Location Privacy Methods in Wireless Sensor Networks

A.C.Priya Ranjani *1 N.Swarna Jyothi *2 B.Sriramulu *3

1Department Of Computer Science and Engineering
Vijaya Institute of Technology for Women, Vijayawada, Andhra Pradesh, India

Abstract: In the last years, wireless sensor networks (WSNs) have gained increasing attention from both the research community and actual users. These networks are likely to be composed of hundreds, and potentially thousands of tiny sensor nodes, functioning autonomously. The emergence of sensor networks as one of the dominant technology has posed numerous unique challenges to researchers. The sensor networks are vulnerable to a vast number of security threats. This security deals with issues like privacy, authentication, confidentiality, integrity and availability. Among these, privacy is of utmost importance. Privacy means each node’s data should be only known to itself. Privacy in WSNs can be classified into two categories content-oriented privacy and context-oriented privacy. Context-oriented privacy protection focuses on protecting contextual information, such as the location and timing information of traffic transmitted in a WSN. Context-oriented privacy comprises of temporal privacy and location privacy. In this paper, we give brief introduction of security threats and survey the major topics related to location privacy in wireless sensor networks and present them.

Keywords: Privacy, Context-privacy, Location privacy, LPR.

I. INTRODUCTION

Wireless sensor network (WSN) consists of spatially distributed autonomous sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants. The development of wireless sensor networks was motivated by military applications such as battlefield surveillance. They are now used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment and habitat monitoring, healthcare applications, home automation, and traffic control. Sensor nodes are great for deployment in hostile environments. In Fig 1, we can how sensor nodes are planted over a geographical area. The data collected from sensor nodes is transmitted to sink and thereby to the actual user.

Fig 1. Sensor nodes scattered in a field

The collected data can either be stored in the network sensors, or transmitted to the sink. Several problems arise when data are stored in sensors. Several problems arise when data are stored in sensors. First, a sensor is equipped with only limited memory or storage space, which prohibits the storage of a large amount of data accumulated for months. Second sensors are battery operated, the stored data will be lost after the sensors are depleted of power. Third, searching for the data of interest in a widely scattered network field is a hard problem. It also has to be implemented at a low cost. Thus, security vulnerability of sensor networks is vastly greater than for other types of networks [1].

Security Requirements in WSN

The objective of confidentiality is required in sensors environment to protect information travelling among the sensor nodes of the network or between the sensors and the base station from disclosure. A sensor network should not leak sensor readings to its neighbors or any other nodes eg., military application [11]. Authentication in sensor networks is essential for each sensor node and base station to have the ability to verify that the data received was really sent by a trusted sender or not. This authentication is needed during the clustering of sensor node in WSN. We can trust the data sent by the nodes in that group after clustering.

Integrity controls must be implemented to ensure that information will not be altered in any unexpected way. The adversary can include forged data or exclude legitimate data. Thus, data integrity ensures that any received data has not been altered in transit. The main requirement for wireless sensor network infrastructure is the availability of the nodes.
as well as the effective availability of the network for a given application. Maintaining privacy in many wireless sensor network applications is a major concern. Each node’s data should be only known to itself. When a sensor network is under a malicious attack, it is possible that some nodes may collude to uncover the private data of other node(s).

Privacy in WSNs can be classified into two categories [3] content-oriented privacy and context-oriented privacy. Context-oriented privacy protection focuses on protecting contextual information, such as the location and timing information of traffic transmitted in a WSN. Context-oriented privacy comprises of temporal privacy and location privacy. The categorization of privacy in WSN’s can be visualized in Fig 2. In this paper, we mainly focus on upholding location privacy at source and the receiver (base station).

II. CONTENT PRIVACY and CONTEXTUAL PRIVACY

Privacy encompasses the right to control information about ourselves, including the right to limit access to that information. Privacy in the context of WSNs involves both privacy of monitored subjects and privacy of nodes and base stations. Privacy in WSNs can be classified into two categories [3] content-oriented privacy (data privacy) and context-oriented privacy. Content-oriented privacy is threatened by an adversary who aims to manipulate and/or read the content of messages sent over a WSN. In contrast, context-oriented privacy is concerned about a protection of contextual information surrounding the content. Typical contextual information is location where the data has been sensed or time of the measurement [3].

Content privacy

Content Privacy or Data-oriented privacy protection focuses on protecting the privacy of data content. Here “data” refers to not only sensed data collected within a WSN but also queries posed to a WSN by users. There are two types of adversaries which may compromise data-oriented privacy. One is an external adversary which eavesdrops the data communication between sensor nodes in a WSN. This type of adversaries can be effectively defended against using the traditional techniques of cryptographic encryption and authentication. The second type is an internal adversary which is also participating node of the WSN, but has been captured and manipulated by malicious entities to compromise private information.

III. CONTEXT PRIVACY SCHEMES IN WSN’S

Context-oriented privacy protection focuses on protecting contextual information, such as the location and timing [4] information of traffic transmitted in a WSN. Location privacy concerns may arise for such special sensor nodes as the data source and the base station. An adversary with knowledge of the location of the data source or base station location may be able to infer the content of the data being transmitted or destroy the sensor network. Timing privacy, on the other hand, concerns the time when sensitive data is created at data source, collected by a sensor node and transmitted to the base station.

A major challenge for context-oriented privacy protection is that an adversary may be able to compromise private information even without the ability of decrypting the transmitted data. In particular, since hop-by-hop transmissions required to address the limited transmission range of sensor nodes, an adversary may derive the locations of base station and data source [5] by observing and analyzing the traffic patterns [8] between different hops (to track down the base station and/or the data source). To address this challenge, the objective of context-oriented privacy protection is to hide the real traffic pattern. Threats against content privacy arise
due to the ability of adversaries to observe and manipulate the content of packets sent over a sensor network. In particular, the location information about senders/receivers may be derived based on the direction of wireless communications. Let us first study about location privacy of data source.

3.1 Location Privacy of Data Source

Before studying the existing techniques for protecting the location privacy of the data source, let us first discuss the example of monitoring Zebras in a habitat. In this application, a large number of zebra-detecting sensors are deployed in a habitat. After sensors detect a zebra, they will generate event messages and transmit them forward them to the base station. Meantime, a hunter also attempts to identify the location of the data source to find the zebra.

The objective of the defender is to hide the location information from being known by the hunters which have the ability to eavesdrop the wireless communication between different sensor nodes. The objective of the hunter is to compromise the location of the data source (and thereby the zebra) by analyzing the traffic flow in the WSN. To retrieve knowledge about the location of the source node, we discuss four techniques, flooding [5], Random walk [5], dummy injection [5,9], and fake data sources [6], against the disclosure of the location of data source in WSNs.

3.1.1 Baseline and probabilistic flooding mechanisms.

The basic idea of this scheme [Fig 3] is for each sensor to broadcast the data it receives from one neighbor to all of its other neighbors. All sensors participate in the data transmission so that it is unlikely for an attacker to track a path of transmission back to the data source [5]. However, the effectiveness of baseline flooding depends on the number of nodes in the transmission path between the data source and the base station. If the path is too short, after an adversary detects the arrival of the first packet at the base station, the adversary can infer that the routing path of this packet is the shortest path between the data source and the base station.

Then, the adversary can track back from the last forwarding sensor along the routing path to the data source. Furthermore, the flooding consumes significant amount of energy in the whole network and hence the lifetime of the WSN may be substantially reduced.

To address the side effect of baseline flooding, probabilistic flooding is proposed in [5], in which not all sensors are involved in forwarding data. Instead, each node forwards/ broadcasts a packet it receives with a pre-determined probability. This scheme not only saves energy but also limits the adversary’s ability to track back to the data source. Nonetheless, the reception of data by the base station is not guaranteed due to the randomness involved in this approach.

3.1.2 Random walk mechanisms.

As one of random walk approaches, Phantom Routing is used [5]. Figs. 3 and 4 illustrate the basic idea of Phantom Routing. Data first performs a few steps of random walk from data source, and then, by employing probabilistic flooding scheme, it is transmitted towards base station.

The advantage of this approach is that even if an adversary is able to track back along the routing path, it would only be able to Fig out the terminal node of the random walk instead of the original data source. Unfortunately, as indicated in [28], the pure random walk approach is not statistically secure for protecting the location of the data source. In particular, we discuss techniques for defense against local and global adversaries.
3.1.3 Dummy data mechanism

To further protect the location of the data source, fake data packets can be introduced along with original packets. In particular, a simple scheme called Short-lived Fake Source Routing was proposed in [18] for each sensor to send out a fake packet with a pre-determined probability. Upon receiving a fake packet, a sensor node just discards it. Although this approach perturbs the local traffic pattern observed by an adversary, it also has limitations on privacy protection. Specifically, to maintain the energy-efficiency of the WSN, the length of each path along which fake data is forwarded is only one hop, therefore, an adversary is able to quickly identify fake paths and eliminate them from consideration. Note that this technique is ineffective against global adversaries that can monitor the transmission rate of each sensor node and thereby identify those that are only sending out dummy data. To address this problem, one possible approach is to globally inject dummy data as well as keeping the transmission of real data the same as that of dummy data. However, this approach may introduce significant delay to the data transmission process.

Two schemes that filter large number of dummy data and reduce energy consumption are introduced. Proxy-based Filtering Scheme (PFS) and Tree based Filtering Scheme (TFS), were proposed in [10] to filter partial dummy data without threatening the source privacy. The whole network is divided into cells. The proxies are responsible for relaying real data from cells around them and filtering dummy data. After filtering all dummy packages from cells and buffering real data, the proxy will send data package, including buffered real data and new generated dummy data, at the same rate of transmission.

3.1.4 Fake data source mechanism.

The basic idea of fake data source is to choose one or more sensor node to simulate the behavior of a real data source in order to confuse the adversaries [6]. The more the fake sources, the better can the identity of real source be protected. Such a technique will also incur more power consumption in the WSN.

3.2 Location Privacy of Receiver

The receiver is the most critical node as it collects data from all sensors. Since all sensors send data to a single node (the receiver), this creates a single point of failure in the network. There are several ways that an adversary can trace the location of a receiver. First, an adversary can deduce the location of the receiver by analyzing the traffic rate. By eavesdropping the packets transmitted [6] at various locations in a sensor network, an adversary is able to compute the traffic densities at these locations, based on which it deduces the location of or the direction to the receiver.

In order to protect the receiver’s location privacy, location-privacy routing protocol (LPR) along with fake packet injection [2] is used. LPR provide path diversity whereas fake packet injection minimizes the information that an adversary can deduce from the overheard packets about the direction towards the receiver. Under such a protection scheme, an adversary can hardly distinguish between real packets and fake packets, or tell which direction is towards the receiver.

Path diversity provided by LPR inevitably leads to longer routing paths, and transmitting fake packets consumes extra energy. The stronger the protection for the receiver is required, the higher the overhead will be.

3.2.1 Location Privacy Routing (LPR)

To support LPR, each sensor divides its neighbors into two lists [2]: a closer list, consisting of neighbors that are closer to the receiver, and a further list, consisting of neighbors that are farther (or at equal distance) from the receiver. Each time the receiver moves to a new position, it broadcasts a beacon packet in the network. This packet carries a hop count whose initial value is zero. When a sensor receives the beacon for the first time, it increments the hop count in the packet by one, records the hop count, and forwards the packet to its neighbors. After the beacon broadcast completes, neighbors exchange their recorded hop counts, based on which they construct their closer/further lists. First, it happens only once a time after the receiver gets to a new position.

An adversary can only make one movement based on this broadcast. Second, due to the assumption that the packet content is protected by an encryption method, the adversary cannot distinguish
between beacon packets and fake packets. After the closer/further lists are built, LPR works as follows.

When a sensor forwards a packet, it selects a neighbor randomly from one of its two lists as the next hop. Because the next hop is randomly chosen, the routing path for packets from the same source node to the receiver is not fixed. On one hand, if the sensors mostly choose their next hops from the closer lists, the routing paths will be short, and the energy efficiency will be good. However, the protection for the receiver’s location privacy becomes weak. On the other hand, if the sensors frequently choose the next hops from the further lists, the energy efficiency will be lower, but the protection for location privacy will be strengthened.

In LPR, each time a sensor forwards a packet, it selects the next hop from the further list with probability \( P_f \) and from the closer list with probability \( 1 - P_f \), where \( P_f \) is a system parameter [2]. By adjusting the value of \( P_f \), one can tune the tradeoff between energy efficiency and location privacy.

Consider a packet that is \( d \) hops away from the receiver. Let \( x_d \) be the expected number of hops the packet has to travel before reaching the receiver. In order for \( x_d \) to be finite, \( P_f \) has to be smaller than 50%. We have the following difference inequality.

\[
x_d \leq x_{d+1} + P_f x_{d+1} + (1-P_f)x_{d-1}
\]

After making one hop, with a probability of \( P_f \), the packet is forwarded to a neighbor further away and has to take up to \( x_{d+1} \) additional hops on average to reach the receiver. With a probability of \( (1-P_f) \), the packet is forwarded to a neighbor in the closer list and has to take \( x_{d-1} \) additional hops on average. Note that the further list includes those neighbors that are at the same distance to the receiver. Therefore, \( P_f x_{d+1} \) is an overestimation for the average additional hops from a neighbor in the further list to the receiver. Consequently, the right side is an upper bound of \( x_d \). Solving the above difference inequality, we have

\[
x_d \leq \frac{x_{d+1}}{1-P_f}
\]

The expected length of the routing path in LPR is \( \frac{d}{1-P_f} \) times that of the shortest path. Below we discuss the strength and weakness of LPR.

**Strength**: In LPR, the next hop from a sensor to receiver is not fixed. Sometimes the next hop does not even point to receiver which makes it harder to perform packet-tracing attack.

**Drawback**: If we apply LPR alone, the protection for location privacy will not be strong enough because the overall traffic trend in the network still points towards the receiver.

### 3.2.2 Fake Packet Injection

The basic idea of fake packet injection is that whenever a sensor node forwards a packet, it also transmits a fake packet to a neighbor that is randomly chosen from its *further list*. Attracted by this fake packet, the adversary may trace to a wrong direction instead of the real next hop.

Each fake packet has a \( TTL_{fake} \) parameter specifying the maximum number of hops it will be forwarded away from the receiver. On one hand, a larger value for \( TTL_{fake} \) will lead to more traffic flowing away from the receiver, increasing the capability of misleading the adversary. On the other hand, a larger value for \( TTL_{fake} \) will also lead to higher energy consumption. It should be emphasized that \( TTL_{fake} \) has to be at least 2 hops. When a node receives a fake packet, it does the following.

1. The node decrements the TTL field of the packet by one.
2. If the TTL field is positive, the node randomly chooses a neighbor from its *further list* and forwards the fake packet to that neighbor.
3. If the TTL field is zero, the node discards the fake packet. The injection of fake packets can effectively enhance the protection of the receiver’s location privacy.

However, the cost is also high. To control the tradeoff between energy consumption and protection strength, parameter \( p_{fake} \) can be used, specifying the probability at which a node generates a fake packet when it forwards a real packet. The higher the value of \( p_{fake} \), the more the number of fake packets that will be generated, and the more the energy that will be consumed.

When LPR is combined with fake packet injection, if the system parameters \( (pf, TTL_{fake}, p_{fake}) \) are appropriately set, then it is very hard for an adversary to perform any analysis based on locally gathered information to infer the direction towards the receiver. A *further* (or *closer*) direction refers to a direction that moves away (or closer to) the receiver.

In Fig 5, node A is the sensor where the adversary resides. Node B is the sensor that forwards a real packet to A. Nodes C1 and C2 are the neighbors to
which A sends the real packet and the fake packet, respectively.

![Fig 5. LPR with fake packets](image)

The adversary can identify the direction of the real packet through the relations among the transmissions made from B to A, from A to C1, and from A to C2. It means that fake packet injection has little effect when working with traditional routing protocols.

However, when LPR is used, the direction to which the real packet is forwarded does not necessarily point towards the receiver. This implies that nodes B, A and C1 are likely to not locate along a line. It is possible that the deviation of C1 from the line of B → A is larger than that of C2, as shown in Fig 5, where R is the receiver. In this case, the adversary can hardly tell which of C1 and C2 is receiving the real packet, he will trace to a wrong way. If the packet itself is fake, which is possible when LPR is applied together with fake packet injection, then no matter where A forwards the packet, the adversary will trace to a wrong way.

4. REFERENCES


