R. Rios, and J. Lopez, "(Un)Suitability of Anonymous Communication Systems to WSN", *IEEE Systems Journal*, vol. 7, pp. 298 - 310, 2013.
http://doi.org/10.1109/JSYST.2012.2221956
NICS Lab. Publications: https://www.nics.uma.es/publications

(Un)Suitability of Anonymous Communication Systems to WSN

Ruben Rios and Javier Lopez
Network, Information, and Computer Security (NICS) Lab
University of Malaga
29071, Spain
ruben@lcc.uma.es, jlm@lcc.uma.es

Abstract

Anonymous communication systems have been extensively studied by the research community to prevent the disclosure of sensitive information from the analysis of individuals' traffic patterns. Many remarkable solutions have been developed in this area, most of which have proven to be effective in the protection of user privacy against different types of attacks. Recently, the privacy preservation problem has also been considered in the realm of wireless sensor networks (WSNs) due to their imminent adoption in real-world scenarios. A special challenge that arises from the analysis of the flow of sensor nodes' communications is the location privacy problem. In this work we concentrate on analyzing the suitability of traditional anonymous communication systems originally designed for the Internet to the original scenario of sensor networks. The results show that, in most cases, traditional solutions do not provide the adequate protection means for the particular problem of location privacy, while other solutions are too resource-consuming for the restricted capabilities of sensor nodes.

Keywords: *Anonymous communication, location privacy, wireless sensor networks.*

1 Introduction

Wireless sensor networks (WSNs) are ad hoc networks composed of inexpensive, autonomous computers which are able to sense the phenomena in their surroundings and transmit this information by means of a wireless antenna. These tiny computers, called sensor nodes, are battery-powered and hardware-constrained, what requires them to cooperate in order to relay the sensed information to a centralized, resourceful device called the base station or sink. By collecting and processing the received data, the base station obtains a clear picture of the environment being monitored.

WSNs are extremely useful in very diverse scenarios given the different types of sensors (e.g., temperature, radiation, vibration, etc.) that might be coupled to sensor nodes. The range of scenarios where this technology is of paramount importance range from military to civilian applications, including environmental monitoring, industrial automation, cargo tracking, and many other. Doubtlessly, the integration of such context-aware technologies will bring tremendous benefits by easing and reducing costs of the management and control of the application scenarios. Nonetheless, there are many security challenges [32] that threaten to impede a successful deployment of the technology under consideration.

Among the various threats to WSNs, privacy is of outermost importance, specially when sensitive information about individuals or corporations can be gathered or inferred from the monitoring system. In the particular case of WSNs two potential privacy threats are likely to appear. On the one hand, the first and most obvious privacy risk is due to the ability of sensor

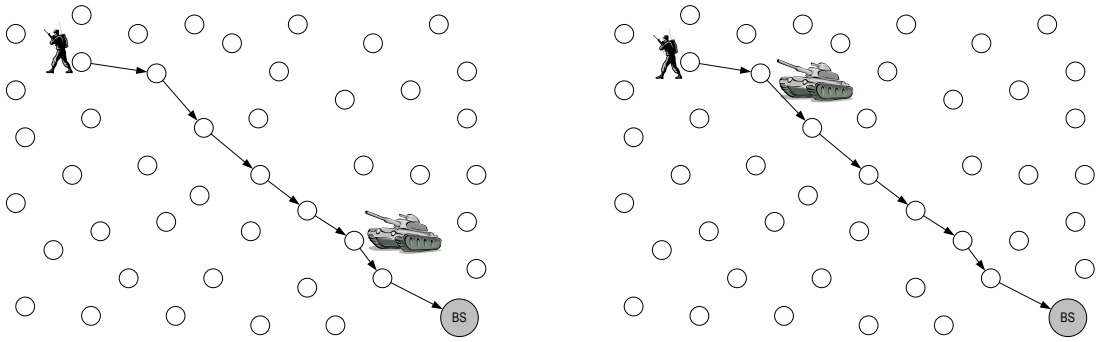

Figure 1: Location Privacy Problem in a Military Scenario

nodes to inadvertently collect information. In this case the privacy invader is the (owner of the) network and the subjects or business being monitored without explicit consent are the victims. On the other hand, the analysis of the communications of the network pose context privacy risks. Context privacy is concerned with the protection of the circumstances surrounding the sensor application [14]. In other words, an adversary might obtain information about the network itself (e.g., type of sensor nodes) and about the environment being monitored (e.g., amount or nature of events[1]).

In particular, the location of the events being monitored by the network is highly valuable information for potential adversaries. The events depend on the application and thus the importance of protecting their location is determined by the criticality of the scenarios. Specially critical scenarios are those involving individuals and valuable assets. Consider, for example, the military scenario depicted in Figure 1, where a sensor network is deployed to monitor both troops and assets (e.g., armaments, tanks, etc.) belonging to a military force. The collected information is sent to the base station in real time for a better military coordination and control. However, the communications generated by the network might be exploited by the enemy to uncover the location of targets. Similarly, the enemy could deduce the location of the base station and thus attack the headquarters. The essentials of the location privacy problem are shown in the depicted scenario, but it is clearly extensible to any other application domains. Actually, and as already stated, the location privacy problem appears because the adversary monitors the communications, and from his observations he is able to determine which nodes are reporting event data and which other nodes are receiving these data.

Anonymous communication systems were devised to preserve users' privacy while communicating on the Internet. The countermeasures proposed by these systems are mainly focused on preventing traffic analysis attacks. Consequently, and in principle, these systems, which were originally proposed for providing anonymity in Internet scenarios, might appear as a viable solution to prevent the disclosure of location information in WSNs. Notwithstanding, the existing literature on WSNs [23, 17, 19, 33] establishes that such systems are not applicable to the sensors domain. We consider that arguments presented in those papers are too vague, mainly focusing on the prohibitive resource consumption for the hardware constrains of sensor nodes. Moreover, we do believe that a more in-depth analysis is necessary before precluding the traditional anonymous communication systems, specially given the maturity of research in this field. As a matter of fact, we think that a strict analysis of the requirements, goals and techniques proposed by these systems, as well as the new requirements and special features in sensor networks, will pave the way for the design and development of solutions that are specific

[1] An event can be defined as the occurrence of a particular phenomenon in the vicinity of a node. For example, in an industrial process an event occurs when the pressure of a valve is higher than a defined threshold, or in a habitat monitoring scenario, an event is that an endangered animal is detected in the surroundings of a node.

for WSN scenarios and that outperform those already existing in the literature.

Consequently, the aim of this work is to provide a rigorous analysis of the requirements, objectives and countermeasures proposed by traditional anonymous communication systems in order to determine the suitability of these solutions to the location privacy problem in WSNs. To that end, the rest of this paper is organized as follows. Section 2 provides background on the design goals of anonymous communication systems and proposes a categorization of these solutions that will guide us during the analysis. Next, in Section 3 we provide a precise definition of the location privacy problem in WSNs as well as the adversarial model and special features of this domain. The insights obtained will allow us to determine which anonymity properties better suit in the protection of location privacy. Subsequently, Section 4 and Section 5 respectively analyze centralized and decentralized anonymous communication systems to have a clearer picture of the overhead introduced by these solutions. In Section 6 we briefly discuss on the factors that limit the application of the previously analyzed solutions to sensor networks. Finally, Section 7 concludes the paper.

2 Anonymous Communication Systems

The present section provides a general overview of the main properties pursued by anonymity solutions. A complete understanding of these properties is essential for the analysis of any anonymity system. Furthermore, this section presents a categorization of traditional anonymous communication systems devised for the Internet that will conduct later discussions.

2.1 Background on Anonymity Properties

The concept of privacy is usually difficult to define mainly due to the subjectivity of the term. Privacy has different interpretations depending on cultural aspects, social condition, and many other factors [3]. A simple definition considers privacy as the right to be left alone [34], however, new definitions have appeared as new ways of invading privacy emerged. With this respect, technology evolution has played a key role in the transformation of the concept because it introduces new means of gathering and disseminating sensitive information. Likewise, different tools and mechanisms have been devised in order to provide various properties [25] that we describe in the following.

Anonymity can be defined as the state of being not sufficiently identifiable within a set of subjects (i.e., the anonymity set) with potentially the same attributes as the original subject. In other words, anonymity mechanisms aim to prevent the disclosure of the identity of the individual who performed an action by having a set of other entities which might have potentially performed that action. In the landscape of anonymous communication systems, the action usually refers to the transmission or reception of messages. Therefore a sender may be anonymous within a set of potential senders. Similarly, a recipient may be anonymous within a set of potential recipients. These properties are known as sender and receiver anonymity, respectively.

Another important property for the protection of subjects' privacy while communicating on the Internet is *unlinkability*. Unlinkability of two or more items of interest means that the adversary cannot sufficiently distinguish whether these items are related or not. By definition, the items of interest might be any element of the system, such as entities or messages. Nonetheless, anonymous communication systems usually strive for the unlinkability of the sender and receiver, which provides the communicating parties the ability to hide with whom they communicate. More precisely, relationship unlinkability suggests that even when the sender and the receiver can each be identified as participants in some communication, they cannot be rec-

	Main goal	Architecture	Techniques							
			SK	PK	LE	RN	PD	PR	FT	MB
Single-proxy [2]	Sender Anonymity	Centralized	√	-	-	√	-	-	-	-
Mixes [7]	Unlinkability		-	√	√	-	√	-	-	-
Onion routing [27]			√	√	√	-	-	-	√	-
Tor [10]			√	√	√	-	-	-	-	-
Crowds [28]	Sender Anonymity	Decentralized	√	-	-	√	-	-	-	-
Hordes [15]			√	√	-	√	-	-	-	√
GAP [4]	Unobservability		√	√	-	√	√	√	√	-
DC-nets [8]			√	-	-	-	-	-	-	√
Herbivore [11]			√	√	-	-	-	-	-	√

Notation	
SK	Symmetric-key encryption/decryption
PK	Public-key encryption/decryption
LE	Layered encryption
RN	Source identity renaming
PD	Temporal packet delay
PR	Packet replay
FT	Fake traffic injection
MB	Multicast or broadcast communications

Table 1: Classification of Anonymous Communication Systems

ognized as communicating with each other. This suggests that the unlinkability property is stronger than anonymity.

Finally, undetectability and unobservability are properties that aim to protect the items of interest per se. *Undetectability* of an item of interest means that the attacker cannot sufficiently determine whether this item exists or not. Similarly, *unobservability* means undetectability of the item against all external entities and, additionally, anonymity of the subjects even against other subjects involved in the item of interest. In anonymous communication systems, the cited properties usually refer to messages as the objects of interest. Therefore, undetectability aims to prevent an adversary from determining whether real messages are being transmitted. Additionally, unobservability implies that even when a subject is able to detect the existence of a message, the parties involved in the communication remain anonymous. A sender is unobservable when the attacker is not able to determine whether any of the senders is transmitting real messages. Likewise, the recipient is unobservable if the adversary cannot conclude whether it is receiving real data messages.

2.2 Classification of Traditional Solutions

Many outstanding anonymous communication systems have been devised to hinder traffic analysis and thus improve the privacy protection of Internet users. These systems were designed with different goals in mind and thus pursue different anonymity properties. We propose a taxonomy of solutions which takes into consideration three major features, namely: (1) the main objective pursued in the design of the anonymous communications system, (2) the architecture of the system, and (3) the main techniques used to achieve the design goals. This taxonomy is presented in Table 1. For the sake of simplicity only the most commonly used techniques have been represented in this table.

Among the numerous anonymous communication systems available, we have selected various outstanding solutions that introduce distinguishing features so that a wide range of techniques and countermeasures are addressed for different adversarial models. From an architectural point of view these solutions can be categorized as either centralized or decentralized. Centralized

solutions are those in which the parties involved in the communication are not an active part of the anonymity system. Data senders communicate with their corresponding recipients by means of a set of devices (i.e., the network core) that forward the messages on their behalf. Contrarily, in decentralized systems every user collaborate in the forwarding process to conceal his own communications and the communications of other participants. Some of these solutions are partially decentralized because they rely on a central server who is in charge of providing all the information necessary to communicate with other members or external entities, while other solutions are fully decentralized and require no central authority. Also, in these solutions, data recipients might be part of the network or external entities.

The proposed categorization also takes into consideration the main goals pursued by these solutions. It is worth mentioning that some of these solutions might have several objectives but only the most relevant are considered in the table for simplicity reasons. For example, mix-net approaches aim to provide sender-receiver unlinkability but in fact they might also provide sender anonymity. Note that when we refer to sender anonymity, we usually refer to anonymity with respect to the data recipient. In many situations users are willing to access services but are reluctant to provide their real identity to potentially distrustful service providers because they fear being tracked or profiled for illegitimate purposes.

Finally, the various techniques employed by these solutions could be used to further divide into new categories. For example, the presented solutions could be organized into high-latency or low-latency solutions depending on the use of temporal packet delays. Instead of increasing the complexity and reducing the readability of this table we decide to list the most common techniques and mark them as used or unused. Further details about these and other techniques are provided in subsequent sections. These sections make use of the architectural perspective (i.e., centralized or decentralized) to organize the exposition and analysis of the solutions under consideration.

3 Location Privacy in WSNs

Prior to the analysis of traditional solutions, this section aims to provide a more precise description of the location privacy problem in the specific scenario of WSNs. Also, it describes the capabilities of the adversarial model in this scenario as well as the network features that cause the problem. Moreover, we discuss on the suitability of the anonymity properties (in Section 2.1) to deal with the location privacy issues in WSNs.

3.1 Problem Overview

Sensor networks are deployed in many diverse application scenarios (e.g., habitat monitoring, healthcare, industrial plants, battlefields) where they are used to monitor and control the phenomena occurring in their vicinity. In these scenarios the information collected by the network is used for a better management and response to anomalous situations. Given the importance of these data, unauthorized entities should not be able to access them and hence sensed data is cryptographically protected before being transmitted to the base station. However, even when data confidentiality mechanisms are in use to conceal the collected data, the mere observation of the operation of the network can reveal important contextual information to an attacker.

By performing traffic analysis attacks an adversary can gain information about the network and the environment being monitored [24]. The type of information that can be obtained depends on the traffic features being analyzed. For example, the observation of the frequency of the wireless signals transmitted by the nodes might disclose information about the sensor platform as well as the potential owner of the network. Similarly, by monitoring the number

and size of the packets being transmitted, the adversary deduces certain features of the sensed data and he potentially infers the purpose of the network. Finally, the routing protocols provide information about the network topology and, consequently, indicates the location of the nodes sending and receiving event data.

The importance of location information in WSNs is twofold. Sensor nodes reporting event data reveal the location of the phenomena being monitored. For example, in tracking applications the adversary can learn the location of the object, animal or individual being tracked and use this information for different purposes, which range from reputation damage to physical violence or theft. In other types of applications, such as control systems, by observing the traffic generated by the network the adversary can learn corporate secrets like the structure of the plant or the behavior of industrial processes. Additionally, protecting the location of the base station is of paramount importance for the operation of the network. The base station is the most important node of the network because it collects and analyses all the data transmitted by the sensor nodes. A malfunctioning or compromised base station renders the whole sensor network unusable. Nonetheless, concealing the location of the base station is not only a matter of physical protection but also a privacy preservation issue because this critical device is usually placed in a strategic location.

3.2 Threat Model

The wireless nature of the communications in WSNs gives an external observer the opportunity to perform traffic analysis attacks without much effort. Two main types of adversaries are considered in the literature, the local and global observer, which basically differ in their range of action.

A local attacker has a limited hearing range, similar to that of an ordinary sensor node, and thus he can only monitor the transmissions of nearby nodes. This type of adversary is equipped with a device (e.g., a directional antenna) that allows him to determine the angle of arrival and signal strength of the packets in his eavesdropping range. Indeed, these techniques are commonly used for localization purposes in sensor networks [37]. Moreover, since event data usually follows regular paths in order to minimize the energy consumption due to the wireless transmissions, the adversary can trace back received packets to the data source. The strategy of a local adversary is to move in the direction of the received packet. Once the immediate data sender is reached, the adversary waits until the next packet is received in order to repeat the operation. In this way, the adversary eventually finds the real data source, where the event is taking place. This process is depicted in Figure 1. Similarly, a local observer might determine the location of the base station by monitoring the time correlation of transmissions between neighboring nodes. When a sensor node sends a message and immediately after a neighboring node forwards it, this is a clear indication of the direction towards the base station. Also, nodes in the proximities of the base station have a higher transmission rate because they send their own data and also forward data from remote nodes, thus the adversary can move in the direction of nodes with higher transmission rates in order to reach the base station.

A global adversary is more powerful in the sense that he is capable of observing all the traffic transmitted by every single sensor node in the network. Upon the detection of an event, the detecting sensor node generates and sends a new message to the base station in order to inform about the existence of such phenomenon. Consequently, a global adversary can easily determine the location of events by simply monitoring which nodes are generating new traffic. This issue can be easily determined by comparing the number of incoming and outgoing packets in a node. Moreover, a global adversary can also determine the location of the base station without moving in the field. Since all communications are addressed to a single base station,

the traffic rate in the vicinity of the base station is significantly higher than in other areas. Therefore, a global observer obtains a very good estimation of the location of the base station by simply monitoring the number of packets being transmitted by the sensor nodes.

Additionally, some adversaries might also compromise a small portion of the nodes in order to obtain the location of source nodes. These compromised nodes are known as internal adversaries since they are ordinary sensor nodes which are collaborating in the routing process. As ordinary nodes they are aware of the shared secrets between neighboring nodes and they will take advantage of this knowledge to observe the contents of data packets. Having access to the packet contents may allow en route nodes to retrieve the original data sender because this information must be contained somewhere in the packets to allow the base station recognize the data source.

3.3 Anonymity Properties in WSNs

Among the various sensitive information that might be gathered by an observer, we concentrate on the location of the nodes reporting or receiving data. The reason for considering these particular problems is that other data, such as the frequency of the wireless signals, cannot be protected by means of anti-traffic analysis mechanisms. These mechanisms are aimed to hide the network traffic patterns in order to prevent the disclosure of the communicating parties. Hindering traffic analysis is the main focus of anonymous communication systems. Therefore, these systems are, in principle, also suitable for protecting the location of the source nodes and the base station in WSNs. However, there are several limitations to the application of traditional solutions to the new domain. Here we concentrate on determining which design goals of traditional anonymity systems are significant in the protection of location privacy in sensor networks.

Firstly, anonymity is only necessary under certain circumstances in WSNs, being even detrimental for the proper operation of the network in some cases. Source anonymity with respect to the recipient is not beneficial for the operation of the network because in most application scenarios the base station (i.e., the recipient) needs to be aware of the original data sender. The base station uses the source ID for the management and control of the environment being monitored. Without this information, the base station cannot identify the origin of the data and thus it is unable to provide the administrator of the network with relevant information about the sensor field. Besides, sender anonymity might be useful to prevent external observers from determining the data source. Hiding this information helps in the protection of the location of events against adversaries who have created a map of the network by patently eavesdropping on every single network node. This problem can be prevented by occasionally changing the nodes IDs so that even if the attacker obtains such a map, it renders useless when the current IDs change their values. Several works already exist on the use of dynamically changing pseudonyms for WSNs [18, 22]. In this situation, the base station is still aware of the pseudonyms of every node and thus is able to identify the occurrence of events in the field. Finally, it might be interesting to prevent compromised sensor nodes (i.e., nodes controlled by the attacker) to obtain the source ID. Also some works have concentrated on this problem [26]. Therefore, source anonymity is only necessary under certain circumstances in WSNs.

Besides, given the existing communications model in sensor networks, the sender-receiver unlinkability property does not make much sense. The normal operation of the network implies many-to-one communications, where any sensor node is a potential sender and the base station is the only receiver. Therefore, the property of relationship unlinkability is lost because in any event transmission the base station is one of the participants. Particularly, in anonymous communication systems, relationship unlinkability is important in terms of the identity of the

sender and the recipient because it gives away information about the behavior and preferences of users. In the case of location privacy in WSNs, all sensor nodes transmit to a single base station so there is no such information gain. The important issue is to determine the location of these nodes and this cannot be done by simply analyzing packets in transit unless this information is given either in the headers or the payload. It is assumed that the attacker has no access to the payload because it is cryptographically protected, but the header might provide information on the source. This issue becomes problematic only in case the adversary already knows the network topology but, as aforementioned, this problem is related to source anonymity not to unlinkability.

In fact, the most natural property for the protection of location privacy is unobservability rather than unlinkability. In WSNs it is necessary to hide the existence of the nodes reporting event data or receiving it. More precisely, the attacker will be unable to determine the location of the communicating nodes if he is unable to sufficiently detect the presence of data messages in the network. Clearly, 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 under the assumption that he has no other information than the observed traffic[2].

In brief, we can state that some anonymity properties are not suitable for the protection of location privacy in WSNs. In any case, the following sections will delve into every particular solution in order to have a clearer understanding of the particular features, the overhead introduced and the techniques proposed by anonymous communication systems originally devised for the Internet.

3.4 Location Privacy Solutions in WSNs

The location privacy problem in WSNs has been mainly tackled by introducing route diversification and fake traffic. These techniques have been used for both the source- and receiver-location privacy problems. Nonetheless, most research has concentrated on the protection of source nodes while concealing the location of the base station has received less attention so far.

Typically, when the adversary has a local hearing range, proposed solutions use random routing protocols so that every packet follows a different path to the base station. This strategy has been extensively used for hiding both source nodes [14, 30] and the base station [9, 13]. Also, the creation of network loops [21, 16] and fake data sources [23, 12] have been used to deviate a mobile adversary from real data paths. Similarly, the authors in [9, 6] propose various methods to create areas of high communication activity in such a way that the adversary believes that he is approaching the base station.

Previous techniques are usually ineffective against adversaries with global eavesdropping capabilities. These adversaries are mainly countered by introducing great amounts of fake traffic [17] which are aimed to hide the presence of real traffic. However, to maximize the lifetime of sensors' batteries the development of new solutions was required. Simulating the presence of events in the field [17, 20], the filtering of bogus traffic [35], and even sophisticated statistical approaches [31, 1] were some of the proposed solutions when protecting the source nodes. Similarly, the injection of fake traffic was used for the receiver-location privacy problem. For example, the authors in [36] proposed to globally homogenize the transmission rate of all sensor nodes regardless of their distance to the base station.

Finally, some research has been conducted to prevent the disclosure of data sources to internal adversaries. Some contributions have been done in the creation of dynamic pseudonyms [18] and en-route identifier modifications [26]. Also, in [29] a trust-based routing approach was

[2]The attacker can benefit from other sources of information, such as visual recognition of the event, to increase his knowledge.

devised to exclude untrustworthy nodes from the data forwarding process.

4 Centralized Anonymous Communication Systems

Centralized anonymous communications systems rely on a set of (partially) trusted devices which are responsible for conveying data from senders to receivers in a privacy-preserving manner. Whenever a user wants to send data anonymously to another party, it does so by contacting any of the devices comprising the anonymity network, which eventually forwards the received data to the final destination on behalf of the user, thus obscuring the real data origin. In short, the parties involved in the communication are not members but clients of centralized anonymity systems. Consequently, there is nothing these solutions can do to protect the location of sender and receivers against global adversaries. Throughout this section only local observers and internal attackers will be considered.

4.1 Single-proxy

Several single-proxy solutions (e.g., Anonymizer [2]) have been proposed to allow Internet users to preserve anonymity when issuing requests to websites or other entities, that is, they provide sender anonymity. These solutions are based on a third-party which acts as intermediary between the user and the real destination. The operation is very simple. First, the user connects to the proxy informing about the intended recipient of the message and then the proxy forwards the user request to the server. Finally, the server responds to the proxy, which sends the reply back to the user[3] (see Figure 2). In this way, the proxy appears as the original data sender to the recipient, thus hiding the identity of the actual sender to the destination.

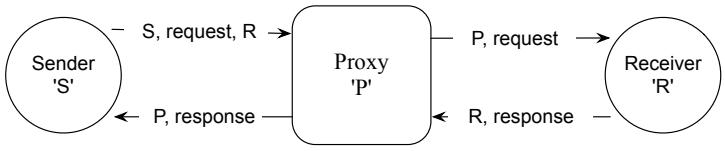

Figure 2: Single-proxy Communications

Single-proxy solutions introduce relatively low computational overhead to source nodes, which are only required to encrypt data messages by means of a key shared with the proxy. On the other hand, nodes serving as proxies decrypt the received data and change the source ID of incoming packets for their own (i.e., rename). This process is repeated at proxy nodes for every data message they receive. Additionally, source nodes can perform end-to-end encryption to conceal event data from the proxy to the base station. In any case, cryptographic operations over event data are only necessary in case the messages contain information that may lead an adversary to the data source. Table 2 summarizes the total number of operations that network nodes will need to perform depending on the scenario. The terminology used in this and subsequent tables corresponds to that described in Table 1. Also, note that these values are with respect to a single sent or received message.

Despite the low overhead introduced, given the threat model described in Section 3.2, the single-proxy approach on its own is not suitable for the location privacy problem. The reason is that a local adversary moves in the direction of received packets regardless of its contents. To prevent this, it is necessary that the attacker cannot capture all the traffic. Phantom Routing

[3]To prevent eavesdropping some single-proxy solutions encrypt communications between the sender and the proxy.

Node	Case	
	Best	Worst
Sources	–	$2SK$
Proxies	$1RN$	$1RN + 1SK$
Sink	–	$1SK$

Table 2: Single-proxy Overhead

[14] achieves this by making source nodes to transmit every data packet on a different paths so that, after several hops, they reach a random intermediate node from which the packets are finally forwarded to the base station. In this sense, this solution is similar to a single-proxy scheme but the actual protection mechanism does not reside on the renaming or the cryptographic operations but on the random intermediate node selection.

4.2 Mixes

The mix is a store-and-forward device that receives public-key encrypted messages and after a sufficiently large time period, outputs a re-ordered batch of messages. In this way, mixes hide the correspondence between inputs and outputs because of temporal storage and decryption of messages. This type of high-latency anonymity solutions were originally devised by D. Chaum [7] for non-interactive communications over the Internet, such as anonymous e-mail transmissions. Usually, mixes are deployed and selected in series (i.e., mix-cascade) or in random order (i.e., mix-net). Figure 3 depicts a series of mixes, where messages M_1, M_2, and M_3 are iteratively encrypted with the public keys of the mixes in reverse order. Every mix decrypts its own encryption layer and outputs a reordered batch of messages after a given delay. In such arrangements a single honest mix preserves the unlinkability between inputs and outputs in the whole path.

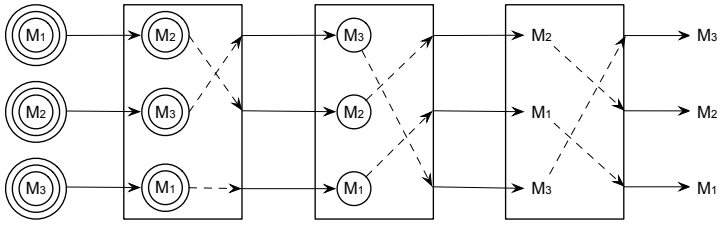

Figure 3: A Mix Cascade

The implementation of Mix models over WSNs present various limitations in terms of the computational overhead introduced. Every source node is required to perform N public-key operations per transmitted packet[4]. Additionally, source nodes are required to have a global network knowledge to be able to determine the transmission paths and perform the public-key encryptions in the right order. Moreover, this means that source nodes must store the public keys (P_K) of all potential mix nodes. Clearly, the size of the network has a tremendous impact on the number of operations and the amount of memory required to store all the necessary information to satisfy the Mix model.

Besides, every intermediate node has to perform 1 public-key decryption per received packet as well as temporarily store a number of messages (T) that depends on the number of events in the field. The amount of memory mounted on a typical sensor node is insufficient to accommodate a large number of messages. Also, after decryption is performed over the received

[4]An extra operation satifies the end-to-end encryption principle. This might be either a public-key or symmetric-key operation

packets these must be padded (PAD) to preserve message indistinguishability, otherwise the adversary would easily determine the message flow direction by monitoring the size of packets. Furthermore, the closer the nodes are to the base station the higher the amount of traffic they receive. Finally, many WSN applications require real-time monitoring capabilities but mixes introduces significant delays at every node thus precluding their use under such scenarios.

A summary of results is provided in Table 3, where for the sake of simplicity only the best case scenario is represented. Note that the worst case (i.e., source nodes perform end-to-end encryption) implies that, for every message transmitted, a source node performs an extra cryptographic operation and, moreover, the base station must share keys with all potential source nodes. Furthermore, we do not consider scenarios where the destination responds to the source. In such cases, the base station would perform the same number of operations as a source node and mix nodes would have to perform roughly the same number of operations as in the forward path. Additional terminology appears in this table: $PERM$ refers to permutation, P_K is public key, N is the number of nodes in a path, and M is the number of paths. Recall that the processing requirements are for every single send or received message.

Node	**Requirements**	
	CPU	RAM
Sources	$N * PK$	$N * M * P_K + topology$
Mixes	$1PK + 1PAD + 1PERM$	$1P_K + T * messages$

Table 3: Mixes Overhead

In addition to the high computational and memory conditions imposed by these schemes, there are other considerations that impede the successful deployment of the Mixes given the types of adversaries considered in WSNs. The main aspect is that the adversary wins if he is able to obtain the location of either the source node or the base station, contrarily to the goal of the adversary in the original scenario where he wants to determine if a particular sender is communicating with a particular recipient. In such scenarios the temporal mix of messages provides the desired property. Nonetheless, in sensor networks it makes no difference whether the adversary reaches one source node or another. The important issue is to reach any of the sources because all of them lead him to an event. Therefore, if the adversary is able to reach the entry point of the mix-cascade he will eventually receive packets from the source node, thus revealing its location. The same applies to the exit point of the mix network and the protection of the base station. These problems might be diminished in the particular case of mix-nets because of the random route selection for every message. Finally, it is worth mentioning that internal adversaries are successfully prevented from determining the source node and the base station unless they are precisely the entry and exit nodes of the mix network, respectively. Any other mix node cannot determine who is the original sender because of the use of layered encryption.

4.3 Onion Routing and Tor

Onion routing [27] is a low-latency anonymous communication system based on a network core composed of onion routers, whose functionality is similar to Chaum's mixes. The main difference between onion routers and mixes is that mixes introduce a significant delay to messages while onion routers provide near real-time multiplexing of several connections in a single data stream. Also, onion routers are connection-oriented, which means that once an anonymous connections is established, that path remains unchanged for a given period of time. In onion routing, anonymous connections (i.e., the data paths) are established by means of a public-key layered data structure, called the onion, which provides the onion routers with the cryptographic material

and the direction of the data traversing that connection. In Tor[10], the second-generation onion routing, the circuit is established incrementally, i.e. node by node, instead of by means of onions. Once the anonymous path is established, application data is repeatedly encrypted with the symmetric keys provided during the path establishment process to each of the onion routers in the circuit. The receiving onion router removes the outer layer of encryption and sends the message to the next hop. An illustration of the architecture and transmission process is provided in Figure 4.

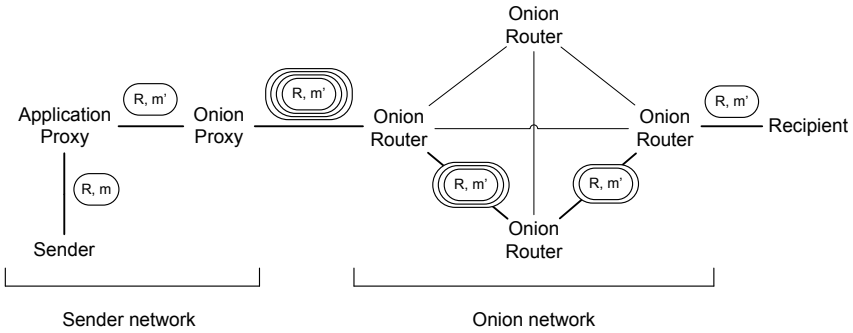

Figure 4: Data Transmission with Onion Routing

These approaches reduce the overhead compared with the Mixes scheme mainly for two reasons: data encryption and decryption is not based on public-key cryptography and the core nodes are not required to temporarily store messages. In this sense, onion routing enables near-real time communications. Nonetheless, the computational and memory requirements are still heavy for sensor nodes. Particularly, source nodes are required to be aware of the network topology as well as the public-keys of every onion router so that they can establish the anonymous path. Moreover, if the path is set by means of an onion, the source must perform several layers of public-key encryption containing the key seed material for the onion node and the next hop in the path. In the case of incremental path establishment, it implies that the source must contact the onion nodes one by one to make authenticated handshakes. This implies even more energy consumption because it requires the exchange of many messages. Once the path is established the data messages must be repetitively encrypted with the symmetric keys agreed on during the anonymous path setup.

Besides taking part in the path establishment process, packet decryption and packet relay, onion nodes keep symmetric encrypted links with neighboring onion nodes so that received data can be multiplexed and conveyed into fixed size packets. In an attempt to further complicate traffic analysis, packet padding and reordering is introduced by onion routing but Tor dismisses the idea because of the level of required resources.

In Table 4 we summarize the computational and memory demands of these schemes. The table considers both the path setup process, which only occurs occasionally, and the transmission period. We place between parenthesis some operations which are only performed by onion routing but not by Tor. Extra terminology is defined: S_K is session key, LE and L_K are link encryption and link key, R is the number of neighbors an onion node shares links with, and S is the number of sessions an onion router handles at every moment. Again, we consider the simplest case where the source does not use end-to-end encryption and the sink does not send responses back to the sources.

These schemes can be regarded as an evolution of the mix-nets approach in the sense that they reduce some of the tight requirements imposed by the original mix design. Despite the overhead reduction, onion routing solutions still present the same limitations with respect to the capabilities of the adversarial model considered in WSNs. The main drawback is that a

Node	Requirements		
	CPU		RAM
	Path setup	Transmission	
Sources	$N * PK$	$N * SK$	$N * M * P_K + N * S_K$ $+topology$
Onion nodes	$1PK$	$1SK + LE$ $(+PAD + PERM)$	$P_K + R * L_K + S * S_K$

Table 4: Onion Routing Schemes Overhead

local adversary will eventually identify the edges of the onion network. This issue allows him to identify the source nodes and the base station if messages follow similar or fixed routes to reach and leave the onion network. Therefore, the best strategy for an adversary is to reach entry or exit nodes and wait for messages to arrive. In general, we can state that the edges of the onion network are the most critical points, also if the adversary is capable of compromising nodes.

5 Decentralized Anonymous Communication Systems

Contrarily to centralized solutions where the communicating parties are not part of the anonymity network, in the solutions considered in this section all members collaborate to conceal the identities of other participants. In this way, there is more cohesion in the network, that is, it is not trivial to identify the communicating parties from mere intermediaries. However, the elimination of a semi-trusted network core introduces new challenges. Still, some of these solutions are only partially decentralized because they rely on a central server, which is in charge of providing all the information necessary to communicate with other participants.

5.1 Crowds and Hordes

Crowds [28] is a partially decentralized solution where a set of geographically diverse users are grouped, and cooperate to issue requests on behalf of its members. Whenever a crowd member (i.e., a jondo) wants to send a message, he chooses a random jondo, possibly itself, as intermediary. The receiving jondo decides, based on some probability, whether to forward the data to another jondo or to finally submit it to the destination. Figure 5 illustrates an example where $jondo1$ issues a request to $Server2$ that is finally submitted by $jondo5$, and also $Server1$ receives a message from $jondo3$ that was originally issued by $jondo5$. Hordes [15] is based on the Crowds model but its main contribution is the incorporation of multicast messages to reduce the latency and overhead on the return paths, that is, from recipients to initiators. Additionally, it uses public-key cryptography to obtain the session keys to be later used for message relay.

The Crowds model presents a low overhead in comparison with other solutions. Instead of requiring computationally heavy mechanisms such as public-key operations, dummy traffic or padding, the Crowds is based on symmetric-key packet re-encryption, sender ID renaming, and random nodes selection[5]. Consequently, any intermediate jondo is only aware of the previous and next hop in the path and, from the receiver's perspective, the message is equally likely to have originated from any crowd member.

Every member must perform one decryption and one encryption for every packet it forwards within the crowd, but if he decides to submit the packet he only needs to decrypt and forward it. In order to perform these operations, crowd members must share keys with any other member. Therefore, the number of keys every node must store is dependent on the size of

[5]Additionally, the user might establish end-to-end encrypted channels to prevent en route eavesdropping.

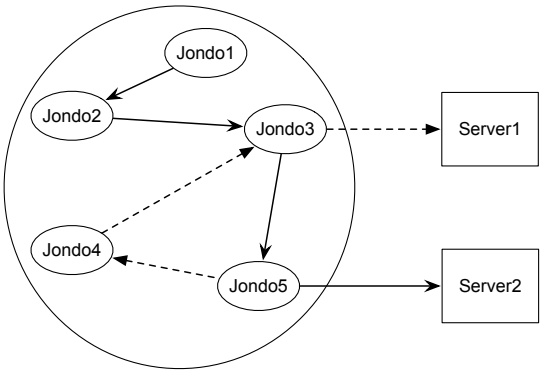

Figure 5: Communications in the Crowd.

the network. Also, for every received message the node changes the sender ID for its own and assigns an identifier to keep track of messages belonging to that path. Sensor nodes must keep a translation table with as many records as the number of paths the node handles because all subsequent packets from this connection will follow the same path.

Table 5 represents the number of operations and memory consumption which introduces the Crowds model to nodes with different roles in the network. Note that even when these roles are separated in the table, a node might have several roles at the same time. A similar table could be constructed for Hordes since we are not considering the communications on the return path. More precisely, in Hordes every participant holds the public key of the server which is used to obtain a signed list of all other members and their public keys. Then every participant chooses a subset of jondos to forward messages and send them a symmetric key encrypted with his own private key. In this way, Hordes not only requires the storage of all participants' keys but also exchange and storage of session keys, which implies more computational operations and more memory usage. For simplicity we provide a single table corresponding to the Crowds solution. In this table, R represents the number of records in the translation table of an intermediate node.

Node	**Requirements**	
	CPU	RAM
Initial	$1SK$	$N * S_K$
Intermediate	$2SK + 1RN$	$N * S_K + R * paths$
Final	$1SK + 1RN$	$N * S_K$

Table 5: Crowds Overhead

In general, the Crowds model imposes relatively low computational and storage needs precisely because of the adversarial model under consideration. This model provides a sufficient protection level against local adversaries which are able to observe the inputs and outputs of a single node but is considered to be static because of the geographically dispersion of the crowd members. This feature makes a big difference during data transmission. The Crowds model considers a random but fixed path for all communications with a given entity, however this possess a serious risk in the WSN scenario where the adversary can move towards the immediate sender of a packet. Identically, by performing time-correlation attacks the adversary could determine the next hop in the path until he finally determines the location of the sink. Besides, internal adversaries are partially countered by means of source renaming at every hop but the main drawback is that renaming also prevents the base station from learning the actual data source unless cryptographically specified in the packet payload. Finally, this model provides

no protection mechanisms against global adversaries, which can easily spot the data sources because they transmit regardless of whether they receive messages. Similarly, the base station can be easily detected because all transactions must be forwarded by nodes in its vicinity.

5.2 GNUnet Anonymity Protocol

The GNUnet Anonymity Protocol (GAP) [4] was originally devised to provide anonymous file-sharing in peer-to-peer networks. GAP is based on the idea of making initiators look like mere intermediaries in order to hide their own actions. To achieve this, every node takes advantage of the traffic generated by other nodes as well as some baseline fake traffic in order to cover the traffic they generate. The operation of a GAP node is depicted in Figure 6. Basically, these nodes perform the following actions: (1) forwarding, which is represented by ordinary arrows within the node; (2) indirection (i.e., sender identity renaming), symbolized by dotted arrows; (3) fake traffic injection, represented by a short dashed arrow; (4) message replay to n random neighbors, and (5) short packet delays as well as (6) message padding.

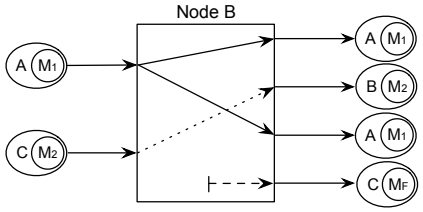

Figure 6: Operation of a GAP node.

In the GAP model a source node must route a sufficient number of packets from other participants to maintain an adequate protection level because the more traffic a node transmits the more unlikely it is to the eyes of an observer that a particular message was originated by that specific node. Received messages can be either forwarded, indirected or dropped. Message forwarding implies no modifications to the message while indirection involves the modification of the sender address and thus the handling of subsequent packets belonging to that connection. However, in this analysis we consider traditional sensor networks where messages only flow from sensors to the base station, thus there is no need to handle replies alleviating the problem of storing large routing tables. Only the forward path will be considered in the rest of this section.

Besides, every node holds a public-key that is used to establish encrypted links between nodes. Public keys are periodically propagated in the network. Additionally, both queries and data traversing the network are encoded using a particular scheme [5] that is similar to a symmetric-key encryption but allows intermediaries to verify if the encoded data matches a specific query or content. In this way, packets change their appearance at every hop but also provides plausible deniability to intermediaries that in general cannot decrypt what they are transmitting. This can also be considered a means of protection against internal adversaries. Finally, to further prevent the correlation between incoming and outgoing messages, small random delays are introduced to received packets and, moreover, packets can be either forwarded or indirected to a random number of nodes. Consequently, this introduces extra traffic in the network, which negatively impacts the lifetime of the nodes.

The requirements imposed to GAP nodes are extremely expensive for hardware-constrained nodes, specially when they have a limited battery supply. The overhead introduced by this solution is briefly presented in Table 6. All nodes in the network should behave similarly but only initiators perform the encryption of data that will traverse the network. This is represented between brackets. The number of replayed packets is represented by R, F indicates the amount

of fake traffic introduced, and again N and T indicate the number of keys that must be stored by the nodes and the number of messages that are temporarily delayed, respectively. These and other values presented in the table are node-dependent and may vary in time.

Node	Requirements
CPU	$[1SK+]\ 2SK(+1RN) + R * PR + F * FT$
RAM	$N * P_K + N * S_K + T * messages$

Table 6: GAP Overhead

This system can protect from both local and global observers because they cannot easily determine the source of messages given the baseline fake traffic injection. On the one hand, a local adversary does not gain any information by following all the messages because this might be fake traffic leading him nowhere. On the other hand, the local adversary is more difficult to deceive because in the presence of continuos events, the traffic load at those network regions will suddenly grow due to the new originated traffic and the packet replay mechanism. Besides, the base station must mimic the behavior of ordinary nodes, replaying and sending traffic but the areas surrounding it would still concentrate more traffic than remote regions. In short, the presented mechanism might be useful in the protection of location privacy in WSNs but the overhead introduced will exhaust the battery of the nodes in a short period of time.

5.3 DC-nets and Herbivore

The Dining Cryptographers (DC) scheme [8] allows a group of users to share information while hiding the actual sender of messages even to other protocol participants. To achieve this, every member needs to share a secret bit with any other participant. For example, in Figure 7, node B share a 0-bit with A and a 1-bit with C. Also, the participants perform the sum modulo 2 (i.e., logic XOR) of their shared secrets. Subsequently, the obtained result is broadcasted to the rest of participants unless the participant is willing to communicate data to the rest of members, in which case it shares the inverse of the result (see node A in Figure 7). The final result is obtained by performing the XOR of all contributions. Every protocol execution is called a round.

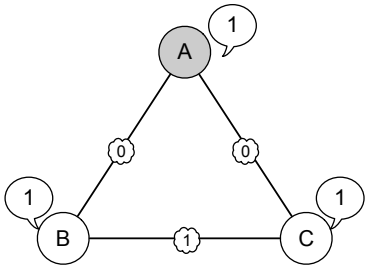

Figure 7: DC-nets communication.

Since every secret is used twice, the final result must be zero if nobody[6] transmitted and one otherwise. Provided that 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. This modification allows the transmission of encrypted messages so that the actual recipient is the only entity capable of determining whether that protocol round

[6]Actually, an even number of simultaneous senders results in zero.

conveyed a real message. The following analysis will be focused on the bit-based version but it might be directly extrapolated to the extended version.

The application of the DC-nets model to WSNs presents several impediments. One of such limitations is that sensor networks communicate wirelessly, which is a highly unreliable medium. The DC protocol is extremely vulnerable to noise and a single erroneous bit leads to undesirable results. Additionally, provided that participants' contributions must be simultaneously broadcasted[7] in order to allow the XOR of their signals, the sensor nodes running the protocol are required to be tightly synchronized and within the transmission range of the other members. This suggests that the data recipient must be either one of the DC participants or an external observer within the communication range. Consequently, only neighboring sensor nodes can run the protocol or they must carefully adjust their transmission power, however, this would deplete their limited batteries in a short time period. As proposed by Herbivore [11], the participants could be hierarchically arranged in order to reduce the complexity of the system. This arrangement would allow sensor nodes to reduce their transmission power but it introduces more synchronization problems and increased delays to reach the destination.

Additionally, there are high memory requirements in the DC model associated with the key sharing process because of the continuous protocol rounds. Two potential solutions exist to the provision of keys: either sensor nodes are preloaded with sufficiently large one-time keys, or they share short keys which are periodically updated by means of a pseudorandom function. In the former case, given the memory limitation in sensor nodes the keys will rapidly expire. In the latter case, the memory cost is traded by computational operations. In any case, the overhead introduced is directly dependent on the topology of the network. A ring topology, as the one presented in Figure 7, requires every node to share 2 random bits, with the right and left participants. On the other hand, in a fully-connected graph every participant shares one bit with every other participant, which adds up $N-1$ random bits, where N is the total number nodes. Note that these values are for a single protocol round (i.e., for the transmission of a single bit). Moreover, a protocol round occurs even if no participant is willing to transmit otherwise an adversary would identify which nodes are interested in transmitting. Clearly, this implies a high waste of bandwidth and energy because of the continuous flow of messages.

Another substantial problem has to do with simultaneous communications. The DC model does not allow various data senders at a time because their messages would collide. This issue highly constrains the usability and nature of sensor networks, which were conceived to provide a highly distributed sensory system. This problem might be reduced by using a slot reservation protocol as proposed in Herbivore, however, this introduces more messages and therefore more energy waste. Moreover, this countermeasure cannot solve the increased delivery time in the communications specially when the sensor networks under consideration are extremely large with a considerable amount of potential data senders. A summary of these and other features constraining the application of this model to WSNs are presented in Table 7, where INV refers to the inversion of the contribution. Note that the computational and memory requirements illustrated are for a single protocol round.

Node	**Requirements**
CPU	$2 * XORs$ $(+1 * INV)$
RAM	[2 *to* $N-1$] bits
Other	Topology restrictions, tight synch, error prone, simultaneity

Table 7: DC-nets Overhead

[7]There are other potential communication techniques besides broadcasting but they imply a significant increase in the number exchanged messages.

Although the computational overhead introduced by the DC-net scheme is rather inexpensive even for sensor nodes, the memory requirements, topological restrictions and the disruption of simultaneous event notification preclude their application to WSNs. Nonetheless, the model is effective in the protection of location privacy because it hides the original data source to all participants and also external (local or global) observers. This could result in a problem for the base station which is unable to identify the data source unless the extended protocol is in use. To this end, the source node would send both event data and its identifier in an encrypted form so that only the base station knows the original sender. Therefore, the location of the data source is protected from disclosure to any other participant including internal adversaries, which are unable to determine the original data sender unless they collude. As a matter of fact, a collusion is successful only if all nodes sharing keys with the potential source node collude which is highly unlikely.

6 Discussion

Previous sections have delved into several features from centralized and decentralized anonymous communication systems that need to be further discussed. This section is intended to provide this final discussion while outlining the most important aspects.

As for the case of centralized solutions, these can be regarded as black box devices where the data sources stand on one side and the data recipients on the other. The communications originated from various sources change their appearance, are delayed or mixed within the network, but still the occurrence of incoming and outgoing messages is evident. In these settings, both source nodes and the base station are clearly exposed to a global observer, simply because they are not part of the network core and thus their actions can be easily detected, which implies the disclosure of their location. Contrarily, local and internal adversaries are placed somewhere within the network core and, in consequence, they cannot identify the communicating nodes so easily. These adversaries count on a partial view of the communications but depending on their situation they might be more likely to uncover the senders and recipients. Specially sensitive are the entry and exit points of centralized systems since in these areas the adversary is capable of distinguishing the source nodes and the base station.

Single-proxy schemes, in particular, are very lightweight because they are primarily based on source renaming at a single intermediate point. However, this together with the potential use of payload encryption for eavesdropping prevention, can protect neither from the trace-back attack performed by local mobile adversaries nor from compromised proxy nodes because they can retrieve the data source from the packet.

Mix-based designs depict a rather different situation. The overhead imposed by mix nodes is significant not only because it demands the use of public-key cryptography but also because the source node performs as many of these operations as nodes in the communication path. Additionally, this implies the knowledge of the public keys of every mix node and the topology of the network to perform the encryptions in the correct order. Moreover, mixes introduce large message delays, which are not suitable for time-critical applications. Regarding the privacy protection, mix cascades present the same problem concerning local adversaries, which are able to follow the paths of messages since they are fixed and they follow any received packets regardless of the appearance or timing. Yet, the random route selection proposed by mix-nets provides some protection means against local adversaries but it might still be insufficient because they can eventually reach either edge of the mix network. From these positions, local attackers are much more likely to succeed. Similarly, internal adversaries which are at the edge of the network are capable of uncovering the communication endpoints. However, the use of layered encryption prevents intermediate nodes in the path from uncovering the data origin.

More precisely, intermediate nodes are only aware of the previous and next hop in the path.

Finally, onion routing solutions reduce some of the computational restrictions imposed by mixes by introducing the path setup process, which allows the establishment of session keys that are later used during the data transmission process. Also, these schemes reduce the delay introduced at every hop by multiplexing the communications of various data sources on a single stream. Although the overhead is reduced, it still demands layered cryptography and great memory requirements. In any case, onion routing schemes present the same problems when countering the adversaries considered in sensor networks.

As for the case of decentralized approaches, their aim is to prevent the aforementioned problems at the edges by making all participants part of the system. In other words, any member of the system is a potential data source as well as a data forwarder. This implies that is not trivial for global observers to determine the communication endpoints and it also introduces the opportunity to more sophisticated internal attacks.

Crowds schemes do not sufficiently protect against global adversaries because the data recipient is not part of the network and data senders start new paths for new data connections, thus altering their behavior and becoming an easy target. Moreover, in order to keep a low overhead, these solutions do not introduce protection mechanisms such as dummy packet injection that might be helpful both against global and local adversaries. Local adversaries can also trace back sources and the base station because the paths are static once created to reduce the chances of internal adversaries.

Contrarily, GAP and DC-nets offer attractive safeguards improving the protection against the various types of adversaries. This implies a significant increase in the number of messages being transmitted, replayed or forwarded, which results in an unaffordable energy waste for battery-powered devices. Additionally, the DC-nets model presents extra limitations in terms of memory requirements, network topology and also the inability to handle simultaneous transmissions, which further preclude its application to the location privacy problem in WSNs.

	Overhead	**Adversary**		
		Global	Local	Internal
Single-proxy [2]	$\downarrow\downarrow$	$\times$	$\times$	$\times$
Mix-nets [7]	$\uparrow\uparrow\uparrow$	$\times$	$\times$	$\surd$
Onion routing [27]	$\uparrow\uparrow$	$\times$	$\times$	$\surd$
Tor [10]	$\uparrow\uparrow$	$\times$	$\times$	$\surd$
Crowds [28]	$\downarrow$	$\times$	$\times$	$\approx$
Hordes [15]	$\downarrow$	$\times$	$\times$	$\approx$
GAP [4]	$\uparrow\uparrow\uparrow$	$\surd$	$\surd$	$\surd$
DC-nets [8]	$\uparrow\uparrow\uparrow$	$\surd$	$\surd$	$\surd$
Herbivore [11]	$\uparrow\uparrow\uparrow$	$\surd$	$\surd$	$\surd$

Table 8: Suitability of Traditional Systems

A visual summary of this discussion is presented in Table 8, where the up and down arrows roughly indicate the overall costs introduced by these systems. The tick, cross and approx. symbols ($\surd$, $\times$ and $\approx$) represent whether these solutions can provide, are not able to provide or could provide some protection against the three adversarial models considered in WSNs.

In general, we can state that centralized approaches are less suitable for the protection of location privacy in WSNs than decentralized approaches given the highly distributed nature of these networks and their characteristic communication pattern. The typical many-to-one communication model makes it difficult to hide the location base station and the source nodes when they are located outside the limits of the centralized network core. A local adversary can eventually determine the entry points of the network core while a global adversary can

directly detect the source and destination of messages. Therefore, decentralized approaches are more convenient since they integrate all the nodes within the anonymizing solution hindering the identification of the current participants to adversaries with either local or global eavesdropping capabilities.

7 Conclusions

This paper presents a rigorous analysis on traditional anonymous communication systems which aims to determine the factors limiting the potential adaptation of these solutions to the location privacy problem in WSNs. Previous works did not provide sufficient analysis on this issue and merely argue, in a very imprecise way, that traditional anonymous communication systems are too resource-consuming for sensor nodes. In this paper we have shown, for the first time, that previous assumptions were not completely true. In particular, we have demonstrated that some solutions are sufficiently lightweight to run in a sensor node, and we have shown that their real weak point is that they do not fit the requirements and the adversarial model considered in the sensors domain. Similarly, another group of solutions are suitable for the protection of location privacy in WSNs but they are rather expensive in terms of computational, memory, and battery requirements. We have drawn these conclusions after providing a precise description of the network features and adversarial models that cause the location privacy problem. Finally, we expect that this work contributes to a better understanding of the originality of the location privacy in sensor networks and serves as the groundwork for the development of new improved solutions in this area.

Acknowledgment

This work has been partially funded by Spanish Ministry of Science and Innovation through the research projects: SPRINT (TIN2009-09237), ARES (CSD2007-00004) and IOT-SEC (ACI2009-0949). The SPRINT Project is co-financed by FEDER (European Regional Development Fund) and the first author is supported by the Spanish Ministry of Education through the National F.P.U. Program.

References

[1] B. Alomair, A. Clark, J. Cuellar, and R. Poovendran. Statistical Framework for Source Anonymity in Sensor Networks. In *GLOBECOM 2010, 2010 IEEE Global Telecommunications Conference*, pages 1 –6, dec. 2010.

[2] Anonymizer, Inc. Hide IP and Anonymous Web Browsing Software. [online], May 2011. `http://www.anonymizer.com/`.

[3] D. Banisar and S. Davies. Privacy and Human rights - An International Survey of Privacy Laws and Practice. [online], May 2012. `http://gilc.org/privacy/survey/`.

[4] K. Bennett and C. Grothoff. GAP - Practical Anonymous Networking. In R. Dingledine, editor, *PET 2003*, volume 2760 of *LNCS*, pages 141–160, Dresden, Germany, 26–28 March 2003. Springer-Verlag.

[5] K. Bennett, C. Grothoff, T. Horozov, and J. T. Lindgren. An Encoding for Censorship-Resistant Sharing, 2003.

[6] S. Chang, Y. Qi, H. Zhu, M. Dong, and K. Ota. Maelstrom: Receiver-location preserving in wireless sensor networks. In Y. Cheng, D. Eun, Z. Qin, M. Song, and K. Xing, editors, *Wireless Algorithms, Systems, and Applications*, volume 6843 of *Lecture Notes in Computer Science*, pages 190–201. Springer Berlin / Heidelberg, 2011.

[7] D. Chaum. Untraceable Electronic Mail, Return addresses, and Digital Pseudonyms. *Commun. ACM*, 24(2):84–88, Feb. 1981.

[8] D. Chaum. The Dining Cryptographers Problem: Unconditional Sender and Recipient Untraceability. *Journal of Cryptology*, 1(1):65–75, 1988.

[9] J. Deng, R. Han, and S. Mishra. Decorrelating wireless sensor network traffic to inhibit traffic analysis attacks. *Pervasive and Mobile Computing*, 2(2):159–186, 2006.

[10] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The Second-Generation Onion Router. In *Proceedings of the 13th conference on USENIX Security Symposium, SSYM'04*, pages 21–21, San Diego, CA, USA, 913 August 2004. USENIX Association, Berkeley, CA, USA.

[11] S. Goel, M. Robson, M. Polte, and E. G. Sirer. Herbivore: A Scalable and Efficient Protocol for Anonymous Communication. Technical Report 2003-1890, Cornell University, Ithaca, NY, February 2003. `http://www.cs.cornell.edu/People/egs/papers/herbivore-tr.pdf`.

[12] A. Jhumka, M. Leeke, and S. Shrestha. On the use of Fake Sources for Source Location Privacy: Trade-Offs Between Energy and Privacy. *The Computer Journal*, 54(6):860–874, 2011.

[13] Y. Jian, S. Chen, Z. Zhang, and L. Zhang. A novel scheme for protecting receiver's location privacy in wireless sensor networks. *Wireless Communications, IEEE Transactions on*, 7(10):3769–3779, October 2008.

[14] P. Kamat, Y. Zhang, W. Trappe, and C. Ozturk. Enhancing Source-Location Privacy in Sensor Network Routing. In *ICDCS 2005. 25th IEEE International Conference on Distributed Computing Systems*, pages 599–608, June 2005.

[15] B. N. Levine and C. Shields. Hordes: A Multicast Based Protocol for Anonymity. *J. Comput. Secur.*, 10(3):213–240, 2002.

[16] L. Lightfoot, Y. Li, and J. Ren. Preserving Source-Location Privacy in Wireless Sensor Network Using STaR Routing. In *GLOBECOM'10*, pages 1–5, 2010.

[17] K. Mehta, D. Liu, and M. Wright. Location Privacy in Sensor Networks Against a Global Eavesdropper. In *IEEE International Conference on Network Protocols, ICNP 2007.*, pages 314–323, Beijing, China, 16–19 Oct. 2007. IEEE.

[18] S. Misra and G. Xue. Efficient anonymity schemes for clustered wireless sensor networks. *International Journal of Sensor Networks*, 1(1):50–63, 2006.

[19] A. A. Nezhad, A. Miri, and D. Makrakis. Location privacy and anonymity preserving routing for wireless sensor networks. *Computer Networks*, 52(18):3433 – 3452, Dec. 2008.

[20] S. Ortolani, M. Conti, B. Crispo, and R. Di Pietro. Events privacy in WSNs: A new model and its application. In *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM)*, pages 1 –9, june 2011.

[21] Y. Ouyang, Z. Le, G. Chen, J. Ford, and F. Makedon. Entrapping Adversaries for Source Protection in Sensor Networks. In *WOWMOM '06: Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*, pages 23–34, Washington, DC, USA, 2006. IEEE Computer Society.

[22] Y. Ouyang, Z. Le, Y. Xu, N. Triandopoulos, S. Zhang, J. Ford, and F. Makedon. Providing Anonymity in Wireless Sensor Networks. In *Pervasive Services, IEEE International Conference on*, pages 145–148, July 2007.

[23] C. Ozturk, Y. Zhang, and W. Trappe. Source-Location Privacy in Energy-Constrained Sensor Network Routing. In *SASN '04: Proceedings of the 2nd ACM workshop on Security of ad hoc and sensor networks*, pages 88–93, Washington, DC, USA, 2004. ACM New York, NY, USA.

[24] S. Pai, S. Bermudez, S. Wicker, M. Meingast, T. Roosta, S. Sastry, and D. Mulligan. Transactional Confidentiality in Sensor Networks. *IEEE Security & Privacy*, 6(4):28–35, July-Aug. 2008.

[25] A. Pfitzmann and M. Hansen. A terminology for talking about privacy by data minimization: Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management. v0.34. [online], Aug. 2010. `http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.34.pdf`.

[26] K. Pongaliur and L. Xiao. Maintaining source privacy under eavesdropping and node compromise attacks. In *Proceedings IEEE INFOCOM 2011*, pages 1656 –1664, april 2011.

[27] M. Reed, P. Syverson, and D. Goldschlag. Anonymous Connections and Onion Routing. *IEEE Journal on Selected Areas in Communications*, 16(4):482–494, May 1998.

[28] M. Reiter and A. Rubin. Crowds: Anonymity for Web Transactions. *ACM transactions on information and system security*, 1(1):66–92, 1998.

[29] R. Shaikh, H. Jameel, B. d'Auriol, S. Lee, Y.-J. Song, and H. Lee. Network Level Privacy for Wireless Sensor Networks. In *ISIAS '08. Fourth International Conference on Information Assurance and Security.*, pages 261–266, Sept. 2008.

[30] M. Shao, W. Hu, S. Zhu, G. Cao, S. Krishnamurthy, and T. La Porta. Cross-layer Enhanced Source Location Privacy in Sensor Networks. In *IEEE Conference on Sensor, Mesh, and Ad Hoc Communications and Networks (SECON '09)*, pages 1–9. IEEE Communications Society, June 2009.

[31] M. Shao, Y. Yang, S. Zhu, and G. Cao. Towards Statistically Strong Source Anonymity for Sensor Networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 466–474, April 2008.

[32] J. Walters, Z. Liang, W. Shi, and V. Chaudhary. *Security in Distributed, Grid, and Pervasive Computing*, chapter Wireless Sensor Network Security: A Survey, pages 367–409. Auerbach Pub, 2007.

[33] H. Wang, B. Sheng, and Q. Li. Privacy-aware routing in sensor networks. *Comput. Netw.*, 53(9):1512–1529, 2009.

[34] S. Warren and L. Brandeis. The Right to Privacy. *Harvard Law Review*, IV(5):193–220, December 1890. doi:10.2307/1321160.

[35] Y. Yang, M. Shao, S. Zhu, B. Urgaonkar, and G. Cao. Towards Event Source Unobservability with Minimum Network Traffic in Sensor Networks. In *WiSec '08: Proceedings of the first ACM conference on Wireless network security*, pages 77–88, New York, NY, USA, 2008. ACM.

[36] B. Ying, J. R. Gallardo, D. Makrakis, and H. T. Mouftah. Concealing of the Sink Location in WSNs by Artificially Homogenizing Traffic Intensity. In *First International Workshop on Security in Computers, Networking and Communications*, pages 1005–1010, 2011.

[37] W. Zhu, Y. Xiang, J. Zhou, R. Deng, and F. Bao. Secure localization with attack detection in wireless sensor networks. *International Journal of Information Security*, 10:155–171, 2011.