J. Lopez, R. Rios, and J. Cuellar, "Preserving Receiver-Location Privacy in Wireless Sensor Networks", *Information Security Practice and Experience (ISPEC 2014)* vol. 8434, pp. 15-27, 2014.
http://doi.org/10.1007/978-3-319-06320-1_3
NICS Lab. Publications: https://www.nics.uma.es/publications

Preserving Receiver-Location Privacy in Wireless Sensor Networks

Javier Lopez[1], Ruben Rios[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. Wireless sensor networks (WSNs) are exposed to many different types of attacks. Among these, the most devastating attack is to compromise or destroy the base station since all communications are addressed exclusively to it. Moreover, this feature can be exploited by a passive adversary to determine the location of this critical device. This receiver-location privacy problem can be reduced by hindering traffic analysis but the adversary may still obtain location information by capturing a subset of sensor nodes in the field. This paper addresses, for the first time, these two problems together in a single solution.

Keywords: Wireless sensor networks, location privacy, traffic analysis, node capture

1 Introduction

Wireless sensor networks (WSNs) are highly distributed networks composed of two types of devices namely, the sensor nodes and the base station [1]. The sensor nodes are matchbox-sized computers which have the ability to monitor the physical phenomena occurring in their vicinity and to wirelessly communicate with devices nearby. To the contrary, the base station is a powerful device that collects all the information sensed by the sensor nodes and serves as an interface to the network. These networks are extremely versatile, making them suitable for countless application scenarios where sensor nodes are unobtrusively embedded into systems for monitoring, tracking and surveillance operations. Many of these applications are critical and thus security and privacy become essential properties.

Privacy problems in WSNs can been categorised as content-oriented or context-oriented [14]. Content-oriented privacy focuses on protecting the privacy of the packet contents. Therefore, the data to be protected may be the actual sensed data [21] or the queries issued to the network [8] by a user. Context-oriented privacy refers to the protection of the metadata associated with the measurement and transmission of data. These data include the time at which sensitive

information is collected (i.e., temporal privacy [13]) and the location of the nodes involved in the communication (i.e., location privacy [15]).

Similarly, there are two main types of location privacy problems affecting sensor networks: source- and receiver-location privacy. The former is concerned with hiding the area where a particular type of data messages are generated. This property is important in applications where these messages are related to the behaviour of individuals, business or valuable assets [12]. On the other side, receiver-location privacy solutions are focused on preventing an adversary from reaching the base station. This is essential for the survivability of the network since an adversary with physical access to this critical device may take control over the network or even render it useless by destroying it.

Preserving receiver-location privacy is especially challenging because all the traffic generated in the network is addressed to this single device using single-path multi-hop communications. This introduces obvious traffic patterns that an attacker can analyse to determine the location of the base station. Several traffic normalisation techniques [5, 20, 16] have been proposed to hinder traffic analysis but these solutions can only provide some means of protection when the attacker is passive. More sophisticated adversaries may also be able to capture a subset of sensor nodes, exploiting the fact that each node stores a routing table that contains information on the location or distance to the base station.

This paper presents a novel receiver-location privacy solution consisting of two complementary schemes that prevent the leakage of information about the location of base station in the presence of traffic analysis and node capture attacks. The first scheme uses a probabilistic approach to guide data packets to the base station and introduces controlled amounts of fake traffic to hide this flow of information. The second scheme consists of a tuneable routing table perturbation algorithm that reduces the negative effects of node capture attacks while ensuring the delivery of data to the base station. To the best of our knowledge there is no solution in the literature that considers both types of attacks to receiver-location privacy simultaneously.

The rest of this paper is organised as follows. Before describing our solution, some related works in the area are analysed in Section 2. Section 3 presents the network and attacker models together with the main assumptions adopted for the rest of this work. Section 4 first overviews the solution and then gives more details about the internals of each of the two schemes that comprise it. Next, Section 4 briefly discusses the benefits and downsides of the proposed solutions. Finally, some conclusions and future lines of research are sketched in Section 6.

2 Related Work

Receiver-location privacy solutions can be classified according to the capabilities of the adversary, which has traditionally been considered to be passive (with a local or global eavesdropping range). This classification can be further divided according to the main techniques used to counter these adversaries, as shown in Fig. 1. Since this work focuses on local adversaries, we only review these solutions

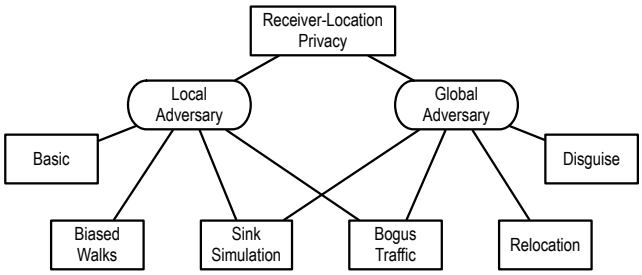


Fig. 1. Classification of Receiver-Location Privacy Solutions

but we refer the reader to [17] for a more exhaustive analysis of location privacy solutions in WSNs.

Deng et al. [6, 5] were the first to propose a set of anti-traffic analysis techniques to protect the base station. They present a set of basic countermeasures consisting of applying hop-by-hop re-encryption, de-correlating packet sending times and establishing a uniform sending rate across the network. These solutions have some limitations and the authors try to alleviate them with the Multi-Parent Routing (MPR) scheme. The idea behind this scheme is balance the traffic load while reducing the distance to the base station at each hop. To that end, each node selects the next node uniformly at random from all its neighbours closer to the base station than itself. However, this countermeasure is insufficient to prevent parent-child correlation and the authors propose to make every node decide whether to send the packet to a random parent or to a random neighbour based on some probability.

A similar approach is proposed by Jian et al. [10, 11]. They suggest to make each sensor node divide its neighbours into two groups: closer and further neighbours. Later, data packets are sent with higher probability to nodes belonging to the group of closer neighbours. This results in a traffic pattern biased towards the base station, which can be noticed by an attacker after a sufficient number of observations. This problem is reduced by injecting fake packets in the opposite direction with some probability. However, the attacker can still determine whether a packet is fake in some situations and therefore he is able to determine that the base station is on the other direction.

Some other approaches try to introduce more randomness in the communication pattern. Deng et al. [7, 5] further improve MPR by generating fake paths of messages with some probability when a node observes the transmission of data packets in their vicinity. As a result, the random path grows fake branches. The main problem of this scheme is that nodes near the base station generate much more fake branches than remote nodes. To address this problem, the authors suggest that sensor nodes may adjust their probability of generating fake traffic based on the number of packet they forward.

Another scheme based on the injection of fake traffic is devised in [19]. Data messages are sent along the shortest path to the base station and when two

of these paths converge at some point, the intersection node sends two fake packets to two fake data sinks after a given period of time. The idea is to have several points in the network that receive a similar number of packets. The main limitation is that the attacker gets a good intuition of the direction to the base station while tracing the shortest path before the first intersection point. This problem is also present in the Bidirectional Tree Scheme [3], which sends data messages along the shortest path and creates several branches of fake messages that flow into and out of the path.

Finally, sink simulation approaches try to emulate the presence of the base station at different locations. These techniques are also based on the generation of fake traffic but addressed to particular network areas. This results in areas with high volumes of traffic (i.e., hotspots) that are intended to draw the adversary away from the real base station. In [2], the base station selects the hotspots by sending some special messages to random locations which are at least h hops away from it. During data transmission, when a node receives a real packet it generates a fake message and forwards it to its closer maelstrom. Deng et al. [7, 5] also devised a similar solution but hotspots are created in a decentralised way by keeping track of the number of fake packets forwarded to each neighbour. New fake traffic is more likely to be sent to neighbours who have previously received more fake traffic. The main drawback of hotspots is the high overhead needed to deceive local adversaries.

The solution presented in this paper is an evolution of our previous work [16], which can be classified as a biased random walk solution with fake packet injection. In the new version we introduce a new mechanism capable of increasing the safety time of the base station in the presence of node capture attacks. None of the solutions in the literature have considered this problem as a threat to receiver-location privacy.

3 Problem Description

This section presents the main assumptions as well as the network model and the capabilities of the attacker adopted for the rest of this paper.

3.1 Network Model

This work considers sensor networks deployed for monitoring purposes. These networks are usually deployed in vast areas and they are composed of a large number of sensor nodes and a single base station.

We assume that the connectivity of the network is relatively high and each node knows its neighbours based on some topology discovery protocol. This allows sensor nodes to build their routing tables in such a way that the node is aware of the distance of each of its neighbours to the base station. During data transmission the node may select the next communication hop from neighbours which are one hop closer, at the same distance than itself, or one hop further

away from the base station. We denote these groups of nodes as $L^{\mathcal{C}}$, $L^{\mathcal{E}}$ and $L^{\mathcal{F}}$, respectively.

Finally, we assume that every node shares pairwise cryptographic keys with each of its neighbours, which enables hop-by-hop re-encryption as well as the identification of fake messages.

3.2 Attacker Model

The adversary is considered to be mobile and capable of performing both traffic analysis and node capture attacks. Its hearing range is limited to a portion of the network, which we represent as $\mathcal{ADV}_n$ being n the number of hops controlled by the adversary. In the literature, the adversary is usually considered to have a monitoring range similar to an ordinary sensor node (i.e., $\mathcal{ADV}_1$).

A passive adversary decides its next move based on its observations. He may choose between a time-correlation or a rate-monitoring attack. In the former, the attacker observes the sending times of a nodes and its neighbours. Since a node forwards a packet immediately after it is received, the attacker may learn which neighbours are closer to the base station. The other attack is based on the assumption that nodes near the base station have a higher forwarding rate. Therefore, the attacker observes which node in its vicinity sends more packets and moves towards it.

An active adversary is interested in capturing nodes in order to retrieve their routing tables since these contain information on the distance to the base station. There are some works in the literature [4, 18] devoted to the modelling and mitigation of node capture attacks but they are focused on the protection of the key distribution mechanisms. Some authors consider a random node capture strategy while others consider the capture of nodes in a particular area. In this work we consider that the attacker starts by capturing nodes at the edge of the network and moves according to the information he obtains. We also assume that the attacker cannot compromise all sensor nodes without being discovered. Only a fraction of the routing tables in the network can be captured.

4 Base Station Cloaking Scheme

This section presents our approach for protecting receiver-location privacy. We start by giving an overview of the two mechanisms comprising our solution and then we describe them in detail.

4.1 Overview

The devised solution consists of two complimentary schemes, a traffic normalisation scheme and a routing tables perturbation scheme. The former is intended to hinder traffic analysis attacks while the latter is used to diminish the threat of node capture attacks.

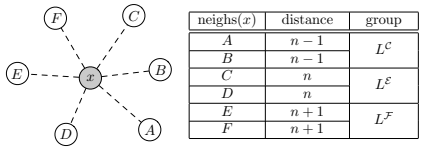
During data transmission, whenever a node has something to transmit, it sends two packets to different random nodes. One of the packets is more likely to be received by a node closer to the base station while the other packet is received by a neighbour at the same distance or further away with high probability. These probabilities are adjusted in such a way that every neighbouring node receives on average the same amount of traffic. Consequently, one of the packets can be used to carry real data and the other one as a mechanism to hide the data flow.

The second scheme complements the first one by introducing some perturbation to the routing tables. In this way, if an adversary is capable of capturing a node and retrieving its routing table he cannot be certain of which nodes in the table are closer to the base station. A parameter is introduced to control the degree of perturbation of the tables since modifying the routing tables affects the efficiency of the data transmission protocol.

4.2 Traffic Normalisation

The transmission protocol must satisfy a series of properties to ensure the security and usability of the system. First, it must guarantee the convergence of data packets to the base station. For this purpose, the expected value of the distance between the data sender and the base station must be larger than the expected value of the distance between the next node and the base station. This is a property for the usability of the system. Second, the protocol must ensure that the traffic generated by a node is evenly distributed among its neighbours. In other words, the average number of messages received by any pair of neighbours must be similar. This property is intended to locally normalise the traffic and thus make it difficult for the adversary to make a decision on its next move based on his observations. Finally, since the protocol sends pair of messages, we impose a third property to make sure that each of them is sent to a different node.

Sensor nodes always send two packets when they have something to transmit. Usually, real packets are sent out in conjunction with fake packets to hide the direction of the data flow because they are indistinguishable to an external observer. We devised a lightweight mechanism that ensures the three properties described above. This mechanism is based on the combinations without repetition of two elements from the routing table. If the routing table is sorted according to the distances of the neighbours to the base station (see Fig. 2a), we achieve that with high probability, any resulting combination has its first element from the list of closer nodes $L^{\mathcal{C}}$. Consequently, if real packets are sent to the first element of the combination the convergence property is satisfied. Additionally, if combinations are picked uniformly at random, the node balances its transmission among all its neighbours since all the elements in the routing table appear in exactly $l-1$ combinations, where l is the number of rows of the table. The combinations resulting from the routing table of node x are depicted in Fig. 2b, where we additionally count the number of combinations where the first and second element of the combination, n_1 and n_2 respectively, belong to each of the groups. From this figure it is easy to see that the probability of sending



neighs(x)	distance	group
A	$n-1$	$L^{\mathcal{C}}$
B	$n-1$	
C	n	$L^{\mathcal{E}}$
D	n	
E	$n+1$	$L^{\mathcal{F}}$
F	$n+1$	

(a) Routing table of node x

n_2 \ n_1	$L^{\mathcal{C}}$		$L^{\mathcal{E}}$		$L^{\mathcal{F}}$		
$L^{\mathcal{C}}$	AB						1
$L^{\mathcal{E}}$	AC	BC					5
	AD	BD	CD				
$L^{\mathcal{F}}$	AE	BE	CE	DE			9
	AF	BF	CF	DF	EF		
	9		5		1		

(b) Resulting combinations

Fig. 2. Neighbours Selection Process

Algorithm 1 Traffic normalisation

Input: $packet \leftarrow receive()$
Input: $combs \leftarrow combinations(sort(neighs), 2)$
Input: $FAKE\_TTL$
1: $\{n1, n2\} \leftarrow select\_random(combs)$
2: **if** $isreal(packet)$ **then**
3: $\quad send\_random(n1, packet, n2, fake(FAKE\_TTL))$
4: **else**
5: $\quad TTL \leftarrow get\_time\_to\_live(packet) - 1$
6: $\quad$ **if** $TTL > 0$ **then**
7: $\quad\quad send\_random(n1, fake(TTL), n2, fake(TTL))$
8: $\quad$ **end if**
9: **end if**

a data packet towards the base station, i.e., $\mathbb{P}(n_1 \in L^{\mathcal{C}}) = 9/15$, is much higher than in order directions.

Since fake traffic is injected in the network, its propagation must be controlled in some way to minimise its impact on the lifetime of the network. In other words, fake traffic must be dropped at some point but this action must not give location information to the adversary. The whole process is represented in Algorithm 1. Upon the reception of a packet, the node first checks whether the packet contains actual data. In case the packet is real, the node selects two neighbours using the above mentioned mechanism and generates a new fake packet that contains a time-to-live value. The data packet is sent to node n_1 and the fake packet to node n_2. If the packet received is fake, the node must behave in the same way, that is, the node sends two packets. However, now both packets are fake. The time-to-live value is decremented at each hop and if the value reaches 0 no packets are forwarded. The initial value of this parameter is globally defined by the network administrator based on the eavesdropping power of the adversary. The goal is to allow fake traffic to propagate beyond the reach of the adversary.

This protocol works correctly under the assumption of highly connected networks where the number of further neighbours do not outnumber the number of closer neighbours. In [16] we showed this conditions are met even for randomly deployed sensor networks since the number of neighbours closer to the base station is roughly the same to the number of nodes further away. Still, the speed of

convergence of data packets is affected by these values. The speed increases as the number of elements in $L^{\mathcal{C}}$ grows for the nodes in the path.

4.3 Routing Tables Perturbation

Routing tables are a fundamental component of almost any data transmission protocol. They contain relevant information regarding the location or distance to the data sink. Our traffic normalisation protocol relies on the table order[3] to determine suitable combinations of neighbours that satisfy the usability and privacy of the system. However, if an adversary gains access to the routing tables he is able to determine which nodes are closer to the base station regardless of the use of traffic analysis techniques.

We propose a routing table perturbation scheme that rearranges the elements of the table to generate some uncertainty on the adversary instead of giving him direct access to this privacy-relevant information. Since the perturbation affects the resulting combinations, it must be carefully tuned in such a way that the convergence property holds. Formally, we must ensure that $\mathbb{P}(n_1 \in L^{\mathcal{C}}) > \mathbb{P}(n_1 \in L^{\mathcal{F}})$. In other words, the routing table must continue to be biased towards the base station after the rearrangement of its elements.

The perturbation degree or bias is an important variable in this scheme because it determines both the speed of convergence of data packets to the base station and the uncertainty level of the attacker. We define the bias of a routing table r, $bias(r) \in [-1, 1]$, as the probability of sending data packets in the direction of or in the opposite direction to the base station. The closer the bias is to 1 the greater it is the probability that data packets are sent to nodes in $L^{\mathcal{C}}$. Likewise, a bias value close to -1 implies that it is highly likely that the resulting combinations have their first elements from $L^{\mathcal{F}}$. Consequently, the bias can be defined as the difference between the probability of sending the data packet to a node in $L^{\mathcal{C}}$ and the probability of sending it to $L^{\mathcal{F}}$.

$$bias(r) = \mathbb{P}(n_1 \in L^{\mathcal{C}}) - \mathbb{P}(n_1 \in L^{\mathcal{F}}) \tag{1}$$

Note that these probabilities depend on the positions of each of the elements in $L^{\mathcal{C}}$ and $L^{\mathcal{F}}$ in the routing table because the position determines the number of combinations that have a particular neighbour as the first element. For example, if we number the rows of the routing table in Fig. 2a from bottom to top, starting from 0, we can see in Fig. 2b that there are no combinations where the first element is F while A is the first element of 5 combinations. Therefore, we can generalise the probabilities in Equation 1 for any subset of elements in the table L as:

$$\mathbb{P}(n_1 \in L) = \frac{1}{C} \sum_{n \in L} pos(n) \tag{2}$$

where C is the total number of combinations resulting from the table. These equations allows us to check that $bias(r) = -1$ when the table is comprised

[3] Knowing the actual distance, as shown in Fig. 2b, is not necessary.

Algorithm 2 Perturbation Algorithm

Input: $br \leftarrow \{L^{\mathcal{C}}, L^{\mathcal{E}}, L^{\mathcal{F}}\}$
Input: $bias$, MAX_ITER
1: $E \leftarrow energy(bias, br)$
2: $i \leftarrow 0$
3: **while** $(i < MAX\_ITER) \wedge (E \neq 0)$ **do**
4: $br' \leftarrow swap(br)$
5: $E' \leftarrow energy(bias, br')$
6: **if** $(E' < E)$ **then**
7: $br \leftarrow br'$
8: $E \leftarrow E'$
9: **end if**
10: $i \leftarrow i + 1$
11: **end while**
12: **return** br

solely of elements in $L^{\mathcal{F}}$ since $C = \sum_{n \in L^{\mathcal{F}}} pos(n)$. Likewise, if all the elements in the table are closer to the base station, then $bias(r) = 1$. In general, the bias must be greater than 0 in order to enable the eventual delivery of messages to the base station.

Finding a particular arrangement of the table that satisfies a particular bias value may be time consuming and may not always be feasible. This problem is conditioned by the number of elements of each group in the original table. Therefore, our perturbation scheme (see Algorithm 2) is modelled as an optimisation algorithm that receives as input a desired bias and the routing table, and returns the closest match. Our algorithm is inspired on evolutionary strategies [9] where simple mutations are applied to the routing table in order to minimise the distance to the desired bias. More precisely, we swap two elements from the routing tables and check whether this reduces the energy (i.e., the distance to the desired solution). In case the new state of the table reduces the energy we keep this arrangement. The process is repeated for a maximum number of iterations or until the energy is zero, which means that the optimal solution is found.

The main advantage of these solutions compared to deterministic algorithms is that the time to find a (pseudo-) optimal solution to the problem is reduced. The search space may be large for very dense network configurations where the routing tables contain a large number of nodes. In these cases, the computation time may be reduced several orders of magnitude. The main downside is that the desired solution is not always found although it converges to it.

The non-deterministic nature of the algorithm provides an additional means of protection to reversing attacks. Since the algorithm does not always reach the same solution, the attacker may not be able to undo the perturbation even if he learns the bias used by the algorithm. Nonetheless, it is more secure to completely remove this value from the node after use since it is no longer necessary for the operation of the data transmission protocol.

5 Discussion

A local eavesdropper eventually finds a data source and starts analysing the traffic it generates. If the attacker chooses to perform a time-correlation attack he moves to the neighbour who forwards the first packet, which may be real or fake. If the packet is real, the attacker is highly likely to reduce its distance to the base station but the probability of following a real packet is lower than the probability of following a fake packet. The reason is that the ratio of fake to real packets is greater or equal to 1 for typical attackers. Also, note that the adversary can only be sure that he followed a fake packet when it is no longer propagated but since they flow in any direction, this provides the attacker[4] with no relevant information. If the strategy is to perform rate monitoring, the attacker moves to the neighbour receiving the larger number of packets but our transmission protocol evenly distributes the traffic, which hinders this attack.

Dealing with an attacker capable of capturing sensor nodes and retrieving its routing table is an ambitious task. First, note that making the nodes store fake routing tables provides no protection to the real table. The reason is that the node must keep a pointer to the real table and the adversary may also have access to it. Even if this information is obfuscated in some way, the adversary can identify which table is in use in various ways. For example, if the node is transmitting packets, the adversary can observe which pairs of neighbours are actually receiving messages. Also, since the routing tables are updated after each topology discovery phase the fake routing tables should take into account topology changes to prevent the attacker from easily distinguishing the real table. The fact that routing tables are periodically updated, also implies that the perturbation must be performed by all nodes. If the decision is determined probabilistically, the adversary could compromise a set of nodes and wait until the next discovery phase to check whether the routing tables changed. In view of this, the attacker could identify which tables are real. In general, keeping the real tables in its original form within the nodes is unsafe.

The main drawback of our perturbation algorithm is that it negatively impacts the performance of the network by slowing the speed of convergence of data packets. However, it is the price to pay for protecting the location of the most important device of the network. Still, the attacker has an advantage that depends on the degree of perturbation introduced to the routing table. Since the resulting table must be slightly biased towards the base station in order to allow data packets to be delivered, the adversary can reproduce the behaviour of the node and generate pairs of messages whose first element is closer to the base station with higher probability. But this is true for any system where the attacker has access to all the secrets. Nevertheless, introducing the perturbation algorithm is much better than not modifying the routing tables. In the latter case, the adversary simply needs to move always to the first neighbour in the

[4] We are assuming that the network is configured correctly and thereby the adversary does not control the whole path of fake messages.

routing table and will reach the base station within the minimum number of captures.

6 Conclusions

This work presents a novel receiver-location privacy solution that reduces the chances of an attacker reaching the base station. The devised solution consists of two complementary schemes that hinder both traffic analysis and node capture attacks. The first scheme is a traffic normalisation protocol that injects controlled amounts of fake traffic to hide the flow of real data packets. This protocol satisfies several usability and security properties that ensure the eventual delivery of data packets to the base station while it is protected from being traced. The second scheme is an evolutionary algorithm that perturbs the routing tables of the nodes to interfere with adversaries capable of gaining location information about the base station after capturing a node. The algorithm is guided by a bias value, which determines the perturbation degree of the tables. This value introduces a tradeoff between the protection against these attacks and the mean delivery time of data packets.

As future work we aim to investigate new mechanisms to reduce the overhead introduced by the traffic normalisation scheme especially when the hearing range of the adversary is large. We also want to explore and develop more sophisticated attacker models and check the robustness of our solution against them. Additionally, we are working on the design of a privacy-friendly topology discovery protocol since traditional solutions reveal the location of the base station. Our final goal is to develop an integral solution capable of protecting both from attackers interested in finding the base station and data sources.

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). The first author is supported by the Spanish Ministry of Education through the F.P.U. Program. Also, special thanks to Martín Ochoa for his valuable comments in preliminary versions of this work.

References

1. Akyildiz, I., Su, W., Sankarasubramaniam, Y., Cayirci, E.: A survey on sensor networks. IEEE Communications Magazine 40(8), 102–114 (2002)
2. 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)

3. 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)
4. Chen, X., Makki, K., Yen, K., Pissinou, N.: Node Compromise Modeling and its Applications in Sensor Networks. In: 12th IEEE Symposium on Computers and Communications (ISCC 2007). pp. 575 –582 (july 2007)
5. 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)
6. Deng, J., Han, R., Mishra, S.: Intrusion tolerance and anti-traffic analysis strategies for wireless sensor networks. In: International Conference on Dependable Systems and Networks. pp. 637–646. DSN '04, IEEE Computer Society, Los Alamitos, CA, USA (2004)
7. Deng, J., Han, R., Mishra, S.: Countermeasures Against Traffic Analysis Attacks in Wireless Sensor Networks. In: 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks (SECURECOMM '05). pp. 113–126 (2005)
8. Di Pietro, R., Viejo, A.: Location privacy and resilience in wireless sensor networks querying. Comput. Commun. 34(3), 515–523 (March 2011)
9. Eiben, A., Smith, J.: Introduction to Evolutionary Computing. Natural Computing, Springer, 2 edn. (2007)
10. Jian, Y., Chen, S., Zhang, Z., Zhang, L.: Protecting receiver-location privacy in wireless sensor networks. In: 26th IEEE International Conference on Computer Communications (INFOCOM 2007). pp. 1955–1963 (2007)
11. 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)
12. 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 (ICDCS '05). pp. 599–608 (2005)
13. Kamat, P., Xu, W., Trappe, W., Zhang, Y.: Temporal Privacy in Wireless Sensor Networks. In: 27th International Conference on Distributed Computing Systems. p. 23. ICDCS '07, IEEE Computer Society, Washington, DC, USA (2007)
14. 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 (SASN '04). pp. 88–93 (2004)
15. 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)
16. Rios, R., Cuellar, J., Lopez, J.: Robust Probabilistic Fake Packet Injection for Receiver-Location Privacy in WSN. In: S. Foresti, M.Y., Martinelli, F. (eds.) 17th European Symposium on Research in Computer Security (ESORICS 2012). LNCS, vol. 7459, pp. 163–180. Springer, Springer, Pisa, Italy (Sept 2012)
17. Rios, R., Lopez, J.: Analysis of Location Privacy Solutions in Wireless Sensor Networks. IET Communications 5, 2518 – 2532 (2011)
18. Vu, T.M., Safavi-Naini, R., Williamson, C.: Securing wireless sensor networks against large-scale node capture attacks. In: Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security. pp. 112–123. ASIACCS '10, ACM, New York, NY, USA (2010)

19. 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
20. Ying, B., Gallardo, J.R., Makrakis, D., Mouftah, H.T.: Concealing of the Sink Location in WSNs by Artificially Homogenizing Traffic Intensity. In: 1st International Workshop on Security in Computers, Networking and Communications. pp. 1005–1010 (2011)
21. Zhang, L., Zhang, H., Conti, M., Di Pietro, R., Jajodia, S., Mancini, L.: Preserving privacy against external and internal threats in WSN data aggregation. Telecommunication Systems 52(4), 2163–2176 (2013)