost is at least

$$\omega_T = \frac{2^b \times C}{(2^b \times C - 1) \times p + 1} \times \epsilon_T$$

where p is the probability of an observation being included in a candidate trace.

Proof: Let $S_T = \{s_1, ..., s_l\}$ be the candidate set identified by the adversary. We have $b = \log_2 \frac{l}{C}$ and thus $l = 2^b \times C$. For any $s_i \in S_T$, let A_i denote the corresponding candidate trace. Let $\mathcal{A} = \cup_{i=1}^{l} A_i$. We know that the communication cost at time T can be estimated by $|\mathcal{A}|$. Note that the probability of an observation being included in another candidate trace is p. We divide the sum of the number of observations from each trace by the average number of times each observation would be counted to obtain:

$$E(|\mathcal{A}|) = \frac{\sum_{i=1}^{l} E(|A_i|)}{p \times (l-1) + 1} = \frac{2^b \times C}{(2^b \times C - 1) \times p + 1} \times \epsilon_T$$

■

Theorem 1 shows the relation between privacy and cost. It tells us the minimum average communication overhead needed to achieve certain privacy. We call a privacy-preserving solution as an *optimal solution* if it can always achieve a given level of location privacy with the minimum communication cost given in Theorem 1.

Note that in applications where the defender only needs to make a few candidate traces in a large sensor network, p is usually very small and negligible. In this case, the communication overhead can be approximated by $2^b \times C \times \epsilon_T$. That is, the overhead increases linearly with the number of candidate traces in the network.

We also note that to achieve a certain level of location privacy without increasing the number of traces, the defender can make p larger. In other words, the defender can have the candidate traces share as many observations as possible. This can be done, for example, by increasing the amount of time that dummy packets are queued at each sensor node. Similar to a technique from anonymous communications, some sensor nodes could wait for several (fake) packets to arrive and forward one instead of many of them. While this would increase p, it would impose larger latencies for data delivery.

To characterize this effect, we simplify our model to understand the effect of different policies on the costs and location privacy in the network. We model the communication in sensor networks as a discrete time system with a granularity of Δ. Specifically, the time line is divided into a number of time intervals with equal length of Δ. The communication between sensor nodes happens at the end of each time interval, i.e., at time $\{\Delta, 2\Delta..., i \times \Delta, ...\}$. A sensor node can receive all the packets targeted to itself and will send or forward no more than one packet at any time interval. Clearly, when a sensor node receives multiple dummy packets during a given interval, it only needs to forward one of them to save the cost. Intuitively, the larger the value of Δ, the more communication cost we can save.

With the above simplification, we model candidate traces as random sets of observations that are picked from all possible observations $\ddot{O}_T^{\Delta} = \{(i, j \times \Delta)\}_{i \in I, 0 \leq j \leq \frac{T}{\Delta}}$ that make the function f_p return the node IDs in I. In other words, the candidate traces are randomly picked from the set

$$\Omega_T^{\Delta} = \{B \mid B \subseteq \ddot{O}_T^{\Delta}, f_p(B) \neq \perp\}.$$

For a candidate trace Y that is random picked from this set, we assume that any observation is equally likely to be included in Y. We further assume that a packet is sent for each observation in a candidate trace. The following lemma uses this model to estimate the value of p.

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

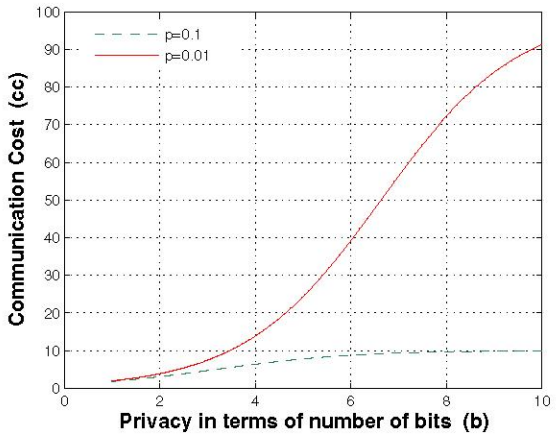


Fig. 1. Relationship between privacy and communication cost. The communication cost is measured by $cc = \frac{\omega_T}{\epsilon_T}$, i.e., the ratio of total messages to messages due to real objects.

Lemma 1: The probability that an observation is included in a candidate trace is given by:

$$p = \frac{\Delta \times \epsilon_T}{N \times T}$$

Proof: Given the assumption of a set of randomly chosen candidate traces, the probability that a candidate trace includes an observation of a given node at a given time interval is the same as the probability that a random $Y \in \Omega_T^{\Delta}$ includes such observation, which can be estimated by $\frac{\epsilon_T}{|\ddot{O}_T^{\Delta}|}$. Since $|\ddot{O}_T^{\Delta}| = |I| \times \frac{T}{\Delta}$, we have $p = \frac{\Delta \times \epsilon_T}{N \times T}$. ■

With Theorem 1 and Lemma 1, we can see that the average communication overhead needed to achieve b-bit of privacy can be re-written as

$$\omega_T = \frac{2^b \times C \times N \times T}{(2^b \times C - 1) \times \Delta + \frac{N \times T}{\epsilon_T}} \quad (1)$$

This tells us that the defender needs to *make ϵ_T, the average number of observations of real objects, as small as possible to save communication overhead*. As a result, the only remaining way for the defender to save communication overhead is to have a large Δ. This, however, is undesirable in many cases since the latency of a real data report reaching the base station may become very large.

In most applications, the traffic generated by a real object increases linearly with T. Hence, we can write $\epsilon_T = \alpha \times T$ for some constant α. Thus, $p = \frac{\Delta \times \alpha}{N}$. As a result, p is often a constant value for a given network.

Figure 1 shows the relationship between privacy and communication cost under a global eavesdropper for different values of p. When p is relatively low, increasing privacy requires significant increases in communication overheads. Effectively, each observation belongs to only a few candidate traces, so there are fewer possible object locations unless many traces are generated. When p is higher, there are more possible locations per trace, and we see that less overhead is needed to provide increasing privacy.

V. Privacy-Preserving Routing Techniques

This section presents two techniques for privacy-preserving routing in sensor networks, a *periodic collection* method and a *source simulation* method. The periodic collection method achieves the optimal location privacy but can only be applied to applications that collect data at a low rate and do not have strict requirements on the data delivery latency. The source simulation method provides practical trade-offs between privacy, communication cost, and latency; it can be effectively applied to real-time applications.

In this paper, we assume that all communication between sensor nodes in the network is protected by pairwise keys so that the contents of all data packets appear random to the global eavesdropper. Many key pre-distribution protocols can be used for our purpose [2], [4], [7]. We also use the technique from [3] to randomize the ID of the destination node in every packet. This prevents the adversary from correlating different data packets to trace the real object.

A. Periodic Collection

As described in Section II, existing privacy-preserving techniques fail against a global eavesdropper. The primary reason is that the presence of a real object in the network will change the traffic pattern, and the global eavesdropper can easily pinpoint where the change happens. An intuitive solution to this problem is to make the network traffic patterns independent of the presence of real objects. To achieve this, our periodic collection method has every sensor node independently and periodically send packets at a reasonable frequency regardless of whether there is real data to send or not. Obviously, the traffic pattern will be independent from the behavior of real objects in the field.

To enable this, every sensor node has a timer which triggers an event every τ seconds, as well as a first-in-first-out (FIFO) queue of size q for buffering received packets that carry real data reports. When an event is triggered by the timer, the node checks if it has any packet in its queue. If so, it dequeues the first packet, encrypts it with the pairwise key it shares with the next hop, and forwards it to that next node. Otherwise, it sends a *dummy packet*, with a random payload, that will not correctly authenticate at the next hop. Since every sensor node only accepts the packets that correctly authenticate, dummy packets do not enter the receiver's queue. When the queue at a sensor node is full, it will stop accepting new packets.

1) Privacy: The periodic collection method provides the optimal location privacy one can ever achieve in the network since the traffic pattern is entirely independent of the activity of real objects. We now present this argument more formally.

Theorem 2: The privacy achieved by the periodic collection method is

$$b = \log_2 \frac{N}{C}$$

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

Proof: Since the object can be anywhere in the field at time T, we know that for any $i \in I$, there exists a candidate trace $A_i \subset \ddot{O}_T$ with $f_p(A_i) = i$. Now, we consider a real object that can generate a candidate trace A_i. If such object is present in the field, it will also produce a set of observations A'_i. According to the definition of f_p, we have $f_p(A'_i) = i$. Since the traffic does not change no matter how the object behaves in the network, we know that $A'_i \subset O_T$. This indicates that $S_T = I$. Hence, we have

$$b = \log_2 \frac{|S_T|}{C} = \log_2 \frac{N}{C}.$$

■

2) Energy consumption: It is generally believed that communication in sensor networks is much more expensive than computation [12]. For a privacy-preserving routing technique, its energy consumption can thus be measured by the additional communication overhead used for hiding the traffic carrying real data.

Since the network starts operation at time 0, the total number of data packets transmitted in the network can be estimated by $\frac{T \times N}{\tau}$. Certainly, a small τ indicates a large amount of additional traffic for our periodic collection method. This means that this method cannot handle real-time applications very well. However, we do believe that it is practical for applications where τ can be set large enough for a reasonable amount of covering traffic.

We now estimate the minimum communication overhead needed to achieve the optimal privacy using Theorem 1. Assume that each real object generates one packet per τ seconds. Let d be the average hop distance from the real object to the base station. The average communication overhead ϵ_T is at least $\frac{d \times T}{\tau}$ since for every real report that reaches the base station, there an average of d nodes are needed to forward packets for the report. According to Equation (1) for calculating the optimal overhead ω_T, since $\Delta = \tau$ and $b = \log_2(\frac{N}{C})$, we have:

$$\begin{aligned} \omega_T &= \frac{2^{log_2 \frac{N}{C}} \times C \times N \times T}{(2^{log_2 \frac{N}{C}} \times C - 1) \times \tau + \frac{N \times T}{(d \times T)/\tau}} \\ &= \frac{\frac{N}{C} \times C \times N \times T}{(\frac{N}{C} \times C - 1 + \frac{N}{d}) \times \tau} \\ &\approx \frac{N \times T}{(1 + \frac{1}{d}) \times \tau} \end{aligned} \quad (2)$$

For a large network, the average hop distance d to the base station can be very large. In this case, we have

$$\omega_T \approx \frac{N \times T}{\tau},$$

which is the overall overhead of the periodic collection method as we discussed before. This shows that for a large network, the performance of periodic collection is very close to the optimal solution in terms of the communication overhead needed to achieve the optimal privacy.

3) Latency: Sensor networks can support a wide range of applications. Different applications may have different requirements that may affect the application of the periodic collection method in real-world scenarios. Example of these requirements include *the latency of a real event being reported to the base station* and *the network lifetime*.

Obviously, parameter τ determines the lifetime of the network. Hence, it should be set judiciously. In general, an application that demands very low latency should set τ to a low value. However, this would lead to high energy consumption as a large number of fake packets will also be generated. Hence, there is a trade off between energy consumption and latency depending on the value of τ.

The queue size q is a factor which determines the number of real packets that a sensor node can buffer. This will affect how well the periodic collection method can handle the situation when the real events are generated frequently in the network. Increasing the value of q will allow the queuing of more real packets and thus will help the network in forwarding more information about real objects to the base station. In other words, the number of packets dropped during the travel to the base station can be reduced. However, we have to realize that having a large q may increase the average latency of a real packet reaching the base station. This occurs because a newly received packet that carries real data may need to wait for a long time before getting forwarded in case of a large queue. We give a detailed simulation study of the effects of different values for parameter q in Section VI.

B. Source Simulation

Though the periodic collection method provides the optimal location privacy, it consumes a substantial amount of energy for applications that have strict latency requirements. It is clearly not well-suited for real-time applications.

In the periodic collection method, every sensor node is a potential source node. To reduce energy consumption, we choose to reduce the number of potential sources in the network. In other words, we will trade off privacy with communication overhead. For this purpose, we propose to *create multiple candidate traces in the network to hide the traffic generate by real objects*. How to determine the number of candidate traces is usually application-dependent. In general, we expect that this number is much smaller than the size of I.

Creating candidate traces in the field is quite challenging in practice. The main problem lies in the difficulty of modeling the behavior of a real object in the field. A poorly-designed model will likely fail in providing privacy protection. For example, as shown in Figure 2, the behavior of a panda is modeled *inaccurately* as a static object in the network. Therefore, the candidate traces are created at places $\{F_1, F_2, ..., F_6\}$. Each of these place will send virtual traffic to the base station, simulating a real panda.

However, the actual panda behavior may be quite different from the model used by the defender. The adversary may notice how the panda moves around in the field. This gives the adversary extra knowledge to identify the possible locations of

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

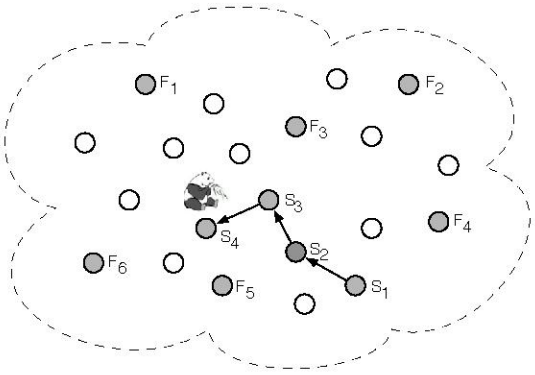


Fig. 2. Movement pattern leaks the location of the panda

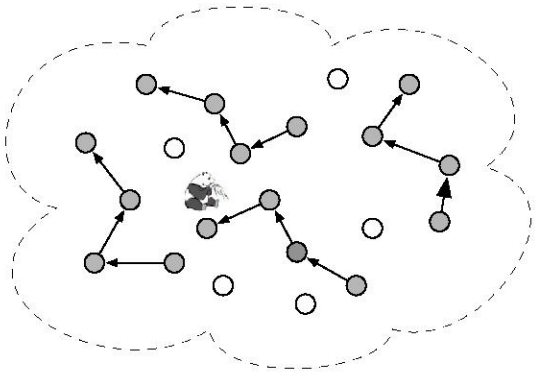


Fig. 3. Simulating virtual objects in the field

pandas. As shown in Figure 2, the panda moves from S_1 to S_4 along the path $\{S_1, S_2, S_3, S_4\}$. In this case, the traffic generated by the panda exhibits a sequential pattern from S_1 to S_4. As a global eavesdropper, it can then quickly determine that the traffic generated from places $\{F_1, F_2, ..., F_6\}$ is likely to be virtual, and the traffic generated from $\{S_1, ..., S_4\}$ is likely to be generated by a real panda. As a result, the adversary can easily tell that the panda is currently close to position S_4.

Fortunately, in most cases, an adversary will not have substantially more knowledge about the behavior of real objects than the defender. Even if the attacker learns about the object behavior over time, the defender will learn the same behavior and can broadcast occasional updates to the object movement model. Thus, it is often reasonable to assume that the adversary and the defender have similar knowledge about the behavior of real objects. We can then create more useful candidate traces in the field to hide real objects. Though it is challenging to model real objects, research on learning and modeling behavior of objects are quite active. We believe that it will not be a very difficult problem to obtain a reasonable behavior model for the object in the field. Modeling of objects is beyond the scope of this paper.

1) Protocol Description: In the source simulation approach, a set of *virtual objects* will be simulated in the field. Each of them will generate a traffic pattern similar to that of a real object. Figure 3 shows the idea of this approach. In this example, pandas move randomly in the field. Both the adversary and the defender have a model of this random movement pattern. After network deployment, each virtual object is treated like a real object, as sensors detect it and send the object's information to the base station. The protocol works in rounds. In the every round, the node simulating the fake panda will randomly pick a sensor node in its neighborhood (including itself) and ask this node to simulate the real panda in the next round. In this way, there will be multiple movement patterns similar to that of real pandas. In Figure 3, there are three such virtual pandas simulating real pandas.

Source simulation then works as follows. Before deployment, we randomly select a set of L sensor nodes and pre-load each of them with a different *token*. Every token has a unique ID. These tokens will be passed around between sensor nodes to simulate the behavior of real objects. For convenience, we call the node holding a token the *token node*. We also assume that the profile for the behavior of real objects is available for us to create candidate traces.

After deployment, every token node will emit a signal mimicking the signal used by real objects for event detection. This will trigger the event detection process in the local area and generate traffic as if a real event was detected. The token node will then determine who in its neighborhood (including itself) needs to simulate the next round of source simulation based on the profile for the behavior of real objects. The token will then be passed to the selected node. The delivery of such token between sensor nodes will be always protected by the pairwise key established between them.

Note that the simulation requests create additional messages that can help the attacker distinguish real objects from virtual ones. To protect against this, we require that nodes that detect the real object also send an extra message in the system each round. Alternatively, we can attach the requests to the messages to the base station, given that all messages would be received by neighbor nodes.

2) Privacy: Assume that the defender can build a model for the behavior of real objects that can always create a useful candidate trace in the network with probability P. In other words, any candidate trace created by the defender in the network will be considered as a valid candidate trace by the attacker with probability P. Let C be the number of real objects in the network. We can see that the set of candidate locations S_T includes an average of $C + L \times P$ node IDs. As a result, the privacy provided by the source simulation approach can be estimated by

$$b = \log_2 \frac{C + L \times P}{C} = \log_2 (1 + \frac{L \times P}{C}).$$

Since we assume that both the adversary and the defender have similar knowledge about the behavior of real objects, we

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

will usually have a value of P that is close to 1. In this case, $b \approx \log_2(1+\frac{L}{C})$.

3) Energy consumption: Since there are L fake sources simulating the real objects in the network, the communication overhead will be increased by a factor of $\frac{L+C}{C}$. Similar to the analysis from Section V-A.2, we use Theorem 1 to estimate the minimum average communication overhead needed to achieve the same location privacy as the source simulation approach under a global eavesdropper.

Note that the source simulation method can be effectively used for real-time applications with privacy protection requirements. We are often required to deliver sensor reports as fast as possible, so we assume that a sensor node will immediately forward the report whenever the channel is free. Hence, Δ is very small ($\Delta << T$). According to Equation (1) for calculating the optimal ω_T, when $b = \log_2(1+\frac{L}{C})$:

$$\begin{aligned}\omega_T &= \frac{2^{log_2(1+\frac{L}{C})} \times C \times N \times T}{(2^{log_2(1+\frac{L}{C})} \times C - 1) \times \Delta + \frac{N \times T}{\epsilon_T}} \\ &= \frac{(1+\frac{L}{C}) \times C \times N \times T}{((1+\frac{L}{C}) \times C - 1) \times \Delta + \frac{N \times T}{\epsilon_T}} \\ &\approx \left(1+\frac{L}{C}\right) \times C \times \epsilon_T \\ &= (L+C) \times \epsilon_T \end{aligned} \tag{3}$$

This shows that the average communication overhead for an optimal solution is approximately increased by a factor of $\frac{L+C}{C}$ to achieve the same location privacy as the source simulation method. Thus, the communication overhead involved in our source simulation approach is very close to the minimum overhead required to achieve the given level of location privacy. Further, it features optimal latencies.

VI. Simulation Model

In this section, we evaluate the performance of our techniques using simulation. We show the performance of the proposed privacy-preserving techniques in terms of energy consumption and latency, and compare our methods with the phantom single-path method [6], a method that is effective only against local eavesdroppers.

In our simulation, we include 5,093 sensor nodes distributed randomly in a square field of 1000×1000 meters. Each sensor node can communicate with other sensor nodes in a radius of 50 meters, while an electronic tag attached to a panda can emit radio signals that can reach sensor nodes within 25 meters. We noticed that, on average, each sensor node has 40 neighbors. As a result, the presence of any panda will be detected by 10 sensor nodes on average. We assume that the base station is located at the center of this field.

The proposed techniques assume a routing protocol for sensor networks, though the choice of routing protocol does not affect our results. For simplicity, we adopt a simple widely-used routing method [3]. In this method, paths are constructed by a beacon packet from the base station. Each node, on receiving the beacon packet for the first time, sets the sender of such beacon packet as its parent. In this way, each node will likely select a parent that is closest to the base station.

For the purpose of simulation, we assume that the network application only needs to detect the locations of pandas and always wants to know the most recent locations. We thus have every sensor node drop a new packet if it has already queued a packet that was generated on the same event.

In our simulation, we assume that the adversary has deployed a network to monitor the traffic in the target network. Specifically, he is able to locate every sensor node in the target network and eavesdrop every packet this node delivers. Though the adversary may face engineering problems in developing methods to collect these observations from its network, we do not believe that this will be a very difficult issue to address. For simplicity, we assume the adversary can always reliably collect all the observations in the network.

We ran each simulation for 6,000 time intervals with τ seconds for each time interval. We selected the initial locations for real pandas randomly. In the experiments, the tag attached to a real panda emits a signal, which will generate an event, for detection at a rate of one per $10 \times \tau$ seconds (10 time intervals). In addition, every panda moves from the current location (x, y) to a random location $(x + a_1, y + a_2)$ every $10 \times \tau$ seconds, where a_1 and a_2 are two random values uniformly selected between 0 and 60.

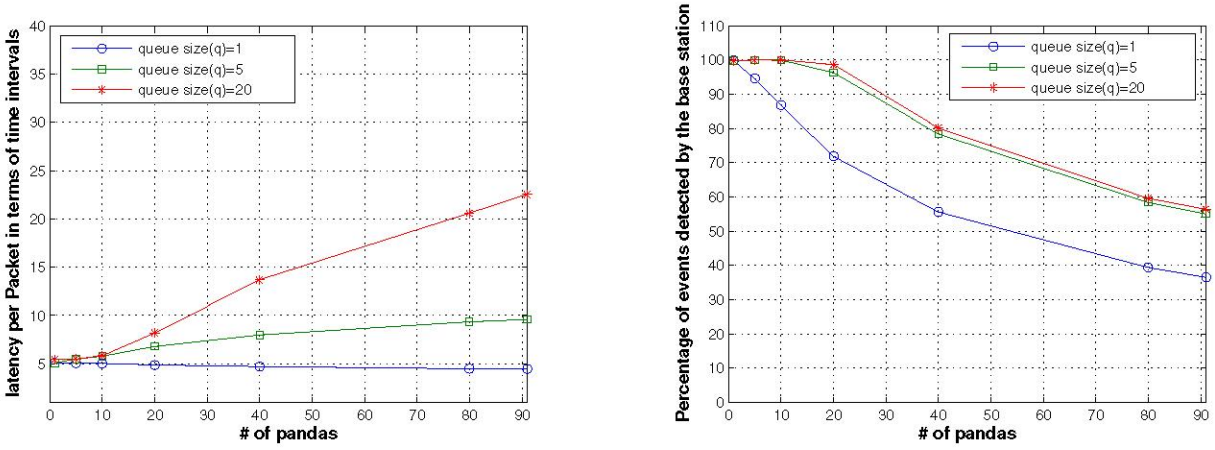
A. Periodic Collection

The analysis of Section V has already shown that the periodic collection method achieves the optimal location privacy. In addition, the communication overhead in the network remains constant and is independent of both the number of pandas and their patterns of movement. Hence, the focus of our simulation evaluation is on the latency and the packet drop rate when there are multiple pandas in the field.

Figure 4(a) shows the latency of packet delivery when there are multiple pandas in the field. We can see that as the number of pandas increases, the latency increases. This is due to nodes close to the base station receiving multiple reports at the same time, therefore requiring them to buffer packets. When the number of pandas grows too large, the buffered packets start being dropped due to the limited size of queue, while the latency of the packets that do arrive at the base station becomes stable after a certain point. When the queue size q decreases, packets traveling long distances have a high probability of getting dropped, making the latency of the packets that do arrive at the base station smaller. This can be seen by a drop in the latency for smaller values of q.

Figure 4(b) shows the percentage of the detected events received by the base station. We can see that the percentage of events received decreases when there are more pandas in the field. Increasing q will certainly increase the percentage of the events forwarded to the base station. However, after a certain point, having a larger q will not offer significant benefit in terms of the packet drop rate. For example, increasing the queue size from 5 to 20 does not help much in forwarding

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.



(a) Latency

(b) Percentage of event detected

Fig. 4. Performance of periodic collection when there are multiple real objects.

more events to the base station. On the other hand, we have already shown in Figure 4(a) that increase q will significantly increase the latency of packet delivery. This is undesirable in some cases. As a result, we believe that q should typically be configured as small as possible while still meeting the requirements of the event drop rate. Overall, the results in Figure 4(a) and Figure 4(b) give a guideline for configuring the queue size q to meet various requirements.

B. Source Simulation

According to the analysis in Section V, the location privacy achieved by source simulation is determined by the number of virtual sources simulated in the network. Thus, the focus of our simulation evaluation is on how much communication cost we have to pay to achieve a given level of location privacy. We use these results to illustrate the efficiency of this technique.

During the simulation, we assume that there is only one panda in the network. Multiple fake pandas are created and simulated in the field. The initial positions of the fake pandas are randomly selected. In addition, we assume that the sensor network is deployed to handle real-time applications. In other words, whenever a sensor node receives a packet, it will forward it to the next hop as soon as possible. We assume that in τ seconds (one time interval), a sensor can process and forward at most one hundred packets from the queue, which is a hundred times faster than the processing speed of the periodic collection method. We set P to 1, which means that the defender and the adversary have the same knowledge about the pandas' behavior.

Figure 5 shows the communication cost involved in our source simulation method to achieve a given level of privacy. We can see that the communication overhead increases as the privacy requirement increases. We also note that the communication overhead is very close to the performance of the optimal solution, which can be derived from Theorem 1. This further validates our analysis and shows that the source simulation method is effective and efficient for achieving privacy in real-time applications. This figure also includes the performance of other approaches for further comparison, which we explain below.

C. Comparison

We now compare the proposed approaches in this paper with previous privacy-preserving techniques–in particular, the phantom single-path routing technique [6]. We focus on location privacy and communication overhead in the following comparison. The simulation result is shown in Figure 5. The performance of the phantom single-path routing is represented by a single point at the left-bottom corner of the figure, and the performance of the periodic collection method is represented by a single point on the right part of the figure.

In terms of privacy, we have already shown that none of the previous methods (including phantom single-path routing) can provide location privacy under the assumption of a global eavesdropper. In contrast, both of our methods provide location privacy against a global eavesdropper. The periodic collection method provides the highest level of privacy and is suitable for applications that collect data at a low rate and do not require real-time data delivery, while the source simulation method can support real-time applications with practical trade-offs between privacy, communication overhead and latency.

We compare the communication overheads through simulation. Figure 5 shows the communication costs involved in different methods. The simulation results are not surprising. The phantom single-path routing technique only introduces small communication overhead, while the periodic collection method involves significant but constant communication cost for a given period of time. The source simulation method is in the middle of these two schemes; it can provide practical trade-offs between privacy and communication cost. We notice that in the figure, the periodic collection method requires less communication overhead to achieve privacy of around $b = 12$ bits when compared with the source simulation method. The

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.

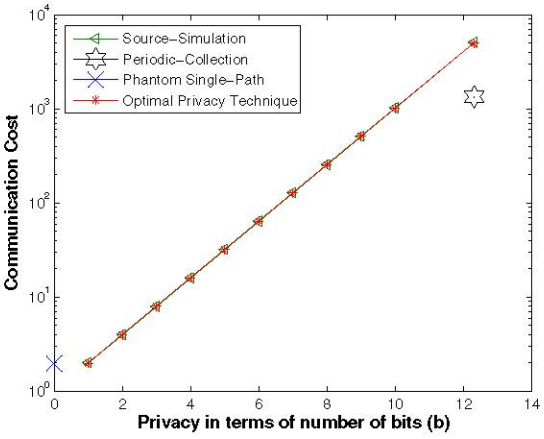


Fig. 5. Communication cost (i.e., the ratio of total messages to messages due to real objects) needed to achieve a given level of privacy for different schemes.

reason is that the source simulation method is configured to support real-time applications with $\Delta \approx \frac{\tau}{100}$ seconds, while for the periodic collection, we have $\Delta = \tau$ seconds.

VII. Conclusions

Prior work on location privacy in sensor networks had assumed that the attacker has only a local eavesdropping capability. This assumption is unrealistic given a well-funded, highly-motivated attacker. In this paper, we formalize the location privacy issues under the model of a global eavesdropper and show the minimum average communication overhead needed for achieving a given level of privacy. We also presented two techniques to provide privacy against a global eavesdropper. Analysis and simulation studies show that they can effectively and efficiently protect location privacy in sensor networks.

There are a number of directions that worth studying in the future. In particular, in this paper, we assume that the global eavesdropper will not compromise sensor nodes; he can only perform traffic analysis without looking at the content of the packet. However, in practice, the global eavesdropper may be able to compromise a few sensor nodes in the field and perform traffic analysis with additional knowledge from insiders. This presents interesting challenges for both of our approaches. In addition, we are also interested in the implementation of our methods on real sensor platforms and the experimental results from real sensor applications.

Acknowledgment The authors would like to thank the anonymous reviewers for their valuable comments.

References

[1] BlueRadios Inc. Order and price info. `http://www.blueradios.com/orderinfo.htm`. Accessed in February 2006.

[2] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *IEEE Symposium on Security and Privacy (S&P)*, pages 197–213, May 2003.

[3] J. Deng, R. Han, and S. Mishra. Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks. In *2004 International Conference on Dependable Systems and Networks (DSN)*, June 2004.

[4] L. Eschenauer and V. D. Gligor. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM Conference on Computer and Communications Security (CCS)*, pages 41–47, November 2002.

[5] J. Hill, M. Horton, R. Kling, and L. Krishnamurthy. The platforms enabling wireless sensor networks. *Commun. ACM*, 47(6):41–46, 2004.

[6] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk. Enhancing source-location privacy in sensor network routing. In *Proceedings of the 25th IEEE International Conference on Distributed Computing Systems (ICDCS)*, pages 599–608, June 2005.

[7] D. Liu and P. Ning. Establishing pairwise keys in distributed sensor networks. In *Proceedings of 10th ACM Conference on Computer and Communications Security (CCS)*, pages 52–61, October 2003.

[8] Star News. Panda poaching gang arrested. *Shanghai Star Telegram*, April 2003.

[9] D. Niculescu and B. Nath. Ad hoc positioning system (APS) using AoA. In *Proceedings of IEEE INFOCOM*, pages 1734–1743, April 2003.

[10] Y. Ouyang, Z. Le, G. Chen, J. Ford, and F. Makedon. Entrapping adversaries for source protection in sensor networks. In *Proceedings of the International Symposium on on World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 23–34, June 2006.

[11] C. Ozturk, Y. Zhang, and W. Trappe. Source-location privacy in energy-constrained sensor network routing. In *Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks (SASN)*, pages 88–93, October 2004.

[12] A. Perrig, R. Szewczyk, V. Wen, D. Culler, and D. Tygar. SPINS: Security protocols for sensor networks. In *Proceedings of Seventh Annual International Conference on Mobile Computing and Networks (MobiCom)*, July 2001.

[13] A. Savvides, C. Han, and M. Srivastava. Dynamic fine-grained localization in ad-hoc networks of sensors. In *Proceedings of ACM MobiCom*, pages 166–179, July 2001.

Authorized licensed use limited to: Univ of Calif Santa Barbara. Downloaded on November 13, 2008 at 16:41 from IEEE Xplore. Restrictions apply.