Security in Wireless Ad-Hoc Networks – A Survey

R. Di Pietro[a,*], S. Guarino[a,**], N. V. Verde[b], J. Domingo-Ferrer[c]

[a]Roma Tre *University, Dept. of Mathematics and Physics, L.go S. L. Murialdo, 1, 00146, Rome, Italy*
[b]Sapienza *University of Rome, Dept. of Computer Science, Via Salaria, 113, 00198, Rome, Italy*
[c]*Universitat* Rovira i Virgili*, Dept. of Computer Engineering and Maths, Av. Paisos Catalans, 26, 43007, Tarragona, Catalonia, Spain*

Abstract

Pervasive mobile and low-end wireless technologies, such as radiofrequency identification (RFID), wireless sensor networks and the impending vehicular ad-hoc networks (VANETs), make the wireless scenario exciting and in full transformation. For all the above (and similar) technologies to fully unleash their potential in the industry and society, there are two pillars that cannot be overlooked: security and privacy. Both properties are especially relevant if we focus on ad-hoc wireless networks, where devices are required to cooperate – *e.g.* from routing to the application layer – to attain their goals.
In this paper, we survey emerging and established wireless ad-hoc technologies and we highlight their security/privacy features and deficiencies. We also identify open research issues and technology challenges for each surveyed technology.

Keywords: Wireless Networks, Ad-hoc Networks, Security, Privacy, Survey

1. Introduction

A wireless network makes use of radio signals to exchange data between two or more physical devices, usually called "nodes" of the network. The lack of wires permits to overcome most limitations of traditional wired networks, allowing deployment in hostile environments or mobile scenarios. When nodes do not depend on any preexisting infrastructure, wireless networks take the name of *wireless ad-hoc networks*. In this case, communications rely on the ability of the nodes to form a multi-hop radio network. Generally speaking, several vulnerabilities can be identified in ad-hoc networks, and at a very abstract level they can be related to one of the following issues:

Vulnerability of the channel Messages can be eavesdropped and fake messages can be injected or replayed into the network, without the hurdle of needing physical access to network components.

Vulnerability of the nodes Nodes may not be physically protected, and are therefore more prone to capture and tamper attacks. If an adversary gets full

*Corresponding author
**Principal corresponding author. Tel: Fax:
Email addresses: dipietro@mat.uniroma3.it (R. Di Pietro), guarino@mat.uniroma3.it (S. Guarino), verde@di.uniroma1.it (N. V. Verde), josep.domingo@urv.cat (J. Domingo-Ferrer)

access to a node, he can (i) steal sensitive information, (ii) reprogram the node and change its behavior, or (iii) physically damage hardware to terminate the node. Due to nodes vulnerability, secret keys cannot be simply issued when the network is deployed, but a secure and efficient key management scheme is crucial.[1]

Absence of infrastructure Ad-hoc networks are supposed to operate independently of any fixed infrastructure. This makes most classical security solutions, based on certification authorities and on-line servers, inapplicable. Security and privacy *de facto* rely on distributed cooperation among (possibly uncooperative) nodes. In this paper, we will focus on issues introduced by *malicious* nodes, that is, nodes that deliberately interfere with the normal behavior of the network. Faulty and selfish nodes are minor sub-cases that we will not address specifically.[2]

Dynamically changing topology The topology of a wireless networks is potentially ever and quickly changing. Sophisticated routing protocols are often needed, but they may introduce new problems that need to be carefully evaluated. Indeed, incorrect routing information can be generated by compromised nodes, or as a result of some topology changes.[3]

The purpose of this paper is not to survey all existing literature about ad-hoc networks. We rather want to focus on security challenges that arise in five different subsets of ad-hoc networks: Wireless Sensor Networks (WSNs), Unattended Wireless Sensor Networks (UWSNs), Wireless Mesh Networks (WMNs), Delay Tolerant Networks (DTNs) and Vehicular Ad-hoc Networks (VANETs). Since these networks share many features, for the sake of clarity and completeness, we will first introduce the main security challenges common to all wireless ad-hoc networks. Afterwards, we will describe the distinctive features of WSNs, UWSNs, WMNs, DTNs and VANETs. For each one of these network technologies, we will highlight the specific security issues they introduce, and detail the solutions proposed so far in the literature.

1.1. Security Requirements of Wireless Ad-Hoc Networks

Let us start with a brief review of the main requirements that all wireless ad-hoc networks typically have to fulfill.

Availability The services provided by the network must be always available (often in a timely manner), despite of any malfunctioning of the system. Resource depletion attacks are the main class of attacks aiming at subverting this property. Resistance to such attacks is therefore of primary importance.

Integrity Any accidental or malicious alteration to the information stored and exchanged in the network must be (promptly) detected, and possibly thwarted.

[1] The reader can refer to [1] for a survey on key management protocols.
[2] For an overview on the problems introduced by selfishness, the reader can refer to [2].
[3] For a survey of secure routing protocols specific to ad-hoc networks, we suggest [3, 4].

Confidentiality Secret information stored and exchanged in the network must not be divulged to unauthorized parties. In some cases, even the existence itself of a communication between two end-points must be hidden. Cryptographic tools are the typical, but not unique, countermeasure to confidentiality threats. In dynamically changing systems, where nodes can join or leave the network, so-called *forward* and *backward secrecy* need to be addressed as well. *Forward secrecy* means to deny access to any future communication to nodes that left the network. Conversely, *backward secrecy* means to ensure that new nodes are not able to access any message sent before they joined the network. Confidentiality must not be confused with privacy. While the former concerns hiding to outer entities data used by the network to provide the intended services, the term privacy usually refers to private (meta)data–not strictly necessary for the network purposes–which must be concealed even to the network authority. Privacy issues only arise in specific scenarios, and we will therefore address them only when dealing with one of those settings.

Authorization Only authorized nodes must be able to gain access to the network, and only authorized entities must be able to enjoy the services provided by the network.

Authentication It must be always possible to verify the identity of the sender of any message exchanged in the network. Unless it is in control of a corrupted node, no attacker should be able to forge a message, though making it indistinguishable from a legitimate message.

Non-repudiation To be able to find and separate compromised nodes, it must be impossible for the sender of a message to successfully challenge the authorship of that message.

Freshness It must be always possible to verify the newness of data exchanged in the network, to prevent any adversary to re-use old messages to mislead network services.

1.2. Attacks to Wireless Ad-Hoc Networks

At a high level, attacks against wireless ad-hoc networks can be classified based on the status of the attacker, on its behavior, and on the purpose of the attack.

Status The first classification is based on whether the attacker is an *outsider* or an *insider*. Outsider attackers are entities that do not belong to the network but want to disrupt the provided service. Insider attackers are legitimate nodes behaving in a malicious way.

Behavior The second classification distinguishes between *passive* and *active* attacks. The former only consist in eavesdropping communications, and monitoring and analyzing the behavior of the network, without interfering with it. The latter include physical access to a portion of the network, and attempts to modify the normal behavior of the network.

Purpose The third categorization depend on the purpose of the attack. Attacks on network *availability* and *service integrity*, aim at disrupting the services provided by the network. Many denial-of-service, routing and physical

Table 1: Distinctive features of several ad-hoc networks

Network	**Distinctive feature**
WSNs	Very high number of nodes, limited computational power
UWSNs	Intermittent sink
WMNs	Integration of many networks
DTNs	Opportunistic contacts and intermittent connectivity
VANETs	Vehicles as mobile nodes

attacks fall within this category. Attacks against *privacy* and *confidentiality* are attacks that try to gain insight on data exchanged in the network and on the network topology. Finally, attacks against *data integrity* try to alter the data that are transmitted. Malicious nodes can inject false messages, modify existing ones, replicate old packets or entire nodes, etc.

1.3. WSNs, UWSNs, WMNs, DTNs and VANETs

Although sharing many common traits, different network technologies present distinctive features, due to specific requirements and challenges imposed by their application setting. The main characteristics of the types of networks addressed in this paper are highlighted in Table 1, and briefly summarized hereunder.

WSNs consist of a (generally large) collection of resource-constrained autonomous *sensor nodes*, appointed to monitor the environment and report the sensed information to one or more trusted gateway nodes, called *sinks*. Unattended WSNs (UWSNs) are characterized by the intermittent presence of the sink, which generally prevents direct offload of the sensed data, requiring a secure and distributed storage infrastructure. Wireless Mesh Networks (WMNs) refer to the ensemble of technologies enabling the interaction of different type of networks. The challenge resides in providing a certain level of security despite having to deal at the same time with many technologies. Delay Tolerant Networks (DTNs) are characterized by the opportunistic contacts and the intermittent connectivity of their nodes. Finally, Vehicular Ad-hoc Networks (VANETs) are mobile ad-hoc networks designed to have vehicles as mobile nodes.

In the following sections, we will address the formerly introduced network paradigms one by one, highlighting their distinctive security challenges and the corresponding possible countermeasures. We will dedicate particular attention to WSNs, partly due to the numerous threats posed by the severe resource constraints and the physical exposure of their nodes, and partly because many of the issues discussed for WSNs will find applications in the other scenarios as well. A comprehensive analysis of the specific features of UWSNs, WMNs, DTNs and VANETs will nevertheless be provided afterwards.

2. Wireless Sensor Networks

A Wireless Sensor Network (WSN) consists of sensors-equipped nodes, called *motes* or simply *sensors*, sensing the environment and reporting the collected data to one or more trusted gateway nodes, called *sinks*. Sinks sometimes play a coordination role, but the frequency and impact of their presence in the network is highly variable according to the setting [5], so motes are often required to self-organize in a distributed way. WSNs are usually sensibly (sometimes even orders of magnitude) larger than similar ad-hoc networks, and are often deployed in hostile environments and over wide geographic areas. Motes have limited computational power, memory and energy supply, which, together with the adverse working conditions, make them particularly prone to failures. Despite many energy harvesting solutions proposed so far, recharging is still considered hardly feasible, and motes are usually regarded as "disposable" devices. Due to the complexity of replacement and management operations, maximizing lifetime and productivity is of paramount importance. In essence, WSNs are ad-hoc networks with additional and more stringent constraints. They need to be more energy-efficient and scalable than other ad-hoc networks, which exacerbates the security challenges.

Initially, the development of WSNs was mainly motivated by military purposes, but nowadays WSNs are becoming pervasive systems, used in several fields, from home automation to border monitoring. However, military applications, together with automated medical systems, still represent the context where security aspects are more relevant. In both cases, the network handles critical information, hence to ensure data availability is crucial. Further, classified military data and private patients health-status information, raise the concern for confidentiality and privacy.

WSN applications need to contrast most security issues communal to conventional networks, like message injection, eavesdropping, impersonation, etc. However, the design of a security infrastructure in WSNs must pervade any layer of the system, from the application layer to the physical layer (that is often considered secure in conventional settings). Further, mainly because of their limited resources, standard techniques such as tamper-proof hardware, secure routing, public-key cryptography, etc., do not suit WSNs. Specific solutions for WSNs are required, that must be conceived with these low-end devices in mind.

There are two specifications available for WSN communication: IEEE 802.15.4 [6] and ZigBee [7]. The first is a standard for low-rate wireless personal area networks that was developed by IEEE (Institute of Electrical and Electronics Engineers) and contains a number of security suites. Basically, it provides access control, integrity, confidentiality and replay protection; however, it does not deal with authentication or key exchange. IEEE 802.15.4 defines a communication layer at level 2 in the OSI (Open System Interconnection) model and its main purpose is to allow communication between two devices. ZigBee is built upon IEEE 802.15.4. This standard defines a communication layer at level 3 and above in the OSI model. Its main purpose is to create a network topology (hierarchy) to let a number of devices communicate among them, and to add extra communication features such as authentication, encryption and association. The ZigBee network layer natively supports star, tree and generic mesh networks.

In the following sections we will provide a categorization of the attacks that can be mounted against WSNs. We will describe in detail several threats, and we will point out the existing countermeasures. Table 2 summarizes the attacks that we will take into consideration, their categorization according to the classifications provided in Section 1.2, and the corresponding countermeasures.

2.1. Attacks Against Network Availability and Service Integrity

Attacks against network availability and service integrity are often referred to as denial-of-service (DoS) attacks: an adversary attempts to disrupt, subvert or destroy the services provided by the network. DoS attacks can have as a target any layer of the sensor network. Indeed, known attacks perform on the physical, the data link, the network and the transport layers. In this section, we will analyze existing DoS attacks layer by layer.

2.1.1. Physical Layer

In WSNs, attacks to the physical layer can target the communication channel or the sensors. In the first case we speak of *jamming* attacks, while in the second of *tampering* attacks.

Jamming. A jamming attack can be seen as noise created by an attacker with the aim of partially or entirely disrupting a legitimate signal. Such noise is generated using a device called *jammer*, able to interfere with the radio frequencies used by the sensors. The jamming activity is effective only if the signal-to-noise ratio is less than 1. Depending on its transmission power, the jammer may disturb the entire network or a smaller portion of it. If ignored in the initial WSN design, a jamming attack can easily disrupt a network, regardless of higher level security mechanisms. Jamming can be classified as follows [8]:

Spot jamming is the simplest jamming technique. The attacker directs all its compromising power against a single frequency. It is usually effective, but it may be avoided by changing the frequency used.

Sweep jamming targets multiple frequencies in quick succession, by rapidly shifting the target frequency. Since the activity of the attacker is not continuous, the effectiveness of this type of attack is limited. However, in WSNs it can force many retransmissions due to packet loss.

Barrage jamming concurrently targets a range of frequencies. However, as the attacked range grows, the output power of jamming is reduced proportionally.

Deceptive jamming consists in fabricating or replaying valid signals on the channel incessantly, thereby occupying the available bandwidth and trying to destroy the network service. It can be applied to a single frequency or a set of frequencies.

Table 2: Attacks against wireless sensor networks

Target	Layer	Attack	Countermeasures	Attacker: Internal	Attacker: External	Attack: Active	Attack: Passive
Network Availability and Service Integrity	Physical	Jamming	Detection techniques, proactive, reactive, and mobile agent-based countermeasures		×	×	
		Tampering	Tamper-proofing, software tamper detection, sensor monitoring		×	×	
	Link	Collision	Forward error-correcting codes		×	×	
		Exhaustion	Rate limitation		×	×	
		Unfairness	Error-correcting codes	×		×	
		Sleep Deprivation	Anti-replay protection, strong link-layer authentication, and broadcast attack protection		×	×	
	Network & Routing	Routing Information	Authentication, MAC	×		×	
		Hello Flooding	Authentication, bi-directionality checking, signal strength		×	×	
		Black Hole	Authentication, REWARD, watchdog and pathrater		×	×	
		Sink Hole Attack	Authentication, monitoring, secure routing		×	×	
		Selective Forwarding	Authentication, IDS, multi-hop acknowledgements, multipath routing		×	×	
		Wormhole Attack	Authentication, packet leashes		×	×	
		Sybil	Authentication, radio resource testing, key validation for random key pre-distribution, position verification		×	×	
	Transport	Flooding	Client puzzles, cryptographic techniques	×		×	
		Desynchronization	Authentication		×	×	
Privacy and Secrecy	Physical	Eavesdropping	Cryptographic techniques		×		×
	Network	Traffic Analysis	Randomized communications	×		×	
Data Integrity	Physical	Node Replication	Emergent properties		×	×	
	Network	Packet Injection	Data authentication	×	×	×	
		Packet Duplication	Data authentication	×	×	×	
		Packet Alteration	Data authentication	×	×	×	

First generation sensor nodes used single-frequency radios, and were therefore vulnerable to narrowband noise, whether unintentional or malicious.[4] More recent motes use direct-sequence spread spectrum to reduce vulnerability to noise.[5] More generally, several countermeasures can be used against the various jamming attacks. Frequency-Hopping Spread Spectrum (FHSS), Direct Sequence Spread Spectrum (DSSS), Hybrid FHSS/DSSS, Ultra Wide Band (UWB) technology, antenna polarization, directional transmission, and regulation of the transmission power are a few examples [9, 10, 11]. However, they do not defeat an adversary with knowledge of the spreading codes or hopping sequence. Indeed, these are not secret, but either standardized (in IEEE 802.15.4) or derivable from node addresses (in Bluetooth). Existing security schemes that address jamming attacks in WSNs can be broadly classified as follows:[6]

- *Detection* techniques aim at instantly detecting jamming attacks. As observed in [12], signal strength, carrier sensing time or packet delivery ratio individually are unable to conclusively detect the presence of a jammer. To improve detection, the authors of [12] introduce the notion of consistency checking, where the packet delivery ratio is used to classify a radio link as having poor utility, and then a consistency check is performed to classify whether poor link quality is due to jamming.
- *Proactive* countermeasures make a WSN immune to jamming attacks rather than reactively respond to such incidents. An example is DEEJAM, a protocol proposed to defend against stealthy jammers using IEEE 802.15.4-based hardware [13]. To contrast adversaries that use hardware with same capabilities as the deployed nodes, it uses four defensive mechanisms altogether:
  - *Frame masking* defends against attackers jamming only when their radio captures a multibyte preamble and a Start of Frame Delimiter (SFD) sequence.
  - *Channel hopping* defends against attackers that try to detect radio activity by periodically sampling the Radio Signal Strength Indicator (RSSI), and start jamming when RSSI is above a programmable threshold.
  - *Packet fragmentation* consists in breaking each packet into short fragments, transmitted separately on different channels and with different SFDs. If the transmission frequency changes fast enough, the attacker cannot start jamming on the right frequency in time.
  - *Redundant encoding* tackles an attacker that blindly jams a single channel using short pulses. It allows packet recovery even if a fragment is corrupted, but energy and bandwidth usage are increased.
- *Reactive* countermeasures enable reaction only upon the incident of a jamming attack. A perfect example is the JAM algorithm proposed in [14], which

[4] *e.g.*, Mica2 and prior motes used the Chipcon CC1000 transceiver, operating at 433 or 900MHz.

[5] *e.g.*, MICAz and Telos motes use the Chipcon CC2420, which operates at 2.45 GHz.

[6] For a comprehensive summary of counteractions against jamming in WSNs, we remand the reader to [8].

enables the detection and mapping of jammed regions to increase network efficiency. In practice, nodes near the border of a jammed region notify their neighbors, which start mapping the region that is currently jammed by exchanging mapping messages. When the jammer moves or simply stops the attack, the jammed nodes recover and notify this change to their neighbors. *Mobile agent-based* countermeasures leverage Mobile Agents (MAs), *i.e.*, autonomous programs that can move from host to host and act on behalf of users towards the completion of an assigned task. An example is the JAID protocol presented in [15], where MAs explore the network incrementally fusing the data as they visit the nodes. Firstly, to identify near-optimal itineraries for the MAs, JAID separates the network into multiple groups of nodes, calculates local near-optimal routes through each group, and assigns these itineraries to individual agent objects. Then, such itineraries are modified using the JAM algorithm, so as to avoid jammed areas, while not harming the efficient data dissemination performed by normally working sensors.

Tampering. A wide range of active attacks, generally carried out by outsiders, all rely on a communal approach: gaining physical access to a subset of sensors by tampering with their hardware. DoS attacks are only one of the possible ways an adversary can leverage tampering. More generally, the purpose may be to modify the behavior of the nodes, to replace them with malicious sensors under the control of the attacker, or to steal confidential data and cryptographic material [16, 17, 18]. To provide a clearer exposition, here we will only discuss resilience to tampering itself. Higher-level countermeasures to attacks for which physical access to the sensors in only a prerequisite will be discussed in the pertaining section. The primary defense against physical tampering focuses on building tamper-resistant sensors [19]. However, although tamper-resistant hardware is becoming cheaper, in most cases it is not a feasible option. As an alternative, softwares were specifically designed to detect tampering attempts, and promptly delete sensitive data (such as cryptographic keys) before executing a self-termination protocol. Tampering with current sensor node hardware has been investigated in [20]. The authors show that attacks that can be executed without interruption of the regular node operation usually have a minor impact. All most serious attacks, which result in full control over a sensor node, require the absence of the node from the network for a substantial amount of time. Therefore, simply monitoring sensor nodes for periods of long inactivity can be considered a good defensive strategy.

2.1.2. Link Layer

In WSNs, several attacks can be mounted on the data-link layer. All such attacks share two main objectives: (i) depleting the energetic resources of the sensors, relying on the fact that most energy consumption in WSNs is due to communication, and (ii) degrading the timeliness of the service.

Link Layer Collision. This attack is very similar to jamming in the physical layer. It occurs when an attacker uses his radio to identify the frequency used by the WSN, and, as soon as he hears the start of a legitimate message transmission, he sends a signal for as little as one octet (or byte) in order to corrupt the entire message [21]. The only evidence of the attack is the reception of an incorrect message, which is detected when a link layer frame fails a cyclic redundancy code (CRC) check. In that case, the link layer automatically discards the entire packet, thereby causing energy and bandwidth waste. A possible countermeasure is provided by forward error-correcting codes (FEC), able to reactively recover lost information [22].

Link Layer Exhaustion. This attack occurs when the attacker manipulates protocol efficiency measures and causes nodes to expend additional energy. Providing a rate limitation by allowing nodes to ignore excessive network requests from a node is an effective countermeasure against this attack.

Unfairness. In an unfairness attack, the adversary transmits a large number of packets when the medium is free, to prevent honest sensors from transmitting legitimate packets. As a result, the quality of service degrades and real-time deadlines are possibly missed. However, this attack is usually considered a weak form of DoS, because it can be limited by using smaller frames, in such a way that the channel is only captured for a small amount of time.

Sleep Deprivation Torture. In WSNs, a sleep mechanism is used by the nodes to adjust their operation mode and extend their lifetime. At full power, a sensor can run for approximately two weeks before exhausting its power resources. To the contrary, if nodes remain in sleep mode and activate as little as possible (*e.g.*, around 1% of the time), their batteries can last even more than a year. As the name suggests, the "Sleep Deprivation Torture" or "denial-of-sleep" attack, firstly introduced in [23], aims at preventing a sensor from sleeping. According to how sleeplessness is induced, it can be classified into two categories [24]: (i) *service request power attacks*, which intensively repeat usual service requests, and (ii) *benign power attacks*, which solicit power-intensive operations on the device under attack. In [25], the authors proposed three ways to lessen the effect of these attacks. The first and most important component of denial-of-sleep defense is *strong link-layer authentication*, which prevents the attackers to send trusted MAC-layer traffic. Existing options for implementing link-layer authentication in WSN include TinySec, which is incorporated into current releases of TinyOS, and the authentication algorithms built into IEEE 802.15.4-compliant devices. The second feature is *anti-replay protection*, usually achieved by maintaining a neighbor table of packet sequence numbers. Unfortunately, such a table can become unwieldy even in moderately sized networks. However, network layer neighbor information can be leveraged to limit the number of neighbors that must be tracked to those from which legitimate traffic is expected. In particular, the authors of [26] suggest to use a protocol called CARP that bounds the size of the neighbor table according to the maximum node degree and the number of clusters that are previously configured. Finally, *broadcast attack protection* allows to detect a denial-of-sleep broadcast attack based on measurements of the ratio of legitimate to malicious traffic, along with the percentage of time that the device is able to sleep.

2.1.3. Network and Routing Layer

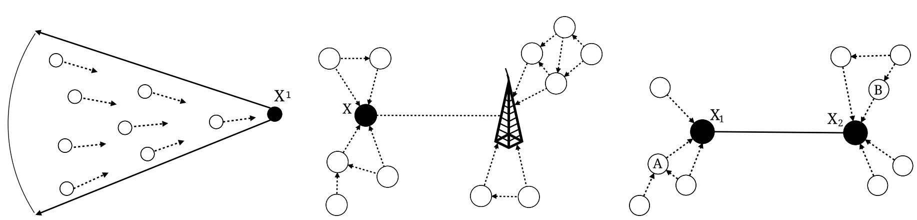
At the network layer, many attacks can disrupt the network availability. We will describe them one by one, together with specific countermeasures. However, it is worth taking into account that in general security at the network layer highly depends on authentication. Due to resource constraints, authentication in WSNs cannot rely on public key cryptography. Based on symmetric keys and hash functions, Zhang and Subramanian [27] proposed a message authentication approach which adopts a perturbed polynomial-based technique to simultaneously accomplish the goals of lightweight and resilience to a large number of node compromises, immediate authentication, scalability, and non-repudiation.

Direct Attacks on Routing Information. A direct attack against the routing layer can try to spoof, alter, or replay routing information. By subverting this information the adversary can change to his favor the data flow. An effective countermeasure against the first two problems is to use a message authentication code (MAC). Counters or timestamps can be used to defend against replay attacks [28]. More generally, the authors of [29] proposed two techniques that mitigate the effects of routing misbehavior: the watchdog and the pathrater. The first is used to identify misbehaving nodes, while the second helps routing to avoid these nodes. Similar countermeasures can as well contrast most of the attacks exposed afterwards.

Hello Flooding. Hello messages are often used to discover neighboring nodes and automatically create a network. Many protocols which use this mechanism make the naive assumption that the sender is within radio range. However, an adversary with a high powered transmitter can corrupt a sensor and make other sensors believe that such a malicious node is in their neighborhood. Data packets routed to the malicious sensor will be indeed sent into oblivion [30], causing both data loss and energy wasting. Figure 1a illustrates such an attack. Generally, a simple countermeasure to the hello flooding attack is to check for bi-directionality of each transmission link. In [31] a method based on signal strength has been proposed to detect and prevent hello flooding attacks.

Black/Sink Hole Attack. The black hole attack works by inducing the sensors to route all the traffic through a set of compromised nodes, that can then drop (or access) all the routed packets. This attack can be detected by listening to and monitoring transmissions by neighbors, and can be tackled using advanced routing algorithms such as REWARD [32]. Black hole attacks can be even more dangerous when the attacker knows the position of the sink, and tries to become the node used by all other nodes to reach the sink. In this case the attack is called Sink Hole Attack, depicted in Figure 1b. To detect sink holes, the authors of [33] proposed an algorithm that firstly finds a list of suspect nodes, and then identifies the intruder in the list through a network flow graph. However, the sink must flood the network with a request message containing the IDs of the affected nodes, and then these nodes have to answer with specific information regarding the correct path, making the algorithm burdensome. In [34] and [35], two other routing protocols against the sink hole attack have been proposed. However, they are respectively based on the Ad-hoc On-demand Distance Vector (AODV) and the Dynamic Source Routing (DSR) protocols, both not very suitable for WSNs. In [36], an intrusion detection system called MintRoute was proposed. It detects sink hole attacks and can be used with the most widely used routing protocol in sensor network deployments.

Wormhole Attack. The wormhole attack leverages a fast and powerful connection (often a wired one) between two faraway compromised nodes to subvert routing information. The adversary can tunnel data between the locations of the two nodes, so as to convince other sensors that they know the quickest path to reach the other side of the network. Figure 1c shows this attack. Most existing ad-hoc network routing protocols, without some specific defensive mechanism, will be severely disrupted by this simple attack. A general solution for detecting and countering wormhole attacks has been introduced in [37], based on *packet leashes*. A leash is any information that is added to a packet in order to restrict its validity. Two types of leashes are proposed: geographical and temporal. The former ensure that the recipient of the packet is within a certain distance from the sender, while the latter establishes an upper bound on the packet's lifetime.



(a) Hello Flooding (b) Sink Hole Attack (c) Wormhole attack

Figure 1: Some examples of attacks at the network and routing layer

Selective Forwarding. When a malicious node does not follow the routing protocol, but acts as a filter forwarding certain messages and dropping others, we face a selective forwarding attack [30]. The black hole attack can be seen as a special case of selective forwarding, where all the packets are dropped. In [38] a centralized intrusion detection scheme based on Support Vector Machines (SVMs) and sliding windows is proposed to tackle selective forwarding. In the scheme presented in [39], instead, detection occurs in both the base station and the source nodes, with alarms raised based on multi-hop acknowledgements from intermediate nodes. Finally, selective forwarding can be tackled using redundant schemes like multipath routing [30, 40]: the same packet is sent along multiple paths to increase the probability of reaching its destination.

Sybil. A Sybil attack consists in a malicious node claiming multiple identities. It was first introduced in peer-to-peer networks [41], but Karlof and Wagner [30] showed it can be a threat in WSNs as well. Fault tolerant schemes, routing and distributed storage algorithms can be easily affected by such an attack. A taxonomy of possible variants of Sybil attacks in WSNs was presented in [42], together with several defensive mechanisms. The main countermeasures are: (i) radio resource testing, that is, asking all nodes to transmit at the same time in a different channel, (ii) key validation for random key pre-distribution, that is, verifying that a node possesses the keys associated to the identity it claims to have, and (iii) position verification, that is, identify Sybil nodes based on the fact that they will appear exactly at the same position.

2.1.4. Transport Layer

All transport layer protocols can be classified into those that provide congestion control mechanisms, and those that provide reliability [43] of the data transfer. The latter are the most relevant, and their main purpose is to guarantee that every packet loss is detected, and that lost packets are retransmitted until they reach their destination. A reliable transport layer protocol can only detect packet losses if there is some kind of feedback in the system. A scheme can use two types of acknowledgements (ACKs): explicit, when a node sends back a confirmation for any packet received, or implicit, when each node verifies the delivery of a packet to a neighbor by overhearing that that neighbor is forwarding the packet. Further, a protocol can use negative acknowledgements ($NACK$s) if nodes are somehow able to realize the non-reception of a packet, and they explicitly send a request for retransmission.

Several transport layer protocols have been explicitly designed for WSNs (*e.g.*, Fusion [44], CODA [45], CCF [46], Siphon [47], ARC [48], Trickle [49], STCP [50], ESRT [51], GARUDA [52], PSFQ [53], DTC [54], RBC [55]).[7] Unfortunately, most of such protocols were designed to ensure reliable communications in the presence of unintentional errors, and not when the network is under attack. Indeed, they fail to provide end-to-end reliability and are subject to increased energy consumption in the presence of an adversary that can replay or forge control packets. When considering attacks at the transport layer, however, we need to assume that the adversary cannot delete both control and data packets (*e.g.*, by jamming), because it would make theoretically impossible to ensure reliable communication [57]. An attack is considered successful if either a packet loss remains undetected or the attacker can permanently prevent the

[7] A detailed description of all existing protocols is out of the scope of this paper. We remand the interested readers to the corresponding references or to the surveys [43, 56].

delivery of the packet. Both ACK and NACK-based schemes are vulnerable to injected control packets, but in general ACK-based protocols cannot even ensure reliability, while NACK-based protocols are only vulnerable to energy depleting attacks. Since the latter type of attacks are in practice less relevant, NACK schemes (*i.e.*, PSFQ [53], [52]) may be preferred to ACK schemes (*i.e.*, [54], [55]). NACK schemes are also more suitable for multi-hop communication, but they have two intrinsic weaknesses. On the one hand, the last fragment of a message is theoretically not protected. This problem is in reality easy to solve, by including the total number of fragments in the first transmitted fragment. More importantly, NACK schemes offer no defense against the loss of the whole message, and there is no satisfactory solution at the moment for this problem [58]. Providing authentication at lower layers can solve many of the above cited problems. At least, by authenticating control packets, it would be more difficult for an attacker to deplete the batteries of the sensors, and thus, to decrease the lifetime of the network. In the following, we will analyze the two main type of attacks to the transport layer [19]: flooding and desynchronization.

Flooding. Flooding attacks exhaust the memory resources of a sensor, by sending many connection establishment requests to the victim, which consequently allocates resources that maintain state for that connections. To reduce the severity of these attacks, *client puzzles* have been introduced [59]: when a client requires the access to a resource, the server answers with a puzzle that the client has to solve in order to gain the required access. Even if puzzles involve a processing overhead, this is often acceptable with respect to excessive communication. A protocol based on client puzzles and suitable for WSNs has been proposed in [60]. It mitigates DoS attacks against broadcast authentication by leveraging a weak authentication mechanism that uses a key chain.

Desynchronization. In a desynchronization attack, the adversary forges messages containing bogus sequence numbers or control flags to disrupt an existing connection between two end-points. By continuously causing retransmission requests, this attack can eventually prevent the end-points from exchanging any useful information, other than quickly drain all the power resources of the attacked nodes. The typical and effective countermeasure to this attack is authentication, whether of the header or of the whole packet.

2.2. Attacks Against Confidentiality and Privacy

The more WSNs become pervasive, the more confidentiality and privacy represent two primary concerns. For example, in military applications confidentiality is a must. On the other hand, in participatory sensing privacy is usually considered a priority with respect to confidentiality. In many other contexts, like automatic health monitoring or commercial applications, privacy and confidentiality are both fundamental [61]. Data confidentiality needs to be enforced through access control policies, to prevent misuse of information by unintended parties. Privacy must be addressed when sensors are not property of the central authority, or in general every time data gathering may involve contextual information which monitored entities do not want to share with the network authority. Confidentiality and privacy issues involve even ethical or legal aspects. However, we will only discuss the technological solutions to enforce such security requirements in WSNs.

2.2.1. Eavesdropping

If end-to-end communications are not protected, anyone is able to discover the communication content by simply eavesdropping on the network's radio frequency range. This way, even passive outsider adversaries can steal private or sensitive information. The standard approach to face this basic attack is cryptography: data are encrypted so that only intended recipients can decrypt the message. Some non-cryptographic approaches have been discussed in the literature [62], but their interest is limited when the adversary can only eavesdrop. Because of sensors having limited computational power, symmetric-key encryption is preferable to public-key. The most used and cited scheme based on symmetric keys is SPINS, introduced in [28]. SPINS is a suite of security protocols optimized for WSNs and consisting of two building blocks: the first assures data confidentiality, two-party data authentication and data freshness, while the second provides an efficient broadcast authentication mechanism. Usual key management schemes for symmetric protocols are unfeasible in many WSNs applications, so most encryption schemes rely on key pre-distribution. When two sensors want to communicate securely, they must first execute a key-discovery phase, to find out which keys they share, and then compute a session key based on such shared keys [63, 64]. However, the authors of [65] highlight a major problem of keys pre-distribution: an attacker can easily obtain a large number of keys by capturing a small fraction of nodes, and leverage such keys to disrupt the authentication mechanism. In particular, if the sink is mobile and cannot therefore be identified by its position, the adversary can deploy a replicated sink preloaded with the compromised keys, which many sensors will confuse with the legitimate sink. To address this issue, the authors propose a new framework relying on two separate key pools, one for the mobile sink to access the network, and one for pairwise key establishment between the sensors. Finally, in WSNs data collected by sensors are usually processed and aggregated at each intermediate node before they reach the sink, to achieve power efficiency by reducing data redundancy and minimizing bandwidth usage. *Data aggregation* is unfortunately in conflict with data confidentiality. The former requires encryption between the originating node and the sink, but apparently intermediate nodes need to access the cleartext, in order to aggregate data. Homomorphic encryption is the natural solution to overcome this impasse, as proposed in [66] and [67].[8] The scheme presented in [66] allow to aggregate data in a confidential and efficient way, relying on a simple but provably secure homomorphic encryption function. The scheme in [67] is able to provide both confidentiality and integrity of the aggregated data.

[8] We suggest [68] for a review of aggregation techniques enforcing confidentiality in WSNs.

2.2.2. *Traffic Analysis*

Encryption alone is not enough to assure *secrecy in a broader sense*. An adversary can analyze overheard data traffic to gain important information about the network topology and the sensed events. Just leveraging traffic analysis, the adversary can identify sensors with special roles [69], or run targeted attacks designed to maximize harm. Deng *et al.* proposed countermeasures against traffic analysis attacks that seek to locate the base station [69]. Recently, Wadaa *et al.* proposed schemes to randomize communications during the network set-up phase, to protect anonymity of sensor network infrastructure [70].

2.3. *Attacks Against Data Integrity*

Data integrity is violated when the adversary corrupts records, and the sink is not able to restore the original sensed data, or at least to detect that data have been manipulated. The simplest attack to compromise sensed data is data erasure, that is, delete any trace of a specific information before it reaches the sink. However, in our classification of security requirements, we considered *data survivability* (the common way to denote resilience to data erasure) a segment of service integrity, rather than data integrity. In standard WSNs, data are off-loaded to the sink as soon as possible, so data erasure requires either compromising the originating sensor before it sends the target information, or intercepting such information along the routing path towards the sink. The former strategy can be easily tackled by letting routing start as soon as data are gathered. The latter is usually implemented using black/sink hole attacks, which can be countered as described in Section 2.1.3. To face more subtle threats to data integrity, tamper-resistance and authentication represent the two basic approaches. On the one hand, if the adversary gains full control of a sensor, fake data injected by that sensor will look legitimate to the sink. On the other hand, if authentication is not used at all, any outsider adversary can alter messages exchanged in the network by simply implementing a man-in-the-middle attack. In general, the success rate of an attack to data integrity depends on the ability of the adversary to capitalize on its resources to circumvent authentication mechanisms. Along this line, *node replication* is the main approach to maximize the impact of sensors corruption. Other active attacks are instead based on *packet injection, replication, and alteration.*

2.3.1. *Node Replication*

When an adversary captures a sensor without being detected, he can use that sensor to inject authenticated, but fake, data. Even if sensors used in typical WSNs are not tamper-proof (mainly for cost reasons), in most application settings the number of sensors that an adversary can concurrently control is however limited. To boost the attack, the adversary can clone the corrupted sensors and insert the replicas in the network. Even if the adversary compromises a single node, he can generate enough replicas to subvert voting or data aggregation protocols. To contrast replication, the network should realize that two different nodes are claiming the same identity. Unfortunately, the distributed nature of most WSNs makes detection challenging when clones of the same sensor are deployed faraway from each other. Centralized monitoring [64, 71] can be a solution: all nodes in the network periodically transfer to a central entity a list of their neighbors, including nodes ID and location. If the same node claims two (or more) conflicting locations, the sensor is considered corrupted and all its credentials are revoked. However, centralized approaches have two drawbacks: the introduction of a single point of failure, and the communication overhead incurred by the nodes that surround the central entity. Emergent properties have been used in contrast to centralized monitoring in [72]. The authors proposed two algorithms extremely resilient to active attacks, and both trying to minimize power consumption by limiting communication.

2.3.2. Packet Injection, Replication and Alteration

To modify data gathered by the network, the adversary has three main alternatives: inject completely false data, replicate previously captured packets, or intercept messages and alter their content. All these attacks can be easily run by insiders, but if the adversary is an outsider they require to break the authentication mechanisms to varying degrees. Injection requires forging from scratch a message that must be indistinguishable from legitimate ones. Replication uses already authenticated massages, but counters or timestamps used to avoid replay attacks need to be counterfeited. Alteration is in general as difficult as injection, but it can result sensibly easier when homomorphic encryption/authentication is used (*e.g.*, for data aggregation). Generally, standard asymmetric authentication protocols are not suitable for WSNs, and are replaced by schemes relying on symmetric keys [64, 71, 73, 74]. In particular, μTESLA [28] is the "micro" version of the Timed Efficient Stream Loss-tolerant Authentication (TESLA) scheme proposed in [75]. It is based on TESLA, but key update and initial authentication are modified to fit for WSNs. The main idea of μTESLA is to use a one-way hash function F to form an "inverse" key chain: the sender chooses the last key K_n of the chain randomly, and applies F repeatedly to compute "previous" keys as $K_i = F(K_{i+1})$, for $i = n - 1$ down to 1. A key is published some time after the corresponding message is sent (therefore, μTESLA requires loosely synchronization between the sensors and the base station, and that each node knows an upper bound on the maximum synchronization error). Since previous keys can be verified through the current key, while the current key cannot be computed from previous keys, an attacker cannot forge keys and authenticate messages. The authors use MD5 as the one-way hash function in TESLA and μTESLA. However, when the adversary corrupts a number of sensors, he can inject fake data which will be correctly authenticated, regardless of the robustness of the cryptographic schemes used. In [76], the authors propose BECAN, a bandwidth-efficient cooperative authentication scheme for filtering injected false data. Such scheme is able to detect the majority of fake data during the routing path to the sink with minor extra overheads at the en-route nodes, obtaining a remarkable reduction of the burden of the sink. Finally, things become even harder when data are aggregated during the routing path. In this case, the authors of [77] propose a secure extension of the robust (against unintentional failures) but insecure (against fake data injection) *synopsis diffusion* algorithm presented in [78]. In particular, the extension is based on a novel lightweight verification algorithm by which the base station can determine if the computed aggregate includes any false contribution.

2.4. Summary of Security Threats and Countermeasures in WSNs

In this section, we discussed the main security threats and countermeasures in WSNs, classifying attacks according to their *target*. Depending on the service provided, secure WSNs need defensive mechanisms to protect (i) network availability and service integrity, (ii) data confidentiality and privacy, and/or (iii) data integrity. When dealing with network and service reliability, we further distinguished the threats based on the attacked *layer*, which sensibly affects the nature of the attack (and of the corresponding defenses). Security mechanisms must perform at each layer, from the physical, to the link, the network, and the transport layer. Most of the issues discussed for WSNs extend to the following network models, which however introduce additional and specific security requirements.

3. Unattended WSN

Unattended Wireless Sensor Networks (UWSNs) were introduced in 2007 [79], to capture all settings where the inaccessibility or the hostility of the deployment scenario would make the assumption of a constant data sink unrealistic. Many examples can be made to highlight the importance of this more specific model: military applications (from the exploration of a battlefield, to the surveillance of a harbor), underground or submarine networks, wildlife monitoring, detection of illegal activities, etc. In all these cases, an intermittent sink is often the only alternative.

The absence of a constant, trusted, central authority, able to both monitor the network and gather sensed data in (quasi) real time, makes data security in UWSNs more challenging than in traditional WSNs. In Section 2, we saw that data integrity and confidentiality in WSNs depend primarily on intrusion detection, encryption, authentication, and multipath routing. data and route them along multiple paths. In fact, in WSNs the sink can supervise the network and (almost) continuously check for sensors malfunctioning or capture. Sensed data are promptly sent to the sink, and do not need to be securely stored in the network. To the contrary, in UWSNs it is natural to assume that the adversary can leverage the absence of the sink to compromise sensors, read, delete or alter sensed and stored information, and disappear without leaving any evidence of its illegal behavior. In other words, intrusion resistance is unfeasible, and the attention is moved to *intrusion detection and recovery*. Further, data sensed while the sink is away is extremely exposed, and it is necessary to enforce *data survivability, confidentiality and authentication* using *secure distributed data processing and storage schemes*. Before discussing more in detail security threats and countermeasures for UWSNs, let us better discuss the *adversary model*, the *cryptographic techniques* that can be used, and the *security requirements* to ensure.

Adversary Model. As we already pointed out, in UWSNs it is natural to assume the presence of an *active* outsider attacker, able to compromise nodes during the absence of the sink without leaving traces. However, the number of sensors that the adversary can corrupt in each interval is limited, since otherwise it could gain complete control of the network and irreparably threaten security. For a similar active but limited adversary, it is fundamental to distinguish between *mobile* or *stationary* attacks. Independently from the fact that the network itself is mobile or stationary, the distinction between mobile and stationary adversaries aims at capturing the ability of the attacker to compromise different sets of sensors. Depending on the adopted model, a mobile adversary can physically move and compromise sensors deployed around him, or somehow "jump" from a set of sensors to another one. A stationary attacker instead chooses a subset of sensors at the very beginning of his attack without changing his target thereafter.

Cryptographic Techniques. As in more general WSNs, symmetric encryption is usually used for data confidentiality and authentication purposes. Simple cryptographic functions are preferable, like one-way hash functions [80] and efficient symmetric scheme such as AES [81] or Skipjack [82]. Skipjack is in particular used for WSNs in the TinySec scheme [83] due to its power efficiency. However, the more stringent security requirements sometimes push towards public key cryptography, which gives more guarantees at the cost of a major resource consumption.

Security Requirements. The three main security requirements in UWSNs are: *Data Survivability and Confidentiality*, *Intrusion Detection and Recovery*, and *Data Authentication*. Since data cannot be off-loaded to the sink in real time, they reside in the network for a longer time than in typical WSNs. This exposes data, raising concern for their integrity and confidentiality. In particular, data *survivability* becomes a major issue because the main objective of an adversary is often to delete sensed data before they reach the sink. Intervals between successive sink visits represent periods of vulnerability, and therefore they give a boost to the activities of an adversary. Frequent intrusions become a necessary assumption, and it is fundamental to be able to detect when nodes are not working as intended, or (even better) to recover compromised sensors. In particular, *self healing schemes* are a remarkable mechanism to restore secure communication with previously corrupted nodes. Finally, the attention paid to data authentication in UWSNs is mainly due to a simple observation: UWSNs cannot use standard data authentication mechanisms that rely on a centralized entity, otherwise, with sufficient time between sink visits, an adversary could easily compromise sensors collected data. In the sequel, we will categorize the main threats based on the security requirements they affect, and describe the corresponding solutions proposed in the literature.

3.1. Data Survivability and Confidentiality

In UWSNs, sensors inability to directly off-load data to the sink makes it easy for an adversary to perform focused attacks aimed at deleting certain target data. Further, the fact itself that UWSNs are often deployed in hostile environments means that it is extremely reasonable that the network is performing some sort of surveillance duties. Consequently, *data survivability* is usually considered the main concern. In this scenario, it is normal to assume a mobile adversary who is actively hunting a certain data item, and who is not afraid to delete/erase any other data he finds.

In [79], the authors proposed a better characterization of the adversary as

Lazy, when the attacker is stationary, and at the beginning of the protocol chooses k nodes to compromise, without ever changing his target thereafter;

Frantic, when the attacker is mobile, and captures a different subset of k randomly chosen nodes each time the sink leaves the network;

Smart, when the attacker is mobile, but only skips between two pre-selected sets of nodes, each of size k.

In the paper, three simple non-cryptographic survival strategies were studied:

DO-NOTHING is the trivial survival strategy, where each sensor simply stores its own sensed data, waiting for the sink arrival;

MOVE-ONCE prescribes that data are moved just once to a new sensor randomly picked among the whole network;

KEEP-MOVING requires that data are continuously and randomly moved from sensor to sensor.

The analysis of all possible attack-survival strategy combinations conducted in [79] highlights that: (i) the DO-NOTHING survival strategy is useless, (ii) when MOVE-ONCE is implemented, a FRANTIC adversary is the most advantaged, and (iii) when KEEP-MOVING is used, a SMART attacker is the most effective one. In [84], resilience to an adversary dubbed ERASER, who wants to indiscriminately erase any information, is analyzed. Surprisingly, the best survival strategy results the DO-NOTHING: moving data only helps the ERASER to encounter and erase all data faster. However, the authors investigated the effects of data replication, showing that with replication the KEEP-MOVING strategy becomes the best solution against an ERASER. In [85], encryption is used to hide contextual information (*e.g.*, the origin and time of collection of a packet), other than the content of sensed data. The rationale is to prevent the adversary from recognizing target data, forcing him to erase data blindly (like the ERASER attacker). An interesting additional result of this analysis is that public key cryptography allows to obtain the same level of security of continuously moving data, by combining moving data just once and re-encrypting them. Replication is deeply discussed in [86], where a pure controlled epidemic technique is used to provide a trade-off between data survivability, optimal usage of sensor resources, and a fast and predictable collecting time. The authors prove that by estimating the maximal power of an attacker it is possible to set up a probabilistic bound on the survivability of the data. This is the first work in the area that considers the collecting time as an issue; consequently, it might open up a new line of research. The problem with replication is that, while enforcing survivability, it affects data confidentiality. In fact, the more replicas of a data are generated, the more easy is for an active adversary to find (and compromise) a sensor which is storing one of such replicas. Alternative non-cryptographic solutions for secure and distributed storage in UWSNs were investigated in [87]. The authors proposed two algorithms: DS-PADV, to protect against adversaries which do not know where the target information is stored, and DS-RADV, a more secure but burdensome scheme to defend from reactive adversary which choose nodes after identifying the target. However, the most promising solution to ensure both survivability and confidentiality of sensed data in UWSNs is represented by secret sharing based schemes. In [88], the authors showed how a similar solution can maximize communication and storage efficiency and data survival degree. They also introduced an enhanced scheme based on network coding to further improve the power consumption efficiency of communication. The importance of secret sharing schemes for distributed secure storage in UWSNs was however really clarified only in [89] and [90]. The authors proposed a detailed analysis of secret sharing schemes in mobile UWSNs (but the study can be easily adapted to static networks, substituting mobility with data routing). In [89], probabilistic bounds are introduced to predict the amount of sensed data that can be reconstructed, only based on the shares stored by a given portion of the network. Such bounds show that secret sharing can indeed provide the desired trade-off between survivability and confidentiality in UWSNs.

3.2. Intrusion Detection and Recovery

In our definition of the adversary model, we stated that it is necessary to assume that the capabilities of the adversary are limited, in that it can only capture a small number of sensors during each period of absence of the sink. However, if the adversary can keep control of sensors captures previously, he will eventually gain control of the whole network in any case. For this reason it is fundamental to detect intrusions, and to try to recover as many corrupted sensors as possible. Data stored in a corrupted sensor are irremediably lost. However, we can restore a secure keyring to prevent the adversary to access data sensed or received by that sensor in the future/past, or to forge new authenticated data. In other words, we are interested in *backward* and *forward secrecy* of the keys. Forward secrecy can be easily obtained through periodic key evolution [91]. In contrast, backward secrecy is much harder to attain since it relies on a source of randomness that the adversary must not control. Solutions based on asymmetric key pre-distribution have been proposed [92], but their feasibility is limited due to the computational cost of asymmetric cryptography. In [93], the authors introduced scheme called DISH, based on symmetric keys. It leverages sensor collaboration to recover from compromise, and maintains the secrecy of collected data. It provides both backward and forward secrecy using a "sponsor" technique: healthy nodes sponsor sick nodes to make them healthy again. Sponsorship in this context means to supply a pseudo-random value to the sponsored node, which the latter uses to renew cryptographic keys. More precisely, in each round, each node requires values from t sponsors, and it uses these values in the next round to update its own symmetric key. The authors consider a mobile adversary that can compromise up to k nodes in each time interval. Two possible strategies are analyzed: the *Trivial Adversary* and the *Smart Adversary*. The former tries to compromise in each time interval a new set of randomly selected sensors that are not yet compromised. The latter selects the subset of nodes to be compromised in such a way to disrupt the sponsor mechanism: he prefers to compromise the sponsors of a sick node in order to maintain it sick. DISH successfully mitigates the effect of sensor compromise. However, it requires many messages to be exchanged in each round. To overcome this issue POSH was presented in [94]. The idea is similar to DISH, but it differs in one main feature: sponsors push instead of being pulled. In other words, instead of nodes explicitly requiring the contribution of t sponsor nodes, the latter voluntarily send their contributions. In this way, the request messages are not longer used, hence decreasing the overall energy consumption.

Previously cited schemes consider an attacker that can compromise up to a fixed number of sensors in each round, randomly picked in the whole network. A more realistic hypothesis is an adversary that can compromise only sensors within its communication or action range. This adversary is analyzed in [95], where the attacker can control a fixed portion of the network deployment area, and compromise all sensors that move within it following a particular mobility model, such as the random way point, or the random jump. The proposed scheme is based on public key cryptography, but it uses an evolution mechanism based on node collaboration to generate one-time symmetric random keys. In particular, the scheme leverages the mobility of the nodes in a way similar to the push mechanism used in POSH. In each round, nodes broadcast a "contribution" that is then used by their neighbors to calculate the next one-time symmetric random key. Another scheme that uses sensor mobility is the one proposed in [96]. However, in this work a different adversary is analyzed, able to roam the network and choose in each round a new portion of the deployment area to compromise. The proposed protocol is similar to the one presented in [95], but the mobility of the adversary leads to different results. The authors show that the proposed scheme depends on (i) the portion of the deployment surface controlled by the adversary, (ii) sensors mobility model, and (iii) the density of the network. Analyses and simulations show that the best self-healing performances are achieved when adopting a sensor mobility model that provides high variability in sensors neighborhoods.

3.3. Authentication

Authentication for unattended sensors was first investigated in [97], where the authors introduced a technique for data aggregation providing forward-secure authentication. However, the scenario analyzed in [97] in not really a network, since communications among sensors are not considered at all. The first scheme that explicitly provides authentication in UWSNs was proposed in [98]. The authors consider a mobile adversary that attempts to replace authentic data with data of his choice. They introduce two techniques that leverage sensor cooperation, and that rely on symmetric cryptography: Co-MAC and ExCo. In Co-MAC, which stands for “Cooperative MAC”, each information is authenticated either by the node that sensed it, and by a set of co-authenticators. The co-authenticators are selected using a Pseudo Random Number Generator (PRNG), and are required to keep the MACs of all data they authenticated. The PRNG relies on a secret seed shared with the sink, which consequently knows which sensors store the MACs corresponding to any data sensed at any round. ExCo stands for “Extensive Cooperation”, and uses a different approach: sensors do not send their data, but they send the MAC of their data to the co-authenticators. When sensors serve as co-authenticator for multiple MACs, it can bundle all such MACs into a single authentication tag. In both Co-MAC and ExCo, to alter authenticated data, the mobile adversary needs to compromise both the originating sensor and all the co-authenticators. The authors show that the probability of a successful attack rapidly drop as the number of the co-authenticators grows. ExCo was finally extended in [99], introducing a mechanism to dynamically adapt the number of co-authenticators.

3.4. Summary of Security Threats and Countermeasures in UWSNs

In this section, we highlighted several security problems that arise in Unattended WSNs due to the intermittent presence of the sink. We delineated more precisely the typical capabilities of the attacker, cryptographic techniques used, and security requirements emerging in UWSNs. Finally, we reviewed the solutions explicitly designed so far for UWSNs, observing that security cannot really rely on cryptography when the network is constantly exposed, but that important results can be obtained through distributed collaboration and storage.

4. Wireless Mesh Networks

With the continuous development of network infrastructures, Wireless Mesh Networks (WMNs) represent a new and fundamental paradigm to model interaction of different types of networks. WMNs need to provide not only adaptive and flexible wireless connectivity to mobile users, but also integration of other wired and wireless networks. Self-healing, self-organization, auto-configuration and easy deployment are the main features of a WMN. Since these properties are communal to other wireless ad-hoc networks, we will see that many solutions designed for different settings result very useful for WMNs as well.

The distinctive feature of WMNs has to be sought in their architecture, depicted in Figure 2. Firstly, nodes are not homogeneous as in the typical ad-hoc scenario, but a WMN is composed by two distinct sets of nodes: *mesh clients* and *mesh routers*. Only the latter are equipped with a gateway, so clients need routers to gain access to any external network. Typical examples of mesh clients are smartphones in a cellular network, or sensors in a sensor

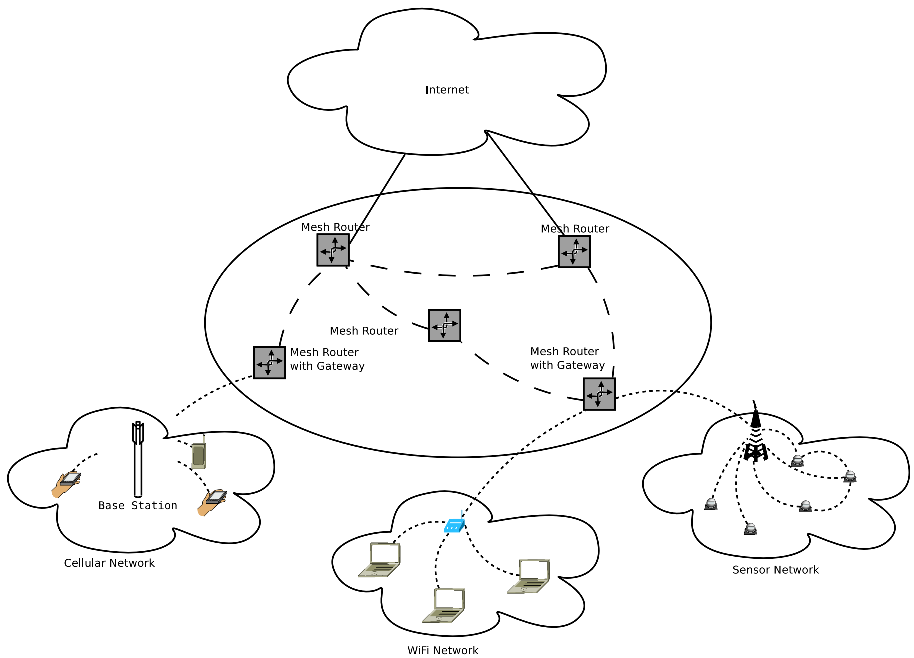


Figure 2: Wireless Mesh Network architecture

network. Mesh routers are instead devices with a minimal mobility and a moderate computational power, but generally not subject to energy constraints. Since routers usually have gateway/bridge functionalities that enable the integration of WMNs with different networks (*e.g.*, WiFi, Cellular, WiMAX, or sensor networks), they need several radio interfaces that use different wireless access technologies. Further, wired clients can use mesh routers connecting to their Ethernet interface. The most common architecture for WMNs is composed by a backbone of mesh routers and many networks of mesh clients, and is denoted *Backbone WMN*. However, mesh clients (without the help of mesh routers) can sometimes communicate in a peer-to-peer fashion, in what is called a *Client WMN*. Client WMN perform routing and provide configuration functionalities as well as end-user applications to other users. A hybrid architecture is also possible, where mesh clients both directly mesh with other clients, or use the backbone to connect to other networks. In this hybrid architecture, clients mesh mechanism improves the coverage and connectivity provided by the backbone.

WMNs are easier to deploy and cheaper than wired networks, though providing good connectivity and large bandwidth to the end users. Therefore, they are attracting considerable attention, especially for many commercial applications. However, they pose two main challenges that still lack of a comprehensive solution: performance when the number of wireless hops increases, and security. The former is promisingly being addressed with novel routing protocols, and multi-radio and multi-channel techniques [100]. Security is instead still not receiving the attention it deserves. In the following, we will describe the main security challenges, as well as the existing countermeasures.

4.1. Similarities with WSNs

Many security aspects of WMNs overlap with those of WSNs. In [101], the authors discuss the common limitations and vulnerable features of WMN and WSN, along with the associated security threats and possible countermeasures. The security challenges that are highlighted are jamming and scrambling, MAC related risks, and routing attacks such as black hole and sleep deprivation to drain the power resources. In [102], security issues in WMNs are investigated, highlighting once again constraints similar to some WSN applications: limited bandwidth, high mobility, and energy and computational constraints of the end nodes. The authors claim that security goals also coincide with those of WSNs: secure routing, intrusion detection systems, and trust and key management. Consequently, they propose to rearrange solutions initially proposed for WSNs or other ad-hoc networks. For a complete survey of these protocols and approaches, the reader can refer to [103].

4.2. Authentication

One of the main requirements of all wireless networks is authentication. On the one hand, solutions based on a Public-Key Infrastructure (PKI) and a single Certification Authority (CA), should be avoided in WMNs. Indeed, it is impractical to establish a single CA that can be trusted by all the nodes of the network.

On the other hand, nodes of a WMN–at least mesh routers–are powerful enough to use public key cryptography, contrary to what happens in sensor networks. An ingenious mechanism that can be used in WMNs to distribute the functionalities of a centralized CA to the whole network is threshold cryptography [104]. Such cryptographic primitive avoids a single point of failure, but allows to tune security upon a threshold t: any t nodes can collectively issue a certificate, but in no way $t-1$ of them can do the same. However, authentication in WMNs must consider that heterogeneous networks may have significant architectural differences. Therefore, WMNs should be able to customize authentication (and, more generally, security) schemes, according to the features of the underlying network clients, but without compromising the overall level of security.

4.3. Routing

Routing is another important issue in WMNs. Indeed, internal or even external attackers can try to induce a misbehavior of one or more nodes, to take advantage of disrupted routing protocols. However, routing in WMNs relies on exactly the same features of other wireless ad-hoc networks: wireless multi-hop links, self-configuration and self-adaptation. Although very few protocols have been proposed specifically for WMNs, the solutions proposed for ad-hoc networks are usually viable for WMNs as well. For example, the IEEE 802.11 standard for wireless LAN mesh networks (802.11s), proposes the well known Ad-hoc On Demand Distance Vector (AODV) protocol [105] as the baseline routing protocol (even if based on a new metric called airtime link metric). Further, more secure protocols can be designed for WMNs upon IEEE 802.11s, like PA-SHWMP [106], which combines a dynamic reputation mechanism with the multi-level security technology, to defend against the attacks caused by compromised nodes while maintaining a reasonable trade-off between security and performance. Even resilience to injection of false routing information and malicious message alteration can be enforce based on algorithms introduced for ad-hoc and sensor networks. Defensive mechanisms performing at any layer of the communication protocol have been proposed, such as ARAN [107], ARIADNE [108], SEAD [109], SAR [110], SAODV [111], SRP [112], etc.[9] Also geographic routing schemes for WMNs may be adopted from ad-hoc and sensor networks. However, since mesh routers are usually static, it is easier to ensure accuracy of routers location necessary for a correct execution of multi-hop routing schemes. A secure routing protocol explicitly designed for WMNs was proposed in [113]. The key contributions of this work are: (i) an accurate estimation of the end-to-end delay in a routing path, used to evaluate the application quality of service; (ii) a link quality estimator, to be used for route selection; (iii) a framework for reliable estimation of the available bandwidth in a routing path, to enable flow admission with guaranteed quality of service; (iv) an improvement of selfish nodes detection and isolation. To better address the

[9] In the literature, several works surveyed these protocols, among which we suggest [3, 4].

issue of selfish nodes in WMNs, the same author proposes a scheme that uses local observations in the nodes for detecting node misbehavior [114]. The scheme is applicable for on-demand routing protocols like AODV, and uses statistical theory of inference and clustering techniques to make a robust and reliable classification (cooperative or selfish) of the nodes based on their neighbors.

4.4. Summary of Security Threats and Countermeasure in WMNs

In this section, we highlighted distinctive and communal elements of WMNs with respect to other ad-hoc networks. Even if many solutions proposed for WSNs can turn useful in WMNs, the integration of many technologies and devices under the same network still leaves many open problems. In particular, when using client meshing, security has to be enforced not only between mesh gateways and heterogeneous client networks, but also between the clients themselves [115]. We briefly described the particular solutions that can or cannot be used in WMNs in the light of the application settings and constraints. However, security in WMNs is a topic that has not yet been completely addressed by the research community.

5. Delay Tolerant Networks

A Delay Tolerant Network (DTN), also called disruption tolerant network, joins locally connected networks using opportunistic (spontaneous) contacts and intermittent interconnectivity. Traditional routing protocols cannot be directly applied in DTNs due to the possible absence of a constant routing path between a source and a destination. Assuming that opportunistic connectivity eventually allows routing data from any source to any destination, the challenge is to understand how to fulfill this goal minimizing delay and security threats.

DTNs were initially developed to support the InterPlanetary Internet (IPN), but were then generalized to many other fields. Traditional models for the Internet rely on the following standard assumptions:

- The existence of a *continuous, bidirectional end-to-end path* between any source and destination;
- The *short duration of round-trips*, with network delay limited to a few seconds or even milliseconds;
- The *symmetry of data rates* in both directions of a network link (asymmetry is possible, but almost always limited);
- The *low error rates* introduced by data routing.

However, these assumptions are rarely concurrently satisfied by contemporary Internet connections, and definitely not satisfied by both the IPN and many WSN scenarios. For example, smartphones can have intermittent connectivity and lower bandwidth than larger devices, a link with a satellite may not be available all day long, or a mobile sensor may be reachable only when it approaches a base station. The TCP/IP protocol that is used on the Internet will not be able to deliver messages to these temporarily unavailable nodes, and it will fail reporting a connection error. Indeed, the Internet is a packet switching network: packets are forwarded from one router to another until they reach their destination. If a path to the destination cannot be found, or if the delay is too long, the connection is aborted. On the contrary, a DTN is an overlay on top of regional networks, including the Internet, that allows communication in the case of intermittent connectivity, long or variable delay, asymmetric data rates and high error rates. For this purpose, it uses a *store-and-forward message switching*: nodes use a persistent storage to temporarily save messages that have to be forwarded to the next hop; when the next hop is available, messages are transfered to the storage device of the next node, until they eventually reach the destination. Messages are deleted from the storage media of a node only when it is sure that they have been transferred to the next node, or when their time to live (usually several hours or days) expires.

The regions that compose a DTN may use different communication protocols with respect to each other, but each of them internally uses a fixed protocol. Communication between regions is implemented leveraging special gateways that connect two or more networks, and translate the traffic from one protocol to another. This translation, and also the store-and-forward mechanism described above, is made possible by using the *bundle layer*. This layer can be seen as a communication layer communal across all DTN regions, and built upon the specific transport layer of each region. A *bundle* is a message composed by three portions: (i) a source-application user data; (ii) control information, provided by the source application for the destination application, describing how to process, store, dispose of, and otherwise handle the user data; and (iii) a bundle header, inserted by the bundle layer. DTN bundle layers communicate with each other using simple sessions with minimal or no round-trips. Any acknowledgement from the receiving node is optional, depending on the class of service selected.

5.1. DTNs Applications

One of the first examples of application of DTNs to wider scenarios is [116]. In that paper, a DTN is designed to allow tracking zebras inside a large, wild area, not covered by cellular service or broadcast communication. The animals wear a collar that includes a Global Positioning System (GPS) and a wireless transmitter. Data are locally stored or moved to other nodes so as to reach a base station as soon as possible. To this end, the authors propose a history-based protocol to identify nodes that registered higher probability of meeting the base station, and to which data should consequently be forwarded. A similar approach is used in [117]. Special nodes called "mules" pick up data from sensors when they are close, buffer them, and then they drop off data to a wired access point. Mules were also used in [118], to provide Internet connectivity to five remote sites in the Swedish mountains. In this case, data mules were set up on board of two helicopters that move daily between an Internet connected host and the five regions. Several other projects test DTN specific ideas on prototypes, mainly focusing on routing aspects of DTNs. For example, in [119] the authors propose a reputation-based protocol for contrasting blackholes. In their scheme, every node locally maintains the reputation of forwarding nodes it comes in touch with, to gradually learn to identify those having the highest reputation. A survey of routing protocols for DTNs can be found in [120] and [121]. Unfortunately, security issues are not taken into account in these works.

Finally, in DTNs applications context-aware mechanisms can help reacting to the changing environmental conditions. Examples of contextual information are (i) the detection of disconnection from known neighbors, (ii) the expected ability of neighbors to deliver a message, (iii) the awareness of neighbors' remaining energy resources and storage space, or (iii) the assignment of priorities to the messages that a node has to deliver. Similar acquired or estimated information can be used to optimize the behavior of DTN mechanisms with respect to metrics such as delivery delay, delivery ratio, traffic overhead and energy consumption [122, 123, 124].

5.2. Security Issues in DTNs

Intermittent connectivity, long or variable delay, asymmetric data rates and high error rates typically noticed in DTNs introduce important security issues. The stressed environment of the underlying networks over which the Bundle Protocol operates makes it important for the DTN to be protected from unauthorized use. Furthermore, DTNs are often deployed in hostile environments, where a portion of the network might become compromised, imposing attention to confidentiality, integrity, and availability.

In DTNs, all couples of nodes (routers and gateway) along a routing path mutually authenticate each other. If a bundle does not pass the authentication check, it is directly discarded [125]. Public-key cryptography is typically used for mutual authentication, users and forwarding nodes having key pairs and certificates. A user certificate also indicates the class-of-service rights of the users: depending on these rights it can require a return receipt, a transfer notification when the bundle is forwarded from one node to another, etc. When the user wants to send a bundle, it signs the bundle itself with its private key. The signature is then checked by the forwarding nodes using the public key of the sender, so as to confirm the authenticity of the sender, the integrity of the message, and the class-of-service rights of the sender. This check is executed in a chain fashion: each forwarding node checks the received signature, and if it is authentic, it replaces this signature with its own signature before forwarding the bundle. In this way, each subsequent forwarding node verifies only the identity of the previous forwarding node, so sender information are implicitly authenticated in a recursive way. A combination of PKI certificates issued by trusted third parties and Certificate Revocation Lists mechanisms is usually assumed. However, this topic in DTNs still needs to be better addressed by researchers. The main reason is surely the disconnected environment typical of DTNs. Each time a signature has to be verified, an end-to-end round trip to a central or replicated lookup database is needed, which delays actual data transmission. When operating across different regions, mutually trusted authorities are required. Furthermore, the management of certificate revocation lists deeply suffers from updates that can be excessively delayed.

A first contribution to developing a practical cryptosystem for DTNs is based on Hierarchical Identity-Based Cryptography (HIBC) [126]. The proposed system is used to create secure channels, to provide mutual authentication, and to allow key revocation. Unlike conventional PKIs, where a user obtains the public/private key pair from a certifying authority, public keys in Identity-Based Cryptography can be any string, but private keys are obtained from a trusted authority called the Private Key Generator (PKG). Hierarchical IBC extends IBC by establishing a cooperative hierarchy of PKGs. In [126], the authors introduce the procedures for initial key establishment and roaming among different regions, and they describe also a simple technique to prevent a user's identity from being compromised due to the loss or theft of a mobile device. Identity Based Cryptography is also used in [127] to provide not only secure communication, but also anonymity. In [128], the authors propose a public key distribution scheme for DTN based on two-channel cryptography. A dynamic virtual digraph (DVD) model is used to study public key distribution with instruments typical of graph theory. By distinguishing between owners and carriers, the proposed scheme realized decentralized public key exchange and authentication. In [129], the use of a PKI is integrated with available social information–knowledge of current and previous affiliations as well as social contacts of peers. The main idea is that some entities have more chances to know the public keys of other entities. This knowledge is used to link a user to a more prominent entity (*e.g.*, an institution or a group of users) that is likely to have a public key already known to the originating user. Social aspects of DTNs are also analyzed in [130] where socially selfish users are considered, who are only willing to forward packets for nodes with whom they have social ties. The authors propose a Social Selfishness Aware Routing (SSAR) algorithm to cope with users selfishness and provide good routing performance in an efficient way. Users' willingness to forward data and their contact opportunity are used as metrics to measure the forwarding capability of the network, and to provide an appreciable trade-off between user demands for selfishness and performance. Social-based routing protocols for DTNs were recently surveyed in [131], highlighting positive and negative social effects.

5.3. Summary of Security Threats and Countermeasures in DTNs

In this section, we provided an overview of the current status of security for DTNs. There are a number of open issues in DTN security. Intermittent connectivity and consequent delays make routing (acknowledgements of successful transmission in particular) extremely challenging. Store-and-forward mechanisms are usually employed, but they require a clever combination of secure storage and secure communication. This can be particularly difficult if the nodes have limited resources, since each security primitive usually involves a delicate trade-off between resource consumption and benefits. The application of cryptographic mechanisms must take into consideration that regional networks may have different limitations and requirements. Equally noteworthy, work on key management is only really beginning recently and standardization is still far. Summing up, security solutions for DTNs are still inadequate, but we pointed out the main directions to follow.

6. Vehicular Ad-hoc Networks (VANETs)

As information and communication technologies (ICT) become increasingly pervasive, vehicles are expected to be equipped in the near future [132, 133] with intelligent devices and radio interfaces, known as On-Board Units (OBUs). OBUs are allowed to talk to other OBUs and the road-side infrastructure formed by Road-Side Units (RSUs). The OBUs and RSUs, equipped with on-board sensory, processing, and wireless communication modules, form a self-organized network with vehicles as nodes, commonly referred to as Vehicular Ad-hoc Network (VANET). Figure 6 depicts a road section with VANET equipment.

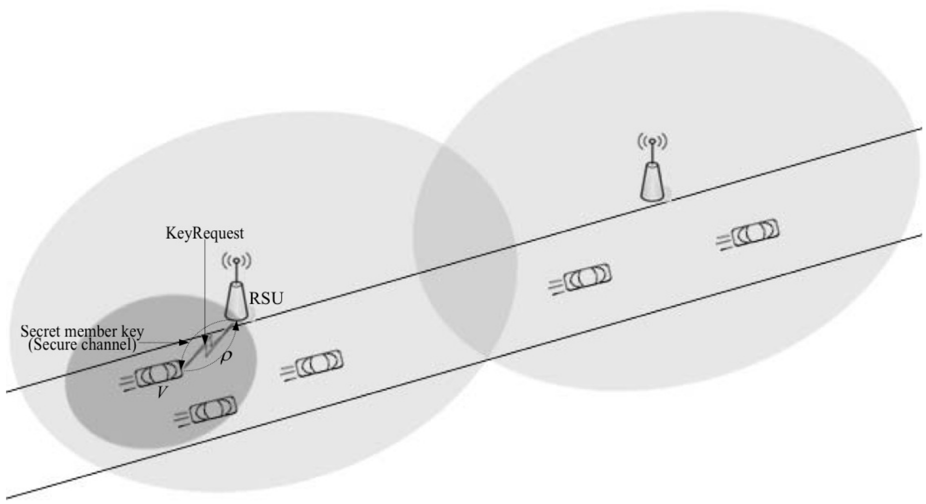


Figure 3: Section of a VANET-enabled road

6.1. Advantages and Problems of VANETs

VANET systems aim at providing a platform for various applications that can improve traffic safety and efficiency, driver assistance, transportation regulation, infotainment, etc. There is substantial research and industrial effort to develop this market. Vehicular communications are supported by the Dedicated Short Range Communications (DSRC) standard [134] in the USA and the Car2Car Communication Consortium [135] in Europe. The U.S. Department of Transportation is investing in the Connected Vehicle Research program (formerly known as IntelliDrive, [136]). In Europe, several projects such as SEVECOM [137] and NoW [138] have been carried out. It is estimated that the market for vehicular communications will reach several billions of euros in the coming years.

The main thrust behind VANET is to improve the safety and the efficiency of traffic. VANETs permit a vehicle to automatically warn nearby vehicles about its movements (braking, lane change, etc.) to avert dangerous situations. These *alert messages* only require a limited dissemination (less than a hundred meters) but have very strong real-time requirements (they must be processed very quickly). VANETs also allow a car to send announcements about road conditions (traffic jams, accidents) to other vehicles so that the latter can take advantage of that information to select routes avoiding troublesome points. Such *announcement messages* require a longer dissemination range. However, their requirement of real-time processing is much less strict than in the case of alerts. These slack time constraints and the computing power of OBUs allow using advanced cryptography to make announcement messages secure and trustworthy.

While the tremendous benefits expected from vehicular communications and the huge number of vehicles are strong points of VANETs, there are still problems to deploy such networks in practice. A very important one is to guarantee the security of vehicle-generated announcements. In what regards security, selfish vehicles may attempt to clear up the way ahead or mess up the way behind with false traffic announcements; criminals being chased may disseminate bogus notifications to other vehicles in order to block police cars. Such attacks may result in serious harm, even loss of lives. Another problem is to protect the privacy of vehicles. VANETs open a big window to observers. It is very easy to collect information about the speed, status, trajectories and whereabouts of the vehicles in a VANET. By mining this information, malicious observers can make inferences about a driver's personality (*e.g.* someone driving slowly is likely to be a calm person), living habits and social relationships (visited places tell a lot about people's lives). This private information may be traded in underground markets, exposing the observed vehicles and drivers to harass (*e.g.* junk advertisements), threats (*e.g.* blackmail if the driver often visits an embarrassing place, like a red-light district) and dangers (*e.g.* hijacks). Finally, VANETs are especially attractive in highly populated urban areas overwhelmed with traffic congestions and accidents. Besides vulnerabilities versus attacks against traffic safety and driver privacy, a large-scale VANET in a metropolitan area raises scalability and management problems.

6.2. Design Goals and Challenges in VANETs

A consequence of the above analysis is that the design goals of VANETs are the following:

- **Security.** The fundamental security functions in vehicular communications consist in ensuring liability for the originator of a data packet. Liability implies that the message originator is held responsible for the message generated. To establish liability without disputes, authentication, integrity and non-repudiation must be provided in vehicular protocols. Authentication allows verifying that the message was generated by the originator as claimed, rather than by an impersonator. Integrity guarantees that the message has not been tampered with after it was sent. Non-repudiation implies that the message originator cannot deny message authorship.

- **Privacy.** In the wireless networks previously described in this paper, privacy refers mostly to confidentiality of the transmitted data. In VANETs the transmitted messages are not private or confidential. Privacy in the VANET context refers to anonymity of the message originator. Hence, there is privacy if, by monitoring the communication in a VANET, message originators cannot be identified, except perhaps by designated parties. Since message authentication requires knowledge of a public identity such as a public key or a license plate, if no anonymity was provided, an attacker could easily trace any vehicle by monitoring the VANET communication. This would be surely undesirable for the drivers. Hence, anonymity should be protected for vehicles *behaving honestly*, that is, not generating untruthful messages. We note in passing that privacy/anonymity is often disregarded as a design goal in this kind of networks, the main focus being on security and scalability (see below).

- **Scalable Management.** For a VANET deployed in a highly populated metropolitan area, managing up to (tens of) millions of vehicles is a substantial concern. Specifically, in such a large VANET, every day some registered vehicles might be stolen or their secret keys might be occasionally leaked. This entails extra burden to manage the system while preserving the liability and the anonymity of vehicles. Hence, it is essential to take the scalable management requirement into consideration when the system is designed.

It is challenging to simultaneously achieve the above design goals. The first challenge derives from the fact that liability and anonymity are conflicting in nature. The liability requirement implies that cheating vehicles distributing bogus messages should be caught. On the other hand, the anonymity requirement implies that attackers cannot trace the original vehicles who generated announcements. Hence, there must be some trade-off between liability and anonymity in a VANET. A well-designed scheme should protect privacy for honest vehicles while allowing the identities of dishonest vehicles to be determined.

Network volatility is another factor that increases the difficulty of securing VANETs. Connectivity among vehicles can often be highly transient due to their high speeds (*e.g.* think of two vehicles crossing each other in opposite directions in a highway). This implies that protocols requiring multiple rounds or strong cooperation such as voting mechanisms may be impractical. Due to their high mobility, vehicles may never again connect with each other after one occasional connection. This puts the public key infrastructure implemented for securing VANETs under strain: if public-key certificates are used, vehicles are confronted to a lot of certificates probably issued by several different certification authorities (CAs); due to mobility, there is little hope that caching the verified certificates of vehicles and CAs will result in any significant speed-up of the next verifications.

The complexity of VANETs deployed in metropolitan areas is another challenge. Transportation systems are governed by a constellation of authorities with different interests, which complicates things. A technically, and perhaps politically, convincing solution is a prerequisite for any security architecture.

Last but not least, the sheer scale of the vehicular network is also challenging: the system has to manage (tens of) millions of nodes of which some may join or leave the VANET occasionally and some may be compromised. This rules out protocols requiring massive distribution of data to all mobile nodes. Furthermore, in case of high vehicular density in metropolitan areas, each node may be flooded with a large number of incoming messages requiring verification.

6.3. Scalability and Service Integrity in VANETs

As mentioned above, scalability is a challenge in VANETs and it has a number of ramifications. The vast number of vehicles and RSUs in a VANET behave simultaneously as information sources and destinations. A way to ensure scalability with the available bandwidth is to *aggregate* the transmitted information as it travels between sources and destinations. In [139] it is proven that any suitable aggregation scheme must reduce the bandwidth at which information about an area at distance d is provided to the cars asymptotically faster than $1/d^2$. Furthermore, the authors show that this bound is tight: for any arbitrary $\epsilon > 0$, there exists a scalable aggregation scheme that reduces information asymptotically like $1/d^{2+\epsilon}$.

When adding security to VANETs (see Section 6.4 below), additional bandwidth is required, because a number of digital signatures need to be appended to each message: one signature for the message originator and possibly another signature for each vehicle endorsing the truthfulness of the message contents. If signatures and the associated public-key certificates are concatenated, as proposed in [140], the size of VANET messages increases linearly with the number of endorsers. If oversignatures are used, *i.e.* each new signature signs previous signatures instead being appended to them, the verifier can only verify the signature by the last signer, but not the previous signatures. In [141], a smart-card based OBU system is proposed whereby the signatures from the originator and the endorsers can be aggregated to save space. In [142], threshold signatures are used which allow combining many partial endorsement signatures into a single standard signature. Nonetheless, the signatures discussed so far require the public-key certificates to be appended to the signatures, which in fact implies a linear growth in message length. Using identity-based cryptography is an effective way to avoid the need of public-key certificates and achieve fixed-length messages (see Section 6.4 below and [143]).

Beyond message aggregation, there are some simple rules to reduce the number of messages generated and verified in a VANET:

- A vehicle should not generate a new message reporting the same information as a message that the same vehicle has previously endorsed;

- A vehicle should not verify a message reporting the same information as a previously verified message.

Since bandwidth is a scarce resource in a VANET, DoS attacks aimed at collapsing the network performance and defeating service integrity are of particular concern. In a DoS attack, the attacker jams the main communication medium and the network is no more available to legitimate users. A DoS attack may be directed at jamming the communication with a specific RSU (vehicle-to-infrastructure or V2I DoS attack) or at jamming the communication medium between the vehicles in an area (vehicle-to-vehicle or V2V DoS attack). Distributed Denial of Service attacks (DDoS) are DoS attacks launched from several locations (usually several vehicles); they are more harmful than DoS by a single vehicle as attackers may co-ordinate and send messages of various types at different times (see [144] for more details on attacks).

6.4. Security and Privacy in VANETs

For VANETs to be viable, the first requirement is to guard them against erroneous information. For example, an attacker may simply put a piece of ice on the vehicle temperature sensor and then a wrong temperature will be reported, even if the hardware sensor is tamper-proof. To counter fraudulent data, detection mechanisms are needed. A general scheme aiming at detection and correction of malicious data was given by Golle *et al.* in 2004 [145]. The authors assume that the simplest explanation of some inconsistency in the received information is most probably the correct one. A specific proposal was made by Leinmüller *et al.* in 2006 [146] focused on verifying the position data sent by vehicles. All position information received from a vehicle is stored for some time period; this is used to perform the checks, the results of which are weighted in order to form a metric on the neighbor's trust. Raya *et al.* [140] and Daza *et al.* [142] introduced a threshold mechanism to prevent the generation of fraudulent messages: a message is given credit only if it was endorsed by a threshold of vehicles in the vicinity.

In addition to guaranteeing correctness of vehicular announcements, VANETs should also provide authentication to establish liability for the prevention, investigation, detection and prosecution of serious criminal offences. To meet this requirement, vehicular communications must be signed to provide authentication, integrity and non-repudiation so that they can be collected as judicial evidence. Several proposals (*e.g.*, [147, 148, 149, 150, 151]) suggest the use of a public key infrastructure (PKI) and digital signatures to secure VANETs. To evict misbehaving vehicles, Raya *et al.* further proposed protocols aimed at revoking certifications of malicious vehicles [152]. A big challenge arising from the

PKI-based schemes in VANETs is the heavy burden of certificate generation, storage, delivery, verification, and revocation.

To guarantee vehicle privacy, some proposals suggest anonymous authentication in VANETs. Among them there are two research lines, *i.e.* pseudonym mechanisms and group signatures.

The pseudonym of a node is a short-lived public key authenticated by a certificate authority (CA) in the vehicular PKI ([153, 154, 155]). The pseudonymity approach mainly focuses on how often a node should change a pseudonym and with whom it should communicate. Sampigethaya *et al.* [156] proposed to use a silent period in order to hamper linkability between pseudonyms, or alternatively to create groups of vehicles and restrict vehicles in one group from listening to messages of other groups. To avoid delivery and storage of a large number of pseudonyms, Calandriello *et al.* [157] proposed self-generating pseudonyms with the help of group signatures locally produced by the vehicles.

One problem with simple anonymity mechanisms in VANETs is the so-called Sybil or "illusion" attack: a single vehicle may abuse anonymity to impersonate several vehicles and generate and provide several endorsements for a message reporting false information. In [142], threshold signatures were used to provide anonymity while thwarting the Sybil attack: at least a threshold amount of partial signatures coming from different groups of vehicles is needed to endorse a message, so that a single vehicle cannot self-endorse a message. Noting that group signatures can be directly used to anonymously authenticate vehicular communications without additionally generating a pseudonym, Guo *et al.* [158] proposed a group signature-based security framework which relies on tamper-resistant devices (requiring password access) for preventing adversarial attacks on vehicular networks. However, neither concrete instantiations nor simulation results are provided. Lin *et al.* [159] introduced a security and privacy-preserving protocol for VANETs by integrating the techniques of group signatures. With the help of group signatures, vehicle-to-vehicle (V2V) communications are authenticated while maintaining conditional privacy. Wu *et al.* [160] distinguished linkability and anonymity of group signatures to improve the trustworthiness of vehicle-generated messages. In fact, group signatures provide the *strongest form of conditional privacy, namely conditional unlinkability*: if two messages bear the same group signature, without co-operation of the group manager, no external observer can decide whether the two messages were signed by different group members or by the same group member; in other words, messages signed by the same group member cannot be linked.

Some recent proposals provide both authentication to establish liability and vehicle privacy in VANETs. When these schemes are implemented in large-scale VANETs in densely populated urban areas, unaddressed challenges remain. Pseudonym-based schemes face the challenge of generating, distributing, verifying and storing a huge number of certificates. Group signature-based schemes in the conventional PKI setting face problems such as how to manage numerous vehicles and especially compromised vehicles. A common concern of both classes of schemes is how to process the large volume of messages received every time unit. These observations call for novel mechanisms to address these challenges in an efficient way. With these challenges in mind, in [143] a set of mechanisms were proposed to address the security, privacy, and management requirements in a large-scale VANET. These conflicting concerns are conciliated by exploiting identity-based group signatures (IBGS) and dividing a large-scale VANET into a number of easy-to-manage smaller groups. In the system, each party, including the group managers (*i.e.* the transportation offices) and the signers (*i.e.* the vehicles), has a unique, human-recognizable identity as its public key, and a corresponding secret key generated by some trusted authority. For instance, the public keys of the administration offices, road-side units [161] and vehicles can be, respectively, the administration name, the RSU geographical address and the traditional vehicle license plate. Certificates are no longer needed because the public key of each party is a human-recognizable identity. This feature greatly reduces the security-related management challenges.

In [143], after registering to transportation offices, any vehicle can anonymously authenticate any message. These vehicle-generated messages can be verified by the identities (*e.g.* the name) of the transportation offices and the public key of the escrow authority. If a message is later found to be false, the identity of the message generator can be traced by traffic police officers. Considering the redundancy in vehicular communications, a selfish verification mechanism is presented to speed up message processing in VANETs. With this technique, although each vehicle may receive a large number of messages, the vehicle only selects for verification those messages affecting its traffic decisions. The selected messages can be verified in a batch as if they were a single one. These speed-up mechanisms are crucial to deploy VANETs in densely populated urban areas.

While group signatures can ensure unlinkability, pseudonyms cannot if they are used more than once: two different VANET messages signed under the same pseudonym are clearly linkable. Using one-time pseudonyms would also provide unlinkability, but changing pseudonyms so often poses an efficiency challenge, because each pseudonym/public key change requires a new private key to be generated. In [162], this challenge was addressed and the authors presented an efficient conditional privacy-preserving protocol for vehicular communications that uses a variant identity-based cryptosystem. Unlike traditional identity-based cryptosystems, this variant requires the master key of the trusted authority to be stored in an *ideally* tamper-proof device embedded into vehicles (that is, such that no attacker can extract any data stored in the device). The fact that each vehicle carries the master key allows efficiently *changing the vehicle pseudonym (pseudo-identity) and private key for each message, which results in unlinkability.*

Yet, the assumption of an ideal tamper-proof device embedded in each car, made in [162], may be too strong to be met in practice. Even if one assumes that an attacker cannot probe into a device, he might collect information related to the master key through powerful side-channel attacks [163, 164], *e.g.*, timing attacks or power analysis attacks. In [165], a protocol called aggregate privacy-preserving authentication (APPA) was proposed to remove the assumption on the existence of ideally tamper-proof devices in each car. It is built on a new notion called one-time identity-based aggregate signature (OTIBAS) and the multiplicative secret sharing technique [166]. The first technique enables vehicles to react to vehicular messages within a very short delay and the seemingly random cryptographic data can be securely and substantially compressed. On the other hand, the multiplicative secret sharing technique can be used to convert a scheme into a leakage resilient one, that is, resistant against side-channel attacks.

However, in [165] the secrets stored in the tamper-proof device cannot be updated, so an *obstinate* attacker might end up learning them. The very recent paper [167] is a follow-up of [165] which further relaxes its tamper-proofness assumption: secrets stored in the tamper-proof device are updated by each RSU once the vehicle enters the RSU's management area. Furthermore, the secrets are only valid for an authorized period; when that period is over, the secrets are deleted. Hence, tamper-proofness is less critical here, as only temporary secrets are stored in the vehicle's device.

6.5. Summary and Further Information

We have briefly described what VANETs are and we have motivated the opportunities and the problems associated with their deployment. While this type of self-organized networks has a big potential to increase traffic safety, it also entails important security, privacy and scalability challenges. Unlike in the other wireless ad-hoc networks previously discussed in this paper, VANET privacy refers to sender anonymity rather than data confidentiality. We have discussed scalability and service integrity, namely how to save bandwidth to improve scalability and how denial-of-service attacks can affect the bandwidth availability in VANETs. Finally, we have ended the section with an overview of the security and privacy solutions for vehicular networks proposed in the literature.

See [133] for a survey of recent developments on vehicle area networks, including VANETs and also intra-vehicle communication. In the `http://vanet.info` web site information on current VANET research and links to important yearly conferences on this topic can be found (*e.g. VNC-IEEE Vehicular Networking Conference*, *ACM VANET*, *Automotive Security*). Important journals in this area are *IEEE Transactions on Vehicular Technology* and *IEEE Transactions on Intelligent Transportation Systems.*

7. Conclusions and Open Research Issues

Wireless ad-hoc networks is an umbrella name that gathers very diverse network technologies with the common features of being self-organized and wireless. These two defining features are the strength and the weakness of such technologies:

- On the positive side, wireless ad-hoc networks are very flexible, relatively cheap and very easily deployable, which explains their great momentum and popularity both for civil and military applications;

- On the negative side, such networks are very vulnerable to attacks against availability, service integrity, security and privacy; indeed, relying on radio communication facilitates eavesdropping, interception and DoS attacks and a self-organized topology without centralized control is prone to attacks against authentication, such as node replication, node suppression, node impersonation, etc.

Beyond the above common pros and cons, there is a great diversity in wireless ad-hoc technologies. At the lower end, we find sensor networks, whose nodes have very limited energy supply and computational power. At the upper end, vehicular networks have vehicles as nodes and the on-board unit of a vehicle is a full-fledged computer with substantial power supply. In spite of the above diversity, data aggregation and encryption turn out to be useful to mitigate the scalability and vulnerability problems of all wireless ad-hoc networks. For low-end networks, symmetric cryptography is the preferred choice, whereas public-key cryptography, including group and threshold cryptography, can be afforded in the high-end networks.

Research challenges depend on each particular network technology and have been identified in the corresponding sections. However, there are a few issues needing further research that pervade several of the described networks. These include making security and privacy compatible with scalability, enhancing bandwidth efficiency, fighting DoS attacks, dealing with node mobility and also reaching worldwide standardization.

[1] A. Hegland, E. Winjum, S. Mjolsnes, C. RONG, O. I. KURE, P. L. SPILLING, A survey of key management in ad hoc networks, IEEE Communications Surveys Tutorials 8 (3) (2006) 48–66.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4020604

[2] E. Cayirci, C. Rong, Security in Wireless Ad Hoc and Sensor Networks, Wiley; 1 edition, 2009.
URL http://www.amazon.com/Security-Wireless-Hoc-Sensor-Networks/dp/0470027487

[3] H. Yih-Chun, A. Perrig, A survey of secure wireless ad hoc routing, IEEE Security & Privacy Magazine 2 (3) (2004) 28–39. doi:10.1109/MSP.2004.1.
URL http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1306970

[4] C. Sreedhar, S. M. Verma, P. N. Kasiviswanath, A Survey on Security Issues in Wireless Ad hoc Network Routing Protocols, International Journal 02 (02) (2010) 224–232.

[5] J. Yick, B. Mukherjee, D. Ghosal, Wireless sensor network survey, Computer Networks 52 (12) (2008) 2292 – 2330. doi:http://dx.doi.org/10.1016/j.comnet.2008.04.002.

URL http://www.sciencedirect.com/science/article/pii/S1389128608001254

[6] IEEE, IEEE Standard 802.15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs), 2006.

[7] ZigBee Alliance, Zigbee specification, ZigBee document 053474r06, version 1.
URL http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:ZIgBee+Specification#0

[8] A. Mpitziopoulos, D. Gavalas, A survey on jamming attacks and countermeasures in WSNs, Surveys & Tutorials 11 (4) (2009) 42–56. doi:10.1109/SURV.2009.090404.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5343062http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5343062

[9] R. Pickholtz, D. Schilling, L. Milstein, Theory of Spread-Spectrum Communications–A Tutorial, IEEE Transactions on Communications 30 (5) (1982) 855–884. doi:10.1109/TCOM.1982.1095533.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1095533

[10] I. Oppermann, L. Stoica, a. Rabbachin, Z. Shelby, J. Haapola, UWB wireless sensor networks: UWEN - a practical example, IEEE Communications Magazine 42 (12) (2004) S27–S32. doi:10.1109/MCOM.2004.1367555.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1367555

[11] W. L. Stutzman, G. A. Thiele, Antenna Theory and Design, 2nd Edition, New York : J. Wiley, 1997.

[12] W. Xu, W. Trappe, Y. Zhang, T. Wood, The feasibility of launching and detecting jamming attacks in wireless networks, Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing - MobiHoc '05 (2005) 46doi:10.1145/1062689.1062697.
URL http://portal.acm.org/citation.cfm?doid=1062689.1062697

[13] A. D. Wood, J. a. Stankovic, G. Zhou, DEEJAM: Defeating Energy-Efficient Jamming in IEEE 802.15.4-based Wireless Networks, 2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (2007) 60–69doi:10.1109/SAHCN.2007.4292818.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4292818

[14] A. D. Wood, J. A. Stankovic, S. H. Son, Jam: A jammed-area mapping service for sensor networks, in: Proceedings of the 24th IEEE International Real-Time Systems Symposium, RTSS '03, IEEE Computer Society, Washington, DC, USA, 2003, pp. 286–.
URL http://dl.acm.org/citation.cfm?id=956418.956593

[15] A. Mpitziopoulos, D. Gavalas, C. Konstantopoulos, G. Pantziou, JAID: An algorithm for data fusion and jamming avoidance on distributed sensor networks, Pervasive and Mobile Computing 5 (2) (2009) 135–147. doi:10.1016/j.pmcj.2008.06.001.
URL http://linkinghub.elsevier.com/retrieve/pii/S157411920800062X

[16] O. Kommerling, M. Kuhn, Design principles for tamper-resistant smartcard processors, in: of the USENIX Workshop on Smartcard, USENIX Association, Chicago, Illinois, 1999, pp. 9–20.
URL http://portal.acm.org/citation.cfm?id=1267117

[17] R. J. Anderson, M. G. Kuhn, Low cost attacks on tamper resistant devices, in: Proceedings of the 5th International Workshop on Security Protocols, Springer-Verlag, London, UK, 1998, pp. 125–136.
URL http://www.springerlink.com/index/5uv5183v0386n75w.pdf

[18] R. Anderson, M. G. Kuhn, Tamper resistance: a cautionary note, in: In Proceedings of the Second Usenix Workshop on Electronic Commerce, USENIX Association, Oakland, California, 1996, pp. 1–11.
URL http://portal.acm.org/citation.cfm?id=1267168

[19] A. D. Wood, J. A. Stankovic, Denial of service in sensor networks, Computer 35 (10) (2002) 54–62. doi:10.1109/MC.2002.1039518.

[20] A. Becher, Z. Benenson, M. Dornseif, Tampering with motes: Real-world physical attacks on wireless sensor networks, Security in Pervasive Computing (2006) 104–118.
URL http://www.springerlink.com/index/1001v40487t55q82.pdf

[21] Y. Law, P. Hartel, J. den Hartog, P. Havinga, Link-layer jamming attacks on S-MAC, in: Wireless Sensor Networks, 2005. Proceeedings of the Second European Workshop on, IEEE, 2004, pp. 217–225. doi:10.1109/EWSN.2005.1462013.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1462013http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1462013

[22] A. Wood, J. Stankovic, A taxonomy for denial-of-service attacks in wireless sensor networks, Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems (2004) 739–763.

[23] F. Stajano, R. J. Anderson, The Resurrecting Duckling: Security Issues for Ad-hoc Wireless Networks, in: Proceedings of the 7th International Workshop on Security Protocols, Springer-Verlag, London, UK, 2000, pp. 172–194.
URL http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:The+Resurrecting+Duckling+:+Security+Issues+for+Ad-hoc+Wireless+Networks#4http://www.springerlink.com/index/a150540577645131.pdf

[24] T. Martin, M. Hsiao, J. Krishnaswami, Denial-of-service attacks on battery-powered mobile computers, in: Second IEEE Annual Conference on Pervasive Computing and Communications, 2004. Proceedings of the, IEEE Computer Society, Washington, DC, USA, 2004, pp. 309–318. doi:10.1109/PERCOM.2004.1276868.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1276868

[25] D. Raymond, R. Marchany, M. Brownfield, S. Midkiff, Effects of Denial-of-Sleep Attacks on Wireless Sensor Network MAC Protocols, IEEE Transactions on Vehicular Technology 58 (1) (2009) 367–380. doi:10.1109/TVT.2008.921621.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4476299

[26] D. R. Raymond, R. C. Marchany, S. F. Midkiff, Scalable, Cluster-based Anti-replay Protection for Wireless Sensor Networks, in: 2007 IEEE SMC Information Assurance and Security Workshop, IEEE, 2007, pp. 127–134. doi:10.1109/IAW.2007.381924.
URL http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=4267552

[27] W. Zhang, N. Subramanian, G. Wang, Lightweight and Compromise-Resilient Message Authentication in Sensor Networks, 2008 IEEE INFOCOM - The 27th Conference on Computer Communications (2008) 1418–1426doi:10.1109/INFOCOM.2008.200.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4509795

[28] A. Perrig, R. Szewczyk, J. Tygar, V. Wen, D. Culler, SPINS: Security protocols for sensor networks, Wireless networks 8 (5) (2002) 521–534.
URL http://portal.acm.org/citation.cfm?id=582464

[29] S. Marti, T. J. Giuli, K. Lai, M. Baker, Mitigating routing misbehavior in mobile ad hoc networks, in: Proceedings of the 6th annual international conference on Mobile computing and networking - MobiCom '00, ACM Press, New York, New York, USA, 2000, pp. 255–265. doi:10.1145/345910.345955.
URL http://portal.acm.org/citation.cfm?doid=345910.345955

[30] C. Karlof, D. Wagner, Secure routing in wireless sensor networks: Attacks and countermeasures, in: Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications, Elsevier, 2003, pp. 113—-127. doi:10.1016/S1570-8705(03)00008-8.
URL http://linkinghub.elsevier.com/retrieve/pii/S1570870503000088http://www.sciencedirect.com/science/article/pii/S1570870503000088

[31] V. Singh, S. Jain, J. Singhai, Hello Flood Attack and its Countermeasures in Wireless Sensor Networks, International Journal of Computer Science 7 (3) (2010) 23.
URL http://www.ijcsi.org/papers/IJCSI-Vol-7-Issue-3-No--11.pdf#page=37

[32] Z. Karakehayov, Using REWARD to detect team black-hole attacks in wireless sensor networks, in: Workshop on Real-World Wireless Sensor Networks, Citeseer, Stockholm, Sweden, 2005.
URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.112.4813&rep=rep1&type=pdf

[33] E. H. Ngai, J. Liu, M. Lyu, On the Intruder Detection for Sinkhole Attack in Wireless Sensor Networks, in: 2006 IEEE International Conference on Communications, Ieee, 2006, pp. 3383–3389. doi:10.1109/ICC.2006.255595.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4024996

[34] D. Dallas, C. Leckie, K. Ramamohanarao, Hop-Count Monitoring: Detecting Sinkhole Attacks in Wireless Sensor Networks, 15th IEEE International Conference on Networks (2007) 176–181doi:10.1109/ICON.2007.4444082.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4444082

[35] A. A. Pirzada, C. McDonald, Circumventing sinkholes and wormholes in wireless sensor networks, in: Conference on Wireless Ad Hoc Networks, 2005.
URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.105.238&rep=rep1&type=pdf

[36] I. Krontiris, T. Dimitriou, T. Giannetsos, M. Mpasoukos, Intrusion detection of sinkhole attacks in wireless sensor networks, Algorithmic Aspects of Wireless Sensor Networks (2008) 150–161.
URL http://www.springerlink.com/index/R1785L8T18773244.pdf

[37] Y.-C. Hu, A. Perrig, D. Johnson, Packet leashes: a defense against wormhole attacks in wireless networks, in: IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and

Communications Societies (IEEE Cat. No.03CH37428), Vol. 3, Ieee, 2002, pp. 1976–1986. doi:10.1109/INFCOM.2003.1209219.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1209219

[38] S. Kaplantzis, A. Shilton, N. Mani, Y. A. Sekercioglu, Detecting Selective Forwarding Attacks in Wireless Sensor Networks using Support Vector Machines, in: 3rd International Conference on Intelligent Sensors, Sensor Networks and Information, Ieee, 2007, pp. 335–340. doi:10.1109/ISSNIP.2007.4496866.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4496866

[39] B. Yu, B. Xiao, Detecting selective forwarding attacks in wireless sensor networks, Proceedings 20th IEEE International Parallel & Distributed Processing Symposium (2006) 8 pp.doi:10.1109/IPDPS.2006.1639675.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1639675

[40] Y. Yu, R. Govindan, D. Estrin, Geographical and energy aware routing: A recursive data dissemination protocol for wireless sensor networks, UCLA Computer Science Department Technical Report, UCLA-CSD TR-01-0023.
URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.21.8533&rep=rep1&type=pdf

[41] J. R. Douceur, The sybil attack, in: Revised Papers from the First International Workshop on Peer-to-Peer Systems, Springer-Verlag, London, UK, 2002, pp. 251–260.
URL http://www.springerlink.com/index/3an0ek5gfan3dtx9.pdf

[42] J. Newso, E. Shi, D. Song, A. Perrig, The sybil attack in sensor networks: analysis & defenses, in: Proceedings of the 3rd international symposium on Information processing in sensor networks, ACM, New York, NY, USA, 2004, pp. 259–268. doi:http://doi.acm.org/10.1145/984622.984660.
URL http://portal.acm.org/citation.cfm?id=984660

[43] C. Wang, K. Sohraby, B. Li, M. Daneshmand, Y. Hu, A survey of transport protocols for wireless sensor networks, Network, IEEE 20 (3) (2006) 34–40.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1637930

[44] B. Hull, K. Jamieson, H. Balakrishnan, Mitigating Congestion in Wireless Sensor Networks, in: Proceedings of the 2nd international conference on Embedded networked sensor systems, ACM, New York, NY, USA, 2004, pp. 134–147.

[45] C. Wan, S. Eisenman, A. Campbell, CODA: congestion detection and avoidance in sensor networks, in: Proceedings of the 1st international conference on Embedded networked sensor systems, ACM, 2003, pp. 266–279.
URL http://dl.acm.org/citation.cfm?id=958523

[46] C. Ee, R. Bajcsy, Congestion control and fairness for many-to-one routing in sensor networks, in: Proceedings of the 2nd international conference on Embedded networked sensor systems, ACM, 2004, pp. 148–161.
URL http://dl.acm.org/citation.cfm?id=1031513

[47] C. Wan, S. Eisenman, A. Campbell, J. Crowcroft, Siphon: overload traffic management using multi-radio virtual sinks in sensor networks, in: Proceedings of the 3rd international conference on Embedded networked sensor systems, ACM, 2005, pp. 116–129.
URL http://dl.acm.org/citation.cfm?id=1098931

[48] A. Woo, D. Culler, A transmission control scheme for media access in sensor networks, in: Proceedings of the 7th annual international conference on Mobile computing and networking, ACM, 2001, pp. 221–235.
URL http://dl.acm.org/citation.cfm?id=381699

[49] P. Levis, N. Patel, D. Culler, S. Shenker, Trickle: A self-regulating algorithm for code propagation and maintenance in wireless sensor networks, in: Proceedings of the 1st conference on Symposium on Networked Systems Design and Implementation-Volume 1, USENIX Association, 2004, pp. 2–2.
URL http://dl.acm.org/citation.cfm?id=1251177

[50] Y. Iyer, S. Gandham, S. Venkatesan, STCP: a generic transport layer protocol for wireless sensor networks, in: Computer Communications and Networks, 2005. ICCCN 2005. Proceedings. 14th International Conference on, IEEE, 2005, pp. 449–454.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1523908

[51] Y. Sankarasubramaniam, O. Akan, I. Akyildiz, ESRT: event-to-sink reliable transport in wireless sensor networks, in: Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing, ACM, 2003, pp. 177–188.
URL http://dl.acm.org/citation.cfm?id=778437

[52] S. Park, R. Vedantham, R. Sivakumar, I. Akyildiz, A scalable approach for reliable downstream data delivery in wireless sensor networks, in: Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing, ACM, 2004, pp. 78–89.
URL http://dl.acm.org/citation.cfm?id=989470

[53] C. Wan, A. Campbell, L. Krishnamurthy, PSFQ: a reliable transport protocol for wireless sensor networks, in: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, ACM, 2002, pp. 1–11.
URL http://dl.acm.org/citation.cfm?id=570740

[54] A. Dunkels, J. Alonso, T. Voigt, H. Ritter, Distributed TCP caching for wireless sensor networks, Tech. rep., SICS Report (2004).
URL http://soda.swedish-ict.se/2344/

[55] H. Zhang, A. Arora, Y.-r. Choi, Reliable bursty convergecast in wireless sensor networks, in: Proceedings of the 6th ACM international symposium on Mobile ad hoc networking and computing, ACM, 2007, pp. 266–276.
URL http://www.sciencedirect.com/science/article/pii/S0140366407002423

[56] F. Yunus, N. Ismail, S. Ariffin, A. Shahidan, N. Fisal, S. Syed-Yusof, Proposed transport protocol for reliable data transfer in wireless sensor network (WSN), in: Modeling, Simulation and Applied Optimization (ICMSAO), 2011 4th International Conference on, IEEE, 2011, pp. 1–7.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5775627

[57] L. Lamport, R. Shostak, The Byzantine generals problem, ACM Transactions on Programming 4 (3) (1982) 382–401.
URL http://dl.acm.org/citation.cfm?id=357176

[58] L. Buttyán, L. Csik, Security analysis of reliable transport layer protocols for wireless sensor networks, in: 8th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops), 2010, 2010, pp. 1–10.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5470633

[59] T. Aura, P. Nikander, J. Leiwo, DOS-resistant authentication with client puzzles, in: Revised Papers from the 8th International Workshop on Security Protocols, Springer, London, UK, 2001, pp. 170–177.
URL http://www.springerlink.com/index/T4DMUWDG8V49LXCL.pdf

[60] P. Ning, A. Liu, W. Du, Mitigating DoS attacks against broadcast authentication in wireless sensor networks, ACM Transactions on Sensor Networks 4 (1) (2008) 1–35. doi:10.1145/1325651.1325652.
URL http://portal.acm.org/citation.cfm?doid=1325651.1325652

[61] M. Ameen, J. Liu, K. Kwak, Security and privacy issues in wireless sensor networks for healthcare applications, Journal of Medical Systems 36 (1) (2012) 93–101. doi:10.1007/s10916-010-9449-4.
URL http://dx.doi.org/10.1007/s10916-010-9449-4

[62] M. Anand, Z. Ives, I. Lee, Quantifying eavesdropping vulnerability in sensor networks, in: Proceedings of the 2nd international workshop on Data management for sensor networks - DMSN '05, ACM Press, New York, New York, USA, 2005, p. 3. doi:10.1145/1080885.1080887.
URL http://portal.acm.org/citation.cfm?doid=1080885.1080887

[63] R. Di Pietro, L. V. Mancini, A. Mei, Energy efficient node-to-node authentication and communication confidentiality in wireless sensor networks, Wireless Networks 12 (6) (2006) 709–721. doi:10.1007/s11276-006-6530-5.
URL http://www.springerlink.com/index/10.1007/s11276-006-6530-5

[64] L. Eschenauer, V. D. Gligor, A key-management scheme for distributed sensor networks, in: Proceedings of the 9th ACM conference on Computer and communications security, ACM Press, New York, New York, USA, 2002, pp. 41–47. doi:10.1145/586115.586117.
URL http://portal.acm.org/citation.cfm?doid=586110.586117

[65] A. Rasheed, R. Mahapatra, The three-tier security scheme in wireless sensor networks with mobile sinks, Parallel and Distributed Systems, IEEE Transactions on 23 (5) (2012) 958–965. doi:10.1109/TPDS.2010.185.

[66] C. Castelluccia, E. Mykletun, G. Tsudik, Efficient aggregation of encrypted data in wireless sensor networks, The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (2005) 109–117doi:10.1109/MOBIQUITOUS.2005.25.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1540992

[67] R. Di Pietro, P. Michiardi, R. Molva, Confidentiality and integrity for data aggregation in WSN using peer monitoring, Security and Communication Networks 2 (2).
URL http://onlinelibrary.wiley.com/doi/10.1002/sec.93/abstract

[68] A. Viejo, J. Domingo-Ferrer, F. Sebé, J. Castellà-Roca, Secure many-to-one communications in wireless sensor networks, Sensors 9 (7) (2009) 5324–5338.

[69] J. Deng, R. Han, S. Mishra, Countermeasures against traffic analysis attacks in wireless sensor networks, in: Proceedings of the First International Conference on Security and Privacy for Emerging Areas in Communications Networks, IEEE Computer Society, 2005, pp. 113–126.
URL http://www.computer.org/portal/web/csdl/doi/10.1109/SECURECOMM.2005.16

[70] A. Wadaa, S. Olariu, L. Wilson, M. Eltoweissy, K. Jones, On providing anonymity in wireless sensor networks, in: Tenth International Conference

on Parallel and Distributed Systems, 2004. ICPADS 2004., Ieee, 2004, pp. 411–418. doi:10.1109/ICPADS.2004.1316121.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1316121

[71] A. Perrig, D. Song, Random key predistribution schemes for sensor networks, Proceedings 19th International Conference on Data Engineering (Cat. No.03CH37405) (April) (2003) 197–213. doi:10.1109/SECPRI.2003.1199337.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1199337

[72] B. Parno, A. Perrig, V. Gligor, Distributed Detection of Node Replication Attacks in Sensor Networks, in: Proceedings of the 2005 IEEE Symposium on Security and Privacy, IEEE Computer Society, Washington, DC, USA, 2005, pp. 49–63. doi:10.1109/SP.2005.8.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1425058

[73] J. Deng, A pairwise key pre-distribution scheme for wireless sensor networks, in: Proceedings of the 10th ACM conference on Computer and communication security - CCS '03, Vol. V, The University of North Carolina at Greensboro, New York, New York, USA, 2005, p. 42. doi:10.1145/948117.948118.
URL http://portal.acm.org/citation.cfm?doid=948109.948118http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:A+Pairwise+Key+Pre-Distribution+Scheme+for+Wireless+Sensor+Networks#0

[74] D. Liu, P. Ning, Establishing pairwise keys in distributed sensor networks, in: Proceedings of the 10th ACM conference on Computer and communication security - CCS '03, ACM Press, New York, New York, USA, 2003, p. 52. doi:10.1145/948117.948119.
URL http://portal.acm.org/citation.cfm?doid=948109.948119

[75] A. Perrig, R. Canetti, J. Tygar, D. Song, Efficient authentication and signing of multicast streams over lossy channels, in: Security and Privacy, 2000. S&P 2000. Proceedings. 2000 IEEE Symposium on, Vol. 28913, IEEE, 2000, pp. 56–73.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=848446

[76] R. Lu, X. Lin, H. Zhu, X. Liang, X. Shen, Becan: A bandwidth-efficient cooperative authentication scheme for filtering injected false data in wireless sensor networks, Parallel and Distributed Systems, IEEE Transactions on 23 (1) (2012) 32–43. doi:10.1109/TPDS.2011.95.

[77] S. Roy, M. Conti, S. Setia, S. Jajodia, Secure data aggregation in wireless sensor networks, Information Forensics and Security, IEEE Transactions on 7 (3) (2012) 1040–1052. doi:10.1109/TIFS.2012.2189568.

[78] S. Nath, H. Yu, P. B. Gibbons, S. Seshan, Synopsis diffusion for robust aggregation in sensor networks, in: IN SENSYS, ACM Press, 2004, pp. 250–262.

[79] R. Di Pietro, L. V. Mancini, C. Soriente, A. Spognardi, G. Tsudik, Catch Me (If You Can): Data Survival in Unattended Sensor Networks, 2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications (PerCom) (2008) 185–194doi:10.1109/PERCOM.2008.31.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4517393

[80] R. Merkle, A certified digital signature, in: Proceedings on Advances in cryptology, Springer-Verlag New York, Inc., Santa Barbara, California, United States, 1989, pp. 218–238.
URL http://www.springerlink.com/index/dyxfp2kd5n6t7fvl.pdf

[81] Nist, FIPS PUB 197: Announcing the Advanced Encryption Standard (AES) (2001).

[82] NIST, SKIPJACK and KEA Algorithm Specifications Version 2.0, Tech. rep. (1998).

[83] C. Karlof, N. Sastry, D. Wagner, TinySec: a link layer security architecture for wireless sensor networks, in: Proceedings of the 2nd international conference on Embedded networked sensor systems, ACM, 2004, pp. 162–175.
URL http://dl.acm.org/citation.cfm?id=1031515

[84] R. Di Pietro, L. V. Mancini, C. Soriente, A. Spognardi, G. Tsudik, Data Security in Unattended Wireless Sensor Networks, IEEE Transactions on Computers 58 (11) (2009) 1500–1511.
URL http://portal.acm.org/citation.cfm?id=1639132

[85] R. Di Pietro, L. V. Mancini, C. Soriente, A. Spognardi, G. Tsudik, Playing hide-and-seek with a focused mobile adversary in unattended wireless sensor networks, Ad Hoc Networks 7 (8) (2009) 1463–1475. doi:10.1016/j.adhoc.2009.04.002.
URL http://linkinghub.elsevier.com/retrieve/pii/S1570870509000341

[86] R. Di Pietro, N. V. Verde, Epidemic data survivability in unattended wireless sensor networks, in: Proceedings of the fourth ACM conference on Wireless network security, ACM, 2011, pp. 11–22.
URL http://ricerca.mat.uniroma3.it/users/dipietro/publications/wisec11-dipietro.pdf

[87] S. Reddy, S. Ruj, A. Nayak, Distributed data survivability schemes in mobile unattended wireless sensor networks, in: Global Communications Conference (GLOBECOM), 2012 IEEE, 2012, pp. 979–984. doi:10.1109/GLOCOM.2012.6503240.

[88] W. Ren, J. Zhao, Y. Ren, Network Coding based Dependable and Efficient Data Survival in Unattended Wireless Sensor Networks, Journal of Communications 4 (11) (2009) 894–901. doi:10.4304/jcm.4.11.894-901.
URL http://ojs.academypublisher.com/index.php/jcm/article/view/2273

[89] R. Di Pietro, S. Guarino, Data confidentiality and availability via secret sharing and node mobility in uwsn, in: INFOCOM, 2013 Proceedings IEEE, 2013, pp. 205–209. doi:10.1109/INFCOM.2013.6566764.

[90] R. Di Pietro, S. Guarino, Confidentiality and availability issues in mobile unattended wireless sensor networks, in: World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2013 IEEE 14th International Symposium and Workshops on a, 2013, pp. 1–6. doi:10.1109/WoWMoM.2013.6583467.

[91] M. Bellare, B. Yee, Forward-security in private-key cryptography, in: Proceedings of the 2003 RSA conference on The cryptographers' track, Springer, San Francisco, CA, USA, 2003, pp. 1–18.
URL http://www.springerlink.com/index/7J2KM6J16V706RY8.pdf

[92] Z. Liu, J. Ma, Y. Park, S. Xiang, Data security in unattended wireless sensor networks with mobile sinks, Wireless Communications and Mobile Computing 12 (13) (2012) 1131–1146. doi:10.1002/wcm.1042.
URL http://dx.doi.org/10.1002/wcm.1042

[93] D. Ma, G. Tsudik, DISH: Distributed Self-Healing, in: SSS '08: Proceedings of the 10th International Symposium on Stabilization, Safety, and Security of Distributed Systems, Springer-Verlag, Detroit, MI, 2008, pp. 47–62.
URL http://portal.acm.org/citation.cfm?id=1484028

[94] R. Di Pietro, D. Ma, C. Soriente, G. Tsudik, POSH: Proactive co-Operative Self-Healing in Unattended Wireless Sensor Networks, in: SRDS '08: Proceedings of the 2008 Symposium on Reliable Distributed Systems, IEEE Computer Society, Naples, Italy, 2008, pp. 185–194.
URL http://portal.acm.org/citation.cfm?id=1475700.1476323#

[95] R. Di Pietro, G. Oligeri, C. Soriente, G. Tsudik, Intrusion-Resilience in Mobile Unattended WSNs, in: 2010 Proceedings IEEE INFOCOM, IEEE Press, San Diego, California, USA, 2010, pp. 1–9. doi:10.1109/INFCOM.2010.5462056.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5462056

[96] R. Di Pietro, G. Oligeri, C. Soriente, G. Tsudik, Securing Mobile Unattended WSNs against a Mobile Adversary, in: 29th IEEE Symposium on Reliable Distributed Systems, IEEE, New Delhi, India, 2010, pp. 11–20. URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5623430

[97] D. Ma, G. Tsudik, Forward-Secure Sequential Aggregate Authentication (Short Paper), in: IEEE Symposium on Security and Privacy, S&P'07, 2007, pp. 86–91. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.112.9520&rep=rep1&type=pdf

[98] R. Di Pietro, C. Soriente, A. Spognardi, G. Tsudik, Collaborative authentication in unattended WSNs, in: Proceedings of the second ACM conference on Wireless network security - WiSec '09, ACM Press, Zürich, Switzerland, 2009, pp. 237–244. doi:10.1145/1514274.1514307. URL http://portal.acm.org/citation.cfm?doid=1514274.1514307

[99] R. Di Pietro, C. Soriente, A. Spognardi, G. Tsudik, Intrusion-resilient integrity in data-centric unattended WSNs, Pervasive and Mobile Computing 7 (4) (2011) 495–508. doi:10.1016/j.pmcj.2010.12.003. URL http://linkinghub.elsevier.com/retrieve/pii/S1574119210001318

[100] L. Badia, M. Conti, S. K. Das, L. Lenzini, H. Skalli, Routing, Interface Assignment and Related Cross-layer Issues in Multiradio Wireless Mesh Networks, in: I. Misra, Sudip and Misra, Subhas Chandra and Woungang (Ed.), Guide to Wireless Mesh Networks, Springer London, 2009, pp. 147–170. doi:http://dx.doi.org/10.1007/978-1-84800-909-7_6.

[101] T. Naeem, K.-K. Loo, Common Security Issues and Challenges in Wireless Sensor Networks and IEEE 802.11 Wireless Mesh Networks, International Journal of Digital Content Technology and its Applications 3 (1) (2009) 88–93. doi:10.4156/jdcta.vol3.issue1.naeem. URL http://nms.dongguk.ac.kr/jdcta/page11.html

[102] M. S. Siddiqui, C. S. V, Security Issues in Wireless Mesh Networks, 2007 International Conference on Multimedia and Ubiquitous Engineering (MUE'07) (2007) 717–722doi:10.1109/MUE.2007.187. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4197357

[103] I. F. Akyildiz, X. Wang, W. Wang, Wireless mesh networks: a survey, Computer Networks 47 (4) (2005) 445–487. doi:10.1016/j.comnet.2004.12.001. URL http://linkinghub.elsevier.com/retrieve/pii/S1389128604003457

[104] Y. Desmedt, Some recent research aspects of threshold cryptography, Information Security (1998) 158–173.
URL http://www.springerlink.com/index/m33tt516174g2706.pdf

[105] C. Perkins, E. Royer, Ad-hoc on-demand distance vector routing, in: Mobile Computing Systems and Applications, 1999. Proceedings. WMCSA'99. Second IEEE Workshop on, IEEE, New Orleans, LA, USA, 1999, pp. 90–100.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=749281

[106] H. Lin, J. Ma, J. Hu, K. Yang, Pa-shwmp: a privacy-aware secure hybrid wireless mesh protocol for ieee 802.11s wireless mesh networks, EURASIP Journal on Wireless Communications and Networking 2012 (1) (2012) 1–16. doi:10.1186/1687-1499-2012-69.
URL http://dx.doi.org/10.1186/1687-1499-2012-69

[107] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, E. M. Belding-Royer, A secure routing protocol for ad hoc networks, in: Proceedings of the 10th IEEE International Conference on Network Protocols, IEEE Computer Society, Washington, DC, USA, 2002, pp. 78–89.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1181388

[108] Y. Hu, A. Perrig, D. Johnson, Ariadne: A secure on-demand routing protocol for ad hoc networks, Wireless Networks 11 (1-2) (2005) 21–38.
URL http://dl.acm.org/citation.cfm?id=1160103

[109] Y. Hu, D. Johnson, A. Perrig, SEAD: Secure efficient distance vector routing for mobile wireless ad hoc networks, in: Proceedings of the 4th IEEE Workshop on Mobile Computing Systems & Applications (WMCSA 2002), 2002.
URL http://www.sciencedirect.com/science/article/pii/S1570870503000192

[110] S. Yi, P. Naldurg, R. Kravets, Security-aware ad hoc routing for wireless networks, in: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing, Long Beach, CA, USA, 2001, pp. 299–302.
URL http://portal.acm.org/citation.cfm?id=501464

[111] M. Zapata, Secure ad hoc on-demand distance vector routing, ACM SIGMOBILE Mobile Computing and Communications Review 6 (3) (2002) 106–107.
URL http://portal.acm.org/citation.cfm?id=581312

[112] P. Papadimitratos, Z. Haas, Secure routing for mobile ad hoc networks, in: SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002), Citeseer, 2002, pp. 1–13.

URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.118.9511&rep=rep1&type=pdf

[113] J. Sen, An efficient and reliable routing protocol for wireless sensor networks, Lecture Notes in Computer Science 6018 (2010) 246–257.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1443555

[114] J. Sen, A Trust-Based Detection Algorithm of Selfish Packet Dropping Nodes in a Peer-to-Peer Wireless Mesh Network, in: Recent Trends in Network Security and Applications, Springer, 2010, pp. 528–537.
URL http://www.springerlink.com/index/W7G1665267R86343.pdf

[115] H. Redwan, K.-H. Kim, Survey of Security Requirements, Attacks and Network Integration in Wireless Mesh Networks, 2008 Japan-China Joint Workshop on Frontier of Computer Science and Technology (2008) 3–9doi:10.1109/FCST.2008.15.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4736502

[116] P. Juang, H. Oki, Y. Wang, M. Martonosi, L. Peh, D. Rubenstein, Energy-efficient computing for wildlife tracking: design tradeoffs and early experiences with ZebraNet, ACM Sigplan Notices 37 (10) (2002) 96–107.
URL http://portal.acm.org/citation.cfm?id=605408

[117] R. C. Shah, S. Roy, S. Jain, W. Brunette, Data MULEs : Modeling a Three-tier Architecture for Sparse Sensor Networks, Proceedings of the First IEEE International Workshop on Sensor Network Protocols and Applications (2003) 30–41.

[118] S. Farrell, Security in the Wild, Internet Computing, IEEE 15 (3) (2011) 86–91.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=5755606

[119] G. Dini, A. L. Duca, Towards a reputation-based routing protocol to contrast blackholes in a delay tolerant network, Ad Hoc Networks 10 (7) (2012) 1167 – 1178. doi:http://dx.doi.org/10.1016/j.adhoc.2012.03.003.
URL http://www.sciencedirect.com/science/article/pii/S1570870512000431

[120] Y. Wang, H. Dang, H. Wu, A survey on analytic studies of Delay-Tolerant Mobile Sensor Networks, Wireless Communications and Mobile Computing 7 (10) (2007) 1197–1208. doi:10.1002/wcm.
URL http://onlinelibrary.wiley.com/doi/10.1002/wcm.519/abstract

[121] Z. Zhang, Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges, IEEE Communications Surveys Tutorials 8 (1) (2006) 24–37.
URL http://www.mendeley.com/research/routing-in-intermittently-connected-mobile-ad-hoc-networks-and-delay-tolerant-networks

[122] C. Mascolo, M. Mirko, SCAR : Context-aware Adaptive Routing in Delay Tolerant Mobile Sensor Networks, in: IWCMC '06: Proceeding of the 2006 international conference on Communications and mobile computing, 2006, pp. 533–538.

[123] G. Sollazzo, M. Musolesi, C. Mascolo, TACO-DTN: a time-aware content-based dissemination system for delay tolerant networks, in: Proceedings of the 1st international MobiSys workshop on Mobile opportunistic networking, ACM, 2007, pp. 83–90.
URL http://dl.acm.org/citation.cfm?id=1247711

[124] B. Pásztor, M. Musolesi, C. Mascolo, Opportunistic mobile sensor data collection with scar, in: Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on, Ieee, 2007, pp. 1–12.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=4428679

[125] K. Fall, A Delay-Tolerant Network Architecture for Challenged Internets, in: Proceedings of the 2003 conference on Applications, technologies, architectures, and protocols for computer communications, 2003, pp. 27–34. doi:http://doi.acm.org/10.1145/863955.863960.

[126] A. Seth, S. Keshav, Practical security for disconnected nodes, in: Secure Network Protocols, 2005.(NPSec). 1st IEEE ICNP Workshop on, IEEE, 2005, pp. 31–36.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1532050

[127] A. Kate, G. M. Zaverucha, U. Hengartner, Anonymity and security in delay tolerant networks, 2007 Third International Conference on Security and Privacy in Communications Networks and the Workshops - SecureComm 2007 (2007) 504–513doi:10.1109/SECCOM.2007.4550373.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4550373

[128] Z. Jia, X. Lin, S.-H. Tan, L. Li, Y. Yang, Public key distribution scheme for delay tolerant networks based on two-channel cryptography, Journal of Network and Computer Applications 35 (3) (2012) 905 – 913, special Issue on Trusted Computing and Communications. doi:http://dx.doi.org/10.1016/j.jnca.2011.03.009.
URL http://www.sciencedirect.com/science/article/pii/S1084804511000634

[129] K. El Defrawy, J. Solis, G. Tsudik, Leveraging social contacts for message confidentiality in delay tolerant networks, in: Proceedings of the 2009 33rd Annual IEEE International Computer Software and Applications Conference - Volume 01, Ieee, 2009, pp. 271–279. doi:10.1109/COMPSAC.2009.43.
URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5254250http://www.computer.org/portal/web/csdl/doi/10.1109/COMPSAC.2009.43

[130] Q. Li, W. Gao, S. Zhu, G. Cao, A routing protocol for socially selfish delay tolerant networks, Ad Hoc Networks 10 (8) (2012) 1619 – 1632, special Issue on Social-Based Routing in Mobile and Delay-Tolerant Networks. doi:http://dx.doi.org/10.1016/j.adhoc.2011.07.007.
URL http://www.sciencedirect.com/science/article/pii/S1570870511001570

[131] Y. Zhu, B. Xu, X. Shi, Y. Wang, A survey of social-based routing in delay tolerant networks: Positive and negative social effects, Communications Surveys Tutorials, IEEE 15 (1) (2013) 387–401. doi:10.1109/SURV.2012.032612.00004.

[132] J. Blau, Car talk, IEEE Spectrum 45 (10) (2008) 16.

[133] M. Faezipour, M. Nourani, A. Saeed, S. Addepalli, Progress and challenges in intelligent vehicle area networks, Communications of the ACM 55 (2) (2012) 90–100.

[134] DSRC-5ghz Band Dedicated Short Range Communications, ASTM E2213-03, http://www.iteris.com/itsarch/html/standard/dsrc5ghz.htm.

[135] Car2Car Communication Consortium, http://www.car-2-car.org/.

[136] U.S. Department of Transportation Connected Vehicle Research program, http://www.its.dot.gov/connected_vehicle/connected_vehicle.htm.

[137] Secure Vehicle Communication, http://www.sevecom.org/.

[138] Network on Wheels, http://www.network-on-wheels.de/.

[139] B. Scheuermann, C. Lochert, J. Rybicki, M. Mauve, A fundamental scalability criterion for data aggregation in VANETs, in: Proceedings of MobiCom 2009-15th Annual Intl. Conf. on Mobile Computing and Networking, ACM, 2009.

[140] M. Raya, A. Aziz, J.-P. Hubaux, Efficient secure aggregation in VANETs, in: ACM International Workshop on Vehicular Ad Hoc Networks-VANET, ACM Press, 2006, pp. 67–75.

[141] A. Viejo, F. Sebé, J. Domingo-Ferrer, Aggregation of trustworthy announcement messages in vehicular ad hoc networks, in: VTC 2009-Spring 69th IEEE Vehicular Technology Conference, 2009.

[142] V. Daza, J. Domingo-Ferrer, F. Sebé, A. Viejo, Trustworthy privacy-preserving car-generated announcements in vehicular ad-hoc networks, IEEE Transactions on Vehicular Technology 58 (4) (2009) 1876–1886.

[143] B. Qin, Q. Wu, J. Domingo-Ferrer, L. Zhang, Preserving security and privacy in large-scale VANETs, in: Information and Communications Security-ICICS 2011, Lecture Notes in Computer Science, Springer, 2011.

[144] I. A. Sumra, I. Ahmad, H. Hasbullah, J.-L. B. A. Manan, Classes of attacks in VANETs, in: 2011 Saudi International Electronics, Communications and Photonics Conference-SIECPC, 2011, pp. 1–5.

[145] P. Golle, D. Greene, J. Staddon, Detecting and correcting malicious data in VANETs, in: ACM International Workshop on Vehicular Ad Hoc Networks-VANET, ACM Press, 2004.

[146] T. Leinmüller, C. Maihöfer, E. Schoch, F. Kargl, Improved security in geographic ad-hoc routing through autonomous position verification, in: ACM International Workshop on Vehicular Ad Hoc Networks-VANET, ACM, 2006, pp. 57–66.

[147] B. Parno, A. Perrig, Challenges in securing vehicular networks, in: 4th Workshop on Hot Topics in Networks-HotNets-IV, 2005, http://conferences.sigcomm.org/hotnets/2005/papers/parno.pdf.

[148] M. E. Zarki, S. Mehrotra, G. Tsudik, N. Venkatasubramanian, Security issues in a future vehicular network, in: European Wireless, 2002, http://www.ics.uci.edu/ dsm/papers/sec001.pdf.

[149] M. Raya, J.-P. Hubaux, The security of vehicular ad-hoc networks, in: ACM Workshop on Security of Ad Hoc and Sensor Networks-SASN, ACM Press, 2005, pp. 11–21.

[150] M. Raya, J.-P. Hubaux, Securing vehicular ad hoc networks, Journal of Computer Security 15 (1) (2007) 39–68.

[151] L. Zhang, Q. Wu, A. Solanas, J. Domingo-Ferrer, A scalable robust authentication protocol for secure vehicular communications, IEEE Transactions on Vehicular Technology 59 (4) (2010) 1606–1617.

[152] M. Raya, P. Papadimitratos, I. Aad, D. Jungels, J.-P. Hubaux, Eviction of misbehaving and faulty nodes in vehicular networks, IEEE Journal of Selected Areas in Communication 25 (8) (2007) 1557–1568.

[153] E. Fonseca, A. Festag, R. Baldessari, R. L. Aguiar, Support of anonymity in VANETs - putting pseudonymity into practice, in: IEEE Wireless Communications and Networking Conference-WCNC, IEEE Press, 2007, pp. 3400–3405.

[154] M. Gerlach, A. Festag, T. Leinmüller, G. Goldacker, C. Harsch, Security architecture for vehicular communication, in: 2nd International Workshop on Intelligent Transportation-WIT 2007, 2007, http://www.network-on-wheels.de/downloads/wit07secarch.pdf.

[155] P. Papadimitratos, L. Buttyan, J.-P. Hubaux, F. Kargl, A. Kung, M. Raya, Architecture for secure and private vehicular communications, in: International Conference on ITS Telecommunications, 2007, pp. 1–6.

[156] K. Sampigethaya, L. Huang, M. Li, R. Poovendran, K. Matsuura, K. Sezaki, CARAVAN: Providing location privacy for VANET, in: Embedded Security in Cars Conference-ESCAR 2005, 2005, http://www.ee.washington.edu/research/nsl/papers/ESCAR-05.pdf.

[157] G. Calandriello, P. Papadimitratos, A. Lioy, J.-P. Hubaux, Efficient and robust pseudonymous authentication in VANET, in: ACM International Workshop on Vehicular Ad Hoc Networks-VANET, ACM Press, 2007, pp. 19–28.

[158] J. Guo, J. Baugh, S. Wang, A group signature based secure and privacy-preserving vehicular communication framework, in: 2007 Mobile Networking for Vehicular Environments, IEEE, 2007.

[159] X. Lin, X. Sun, P.-H. Ho, X. Shen, GSIS: A secure and privacy preserving protocol for vehicular communications, IEEE Transactions on Vehicular Technology 56 (6) (2007) 3442–3456.

[160] Q. Wu, J. Domingo-Ferrer, U. González-Nicolás, Balanced trustworthiness, safety and privacy in vehicle-to-vehicle communications, IEEE Transactions on Vehicular Technology 59 (2) (2010) 559–573.

[161] J.-H. Lee, H. Lee-Kwang, Distributed and cooperative fuzzy controllers for traffic intersections group, IEEE Transactions on Systems, Man and Cybernetics 29 (2) (1999) 263–271.

[162] C. Zhang, R. Lu, X. Lin, P.-H. Ho, X. Shen, An efficient identity-based batch verification scheme for vehicular sensor networks, in: IEEE INFOCOM 2008, IEEE, 2008, pp. 246–250.

[163] P. Kocher, Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems, in: Proceedings of CRYPTO 96, Lecture Notes in Computer Science, Springer, 1996, pp. 104–113.

[164] F.-X. Standaert, T. Malkin, M. Yung, A unified framework for the analysis of side-channel key recovery attacks, in: Proceedings of EUROCRYPT 2009, Lecture Notes in Computer Science, Springer, 2009, pp. 443–461.

[165] L. Zhang, Q. Wu, B. Qin, J. Domingo-Ferrer, Appa: Aggregate privacy-preserving authentication in vehicular ad hoc networks, in: Proceedings of ISC 2011, Lecture Notes in Computer Science, Springer, 2011, pp. 293–308.

[166] E. Kiltz, K. Pietrzak, Leakage resilient ElGamal encryption, in: Proceedings of ASIACRYPT 2010, Lecture Notes in Computer Science, Springer, 2010, pp. 595–612.

[167] L. Zhang, Q. Wu, J. Domingo-Ferrer, C. Hu, B. Liu, B. Qin, Distributed aggregate privacy-preserving authentication in VANETs, manuscript (2014).