Quantifying Source Location Privacy Routing Performance via Divergence and Information Loss

Matthew Bradbury* and Arshad Jhumka†
*School of Computing and Communications, Lancaster University, Lancaster, UK
†Department of Computer Science, University of Warwick, Coventry, UK
Email: M.S.Bradbury@lancaster.ac.uk, H.A.Jhumka@warwick.ac.uk

***Abstract*—Source location Privacy (SLP) is an important property for security critical applications deployed over a wireless sensor network. This property specifies that the location of the source of messages needs to be kept secret from an eavesdropping adversary that is able to move around the network. Most previous work on SLP has focused on developing protocols to enhance the SLP imparted to the network under various attacker models and other conditions. Other works have focused on analysing the level of SLP being imparted by a specific protocol. In this paper, we introduce the notion of a routing matrix which captures when messages are *first* received. We then introduce a novel approach where an optimal SLP routing matrix is derived. In this approach, the attacker's movement is modelled as a Markov chain where measures of conditional entropy and divergence are used to compare routing matrices and quantify if they provide high levels of SLP. We propose the notion of a *properly competing paths* that causes an attacker to *divert* when moving towards the source. This concept provides the basis for developing a *perturbation model*, similar to those used in privacy-preserving data mining. We formally prove that properly competing paths are both necessary and sufficient in ensuring the existence of an SLP-aware routing matrix and show their usage in developing an SLP-aware routing matrix. Further, we show how different SLP-aware routing matrices can be obtained through different instantiations of the framework. Those instantiations are obtained based on a notion of information loss achieved through the use of the perturbation model proposed.**

***Index Terms*—Source Location Privacy; Wireless Sensor Networks; Entropy; Divergence; Perturbation.**

I. Introduction

Wireless sensor networks (WSNs) present a difficult challenge in creating secure and private applications due to their potential to expose important information owing to the broadcast nature of wireless communications. Even if encryption is used to protect the *content* of a message the *context* of the broadcast is still exposed for a malicious eavesdropper to exploit. One such problem that arises from context information leakage is where an attacker can monitor the pattern of broadcasts to gain knowledge about the location of the source of messages.

The Source Location Privacy (SLP) problem was initially introduced in terms of the panda-hunter game [3] where a WSN is deployed in a panda's habitat to monitor them. Using a directional antenna, an attacker in the network can identify the direction of the proximate source of a message and use this information to trace messages hop-by-hop through the habitat to find the ultimate source of the messages and thus the panda (or another valuable asset). SLP protection schemes aim to protect against this attack via various techniques, such as increasing the time an attacker would take to capture the source by changing the routing protocol. This assumes an approximate time the source (i.e., panda) will stay stationary.

There has been much work on providing SLP [4, 5] with many techniques using large-scale simulations to evaluate their performance. Several works [6–12] have developed models to analyse the privacy provided by their technique or protocol. Many of which tend to be for SLP techniques that provide privacy against an attacker with global visibility. There are two issues here: the first is that the modelling performed is for a single specific SLP routing protocol meaning that its results are not useful in analysing a broad range of other routing protocols. The second is that more general analyses focus on SLP techniques that defend against an attacker with global visibility of the network. This means that there is a lack of analysis of *arbitrary routing protocols against attackers with local visibility* present in the network. This adversary is equivalent to the "Patient Adversary" [13].

While adversaries with global visibility over the network are significantly more powerful compared to adversaries with local visibility, the resources required are also significantly more expensive. A global adversary would need to deploy their own network or use expensive equipment to localise transmissions from far away to monitor the network. On the other hand, a local adversary requires cheap equipment such as a directional antenna, software defined radio, and a laptop. This makes the barriers to a local adversary performing an attack much lower.

To resolve the lack of investigation in analysing routing protocols against attackers with local visibility, in this paper we introduce the notion of a *routing matrix* which captures when a message is *first* received, and then introduce a novel approach to SLP quantification using the information theoretic measures of conditional entropy and divergence to compare the performance of arbitrary routing matrices. Instead of an attacker with global visibility, this work focuses on a *mobile eavesdropping* attacker that is present in the network due to the lower cost to perform this attack. The analysis quantifies how much information is lost to this class of attacker. The attacker's position is modelled as a stochastic process and its movement is modelled by a Markov chain which is derived from a Markov chain representing the routing matrix.

A suitable definition of information loss is used with conditional entropy and divergence of random variables to determine

This paper is an extension of [1] and parts were included in [2].

how to produce an SLP-aware routing matrix. Specifically, a perturbation model is created such that a protectionless routing matrix, which is not SLP-aware, is transformed into an SLP-aware routing matrix whereby (i) a source can still do convergecast communication, and (ii) the attacker cannot reach the source within a specified time limit (the safety period). Manual effort is required to design a routing protocol that reflects the behaviour of the derived SLP-aware routing matrix. In essence, our approach takes a clear time series (non SLP-aware routing matrix) and applies the perturbation technique to the clear time series, generating a noisy time series (SLP-aware routing matrix). In this work a time series is a sequence of network edges representing message transmissions or attacker movements ordered by the time at which they occur. A clear time series is where attacker movements lead to the source being captured and a noisy time series is where noise has been applied such that the source is not captured within a time limit.

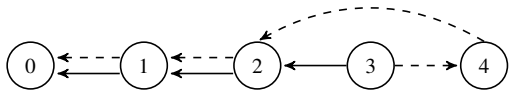


Figure 1: Visualisation of a clear attacker movement time series (solid line) and noisy movement time series (dashed line).

The perturbation occurs in such a way that the attacker learns little information about identifying the source from the noisy time series. Such an approach is beneficial as it does not make any assumptions (i) about the nature of the attacks, or (ii) any particular protocol implementation. An example time series perturbation for a 4-node network could be to apply noise to the time series $\langle(3,2)\cdot(2,1)\cdot(1,0)\rangle$ such that the path is longer and makes use of a redundant node $\langle(3,4)\cdot(4,2)\cdot(2,1)\cdot(1,0)\rangle$, which is visualised in Figure 1. The aim is analogous to differential privacy [14], where the application of noise reduces the utility of observations to the adversary. Unlike in privacy-preserving data mining where there is trade-off between information loss and privacy loss [15], no such trade-off is required here. This means that information loss can be maximised, if possible, to minimise privacy loss.

Furthermore, a novel concept called *proper competing paths* is proposed to capture the problem of whether the attacker can be *stalled* when moving towards the source. Proper competing paths are central to our perturbation model, as (i) wherever proper competing paths exist there is an increased entropy at that point, and (ii) noisy time series made up of proper competing paths are more likely to have high divergence with the corresponding clear data time series. The analysis approach developed will be applied to *DynamicSPR* [16] that provides SLP against a local attacker using fake sources. The following contributions are made in this paper:

1) The design of an SLP-aware routing matrix is formalised as a transformation problem.
2) Using conditional entropy and divergence, the requirements necessary to minimise the amount of information leaked by a noisy time series is derived.
3) The concept of proper competing paths is developed, which underpin the perturbation model proposed.
4) Two heuristics are proposed to (i) compute the set of proper perturbation paths, and to (ii) transform a protectionless routing matrix to be SLP-aware.

The remainder of this paper proceeds as follows: Section II presents related work on SLP and then Section III contains a description of the network, privacy and attacker models. In Section IV the problem statement is outlined and Section V details the analysis used to guide SLP protocol development. The perturbation model is described in Section VI and example case studies are presented in Section VII. In Section VIII considerations with using the model are discussed and Section IX concludes the paper.

II. Related Work

In the seminal work *phantom routing* was proposed [3] in which a directed random walk is performed away from or towards a landmark node until a message reaches a *phantom node* before that message is flooded in the network. This provides privacy by delaying the attacker on its way to the source. An energy-optimised version used a single path route from the phantom node to the sink [13]. This technique aims to protect against an adversary with a limited (i.e., local) view of the network. There has since been much work improving the directed random walk phase of phantom routing such as angle-based techniques [17, 18] which calculate angles between certain nodes to influence the direction of the walk. Other approaches route messages in a ring around the source before being forwarded to the sink [19, 20].

Another approach to providing SLP has been to use *fake sources*, which are nodes in the network which send messages encrypted and padded to be indistinguishable to normal messages sent from the source. There are several approaches [21, 22] with current techniques focusing on dynamically determining good parameters online [16]. A criticism of fake sources is that they tend to use more energy compared to phantom routing, although this is due to a lack of investigation into applying duty cycling to SLP techniques [23]. Other techniques consist of a hybrid between generating fake messages and messages taking alternate paths through the network. One example is tree-based diversionary routing [24] which imposes a tree structure on the network and then routes fake messages through the tree. The idea of fogs or clouds [25, 26] is similar, where a normal message is routed through a group of nodes called a fog and then onwards to other fogs. Fake messages are used to provide additional privacy. In general, limited work in the literature has consider multiple assets due to the challenges this problem poses [27].

The majority of SLP techniques demonstrated their performance by simulation, however, other approaches make use of information theoretic, statistical models or analysis to either assist in designing a SLP protocol or evaluating how well it performs. For example, a protection scheme called Periodic [11] provides perfect protection against an attacker with a global view of the network. Periodic achieves this against an adversary with global visibility by having every node sending a message after a fixed period. This was demonstrated by measuring the privacy loss of traces of source detections.

Following the global visibility theme, alternate approaches have provided *statistically strong* SLP [12] in contrast to the perfect SLP previously [11]. This model and solution aimed to make the distribution of message broadcasts from nodes indistinguishable from a statistical distribution (using the Anderson-Darling Test).

Other global protection schemes [9, 10, 28] have performed an analysis of their algorithms to justify their effectiveness. Global protection schemes tend to be easier to formalise and reason about compared to providing SLP against a local attacker and there have been several analysis approaches. Such as by quantifying the amount of source-location information that an individual message can leak to attackers [6]. In this work, the effect of multiple routing paths originating from the source node was evaluated, showing that a higher number of long paths increases the SLP provided.

Nezhad et al. [7] performed an analysis to determine the safety period for their technique (the higher the safety period the greater the SLP provided). Their analysis focused on a tree-based scheme and how an attacker would navigate it. Their analysis requires a bounded time for message forwarding and that the number of messages repeated follows the Poisson probability distribution.

A different approach was taken by Armenia et al. [8] with a information theoretic analysis, where the location of the asset was modelled as a random variable over the set of network nodes. Using a matrix of message forwarding probabilities the quantity of location information leaked was calculated. This solution is general as it does not rely on analysing individual paths, but instead analyses the overall protocol. However, the formalisation of the routing matrix is unlikely to be applicable to complicated local routing schemes.

Analyses of protocols against adversaries with global visibility succeeded in assisting with the techniques they accompanied, but, there are deficiencies in applying these strategies to adversaries with local visibility. For example, a network may contain many separate routing paths, but if an attacker never encounters them then they will not increase the location privacy as Armenia et al. [8] says they should. Also, many analyses focus on paths that solely originate from the source, none have taken the perspective of fake sources or multiple real sources and how they affect the information leakage. Another issue is that the aim of these analyses has been to evaluate a single protocol, none that we are aware of is designed to be generic enough to evaluate a wide range of protocols.

Name	Symbol	Description
Source	s	The node which generates messages about a detected asset.
Sink	q	Destination for messages generated by the source.
Adversary	$\mathbb{A}$	The adversary.
Capture Time	$\mathcal{TT}$	Expected time taken for $\mathbb{A}$ to capture the source.
Capture Likelihood	δ	Likelihood of $\mathbb{A}$ capturing s.
Safety Period	P_{safety}	Minimum time $\mathbb{A}$ needs to not capture the source in to be considered secure.
Routing Protocol	Q	Sets of paths messages can take from sink to source.
Routing Matrix	$\mathcal{R}$	Matrix of nodes receiving a message *first* from a 1-hop neighbour.
Attacker Routing Matrix	$\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}$	Matrix of attacker movement, derived from $\mathcal{R}_{\mathcal{X}}$.
Transition	(n_1, n_2)	Node n_2 receives a message from n_1 first.
Transitions	Γ	The set of all transitions.
Adversary Location RV.	$\mathbb{A}_{\mathcal{X}\lambda}$	A random variable of the attacker's location at time λ for either routing matrix.
Protectionless Transition RV.	$\mathcal{P}_{\lambda}$	A random variable of the transition taken at λ under $\mathcal{R}_{\mathcal{P}}$.
SLP Transition RV.	$\mathcal{S}_{\mu}$	A random variable of the transition taken at μ under $\mathcal{R}_{\mathcal{S}}$.

Protectionless and SLP-aware variants exist for Routing Protocols ($Q_{\mathcal{P}}$, $Q_{\mathcal{S}}$), Routing Matrices ($\mathcal{R}_{\mathcal{P}}$, $\mathcal{R}_{\mathcal{S}}$). To indicate that either a protectionless or SLP-aware variant could be used the $\mathcal{X}$ symbol is used (e.g, $\mathcal{R}_{\mathcal{X}}$).

Table I: Summary of Common Symbols

III. Preliminaries and Models

In this section, we present our system model, attacker model and the formal machinery used in this paper. A summary of the symbols used is shown in Table I.

A. System Model

We model a WSN as an undirected graph $G = (V, E)$, where the set of vertices V represents the set of wireless sensor nodes (or nodes) and the set of links between nodes is represented by the set E of edges. Each node u can directly communicate with another node v when $(u, v) \in E$, where we say that u and v are called neighbours and that a link exists between them. As the graph is undirected, it means that the links are bidirectional. We assume every link to be *reliable*, i.e., if a node u sends a message m to a neighbour node v, i.e., $(u, v) \in E$, then v will eventually receive m. Given a link (or transition) $n = (u, v)$, we denote by n_1 the start node of n, i.e., $n_1 = u$ and by n_2 the end node of n, i.e., $n_2 = v$. We also assume network links do not change over time, this means links between nodes are not added or removed.

A *path* $\langle n^1 \cdot n^2 \cdots n^j \rangle$ is a sequence of transitions from the initial node n^1_1 to the final node n^j_2, where each transition is an ordered pair of nodes $n = (n_1, n_2) \in E$. For a path $p = \langle n^1 \cdot n^2 \cdots n^j \rangle$ to be valid, it must be the case that the end node of the previous transition should be the same as the start node of the next transition, i.e., $n^i_2 = n^{i+1}_1, \forall i, 1 \leq i < |p|$. A path p is acyclic if $\forall n^i, n^j \in p : (n^i_1 \neq n^j_2)$. We denote the set of all acyclic paths from node u to node v by $\textsc{Paths}(u, v)$.

B. Timing Information

To ease explanation and understanding, we assume a fictitious global clock to which every node is synchronised and assume time to be split into equal sized slots. In each slot, nodes in the network will transmit in such a way that no message collision will result. We call the size of a slot a time unit. A sequence of contiguous slots of a specified size is called a period.

Algorithm 1 Patient Adversary [13] executed by the local adversary present in the network

```
pos ← q                                              ▷ Start at the sink
receive Message⟨msg⟩ from n →
    if IsFirstNewMessage(msg) then                   ▷ First time receiving msg?
        if pos ≠ s then                              ▷ Have not already found source
            pos ← n                                  ▷ Move to proximate sender
```

Each node senses the environment, and when a node detects an event of interest (e.g., the presence of an asset), the node acts as a *source*, which we denote s, and sends messages containing information on the event. This modelling focuses on the case where there is only a *single* source node in the network (i.e., there is only a single asset to protect). Once an event is detected, the source will frequently transmit messages, we do not place any requirements on how often a message will be sent through the network.

We assume that a first message is sent by the source in the first slot, i.e., time 1. When time is in the k^{th} slot, we say that time $t = k$. When a node transmits a message at time t, its neighbours receiving the message will (possibly) forward it in the following time slot, i.e., at time $t + 1$. All messages are ultimately routed to a special node in the network called a *sink*, which we denote by q.

C. Message Routing

In a multi-hop WSN represented as $G = (V, E)$, a message generated by the source s will need to be routed to the sink q via a series of single-hop forwarding. This message forwarding process is repeated until the message is received at the single sink in the network. The message is routed towards the sink q using a multi-hop routing protocol Q. A protectionless routing protocol $Q_{\mathcal{P}}$ is a set of shortest paths from all possible source nodes $s \in V$ to the sink $q \in V$, $s \neq q$, that does not provide privacy. This is the union of the set of all shortest paths from s to q denoted by $Q^s_{\mathcal{P}}$.

$$Q_{\mathcal{P}} = \bigcup_{s \in V} Q^s_{\mathcal{P}} \tag{1}$$

$$Q^s_{\mathcal{P}} = \{\, p \mid \forall p, p' \in \textsc{Paths}(s, q), |p| \leq |p'| \wedge p \neq p' \,\} \tag{2}$$

If there are multiple shortest paths from a given source s, i.e., $|Q^s_{\mathcal{P}}| \geq 2$, then we assume that messages travel along all the paths (not necessarily simultaneously).

Under a routing protocol a receiver node r will typically receive a message first along a shortest path from the source to r to minimise energy cost. Hence, this is why we will focus on the first new message that nodes receive. Nodes may receive a message more than once, when it is received for the first time and again from the next neighbour node that is forwarding the message onwards. In this model, a node *cannot* receive a message first along a route in $Q^s_{\mathcal{P}}$ which is not the shortest path from the source s to the sink q, i.e, q cannot receive a message from s along a path p first if $p \notin Q^s_{\mathcal{P}}$.

D. Attacker Model: Routing-Based Eavesdropper

When a node s is sending a message m to a neighbour node r (on route to the sink) at time t, if an attacker is located at r, the attacker will hear the message m coming from s and can move to s from r. The attacker moves instantaneously, so at time $t+1$ the attacker will already be co-located with s. Specifically, wherever the attacker is located, upon eavesdropping the *first* new message at that location the attacker moves to the neighbour who relayed the message. The attacker moves in a direction opposite to the flow of message from source to sink. Thus, when the attacker hears a new message, it makes a step towards the source. This process can be repeated a number of times until the attacker reaches the source node, whereby it captures the asset. This is the "Patient Adversary" as defined by Kamat et al. [13, Algorithm 1] and described in pseudo-code in Algorithm 1. In this paper we assume the attacker starts at the sink q, as this is the one node in the network guaranteed to receive messages from the source.

The distributed eavesdropper attacker is modelled using a Markov chain. This means the attacker is memoryless and does not keep track of history information and it may revisit a node that it has previously visited. Thus, the path an attacker takes to capture an asset may contain loops.

Definition 1 (Capture Time): Given a network $G = (V, E)$, a source s, a routing protocol Q^s that sends messages from s, and an attacker $\mathbb{A}$ that starts at the sink q, the *capture time*, $\mathcal{TT}$, of s by $\mathbb{A}$ is equal to the length of the shortest path that joins q to s, measured in time units, under Q^s.

Since an attacker's attempts to capture the source should only be considered within some time bound [16], a *safety period* is needed to capture the time window during which the attacker's movements are considered. The safety period intuitively captures the time during which the asset is considered static at a given location, i.e., the asset is located at that single location. The safety period is typically set to be the product of a *safety factor*, denoted by ϕ, and the time to capture $\mathcal{TT}$. Thus, $P_{safety} = \phi\mathcal{TT}$. ϕ can be obtained using field data or using the estimated behaviour of the asset. For example, if the asset is static, then $\phi = \infty$. On the other hand, if the asset is very mobile, then $\phi \leq 1$. Typically, for semi-mobile, the safety factor is set to $1 \leq \phi < 2$, e.g., [16]. In this paper, we will focus on $\mathcal{TT} \leq P_{safety} < 2\mathcal{TT}$.

Definition 2 (Protectionless and SLP Routing Protocol): Given a network $G = (V, E)$, a source s, a sink q, a routing protocol Q^s sending messages from s, a safety period P_{safety}, and an adversary $\mathbb{A}$ that starts at q, we say that Q^s is protectionless if there exists a path where the adversary reaches s within P_{safety} time units, hence Q^s is denoted $Q^s_{\mathcal{P}}$. We say that Q^s provides SLP if the adversary does not reach s before P_{safety} time units along all paths, hence Q^s is denoted $Q^s_{\mathcal{S}}$, i.e., the attacker can only reach the source s after the P_{safety}.

E. Formal Preliminaries

The notations and definitions used in the rest of the paper will now be described before the analysis of the routing matrix transformation is subsequently presented. The discrete time

Algorithm 2 Convert Routing Protocol to Routing Matrix

Input: $Q_{\mathcal{P}}$ ▷ Routing Protocol (all shortest paths from source to sink)
Input: $G = (V, E)$ ▷ The graph containing network nodes and edges
Output: $\mathcal{R}_{\mathcal{P}}$ ▷ The protectionless routing matrix
to_nodes $\leftarrow \emptyset$ ▷ A mapping from a node to a set of nodes
edges_taken $\leftarrow \emptyset$ ▷ A set of edges
for path $\in Q_{\mathcal{P}}$ **do**
  for $(x, y) \in$ path **do**
    to_nodes$[y] \leftarrow$ from_nodes$[y] \cup \{x\}$
    edges_taken $\leftarrow$ edges_taken $\cup \{(x, y)\}$
for $x \in V$ **do**
  for $y \in V$ **do**
    if $(x, y) \in$ edges_taken $\wedge$ |to_nodes$[y]| > 0$ **then**
      $[\mathcal{R}_{\mathcal{P}}]_{xy} \leftarrow \frac{1}{|\text{to\_nodes}[y]|}$
    else
      $[\mathcal{R}_{\mathcal{P}}]_{xy} \leftarrow 0$

domain is denoted by $\mathcal{T} \subseteq \mathbb{Z}^{\geq 0}$. The attacker can receive a message and move to a single new location in one time unit.

We have defined a routing protocol $Q^s_{\mathcal{P}}$ for single source s to be a set of paths. For this paper we will also need to represent the actions that a message originating at s takes in terms of a matrix $\mathcal{R}$. Here, we take a receiver-centric approach that captures when a node will *receive* a message for the *first time*. This allows us to represent message reception transitions from a specific node. We have chosen this approach in order to align with an analysis from the attacker's perspective, as the nodes from which messages are received is more pertinent than understanding to which node a message is sent.

Definition 3 (Routing Matrix): $\mathcal{R}$ is a $|V| \times |V|$ routing matrix which represents a routing protocol Q, where V is the set of nodes in the network. $\mathcal{R}_{ij}$ represents the probability that node j receives a message from i *first*.

This definition differs from Armenia et al. [8] where routing matrices contain the probability that the routing algorithm chooses the next node. We will typically focus on two different routing matrices: one for protectionless routing $\mathcal{R}_{\mathcal{P}}$ and another for SLP-aware routing $\mathcal{R}_{\mathcal{S}}$. Algorithm 2 can be used to convert from our definition of a routing protocol to this routing matrix. Because attackers are initially located at the sink and they leverage the routing matrix to locate the source, we also define an attacker routing matrix.

Definition 4 (Attacker Routing Matrix): Given a routing matrix $\mathcal{R}$, an attacker routing matrix under $\mathcal{R}$, denoted by $\mathcal{R}^{\mathbb{A}}$, specifies the transitions an attacker could take and is given by the transpose of $\mathcal{R}$, with transitions (n, n) added with probability 1 for each node n when there are no edges that leave n in $\mathcal{R}$. These are added to so an attacker remains stationary when there are no messages sent to node n for them to follow, as is specified by the Patient Adversary.

There is an attacker routing matrix for both protectionless routing ($\mathcal{R}^{\mathbb{A}}_{\mathcal{P}}$) and SLP-aware routing ($\mathcal{R}^{\mathbb{A}}_{\mathcal{S}}$). The matrix $\mathcal{R}_{\mathcal{X}}$ indicates that the equation can be calculated for either $\mathcal{R}_{\mathcal{S}}$ or $\mathcal{R}_{\mathcal{P}}$. The transpose of matrix $\mathcal{R}_{\mathcal{X}}$ is indicated by $\mathcal{R}^{\top}_{\mathcal{X}}$.

$$\left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{ij} = \begin{cases} 1 & \text{if } i = j \wedge 0 = \sum_{k \in V} \left[\mathcal{R}^{\top}_{\mathcal{X}}\right]_{ik}, \\ \left[\mathcal{R}^{\top}_{\mathcal{X}}\right]_{ij} & \text{otherwise.} \end{cases} \quad (3)$$

Definition 5 (Attacker Transitions): The set of all possible transitions an attacker could take is a set of ordered pairs of nodes $\Gamma \subseteq V \times V$. The set of possible transitions that an attacker could take in $\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}$ is

$$\Gamma_{\mathcal{X}} = \left\{ (i, j) \mid (i, j) \in \Gamma \wedge \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{ij} > 0 \right\}. \quad (4)$$

$\mathcal{P}_\lambda$ is a random variable of attacker transitions $\Gamma_{\mathcal{P}}$, that occur at time λ, under a protectionless routing matrix $\mathcal{R}_{\mathcal{P}}$. $\mathcal{S}_\mu$ is a random variable of attacker transitions $\Gamma_{\mathcal{S}}$, that occur at time μ, under an SLP routing matrix $\mathcal{R}_{\mathcal{S}}$. We use $\mathcal{X}$ in equations to indicate that either $\mathcal{P}$ or $\mathcal{S}$ could be used to perform the calculation where $\mathcal{X}$ occurs.

The trace of clear time-series data of an attacker movement under a protectionless routing $\mathcal{R}_{\mathcal{P}}$ is a stochastic process $\mathbb{A}_{\mathcal{P}} = \{\mathbb{A}_{\mathcal{P}i}\}_{i \in \mathcal{T}}$, where the $\mathbb{A}_{\mathcal{P}i}$'s form a sequence of random variables of attacker positions in the network ($\mathbb{A}_{\mathcal{P}i} \in V$). The trace of noisy time-series data generated by SLP-aware routing $\mathcal{R}_{\mathcal{S}}$ is a stochastic process $\mathbb{A}_{\mathcal{S}} = \{\mathbb{A}_{\mathcal{S}i}\}_{i \in \mathcal{T}}$, where the $\mathbb{A}_{\mathcal{S}i}$'s form a sequence of random variables of attacker positions in the network ($\mathbb{A}_{\mathcal{S}i} \in V$).

In this paper, we focus on acyclic paths which typically have finite length, however, a finite path can be converted into an infinite path through the introduction of loops. For example, when a finite path terminates at the source, it can be augmented through the infinite repetition of the final node, i.e., $\langle n^1 \cdot n^2 \cdots (s, s) \cdot (s, s) \cdot (s, s) \cdots \rangle$. This is the only type of cycles allowed in paths. For a finite path p, the number of transitions (or path length) is denoted by $|p|$. The prefix of path p of length l is denoted by $l \uparrow p$.

Definition 6 (Source-converging and sink converging paths): A path $p = \langle n^1 \cdot n^2 \cdots n^j \rangle$ is a *source-converging* path if p ends at the source, i.e., $n^j_2 = s$. A path $p = \langle n^1 \cdot n^2 \cdots n^j \rangle$ is a *sink-converging* path if p ends at the sink, i.e., $n^j_2 = q$.

In this paper, unless specified otherwise, a path means an acyclic source-converging path.

F. Attacker Transition Probabilities

The probability functions for attacker movement are now defined. The probability that the attacker starting at node i at time t reaches j in exactly λ steps is:

$$\Pr\left(\mathbb{A}_{\mathcal{X}t+\lambda} = j \mid \mathbb{A}_{\mathcal{X}t} = i\right) = \left[\left(\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right)^{\lambda}\right]_{ij}. \quad (5)$$

In related work the attacker is assumed to start at the sink q. This model supports the attacker starting at an arbitrary location. However, this paper assumes that there is one starting location with a probability of 1, $\Pr\left(\mathbb{A}_{\mathcal{X}0} = q\right) = 1$.

The probability that an attacker takes a transition $n = (n_1, n_2)$ at time λ when its starting location is q can be calculated via:

$$\begin{aligned} &\Pr\left(\mathcal{X}_{t+\lambda} = n \mid \mathbb{A}_{\mathcal{X}t} = q\right) \\ &= \Pr\left(\mathbb{A}_{\mathcal{X}t+\lambda} = n_2 \mid \mathbb{A}_{\mathcal{X}t+\lambda-1} = n_1\right) \times \\ &\quad \Pr\left(\mathbb{A}_{\mathcal{X}t+\lambda-1} = n_1 \mid \mathbb{A}_{\mathcal{X}t} = q\right) \\ &= \left[\left(\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right)\right]_{n_1 n_2} \left[\left(\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right)^{\lambda-1}\right]_{qn_1} \end{aligned}. \quad (6)$$

The intuition is to calculate the probability the attacker reaches node n_1 at time $\lambda - 1$ and then takes the transition n at λ.

The probability that an adversary takes a walk $x_{0:n} = \langle x_0, \ldots, x_n \rangle$ through the network is:

$$\Pr\left(\mathbb{A}_{\mathcal{X}} n = x_n, \ldots, \mathbb{A}_{\mathcal{X}} 2 = x_2, \mathbb{A}_{\mathcal{X}} 1 = x_1, \mathbb{A}_{\mathcal{X}} 0 = x_0\right) = \Pr\left(\mathbb{A}_{\mathcal{X}} 0 = x_0\right) \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_0,x_1} \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_1,x_2} \ldots \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_{n-1},x_n} \quad (7)$$

The intuition is to calculate the product of the probability of starting at x_0 and the probabilities of taking each transition between pairs of locations along the route. An example distribution over the attacker start location is our assumption of the adversary starting at the sink q with probability 1.

G. Adversary Model is Markovian

In order to model the adversary using a Markov chain, we first show that Algorithm 1 implements Equation (3), and then we show that Equation (3) is Markovian (i.e., has the Markov property) when receiving messages from the routing matrix.

Corollary 1: The Patient Adversary in Algorithm 1 follows the routing protocol defined in Equation (3).

We use $\textsc{Recv}(\mathrm{msg}, n)$ to indicate that the adversary received a message msg from n. We can observe that any transition taken by an attacker is valid under the routing matrix $\mathcal{R}_{\mathcal{X}}$, i.e., the $\left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{ij}$ between two consecutive positions i and j of an attacker is greater than 0. Firstly, $\exists n \in V, \textsc{Recv}(\mathrm{msg}, n) \implies \textsc{IsFirstNewMessage}(\mathrm{msg})$ is true due to the definition of $\mathcal{R}_{\mathcal{X}}$ specifying that messages are received *first*, so every received message must be new. Secondly, from Equation (3) and the definition of $\mathcal{R}_{\mathcal{X}}$ the following are trivially true:

$$\begin{aligned}
\mathrm{pos} = \mathrm{s} &\implies \bigwedge_{n \in V} \neg\textsc{Recv}(\mathrm{msg}, n) \ , \\
\exists n \in V, \textsc{Recv}(\mathrm{msg}, n) &\implies \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{\mathrm{pos},n} > 0 \ , \\
\forall n \in V \setminus \{\mathrm{pos}\}, \neg\textsc{Recv}(\mathrm{msg}, n) &\implies \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{\mathrm{pos},n} = 0 \text{ , and} \\
\bigwedge_{n \in V} \neg\textsc{Recv}(\mathrm{msg}, n) &\implies \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{\mathrm{pos},\mathrm{pos}} = 1 \ .
\end{aligned}$$

Definition 7 (Markov Property): Given a stochastic process $X = \{X_t\}_{t \in \mathbb{Z}^{\geq 0}}$, then X is said to be Markovian if $\Pr\left(X_n = x_n \mid X_{n-1} = x_{n-1}, \ldots, X_0 = x_0\right) = \Pr\left(X_n = x_n \mid X_{n-1} = x_{n-1}\right)$ for all sequences of events $x_{0:n}$ [29].

Lemma 1 (Patient Adversary is Markovian): The stochastic process that implements the Patient Adversary (Algorithm 1) is Markovian.

Proof: Given a routing matrix $\mathcal{R}_{\mathcal{X}}$, a stochastic process of the attacker $\mathbb{A} = \{\mathbb{A}_t\}_{t \in \mathcal{T}}$, and the attacker routing matrix in Equation (3). Using the conditional probability definition $\Pr\left(A \mid B\right) = \frac{\Pr(A,B)}{\Pr(B)}$, we show that the Markov property holds:

$$\begin{aligned}
&\Pr\left(\mathbb{A}_n = x_n \mid \mathbb{A}_{n-1} = x_{n-1}, \ldots, \mathbb{A}_0 = x_0\right) = \\
&\frac{\Pr\left(\mathbb{A}_n = x_n, \mathbb{A}_{n-1} = x_{n-1}, \ldots, \mathbb{A}_0 = x_0\right)}{\Pr\left(\mathbb{A}_{n-1} = x_{n-1}, \ldots, \mathbb{A}_0 = x_0\right)} = \\
&\frac{\Pr\left(\mathbb{A}_{\mathcal{X}} 0 = x_0\right) \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_0,x_1} \ldots \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_{n-2},x_{n-1}} \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_{n-1},x_n}}{\Pr\left(\mathbb{A}_{\mathcal{X}} 0 = x_0\right) \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_0,x_1} \ldots \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_{n-2},x_{n-1}}} = \\
&\left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{x_{n-1},x_n} = \\
&\Pr\left(\mathbb{A}_n = x_n \mid \mathbb{A}_{n-1} = x_{n-1}\right) \ .
\end{aligned}$$

■

IV. Problem Statement

Using these definitions, we now formally state the problem. Given an attacker that is initially located at the sink and eavesdrops messages along a route between the sink and the source, the problem is to transform a protectionless routing matrix into a routing matrix that provides SLP. It is initially necessary to determine the (maximum) probability δ that the attacker will reach the source and capture the asset within a specified maximum time bound, termed as P_{safety}. The attacker is assumed to use the routing protocol to achieve its objective of reaching the source node. This means that the attacker will not randomly choose moves to take that are not possible when responding to the routing protocol. If the attacker does not make use of the routing protocol, then shortest path routing should be used.

Formally, the problem specification is as follows. Given:

Definition 8 (SLP Transformation Problem):

- A wireless sensor network $G = (V, E)$, where V is the set of nodes in the network and E is the set of wireless links between nodes,
- A mobile eavesdropping attacker $\mathbb{A}$ that is initially located at the sink $q \in V$,
- A single source location $s \in V$,
- A safety period P_{safety} (the upper time bound),
- A maximum capture threshold probability δ that determines the SLP level required, and
- A protectionless routing protocol $Q^s_{\mathcal{P}}$ that routes messages from s to q,

The objective is to derive a protectionless routing matrix $\mathcal{R}_{\mathcal{P}}$ from $Q^s_{\mathcal{P}}$ which is then transformed into a SLP-aware routing matrix $\mathcal{R}_{\mathcal{S}}$ such that:

- *Liveness*: There exists at least one sink-converging path from the source s to the sink q using $\mathcal{R}_{\mathcal{S}}$.
- *Privacy*: $\mathbb{A}$ reaches s with probability of at most δ within P_{safety} using $\mathcal{R}_{\mathcal{S}}$.

The liveness property ensures that the resulting routing protocol $\mathcal{R}_{\mathcal{S}}$ delivers messages to the sink and the privacy property ensures that the asset cannot be realistically caught within a certain time period $\mathcal{TT}$ and probability δ.

A routing protocol $Q_{\mathcal{S}}$ is called a P_{safety}-δ-SLP routing protocol (or simply an SLP-aware routing protocol) if it prevents the attacker from finding the source within P_{safety} time units with probability of at most δ. Specifically, the objective is to understand the steps required to transform a protectionless routing protocol into an SLP-aware routing protocol, i.e., to determine how messages should be received first in a P_{safety}-δ-SLP $Q_{\mathcal{S}}$. $Q_{\mathcal{P}}$ and $Q_{\mathcal{S}}$ do not need to specify routing protocols with similar transitions, except that there must be a path from the source to the sink.

One way towards solving this problem is to first develop a protocol and then perform a performance analysis of it to determine its efficacy [8]. Such an analysis identifies the level of SLP the algorithm is capable of providing and also allows the protocol to be refined based on the results of the analysis. However, while this technique is effective in demonstrating the performance of a specific technique in practice, it is *not suitable* for abstractly investigating optimal techniques in general.

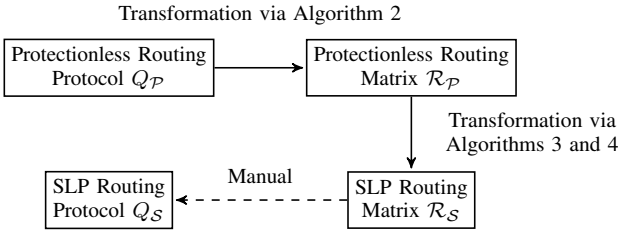


Figure 2: Methodology where $Q_{\mathcal{P}}$ is transformed into $\mathcal{R}_{\mathcal{P}}$ and $\mathcal{R}_{\mathcal{P}}$ is transformed into $\mathcal{R}_{\mathcal{S}}$, but algorithms to create paths in $Q_{\mathcal{S}}$ satisfying $\mathcal{R}_{\mathcal{S}}$ need to be implemented.

As an attacker takes a step along a single hop within a given time unit, its movement is modelled as a time series. This allows the SLP problem to be analysed from the perspective of privacy protection of time-series data. By structuring the analysis in this way, routing protocols and matrices can be abstractly considered in order to determine how to provide SLP. This paper will therefore consider the problem of quantifying the protection of time-series data (reception of messages and attacker movement) which has been perturbed by some arbitrary model. As the perturbation model is related to the transformation of $\mathcal{R}_{\mathcal{P}}$ into $\mathcal{R}_{\mathcal{S}}$, this provides a foundation for the manual design of an SLP routing protocol that potentially minimises privacy loss, i.e., a P_{safety}-0-SLP routing protocol. More precisely, our approach guarantees the generated routing matrix is SLP-aware by design. The remainder of this paper will focus on $\delta = 0$.

Figure 2 shows our methodology where the protectionless routing protocol $Q_{\mathcal{P}}$ is first translated into an abstract representation of a routing matrix $\mathcal{R}_{\mathcal{P}}$. The next step is for the protectionless routing matrix to be translated into an SLP-aware routing matrix $\mathcal{R}_{\mathcal{S}}$ via Algorithms 3 and 4. However, we do not specify how to convert from the routing matrix $\mathcal{R}_{\mathcal{S}}$ into an algorithm that can produce the paths that should be present in $Q_{\mathcal{S}}$. Instead we intend for the approach used by Bradbury and Jhumka [30] to be applied, where $\mathcal{R}_{\mathcal{S}}$ is used to guide the manual design of the SLP-aware routing protocol.

V. Problem Analysis

In this section, how a routing protocol can provide a high level of SLP is identified. To do this, a measure of privacy is needed to evaluate the level of SLP enhancement provided by a given solution. There are several potential definitions for privacy metrics, including information theoretic-based metrics and metrics involved with controlling statistics disclosure [15].

A. Routing Entropy

Initially, we focus on the entropy of a single random variable $H(\mathcal{X}_\lambda)$ and subsequently the divergence between two random variables $\mathcal{P}_\lambda$ and $\mathcal{S}_\mu$ to understand how routing protocols differ. Specifying λ and μ allows the difference between two protocols at different times to be examined.

The entropy of a random variable $H(\mathcal{X}_\lambda)$ indicates the uncertainty of an attacker taking a transition at time λ. In order to calculate the entropy, the starting location q of the attacker needs to be specified, hence we use entropy conditioned on the start location of the adversary. This means that the conditional entropy at λ will differ depending on where the attacker starts.

$$H(\mathcal{X}_\lambda \mid \mathbb{A}_{\mathcal{X}0} = q) = -\sum_{n \in \Gamma} L\left(\Pr\left(\mathcal{X}_\lambda = n \mid \mathbb{A}_{\mathcal{X}0} = q\right)\right), \quad \text{where } L(x) = x \log_2 x \ . \tag{8}$$

B. Routing Differences: Jensen-Shannon Divergence

To measure how much $\mathcal{R}_{\mathcal{S}}$ differs from $\mathcal{R}_{\mathcal{P}}$, a divergence measure can be used. Measures such as the Kullback–Leibler (KL) divergence [31] cannot be used because the invariant ($\forall n \in \Gamma : \Pr(\mathcal{S}_\mu = n) = 0 \implies \Pr(\mathcal{P}_\lambda = n) = 0$) required to use KL divergence does not always hold as there may be no link between the SLP-aware routing matrix and the protectionless routing matrix. Instead, an alternative measure such as the Jensen–Shannon divergence [32] (JSD) can be used as it does not assume such a link between routing matrices. The definition for JS divergence is shown in Equation (9) with $\mathcal{P}_\lambda$ and $\mathcal{S}_\mu$ weighted equally. These are conditioned on the attacker's start location, but it is omitted for brevity.

$$JSD(\mathcal{P}_\lambda \parallel \mathcal{S}_\mu) = H\left(\frac{\mathcal{P}_\lambda + \mathcal{S}_\mu}{2}\right) - \frac{H(\mathcal{P}_\lambda)}{2} + \frac{H(\mathcal{S}_\mu)}{2} \ . \tag{9}$$

Using JSD indicates how effective the transformation from $\mathcal{R}_{\mathcal{P}}$ to $\mathcal{R}_{\mathcal{S}}$ is at specific points in time (i.e., λ and μ). As the log base used to calculate entropy is 2 and the divergence is being calculated for two probability distributions, the divergence is bounded: $0 \leq JSD(\mathcal{P}_\lambda \parallel \mathcal{S}_\mu) \leq \log_2(2)$. This means that the upper bound of the divergence is 1. A higher divergence means that there are more differences between the two routing protocols. Ideally the JSD would equal 1 when $\lambda = \mu$ for a sufficient number of transitions (i.e., the safety period), indicating the two have fully diverged for this time.

C. Expected Capture Time and Capture Probability

A useful application of Markov chains is the ability to calculate the expected hitting time. This translates well to SLP as it is useful to know the expected capture time of a routing matrix. Using the hitting probability (h_{ij}) and expected hitting time of a Markov chain $E\left[\mathbb{A}_{\mathcal{X}t} = j \mid \mathbb{A}_{\mathcal{X}0} = i\right]$ (also written as $\mathcal{TT}_{ij}$), the expected capture time t when the attacker starts at i and the source is at j can be calculated. These equations can be calculated for $\mathbb{A}_{\mathcal{P}}$ and $\mathbb{A}_{\mathcal{S}}$.

$$h_{ij} = \begin{cases} 1 & \text{if } i = j, \\ \sum\limits_{k \in V \setminus \{i\}} \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{ik} h_{kj} & \text{otherwise.} \end{cases} \tag{10}$$

$$\mathcal{TT}_{ij} = \begin{cases} \infty & \text{if } h_{ij} < 1, \\ 0 & \text{if } i = j, \\ 1 + \sum\limits_{k \in V \setminus \{i\}} \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{ik} \mathcal{TT}_{kj} & \text{otherwise.} \end{cases} \tag{11}$$

The probability δ that the attacker reaches the source at j within P_{safety} hops when starting at i can be calculated using

$$\delta_{ij}^{P_{safety}} = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{if } i \neq j \wedge P_{safety} = 0, \\ \sum\limits_{k \in V \setminus \{i\}} \left[\mathcal{R}^{\mathbb{A}}_{\mathcal{X}}\right]_{ik} \delta_{kj}^{P_{safety}-1} & \text{otherwise.} \end{cases} \tag{12}$$

D. Information Loss from Privacy Preserving Data Mining

While the JSD works well for calculating the divergence at specific times, it doesn't provide a summary of an SLP-aware routing matrix. So, a notion of dissimilarity between $\mathcal{R}_{\mathcal{P}}$ and $\mathcal{R}_{\mathcal{S}}$ is needed which is indicative of the SLP level provided by $\mathcal{R}_{\mathcal{S}}$. The notion of information loss varies inversely with privacy loss [33], i.e., the higher the information that is lost or the more perturbed the clear data time-series is, the less privacy is lost. However, by diverting onto the noisy time series for message transmission (and therefore attacker movement) to maximise the information loss, a cost will be incurred. These costs (such as, energy, latency, packet delivery ratio, and others) have been widely investigated in the SLP literature [4].

To this end, the definition of information loss in Equation (13), which is used in privacy-preserving data mining [15, 34], is adapted to the SLP problem in Equation (14). Here, $D_{\mathcal{P}}$ and $D_{\mathcal{S}}$ represent the clear and noisy domains respectively, and $f_D(i)$ represents the frequency of the data item i in domain D.

$$IL(D_{\mathcal{P}}, D_{\mathcal{S}}) = \frac{\sum_{i=1}^{n} |f_{D_{\mathcal{P}}}(i) - f_{D_{\mathcal{S}}}(i)|}{\sum_{i=1}^{n} f_{D_{\mathcal{P}}}(i)} \ . \tag{13}$$

Since transitions do not contribute to attacker information gain after the safety period has elapsed, the information loss definition is adapted to only include transitions in $\mathcal{R}_{\mathcal{S}}$ that occur before the safety period expires.

$$IL(D_{\mathcal{P}}, D_{\mathcal{S}}) = \frac{\sum_{i=1}^{n} |f_{D_{\mathcal{P}}}(i) - f_{D_{\mathcal{S}}}(i^{P_{safety}})|}{\sum_{i=1}^{n} f_{D_{\mathcal{P}}}(i)} \ . \tag{14}$$

where $f_{\mathcal{D}_{\mathcal{P}}}(i)$ and $f_{\mathcal{D}_{\mathcal{S}}}(i^{P_{safety}})$ are defined as:

$$f_{\mathcal{D}_{\mathcal{P}}}(i) = \begin{cases} 1 & \text{if transition } i \text{ is used in } \mathcal{R}_{\mathcal{P}}, \\ 0 & \text{otherwise.} \end{cases} \tag{15}$$

$$f_{\mathcal{D}_{\mathcal{S}}}(i^{P_{safety}}) = \begin{cases} 1 & \text{if } i \text{ is not taken within} \\ & P_{safety} \text{ steps in } \mathcal{R}_{\mathcal{S}}, \\ 0 & \text{otherwise.} \end{cases} \tag{16}$$

Equation (14) states that the more dissimilar the set of transitions taken within P_{safety}, the greater the information loss, hence the lower the privacy loss. If $IL(D_{\mathcal{P}}, D_{\mathcal{S}}) = 1$ (i.e., is maximum), then it implies that $D_{\mathcal{P}} \cap D_{\mathcal{S}} = \emptyset$. Therefore, to minimise privacy loss, $\mathcal{R}_{\mathcal{P}}$ and $\mathcal{R}_{\mathcal{S}}$ cannot share any transitions. More specifically, it means that, though $\mathcal{R}_{\mathcal{N}}$ and $\mathcal{R}_{\mathcal{S}}$ can share transitions, an attacker cannot take some transition in $\mathcal{R}_{\mathcal{P}}$ under $\mathcal{R}_{\mathcal{S}}$ within P_{safety} time units. In order to obtain $\mathcal{R}_{\mathcal{S}}$, $\mathcal{R}_{\mathcal{P}}$ has to be transformed in such a way that *for a certain duration*, for any transition (i, j) unique to $\mathcal{R}_{\mathcal{S}}$, an attacker at location j needs to receive a message from node i *first*.

VI. Perturbation via Proper Competing Paths

Until now, we have conceptually studied how to minimise privacy loss in a routing protocol, by maximising information loss. To understand how to concretely transform $\mathcal{R}_{\mathcal{P}}$ into $\mathcal{R}_{\mathcal{S}}$, i.e., to understand how $\mathcal{R}_{\mathcal{P}}$ can perturbed into $\mathcal{R}_{\mathcal{S}}$, we introduce the concept of *competing paths*. A visualisation of the concept is shown in Figure 3.

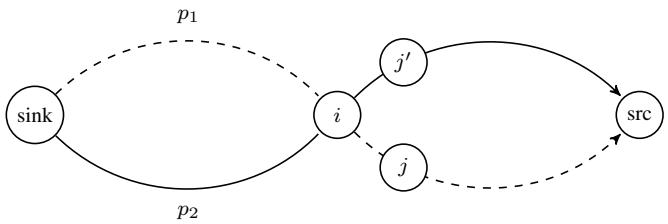


Figure 3: Demonstration of two paths p_1 and p_2 the attacker could take that compete at node i.

A. Competing Paths

Definition 9 (Competing Paths): Given a network $G = (V, E)$ and a protectionless routing matrix $\mathcal{R}_{\mathcal{P}}$, two distinct paths p_1 and p_2 under $\mathcal{R}_{\mathcal{P}}$ compete at a node $n \in V$ iff the all of following are satisfied:

- p_1 and p_2 are source-converging paths,
- $\exists (i, j), (i, j') \in E : (i, j) \in p_1 \wedge (i, j') \in p_2 \wedge i = n$, and
- $\forall j, j' \in V : j \neq j' \implies [\mathcal{R}_{\mathcal{P}}]_{jn} > 0 \wedge [\mathcal{R}_{\mathcal{P}}]_{j'n} \geq 0$.

Definition 10 (Junction Node): A node $n \in V$ is called a *junction* node if multiple paths compete at n.

The idea of competing paths is that if one path is part of the clear data time-series, then the other can be used in the noisy data time-series. Specifically, it means that if the attacker has reached a junction node n for following a given path p_1 under $\mathcal{R}_{\mathcal{P}}$, then the attacker can be made to follow an alternative path p_2 (from n onwards) under $\mathcal{R}_{\mathcal{S}}$. In this definition the node n is called a *junction node*, p_1 is called a normal path, and p_2 a perturbed path. The example in Figure 4a can be considered. Since $[\mathcal{R}_{\mathcal{P}}]_{2,5} = 0.5$ and $[\mathcal{R}_{\mathcal{P}}]_{4,5} = 0.5$, then paths $\langle (5, 2) \cdot (2, 1) \rangle$ and $\langle (5, 4) \cdot (4, 1) \rangle$ compete at node 5. The more paths that compete at a junction node, the higher the entropy at that node.

Corollary 2: When there are two or more unique source-converging paths in $\bigcup_{v \in V} \textsc{Paths}(v, q)$, they all compete at the sink q.

B. Proper Competing Paths

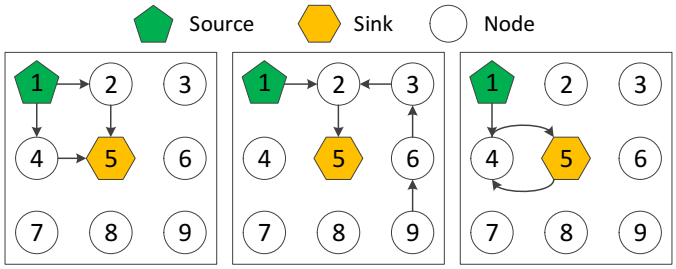
However, not all competing paths are capable of preventing the attacker from reaching the source within the safety period. For example, in Figure 4a the two paths both compete at node 5, but neither prevent the attacker from reaching the source within a safety period of 4 time units. The notion of competing paths is thus strengthened to that of *proper competing paths*.

Definition 11 (Proper Competing Paths): Given a network $G = (V, E)$ and a protectionless routing protocol $\mathcal{R}_{\mathcal{P}}$, two distinct paths p_1 and p_2 under $\mathcal{R}_{\mathcal{P}}$ compete properly at a node $n \in V$ iff all of the following are satisfied:

- p_1 and p_2 are source-converging paths,
- $\exists (i, j), (i, j') \in E : (i, j) \in p_1 \wedge (i, j') \in p_2 \wedge i = n$, and
- $\forall j, j' \in V : j \neq j' \implies [\mathcal{R}_{\mathcal{P}}]_{jn} > 0 \wedge [\mathcal{R}_{\mathcal{P}}]_{j'n} = 0$.

Definition 12 (Proper Junction Node): A node n is a proper junction node if multiple proper competing paths compete at n.

Here, for two proper competing paths, the attacker cannot receive the message first along one of these paths. Thus, path p_1



(a) JN at 5 (b) JN at 2 (c) JN at 4 and 5

Figure 4: Examples of competing paths and their Junction Nodes (JNs).

should be perturbed into path p_2 in the noisy data time-series. The intuition is that the attacker, at a proper junction node, has two distinct choices and one of those choices is one it would unlikely have made under normal circumstances. As before, p_1 is called a normal path and p_2 a properly perturbed path.

C. Existence of Proper Competing Paths

An issue with SLP routing protocol is that they require an element of redundancy [35] in order to provide SLP. Therefore, we now show that proper completing paths cannot always exist and under what circumstances a proper competing path needs to exist for an SLP routing protocol to exist.

Lemma 2: Given a network $G = (V, E)$, a source s, an attacker $\mathbb{A}$ that starts at the sink q, a protectionless routing protocol $Q^s_{\mathcal{P}}$ from s, safety period P_{safety}, and a path $p_1 \in Q^s_{\mathcal{P}}$ with $|p_1| \leq P_{safety}$, then there does not exist a path $p_2 \in Q^s_{\mathcal{P}}$ such that p_1 and p_2 properly compete at n.

Proof: We prove this by contradiction by assuming that such a p_2 exists (properly competing) and then showing that $p_2 \notin Q^s_{\mathcal{P}}$. We assume that $p_2 \in Q^s_{\mathcal{P}}$. Since p_1 and p_2 properly compete at n, and given that $p_1 \in Q^s_{\mathcal{P}}$, it means that the probability of q (hence $\mathbb{A}$) receiving the message first along p_1 is 1. Thus, the probability of receiving first along p_2 is 0, meaning that $p_2 \notin Q^s_{\mathcal{P}}$, contradicting our assumption. ■

Intuitively, the result suggests that, if $Q^s_{\mathcal{P}}$ had such a path p_2, $Q^s_{\mathcal{P}}$ would not have been protectionless. This result also suggests that such a path p_2 will properly compete if $p_2 \notin Q^s_{\mathcal{P}}$. This leads to our next important result which shows when an SLP routing protocol exists.

Theorem 1: Given a network $G = (V, E)$ with a source s, an attacker $\mathbb{A}$ that starts at the sink q, safety period P_{safety}, a protectionless routing protocol $Q^s_{\mathcal{P}}$ from s, then there exists a P_{safety}-0-SLP routing protocol $Q^s_{\mathcal{S}}$ if and only if there exists a path $p \notin Q^s_{\mathcal{P}}$ with $|p| > P_{safety}$ and, $\forall p_1 \in Q^s_{\mathcal{P}}$, p_1 properly competes with p at q.

Proof: [$\Leftarrow$] We assume that there exists a path $p \notin Q^s_{\mathcal{P}}$ with $|p| > P_{safety}$ and, $\forall p_1 \in Q^s_{\mathcal{P}}$, p_1 properly competes with p at q. Since p and p_1 properly compete at q, it means that q never received a message first along p. Also, given that $|p| > P_{safety}$, it means that the attacker cannot reach the source before P_{safety} has elapsed, i.e., the attacker $\mathbb{A}$ cannot capture the source before P_{safety}. Thus, all paths such as p are included in $Q^s_{\mathcal{S}}$. Hence, $Q^s_{\mathcal{S}}$ is P_{safety}-0-SLP.

[$\Rightarrow$] As $Q^s_{\mathcal{S}}$ is P_{safety}-0-SLP, it means that all paths $p \in Q^s_{\mathcal{S}}$, $|p| > P_{safety}$, implying that $p \notin Q^s_{\mathcal{P}}$. Because $p \notin Q^s_{\mathcal{P}}$, it means q cannot receive messages first along p in $Q^s_{\mathcal{P}}$. Thus, $\forall p_1 \in Q^s_{\mathcal{P}}$, p and p_1 properly compete at q. ■

The intuition is that p_1 is a path that an attacker may follow under the protectionless protocol $Q^s_{\mathcal{P}}$ to capture the asset, while p provides a *diversion* via a path that the attacker will not normally follow which contains enough moves to adequately delay the attacker. Further, p also captures the fact that p_1 can be perturbed into p at the identified proper junction node. A path p will need to be guaranteed to exist under a SLP-aware routing protocol $Q^s_{\mathcal{S}}$.

Corollary 3: The length of a path in the P_{safety}-0-SLP routing protocol is bounded below by the safety period P_{safety} and bounded above by the number of nodes in the network.

There are different network topologies where such conditions exist. For example, a network with the sink in the centre of the network and the source at the extremity of the network (e.g., Figure 4a) where paths $\langle (1,2) \cdot (2,5) \rangle$ and $\langle (1,4) \cdot (4,5) \rangle$ are in $\mathcal{R}_{\mathcal{P}}$. Path $p = \langle (1,2) \cdot (2,3) \cdot (3,6) \cdot (6,5) \rangle \notin \mathcal{R}_{\mathcal{P}}$ would properly compete with $\langle (1,4) \cdot (4,5) \rangle$ at the sink.

Using Figures 4a and 4c as an example, there are two paths p_1 and p_2. Setting $P_{safety} = 4$ the paths are expanded out to p'_1 and p'_2. The attacker reaches the asset at node 1 in p_1 within P_{safety} steps but does not in p_2.

$$
\begin{aligned}
p_1 &= \langle (5,4) \cdot (4,1) \rangle \\
p'_1 &= \langle (5,4) \cdot (4,1) \cdot (1,1) \cdot (1,1) \cdot (1,1) \cdot (1,1) \cdot (1,1) \rangle \\
p_2 &= \langle (5,4) \cdot (4,5) \cdot (5,4) \cdot (4,5) \cdot (5,4) \cdot (4,1) \rangle \\
p'_2 &= \langle (5,4) \cdot (4,5) \cdot (5,4) \cdot (4,5) \cdot (5,4) \cdot (4,1) \cdot (1,1) \rangle
\end{aligned}
$$

D. Generating Proper Competing Paths

In order to derive $\mathcal{R}_{\mathcal{S}}$ from $\mathcal{R}_{\mathcal{P}}$ a proper junction node now needs to be selected through which a path in $\mathcal{R}_{\mathcal{P}}$ can be perturbed. The heuristic in Algorithm 3 is proposed to generate the set of properly perturbed paths at possible proper junction nodes. Normal paths in $\mathcal{R}_{\mathcal{P}}$ are iterated and all nodes are considered as proper junction node candidates. Those that meet the appropriate conditions (Line 12 in Algorithm 3) are used to generate a properly perturbed path. The SLP-aware routing matrix $\mathcal{R}_{\mathcal{S}}$ can then be generated using Algorithm 4. Note that the heuristic does not define certain methods (such as CHOOSE) as different definitions will lead to different SLP-aware routing protocols being produced.

The set of properly perturbed paths that Algorithm 3 generates is shown below, and the CHOOSE function picks the fourth path in Algorithm 4 in the following examples.

$$
\begin{aligned}
\{\ &\langle (1,2) \cdot (2,3) \cdot (3,6) \cdot (6,9) \cdot (9,8) \cdot (8,5) \rangle, \\
&\langle (1,2) \cdot (2,3) \cdot (3,6) \cdot (6,9) \cdot (9,8) \cdot (8,7) \cdot (7,4) \cdot (4,5) \rangle, \\
&\langle (1,4) \cdot (4,7) \cdot (7,8) \cdot (8,9) \cdot (9,6) \cdot (6,3) \cdot (3,2) \cdot (2,5) \rangle, \\
&\langle (1,4) \cdot (4,7) \cdot (7,8) \cdot (8,9) \cdot (9,6) \cdot (6,5) \rangle\ \}
\end{aligned}
$$

Theorem 2 (Privacy Loss): Given, a network $G = (V, E)$, a source location $s \in V$, a sink location $q \in V$, a distributed eavesdropper attacker $\mathbb{A}$ that is initially located at q, a safety factor ϕ, and a protectionless routing matrix $\mathcal{R}_{\mathcal{P}}$ with expected

Algorithm 3 Generating set of properly perturbed paths with a length of at least P_{safety} hops.

function PROPERLYPERTURBEDPATHS($\mathcal{R}_\mathcal{P}$, ϕ)
  PPP $\leftarrow \emptyset$
  N $\leftarrow$ set of all normal paths from sink q to source s in $\mathcal{R}^{\mathbb{A}}_\mathcal{P}$
  $\mathcal{TT} \leftarrow E\left[\mathbb{A}_{\mathcal{P}t} = s \mid \mathbb{A}_{\mathcal{P}0} = q\right]$ ▷ Expected capture time
  $P_{safety} \leftarrow \lceil \mathcal{TT} \times \phi \rceil$ ▷ ϕ is the safety factor
  for path $\in$ N **do**
    used $\leftarrow \langle\rangle$
    for $(i, j) \in$ path **do** ▷ For each transition in the path
      if $i = s$ **then** ▷ End when the source is reached
        break
      ▷ Find an unused node n to perturb p though
      for $n \in V : \left[\mathcal{R}^{\mathbb{A}}_\mathcal{P}\right]_{in} = 0$ **do**
        pp $\leftarrow \{\, p \mid$ generate path p from q to s where
          $(|\text{used}| \uparrow p) = \text{used} \wedge$ ▷ Path starts the same
          $p^{|\text{used}|+1} = (i, n) \wedge$ ▷ Path goes via n
          $(\lceil \mathcal{TT} \rceil \uparrow p) \notin \mathcal{R}_\mathcal{P} \wedge$ ▷ Eliminate paths in $\mathcal{R}_\mathcal{P}$
          $|p| > P_{safety} \,\}$ ▷ $\mathbb{A}$ does not reach s within P_{safety}
        ▷ Reverse the path
        pp $\leftarrow \{\, \langle (n_2, n_1) \mid n \in \text{REVERSE}(p) \rangle \mid p \in \text{pp} \,\}$
        PPP $\leftarrow$ PPP $\cup$ pp
      used $\leftarrow$ used $\frown \langle (i, j) \rangle$
  return PPP

Algorithm 4 Generating SLP-aware routing matrix

▷ Generate a set of paths from the source s to the sink q for messages to follow. The attacker follows the reverse of this path.
function PERTURBNORMALROUTING($\mathcal{R}_\mathcal{P}$, ϕ)
  $\mathcal{R}_\mathcal{S} \leftarrow \mathcal{R}_\mathcal{P}$
  PPP $\leftarrow$ PROPERLYPERTURBEDPATHS($\mathcal{R}_\mathcal{P}$, ϕ)
  p $\leftarrow$ CHOOSE(PPP) ▷ Choose one path, fails if no paths
  for $(i, j) \in p$ **do**
    for $k \in V \setminus \{i\}$ **do** ▷ j must receive from i
      $[\mathcal{R}_\mathcal{S}]_{kj} \leftarrow 0$
    $[\mathcal{R}_\mathcal{S}]_{ij} \leftarrow 1$ ▷ $\mathbb{A}$ follows chosen properly perturbed path
  ▷ Remove paths from s that no longer terminate at q
  return REMOVENONTERMINATINGPATHS($\mathcal{R}_\mathcal{S}$)

time to capture $\mathcal{TT}$, then $\mathcal{R}_\mathcal{S}$ generated with Algorithm 4 (if such a routing matrix exists) results in at least $(\phi - 1)\mathcal{TT} + 2$ steps where $JSD(\mathcal{P}_\lambda \parallel \mathcal{S}_\lambda) = 1$.

Lemma 3: At least $(\phi - 1)\mathcal{TT} + 2$ steps completely diverge in $\mathcal{R}_\mathcal{S}$ compared to $\mathcal{R}_\mathcal{P}$.

Proof: By construction. As the paths generated must be longer than $\phi\mathcal{TT}$ (the safety period), the minimum length of a properly perturbed path must be $\phi\mathcal{TT} + 1$. The maximum number of overlapping steps the perturbed path can have is $\mathcal{TT} - 1$. So the minimum number of diverged steps that the path must have is $\phi\mathcal{TT} + 1 - (\mathcal{TT} - 1) = (\phi - 1)\mathcal{TT} + 2$. ■

Lemma 4: When a transition diverges at λ, for all $n \in \Gamma$:

$$\Pr(\mathcal{S}_\lambda = n) = 1 \implies \Pr(\mathcal{P}_\lambda = n) = 0 \text{ and} \tag{17}$$
$$\Pr(\mathcal{P}_\lambda = n) > 0 \implies \Pr(\mathcal{S}_\lambda = n) = 0\ . \tag{18}$$

These properties lead to $JSD(\mathcal{P}_\lambda \parallel \mathcal{S}_\lambda) = 1$.

Proof: There are three cases that need to be considered:
Case 1: $\Pr(\mathcal{S}_\lambda = n) = 1$, where it is also the case that $\Pr(\mathcal{P}_\lambda = n) = 0$ due to Equation (17).

$$\frac{L\left(\Pr(\mathcal{P}_\lambda = n)\right)}{2}^{0} + \frac{L\left(\Pr(\mathcal{S}_\lambda = n)\right)}{2}^{0} - L\left(\frac{\Pr(\mathcal{P}_\lambda = n) + \Pr(\mathcal{S}_\lambda = n)}{2}\right)^{0.5}. \tag{19}$$

So when n is taken under $\mathcal{R}_\mathcal{S}$ the expression inside the sum has the value $-L(0.5) = 0.5$.
Case 2: $\Pr(\mathcal{P}_\lambda = n) > 0$, where it is also the case that $\Pr(\mathcal{S}_\lambda = n) = 0$ due to Equation (18).

$$\frac{L\left(\Pr(\mathcal{P}_\lambda = n)\right)}{2} + \frac{L\left(\Pr(\mathcal{S}_\lambda = n)\right)}{2}^{0} - L\left(\frac{\Pr(\mathcal{P}_\lambda = n) + \Pr(\mathcal{S}_\lambda = n)}{2}\right)^{0}$$
$$= \frac{L\left(\Pr(\mathcal{P}_\lambda = n)\right)}{2} - L\left(\frac{\Pr(\mathcal{P}_\lambda = n)}{2}\right) = \frac{\Pr(\mathcal{P}_\lambda = n)}{2}\ . \tag{20}$$

As $1 = \sum_{n \in \Gamma} \Pr(\mathcal{P}_\lambda = n)$ then $0.5 = \sum_{n \in \Gamma} \frac{\Pr(\mathcal{P}_\lambda = n)}{2}$.
Case 3: $\Pr(\mathcal{P}_\lambda = n) = 0 \wedge \Pr(\mathcal{S}_\lambda = n) = 0$

$$\frac{L\left(\Pr(\mathcal{P}_\lambda = n)\right)}{2}^{0} + \frac{L\left(\Pr(\mathcal{S}_\lambda = n)\right)}{2}^{0} - L\left(\frac{\Pr(\mathcal{P}_\lambda = n) + \Pr(\mathcal{S}_\lambda = n)}{2}\right)^{0}. \tag{21}$$

This leaves the summation to obtain the Jensen-Shannon divergence as:

$$1 = \sum_{n \in \Gamma} \begin{cases} 0.5 & \text{if } \Pr(\mathcal{S}_\lambda = n) = 1 \\ \frac{\Pr(\mathcal{P}_\lambda = n)}{2} & \text{if } \Pr(\mathcal{P}_\lambda = n) > 0 \\ 0 & \text{if } \Pr(\mathcal{P}_\lambda = n) = 0 \wedge \Pr(\mathcal{S}_\lambda = n) = 0\ . \end{cases} \tag{22}$$

■

VII. CASE STUDIES

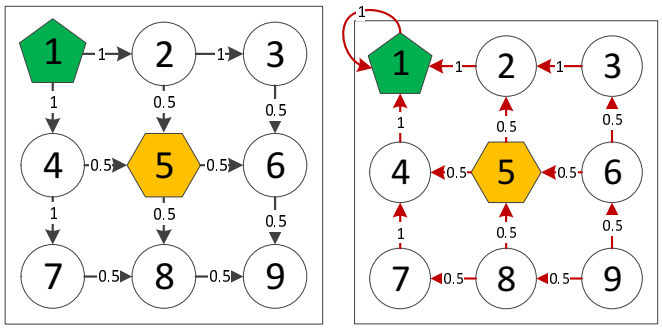
A. Using Algorithm 4 to perturb protectionless flooding

In this section, a case study is used to show the viability of this approach. Using the network shown in Figure 5c for $\mathcal{R}_\mathcal{P}$ and Figure 6c for $\mathcal{R}_\mathcal{S}$, with the sink at the centre of the network and the source on the border. Here, $[\mathcal{R}_\mathcal{P}]_{1,2} = 1$ means that node 2 will receive a message first from node 1 and $[\mathcal{R}_\mathcal{P}]_{2,5} = 0.5$ means there is a 50% chance that node 5 will receive a message first from node 2 (the other possibility is from 4). The set of normal paths is given by $\{\langle (5,2) \cdot (2,1) \rangle, \langle (5,4) \cdot (4,1) \rangle\}$. the safety period is 4 as the expected capture time is 2 and the safety factor is 2.

In this example, the first proper junction node is node 5, the sink. Any properly perturbed path will start with transition $(5, 6)$ or $(5, 8)$. Applying the heuristic to generate $\mathcal{R}_\mathcal{S}$ means that relevant transition probabilities need to be replaced. This is shown in the matrix $\mathcal{R}_\mathcal{S}$, where the old value is replaced by the new value shown in bold. For example, it means that $[\mathcal{R}_\mathcal{S}]_{6,5}$ needs to be set to 1 to ensure that the attacker moves to node 6 from the sink, rather than moving towards either node 2 or 4. This is performed for each transition in one of the properly perturbed paths.

An attacker will now take the path $\langle (5,6) \cdot (6,9) \cdot (9,8) \cdot (8,7) \cdot (7,4) \cdot (4,1) \rangle$, meaning that the attacker requires 6 transitions to reach the source, which is more than the safety period. Hence, it means that the attacker cannot catch the source before the safety period expires.

The information loss of $\mathcal{R}_\mathcal{S}$ compared to $\mathcal{R}_\mathcal{P}$ is calculated using Equation (14). There are twelve transitions present in the domain of $\Gamma_\mathcal{P}$ shown in $D_\mathcal{P}$. There are six transitions in the domain of $\Gamma_\mathcal{S}$ (shown in $D_\mathcal{S}$), of which only four are reachable within the safety period (shown in $D_\mathcal{S}^{P_{safety}}$). Table II shows



(a) $\mathcal{R}_{\mathcal{P}}$ (b) $\mathcal{R}_{\mathcal{P}}^{\mathbb{A}}$

Receiving Nodes

Sending Node

	1	2	3	4	5	6	7	8	9
1	0	1	0	1	0	0	0	0	0
2	0	0	1	0	0.5	0	0	0	0
3	0	0	0	0	0	0.5	0	0	0
4	0	0	0	0	0.5	0	1	0	0
5	0	0	0	0	0	0.5	0	0.5	0
6	0	0	0	0	0	0	0	0	0.5
7	0	0	0	0	0	0	0	0.5	0
8	0	0	0	0	0	0	0	0	0.5
9	0	0	0	0	0	0	0	0	0

(c) Routing matrix $\mathcal{R}_{\mathcal{P}}$

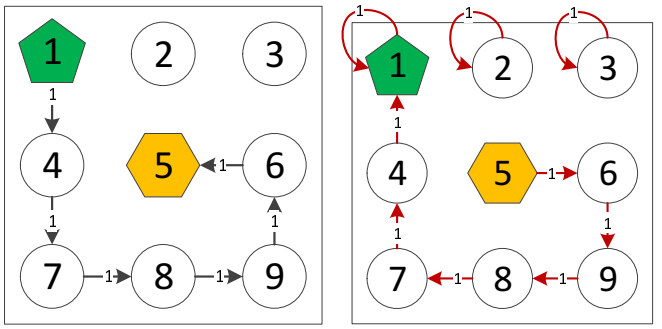
Sending Node (Moving To)

Recv. Nodes (Moving From)

	1	2	3	4	5	6	7	8	9
1	1	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0	0
3	0	1	0	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0
5	0	0.5	0	0.5	0	0	0	0	0
6	0	0	0.5	0	0.5	0	0	0	0
7	0	0	0	1	0	0	0	0	0
8	0	0	0	0	0.5	0	0.5	0	0
9	0	0	0	0	0	0.5	0	0.5	0

(d) Attacker movement matrix $\mathcal{R}_{\mathcal{P}}^{\mathbb{A}}$

Figure 5: *Protectionless* flooding route



(a) $\mathcal{R}_{\mathcal{S}}$ (b) $\mathcal{R}_{\mathcal{S}}^{\mathbb{A}}$

Receiving Nodes

Sending Node

	1	2	3	4	5	6	7	8	9
1	0	~~1~~ **0**	0	1	0	0	0	0	0
2	0	0	~~1~~ **0**	0	~~0.5~~ **0**	0	0	0	0
3	0	0	0	0	0	~~0.5~~ **0**	0	0	0
4	0	0	0	0	~~0.5~~ **0**	0	1	0	0
5	0	0	0	0	0	~~0.5~~ **0**	0	~~0.5~~ **0**	0
6	0	0	0	0	~~0~~ **1**	0	0	0	~~0.5~~ **0**
7	0	0	0	0	0	0	0	~~0.5~~ **1**	0
8	0	0	0	0	0	0	0	0	~~0.5~~ **1**
9	0	0	0	0	0	~~0~~ **1**	0	0	0

(c) Routing matrix $\mathcal{R}_{\mathcal{S}}$

Sending Node (Moving To)

Recv. Nodes (Moving From)

	1	2	3	4	5	6	7	8	9
1	1	0	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0
4	1	0	0	0	0	0	0	0	0
5	0	0	0	0	0	1	0	0	0
6	0	0	0	0	0	0	0	0	1
7	0	0	0	1	0	0	0	0	0
8	0	0	0	0	0	0	1	0	0
9	0	0	0	0	0	0	0	1	0

(d) Attacker movement matrix $\mathcal{R}_{\mathcal{S}}^{\mathbb{A}}$

Figure 6: An example SLP-aware route

which transitions are present in both $D_{\mathcal{P}}$ and $D_{\mathcal{S}}^{P_{safety}}$. Using this result, (where ✗ is a 1 and ✓ is a 0) the information loss is calculated in Equation (26).

These results show that the information loss from the attacker's perspective is 83%, meaning it can only observe 17% of the original routing matrix. By reducing the amount of information the attacker gains, it can be prevented from finding the source within the safety period.

$$D_{\mathcal{P}} = \{(1,2),(1,4),(2,3),(2,5),(3,6),(4,5),(4,7),(5,6),(5,8),(6,9),(7,8),(8,9)\} \quad (23)$$

$$D_{\mathcal{S}} = \{(1,4),(4,7),(7,8),(8,9),(9,6),(6,5)\} \quad (24)$$

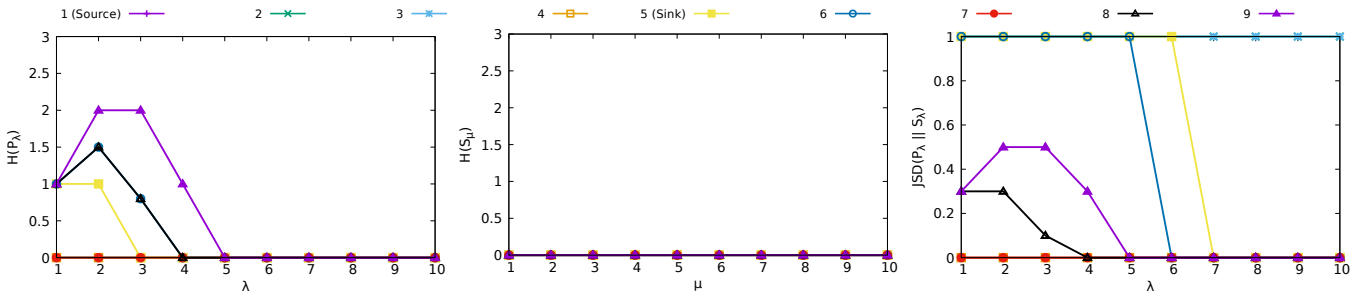
$$D_{\mathcal{S}}^{P_{safety}} = \{(7,8),(8,9),(9,6),(6,5)\} \quad (25)$$

$D_{\mathcal{P}_1}$	1	1	2	2	3	4	4	5	5	6	7	8
$D_{\mathcal{P}_2}$	2	4	3	5	6	5	7	6	8	9	8	9
$D_{\mathcal{S}}^{P_{safety}}$	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓

Table II: Are the transitions from $D_{\mathcal{P}}$ in $D_{\mathcal{S}}^{P_{safety}}$?

$$IL(D_{\mathcal{P}}, D_{\mathcal{S}}) = \frac{1+1+1+1+1+1+1+1+1+1+1+1+0+0}{12} = \frac{10}{12} = \frac{5}{6} \quad (26)$$

In addition to using the measure of information loss, the two routing matrices $\mathcal{R}_{\mathcal{P}}$ and $\mathcal{R}_{\mathcal{S}}$ can be compared using JSD. The entropy for the two example routing matrices in Figures 5c and 6c is shown in Figures 7a and 7b. Because there is only one path between the sink and source in $\mathcal{R}_{\mathcal{S}}$ the attacker will always take the same path, hence the entropy is 0 for all starting locations as there is no uncertainty in its actions. When there are multiple paths to the source from the



(a) Entropy of $\mathcal{P}_\lambda$ (b) Entropy of $\mathcal{S}_\mu$ (c) JSD between $\mathcal{P}_\lambda$ and $\mathcal{S}_\lambda$

Figure 7: Entropy and Jensen–Shannon divergence at different times based on the attacker's starting position

sink in $\mathcal{R}_\mathcal{P}$ the entropy is non-zero, with a higher number of paths leading to a higher entropy. This can be observed when the attacker starts at node 9 as entropy is greater than when it starts at node 5.

So far, the analysis has focused on individual examples of routing. To compare $\mathcal{R}_\mathcal{S}$ to $\mathcal{R}_\mathcal{P}$ Figure 7c shows the JSD at the same times for the two attacker movement matrices, which demonstrates a number of interesting aspects. First, is that the actions the attacker takes when starting at 2 or 3 have completely diverged. This is because no messages are routed through 2 or 3 in $\mathcal{R}_\mathcal{S}$. Nodes 1, 4, and 7 have no divergence as the actions the attacker could take when starting at those locations are the same for both routing matrices. The remaining starting points (5, 6, 8 and 9) diverge early on, but converge at a later time when the attacker reaches the source. The most interesting of these four points is when the attacker starts at node 5. The aim in transforming $\mathcal{R}_\mathcal{P}$ into $\mathcal{R}_\mathcal{S}$ was to prevent the attacker from reaching the source within the safety period of 4 steps. Figure 7c shows that $\mathcal{R}_\mathcal{P}$ and $\mathcal{R}_\mathcal{S}$ have fully diverged until the 7^{th} step when the attacker under either routing matrix would then be at the source.

Figure 8 shows the JSD at different times in $\mathcal{P}_\lambda$ and $\mathcal{S}_\mu$ (rather than when $\lambda = \mu$ in Figure 7) when the attacker starts at nodes 1 to 9. These graphs allow routing matrices to be compared in more detail at specific times by showing how diverged the actions an attacker could take in $\mathcal{R}_\mathcal{P}$ at λ compared to the actions it could take in $\mathcal{R}_\mathcal{S}$ at μ. Where 1 means that the actions have fully diverged and 0 means that the actions are the same. The height of the bottom two rows in Figure 8d (when the attacker starts at 5) that diverge is the same as the expected capture time in $\mathcal{R}_\mathcal{P}$, and the width of the first 6 columns is the same as the expected capture time in $\mathcal{R}_\mathcal{S}$. This figure is not symmetric and there is 0.3 divergence at $(\mathcal{S}_6, \mathcal{P}_2)$ because at that point the attacker would take the $(4, 1)$ edge in both routing matrices, but full divergence at $(\mathcal{S}_2, \mathcal{P}_6)$ because by time 6 the attacker has captured the source in $\mathcal{P}$, but at time 2 the attacker is at node 9 in $\mathcal{S}$. While the perturbation works well for when the attacker starts at node 5, it does not increase the time to capture when the attacker starts at node 9. This is a weakness of this example perturbation as the algorithm that performs the perturbation focuses on a single starting location. Other SLP techniques may be able to perturb a wider range of nodes sufficiently.

Finally, the expected capture times shown in Table III can be calculated using Equation (11) and the expected capture probabilities are shown in Figure 9 which is calculated using Equation (12). Here the capture probabilities for SLP are strictly better than Protectionless. The results for when the attacker starts at the sink (node 5) in the two examples is shown in bold. When the attacker starts at nodes 5 and 6 it will experience a longer time before capturing the source in the SLP-aware protocol compared to the protectionless protocol. If the attacker starts at nodes 2 or 3 in $\mathcal{R}_\mathcal{S}$ it would never reach the source as there are no transitions to either of those nodes that an attacker could use to follow a message. This is also shown by Figure 9b as no matter the safety period, the capture probability is 0.

Attacker starts at i	1	2	3	4	**5**	6	7	8	9
$\mathbb{E}(\mathbb{A}_{\mathcal{P}t} = 1 \mid \mathbb{A}_{\mathcal{P}0} = i)$	0	1	2	1	**2**	3	2	3	4
$\mathbb{E}(\mathbb{A}_{\mathcal{S}t} = 1 \mid \mathbb{A}_{\mathcal{S}0} = i)$	0	∞	∞	1	**6**	5	2	3	4

Table III: Expected capture time t of the source at node 1 for $\mathcal{R}_\mathcal{P}$ and $\mathcal{R}_\mathcal{S}$ with different starting locations for $\mathbb{A}$.

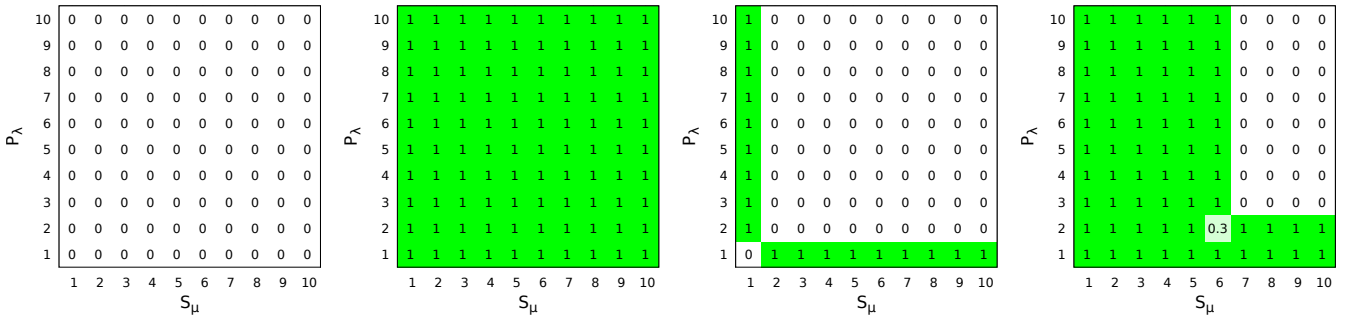
B. DynamicSPR *Case Study*

Dynamic and *DynamicSPR* have an interesting structure when considered in terms of competing paths because rather than perturbing the protectionless routing matrix additional routing matrices are added. So, a sequence of routing matrices is used to model the routing protocol for *Dynamic* and *DynamicSPR*. The first routing matrix will represent flooding $\langle$normal$\rangle$ messages from the source node. Depending on the approach there will either be one, two or randomly one or two entries in the sequence of routing matrices that represent the flood of $\langle$fake$\rangle$ messages. This pattern will repeat with the routing matrices for $\langle$fake$\rangle$ messages changing as the fake sources change location.

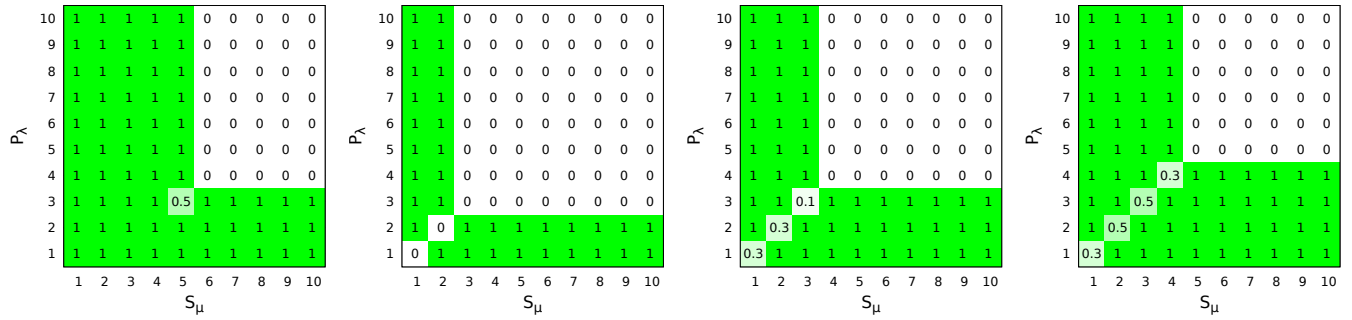
Equations (27) and (28) respectively show example sequences of routing matrices where 1 and 2 $\langle$fake$\rangle$ messages are sent per source period (`Fixed1` and `Fixed2`). While these sequences of routing matrices are infinite in length, only the finite number of entries before the safety period would be considered in the analysis. This would be the first $2P_{safety}$ entries for `Fixed1` and $3P_{safety}$ for `Fixed2`. The reason for this is that 2 routing matrices would occur within a source period for `Fixed1` and 3 would occur for `Fixed2`.

$$\mathcal{R}_1 = [\mathcal{R}_\mathcal{P}, \mathcal{R}_{\mathcal{F}_1}, \mathcal{R}_\mathcal{P}, \mathcal{R}_{\mathcal{F}_2}, \dots] \quad (27)$$

$$\mathcal{R}_2 = [\mathcal{R}_\mathcal{P}, \mathcal{R}_{\mathcal{F}_1}, \mathcal{R}_{\mathcal{F}_1}, \mathcal{R}_\mathcal{P}, \mathcal{R}_{\mathcal{F}_2}, \mathcal{R}_{\mathcal{F}_2}, \dots] \quad (28)$$

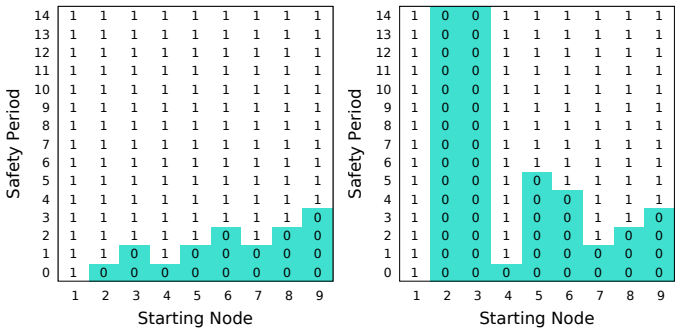


(a) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 1$ (b) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 2$ and 3 (c) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 4$ (d) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 5$



(e) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 6$ (f) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 7$ (g) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 8$ (h) $\mathbb{A}_{\mathcal{P}0} = \mathbb{A}_{\mathcal{S}0} = 9$

Figure 8: Jensen–Shannon divergence between $\mathcal{P}_\lambda$ and $\mathcal{S}_\mu$ at different times for different starting locations



(a) Protectionless (Figure 5) (b) SLP (Figure 6)

Figure 9: Capture probabilities when varying start location and safety period

$D_{\mathcal{N}}^{P_{safety}}$	1 2	1 4	2 3	2 5	3 6	4 5	4 7	5 6	5 8	6 9	7 8	8 9
$D_{\mathcal{N}}^{P_{safety}}$	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
$D_{\mathcal{F}_1}^{P_{safety}}$	✗	✗	✗	✗	✗	✗	✓	✗	✗	✓	✗	✓
$D_{\mathcal{F}_2}^{P_{safety}}$	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗

Table IV: Are the transitions from $D_{\mathcal{P}}$ in $D_{\mathcal{P}}^{P_{safety}}$, $D_{\mathcal{F}_1}^{P_{safety}}$ and $D_{\mathcal{F}_2}^{P_{safety}}$?

In terms of competing paths, an advantage of *Dynamic* and *DynamicSPR* is that the two fake routing matrices compete at all nodes that an attacker is likely to be located at when considering the protectionless routing protocol. This means that there is a large number of opportunities for the attacker to be pulled away from the real source.

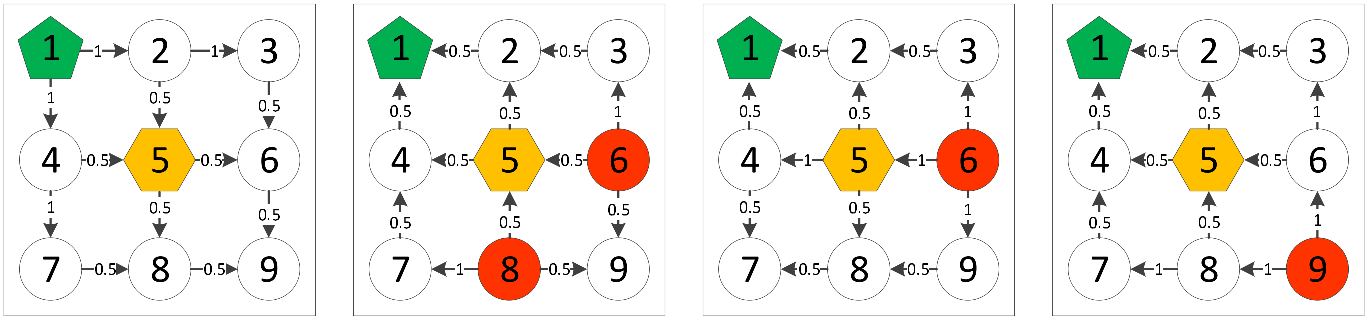
$$D_{\mathcal{F}_1}^{P_{safety}} = D_{\mathcal{F}_1} = \{(2,1),(3,2),(4,1),(4,7),(5,2),(5,4),(6,3),(6,5),(6,9),(8,5),(8,7),(8,9)\} \quad (29)$$

$$D_{\mathcal{F}_2}^{P_{safety}} = D_{\mathcal{F}_2} = \{(2,1),(3,2),(4,1),(5,2),(5,4),(6,3),(6,5),(7,4),(8,5),(8,7),(9,6),(9,8)\} \quad (30)$$

$$IL(D_{\mathcal{P}}, D_{\mathcal{N}}) = \frac{0}{12} \quad (31)$$

$$IL(D_{\mathcal{P}}, D_{\mathcal{F}_1}) = \frac{9}{12} \quad (32) \qquad IL(D_{\mathcal{P}}, D_{\mathcal{F}_2}) = \frac{12}{12} \quad (33)$$

The information loss of this sequence of routing protocols is calculated in Equations 31 to 33. In this example the safety period is 4 because the attacker would capture the source within 2 moves under *Protectionless* flooding $\mathcal{R}_{\mathcal{P}}$ and the safety factor is set to 2. The set of transitions (shown in Equation (23)) remains the same when a safety period of 4 is used. As flooding is used as the base routing protocol it leaks maximal information to the attacker when ⟨normal⟩ messages are sent. This is because this matrix is equal to the *Protectionless* routing matrix. However, the first and second ⟨fake⟩ message routing matrices provide high information loss of 75% and 100% respectively, as very few paths are shared with *Protectionless* flooding.



(a) Flooding (b) *Dynamic* 1 (c) *DynamicSPR* 1 (d) *Dynamic* 2

Figure 10: The *Protectionless* and two *Dynamic* and *DynamicSPR* routing matrices. With fake sources shown in red.

VIII. Discussion

A. Maximal Information Loss

There are different ways to obtain the relevant proper perturbed paths. For example, a path with no loops might be obtained by one method while another path with loops might also be considered. Since the notion of paths captures source-converging paths, it means that Equation (13) will never be a maximum as $D_{\mathcal{S}}$ will contain some elements of $D_{\mathcal{P}}$. However, minimising the number of common elements will result in high information loss. Specifically, it is better to introduce loops in (the non-overlapping elements of) $D_{\mathcal{S}}$ than in $D_{\mathcal{P}}$ as this will reduce the number of common elements.

B. Selecting Proper Junction Points

In this work a heuristic was proposed to select proper junction points to generate the set of properly perturbed paths. This heuristic does not specify how to choose which junction point should be used to generate the perturbed path. This is because there may be several junction points that could potentially be used to perturb the path, and some technique will need to be used to select them. Determining an optimal selection of junction nodes is likely to be a difficult problem to solve as each junction point could be used by multiple paths and produce multiple properly perturbed paths. The perturbed paths generated would need to ensure that they do not compete in a way that would lead the attacker towards the source.

C. Unreliable Links

In this model, links have been assumed to be bidirectional and lossless, such that the sum of each column that involves nodes that receive messages adds to 1. However, when links become unidirectional or lossy (e.g., due to message collisions), the sum may be less than 1. This also means that the domain $D_{\mathcal{S}}$ may differ when links are bidirectional. In this case, some of the techniques proposed will have to be adapted to specifically account for unidirectional links. On the other hand, if the unidirectional nature of links is transient the current framework can still work if nodes are made to perform retransmissions (at the link-layer level). However, this technique will not work if message collisions occur. Most often, sensor nodes are not equipped with collision detectors and it is entirely possible that the matrix $\mathcal{R}_{\mathcal{P}}$ is different to the one assumed, as a node j may receive a message from node i first (in practice) rather than from node k (as specified by $[\mathcal{R}_{\mathcal{P}}]_{kj} = 1$ and $[\mathcal{R}_{\mathcal{P}}]_{ij} = 0$).

D. Evaluating Performance

In this paper we have developed two ways (information loss and divergence) via which routing matrices can be compared. These techniques are appropriate for investigating these abstract representations of SLP-aware routing protocols. However, what is unchanged is that real-world performance will still need to be evaluated using a deployment on appropriate hardware and in a realistic environment. This analysis cannot assess the performance that an implementation would produce along metrics such as energy usage, latency, and others, however, it provides the capability to evaluate performance abstractly earlier in the design of SLP routing protocols.

E. Multiple Adversaries

In this work we have focused on modelling the location of a single adversary that starts at the sink. This location is used because it is the only location in the network that is guaranteed to receive a message. There is the potential for more powerful local adversaries to exist, for example, when there are multiple adversaries. However, when there are multiple adversaries it no longer makes sense for them all to start at the sink as they would all take the same actions. This means that the adversaries need to start distributed throughout the network, which can be modelled via the probability distribution over the initial start location $\Pr(\mathbb{A}_{\mathcal{X}0} = n)$.

This work could be adjusted to analyse the case of multiple adversaries. Assuming adversaries take independent actions, Equation (6) would need to account for the probability distribution of adversary start locations. Additionally, metrics such as the expected time to capture in Equation (11) and capture probability in Equation (12) could be updated for multiple adversaries. Finally, Algorithms 3 and 4 would need to consider multiple adversaries. A downside is that the transformation process is specific to the distribution of adversary start positions, so the produced routing matrix would be specific to the distribution over adversary start position as is the case with our assumption that the adversary starts at the sink.

In this work these algorithms have generated an SLP routing matrix that performs the same or better irrespective of the adversary's start location (see Figure 9). This means that, without changes, the generated routing matrix would not perform worse if multiple independent adversaries exist in the network.

IX. Conclusion

In this paper, the SLP problem in WSNs has been addressed from a conditional entropy and divergence viewpoint. One major advantage of using such an approach is that it allows specific protocols to be abstracted away, focusing instead on the information leaked by the network or gained by attacker. While several other works have focused on analysing specific routing protocols or privacy metrics, this approach focused on understanding the basis of routing transformations to maximise the routing divergence. The framework is novel in that it allows the SLP-aware routing matrix to be configured in different ways, to give rise to potentially different SLP-aware routing protocols. Overall, the technique is useful as it provides a way to reason about the performance of arbitrary SLP-aware routing protocols (as a routing matrix) against a local attacker, which was previously lacking in the existing work. In future work, this approach could be applied to novel context privacy problems in different domains.

Data Statement

Code for Algorithms 2 to 4 and Figures 7 to 9 can be found at https://github.com/MBradbury/slp-divergence.

Acknowledgment

This research was supported in part by the Engineering and Physical Sciences Research Council (EPSRC) [DTG grant EP/M506679/1] and Security Lancaster.

References

[1] M. Bradbury and A. Jhumka, "Understanding Source Location Privacy Protocols in Sensor Networks via Perturbation of Time Series," in *IEEE INFOCOM*, Atlanta, GA, USA, 01–04 May 2017, pp. 1611–1619.

[2] M. Bradbury, "Near Optimal Routing Protocols for Source Location Privacy in Wireless Sensor Networks: Modelling, Design and Evaluation," Ph.D. dissertation, University of Warwick, Coventry, UK, May 2018. [Online]. Available: http://wrap.warwick.ac.uk/115772

[3] 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*, ser. SASN '04. Washington DC, USA: ACM, 25 October 2004, pp. 88–93.

[4] M. Conti, J. Willemsen, and B. Crispo, "Providing Source Location Privacy in Wireless Sensor Networks: A Survey," *IEEE Communications Surveys and Tutorials*, vol. 15, no. 3, pp. 1238–1280, 2013.

[5] R. Rios, J. Lopez, and J. Cuellar, *Foundations of Security Analysis and Design VII: FOSAD 2012/2013 Tutorial Lectures*. Cham: Springer International Publishing, 2014, ch. Location Privacy in WSNs: Solutions, Challenges, and Future Trends, pp. 244–282.

[6] Y. Li, J. Ren, and J. Wu, "Quantitative Measurement and Design of Source-Location Privacy Schemes for Wireless Sensor Networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 23, no. 7, pp. 1302–1311, Jul. 2012.

[7] A. A. Nezhad, A. Miri, and D. Makrakis, "Location privacy and anonymity preserving routing for wireless sensor networks," *Computer Networks*, vol. 52, no. 18, pp. 3433–3452, Dec. 2008.

[8] S. Armenia, G. Morabito, and S. Palazzo, "Analysis of Location Privacy/Energy Efficiency Tradeoffs in Wireless Sensor Networks," in *Proceedings of the 6th International IFIP-TC6 Conference on Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet*, Atlanta, GA, USA, May 2007, pp. 215–226.

[9] H. Park, S. Song, B. Y. Choi, and C. T. Huang, "PASSAGES: Preserving Anonymity of Sources and Sinks against Global Eavesdroppers," in *2013 Proceedings IEEE INFOCOM*, Turin, Italy, 14–19 April 2013, pp. 210–214.

[10] A. Proaño, L. Lazos, and M. Krunz, "Traffic Decorrelation Techniques for Countering a Global Eavesdropper in WSNs," *IEEE Transactions on Mobile Computing*, vol. 16, no. 3, pp. 857–871, Mar. 2017.

[11] K. Mehta, D. Liu, and M. Wright, "Protecting Location Privacy in Sensor Networks against a Global Eavesdropper," *IEEE Transactions on Mobile Computing*, vol. 11, no. 2, pp. 320–336, Feb. 2012.

[12] Y. Yang, M. Shao, S. Zhu, and G. Cao, "Towards Statistically Strong Source Anonymity for Sensor Networks," *ACM Transactions on Sensor Networks*, vol. 9, no. 3, pp. 34:1–34:23, Jun. 2013.

[13] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk, "Enhancing Source-Location Privacy in Sensor Network Routing," in *25th IEEE International Conference on Distributed Computing Systems*. Columbus, OH, USA: IEEE, 6–10 June 2005, pp. 599–608.

[14] C. Dwork, F. McSherry, K. Nissim, and A. Smith, "Calibrating Noise to Sensitivity in Private Data Analysis," in *Theory of Cryptography*, S. Halevi and T. Rabin, Eds. Berlin, Heidelberg: Springer, 2006, pp. 265–284.

[15] E. Bertino, D. Lin, and W. Jiang, *A Survey of Quantification of Privacy Preserving Data Mining Algorithms*. Springer US, 2008, ch. 8, pp. 183–205.

[16] M. Bradbury, A. Jhumka, and M. Leeke, "Hybrid Online Protocols for Source Location Privacy in Wireless Sensor Networks," *Journal of Parallel and Distributed Computing*, vol. 115, pp. 67–81, May 2018.

[17] P. Spachos, D. Toumpakaris, and D. Hatzinakos, "Angle-Based Dynamic Routing Scheme for Source Location Privacy in Wireless Sensor Networks," in *79th Vehicular Technology Conference (VTC Spring)*. Seoul, South Korea: IEEE, 18–21 May 2014, pp. 1–5.

[18] R. Manjula and R. Datta, "A novel source location privacy preservation technique to achieve enhanced privacy and network lifetime in WSNs," *Pervasive and Mobile Computing*, vol. 44, pp. 58–73, Feb. 2018.

[19] L. Yao, L. Kang, F. Deng, J. Deng, and G. Wu, "Protecting source–location privacy based on multirings in wireless sensor networks," *Concurrency and Computation: Practice and Experience*, vol. 27, no. 15, pp. 3863–3876, Jun. 2015.

[20] G. Han, M. Xu, Y. He, J. Jiang, J. A. Ansere, and W. Zhang, "A dynamic ring-based routing scheme for

source location privacy in wireless sensor networks," *Information Sciences*, vol. 504, pp. 308–323, Dec. 2019.

[21] Z. Hong, R. Wang, S. Ji, and R. Beyah, "Attacker Location Evaluation-based Fake Source Scheduling for Source Location Privacy in Cyber-Physical Systems," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 5, pp. 1337–1350, May 2018.

[22] Y. He, G. Han, M. Xu, and M. Martínez-García, "A Pseudo-Packet Scheduling Algorithm for Protecting Source Location Privacy in the Internet of Things," *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9999–10 009, Jun. 2021.

[23] M. Bradbury, A. Jhumka, and C. Maple, "A Spatial Source Location Privacy-Aware Duty Cycle for Internet of Things Sensor Networks," *ACM Transactions on Internet of Things*, vol. 2, no. 1, pp. 1–32, Feb. 2021.

[24] J. Long, M. Dong, K. Ota, and A. Liu, "Achieving Source Location Privacy and Network Lifetime Maximization Through Tree-Based Diversionary Routing in Wireless Sensor Networks," *IEEE Access*, vol. 2, pp. 633–651, Jun. 2014.

[25] M. Dong, K. Ota, and A. Liu, "Preserving Source-Location Privacy through Redundant Fog Loop for Wireless Sensor Networks," in *13th IEEE International Conference on Dependable, Autonomic and Secure Computing (DASC)*. Liverpool, UK: IEEE, 26–28 October 2015, pp. 1835–1842.

[26] G. Han, X. Miao, H. Wang, M. Guizani, and W. Zhang, "CPSLP: A Cloud-Based Scheme for Protecting Source-Location Privacy in Wireless Sensor Networks Using Multi-Sinks," *IEEE Transactions on Vehicular Technology*, pp. 2739–2750, Mar. 2019.

[27] M. Raja, T. Koduru, and R. Datta, "Protecting Source Location Privacy in IoT Enabled Wireless Sensor Networks: the Case of Multiple Assets," *IEEE Internet of Things Journal*, vol. 9, no. 13, pp. 10 807–10 820, Jul. 2021.

[28] Q. Zhou, X. Qin, and X. Xie, "Hiding Contextual Information for Defending a Global Attacker," *IEEE Access*, vol. 6, pp. 51 735–51 747, Sep. 2018.

[29] R. Durrett, *Probability: Theory and Examples*, 5th ed., ser. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 2019, ch. Markov Chains, pp. 232–285.

[30] M. Bradbury and A. Jhumka, "A Near-Optimal Source Location Privacy Scheme for Wireless Sensor Networks," in *16th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*. Sydney, NSW, Australia: IEEE, 01–04 August 2017, pp. 409–416.

[31] S. Kullback and R. A. Leibler, "On Information and Sufficiency," *The Annals of Mathematical Statistics*, vol. 22, no. 1, pp. 79–86, Mar. 1951.

[32] J. Lin, "Divergence measures based on the Shannon entropy," *IEEE Transactions on Information Theory*, vol. 37, no. 1, pp. 145–151, Jan. 1991.

[33] D. Agrawal and C. C. Aggarwal, "On the Design and Quantification of Privacy Preserving Data Mining Algorithms," in *Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, ser. PODS '01. Santa Barbara, California, USA: ACM, May 2001, pp. 247–255.

[34] E. Bertino, I. N. Fovino, and L. P. Provenza, "A Framework for Evaluating Privacy Preserving Data Mining Algorithms," *Data Mining and Knowledge Discovery*, vol. 11, no. 2, pp. 121–154, Sep. 2005.

[35] A. Jhumka and M. Bradbury, "Deconstructing Source Location Privacy-aware Routing Protocols," in *Proceedings of the Symposium on Applied Computing*, ser. SAC'17, Marrakech, Morocco, 03–07 April 2017, pp. 431–436.