Using Data Mules to Preserve Source Location Privacy in Wireless Sensor Networks

Na Li,[1] Mayank Raj,[1] Donggang Liu[2], Matthew Wright[2], Sajal K. Das[1]

[1] Center for Research in Wireless Mobility and Networking (CReWMaN)
[2] The Information Security Lab (iSec)
Computer Science and Engineering Department,
University of Texas at Arlington
Email: {na.li,mayank.raj}@mavs.uta.edu, {dliu,mwright,das}@uta.edu

Abstract. Wireless sensor networks (WSNs) have many promising applications for monitoring critical regions, such as in military surveillance and target tracking. In such applications, privacy of the location of the source sensor is of utmost importance as its compromise may reveal the location of the object being monitored. Traditional security mechanisms, like encryption, have proven to be ineffective as location of the source can also be revealed by analysis of the direction of traffic flow in the network. In this paper, we investigate the source-location privacy issue. We first propose a semi-global eavesdropping attack model which we show as being more realistic than the local or global eavesdropping attack model discussed in literature. In this model, we use a linear-regression based traffic analysis technique and show that it is effective in inferring the location of the data source under an existing source-location preserving technique. To measure source location privacy against this semi-global eavesdropping, we define an α-angle anonymity model. Additionally, we adapt the conventional function of data mules to design a new protocol for securing source location privacy, called the Mules-Saving-Source (MSS) protocol, which provides α-angle anonymity. We analyze the delay incurred by using data mules in our protocol and examine the association between privacy preservation and data delay in our protocol through simulation.

Keywords: source location privacy, data mules, alpha-angle anonymity, wireless sensor networks, mules saving source protocol

1 Introduction

In recent years, WSNs have played an important role in a number of security applications, like remotely monitoring objects etc. In such applications, the location of the monitored object is tightly coupled with the sensor that detects it, called the data source. Therefore, preserving the location of data source is important for protecting the object from being traced. Such a preservation cannot be simply accomplished by encrypting the data packets as the location of the data source can be disclosed by analyzing the traffic flow in WSNs.

The problem of preserving source-location privacy can be explained using the "Panda-Hunter Game" [1], in which the sensors are deployed in the forest to monitor the movement of pandas. Each panda is mounted with an actuator which signals to the surrounding sensors in its communication range. When the sensor close to the panda receives the signal, it creates and sends data reports to the base station over the wireless network. A hunter who is monitoring the wireless communication between the sensors will be able to identify the direction of incoming traffic flow and trace back the data transmission path to locate the data source, thus catching the panda. In fact, any WSN used for such monitoring applications are vulnerable to such kind of traffic analysis based attacks.

There have been extensive techniques proposed to preserve source-location privacy against different attack models, *the local-eavesdropping model* and *the global-eavesdropping model*. Local-eavesdropping [1, 8, 7, 5] assumes the attacker's ability to monitor the wireless communication is limited to a very small region, up to very few hops. In global-eavesdropping model [6, 12, 11], the attacker is assumed to be capable of monitoring the traffic over the entire network. We believe both of them are unrealistic, because the former stringently restricts the attacker's ability, while the latter exaggerates it, considering resources and cost required for launching such an attack.

In this paper, we propose a more practical attack model, *the semi-global eavesdropping model*, in which the attacker is able to eavesdrop on wireless communications in a substantial area that is much smaller than the entire monitoring network. This attack model allows the attacker to gather substantially more information than a local eavesdropper. As shown in Section 3, this attack allows the attacker to overcome defenses that defeat a local eavesdropper. Meantime, without the ability of monitoring the entire network, system designers can consider alternatives to network flooding and other approaches against global eavesdropping model that suffer from high communication overhead.

Under the semi-global eavesdropping model, we explore a novel protocol for preserving source-location privacy by using data mules. Traditionally, data mules are used in WSNs for reducing energy consumption due to the data transmission between sensors and facilitating communication in disconnected networks. A data mule picks up data from the data source and then delivers them directly to the base station. We adapt the functionality of data mules so that they not only maintain their traditional functionality, but also facilitate the preservation of the location privacy of data sources. Our main contributions in this paper are summarized as follows: (1) we propose a new attack model, called semi-global eavesdropping; (2) we introduce a linear-regression based traffic analysis approach to enable the attacker to infer the direction of data source, and demonstrate its effectiveness by breaking an existing routing protocol of preserving source-location privacy; (3) we define the α-angle anonymity model for studying the source-location preservation; (4) we propose a novel protocol, called Mules-Saving-Source protocol, that uses data mules to achieve α-angle anonymity. The protocol is evaluated by an absorbing Markov Chain based model; (5) we conduct a comprehensive set of simulations to evaluate our protocol performance.

The roadmap of this paper is given as follows. We describe the system model and network scenario in Section 2. In Section 3 we introduce the attack model as well as our proposed linear-regression based approach to analyze traffic, followed by the α-angle anonymity model for preserving source location privacy. In Section 4, we present the Mules-Saving-Source protocol to protect the location of data source. In addition, we theoretically analyze the data delay introduced by our protocol in Section 5. Finally, we evaluate our protocol performance by analyzing the results from a comprehensive set of simulations in Section 6, followed by related works in Section 7 and conclude the paper in Section 8.

2 System Model

The terrain of our underlying network is a finite two-dimensional grid, which is further divided into cells of equal size. The network is composed of one base station, static sensors, and mobile agents, called data mules.

Static sensors - All static sensors are homogeneous with the same lifetime and capabilities of storage, processing as well as communication. They are deployed uniformly at random in the cells, and assumed to guarantee the connectivity of the network.

Data mules - Data mules are the mobile agents which can be artificially introduced in the network [10]. We assume they move independently and do not communicate with each other. Also, they are assumed to know their own locations when they are moving all the time. Their mobility pattern can be modeled as a random walk on the grid, whereby in each transition it moves with equal probability to one of the horizontally or vertically adjacent cells. After a data mule moves into a cell, it stays there for t_{pause} time period before its next transition. At the beginning of the pause interval, the data mule announces its arrival by broadcasting Hello Message. Only data source will respond and relay buffered data to the data mule. We assume the data mule does not communicate with sensors when moving. The data mule's communication range is larger than the of a sensor, thus a data source which cannot directly transmit data to the data mule will use multi-hop routing.

3 Preliminaries

In this section, we will first introduce our attack model and then propose a linear-regression based approach for analyzing data traffic. Furthermore, we will demonstrate the effectiveness of out attack model by compromising the phantom routing protocol [1]. Finally, we will define the α-angle anonymity model for studying the location privacy preservation of data source.

3.1 Attack Model

We assume the attacker is capable of launching only passive attacks, in which he can only monitor the traffic transmission but not decrypt or modify data packets. Suppose the attacker monitors the radio transmissions between sensors in a circular area of radius R_{att}, as shown in Fig. 1. The larger the monitoring area, the stronger the attacker. If the monitoring area is large enough to cover the whole network, it is global eavesdropping; on the other hand, if the area is

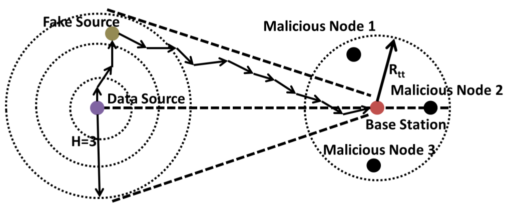

Fig. 1: Traffic flow in phantom routing

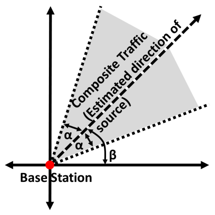

Fig. 2: α-angle anonymity

limited only to a few hops, it is local eavesdropping. However, we define *semi-global eavesdropping* as whose strength lies in between the two extreme attack models. In addition, we believe without any prior knowledge of source location, the attacker is inclined to launch the attack by collecting traffic data from around the base station. Intuitively, since the whole network traffic converges to the base station, it serves as the ideal point for starting the attack. Admittedly, the attacker can make an initial estimation of the direction of data source and move in that direction. Meantime, he can keep updating his estimation with more traffic observed as he moves, until he finds the data source. However, in this paper we aim to discourage the attacker even from making a good initial estimation before he starts moving.

3.2 Linear-regression Based Traffic Analysis

The attacker starts observing the incoming data transmission around the base station and analyzes it to estimate the direction of the data source. However, this estimation is not straightforward as the observed transmission paths are not linear or constant due to the multi-hop data routing as well as the randomness introduced in the secure routing protocols, such as the random H hops in phantom routing[1]. To address this issue, we apply the linear-regression [9] to find the best fit line representing the location of the sensors in the transmission path for a single data packet, as observed by the attacker. Since the packets are forwarded towards and delivered to the base station, we force each regression line to pass through the base station. The regression lines are used to estimate the direction of the incoming traffic which helps us in inferring the direction of data source. Ideally, in the absence of any spatial randomness introduced in the routing path, walking along the regression line would reveal the location of the source.

To estimate the direction of data source, we define a *traffic vector* with unit magnitude for each data packet observed by the attacker. The vector's direction is given by the direction of the regression line representing the transmission path of the data packet. By doing so, we have a traffic vector for each transmission path observed by the attacker. The direction of the data source can be inferred from the direction of the composite vector formed by summing up the traffic vectors defined for each transmission path.

Compromising Phantom Routing - We claim that using our proposed attack model and traffic analysis approach, the direction of the composite vector will reveal the direction of the data source in phantom routing. *Phantom Routing Protocol* proposed in [1] requires each generated data packet to be first randomly routed H hops from data source, and then forwarded to the base station along the shortest path. By using that protocol, a backtracking attacker will fail to find out the real data source due to random H-hop routing. We define fake sources in phantom routing as the last sensor at the H^{th} random hop in the first phase of the protocol. Assuming all sensors are deployed uniformly at random in the network, the location distribution of fake sources statistically forms concentric rings centered around the real data source wherein the sensors in the same ring have similar probability to become fake sources. This is due to two facts: (1) symmetric deployment of sensors around the real data source, and (2) in the first phase of the routing, the next-hop sensors are selected uniformly at random from the surrounding sensors within the transmission range.

Since the fake sources are distributed symmetrically around the data source, the composite traffic vector for all data transmission paths gives the direction of the data source. We carried out simulations to confirm our analysis. We configured phantom routing with $H = 8$ and analyzed the cost of launching the attack over 1000 trials. Firstly, we studied the cost of the attack as the attacker's monitoring area. As shown in Fig. 3, when the attacker's monitoring area is restricted to a few hops of 2 or 4 (i.e., local eavesdropping), the estimated error is very high as compared to the scenarios when the attacker observes transmissions over a larger number of hops. Therefore, the attacker will move further away from the source as he moves along the estimated direction, thus he is being biased. Hence, the protocol will provide defense against the local eavesdropping adversary. The larger the monitoring area, the more accurate the inference of the source direction. We further analyzed the cost as the amount of data packets required for the attacker to make a good estimation of the data source direction. As shown in Figure 4, a semi-global eavesdropper is able to infer the source direction without observing a large amount of data transmission, i.e. around 60 packets. Therefore, one can see that our attack model is effective in compromising the phantom routing.

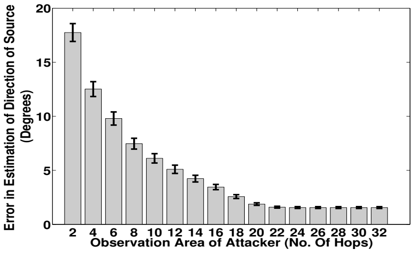

Fig. 3: Error in Estimated Direction of Source

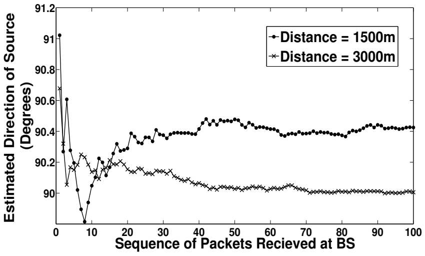

Fig. 4: Estimated Direction of Source with Varying Distance from BS

3.3 α-Angle Anonymity

In order to anonymize source location privacy under semi-global eavesdropping attack, we introduce α-angle anonymity model in Definition 1. This model ensures the preservation of source location privacy by enlarging the inference space from which the attacker estimates the real direction of the data source. The inference space is determined by the system variable α. The value of α can be open to the public, even including the attacker, however, this should not threaten the privacy of source location. According to the definition, we can see that the larger the value α, the larger the inference area. The shaded area in Fig. 2 represents the attacker's inference space. Given a larger inference space, the attacker cannot deterministically estimate the real direction or location of data source, thereby the source location privacy being preserved.

Definition 1. ***α-Angle Anonymity.*** *A protocol is α-angle anonymous if the real direction of data source is equally likely distributed in the angle range $[\beta - \alpha, \beta + \alpha]$, where β is the angle of the direction inferred by the attacker based on his observation.*

4 Mules-Saving-Source Protocol

To protect the source location privacy against a semi-global eavesdropper, we design a protocol, called *Mules-Saving-Source* protocol, achieving α-angle anonymity. Our protocol exploits the random mobility of data mules to establish a data transmission pattern which effectively preserve the location privacy of data source. Specifically, we modify the traditional function of data mules by having them hand data to regular sensors at only specific locations in the network, from where data will be further routed towards base station along the shortest paths. The specific sensors will be selected so as to bias the direction of composite traffic to be derived by the attacker based on data transmission he observes around base station.

In fact, solely allowing data mules to directly deliver data to base station can thoroughly preserve source location privacy against a semi-global eavesdropper. This is because the data transmission between data source and base station is completely hidden by the random movement of the data mules which ferry data. However, its disadvantage is the non-trivial delay caused by data mules, which may not be tolerable especially in large-scale wireless sensor networks. In this section, we first describe our protocol and then prove it to be α-angle anonymous. Note that we predefine a coordinate system with the base station as the origin, which is assumed to be known by data mules. Our protocol includes three phases: 1) picking a fake direction at source, 2) carrying and unloading data by data mules, and 3) routing data to the base station.

Phase 1. Picking a fake direction at source - When a target is detected by the sensors, they coordinate among themselves and let the one closest to the target become the data source. The coordination protocol has been well studied in literature [13] and its discussion is out of the scope of this paper. The data source periodically generates and sends data reports towards base station. Additionally, it generates a value of β as the fake direction of data source to be used for biasing the attacker's observation in the traffic flows coming towards base station. Specifically, the data source selects β from the range $[\theta - \alpha, \theta + \alpha]$ uniformly at random, where θ is the absolute angle between the direction of data source and the direction of x-axis in our coordinate system, and α is a value preset to configure the privacy preservation level. The β angle is known only by the data source initially.

Phase 2. Carrying and unloading data by data mules - When a data mule moves into a cell, only the data source in its communication range responds with the buffered packets. Along with the data, the data source also sends the value of β angle the data mule. After getting the data, the data mule roams around the network until reaching certain location, called *dropping point*. Dropping point is referred to as any point located on the *dropping line* drawn from base station at an angle β in the coordinate system. Upon arriving in a cell intersecting with the dropping line, the data mule unloads the data to the sensor closest to the dropping line present within the cell.

Phase 3. Routing data at sensors - After data packets are offloaded to a sensor by the data mule, they are routed towards base station along the shortest path. Ideally, the transmission path is along the dropping line. However, due to the nonlinear multi-hop routing, data transmission may have trivial deviation from the dropping line, which should not affect the privacy preservation. One can see the traffic flow will go towards base station roughly along the direction with a β angle, thereby successfully biasing the attacker's inference of data source direction.

Given the MSS protocol, let us further demonstrate its effectiveness for preserving source location privacy by Theorem 1.

Theorem 1. *Mules-Saving-Source protocol is α-angle anonymous for source location privacy.*

Proof. In MSS, since all data from a data source are forced to come towards base station along the fake direction at an angle of β, their composite traffic vector will be along the same direction. Although the attacker knows the rule of picking the fake direction - $\gamma - \alpha \leq \beta \leq \gamma + \alpha$, where γ is the absolute angle of the source direction, which is unknown to the attacker. He can only deduce the data source lies in a region given by $\beta - \alpha \leq \gamma \leq \beta + \alpha$. Therefore, MSS achieves α-angle anonymity in terms of Definition 1.

5 Protocol Analysis

In order to model the mobility pattern of data mules we use a discrete-time Markov chain model, similar to the model proposed in [10]. Each state in the Markov chain represents the condition when the data mule is present at a specific cell in the network. The probability (P_{ij}) of a data mule transiting from one state s_i to another state s_j for Markov chain with state space S is:

$$P_{ij} = \begin{cases} \frac{1}{q}, \text{ if } s_i \text{ and } s_j \text{ are adjacent} \\ 0, \text{ Otherwise} \end{cases} \tag{1}$$

where two states being adjacent means their corresponding cells are adjacent to each other either horizontally or vertically, and q is the number of adjacent states of s_i. Additionally, we assume the mobility pattern of data mules has

Table 1: Terminology Table

Notations	**Definitions**
N_{mules}	Number of data mules in the network
L_n	The length of the network ($L_n \times L_n$) edge
L_c	The length of each cell ($L_c \times L_c$) edge
v_{mule}	The velocity of each data mule moving
t_{move}	The transition time of data mule from one cell to another (L_c/v_{mule})
t_{pause}	The pausing time of data mule in each cell it stays
D_{src}	Random variable for the buffering time at data source
D_{mule}	Random variable for the carrying time at data mule
E_{absorb}	the expected number of transitions until reaching any absorbing state
$E_{D_{mule}}$	the expected delay at data mule

achieved stationary distribution. In Table 1 we describe the notations used in our analysis.

In the following subsections, we will compare our MSS protocol with direct delivery protocol (DD) with respect to data delay. Basically, the end to end delay for data delivery in data-mule based protocols consists of two parts: (1) D_{src} - the time period data source buffers a data packet until a data mule picks it up, and (2) D_{mule} - the time duration when a data mule carries a packet. In contrast with those two types of delay, transmission delay at static sensors in WSNs is trivial and thus ignorable. For the sake of simplifying our analysis explanation, we define one time unit (t_{unit}) as the total time spent in one transition, $t_{unit} = t_{move} + t_{pause}$.

5.1 Buffering Delay at Data Source

For the buffering delay at data source, both MSS and DD protocols follow the same analysis, because this delay will not be affected by where a data mule will drop data, either to base station in DD or to the dropping points in MSS. One analytic result is derived in [10], formalizing the distribution of buffering delay at data source (D_{src}), as shown in Equation 2, where one time unit is given by t_{unit}, and the buffering capacity at data source is assumed to be unlimited. From Equation 2, one can see the the probability of having small buffering delay at the data source increases with the number of data mules and decreases with the number of cells in the network $(\frac{L_n}{L_c})^2$.

$$P\{D_{src} \leq t\} \approx 1 - \exp\left(\frac{-t}{0.68\frac{L_n^2}{L_c^2 \times N_{mules}}\log(\frac{L_n}{L_c})}\right) \tag{2}$$

5.2 Carrying Delay At Data Mules

The carrying delay at data mule in MSS protocol differs from that in DD protocol. We will first introduce a result concluded in [10], which is relevant to the carrying delay in DD. Then, we will present our model to evaluate the carrying delay at data mule for both MSS and DD protocols. Equation 3 derived in [10] formulates the delay distribution of carrying data at data mule with base station assumed to be located at the center of the network. Such a delay increases with the number of cells in the network, $(\frac{L_n}{L_c})^2$. Additionally, it is associated with the moving speed of data mule, which is captured by the time unit definition.

$$P\{D_{mule} \leq t\} \approx 1 - \exp\left(\frac{-t}{0.68(\frac{L_n}{L_c})^2 \log(\frac{L_n}{L_c})}\right) \tag{3}$$

Note the analysis given in Equation 3 is the distribution of the delay of carrying data at data mules, given a random data source. However, we can also analyze this delay for a specific data source by using an absorbing Markov chain so that we can compare the performance of DD and MSS protocols. In the following, we will first introduce some basic concepts regarding absorbing Markov chain in Definition 2, and then present our model.

Definition 2. *Absorbing Markov Chain. A state s_i of a Markov chain is called an absorbing state if the probability of staying in the current state s_i after transitioning into it is one. The rest of the states which are not absorbing are called transient states. An absorbing Markov chain is one if it has at least one absorbing state and all absorbing states are reachable from each transient states.*

According to Definition 2, DD can be modeled as an absorbing Markov chain with one absorbing state, in which the absorbing state is the state when a data mule reaches the cell having base station in. We call it an absorbing state due to the fact that when the data mule reaches that state, the data it is carrying are delivered to the base station and do not transit to adjacent cells with it any more. Therefore, data transition in DD can be modeled as an absorbing Markov chain with a single absorbing state. On the other hand, our MSS protocol can be modeled as an absorbing Markov chain with multiple absorbing states. All cells which intersect with the dropping line form the absorbing states because the data carried by the data mule are unloaded to static sensors in those cells and no longer transit to adjacent cells with the data mule. Therefore, data transition in MSS can be modeled as an absorbing Markov chain with multiple absorbing states. We can see that in both of our models, any absorbing state can be reached from any transient state with at least one transition.

The Expected Carrying Delay at Data Mules Based on our absorbing Markov chain models, the expected delay of carrying data at data mules is associated with the expected number of transitions until it reaches the absorbing state by $E_{D_{mule}} = E_{absorb} \times t_{unit}$. Therefore, we will focus on estimating E_{absorb} in the following analysis. Given an absorbing Markov chain, *Canonical Form* of transition matrix can be derived as shown in Equation 4, where there are t transient states and r absorbing states.

$$\mathbf{P} = \begin{pmatrix} \mathbf{Q} & \mathbf{R} \\ \mathbf{0} & \mathbf{I} \end{pmatrix} \tag{4}$$

where $\mathbf{I}$ is an r-by-r identity matrix, $\mathbf{0}$ is an r-by-t zero matrix, $\mathbf{R}$ is a nonzero t-by-r matrix representing the transition probabilities from transient states to absorbing states, and $\mathbf{Q}$ is a t-by-t matrix of the transition probabilities between transient states. Once the absorbing state is reached the data remain in the absorbing state with probability 1. Based on the canonical form, the *Fundamental Matrix* N for an absorbing Markov chain is defined as $\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1}$.

From [2], we know that the expected number of transitions before the chain is absorbed is given as $\mathbf{w} = \mathbf{Nc}$, where each i^{th} entry w_i in $\mathbf{w}$ is the expected number of transitions before the chain is absorbed, given the fact that the chain starts from the i^{th} transient state in $\mathbf{P}$, and $\mathbf{c}$ is a column vectors with all entries of one.

By using the aforementioned formulas well-developed in the field of Markov Chains, we can derive the expected number of transitions until the chain is absorbed starting from any state to compare DD and MSS protocols. One can see that since the MSS protocol has more absorbing states than the DD protocol, and furthermore, the absorbing state in DD has been included in the absorbing states in MSS. Therefore, the expected number of transitions before the chain is absorbed in MSS is less than that in DD. As a matter of fact, in MSS the number of absorbing states as well as the locations of their corresponding cells in the network determines the expected number of transitions. The locations of these cells are actually specified by β angle.

The Lower Bound Of Carrying Delay At Data Mules Required In this section, we will analyze the minimum delay of carrying data required to ensure data delivery. The minimum delay depends on the minimum number of transitions needed to enter the absorption state, called the lower bound of transition number (LB), and can be determined by $LB \times t_{unit}$. For the sake of simplifying our analysis description, we index each cell in the network by the indices of its row and column, respectively. For example, a cell in row i and column j is indexed as (i, j), and represented as $cell_{(i,j)}$. Suppose the data source is located in $cell_{(i_{src}, j_{src})}$, and base station is in $cell_{(i_{bs}, j_{bs})}$. The cells corresponding to the multiple absorbing states are represented by $cell_{(i_{ab_k}, j_{ab_k})}$ $(k = 1, ...n_{ab})$, where n_{ab} is the total number of absorbing states. In DD protocol $n_{ab} = 1$. We propose Theorem 2 which addresses the relation of the lower bounds of transitions for both of the protocols. Before proving Theorem 2, let us introduce Lemma 1 first. Then from Theorem 2 we could see the minimum delay required to deliver data in MSS protocol is smaller than that in DD protocol.

Lemma 1. *Given $cell_{(i_{src}, j_{src})}$ and $cell_{(i_{ab_k}, j_{ab_k})}$ which is associated with any absorbing state s_{ab_k}, $LB_{s_{ab_k}} = |i_{src} - i_{ab_k}| + |j_{src} - j_{ab_k}|$.*

Proof. Consider the mobility pattern of data mules, the minimum number of transitions required to reach the absorbing state s_{ab_k} from s_{src} is determined by the Manhattan Distance of the two cells, $cell_{(i_{src},j_{src})}$ and $cell_{(i_{ab_k},j_{ab_k})}$, which is exactly $|i_{src} - i_{ab_k}| + |j_{src} - j_{ab_k}|$.

Theorem 2. *The lower bound of transition number in MSS protocol, LB_{mss}, is not larger than that in DD protocol, LB_{dd}.*

Proof. According to the definition of the lower bound transition number, we know that $LB_{mss} = \min(LB_{s_{ab_k}})$, where s_{ab_k} is any absorbing state. Since base station is one of those absorbing states, $LB_{mss} \leq LB_{s_{base\_station}}$. From $LB_{s_{base\_station}} = LB_{dd}$, we get $LB_{mss} \leq LB_{dd}$. Therefore, the theorem holds.

6 Simulation and Results

Table 2: Simulation Scenario

Total no of sensors	10000
Total deployment area	10^8 sq. m
Communication range of sensors	200 m
Data generation rate	1 per s
Speed of data mules	25m/s
Pause time of data mules	1 second
Dimensions of logical cells of data mules	500mx500m
Communication range of data mules	250 m
Total simulation time	15000 seconds

A comprehensive set of simulations were conducted using a customized C++ based simulator to evaluate the performance of MSS protocol. Specifically, we investigated the association of data delay with the number of data mules, as well as with the α value. The latter indicates the trade-off between privacy preservation and data delay. Furthermore, we validated the accuracy of our proposed analytical models for both direct delivery (DD) protocol and our MSS protocol through our simulations. The simulation configuration is detailed in Table 2. We

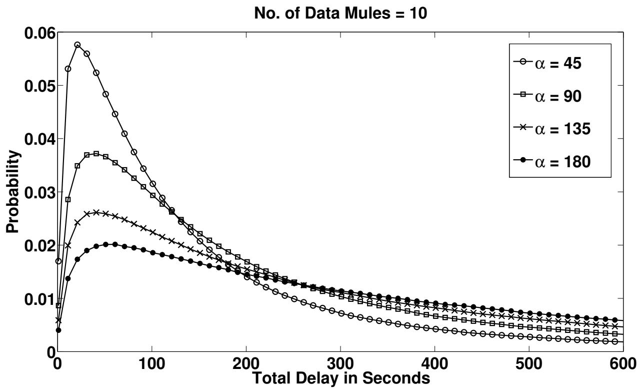

Fig. 5: PDF of total delay with 10 data mules

assume the base station is located at the center of the network at $(5000, 5000)$, and the data source is at $(5000, 8000)$. Sensors are deployed uniformly at random in the network. At the beginning of the network, tree routing topology is constructed by the broadcasting of a message over the entire network, which is initiated from base station. In our simulations, data mules are initially deployed uniformly at random over the network, and then move based on the mobility pattern introduced in Section 5. The data mules move in logical cells of size $500m \times 500m$.

We performed simulations with varying number of data mules and different α values to study their impact on the data delay caused by data mules carrying data. We performed 1000 trials for each simulation configuration, and all plots shown in this section are the averaged results over these 1000 trials of experiments. Based on our results, we plotted the PDF of the total data delay - the time from when the data is generated to the time when it is delivered to base station, with varying number of data mules as shown in Fig. 5 and Fig. 6.

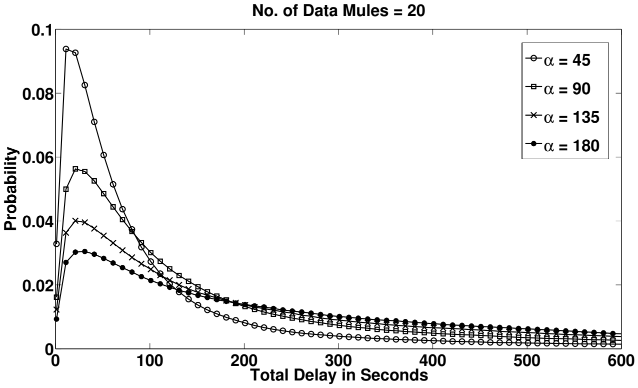

Fig. 6: PDF of total delay with 20 data mules

6.1 Delay And Data Mules

As we mentioned in Section 5, the total data delay is composed of two parts, the buffering delay at data source and the delay of carrying data at data mules. For the buffering delay at data sources, we can see from Fig. 7, given a value of α, the more data mules introduced to the network, the less the buffering delay at data source. On the other hand, from Fig. 8, we observe that the increasing of the number of data mules does not significantly impact on the delay of carrying data at data mules. The reason is that after a data mule picks up the data from the data source, the carrying delay solely depends on the mobility pattern of the data mule, rather than the number of data mules. Therefore, as displayed in Fig. 9, the total expected delay increases as well with the increasing of the number of data mules.

6.2 Delay And Privacy Preservation

The value of α represents the degree of the privacy to be preserved. Given a large value of α, the attacker has to infer the real direction of data source from

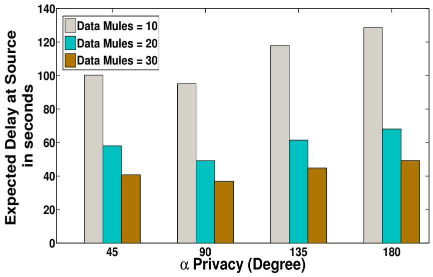

Fig. 7: Expected delay at source

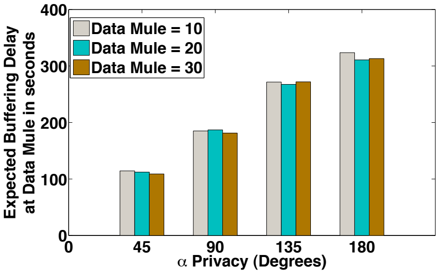

Fig. 8: Expected delay at data mules

a wider inference space, which reduces the possibility of succeeding. However, increasing the value of α has adverse impact on data delay. From Fig. 7 we observe that given a specific number of data mules, varying α does not have a significant influence on the buffering delay at data source. This is because the buffering delay at source is impacted by the network configuration, such as the number of data mules rather than the α value as shown in Equation 2. On the other hand, given a specific number of data mules, the larger the value of α, the longer the delay of carrying data at data mules, as shown in Fig. 8. This is because setting a larger value to α leads to the possibility of selecting a β value more deviating from the data source, so that data mules choose dropping points further away from the real source to offload data, thus causing a larger delay of carrying data. Due to the impact of α value on both of the two parts of delay, the total delay therefore increases with the value of α, as displayed in Fig. 9, Fig. 5 and Fig. 6.

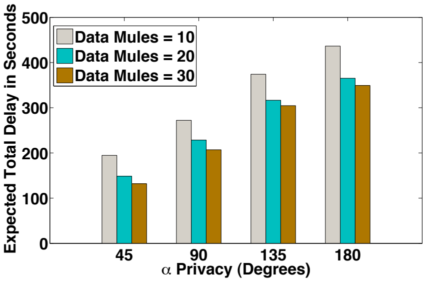

Fig. 9: Expected total delay with varying number of data mules

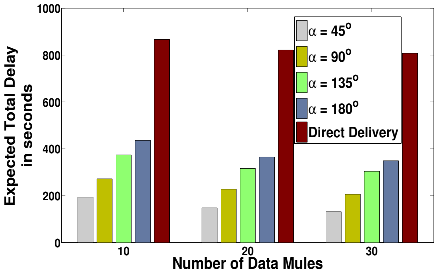

Fig. 10: Expected total delay with varying α

6.3 MSS And Direct Delivery

According to the analysis given in Section 5, we already know that DD protocol leads to a larger data delay compared with our MSS protocol. We further compare them through the simulations. As shown in Fig. 10, given a fixed number of data mules, the total delay of DD is much larger than that of MSS. Thus, although DD protocol guarantees the complete preservation of the location privacy of data source, it causes high delay, as compared to MSS protocol.

7 Related Work

We discuss the techniques for preserving the location privacy of data source by categorizing them based on different attack models they counteract, namely *local-eavesdropping model* and *global-eavesdropping model*. For a more comprehensive taxonomy of techniques of preserving privacy in WSNs, we refer readers to the state-of-the-art survey [3].

For local-eavesdropping based attack, flooding based approach was first introduced in [8], where each sensor broadcasts data that it receives to all its neighbors. However, this technique suffers from high communication overhead for sensors. In [7], cyclic entrapment technique is introduced to create looping routes between data source and base station aiming at trapping the attacker in loops when he backtracks along the data transmission path.

In [5], each data packet is first relayed to a randomly selected intermediate sensor in the network and then is forwarded towards base station along the shortest path. For global-eavesdropping based attack, the authors in [6] create $k-1$ fake sources in the network to anonymize the real data source. Additionally, *proxy-based* technique is proposed in [12] wherein a set of proxies are distributed in the network partitioned into cells. Each cell sends traffic including both real and fake packets to its nearest proxy by following an exponential distribution. The proxies filter out some dummy packets they collected, and then send the remaining data to base station. A similar idea is brought up in [11] in which rather than relying on proxies, cluster-heads first aggregate data and then report them to base station.

The authors in [4] propose a mixing ring-based technique, in which a closed circular routing path is formed around base station. Data source first routes the data packet to a random intermediate sensor in this ring, which provides local source-location privacy preservation. Then the data is routed along the ring and will be forwarded towards base station by any ring-node with a given probability. However, it is difficult to predetermine the size of the ring without knowing the attacker's monitoring ability.

Different from the above line of research, our proposed protocol uses data mules to deliver data, therefore reducing energy consumption on the communication among sensors. Furthermore, since there exists no physical data transmission path between the data source and the sensors at which data mules unload data, the attacker cannot backtrack the transmission to locate the data source. Additionally, our protocol allows the system designer to configure the privacy level as desired based on the tolerable delay in data delivery, the network area and the number/speed of data mules.

8 Conclusion

In this paper, we were focused on the location-privacy preservation of data source in WSNs. Different from the literature, we defined a more practical attack model, namely semi-global eavesdropping, whose strength lies in between local-eavesdropping and global-eavesdropping. Besides, we proposed a linear-regression based approach to enable the attacker to analyze data traffic in the network. Based on our traffic analysis approach, we demonstrated the vulnerability of phantom routing under the semi-global eavesdropping attack. Furthermore, we defined the α-angle anonymity model for measuring the privacy preservation of source location in WSNs. Under the semi-global eavesdropping attack model, we designed a protocol for preserving the location of data source, called Mules-Saving-Source protocol, which is proved to be α-angle anonymous. Additionally, we theoretically analyze the data delay for both our MSS protocol and the direct delivery protocol by an absorbing Markov chain model. Finally, by conducting a comprehensive set of simulations we evaluate our protocol performance and drew a couple of conclusions: (1) the increasing of the number of date mules leads to the decreasing of data delay; (2) a higher degree of privacy preserving, represented by a larger value of α, leads to a longer data delay.

9 Acknowledgment

This work is partially supported by US National Science Foundation (NSF) grant IIS-0326505, CNS-0721951 and CNS-0916221. Any opinions, findings and conclusions or recommendations expressed in this paper are those of the authors and do not necessarily reflect those of the National Science Foundation.

References

1. Kamat, P., Zhang, Y., Trappe, W., Ozturk, C.: Enhancing source-location privacy in sensor network routing. In: Distributed Computing Systems, 2005. ICDCS 2005. Proceedings. 25th IEEE International Conference on. pp. 599 –608 (June 2005)
2. Kemeny, J.G., Snell, J.L.: Finite Markov Chains. Springer-Verlag (1976)
3. Li, N., Zhang, N., Das, S.K., Thuraisingham, B.: Privacy preservation in wireless sensor networks: A state-of-the-art survey. Ad Hoc Networks 7(8), (2009)
4. Li, Y., Ren, J.: Mixing ring-based source-location privacy in wireless sensor networks. In: Proceedings of ICCCN. pp. 1–6 (August 2009)
5. Li, Y., Ren, J.: Source-location privacy through dynamic routing in wireless sensor networks. In: Proceedings of INFOCOM (2010)
6. Mehta, K., Liu, D., Wright, M.: Location privacy in sensor networks against a global eavesdropper. In: Proceedings of ICNP. pp. 314–323 (October 2007)
7. Ouyang, Y., Le, X., Chen, G., Ford, J., Makedon, F.: Entrapping adversaries for source protection in sensor networks. In: World of Wireless, Mobile and Multimedia Networks, 2006. WoWMoM 2006. International Symposium on a (2006)
8. Ozturk, C., Zhang, Y., Trappe, W.: Source-location privacy in energy-constrained sensor network routing. In: Proceedings. 2nd ACM workshop on Security of ad hoc and sensor networks, SASN (2004)
9. S., R.M.: Introductory Statistics. Academic Press Title (2010)
10. Shah, C.R., Roy, S., Jain, S., Brunette, W.: Data mules: modeling and analysis of a three-tier architecture for sparse sensor networks. Ad Hoc Networks 1(2-3), 215–233 (2003)
11. Yang, W., Zhu, W.: Source location privacy in wireless sensor networks with data aggregation. In: Proceedings of UIC. pp. 1–6 (August 2010)
12. Yang, Y., Shao, M., Zhu, S., Urgaonkar, B., Cao, G.: Towards event source unobservability with minimum network traffic in sensor networks. In: Proceedings of WiSec (2008)
13. Zhao, F., Shin, J., Reich, J.: Information-driven dynamic sensor collaboration for tracking applications. IEEE Signal Processing Magazine 19, 61–72 (March 2002)