R. Rios, J. Lopez and J. Cuellar, "Location Privacy in WSNs: Solutions, Challenges, and Future Trends", *Foundations of Security Analysis and Design VII*, vol. 8604, pp. 244-282, 2014.
http://doi.org/10.1007/978-3-319-10082-1_9
NICS Lab. Publications: https://www.nics.uma.es/publications

Location Privacy in WSNs: Solutions, Challenges, and Future Trends

Ruben Rios[1], Javier Lopez[1], and Jorge Cuellar[2]

[1] Network, Information and Computer Security (NICS) Lab, Universidad of Málaga, Spain
[2] Siemens AG, Munich, Germany
{jlm,ruben}@lcc.uma.es
jorge.cuellar@siemens.com

Abstract. Privacy preservation is gaining popularity in Wireless Sensor Network (WSNs) due to its adoption in everyday scenarios. There are a number of research papers in this area many of which concentrate on the location privacy problem. In this paper we review and categorise these solutions based on the information available to the adversary and his capabilities. But first we analyse whether traditional anonymous communication systems conform to the original requirements of location privacy in sensor networks. Finally, we present and discuss a number of challenges and future trends that demand further attention from the research community.

Keywords: Wireless sensor networks, location privacy, traffic analysis, survey

1 Introduction

The miniaturisation of electro-mechanical systems has led to the creation of tiny, inexpensive computers capable of feeling their environment in the same way as humans experience the world through our senses. These matchbox-sized computers are called sensor nodes and they can cooperate and communicate wirelessly with other nodes nearby forming a wireless sensor network (WSN). The data collected by the sensor nodes are transmitted to a powerful device called the base station or data sink, which serves as an interface to the network.

These systems have been successfully applied to numerous application scenarios where sensor nodes are unobtrusively embedded into systems for monitoring, tracking and surveillance operations [13]. However, sensor nodes are highly vulnerable to a number of threats and attacks [45] due to their hardware limitations, which may limit their applicability to scenarios where security and privacy are essential properties. Particularly sensitive scenarios are those involving individuals, businesses and relevant assets.

A first line of defence against attacks is to protect the data traversing the network from modifications and eavesdropping but even if secure confidentiality and integrity mechanisms are in place, an adversary can attack the network

in another way. By silently observing and analysing the communications, the adversary can obtain contextual information associated with the measuring and transmission of data [33]. These metadata are inherently more difficult to protect than the data contained in the packets' payload. Indeed, the mere presence of messages may reveal sensitive information related to the application scenario. For example, the transmission of messages by a sensor node used for monitoring the structural health of a fuel pipeline is an indicator of internal corrosion.

A noteworthy piece of contextual information that may be leaked to the attacker is the location of relevant nodes in the network. The location of data sources reveals the area where special phenomena are being observed. These phenomena may be related to individuals, endangered animals, valuable cargo, etc., and as a result, the adversary obtains the location of those entities and goods. On the other hand, the location of the base station is relevant for several reasons. The base station is the most critical device in the network and if the adversary is able reach it, he may be able take control of the network or even render it completely useless by destroying it. Besides its importance for the survivability of the network, the location of the base station is strategically significant because it is most likely housed in a highly-sensitive facility. In a scenario where a WSN is deployed to monitor the behaviour of whales in the middle of the ocean, finding the base station leads to the ship where the biologists are analysing the results.

Location privacy schemes can be categorised following two main criteria, which are (a) what information is available to the adversary, and (b) what are the capabilities of the adversary to be countered. There are basically two items of interest which may help the adversary to locate targets, namely the identities of the nodes and the traffic pattern. Packet headers contain the identifiers of the source and destination of a transaction, therefore obscuring this information is the first step in achieving location privacy. Although these data are effectively protected, the attacker can still obtain location information by analysing the traffic generated by the network. The strategy of the adversary is determined by his goal and capabilities. The literature usually considers an external and passive attacker with either local or global eavesdropping capabilities. Occasionally, the attacker is also capable of compromising a small portion of the sensor nodes, thus becoming an internal adversary. As a result, we propose a taxonomy of solutions (see Fig. 1) that will guide the exposition of subsequent sections.

The rest of this work is organised as follows. Prior to the analysis of location privacy solutions in WSNs, Section 2 studies whether traditional anonymous communication systems devised for computer networks can adjust to the specific requirements and adversaries considered in sensor networks. Then, Section 3 examines two approaches to node identity protection based on the creation and use of pseudonyms. Section 4 dives into source-location privacy solutions, paying attention both to external and internal adversaries. Similarly, Section 5 analyses solutions for the protection of the base station against local and global observers. Finally, Section 6 presents and discusses a number of open issues and future areas of research, and Section 7 concludes this paper.

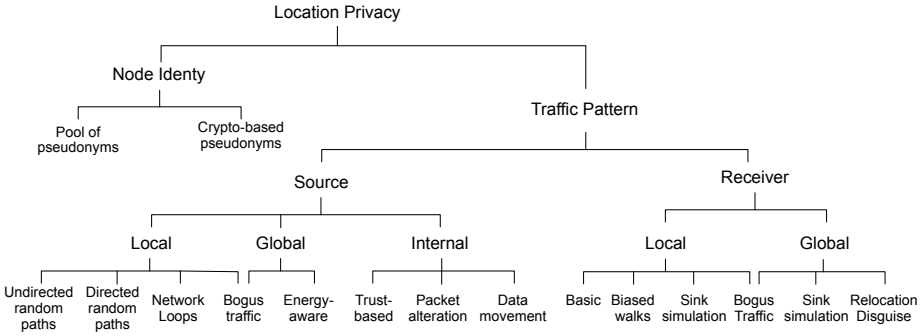


Fig. 1. Taxonomy of Location Privacy Solutions in WSNs

2 Computer-based Anonymity Systems

Anonymous communication systems for computer networks were originally devised to hinder traffic analysis attacks. Therefore, it appears feasible to use these solutions to protect location privacy in sensor networks as this problem is caused by the peculiar traffic pattern of this networks. First, we need to analyse the anonymity requirements in both scenarios. After that, we select several renowned anonymous communication systems to study whether these can be implemented in resource-constrained sensor nodes and also whether the deployment of these solutions limit in any way, the usability or functionality of the network.

2.1 Anonymity Requierements

There are several anonymity properties that may help entities to preserve their privacy when communicating with other entities [34]. These properties provide different levels of anonymity ranging from avoiding the identification of a given subject within a set of other subjects to the impossibility of proving the participation of a specific subject in a given communication. The most usual property implemented by traditional anonymous communication systems is the *unlinkability* of senders and receivers, which is intended to prevent an adversary from identifying which entities are communicating with whom, since this allows him to learn the habits and interests of a specific individual. However, this property is not necessary in WSNs since an external adversary already knows that all sensor nodes communicate with the base station.

Some other solutions focus on providing sender *anonymity* with respect to the receiver. The goal is to prevent ill-intentioned service providers from collecting data from users for the purpose of tracking and profiling. In WSNs, the enforcement of this property is not only unnecessary but also detrimental to the normal operation of the network. The reason is that the base station needs to know the identities of the nodes generating data messages in order to faithfully identify the location of relevant events in the field. Nevertheless, source

anonymity is also suitable for systems where the communications traverse some potentially malicious (i.e., honest but curious) nodes interested in learning the actual data sender. This type of anonymity is important in WSNs where some nodes are compromised by the attacker and try to obtain the source node identifier. Therefore, source anonymity is only necessary in certain circumstances.

While *unobservability* is a very strong notion of privacy and is only rarely necessary in computer networks, it becomes the most natural way of protecting location privacy in WSNs. It is imperative to hide the existence of the nodes reporting on or receiving event data. If the adversary cannot sufficiently detect the presence of data messages in the network, he will be unable to determine the location of the nodes taking part in the communication. Consequently, if the attacker is not able to ascertain the existence of messages, he will not be able to determine who is the sender or recipient of that message by simply performing traffic analysis attacks.

In general, we can state that some anonymity properties are unsuitable or unnecessary for protecting location privacy in WSNs, in fact, they might even be counterproductive in particular cases.

2.2 Overhead Analysis

The aforementioned properties have been satisfied by anonymous communication schemes through different techniques, which incur notorious computational and communication overhead to the system. These techniques range from simple identity renaming to more complex operations such as layered encryption, fake traffic injection, and tightly-synchronised broadcast communications. Moreover, anonymous communications systems can be categorised based on their architecture as centralised or decentralised, depending on whether the users are members of the system that collaborate in the anonymisation process or not. Here we have selected three solutions that not only cover a wide range of techniques and features but also pursue different anonymity properties and architectures.

Mix-nets [6] are high-latency centralised systems composed of a set of store-and-forward devices (i.e., mixes) that prevent the correlation between incoming and outgoing messages. Whenever a user wants to communicate with another user, he selects a series of mixes and recursively adds a layer of (public-key) encryption to the message for each mix in reverse order. In this way, each mix device only knows its predecessor and successor in the path. This scheme is extremely effective for ensuring unlinkability in delay-tolerant applications but its is not suitable for WSNs, where real-time monitoring capabilities are usually necessary. Moreover, there are some other limitations with respect to the memory and computational requirements imposed by the scheme. Data sources are required to perform $n+1$ public-key operations per data packet, being n the path length, but they also need to have a complete knowledge of the topology of the mix-net in order to apply the layers of encryption in the right order. Additionally, each intermediate node is required to not only perform one decryption per packet but also to store a number of packets for a long period of time. Finally, a centralised scheme cannot protect from global adversaries and, for the particular

case of mix-nets, it cannot protect itself from local adversaries either because the attacker can eventually reach the edge of the mix-net and from there locate the data sources.

Crowds [38] is a decentralised scheme where a set of users collaborate to issue requests to servers in order to provide anonymity to its members. After joining the crowd, any of its members can initiate requests to different servers, which are delivered by a random member. Whenever a crowd member wants to send a message, it chooses a random member, possibly itself, to act as an intermediary. The recipient decides, based on some biased probability, whether to forward the data to another member or to finally submit it to the destination. Subsequent requests from the same data source and same destination follow the same path. Messages are re-encrypted and the sender identity is replaced at every hop. Although this model is far less complex than the previous one from a computational point of view, it still has high memory demands. Each node must hold $n-1$ shared keys (i.e., one key per crowd member) and a translation table containing all the paths that have the node as an intermediary, as paths are static. This is, indeed, an important drawback to its application in WSNs because static paths can be easily traced back by local adversaries. Even though this is a decentralised solution, global adversaries might be able to identify data sources since new traffic is only generated in the presence of real events, and also, the base station as all the traffic is addressed to it. However, this scheme does provide some means of protection against Internal adversaries due to the identity renaming mechanism.

DC-nets [7] is a decentralised solution based on simple calculations that allows a group of users to share information while hiding the actual sender (and recipient) of messages even from other protocol participants. To this end, each member shares bitwise keys with any other participant and all members simultaneously broadcast the result of the bitwise sum of their secrets. The key point is that if a participant has something to say he inverts this result before broadcasting it. Each secret is used twice so the final result must be zero if no one has inverted his result. Since the initial shared bits are secret, there is no way to determine the actual sender. Although the original protocol considers the transmission of a single data bit, the DC scheme can be easily extended to transmit string messages by sharing random numbers instead of random bits. The application of the DC-nets model in WSNs has several impediments. First, the need for a tight and reliable broadcast channel that covers all sensor nodes and the base station. Second, the high memory overhead required to store one-time secrets for multiple protocol rounds and the high waste of bandwidth and energy due to the continuous rounds even when no participant is willing to transmit. Another substantial problem has to do with simultaneous communications. The scheme does not support multiple transmissions at the same time, which would highly constrain the usability and nature of sensor networks.

Table 1 presents a summary of the this analysis. It indicates that even though some solutions are sufficiently lightweight to run in sensor nodes, the true weak point is that the solutions do not fit the requirements and the adversarial models

		Adversary		
	Limitations	Global	Local	Internal
Mix-nets	high	×	×	✓
Crowds	low	×	×	≈
DC-nets	high	✓	✓	✓

Table 1. Suitability of some Anonymous Communication Systems

under consideration. Similarly, another group of solutions are suitable for the protection of location privacy in WSNs but they are rather expensive or they present important limitations. As a result, new tailored solutions have been designed specifically for WSNs.

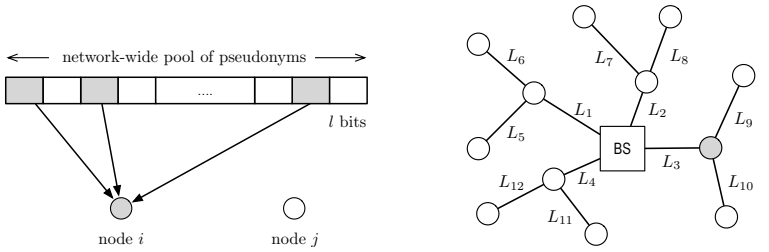
3 Node Anonymity

Packet headers consist of various data fields containing, among other things, the identifiers of the data sender and the destination. These data are sent in clear text to enable intermediate nodes to perform routing tasks. Thus, after a sufficient number of observations, an attacker can elaborate a map of the network relating node identifiers to locations in the field. Being in possession of such a network map, the attacker may simply wait next to the base station for incoming messages and easily obtain the location where events occur.

Several techniques have been proposed to provide node anonymity, most of which are based on the use of dynamic pseudonyms. Some authors have approached the management of pseudonyms by means of pools of pseudonyms while others have turn to cryptographic mechanisms for the same purpose. Note that most of the solutions fall into the second category since the use of cryptographic techniques for the creation of pseudonyms have several benefits over the use of network pools. Next, we review these solutions in detail.

3.1 Pool of pseudonyms

Misra and Xue [26] were the first authors to provide a set of solutions for node identity protection. The Simple Anonymity Scheme (SAS) is based on a network-wide pool of pseudonyms which the base station divides into subranges of l bits and provides each node with a random set of them (see Fig. 2a). Each node builds a pseudonyms table where it stores pseudonym ranges for incoming and outgoing messages for each neighbour and their corresponding secret keys. When the node wants to communicate with a specific neighbour, it selects a random value from the range of pseudonyms belonging to that node and concatenates the index of the row from where it picked the pseudonym. The recipient node checks whether the received pseudonym belongs to the incoming range corresponding to the given index and, that being the case, it uses the shared key to decrypt the message. The principal limitation to SAS is the large memory space necessary



(a) Distribution of pseudonyms in SAS (b) Label assignment in DCARPS

Fig. 2. Pool-Based Approaches

to store a sufficiently large pseudonym space, especially in densely populated networks.

Nezhad et al. [27, 28] proposed a label switching protocol as part of their DCARPS anonymous routing protocol. After each topology discovery phase, the base station obtains an updated map of the network and assigns labels (i.e., identifiers) to each and every network link, as depicted in Fig. 2b. These labels serve as pseudonyms and whenever a node has to send a packet to the base station, it uses the label assigned to the link connecting it to a neighbour that is closer to the base station. Upon the reception of the packet, the neighbour node, checks whether the label corresponds to one of its input labels. If the label is known to the node, it replaces the input label with its own output label. For example, the grey node in Fig. 2b checks whether an incoming message has either label L_9 or L_{10} and, in the case it does, it forwards the packet after changing the original label with L_3. The main drawback to this labelling solution is that labels are modified only after a topology change has been discovered, which allows the attacker to correlate labels with specific nodes, thus completely compromising anonymity.

3.2 Cryptographic pseudonyms

The second solution by Misra and Xue [26] is intended to reduce the amount of memory needed by SAS at the expense of increased computational overhead. The Cryptographic Anonymity Scheme (CAS) uses a keyed hash function to generate the pseudonyms. Before the deployment of the network, each node is assigned a pseudo-random function, a secret key and a random seed shared with the base station. After deployment, each pair of neighbours agree upon a random seed and a hash key that they store together with a sequence number. Whenever a node wants to send data to the base station, using a neighbour as intermediary, it creates a message $M = \{sID, rID, EncryptedPayload, seq\}$, where sID and rID are the pseudonyms generated after applying the keyed hash functions to the random seed and the sequence number shared with the base station and the intermediary, respectively. This scheme is more memory efficient but it imposes a computational overhead, not only to the intended recipient but also to any

neighbour receiving the packet which need to compute a keyed hash value before discovering it is not addressed to them.

The CAS scheme assume that an attacker cannot compromise the secrets shared between the nodes. To reduce the impact of secrets being compromised, Ouyang et al. [31] propose two methods based on keyed hash chains. The Hashing-based ID Randomisation (HIR) scheme, uses the result of applying a keyed hash function to the true identifier of the node as pseudonym. More precisely, each node shares pairwise keys with uplink and downlink neighbours and creates, for each link, the keyed hash identifier of the uplink node of that neighbour. After the transmission or reception of a message on a particular link, the node rehashes the value contained in the table to generate a fresh pseudonym. Additionally, packets convey another identifier used for the base station to be able to identify the original data source. This value is also an element of a hash chain keyed with a secret shared with the base station. Since hash values are assumed to be non-invertible, this solution provides backwards secrecy, but if the adversary compromises the key used by the hash functions, he can generate future pseudonyms. The second solution, Reverse HIR (RHIR), attempts to reduce this problem by creating the hash chain during the initialisation and then using the elements of the chain in reverse order. Once a pseudonym has been used, it is no longer needed and it can be deleted from the memory. In this way, the attacker cannot generate any fresh pseudonyms even if he compromises the key. The main drawback to this solution with respect to the previous one lies in the need for increased memory space to accommodate a lengthy hash chain.

Later, Jiang et al. [16] introduced the Anonymous Path Routing (APR) protocol. One of the elements of this scheme, namely the anonymous one-hop communication, introduces an enhancement that improves the resilience against secret compromise attacks compared to previous solutions. In this scheme each node creates a table to keep the uplink and downlink hidden identities of each neighbour. These identities are calculated by hashing the values of the secret keys, identities, a sequence number and a nonce shared by the nodes. The novelty of this approach is that not only the hidden identities are updated (i.e., rehashed) after each successful transmission between neighbouring nodes but also the keys shared between the nodes. The same idea has been developed by Chen et al. [9] in the Efficient Anonymous Communication (EAC) protocol. The problem with this is scheme is that, nodes exchange with their neighbours the keys and nonces they share with the base station to update the pseudonyms used for one-hop communications. This allows any node to determine whether the true source of the packet is a neighbouring node as well as to impersonate any of its neighbours.

Finally, it is important to highlight that node anonymity is only a first line of defence to preserve location privacy. An adversary can perform more sophisticated attacks to obtain location information from the analysis of traffic patterns. In the following sections we concentrate on the most important solutions that have been developed to diminish the threat of different types of adversaries.

4 Source-Location Privacy

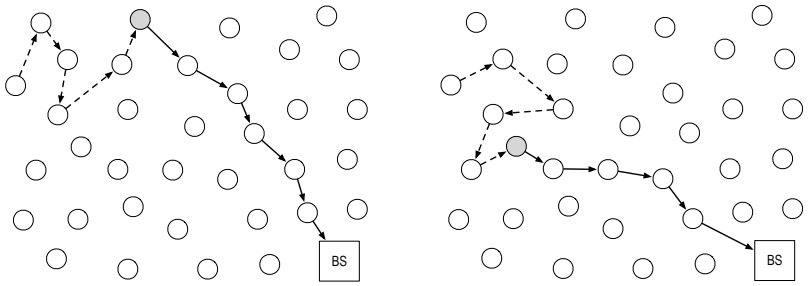
Source-location privacy refers to the ability to hide the location of data sources, which results in the protection of the physical location of the events being monitored since they may be related to individuals or valuable resources. This problem has drawn the attention of the research community and plenty of solutions have been devised for countering passive adversaries with a local or a global view of the communications, but only a few authors have concentrated on the threat of internal attackers.

4.1 Local Adversaries

A local adversary can only monitor a small portion of the network, typically the equivalent of the hearing range of an ordinary sensor node. Therefore, they must turn to moving in the field using a *traceback attack* in an attempt to reach the target by moving along the path of messages from the source to the base station in reverse order. This attack is successful because data packets tend to follow the same path over and over again. Consequently, most of the solutions to this problem are based on the randomisation of routes although some schemes also take advantage of bogus traffic to mislead the adversary. Note that some solutions may belong to more than one category.

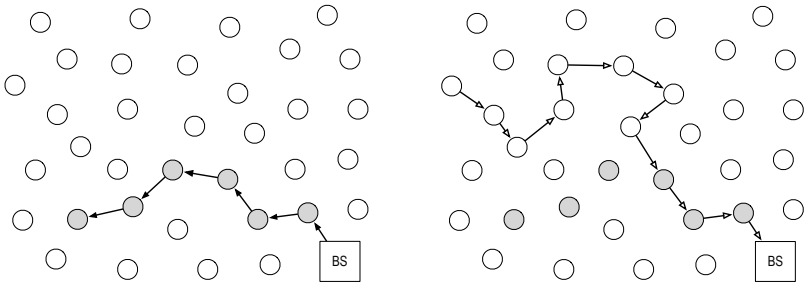
Undirected Random Paths The first solution to provide source-location privacy was devised by Ozturk et al. and is called Phantom Routing [32]. Phantom Routing proposes making each packet undergo two phases, a walking phase and a flooding phase. In the walking phase, the packet is sent on a random walk for h hops until it reaches a node, which is called the phantom source. Then, in the next phase, the phantom source initiates a baseline or probabilistic flooding, which eventually delivers the packet to the base station. This two-phase process picks random phantom sources for each new message thereby originating different paths. Later, a new version of protocol, called Phantom Single-Path Routing [17] replaced the flooding in the second phase by a single-path routing, which results in even longer safety periods due to the fact that the adversary misses some packets. Fig. 3 depicts the transmission of two messages using this solution, where dashed arrows represent the walking phase and the ordinary arrows represent the single-path phase. The grey node is the phantom source. The main limitation to Phantom Routing protocols is in the walking phase. Pure random walks tend to stay close to the source node and the definition of a larger value of h does not provide a direct improvement in the safety period, it only increases the energy waste. This problem is represented in Fig. 3, where phantom sources are within a distance of two or three hops regardless of the definition of a 5-step random walk.

Xi et al. [49] state that using pure random walks is desirable because routing decisions are independent from the source location but also impractical since the average delivery time of messages goes to infinity. The idea behind GROW is



(a) First transmission (b) Next transmission

Fig. 3. Phantom Single-Path Routing with $h = 5$

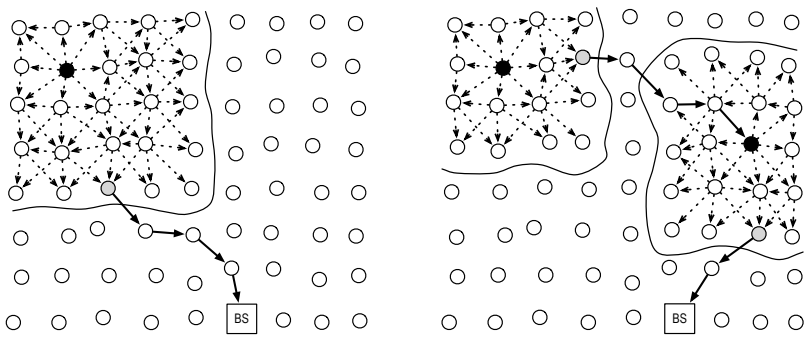


(a) Path of receptors (b) Data transmission

Fig. 4. Operation of the GROW Scheme

using two random walks as the probability of them not intersecting decreases exponentially in time. First, it creates a permanent path of receptors by transmitting a special packet on a random walk from the base station. Then, the source nodes send data packets on a greedy random walk that will eventually hit a node from the path of receptors. From there, the packet is forwarded to the base station following the established path in reverse order. This process is illustrated in Fig. 4. Despite being designed as a greedy algorithm, one of the main limitations of GROW is the substantial delivery time of the packets.

Cross-layer routing [42] was designed to further mitigate the problem of random walks staying close to the data source. This approach is basically a Phantom Routing that hides the walking phase by routing data using the beacon frames from the data link layer. Since beacons are transmitted regardless of the occurrence of events, the attacker is unable to distinguish legitimate beacons from those containing event data. At the end of the walking phase, event data reaches a pivot node that sends the data to the base station using the implemented routing protocol. The operation of the protocol is depicted in Fig. 5a, where the dotted arrows represent the beacon frames, solid arrows represent the routing phase, and the black and grey circles represent the source and the pivot node,



(a) Single cross-layer approach (b) Dual cross-layer approach

Fig. 5. Cross-Layer Routing Schemes

respectively. The main limitation to this approach lies in the tradeoff between the level of protection it can provide and the delay introduced by large beaconing areas[3]. Therefore, the larger the beaconing area is the better the protection but also the longer the delay.

An attacker may be able to reach the edge of the beaconing area and, from there, reach the data source if the network administrator turns to small values for h to boost the delivery time. A double cross-layer solution is proposed to further enhance location privacy in these circumstances. In this version of the protocol, instead of sending the data directly to the base station, the pivot node sends the data to another randomly chosen node using the routing layer. Then, this random node chooses a new pivot node and starts a second beaconing phase. Thus, the attacker cannot easily reach the edge of the beacon area to which the original data source belongs. The dual cross-layer approach is represented in Fig. 5b.

Based on the same idea of hiding the walking phase, Mahmoud and Shen propose creating a cloud of fake traffic around the data source to hinder traceback attacks [23]. During the network setup, sensor nodes choose a group of nodes at different distances to later become fake source nodes, similar to phantom sources or pivot nodes. Also, each node divides its immediate neighbours in several groups in such a way that the neighbours from the same group are in different directions. During the data transmission phase, for each message, the source node chooses one of its fake sources and sends the message to the group where there is a member which knows how to reach it. As the packet travels to the fake source, it generates fake traffic to cover the route. A node from the addressed group that does not know where the fake source is, generates a fake message and picks one of its groups at random to broadcast it. The fake message lasts for h hops, generating clouds with dynamic shapes. Compared to

[3] Beacon frames are sent out at intervals ranging from milliseconds to hundreds of seconds.

the previous scheme, this solution consumes substantially more energy but it reduces the delivery time.

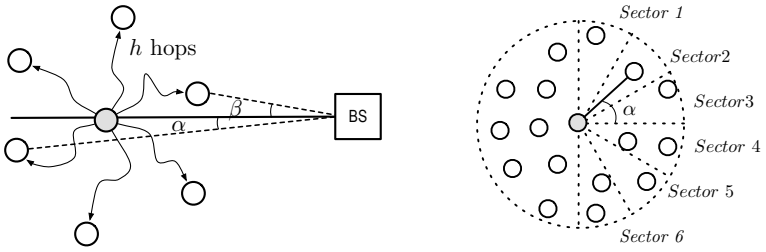
Directed Random Paths Instead of simply sending packets at random, some authors have proposed using mechanisms to guide the walking phase. The first solution to have considered this is Phantom Routing itself [32]. The authors suggest changing the pure random walk in favour of a directed random walk. To that end, each node separates its neighbours into two groups depending on whether they are in the same direction or in the opposite direction to the base station. Thus, during the walking phase, the next hop in the path is still selected uniformly at random but only from the set of nodes in the direction of the base station. By introducing this simple mechanism they prevent packets from looping in the vicinity of the source thereby increasing the level of protection.

Yao and Wen devised the Directed Random Walk (DROW) in [53]. The idea behind this solution is quite simple, any sensor node having a data packet to transmit must send it to any of its parent nodes (i.e., a node closer to the sink) with equal probability. Therefore, the level of protection is highly dependent on the connectivity of the network. In 2010, Yao alone published another paper describing the Directed Greedy Random Walk (DGRW) [52], which is basically a copy of DROW with a different name. Also, the Forward Random Walk (FRW) [8] does exactly the same thing. However, the Chen and Lou argue that this solution cannot obtain a high level of protection and it would be necessary to inject dummy messages in the network to reduce the chances of the adversary.

Interestingly, Wei-Ping et al. [48] observed that long random walks do not necessarily increase the protection unless the phantom sources are not placed close to the straight line between the data source and the sink. The reason is that if phantom sources are close to this line too often, the single paths originated by them will be very similar to each other and thus the attacker has more opportunity to overhear packets. This problem is depicted in Fig. 6a, where the curly lines represent directed random walks from the source node to the phantom sources and the dashed lines represent the single-path routing phase. To prevent this situation, in Phantom Routing with Locational Angle (PRLA) a sensor node assigns its neighbours forwarding probabilities based on their inclination angles in such a way that neighbours with larger angles will be more likely to receive messages. A major downside to this work is that it is not fully clear how the nodes obtain the inclination angles[4] of their neighbours without built-in geolocation devices or directional antennas.

Wang et al. [46] devised a solution, called the Weighted Random Stride (WRS), which is similar to PRLA in the sense that both of them make routing decisions probabilistically based on the inclination angle of its neighbours. Data paths are guided by two parameters, a forwarding angle and a stride. The forwarding angle determines the next neighbour in the path while the stride defines the number of hops for a particular forwarding angle. The node receiving a expired stride selects a new forwarding angle and starts a new stride. In practice,

[4] The authors claim that the inclination angle is calculated in terms of the hop count.



(a) Problematic selection of phantom sources

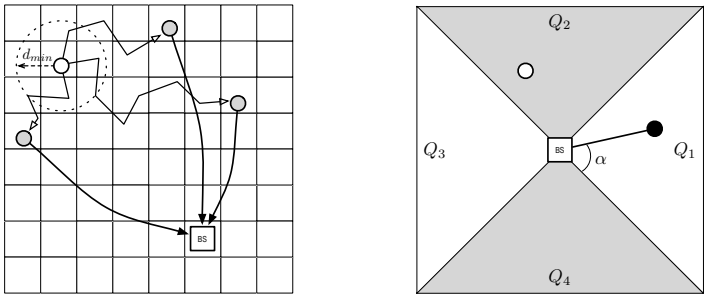
(b) Sector-based neighbour selection in WRS

Fig. 6. Angle-based privacy solutions

sensor nodes divide their neighbours into closer and further nodes and these into sectors. Sectors with larger inclination angles are prioritised. For example, in Fig. 6b, sectors 1 and 6 are more likely to be chosen than sectors 2 and 5, and sectors 3 and 4 are the least likely. The main difference between this approach and PRLA is that in WRS there are no phantom sources from where the packets are finally routed to the base station using a single-path approach.

Besides the WRS routing, Wang et al. [46] designed the Random Parallel routing, which assigns each sensor node n parallel routing paths to the base station. Messages are evenly distributed to different paths in such a way that the adversary traceback time is the same at any path. The underlying idea is that once the adversary chooses one of the paths he is forced to stay on that path. This increases the traceback time, which is now equivalent to the sum of all the parallel paths, without delaying message delivery. In a real-world setting, the generation of n truly parallel paths is a complex task, especially in large-scale sensor network deployments. Moreover, since the paths are parallel to each other, retrieving several packets from any of the paths provides a good idea of the direction to the source. This would significantly reduce the expected traceback time for the adversary.

Li et al. [19] proposed Routing through a Random selected Intermediate Node (RRIN) to the problem of selecting phantom sources close to the data source. The authors assume that the network is divided into a grid and that each node knows its relative location (i.e., cell position) as well as the grid dimensions. In this way, the source node can pick a random point in the field and send the packet to that location. The node closest to that location becomes the intermediate node. They devised two versions of RRIN. In the first version, the intermediate point is chosen uniformly at random but it is forced to be placed at least at a distance d_{min} from the source as shown in Fig. 7a. The main drawback to this scheme is that the probability of being selected as an intermediate node is proportional to the distance to the data source. Additionally, no mechanism prevents them from being picked from the proximities of the source-destination shortest path, which was one of the problems addressed by PRLA and WRS. In the second version of RRIN, any location in the network has the same probability of being selected



(a) Distance-based RRIN

(b) Quadrant-based multi-intermediate selection

Fig. 7. Routing through Random selected Intermediate Nodes

as the random intermediate point. The consequence is that some intermediate nodes will be very close to the data source thus exposing its location while some others will be extremely far resulting in energy-intensive paths.

The RRIN scheme has been extended and used in several other papers. The Sink Toroidal Routing (STaR) routing protocol [22] is also designed to improve upon the initial RRIN designs. More precisely, the goal is to reduce the energy cost associated with the selection of pure random intermediate nodes in the field. To that end, the source node picks random points within a toroidal region around the base station, which guarantees that intermediate nodes are, at most, a given distance from the destination but also not too close in order to prevent traceback attacks. The main drawback to this solution again has to do with the selection of problematic intermediate nodes not only between the source and the base station but also behind it.

In [20], Li et al. propose two schemes that use multiple random intermediate nodes instead of a single one. In the angle-based multi-intermediate node selection, the source node selects a maximum angle β to limit the location of the last intermediate node within the range $(-\beta, \beta)$. Once the maximum angle has been determined, the source node uniformly chooses a random angle θ between itself and the node with respect to the base station, such that $\theta \in (-\beta, \beta)$. Then, the data source selects the rest of the n intermediate nodes to be evenly separated between itself and the final intermediate node. In the quadrant-based multi-intermediate node selection, each sensor node divides the network into four quadrants in such a way that it is placed in the first quadrant and the base station is in the middle. The source node location is determined within the first quadrant based on a random angle α. The last intermediate node is selected to be somewhere within its adjacent quadrants, namely quadrant 2 and 4 as shown in Fig. 7b. Both extensions ensure that nodes are neither selected from behind the base station nor close to the shortest-path between the data source and the destination. However, it is not fully clear why it is necessary to use multiple intermediate nodes instead of a single intermediary.

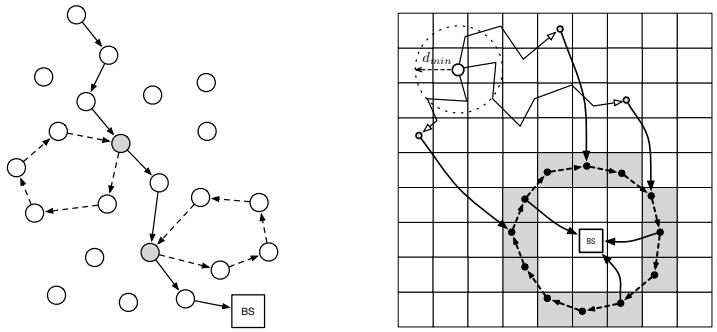
Finally, Rios and Lopez [40] realised that the message delivery delay and energy consumption incurred by existing solutions could be significantly reduced. The Context-Aware Location Privacy (CALP) scheme takes advantage of the ability of sensor nodes to perceive the presence of a mobile adversary in their vicinity in order to dynamically modify the routing paths. The routing process operates as usual but upon the detection of an adversary in the vicinity of a node the CALP mechanism is triggered. The detecting node informs its neighbours about the presence of the adversary and they modify their routing tables to circumvent the area controlled by him. Two strategies are devised depending on the way forwarding decisions are made. The strict version blocks the transmission of packets if the adversary is too close, thus avoiding the capture of packets but it might cause large delays. The second version is more permissive as it only penalises the transmission of packets within an area close to the adversary but it reduces the delay.

Network Loop Methods A completely different approach to deceive local adversaries consists of the creation of network loops. A network loop is basically a sequence of nodes that transmit messages in a cycle in order to keep the adversary away from the real direction towards the data source.

The Cyclic Entrapment Method (CEM) [30] sets traps in the form of decoy messages to distract the adversary from the true path to the data source for as long as possible. After the deployment of the network, each sensor node decides whether it will generate a network loop with a given probability. Then, the node selects two neighbouring nodes and sends a loop-creation message that travels h hops from the first to the other neighbour. All the nodes receiving this message become loop members. During the normal operation of the network, a loop remains active as long as a loop member receives a real packet. Interestingly, when CEM is used in conjunction with single-path routing (see Fig. 8a), real traffic reaches the base station in the shortest time possible without incurring extra delays. During a traceback attack, when reaching a fork in the path the adversary must decide which packet to follow. If he picks the fake message he is trapped in the loop for h hops. However, an skilled adversary might avoid loops since packets with a larger inclination angle are more likely to lead to a loop.

In the information Hiding in Distributed Environments (iHIDE) scheme [18] the sensor network consists of a set of ring nodes which are inter-connected with each other and with the base station by means of a network bus. This arrangement is similar to the one depicted in Fig. 8a but in iHIDE all sensor nodes are either bus or ring nodes. During the data transmission period, a source node that wishes to communicate data to the sink first sends the data to the next ring member in a (counter-)clockwise direction[5]. When the bus node receives the packet, it forwards it to the next bus node closer to the sink but the packet continues to loop in the same ring for a random number of hops. As the packet travels through the bus, each bus node decides, based on a given probability, to

[5] In the case that the sensor node belongs to multiple rings simultaneously it randomly selects one of them to forward the message.



(a) Cyclic Entrapment Method (b) Network Mixing Ring

Fig. 8. Network Loops Methods

forward the packet into its own ring or to directly submit it to the next bus node. The main limitation to iHIDE is that the adversary can wait until he observes that a bus node just forwards a message to the next bus node. This implies that somewhere in a previous ring there is a data source.

The Network Mixing Ring (NMR) scheme [21] creates a virtual ring of nodes surrounding the base station whose aim is to mix up real messages with fake traffic in order to mislead the adversary. This scheme consists of two phases. In the first phase, the source nodes picks an intermediate node using the RRIN approach (see Section 4.1). In the second phase, the intermediate sends the packet to the network mixing ring. Once there, the packet is relayed clockwise for a random number of hops before being finally submitted to the base station. Within the mixing ring there are a few nodes that generate vehicle messages, which are re-encrypted at every hop. These messages carry several bogus data units, which are replaced as real messages enter the ring. The whole process is depicted in Fig. 8b, where the grey cells represent the area defining the network mixing ring. A major limitation to this scheme is that ring nodes are likely to deplete their batteries soon, thus isolating the sink from the rest of the network.

To diminish the energy imbalance between ordinary sensor nodes and ring nodes, the authors propose predefining several rings and activating only one at a time according to the residual energy of their members. Additionally, they briefly discuss the possibility of having several active rings simultaneously to improve the level of protection of the data sources. This idea have been continued by Yao et al. [54]. Whenever a sensor nodes has something to transmit it picks two random rings (one closer and one farther), and an angle α between zero and π. Then, it sends out the packet to the farther ring and once there it is relayed counterclockwise until the angle is reached. From this point, the packet is sent to the closer ring and once more travels counterclockwise for an angle $\beta = \pi - \alpha$. Finally, the packet is routed directly to the base station. During this process, fake packets are injected by the nodes on contiguous rings to further complicate

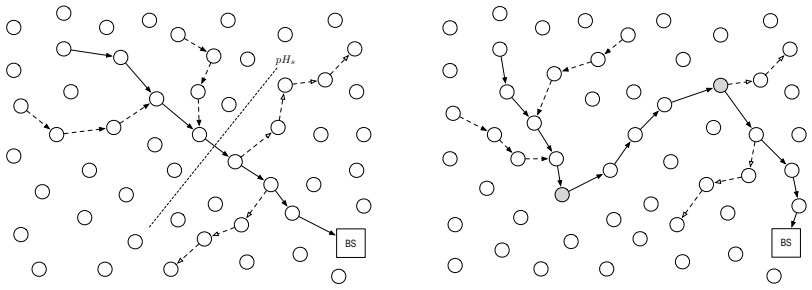
traffic analysis. Clearly, these ring-based solutions require the network to be densely populated in order to enable the creation of full rings.

Fake Data Sources The idea of using fake data sources was first suggested by Ozturk et al. [32]. They proposed two strategies, namely Short-lived and Persistent Fake Source, to simulate the presence of real events in the field by making some sensor nodes to behave as true data sources. In the first strategy, whenever a sensor node receives a real message it decides, based on a particular probability distribution, whether to generate a fake message and flood the network with it. This scheme provides a poor privacy protection since fake data sources are ephemeral. The second strategy aims to prevent this by creating persistent sources of fake messages. Each sensor node decides with a probability to become a fake data source. The efficiency of this strategy is very much dependent on the positioning of the fake data source. If fake data sources are far from a real data source it helps, otherwise it may lead the adversary to the real data source.

Chen and Lou [8] designed several solutions to protect location privacy based on the use of fake messages, namely the Bidirectional Tree (BT) scheme, the Dynamic Bidirectional Tree (DBT) scheme, and the Zigzag Bidirectional Tree (ZBT) scheme. These solutions are intended to protect both source- and receiver-location privacy simultaneously but we cover them here in full detail to avoid the duplication of contents across different sections. In the BT scheme, real messages travel along the shortest path from the source to the sink and several branches of fake messages flow into and out of the path. To that end, before the transmission of data messages, the source node sends a packet containing its own hop count H_s along the shortest path. Those nodes in the path whose distance to the sink is greater than $(1-p)H_s$, being p a network-wide parameter, will generate an input branch[6] with a given probability. Similarly, the nodes satisfying pH_s will choose whether to generate an output branch. This solution is depicted in Fig. 9a, where dashed arrows represent (input or output) fake branches. The idea behind the creation of fake branches is to misdirect the adversary from the real path but is not difficult for a skilled adversary to realise that nodes deviating from the already travelled path are fake branches.

To prevent the adversary from easily obtaining directional information, the DBT scheme suggests that when a node receives a real message it must decide the next hop uniformly at random its neighbours closer to the base station. Similar to the BT scheme, fake branches are created but in this case, input branches are generated with a given probability when the hop count is smaller than $H_s/2$, and output branches otherwise. In the ZBT real packets zigzag along three segments: from the source node to a source proxy, from there to a sink proxy, and finally to the real sink. During the data transmission phase, each node in the path generates fake branches with a given probability. In the segment from the source node to the source proxy, the fake packets flow into the path, and in the segment from the sink proxy to the sink, the packets flow out. No branches are generated in the segment connecting the source and sink proxies. The operation of the

[6] The authors do not specify how sources of fake input data are selected.



(a) Bidirectional Tree Scheme (b) Zigzag Bidirectional Tree Scheme

Fig. 9. Bidirectional Tree Schemes

ZBT scheme is depicted in Fig. 9b, where grey nodes represent the source and sink proxy nodes. This scheme presents the same limitation as the original BT scheme, that is, fake branches can be eventually discarded. Either the attacker discards a fake branch after tracing it or due to a unusual inclination angle.

Jhumka et al. [14] developed two solutions, namely fake source (FS) 1 and 2, to investigate the effectiveness of using fake data sources. Both solutions are built on top of a baseline flooding protocol. In FS1, the data source floods the network with a message containing the event data and a hop count. When this packet reaches the base station, it generates an away message containing the distance between itself and the data source, and floods the network with it. The away message is intended to reach all nodes at the same distance as the source to the sink and make them transmit a choose message. This new message is forwarded to nodes further away, which decide to forward it based on a given probability. When the hop count of the choose message reaches 0, it generates a random number and, if above a given threshold, the node becomes a fake data source. The FS2 protocol is very similar to FS1, the difference is that in FS2 all the nodes that receive a message forward it, while in FS1 the forwarding of messages is determined by a given probability. Consequently, more nodes are likely to become fake data sources in FS2 and thereby the level of protection achieved by this scheme is better at the expense of increased energy consumption.

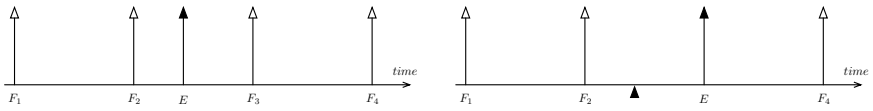
4.2 Global Adversaries

The aforementioned techniques are only effective against adversaries performing traceback attacks with a limited hearing range. Global adversaries are capable of monitoring all the traffic generated and forwarded in the network. Such adversaries can easily detect the data sources among mere intermediaries because sensor nodes are programmed to report event data to the base station as soon as it is detected.

There are two main approaches to hide the location of data sources from global adversaries, either using fake packets or introducing significant delays in the transmissions. Most solutions have concentrated on the injection of bogus

traffic and a huge research effort has been devoted to making these solutions as energy-efficient as possible.

Bogus Traffic The threat of global adversaries was first considered by Mehta et al. in [24], where they proposed the Periodic Collection scheme. This scheme hides the presence of events in the field by making every node transmit fake messages at regular intervals. However, it is not as simple as sending fake messages at a constant rate because the occurrence of an event message would change the transmission pattern, as shown in Fig. 10a. This figure depicts a timeline where the transmissions of real and fake packets are represented by arrows with white or black heads, respectively. In the Periodic Collection scheme, sensor nodes transmit messages at a given rate $\mathcal{R}$ regardless of the presence of events. Instead of transmitting a message immediately after the detection of an event, the message is temporarily stored until the next scheduled transmission time, as shown in Fig. 10b. Since real and bogus traffic are indistinguishable from each other, this method provides perfect event source unobservability because the transmission rate is not altered by the presence of events.



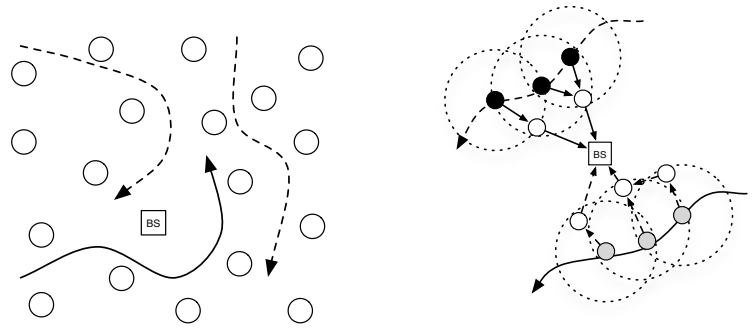
(a) Flawed fake injection mechanism (b) Perfect event source unobservability

Fig. 10. Periodic Fake Packet Injection

As event messages need to be delayed until the next scheduled transmission time, this poses a serious limitation in time-critical applications. Intuitively, the delivery delay can be reduced by changing scheduling in order to have shorter inter-transmission times. However, this impacts negatively on the energy waste of the network. Therefore, the transmission rate must be carefully adjusted in order to ensure the durability of the network without incurring an excessive delay.

Energy-Aware Approaches There has been an extensive body of research which focuses on reducing the overhead imposed by the injection of fake messages at regular intervals. These proposed solutions have approached the problem in different ways: simulating the presence of events in the field, filtering out fake traffic, using already existing traffic to convey event data, and sending messages according to a given probability distribution.

The Source Simulation scheme [24] is based on the idea of saving energy by reducing the number of nodes transmitting fake messages. Instead of making all nodes send out messages at regular intervals, the network simulates the presence



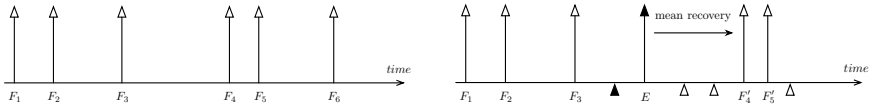
(a) Real and fake mobile events (b) Modelling of event propagation

Fig. 11. Unobservable Handoff Trajectory

of real events in the field. During network deployment, a set of L nodes are preloaded, each with a different token. These nodes generate fake traffic during the data transmission phase and after a predefined period of time, the token is passed to one of its neighbours (possibly itself) depending on the behaviour of real objects. The size of L determines the level of protection as well as the energy consumed by the network. The main problem with this approach lies in the difficulty of accurately modelling the movement of an object so it appears as real to the adversary.

The Unobservable Handoff Trajectory (UHT) [29] is another solution that simulates the presence of objets in the field. The UHT is a decentralised and self-adaptive scheme that generates fake mobile events with the same probability distribution as real events. Real events follow a Poisson distribution and fake events are generated in such a way that the overall distribution is not affected. The generation of dummy events starts at the perimeter of the network and propagates for a number of hops according to the length of real events (see Fig. 11a). Each perimeter node decides to generate a new dummy event independently based on the number of perimeter nodes and the number of real events they observe over a time window. After being created, fake messages must be propagated. This process is based on the fact that all the neighbours of a fake node receive the fake packets sent towards the base station. This packet contains who will be the next fake source in the path and also the length of the current event. The propagation is represented in Fig. 11b, where fake sources are shaded in grey and real sources in black while fake and real messages are represented with dashed and ordinary arrows, respectively.

Besides the cross-layer scheme described in Section 4.1, Shao et al. [42] proposed another version of the same solution that can protect against global adversaries. This alternative protocol is very similar to the Periodic Collection proposed by Mehta et al. but the main difference is that instead of using ordinary network traffic it takes advantage of the beaconing phase. This scheme also provides perfect event source unobservability at no additional cost since event



(a) Predefined message distribution (b) Message distribution adjustment

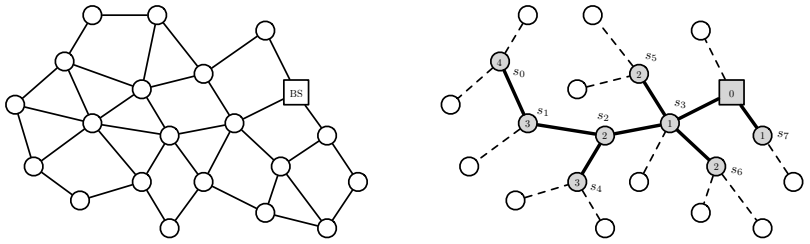
Fig. 12. Statistically strong source unobservability

data is hidden within beacon frames, which are periodically broadcast regardless of the occurrence or not, of events in the field. However, since the time between consecutive beacons is relatively large, the solution is only practical for some applications where no tight time restrictions exist.

In order to reduce network traffic while maintaining source unobservability, Yang et al. [50] proposed a bogus traffic filtering scheme. In this solution, any node sends real or fake messages at a given rate and some nodes operate a filtering proxies. Proxy nodes discard bogus traffic and temporarily buffer and re-encrypt real traffic before forwarding it. If there are no real messages available, a proxy node sends encrypted dummy messages. In the Proxy-based Filtering Scheme (PFS) selects a number of proxies and traffic is filtered by only them. In the Tree-based Filtering Scheme (TFS) packets can be processed by several proxy nodes as the move towards the base station, thus reducing fake traffic at the expense of increased network delay. A drawback to this solution is that an attacker can still use rate monitoring techniques to identify the proxy nodes, which are important for the operation of the network.

Another branch of research has concentrated on the concept of statistically strong source unobservability. This concept was introduced by Shao et al. [43] to relax the tight requirements of perfect event source unobservability while maintaining a statistical assurance on the protection of data source. Before deployment, sensor nodes are configured to transmit according to a message distribution F_i, as depicted in Fig. 12. During the data transmission phase, when an event E occurs, the real message can be transmitted before the next scheduled transmission, F_4, without altering the parameters (e.g., the mean and variance) of the distribution. This process is depicted in Fig. 12b. Sensor nodes keep a sliding window of previous inter-message delays $\{\delta_1, \delta_2, ..., \delta_{n-1}\}$ and, upon the occurrence of an event, δ_n is set to a value very close to 0 and gradually incremented by a small random number until the whole sliding window passes a goodness of fit test. Thus, the real event transmission can be sent ahead of the scheduled time without alerting the adversary even if he performs statistical tests on inter-message delays. The solution includes a mean recovery mechanism which delays subsequent transmissions because the presence of bursts of real messages might skew the mean of the distribution.

Recently, Alomair et al. [2] showed that a global adversary has more efficient ways of breaking statistically strong unobservability. Instead of focusing on the inter-message delays of a single sliding window, the attacker might try to spot differences between any two sliding windows (i.e., intervals) in order to detect



(a) A graph representation of the WSN (b) A potential MCDS

Fig. 13. Minimum Connected Dominating Set

the presence of real events. The strategy of the adversary is to identify short inter-message delays followed by long inter-message delays. These patterns are common in intervals containing real events because the delay of real messages is usually shorter than the mean in order to reduce the latency, and subsequent messages are delayed in order to adjust the mean of the distribution, as proposed in [43]. To the contrary, inter-message delays are independent identically distributed random variables in fake intervals. Consequently, by counting the number of short-long inter-message delays an attacker might be able to distinguish intervals containing real events. The solution proposed by Alomair et al. is to make fake intervals resemble intervals with real events by introducing some statistical interdependence between fake inter-message delays.

Proano and Lazos [36] pointed out that since a global vision is obtained by means of an adversarial network, the attacker cannot exactly determine the transmission rate of each and every sensor node. As a result, not all sensor nodes need to be active sources of fake traffic to deceive the adversary. They suggest reducing the number of fake data sources by partitioning the network into a minimum connected dominating set (MCDS) rooted at the base station. In a MCDS each node either belongs to the MCDS or is one hop away from it, as depicted in Fig. 13. In this way, the nodes in the MCDS transmit (real or fake) traffic at a given rate and the rest of the nodes regulate their transmissions in order to conform to the statistical traffic properties observed by an eavesdropper. Later, in [37], the same authors added a deterministic assignment scheme for coordinating sensor transmissions and thus reduce end-to-end delay for real packets. Nodes deeper in the MCDS are scheduled to transmit sooner, so that any real packet reaches the sink at the end of each interval. For example, in Fig. 13b, each time interval is divided into four subintervals since the maximum depth of the MCDS is four. Sensor node s_0 transmits at the first subinterval, node s_1 at the next subinterval, and so on.

Previous solutions have countered a passive global attacker. Yang et al. [51] consider a global attacker who, upon detecting suspicious cells devises an optimal route to visit these spots. Tthey propose two potential strategies to find a (pseudo-)optimal route to visit all suspicious cells. The first strategy is based on a greedy algorithm, which ends in polynomial time but is not globally op-

timal, and the second one is a dynamic programming algorithm, which finds the optimal solution but requires an exponential time to finish. Subsequently, the authors evaluate the impact of the proposed attacker model to two existing solutions: statistically strong source unobservability and source simulation. They conclude that the former behaves well when the rate of real messages to be delivered is low while the latter approach is suitable when the rate is high. As a result, Yang et al. propose a dynamic approach that combines the merits of both solutions by switching from the one to the other based on the load of the network.

4.3 Internal Adversaries

Some adversaries might be able to compromise and control a subset of nodes from the legitimate network. These nodes become internal adversaries since they can participate in the same tasks performed by any other network node and provide the attacker with any information contained in the packets they forward. The solutions devised to deal with these types of attackers are very limited and their approaches rather diverse.

The Identity, Route and Location privacy (IRL) algorithm [41] is as a network-level privacy solution. The primary goal of this solution is to provide source anonymity and location privacy as well as provide assurance that packets reach their destination. Although the authors do not consider the threat of internal adversaries, one of its features is suitable for just this purpose. The authors introduce the notion of trust and reputation to prevent routing through misbehaving adversaries. First, each node classifies its neighbours into four groups depending on their position with respect to the base station. Additionally, each node classifies its neighbours as either trustworthy or untrustworthy. When a node wants to transmit, it selects random trustworthy nodes which are closer to the base station. If no trustworthy nodes are found it tries with nodes at the same distance or in the opposite direction. In the case no trustworthy nodes are found, the node simply drops the packet. Therefore, each message follows a random path composed of trustworthy nodes only. Additionally, dishonest en-route nodes are unable to determine whether the sender is the real data source or a mere intermediary since nodes replace the identifier of received packets with their own at every hop.

Pongaliur and Xiao [35] propose to modify packets headers at dynamically selected nodes in the route to the base station to protect the identity of the data source from internal adversaries. When a node creates a packet it includes a pseudonym instead of its real identifier. This pseudonym is a value from a hash chain used in reverse order obtained from the real identifier of the node. The packets also include a random value that is used by intermediate nodes to determine whether to replace the identifier carried in the packet by their own pseudonym[7]. Additionally, a rehashing node concatenates the replaced identifier

[7] A hash function is applied to the random value and the result is used as input to a mapping function which returns 0 or 1 with a given probability.

to the payload and encrypts it with its own shared with the base station. An extra field is used for verifying the validity of the modifications. To that end, the base station needs to keep track of the hash chains of all the nodes in order to find the key corresponding to each of concatenated the hash values. Another limitation to this approach is that an internal adversary can estimate its distance to the data source based on the rehashing probability and the size of the payload.

The last solution is called *p*DCS [44] and its aim is to provide security and privacy in Data-Centric Sensor (DCS) networks, where the data collected by sensing nodes is forwarded and kept at storage node until the base station queries for them. Sensing nodes know where to send the data by means of a a publicly known mapping function. Since this function is public an attacker can easily determine which nodes to compromise to obtain a particular type of data. After compromising such nodes, he can also identify the location where the data was originally collected. *p*DCS is intended to protect against this type of threat. The scheme is based on the use of a secure mapping function[8] and the storage of encrypted data in a remote location. In the case the adversary compromises a storage node he is not able to decrypt the data contained in it because these data are encrypted with the key of the sensing nodes which collected them. If a sensing node is compromised, the attacker cannot determine where previous data was stored because the secure mapping function prevents this from happening. Moreover, when a node is found to be compromised there is a node revocation mechanism in order to prevent the attacker from obtaining the location of future event data. Finally, the authors suggest protecting the flow of data from the sensing to the storage node by means of any existing source-location privacy solution.

5 Receiver-Location Privacy

Receiver-location privacy refers to the protection of the destination of messages but it primarily concentrates on hiding the location of the base station. The location of the base station is exposed due to the peculiar communication pattern of WSNs: each sensor node transmits data messages to this single point. Intuitively, the solution is to normalise the traffic load by making each sensor node transmit, on average, the same number of messages but this incurs a prohibitive network overhead. In the following we analyse proposals dealing with local adversaries followed by solutions considering the threat of global adversaries. To the best of our knowledge, there are no solutions in the literature that study the threat of internal adversaries.

5.1 Local Adversaries

A local attacker usually starts at a random position in the network[9] and moves around until he overhears some transmissions in the area surrounding him. The

[8] A secure mapping function is basically a keyed hash function that uses as input the type of event and other secret information shared by a group of nodes.

[9] Starting at the edge of the network is, in our opinion, more realistic.

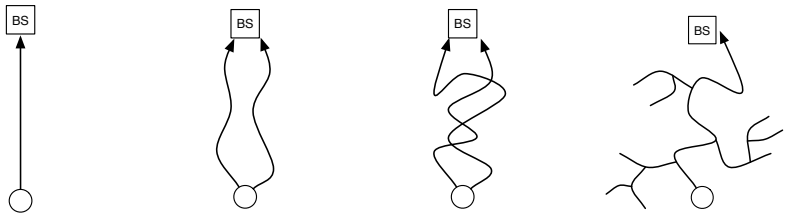
typical types of attacks performed by an adversary who wishes to find the sink are: content analysis, time correlation, and rate monitoring. Content analysis tries to obtain information from the packet headers or payload. Additionally, an attacker can observe the packet sending times of neighbouring nodes in order to determine the direction to the base station.Finally, in a rate monitoring attack, the strategy of the adversary is to move in the direction of those nodes with higher transmission rates since nodes in the vicinity of the base station receive more packets than remote nodes.

Next we analyse some basic countermeasures against the aforementioned attacks followed by a set of more advanced solutions. Most of these solutions aim to balance the amount of traffic between all network nodes by selecting the next hop based on some probability while other solutions attempt to disguise or emulate the presence of the base station at different locations. Again, some solutions may fall into several categories depending on the features analysed.

Basic Countermeasures Some basic countermeasures have been proposed in [12] to prevent the aforementioned attacks. First, content analysis can be hindered by applying secure data encryption on a hop-by-hop basis. This process should be applied throughout the whole lifetime of the network but it is not easy to satisfy this requirement until each node shares pairwise keys with all its neighbours. Thus, they propose an ID confusion technique to conceal the source and destination during the route discovery phase. This technique is based on reversible hash functions so that when a node x sends a message to node y, it randomly selects an element from $C_x = \{h_x : x = H(x)\}$ as the source address, and an element from $C_y = \{h_y : y = H(y)\}$ as the destination address. Finally, it encrypts the whole packet with a network-wide shared key pre-loaded on all sensor nodes. A receiving node decrypts the message and, by reverting the hash function, it obtains the true sender and intended recipient.

During data transmission, sensor nodes must ensure that packets change their appearance as they move towards the base station. Each node in the path must decrypt any received packet and then re-encrypt it with the key shared with the next node in the route. However, even if the attacker cannot observe the contents of the packets, he can learn some information from packet sending times and eventually infer the relationship between parent and child (i.e., closer and further) nodes. To prevent this, Deng et al. [11] propose applying random delays to the transmission of packets. Additionally, the authors suggest creating a uniform sending rate to prevent rate monitoring attacks. This can be achieved by making a parent node accept packets from a child node only if its own packet has been forwarded. In the case the parent node has nothing new to send, it can simply continue to send the same packet or inject dummy traffic.

There are some limitations to these basic countermeasures that require the development of further solutions. The following schemes aim to reduce these limitations.



(a) Single-path (b) MPR (c) MPR+RW (d) MPR+RW+FP

Fig. 14. Schematic of Several Multi-Parent Routing Techniques

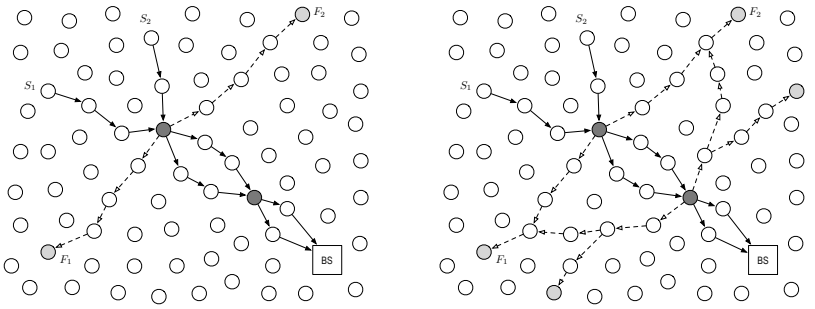
Biased Random Walks This category brings together solutions where the routing process is random but somehow biased towards the base station. The first solution is also presented by Deng et al. [11] and is called Multi-Parent Routing (MPR). The MPR consists of making each sensor node pick the next element in the path uniformly at random from its set of parent nodes. See in Fig. 14 a comparison between a single-path routing and a MPR scheme. In Fig. 14a all transmissions use the same transmission path, which is represented by a straight arrow, while in Fig. 14b the paths followed by two different packets are represented. The MPR scheme obtains a better load balance as data packets spread within a band of nodes next to the shortest path from the data source to the base station. However, the traffic flow still points to the base station as the next communication hop is always selected from the list of parent nodes. To further diversify routing paths, the authors suggest combining MPR with a random walk (RW) routing scheme. In this version of the protocol, nodes forward packets to a parent node with probability p_r and to a randomly chosen neighbour with probability $1 - p_r$. Consequently, packets may not only travel towards the base station but in any other direction. In Fig. 14c we depict two routing paths which at some points move in the opposite direction to the base station. This scheme provides better security at the cost of a higher message delivery delay.

Similarly, Jian et al. [15] propose to make every sensor node divide its neighbours two groups. The first group contains nodes which are closer to the base station and the second group contains the rest of their neighbours. So, nodes forward packets to further nodes with probability $P_f < 1/2$ and to closer nodes with probability $1 - P_f$. This implies that the transmission is biased and the attacker is able to infer the direction to the data sink. To prevent this, the authors inject fake packets in the opposite direction to the base station with probability P_{fake} after receiving a real packet. This packet travels for several hops away from the base station. In general, the adversary cannot distinguish real from fake traffic which makes this solution secure since packets flow in any direction with an even probability. However, if the adversary observes a node that does not forward a packet he knows that it is a fake packet. As fake packets are sent to further neighbours exclusively, the adversary learns that the base station is in the opposite direction.

Rios et al. [39] devised an new strategy that solves the previous problem. They suggest to send a pair of messages (real and fake) for every transmission in such a way that real traffic is more likely to be sent towards the base station and fake traffic is used to compensate the message rate for every neighbour. When fake traffic is received by a node, it continues sending two messages, both of which are fake, for a number of hops that depends on the hearing range of the adversary. The branches of fake traffic must reach out of the hearing range of the adversary. Now, if the adversary observes a node that drops a received packet he knows that this packet is fake but he is unable to determine the direction to the base station since fake packets are sent in any possible direction.

Fake Traffic Injection Deng et al. [11] proposed new ways of improving MPR based on the injection of fake traffic. Fractal Propagation (FP) was designed to be used in conjunction with MPR and RW. When a sensor node observes that a neighbouring node is forwarding a data packet to the base station, it generates a fake packet with probability p_c and forwards it to one of its neighbours. The durability of fake packets is controlled by means of a global time-to-live parameter K. Also, if a node observes a fake packet with parameter k $(0 < k < K)$ it propagates another fake packet with time-to-live parameter $k - 1$. Fig. 14d shows the trace resulting from the transmission of a single packet using the three mechanisms together. The main problem of the FP scheme is that nodes in the vicinity of the base station generate much more fake traffic than remote nodes. To address this problem, the authors propose the Differential Fractal Propagation (DFP), where sensor nodes adjust their probability of generating fake traffic p_c according to the number of packets they forward. Besides reducing the energy waste, this scheme provides better privacy protection because it balances the network traffic load more evenly.

Yao et al. propose in [55] a new fake packet injection scheme. Real packets are sent to the base station using the shortest path and when two paths of real messages intersect at some point, the node receiving these packets sends two fake packets to two fake data sinks after a timer expires or a packet counter reaches a certain threshold. In this way, real and fake data sinks receive a similar number of packets. Moreover, when a packet reaches subsequent intersection points, the intersection node sends N_f packets to some random destinations. This process is depicted in Fig. 15, where dark grey nodes represent intersection nodes, light grey nodes are fake sinks or some random data destinations. Ordinary arrows symbolise real data packets while dashed arrows represent fake packets. In Fig. 15a the first intersection node transmits fake traffic to both fake data sinks. Meanwhile, the second intersection node introduces fake traffic to other random destinations as well. The main problem of Yao et al.'s approach is an attacker starting from a data source and tracing packets can trivially reach the first intermediate node. From that point, he can distinguish fake paths since they may imply an abrupt change in the angle of transmission. This problem has already been discussed for other solutions.



(a) Injection at first intersection node (b) Injection at second intersection node

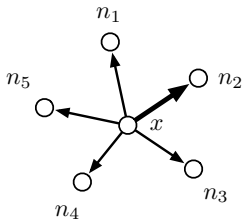
Fig. 15. Yao et al. Fake Packet Injection Scheme

Sink Simulation Some approaches try to emulate the presence of the base station at different points in the field. Simulation techniques are based on the generation of fake traffic but, instead of being transmitted in random directions, it is addressed to particular network locations. This results in a concentration of high volumes of fake traffic, called hotspots, the objective of which is to draw the adversary away from the true data sink. The main challenge is to create hotspots that are evenly distributed throughout the network with a minimum overhead.

Maelstrom [5] is one of such solutions that generates a number of fake data sinks. After deployment, the base station sends N special configuration packets, each of which is configured to travel H_s hops away from the base station. After that, each of these packets travel H_r random hops to any node on the same level or further away. The final recipients of these packets become the centre of a maelstrom area and announce this by sending a discovery packet to nearby nodes. During data transmission, when a node receives a real packet it generates, with a probability, a fake message and forwards it to its closest maelstrom. However, once an intelligent attacker reaches a maelstrom area he can discard it as the true data sink.

A similar approach is proposed by Biswas et al. [3]. The idea is to evenly distribute multiple fake data sinks with the largest number of neighbours, since this implies more incoming traffic. During data transmission, each node is configured to transmit a fixed number of messages either real or fake so that after a given time period all nodes have sent the same amount of traffic. Fake traffic is directed to fake data sink by its neighbours except for nodes which are not immediate neighbours, where the selection of a fake destination is done in a round-robin fashion. The result should be that fake base stations receive at least the same amount of traffic as the actual base station. This approach may deal with naive rate monitoring adversaries but it can be defeated by informed adversaries.

Finally, Deng et al. [11] refined their fractal propagation solutions and created a new scheme called Differential Enforced Fractal Propagation (DEFP) that is



neighs(x)	tickets	Probability
n_1	1	1/8
n_2	4	1/2
n_3	1	1/8
n_4	1	1/8
n_5	1	1/8

Fig. 16. Decentralised Hotspot Generation in DEFP

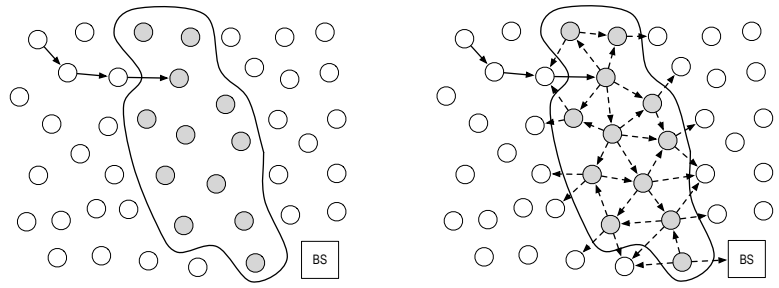
capable of creating hotspots in a decentralised and dynamic way. Sensor nodes keep track of the number of fake packets forwarded to each neighbour and new fake traffic is more likely to be sent to neighbours who have previously received more fake traffic, as shown in Fig. 16. In this way there is no need for a central authority or a complex coordination system to establish where the hotspots should be placed. Another interesting feature of this solution is that the hotspots can be deactivated by simply resetting the forwarding probabilities of each node. After that, new hotspot locations are likely to appear, which prevents smart attackers from discarding fake data sinks (i.e., hotspots) until they find the real base station.

5.2 Global Adversaries

The aforementioned techniques are considered to be effective only in a local adversarial model but some of them may also provide some means of protection against global adversaries. As a matter of fact, they can be useful if the global adversary has no real-time analysing capabilities.

Again, the injection of fake traffic is one of the main approaches for protecting from global adversaries. Making the base station mimic the behaviour of sensor nodes, simulating the presence of several data sinks, and moving the base station to a different location might also be useful solutions.

Bogus Traffic As mentioned in Section 5, flooding the network with messages is a simple yet efficient mechanism to protect the location of the base station. The main drawback to flooding is the high communication cost associated with the retransmission of the same message to every corner of the network. Backbone flooding [25] reduces the communication cost by limiting the transmissions within a backbone area. The backbone area consists of a sufficient number of adjacent nodes to achieve a desired level of privacy. Any data packet generated in the network is addressed to the backbone, where it spreads to all its members. Since data sinks must be located at least within the range of a backbone member, they overhear all messages. A major limitation to this approach is that the backbone is static. The authors suggest to alleviated this problem by (a) periodically rebuilding the backbone or (b) defining several backbones from the beginning.



(a) Data packet reaches the backbone (b) Flooding within the backbone

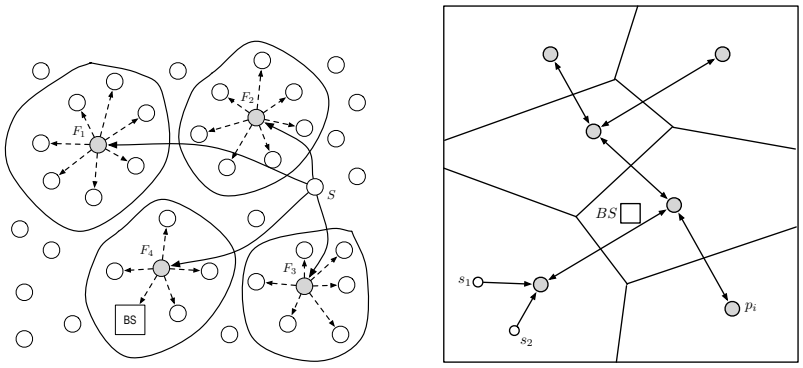
Fig. 17. Backbone Flooding

Fig. 17 illustrates the transmission of a data packet and its propagation within the backbone area.

The scheme called Concealing Sink Location (CSL) [56] follows a different strategy. The idea is to make each sensor node transmit at the same rate regardless of its distance to the base station. This rate is calculated for nodes at distance i from the sink by counting the number of nodes at distance greater than i and dividing it by the number of nodes at distance i. This ratio represents that each node must send its own traffic and forward the traffic from nodes further away. The number of nodes at a given distance i is estimated via geometric analysis considering the size of the deployment area and a uniform distribution of the nodes in the field. However, these estimations may differ significantly from the reality. Also, it is important to note that the authors assume that sensor nodes have a similar transmission rate for real messages but this might not be the case in the presence of bursts of messages.

A similar approach is followed in [**?**], where the transmission rate of nodes is calculated based on the number of child nodes an immediate neighbour of the sink has. The idea is to make all sensor nodes transmit as many messages as a sink neighbour has to since they are the busiest nodes. When a sensor node receives a fake packet it simply drops it, while if the packet is real, it buffers it temporarily. In the meantime the sensor node generates fake traffic to satisfy the overall transmission rate. The authors claim that by generating that much traffic the lifetime of the network is not reduced. The argument is that all nodes in the network will deplete their batteries at the same time and not only the sink neighbours. However, they have not considered that in this way the transceivers of the nodes are active most of the time and they need to decrypt much more messages. Also, they have not considered collisions and packet retransmissions.

Sink Simulation Sink simulation has also been suggested as a mechanism to protect from global adversaries. Mehta et al. [25] propose simulating the presence of several data sinks in the field. During the deployment k of sensor nodes are picked as fake data sinks and the true data sinks are manually placed within the



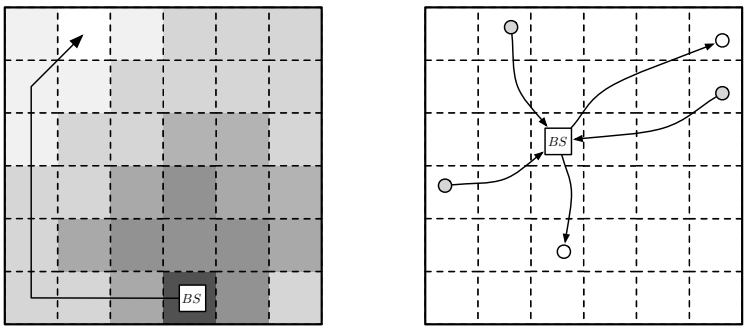
(a) Mehta et al.'s sink simulation (b) Network partition and tree formation

Fig. 18. Examples of Sink Simulation Approaches

communication range of some of these. The number of fake sinks must outnumber the number of true sinks. When a source node detects event data, it send them to all the fake data sinks, which on reception broadcast the message locally. This process is illustrated in Fig. 18a, where the data source S sends messages to $F_1, \ldots, F_2$ and each of them broadcast the message locally. Since all fake sinks receive the same amount of traffic, they are all equally likely to be next to a true data sink. The larger the value of k the better the protection but the higher the volume of traffic in the network.

The solution in [4] is also based on the concept of k-anonymity. The idea is to have at least k nodes with a communication pattern similar to the nodes around the base station. To that end, the network is partitioned into k Voronoi regions, each of which contains a node that collects all the information sensed in that region. These nodes p_i are organised as an Euclidean minimum-spanning tree and the data they received from their own region is forwarded to all other tree members. Fig. 18a shows a Voronoi partition of the network for the designated nodes p_i, in grey. Note that all nodes connecting the designated nodes see all the network traffic and thus the base station simply needs to be placed close to one of them. As a result, the uncertainty of the attacker is much greater than in the previous scheme for the same value of k. However, the nodes forming the tree are highly likely to deplete their batteries much sooner than the rest of the nodes.

Wang and Hsiang [47] propose another solution that starts by generating a shortest-path tree rooted at the base station. After that, neighbouring leaf nodes establish communication links to generate network cycles. During data transmission, the shortest-path tree is used to transmit data to the base station and, simultaneously, fake packets are injected into the cycles. Fake traffic continues moving along the cycle until it is completed. When several cycles intersect at a node it creates a hotspot since it receives all the bogus traffic from the cycles. The authors include a mechanism to limit the number of cycles by



(a) Safest route in the RIA scheme (b) Selective packet retransmission

Fig. 19. Relocation and Disguise Examples

allowing leaf nodes to establish links only if their least common ancestor is at least h hops away from both nodes. In this way, each of the hotspots receive more traffic. Even though the authors assume a global adversarial model, this solution does not seem suitable for that purpose. The main problem is that the true sink behaves differently from the rest of the artificial hotspots. While the transmission rate of the base station is negligible, fake hotspots must forward the real data packets coming from its child nodes.

Relocation and Disguise As far back as 2003, Deng et al. [12] suggested the reallocation of the base station for enhanced security. They assume that the base station has complete knowledge of the topology of the network and thus it may calculate an optimal future location that maximises its security. Actually, they do not address a global eavesdropper but a compromised node dropping packets. Therefore, we refer the reader to their paper for further details.

Possibly motivated by the approach just mentioned, Acharya and Younis propose the Relocation for Increased Anonymity (RIA) scheme [1], where the base station finds a new location by considering both the impact over network performance and its own level of protection. The base station calculates a score for each cell based on the node density and the threat level (i.e., transmission rate). The rationale behind this scoring mechanism is that by moving the base station to a cell with a low threat , the cells with high activity need to send packets to remote areas, which increases the delivery time and consumes more energy. Likewise, if there is a low transmission rate due to a reduced node density, moving the base station to that cell would cause the few nodes in the cell to become overwhelmed with traffic. Once the base station knows which is the most suitable cell to reside in, it follows the safest route to reach the final destination. In Fig. 19a we depict the path selected by the base station for relocation based on the scores of each of its cells, the cells with higher scores are depicted in a lighter colour.

Mimicking the behaviour of ordinary sensor nodes is another way of hiding the base station from global adversaries. The Base-station Anonymity increase through selective packet Re-transmission (BAR) [1] suggests to make the base station decide whether to forward the packets it receives for several hops. The length of the walk is dynamically adjusted based on the level of threat perceived by the base station. If the base station needs to increase its level of protection it defines longer walks. The general idea is that by doing this, the number of transmissions in remote cells increase and thus the attacker cannot clearly identify the actual location of the base station based on the transmission rate of a cell. An example of this approach is illustrated in Fig. 19b, where source nodes and destination nodes are represented as grey and white circles, respectively. The main problem with this approach is that by forwarding packets to random remote locations, the base station is also increasing the transmission rate of the cells in its vicinity. Consequently, the attacker may still spot the base station as the cell with the highest transmission rate.

Finally, the Decoy Sink Protocol [10] combines indirection and data aggregation to reduce the amount of traffic received by the base station. Instead of sending the data to the base station directly, sensor nodes are programmed to transmit their packets to an intermediate node (i.e., the decoy sink) and, on their way, the data are aggregated. Finally the decoy sink sends the result of the aggregation to the base station. Although this may prevent the attacker from determining the location of the true data sink, this scheme exposes the location of the decoy sink. If the goal of the attacker is to compromise the base station, he obtains a similar result by compromising the decoy sink. Also, if he destroys it the protocol stops working. This problem is contemplated by the authors and they suggest picking several random nodes during the initialisation of the network to operate as decoy sinks. During the transmission period, sensor nodes send all their readings to a particular decoy sink for a pre-established period of time. This version of the protocol adds robustness to the network and balances the traffic load but the attacker is still able to ultimately achieve his original goal.

6 Challenges and Future Trends

Privacy preservation in WSNs has proven to be an extremely challenging task and regardless of the number of solutions that have been devised there are several open questions that need further attention:

- **Cost-effective solutions**. The main approach to location privacy is to increase the number of transmissions in order to mislead the adversary from the target in some way. However, sending more packets implies more energy waste and increased delays. This overhead is normally related to the level of protection provided by the solutions but sending more packets does not always increase privacy, as shown by angle-based privacy solutions. Moreover, many solutions are incapable of completely deceiving the adversary and can

only guarantee a longer safety period until the adversary eventually finds the target. Consequently, it is necessary to devise and develop new solutions that keep to a reasonable energy budget without sacrificing the level of protection. Some solutions based on innovative techniques already exist (e.g., cross-layer routing and context-aware location privacy) but there is still room for original research in the area.

- **Holistic privacy**. Despite the number of solutions existing in the literature devoted to protecting source- and receiver-location privacy, there is no single scheme capable of effectively and efficiently providing an integral solution to both problems simultaneously. While source-location privacy can be achieved by hiding the transmissions of real packets, receiver-location privacy demands a homogeneous traffic load in the network. Therefore, a naive solution to these problems is to use baseline flooding together with fake data sources. However, this approach is too energy consuming for ordinary sensor networks, where the energy budget is rather limited. How to solve this problem in an energy-efficient way demands further attention from the research community.
- **Interoperability framework**. Another open problem in the literature is the lack of a unified framework for quantifying location privacy for comparing different solutions. Currently, different authors resort to different approaches such as measuring entropy, game theory, evidence theory, numerical analysis, and simulations. However, it is not trivial to provide a formal model that accurately represents the behaviour of the system, especially in the context of a local adversary. Although it is possible to measure the privacy loss in one step, the information leak accumulates in a way that remains intractable as the adversary moves in the field. Probably, this is the reason why simulations is the most common approach to proving the correctness of solutions. But simulation results are not easily reproducible because either the simulator is not standardised or the code is not made publicly available, or both. Thus, defining an interoperability framework is a challenging area of research that may help to devise new contrasted solutions.
- **More skilled adversaries**. Also in relation with the previous issue, it is necessary to formally and faithfully define the capabilities and actions that may be performed by the adversary. The traditional approach is to define an adversary with a predefined strategy that remains unaltered. An appropriate model for representing the knowledge of the adversary does not exist. At most, the adversary knows whether he has visited a specific node before or not. The adversary does not use or infer new information based on previously known data or additional sources of information. For example, the adversary might use the routing tables of the nodes to compromise receiver-location privacy. In this regard, the adversarial model considered in the literature is mostly passive and does not interfere with the normal operation of the network. Particular attention must be paid to adversaries who can inject, modify, reply, or block messages from a portion of the network given the hardware limitations of sensor nodes. Also, more research must be conducted

to devise solutions against internal adversaries, which are not only capable of obtaining contextual information but also payload contents.
- **Dynamic environments and future scenarios**. All the solutions analysed here only consider static networks. Once placed, sensor nodes are not reallocated to another location. However, the Internet of Things opens the door to new scenarios where everyday objects are fitted with computational power and limited batteries. This will result in one of the most promising areas for innovation. In this landscape, mobility is of paramount importance but it may also imply intermittent network connectivity and the use of untrustworthy data relays to reach the base station. Moreover, it is possible that not only the base station has a connection to the outside world, but the sensor nodes could also be directly connected to the Internet. Similarly, new types of adversaries might appear. Therefore, we believe that the integration of sensor networks with the Internet will result in a prolific area of study.

Note that this paper has focused on location privacy but there are more metadata that may be leaked from the operation of the network. For example, it is important to hide the moment in time when an event takes place (i.e., temporal privacy) since it allows an adversary to predict future behaviours of the elements being monitored by the network. Also, there is also room for innovation and research in content-oriented privacy, which is primarily aimed to hide packets contents while enabling data-aggregation. Finally, another related issue that requires further attention is query privacy, namely, preventing the disclosure of a query based on the nodes that respond to it.

7 Conclusions

This paper has presented a taxonomy of solutions for location privacy in Wireless Sensor Networks. The taxonomy is organised based on the information to be protected and the capabilities of the adversary that may want to compromise location privacy. More than 50 papers have been analysed including solutions for node anonymity, source-location privacy, and receiver-location privacy. In general, local adversaries are countered by means of random walk routing solutions, which are ineffective against global adversaries. Dummy traffic injection is the typical approach to provide protection against more powerful adversaries but the overhead imposed by these solutions is overly high. Internal adversaries have not received sufficient attention yet.

Prior to analysing solutions we have studied whether traditional anonymous communication systems are suitable for protecting location privacy in WSNs. This study has first considered which anonymity requirements are desirable for the sensors' domain and then we have studied the overhead and limitations imposed by some renowned anonymous communication systems. From this, we have shown that some of these solutions are sufficiently lightweight to run in sensor nodes but either the anonymity requirements or the adversarial model differ from the ones considered in WSNs. To the contrary, other solutions are

suitable for the location privacy problem but impose a high overhead or limit the usability of the network.

At the end of this paper we present a number of challenges and open issues that must be addressed by the research community to facilitate the acceptance of sensor networks and other foreseeable technologies.

Acknowledgements

This work has been partially funded by the European Commission through the FP7 project NESSoS (FP7 256890), the Spanish Ministry of Science and Innovation through the ARES project (CSD2007-00004) and the Andalusian Government PISCIS project (P10-TIC-06334).

References

1. Acharya, U., Younis, M.: Increasing base-station anonymity in wireless sensor networks. Ad Hoc Networks 8(8), 791–809 (2010)
2. Alomair, B., Clark, A., Cuellar, J., Poovendran, R.: Towards a Statistical Framework for Source Anonymity in Sensor Networks. IEEE Transactions on Mobile Computing 12(2), 248 – 260 (2012)
3. Biswas, S., Mukherjee, S., Mukhopadhyaya, K.: A Countermeasure against Traffic-Analysis based Base Station Detection in WSN (2008), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.98.948
4. Chai, G., Xu, M., Xu, W., Lin, Z.: Enhancing sink-location privacy in wireless sensor networks through k-anonymity. International Journal of Distributed Sensor Networks 2012, 16 (2012)
5. Chang, S., Qi, Y., Zhu, H., Dong, M., Ota, K.: Maelstrom: Receiver-Location Preserving in Wireless Sensor Networks. In: Wireless Algorithms, Systems, and Applications, LNCS, vol. 6843, pp. 190–201. Springer (2011)
6. Chaum, D.: Untraceable Electronic Mail, Return addresses, and Digital Pseudonyms. Commun. ACM 24(2), 84–88 (Feb 1981)
7. Chaum, D.: The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability. Journal of Cryptology 1, 65–75 (1988)
8. Chen, H., Lou, W.: From Nowhere to Somewhere: Protecting End-to-End Location Privacy in Wireless Sensor Networks. In: 29th International Performance Computing and Communications Conference. pp. 1–8. IPCCC'10, IEEE (2010)
9. Chen, J., Du, X., Fang, B.: An Efficient Anonymous Communication Protocol for Wireless Sensor Networks. Wireless Communications and Mobile Computing 12(14), 1302–1312 (Oct 2012)
10. Conner, W., Abdelzaher, T., Nahrstedt, K.: Using Data Aggregation to Prevent Traffic Analysis in Wireless Sensor Networks. In: Distributed Computing in Sensor Systems, LNCS, vol. 4026, pp. 202–217. Springer (2006)
11. Deng, J., Han, R., Mishra, S.: Decorrelating wireless sensor network traffic to inhibit traffic analysis attacks. Pervasive and Mobile Computing 2(2), 159–186 (2006)
12. Deng, J., Han, R., Mishra, S.: Enhancing Base Station Security in Wireless Sensor Networks. Tech. Rep. CU-CS-951-03, University of Colorado (2003), http://www.cs.colorado.edu/~mishras/research/papers/tech03-1.pdf

13. Gómez, C., Paradells, J., Caballero, J.E.: Sensors Everywhere: Wireless Network Technologies and Solutions. Fundación Vodafone España (2010), http://fundacion.vodafone.es/static/fichero/pre_ucm_mgmt_002618.pdf, iSBN 978-84-934740-5-8
14. Jhumka, A., Leeke, M., Shrestha, S.: On the Use of Fake Sources for Source Location Privacy: Trade-Offs Between Energy and Privacy. The Computer Journal 54(6), 860–874 (2011)
15. Jian, Y., Chen, S., Zhang, Z., Zhang, L.: A novel scheme for protecting receiver's location privacy in wireless sensor networks. IEEE Transactions on Wireless Communications 7(10), 3769–3779 (October 2008)
16. Jiang, J.R., Sheu, J.P., Tu, C., Wu, J.W.: An Anonymous Path Routing (APR) Protocol for Wireless Sensor Networks. Journal of Information Science and Engineering 27(2), 657–680 (2011)
17. Kamat, P., Zhang, Y., Trappe, W., Ozturk, C.: Enhancing Source-Location Privacy in Sensor Network Routing. In: 25th IEEE International Conference on Distributed Computing Systems. pp. 599–608. ICDCS 2005 (June 2005)
18. Kazatzopoulos, L., Delakouridis, K., Marias, G.F.: A privacy-aware overlay routing scheme in wsns. Security and Communication Networks 4(7), 729–743 (Jul 2011)
19. Li, Y., Lightfoot, L., Ren, J.: Routing-Based Source-Location Privacy Protection in Wireless Sensor Networks. In: IEEE International Conference on Electro/Information Technology. pp. 29–34. EIT'09 (2009)
20. Li, Y., Ren, J.: Providing Source-Location Privacy in Wireless Sensor Networks. In: 4th International Conference on Wireless Algorithms, Systems, and Applications. pp. 338–347. WASA '09, Springer-Verlag, Berlin, Heidelberg (2009)
21. Li, Y., Ren, J., Wu, J.: Quantitative measurement and design of source-location privacy schemes for wireless sensor networks. IEEE Transactions on Parallel and Distributed Systems 23, 1302–1311 (July 2012)
22. Lightfoot, L., Li, Y., Ren, J.: STaR: design and quantitative measurement of source-location privacy for wireless sensor networks. Security and Communication Networks (Online Mar 2012)
23. Mahmoud, M., Shen, X.: A Cloud-Based Scheme for Protecting Source-Location Privacy against Hotspot-Locating Attack in Wireless Sensor Networks. IEEE Transactions on Parallel and Distributed Systems 23(10), 1805–1818 (2012)
24. Mehta, K., Liu, D., Wright, M.: Location Privacy in Sensor Networks Against a Global Eavesdropper. In: IEEE International Conference on Network Protocols. pp. 314–323. ICNP 2007, IEEE, Beijing, China (16–19 Oct 2007)
25. Mehta, K., Liu, D., Wright, M.: Protecting Location Privacy in Sensor Networks Against a Global Eavesdropper. IEEE Transactions on Mobile Computing 11(2), 320–336 (2012)
26. Misra, S., Xue, G.: Efficient anonymity schemes for clustered wireless sensor networks. International Journal of Sensor Networks 1(1), 50–63 (2006)
27. Nezhad, A.A., Makrakis, D., Miri, A.: Anonymous Topology Discovery for Multihop Wireless Sensor Networks. In: 3rd ACM workshop on QoS and security for wireless and mobile networks. pp. 78–85. Q2SWinet '07, ACM, New York, NY, USA (2007)
28. Nezhad, A.A., Miri, A., Makrakis, D.: Location privacy and anonymity preserving routing for wireless sensor networks. Computer Networks 52(18), 3433 – 3452 (Dec 2008)
29. Ortolani, S., Conti, M., Crispo, B., Di Pietro, R.: Events privacy in WSNs: A new model and its applications. In: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM). pp. 1 –9 (june 2011)

30. Ouyang, Y., Le, Z., Chen, G., Ford, J., Makedon, F.: Entrapping Adversaries for Source Protection in Sensor Networks. In: 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks. pp. 23–34. WOWMOM '06, IEEE Computer Society, Washington, DC, USA (2006)
31. Ouyang, Y., Le, Z., Xu, Y., Triandopoulos, N., Zhang, S., Ford, J., Makedon, F.: Providing Anonymity in Wireless Sensor Networks. In: IEEE International Conference on Pervasive Services. pp. 145–148 (July 2007)
32. Ozturk, C., Zhang, Y., Trappe, W.: Source-Location Privacy in Energy-Constrained Sensor Network Routing. In: 2nd ACM workshop on Security of ad hoc and sensor networks. pp. 88–93. SASN, ACM New York, NY, USA, Washington, DC, USA (2004)
33. Pai, S., Bermudez, S., Wicker, S., Meingast, M., Roosta, T., Sastry, S., Mulligan, D.: Transactional Confidentiality in Sensor Networks. IEEE Security & Privacy 6(4), 28–35 (July-Aug 2008)
34. Pfitzmann, A., Hansen, M.: A terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management (Aug 2010), `http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf`, v0.34
35. Pongaliur, K., Xiao, L.: Sensor Node Source Privacy and Packet Recovery Under Eavesdropping and Node Compromise Attacks. ACM Transactions on Sensor Networks 9(4), 50:1–50:26 (Jul 2013)
36. Proano, A., Lazos, L.: Hiding contextual information in wsns. In: IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks. pp. 1 –6. WoWMoM (june 2012)
37. Proano, A., Lazos, L.: Perfect Contextual Information Privacy in WSNs under Colluding Eavesdroppers. In: 6th ACM Conference on Security and Privacy in Wireless and Mobile Networks. WiSec, ACM, Budapest, Hungary (April 17-19 2013)
38. Reiter, M., Rubin, A.: Crowds: Anonymity for Web Transactions. ACM transactions on information and system security 1(1), 66–92 (1998)
39. Rios, R., Cuellar, J., Lopez, J.: Robust Probabilistic Fake Packet Injection for Receiver-Location Privacy in WSN. In: Foresti, S., Yung, M., Martinelli, F. (eds.) 17th European Symposium on Research in Computer Security (ESORICS 2012). LNCS, vol. 7459, pp. 163–180. Springer, Pisa (Italy) (Sept 2012)
40. Rios, R., Lopez, J.: Exploiting context-awareness to enhance source-location privacy in wireless sensor networks. The Computer Journal 54(10), 1603–1615 (2011), impact Factor: 0.79
41. Shaikh, R., Jameel, H., d'Auriol, B., Lee, S., Song, Y.J., Lee, H.: Network Level Privacy for Wireless Sensor Networks. In: 4th International Conference on Information Assurance and Security. pp. 261–266. ISIAS '08 (Sept 2008)
42. Shao, M., Hu, W., Zhu, S., Cao, G., Krishnamurthy, S., La Porta, T.: Cross-layer Enhanced Source Location Privacy in Sensor Networks. In: IEEE Conference on Sensor, Mesh, and Ad Hoc Communications and Networks. pp. 1–9. SECON '09, IEEE Communications Society (June 2009)
43. Shao, M., Yang, Y., Zhu, S., Cao, G.: Towards Statistically Strong Source Anonymity for Sensor Networks. In: 27th IEEE Conference on Computer Communications. pp. 466–474. INFOCOM 2008 (April 2008)
44. Shao, M., Zhu, S., Zhang, W., Cao, G., Yang, Y.: pdcs: Security and privacy support for data-centric sensor networks. Mobile Computing, IEEE Transactions on 8(8), 1023–1038 (Aug 2009)

45. Walters, J., Liang, Z., Shi, W., Chaudhary, V.: Security in Distributed, Grid, and Pervasive Computing, chap. Wireless Sensor Network Security: A Survey, pp. 367–409. Auerbach Pub (2007)
46. Wang, H., Sheng, B., Li, Q.: Privacy-aware routing in sensor networks. Computer Networks 53(9), 1512–1529 (2009)
47. Wang, H.J., Hsiang, T.R.: Defending Traffic Analysis with Communication Cycles in Wireless Sensor Networks. In: 10th International Symposium on Pervasive Systems, Algorithms, and Networks. pp. 166 –171. ISPAN (2009)
48. Wei-Ping, W., Liang, C., Jian-Xin, W.: A source-location privacy protocol in WSN based on locational angle. In: IEEE International Conference on Communications. pp. 1630–1634. ICC '08, IEEE Communications Society, Beijing (19–23 May 2008)
49. Xi, Y., Schwiebert, L., Shi, W.: Preserving Source Location Privacy in Monitoring-Based Wireless Sensor Networks. In: 20th International Parallel and Distributed Processing Symposium. p. 8 pp. IPDPS 2006 (April 2006)
50. Yang, Y., Shao, M., Zhu, S., Urgaonkar, B., Cao, G.: Towards Event Source Unobservability with Minimum Network Traffic in Sensor Networks. In: 1st ACM conference on Wireless network security. pp. 77–88. WiSec'08, ACM, New York, NY, USA (2008)
51. Yang, Y., Zhu, S., Cao, G., LaPorta, T.: An Active Global Attack Model for Sensor Source Location Privacy: Analysis and Countermeasures. In: Security and Privacy in Communication Networks, Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 19, pp. 373–393. Springer Berlin Heidelberg (2009)
52. Yao, J.: Source-location privacy based on directed greedy walk in wireless sensor networks. In: Wireless Communications Networking and Mobile Computing (WiCOM), 2010 6th International Conference on. pp. 1–4 (sept 2010)
53. Yao, J., Wen, G.: Preserving source-location privacy in energy-constrained wireless sensor networks. In: Proceedings of the 28th International Conference on Distributed Computing Systems Workshops. pp. 412–416. ICDCSW '08, IEEE Computer Society, Washington, DC, USA (2008)
54. Yao, L., Kang, L., Deng, F., Deng, J., Wu, G.: Protecting source-location privacy based on multirings in wireless sensor networks. Concurrency and Computation: Practice and Experience (Online Jun 2013)
55. Yao, L., Kang, L., Shang, P., Wu, G.: Protecting the sink location privacy in wireless sensor networks. Personal and Ubiquitous Computing pp. 1–11 (2012), 10.1007/s00779-012-0539-9
56. Ying, B., Gallardo, J.R., Makrakis, D., Mouftah, H.T.: Concealing of the Sink Location in WSNs by Artificially Homogenizing Traffic Intensity. In: The First International Workshop on Security in Computers, Networking and Communications (INFOCOM Workshops). pp. 988 – 993 (April 2011)