# Security in Delay-Tolerant Mobile Cyber-Physical Applications

Naércio Magaia naercio.magaia@tecnico.ulisboa.pt
Paulo Pereira prbp@inesc.pt
Miguel Correia miguel.p.correia@tecnico.ulisboa.pt
INESC-ID, Instituto Superior Técnico, Universidade de Lisboa
Rua Alves Redol 9, 1000-029 Lisboa, Portugal

**Abstract**: In delay-tolerant mobile cyber-physical systems, mobile sensing and computing devices interact following the delay tolerant network (DTN) routing paradigm, storing (or buffering) messages and forwarding them to other nodes until they reach their target, which can be a server or a cloud application, for further processing or storage. Cyber-security threats and the self-organizing nature of DTN environments pose a set of security challenges to the design of delay-tolerant mobile cyber-physical applications. On the one hand, the increase in the interaction between the physical and cyber-systems increases the exposure of physical systems to cyber systems' security threats. On the other hand, finding a suitable node to forward messages may incur in the consumption of nodes' limited resources. Trust establishment and the fairness of contributions among nodes also raise additional concerns in such environments. In this chapter, a comprehensive overview of security mechanisms in delay-tolerant mobile cyber-physical applications is presented. Topics like authentication, confidentiality, integrity, availability, privacy, trust, and cooperation enforcement are analyzed.

## I. Introduction

Unlike conventional embedded systems where the emphasis tends to be on the computational elements, a *Cyber-Physical System* (CPS) [1] is typically designed as a network of physically distributed embedded sensor and actuator devices equipped with computing and communicating capabilities to process and react to stimuli from the physical world and make decisions that also impact the physical world. Example applications of CPSs include high confidence medical devices and systems, assisted living, traffic control and safety, advanced automotive systems, process control, energy conservation, environmental control, avionics, instrumentation, critical infrastructure control (for example, electric power, water resources, and communications systems), distributed robotics (for example, telepresence, telemedicine), defense systems, manufacturing, and smart structures [2].

The broad dissemination of mobile devices with substantial computation resources (e.g., processing and storage capacity), a variety of sensors (e.g., cameras, GPS, speakers, microphone, light and proximity sensors, etc.), multiple communication mechanisms (e.g., Cellular, Wi-Fi, Bluetooth) allowing interconnection to the internet as well as to other devices, made possible the rise of a noticeable subcategory of CPSs, known as *mobile cyber-physical systems* [3]. In mobile CPSs, since CPSs are combined with mobile devices with internet access, tasks requiring more resources than those locally available can now be executed using the mobile link, either to access a server or cloud environment, fostering a new range of applications such as augmented reality, interaction with social networks, health monitoring and so on [3].

*Delay-tolerant mobile cyber-physical applications* are mobile cyber-physical applications that use the *Delay-Tolerant Network* (DTN) [4] paradigm to communicate. DTNs are networks in

which end-to-end connectivity between a source and target nodes might never exist, so nodes have to store (or buffer) data packets and forward them to others until they reach their target. This lack of end-to-end connectivity is the main difference in relation to Mobile Ad hoc Networks (MANETs), where continuous end-to-end connectivity has to be established prior to the forwarding of messages. Due to nodes' mobility and/or network's dynamics (nodes joining or leaving the network, e.g., due to devices turned off or run out of battery), end-to-end connectivity is not guaranteed, even in real MANETs. However, in the DTN routing paradigm (a store-carry-and-forward approach), mobility related issues are no longer seen as obstacles, since nodes can carry messages with them while moving until an appropriate forwarder is found. The DTN routing strategy allows messages to be relayed among nodes until the destination is reached, or they are discarded.

Nevertheless, DTN routing involves the challenging task of finding suitable nodes to forward messages. A variety of network information is used to address this problem, namely: (1) dynamic network information, e.g., location information, traffic information and encounter information; (2) static network information, e.g., social relations among nodes. Static network information, like social ties and behaviors between nodes tend to be more stable over time, and when used they facilitate message transmission [5]. A considerable number of social-aware or social-based routing protocols have been proposed that use static information, i.e., they use social relations among nodes to determine the most appropriate node to forward messages.

In CPSs, due to the increase in the interaction between the physical and cyber systems, physical systems are increasingly exposed to cyber security threats. Similarly, in decentralized and self-organized networks, *security* aspects like confidentiality, integrity, authentication, privacy, trust, and cooperation enforcement arise. To deal with confidentiality, integrity, and privacy, nodes should encrypt or somehow protect information, as forwarding decisions can be made based on the packet's content and/or context. Due to nodes' resource scarcity and to the fact that they are controlled by rational entities, nodes might misbehave. Node misbehavior, malicious or selfish, can significantly impact network performance [6] [7]. Decision making in various fields, e.g., commerce or trade, becomes much simpler with the use of reputation and trust. On the one hand, trust can be seen as the belief a node has in the peer's qualities; on the other hand, reputation can be seen as a peer's perception about a node [8].

The authors of [5] [9] proposed a taxonomy to classify mechanisms to stimulate cooperation among nodes in DTNs, i.e., cooperation enforcement. Cooperation enforcement schemes are categorized as: reputation-based, remuneration-based (also called credit-based) and game theory-based (hereafter called game-based). In *reputation-based schemes*, nodes use others' reputation records to make forwarding decisions. Good or bad reputation is gained by forwarding or not messages from other nodes, respectively. Nodes with bad reputation (i.e., misbehaving) are often excluded from the network. Some form of credit (or reward) is used to control message forwarding in *remuneration-based schemes*. Nodes that forward others' messages are rewarded. The earned credit is used to obtain forwarding services from other nodes. In *game-based schemes*, forwarding decisions are modeled by game theory, where each node follows a strategy aiming at maximizing its benefits and minimizing its resource consumption. That can be accomplished, for example, by maximizing its delivery probability or perhaps by minimizing its end-to-end delay. A well-known strategy is Tit-for-Tat, in which a node forwards as many messages for a neighbor as the neighbor has forwarded its.

Besides selfishness, which is common in self-organized environments, CPSs are prone to the following attacks [10]: eavesdropping, compromised-key attacks, man-in-the-middle attacks, and Denial-of-Service (DoS) attacks. Eavesdropping happens if an attacker intercepts any information communicated by the system. For example, in medical CPS applications, privacy issues arise if patient's health data is disclosed. If an attacker obtains a secret code (key), used to decrypt secure

information, the attacker gains access to secured communications without the perception of the sender or receiver, by means of the compromised key. In a man-in-the-middle attack, falsified messages can be sent to the system that may cause it to go into an incorrect state. An example can be a false message sent to the system control saying that the system is fine, while some actions are required. A DoS attack consists in making the system unavailable.

This chapter is organized as follows. Section II presents security threats of cyber systems and DTNs. Section III presents security requirements of such systems. Section IV presents mechanisms to enhance trust and cooperation enforcement in DTNs. Section V surveys security mechanisms for delay-tolerant mobile cyber-physical applications. Section VI presents some examples of delay-tolerant mobile cyber-physical applications. Finally, Section VII presents concluding remarks.

## II. Security threats

Cyber-Physical Systems (CPS) consist of sensing, processing and communication platforms tightly coupled with physical processes. A CPS is composed of a physical process and a cyber-system, in which the physical system is controlled and monitored by the cyber-system. CPSs often aim to monitor the behavior of a physical process and to trigger actions in order to change its behavior, resulting in a correct and better physical environment. However, due to the interaction between physical and cyber systems, physical systems are exposed to cyber systems' security threats. Some examples of real attacks to CPSs or existing vulnerabilities are: hackers have broken into the air force control mission-support systems of the U.S. Federal Aviation Administration several times [11]; hackers can hack wirelessly networked medical devices implanted in the human body [12]; hackers have penetrated power systems in several regions outside the United States [13]; and CarShark [14] – a software tool – that can kill a car engine remotely, turn off its brakes, make its instruments give false readings, and insert fake data packets to carry out attacks.

Mobile CPSs are those in which CPSs are combined with mobile devices, such as smartphones and tablets, allowing interconnection to the Internet as well as to other devices in a decentralized manner. In some scenarios, these interconnections are made in a delay-tolerant manner, which fostered the appearance of delay-tolerant mobile cyber-physical applications.

In decentralized networks, like DTNs, forwarding decisions are made individually by each node (or entity), which may incur in the consumption of nodes' limited resources, e.g., battery power, bandwidth, processing, and memory, all over the network. These networks share a tricky notion of being self-organized and/or self-managed, but they require for their correct operation that each node gives its own contribution[1]. Some concerns arise in these networks (1) in the establishment of trust between entities, or (2) to stimulate their cooperation, or even (3) due to the fairness of their contributions.

There are two possible types of nodes' misbehavior: selfish and malicious. When a node manifests a *selfish* behavior, it aims to maximize its benefits by using the network while saving its own resources (e.g., battery power). As such, cooperation enforcement schemes can be leveraged to foster cooperation. If a node manifests a *malicious* behavior, it tends to maximize the damage caused to the network for its own benefit. A way to deal with such misbehavior is by detecting and isolating those nodes from the network.

Selfish behaviors can be classified as individual or social selfishness [5] [9]. A node presents individual selfishness if it only aims at maximizing its own utility, hence disregarding a system-wide criteria. Social selfishness is manifested when nodes only forward messages of others to whom they have social ties with.

[1] It is assumed that nodes have equivalent privileges and responsibilities.

Table 1
A summary of potential attacks to delay-tolerant mobile cyber-physical applications

| *Attack type* | *Attack Description* |
|---|---|
| Individual selfishness [5] [9] | A selfish node only aims at maximizing its own utility, disregarding the system-wide criteria. |
| Social selfishness [5] [9] | Nodes only forward messages of others to whom they have social ties with. |
| Eavesdropping [15] [16] | Message content is made available or disclosed to unauthorized parties. |
| Compromised-key attack [10] | An attacker obtains a cryptographic key and uses it to read an encrypted communication. |
| Traffic analysis [15] [17] | Extracting unauthorized information by analyzing communication patterns (but not their content). |
| Routing loop attacks [22] | Modifying routing packets so they do not reach their destination. |
| Wormhole attacks [22] | A set of malicious nodes creates a worm link to connect distant network points with low-latency, causing disruption in normal traffic load and end flow. |
| Black-hole attacks [22] | A malicious node drops all packets forwarded to it and responds positively to incoming route requests despite the fact of not having proper routing information. |
| Gray-hole attacks [22] | Special case of a black-hole attack where a malicious node selectively drops packets. |
| Denial-of-Service (DoS) [22] | Attacks that obstruct the normal use or management of a service. |
| False information or false recommendation [22] | Colluding and providing false recommendation/information in order to isolate good nodes while keeping bad ones connected. |
| Incomplete information | Consists in not cooperating to provide proper or complete information. |
| Packet modification/insertion | Consists in the modification or malicious insertion of packets. |
| Newcomer attacks [22] | A malicious node registers as a new user to discard its bad reputation or distrust. |
| Sybil attacks [22] | Consists in using multiple network identities. |
| Blackmailing [22] | Consists in using a majority-voting scheme trying to cause routing topology change. |
| Replay attacks [22] | Consists in maliciously or fraudulently repeating or delaying valid transmitted packets. |
| Selective misbehaving attacks [22] | Consists in selectively misbehave to other nodes. |
| On-off attacks [22] | Disrupting services by behaving correctly/incorrectly in alternation. |
| Conflicting behavior attacks [22] | Behave differently to different nodes to cause contradictory opinions. |
| Credit forgery attack (or layer injection attack) [50] | Forge valid credit in order to reward itself for work it did not do or for more than it has done. |
| Nodular tontine attack (or layer removal attack) [50] | Remove one or more layers of a multilayer credit generated by previous forwarding nodes. |
| Submission refusal attack [50] | If the source and last intermediate node collude, the latter may refuse to submit the received credit from a virtual bank, and receive another compensation from the source node. |

Besides selfishness, entities on a self-organized environment are also prone to other forms of attacks such as flooding and cheating. A flooding attack consists in exhaustively using others' resources, e.g., by initiating an enormous amount of requests, in order to render the network useless. A cheating (or retention) attack consists in gaining an unfair advantage over other nodes by holding essential system's data.

Yet another way of classifying attacks is based on the nature of the attacks and the type of attackers. As such, they can be classified as passive or active [15]. Passive attacks happen if an unauthorized party gains access to a message without modifying its contents. There are two types of passive attacks, namely: (1) the release of message contents [16], which happens if message contents are made available or disclosed to unauthorized parties, and (2) traffic analysis [17], which allows an attacker to infer communication patterns, thus guessing the nature of the communication that was taking place. Active attacks are characterized by an unauthorized party modifying the contents of the message.

Table 1 summarizes potential attacks to delay-tolerant mobile cyber-physical applications. As an example, Figure 1, as in [18], shows the average delivery probability (i.e., the fraction of successfully received packets over all packets sent) for 5 different percentages of nodes performing a black-hole attack on a map-based mobility model [19] of the Helsinki City, for 8 DTN routing protocols, namely, Direct Delivery (DD), First Contact (FC), Epidemic, PRoPHET, MaxProp, RAPID and Spray and Wait (SnW). DD it is not affected by black-hole attacks, as it can only deliver a message if it meets the final destination. SnW delivers more messages than DD as it sprays multiples copies to intermediate nodes which may come in contact with the destination node sooner. Naturally, it is vulnerable to attacks by relying on other nodes for forwarding messages. As FC forwards only one copy of the message to the first node met, it is the most affected routing protocol by black-hole

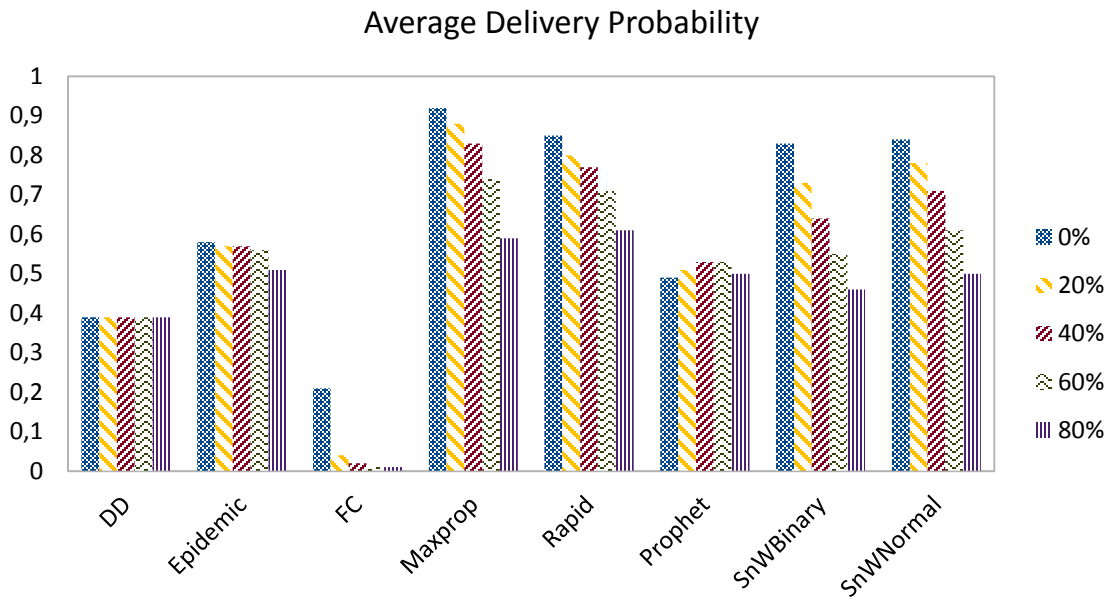


Figure 1 Average delivery probability as function of the percentage of nodes performing a black-hole attack.

attacks. It was also observed that black-hole misbehaving nodes reduce congestion (e.g., PRoPHET with less than 60% of misbehaving nodes), i.e., misbehaving nodes cause a reduction of the number of message copies circulating in the network as they drop them. Epidemic and Prophet are less affected in comparison with other routing protocols by black-hole misbehaving nodes, due to their unlimited copy algorithms. But, MaxProp and Rapid also suffer from black-hole attacks because of the contact pattern (i.e., shorter contact durations) since longer messages require more contacts to be transmitted.

## III. Security Requirements

As a wireless medium is accessible (or open) for everyone within communication range, misbehaving nodes might attempt to compromise message contents, justifying the existence of security requirements [15] such as authentication, confidentiality, integrity, availability and privacy, in such environments.

### *A. Authentication*

*Authentication* assures that the communication is genuine. There are two possible cases: (1) the single message case, e.g., a warning message, where the objective is to assure the source's legitimacy to the message's destination, i.e., that the message is from the source that it claims to be from; (2) the ongoing interaction case, e.g., the connection of an entity to another, where the objective is twofold: first, it is necessary to assure the entities' authenticity, i.e., that each entity is who it claims to be, and second, it is also necessary to assure that a third party cannot interfere with the connection, even if masquerading as one of the legitimate parties. In a CPS, authentication assures the authenticity of all the related processes such as sensing, communication and actuation.

### *B. Confidentiality*

*Confidentiality* assures the protection of the transmitted data from passive attacks, since, and as previously stated, the wireless medium is accessible for everyone within communication range.

Regarding the content of the data transmission, several levels of protection can be identified, in which the broadest level assures the protection of all user data transmitted between two nodes over a period of time [15].

Confidentiality may also aim to assure protection against traffic analysis. The idea is that an attacker shall not able to perceive traffic flow information such as the source and destination, frequency, length, or even other traffic's characteristics.

As an example, consider a delay-tolerant mobile cyber-physical application, such as health monitoring, in which the personal health records need to be transmitted from the Personal Health Record system to the doctor or the front medical devices. By means of confidentiality, the system prevents third parties from inferring data about the patients' personal health records either by reading their content or by understanding to what kind of doctor it is being sent to.

### *C. Integrity*

*Integrity* assures that the transmitted data is received as sent, without duplication, insertion, modification, reordering, or replay. Similarly to confidentiality, integrity can apply to: (1) a messages' stream, (2) a single message, or even (3) certain fields of the message. And as before, the complete stream protection is the best approach.

There are two types of integrity [15]: (1) connection-oriented and (2) connectionless. The connection-oriented integrity type addresses both message stream modification and denial-of-service. The destruction of data is also covered under this type. The connectionless integrity type deals with individual messages without considering any larger context. It usually only offers protection against the modification of messages.

Integrity is compromised in a CPS when, for instance, false data is received by the system control stating that the system is fine, whereas urgent actions are required.

### *D. Availability*

*Availability* refers to a system's property of being accessible and usable when requested by an authorized entity, in accordance to the system's design. Loss or reduction of availability can be caused by a variety of attacks. An example of such attacks is DoS (see Table 1). With automated countermeasures, e.g., authentication and encryption, or some sort of physical action, it is possible to prevent/recover from the loss of availability resulting from such attacks. For example, consider a smart grid infrastructure where information such as price information, control commands and meter data flows to/from a smart meter. In such scenario, availability of price information and control commands is critical. The former, because of the serious financial and possible legal implications, and the latter, as it is necessary to turn the meter back on after completing the payment of the electric bill. Instead, the availability of the meter data may not be critical as the data can be accessed later on.

### *E. Privacy*

*Privacy* can be understood as the users' willingness to disclose or not its information, to others (family, friends, or even the general public). As a property, privacy is the confidentiality of personal information.

The lack of infrastructure in decentralized and self-managed environments leverages nodes' forwarding decisions, allowing the opportunist exploitation of any other nodes in their vicinity to help messages reach their intended destinations. In such environments, message forwarding relies on the nodes' participation in the network.

If some nodes are deemed untrusted, privacy issues may rise due to passive attacks, which are in the nature of eavesdropping on transmissions.

## IV. Trust and Cooperation enforcement schemes

In this section, a survey of mechanisms to foster cooperation in decentralized and self-managed networks, such as DTNs, is presented.

### *A. Trust*

According to the *Oxford Dictionary of English* [20], trust is a "firm belief in the reliability, truth, or ability of someone or something". In the literature, many other definitions of trust can be found according to various disciplines. For example, in line with *social sciences* [21], trust can be defined "as the degree of subjective belief about the behavior of a particular entity"; in line with *economics* – as shown in the Prisoner's Dilemma – trust is based on the assumption that humans are rational and strict utility maximizers of their own interest, therefore selfish. Even though, the emergence of altruistic behavior can be seen in initially purely selfish mechanisms. In line with *communication and networking* [22], trust relationships among participating nodes are important, as with them cooperative and collaborative environments can be built, which improve system objectives in terms of scalability, re-configurability, reliability, dependability, or security.

Trust management is necessary in order for participating nodes without previous interactions to form a decentralized and self-organized network (e.g., MANET or DTN) with an acceptable level of trust relationships between them [22]. As stated in [23], "trust management provides a unified approach for specifying and interpreting security policies, credentials, and relationships." Some applications of trust management in decision-making situations are intrusion detection, authentication, access control, key management, and the isolation of misbehaving nodes for effective routing [22].

*Trust management* is composed of trust establishment, trust update and trust revocation. A *trust establishment* process consists of the representation, evaluation, maintenance, and distribution of trust among nodes. A *trust update* process consists in modifying a node's trust values as a consequence of their collaboration with others, thus favoring trustworthy nodes and penalizing untrustworthy ones. *Trust revocation* consists in dropping/cancelling trust relationships among nodes.

In general, the main properties of trust in decentralized and self-managed environments [22] are: (1) *dynamicity*, since trust establishment is based on temporally and spatially local information; (2) *subjectivity*, due to network's dynamics, a trustee node may be assigned different levels of trust as a result of different experiences; (3) *incomplete transitivity*, trust's transitivity among two entities (trustor and trustee) and a third party is guaranteed if a trustor trusts the trustee and the trustee's recommendation of the third party; (4) *asymmetry*, entities with different capacities (e.g., energy or computation power) may not trust each other; (5) *context-dependency*, trust types depend on the foreseen task.

The properties presented above should be taken into account during the design of a trust-based framework. Other important aspects to consider for decentralized and self-managed environments are: (*i)* an entity decision procedure of trust should be *fully distributed* and based on a cooperative evaluation with uncertainty and incomplete evidence, since a Trusted Third Party (TTP) is unreliable in such environments; (*ii*) trust's determination should be flexible to membership changes and to deployment scenarios, therefore in a *highly customizable manner*; (*iii*) *selfishness* should be taken into account by a trust decision framework, hence no assumption that all nodes are cooperative should be made; (*iv*) also because of network's dynamics, trust should be established in a *self-organized and reconfigurable manner*.

Additional care should be taken to ensure that a trust management system is not easily subverted, attacked or compromised. It is important to mention that trust management schemes

are devised to detect misbehaving nodes (i.e., selfish nodes along with malicious ones). In addition, if the available information or evidence does not provide a certain level of trust, the trust engine should be robust enough to gracefully degrade.

### *B. Cooperation enforcement*

Cooperation enforcement [24] (incentive schemes/mechanisms) can be leveraged to manage and organize decentralized and self-managed systems, therefore compensating for the nonexistence of a central or dedicated entity. As a consequence, it is possible to deal with the security challenges previously mentioned. Even though, a cooperative behavior may result in an increase of the nodes' resource consumption, e.g., forwarding nodes may incur in additional energy and bandwidth usage during packets' transmissions and receptions. It was demonstrated in the context of MANETs that cooperation can succeed over competition [25]. So, the idea is to guarantee that a cooperative behavior is overall more beneficial than a passive or malicious uncooperative behavior. It is often desirable that these mechanisms distinguish uncooperative behaviors due to valid reasons such as energy shortage, crashing, etc., from malicious uncooperative ones.

The use of cooperation enforcement schemes in MANETs has been exhaustively researched [26] [27] [28] [29] [30] [31] [32] [33]. Cooperation enforcement schemes have been envisaged for many application domains, namely infrastructure-based P2P applications (e.g., file sharing, distributed processing, and data backup), wireless networks (e.g., DTNs, Wireless Sensor Networks (WSNs), MANETs, wireless ad hoc backup, and nomadic computing) and Web commerce (e.g., auction sites, review and recommendation sites) [24].

Cooperation enforcement schemes can be categorized as reputation-based, remuneration-based and game-based schemes. The following sections discuss in detail each one of these categories.

#### *1) Reputation-based schemes*

Reputation-based schemes are those in which the decision to interact depends on the other node's reputation.

*Architecture*. The architecture of the reputation management system, an integral part of the reputation mechanism, can be centralized, decentralized or hybrid [24]. In the centralized approach, a central authority collects nodes' information, derives and provides the scores for all participants. In the distributed approach, since no central authority is available, nodes' ratings are stored in a distributed fashion. The evaluation of reputation is commonly based on subsets of information (e.g., neighbor nodes information), which may be prone to inconsistencies if compared to the centralized approach. Nevertheless, a distributed management system scales better, and is more common in self-organized networks, than the centralized one. A hybrid approach is a combination of both.

*Operations*. Reputation-based mechanisms are composed of three phases: collection of evidence, cooperation decision and cooperation evaluation [24]. In the first phase, a node collects reputation information by observing, experiencing and/or by means of recommendations of third parties. In the second phase, a node evaluates the collected information in order to decide if it should cooperate or not, based on the other node's reputation. Some evaluation methods used in this phase are: voting schemes, average ratings, Bayesian based computation, flow mode, etc [24]. In the last phase, the degree of cooperation between nodes is evaluated. It is done after their interaction, and consists in rewarding nodes that presented a good behavior by adequately increasing their local reputation. Consequently, nodes with bad reputation are isolated, hence not receiving others' services.

*Attacks and counter-measures*. Misbehaving nodes can cause a variety of attacks such as individual and social selfishness, DoS, functionality attacks (e.g., subversion attacks), and single or group attacks to the reputation system (e.g., a liar or collusion[2], respectively). For example, the CORE [34] mechanism can be used to guard against the impact of liars. In [35], the Watchdog and Pathrater are used to identify misbehaving nodes and selecting paths to avoid them, respectively. For a more detailed discussion of attacks and countermeasures, please refer to [36].

*2) Remuneration-based schemes*

Remuneration-based schemes are those in which cooperating nodes should receive an equivalent complement (remuneration), and misconduct is punished with a penalty. The exchange of services, for some sort of payment, calls for a TTP (e.g., bank) to manage the process.

*Architecture*. A remuneration-based mechanism is composed of four operations: negotiation, cooperation decision, cooperation evaluation and remuneration [24]. During negotiation, nodes decide on the terms of their interaction. This can be done only between them, or between them and a TTP. The cooperation decision, i.e., if a node can or not cooperate, is taken based on the outcome of the negotiation. The cooperation evaluation is twofold: on the one side, the service requesting party decides based on the acceptability of the service to the request; on the other side, the service providing party, decides based on the acceptability of the remuneration. At last, the collaborating node is remunerated. There are three types of remuneration envisaged in this scheme: virtual currency units, real money and bartering units [24].

*Fair exchange*. A fair exchange protocol offers ways to guarantee that at the end of the exchange held by two or more nodes, either all of them have received what they were expecting or none of them has received anything. The correctness of the exchange depends on the availability of a neutral TTP. There are two types of protocols: online and offline fair exchange protocols [24]. In the former, the TTP constitutes a bottleneck as it mediates every interaction between the nodes. In the latter, the TTP is used as an intermediary if and only if one of the nodes has doubts about the fairness of the exchange.

*3) Game-based schemes*

Game-based schemes are those in which forwarding decisions are modeled using game theory. Game theory provides guidelines on how to model situations such as social dilemmas (e.g., the prisoner's dilemma). It also provides insights on how individual node-to-node interactions, without a centralized entity, can still spawn cooperation towards a more efficient outcome. Classical (rational) game theory assumes that nodes (i.e., players) have well-defined and consistent goals, that can be described by an utility function, which can be seen as a measure of the players' satisfaction resulting from a certain game outcome. So, in these schemes, each node follows a strategy aiming at maximizing its benefits and minimizing its resource consumption. For example, a node decides to forward a message if the (direct or indirect) result of that action maximizes its delivery probability, or possibly it minimizes its end-to-end delay.

Below, some definitions are presented.

*Definition 1. Game*. A game constitutes a formal description of a strategic interaction between players.

[2] According to the *Oxford Dictionary of English* [20], collusion is a secret or illegal cooperation or conspiracy in order to deceive others.

*Definition 2. Player*. A player is an entity entitled to its own decisions and subsequent actions. It can also be interpreted as a node or as a group of nodes making decisions.

*Definition 3. Action*. An action is the act of performing a move in the game.

*Definition 4. Payoff*. A payoff (or utility), typically represented by a (positive or negative) number, reflects the desirability of an outcome to a player. As a consequence, it incorporates the player's attitude towards the risk.

*Definition 5. Strategy*. A strategy represents a set of actions that can be performed by the player during the game.

*Definition 6. Payoff Matrix*. A payoff matrix is a matrix that represents: the players, their strategies and the payoffs for each player with every possible combination of actions.

A plethora of ways to classify games is available nowadays [37] [38]. Some classification examples are: (1) according to the level of cooperation; (2) according to the symmetry of the payoff matrix (or according to the dependency between the strategy and the player); (3) according to the sum of the players' payoffs; (4) according to the amount of information known in advance; (5) according to how the plan of action is chosen; and (6) according to the number of players.

In relation to (1), games can be classified as cooperative or non-cooperative. In non-cooperative games, the game describes (or focuses on) the strategy of the node, where a node has to make a decision if it is going to cooperate or not with another random node. If the cooperation decisions are taken by a group of nodes, these types of games are called cooperative games. In relation to (2), games can be classified as symmetric or asymmetric. Symmetric games (also called matrix games) are those in which the strategy options and payoffs do not depend on the players, but only of the other strategies employed. All players have the same strategy set [39]. The games where the strategies of the players are not identical are known as asymmetric ones (also called bi-matrix games). In relation to (3), games can be classified as zero sum or non-zero sum games. In zero-sum games, the sum of all players' payoffs is zero, for any possible outcome. Thus, a player's benefit is equal to the loss of other players. However, if for any outcome, the sum of all players' payoffs is greater or less than zero, they are called non-zero sum games. In relation to (4), games are classified as of perfect or imperfect information. If all previous players' moves are known, the game is of perfect information. A concept similar to a perfect information game is a complete information game, in which all players' strategies and payoffs are known, excluding their actions. In relation to (5), games can be classified as strategic or extensive. If each player chooses (once and for all) its action's plan, and all players' decisions are simultaneously made, it is called a strategic game. In extensive games, each player can choose its action's plan while taking decisions, which does not have to happen at the beginning of the game. In relation to (6), games can be classified as two-player or multi-player. As the name suggest, the two-player games are those played by exactly two players. Instead, if there are more than two players, the game is called a multi-player one.

*a) Social dilemmas*

Classical game theory is based on the following key assumptions: (1) player's perfect rationality, i.e., players have well-defined payoff functions, being fully aware of their own and their opponents' strategy options and payoff values; and (2) that this is common knowledge, meaning that all players are aware of their own rationality, and of the rationality of other players; and that all players are

aware that all players are aware that all are rational, etc., ad infinitum. A strategy profile of a game is said to be a Nash Equilibrium (NE), if and only if no player has an unilateral incentive to deviate and play another strategy, since there is no way it could be better off given the others' choices.

Social dilemmas occur in certain situations where the game has a single NE, which is not Pareto efficient[3] so that the sum of individual utilities (social welfare) is not maximized in equilibrium. One of game theory's main tasks is to provide guidelines on how to solve social dilemmas, and to provide insights on how player-to-player interactions (excluding the intervention of a central entity) may still generate an aggregate cooperation towards a more efficient outcome in many real-life situations.

The most popular dilemmas of cooperation are: the snowdrift game, the stag-hunt game, and the prisoner's dilemma [40].

*Stag-Hunt*. In *A Discovery in Inequality* of 1755, Rousseau described the Stag-Hunt game's story. In it, each hunter prefers stag *S* over hare *H*, and hare over nothing. According to game theory, the highest income is achieved if each hunter chooses hunting stag, but the chance of a successful stag hunt increases with the number of hunters. So, there is nearly no chance of catching a stag alone, but the odds of getting a hare does not depend on others. Hence, the payoff matrix for two hunters is given by:

| | | $Hunter_2$ | |
|---|---|---|---|
| | | *H* | *S* |
| $Hunter_1$ | *H* | *(1,1)* | *(2,0)* |
| | *S* | *(0,2)* | *(3,3)* |

*Prisoner's Dilemma*. In 1950, Tucker aiming at showing the difficulty of analyzing certain kinds of games previously studied by Dresher and Flood, came up with the Prisoner's Dilemma story: "Two burglars ($B_1$ and $B_2$) are arrested after their joint burglary and held separately by the police. However, the police does not have sufficient proof in order to have them convicted, therefore the prosecutor visits each of them and offers the same deal: if one confesses (called *defection D* in the context of game theory) and the other remains silent (called *cooperation C* – with the other prisoner), the silent accomplice receives a 3-year sentence and the confessor goes free. If both stay silent then the police can only give both burglars 1-year sentence for minor charges. If both confess, each burglar receives a 2-year sentence."

Let *P* – Punishment for mutual deflection, *T* – Temptation to defect, *S* – Sucker's payoff and *R* – Reward for mutual cooperation, such that the matrix payoff is:

| | | $B_2$ | |
|---|---|---|---|
| | | *D* | *C* |
| $B_1$ | *D* | *(P,P)* | *(T,S)* |
| | *C* | *(S,T)* | *(R,R)* |

$S < P < R < T$ is the ranking ordering satisfied by the matrix elements.

[3] Pareto efficient is a NE refinement concept used to provide equilibrium selection in cases where the NE concept alone could provide multiple solutions to the game.

*Snowdrift*. Two drivers trapped on opposite sides of a snowdrift have the following options: (1) cooperation, by getting out and shoveling; (2) defection, by remaining in their cars. If both drivers decide to shovel, each of them gets the benefit $b$ of getting home and both share the work's cost $c$. Therefore, each receives a mutual cooperation reward $R = b - c/2$. If they both choose to defect, none of them gets home and they both obtain no benefit ($P = 0$). If only one of them shovels, then both get home but the defector's income becomes $T = b$, as it was not reduced by shoveling, while the cooperator gets $S = b - c$.

Snowdrift is mathematically equivalent to the hawk-dove and the chicken games.

*b) Strategies*

*Tit-for-Tat*. Rapoport proposed the tit-for-tat strategy for the Axelrod computer tournament in 1984. This strategy, for the Iterated Prisoner's Dilemma, starts by cooperating in the first step and subsequently repeating the opponent's previous action. In the long run, the tit-for-tat strategy cannot be exploited, since it retaliates defection (never being also the first to defect) by playing defection until the co-player decides on cooperating again. Thus, the extra income gained by the opponent during its first defection is returned to tit-for-tat. It is also a forgiving strategy as it is willing to cooperate again, defecting only if the opponent defects. This strategy effectively helps to maintain cooperative behaviors in multi-player evolutionary Prisoner's Dilemma games [40]. Tit-for-Tat modified versions were proposed to overcome the drawbacks of the strategy's determinism (e.g., in noise environments) such as the Tit-for-Two-Tats, which only defects if its opponents has defected twice in a row, and the Generous (Forgiving) Tit-for-Tat, which cooperates with some probability even if the co-player has previously defected.

*Win-Stay-Lose-Shift*. The concept of Win-Stay-Lose-Shift (WSLS) was introduced by Thorndike in 1911. WSLS strategies make use of a heuristic update rule depending on a direct payoff criterion (aspiration level) that dictates when the player should change its planned action. The player maintains its original action, if the recent rounds' average payoff is above the aspiration level, changing to a new one if not. By doing so, the aspiration level differentiates between winning and losing situations. A random choice can be performed, if there are multiple changing alternatives. An example of WSLS strategy is Pavlov [41], which was demonstrated to be able to defeat Tit-for-Tat in noisy environments (e.g., for the Iterated Prisoner's Dilemma), due to its ability to correct mistakes and exploit unconditional cooperators.

## V. Security in Delay-Tolerant Mobile Cyber-Physical Applications

Security in delay-tolerant mobile cyber-physical applications has to be handled at two levels. First, *protection mechanisms* have to be used for ensuring authentication, confidentiality, integrity, and privacy. Second, *cooperation enforcement and trust mechanisms* have to be used to guarantee that misbehaving nodes do not impair communication. The following two sections survey work in these areas.

### *A. Protection*

Protection mechanisms for ensuring authentication, confidentiality, integrity, and privacy have been widely studied for decades, and can today be better understood by resorting to textbooks in network security. Here, we are not going to focus on these mechanisms in general, but just on specific requirements of CPSs, following Wang *et al.* [10]:

- *Sensing:* the integrity of the readings from environmental parameters and settings has to be assured, as they are often critical for CPS applications. Well-known mechanisms, such as

physical shielding and access control, are extremely important to enforce this requirement. Moreover, more recent, hardware-based mechanisms, such as the Trusted Computing Group's Trusted Platform Module (TPM) [42] and ARM's Trustzone [43], can be used. The former is a chip in x86 motherboards that allow attesting the integrity of the software of a node. The latter is a functionality of recent ARM processors, which allows running software components in an isolated environment.

- *Actuation:* CPSs not only sense but also actuate on physical processes, which is even more critical, as it may have costs in terms of human health [44] or even the destruction of important devices [45]. This requirement ensures that actuation takes place under appropriate authorization. Besides that, mechanisms used to ensure the integrity of the sensing requirement may also be used.
- *Communication:* the networked nature of some CPS applications allows the establishment of a network for data fusion, the delivery of data to back-end servers (e.g., cloud-based ones), or to take coordinated response actions. This requirement imposes the use of secure (inter- and intra-)CPS communication mechanisms for protection from both active and passive adversaries. It encompasses: a confidentiality and privacy protection scheme (involving encryption) to prevent eavesdropping and stealing of user's private information; a mutual authentication protocol, to address authenticity issues; an access control and authorization scheme, to address unauthorized access issues; a key management scheme, to generate and distribute cryptographic keys; and an intrusion detection and prevention mechanism, to detect intrusions and block DoS attacks.
- *Computing:* data collected and processed in a CPS platform has to be secured against physical and cyber tampering and invalid access. This can be accomplished using encryption, digital signatures, and access control.
- *Feedback:* any CPS uses control loops that involve feedback data. This requirement is about the protection of the CPS's control system providing feedback for performing actuation. Most mechanisms already mentioned also apply for this requirement.

In DTNs, security mechanisms can be applied on a hop-by-hop (between two nodes that meet) or end-to-end (between sender and receiver) basis, depending on specific goals [46]. For instance, these mechanisms are important for nodes to verify if they are not relaying/forwarding data that was modified by another node in the path. Specific DTN requirements are:

- *Authentication:* it has to be possible to distinguish legitimate nodes (i.e., nodes that belong to the DTN) from unauthorized ones.
- *Confidentiality:* sensitive information cannot be disclosed to unauthorized third parties during its propagation through the DTN.
- *Integrity:* transmitted data cannot be modified while in transit through the DTN.
- *Privacy:* is more related to specific DTN application requirements. Sensitive information of an entity controlling/owning a DTN node cannot be disclosed to other entities.

All these requirements can be enforced using mostly cryptographic methods (availability being the exception). Confidentiality, integrity, and privacy can be mostly obtained using *symmetric cryptography* (e.g., AES) and *message authentication codes* (e.g., HMACs based on SHA-3). However, these methods require key distribution, which typically requires *public-cryptography*, also required for authentication.

Public-cryptography is usually based on a Public Key Infrastructure (PKI) and a set of Certification Authorities (CAs), as in the World Wide Web. However, this is not an optimal approach due to the disconnected nature of DTN environments, where there is no end-to-end connectivity. Decentralized mechanisms, such as Identity-Based Cryptography (IBC) [16], in which each node's identity acts as a key, are more suitable to DTNs in alternative to a PKI. Yet, IBC may not be feasible

in certain DTN environments, as it calls for a global TTP for private key generation, to guarantee that new nodes can enter the network. Some alternatives have been proposed for such environments such as the use of trusted social contacts [47], or obfuscating routing information [48].

### *B. Cooperation Enforcement and Trust*

In a DTN, nodes can be controlled by rational users or entities. Due to resources' scarcity, these entities may attempt to maximize their utilities and preserve their resources, i.e., behave selfishly by only forwarding messages for nodes with whom they have social ties with, which can significantly impact network performance [6] [7]. In other words, selfish entities (or nodes) may be faithful to the nodes from the same group (based on a common interest), and uncooperative to outsiders. However, despite the fact of selfish behavior being harmful, selfish nodes can also be used to control message overhead in resource-limited networks.

*Selfishness* [49] measures the level of interaction or cooperation among nodes. However, the lack of connectivity, or large transmission delays, that are typical in DTNs, makes the task of designing cooperation enforcement schemes for DTNs more challenging. Consequently, a selfish behavior is difficult to identify and measure which is aggravated by the delayed feedback information. Still, recently some DTN routing solutions have been proposed to address this problem [50] [51] [52] [53] [54] [55].

As stated before, cooperation enforcement schemes for DTNs are categorized as: reputation-based, remuneration-based and game-based. Next, cooperation enforcement schemes to handle selfishness in DTNs are reviewed.

#### *1) Reputation-based mechanisms*

*RCAR*. The authors of [51] proposed a reputation-based extension to the Context Aware Routing (CAR) [56] protocol, called RCAR, to address the problem of black-holes in DTNs.

In RCAR, every node keeps a local notion of reputation, therefore avoiding the overhead and technical complication associated with a centralized reputation management system in decentralized networks. Upon message forwarding, each node estimates the likelihood of selecting forwarding nodes based on the node's reputation, that is, based on past interactions with possible forwarding nodes.

The reputation management system employs both data and acknowledgment messages. The data messages' format incorporates the nodes' list (*nlist*), i.e., the list of nodes a message has passed through, as well as a list of digital signatures (*slist*) that is used to prove the integrity and authenticity of every node in *nlist*. The update mechanism, employed by the proposed reputation management system, does not disseminate updates based on broadcast/multicast mechanisms, which are expensive.

Reputation is maintained by means of three mechanisms: acknowledgments, nodes' list and aging. Since each message contains a list of forwarding nodes the message has traversed, upon message reception, nodes should update the reputation of the forwarding nodes in *nlist*. Despite that, the sender waits for an acknowledgment from the destination node, and only increases the reputation of the forwarding node upon reception of the acknowledgment. At last, the aging mechanism is used to decrease the reputation of all nodes. As messages can get lost, there is no way a node can know the reasons behind it, e.g., message drop due to a black-hole node (which node misbehaved?), or due to buffer overflow, or even due to Time-To-Live (TTL) expiration. To fulfill DTN requirements, the aging mechanisms decrease period is dynamically updated using Kalman filters [57].

*Give2Get*. In [52], the authors proposed two strategy proof forwarding protocols for Pocket Switched Networks (PSNs) [58] of selfish individuals, namely, Give2Get Epidemic and Give2Get Delegation. The proposed protocols are strategy proof, which means that the strategies of following the protocol are Nash Equilibria. So, no individual has any incentive to deviate.

In Epidemic Forwarding [59], nodes use every contact opportunity to forward messages. When a node is in contact with another one, and it has a message that the other does not have, the message is relayed to the other node. If selfish nodes that simply drop messages are considered in the network (also called message droppers), epidemic forwarding performance degrades [7].

In Delegation Forwarding [60], nodes have associated to them a forwarding quality, which may depend on the message's destination. A message, upon its generation, is associated with the forwarding quality of the sender. Then, when a relay node gets in contact with another node, it checks whether the forwarding quality of the other relay node is higher than that of the forwarding quality of the message. If so, it creates a replica of the message, labels both messages with the forwarding quality of the other relay node, keeps one of them and forwards the other message to the other relay node. If not, the message is not forwarded.

The idea behind G2G Epidemic Forwarding is that the protocol works correctly even if all nodes in the network are selfish. That can be accomplished if no selfish node has a better choice than following the protocol truthfully, thus being a NE. G2G Epidemic Forwarding consists of three phases: message generation, relay and test. In the message generation phase, the message is modified, that is, the message sender is hidden from every possible relay except the destination, so that the relay candidate has no interest in not accepting it[4]. In the relay phase, nodes collect proof of relay (POR) to show to the source and/or previous relays during the test phase. In the test phase, nodes present evidence of their correct behavior, making it impossible for relays to drop messages. If no evidence is presented by the relay, a proof of misbehavior (PoM) is generated and the relay is removed from the network.

G2G Delegation Forwarding makes use of G2G Epidemic Forwarding techniques with the intention of stopping message droppers. Since G2G Epidemic Forwarding techniques are not enough for the algorithms to be NE, due to the fact of selfish nodes (1) being able to lie about the forwarding quality (i.e., being liars), and (2) being able to change the forwarding quality of messages, e.g., set it to zero, therefore getting rid of the message sooner (i.e., being cheaters). Two approaches have been devised for G2G Delegation Forwarding: Delegation Destination Frequency (DDF) and Delegation Destination Last Contact (DDLC). In the former, a node forwards a message to another node if the other node has contacted the destination of the message more frequently than any other node in the nodes' list inside the message. In the latter, a node forwards a message to another node if the other node has contacted the destination of the message more recently than any other node in the nodes' list inside the message.

*2) Remuneration-based mechanisms*

*SMART*. The authors of [50] proposed a secure multilayer credit-based incentive (SMART) scheme to stimulate bundle forwarding cooperation among selfish nodes, which can be implemented in a distributed manner without relying on any tamperproof hardware in a DTN environment. In SMART, intermediate nodes, without the involvement of the sender, can transfer/distribute credit since a forwarding path cannot be predicted by the sender (unlike in MANETs where forwarding paths are known a priori), and due to nodes' mobility, intermediate nodes and the sender may be disconnected.

[4] The authors made the following assumptions: (1) every node is selfish and accepts messages destined to it; (2) there are no Byzantine nodes in the network, i.e., nodes that behave arbitrarily; (3) selfish nodes do not collude; (4) nodes are capable of making use of public key cryptography.

SMART assumes the existence of two main entities: an Offline Security Manager (OSM), which is responsible for key distribution, and a virtual bank (VB), i.e., a special network component, such as a roadside unit in vehicular networks [61] or the information publisher in social networks [62], which takes charge of credit clearance. DTN nodes submit collected coins to the VB, by exploiting opportunistic links to these special network components. Upon joining the network, every node should register with the OSM. During the clearance phase, nodes should submit the collected layered coins to the VB in order to receive their rewards.

SMART is based on the concept of a layered coin – composed of multiple layers, where each layer is generated by the source/destination or an intermediate node – providing virtual electronic credits to charge for and reward the provision of data forwarding in DTN environments. The first layer, also known as the base layer, is created by the source node, and it indicates: the payment rate (credit value), the remuneration conditions, and the class of service (CoS) requirements, besides other reward policies. In the following propagation process, a new layer, also known as the endorsed layer, which is built on the previous one, is created by each intermediate node by appending a non-forgeable digital signature. It implies that the forwarding node agrees in providing forwarding services underneath the predefined CoS requirement, thus being accordingly remunerated in accordance to the reward policy. By checking the signature at each endorsed layer, it is easy to check the propagation path and determine each intermediate node. If the provided forwarding service fulfills remuneration conditions defined in the predefined reward policy, in the rewarding and charging phase, each forwarding node along a single/multiple path(s) will share the credit defined in this coin depending on single/multi-copy data-forwarding algorithms and its forwarding results.

Four aspects were considered in SMART's design, namely: effectiveness, by stimulating cooperation among selfish nodes; security, by being robust to various attacks; efficiency, by not introducing extra communication and transmission overhead; and generality, by being compatible with most existing DTN routing schemes. A trade-off had to be taken into account between security and performance. Security, as intermediate nodes manage all security issues related to a coin, during the forwarding process, a(n) (individual or social) selfish node(s) may attempt to maximize its expected benefit by cheating the system. The final aspect is performance, due to the extra computation and transmission overhead as a result of any security functionality, during the design of a secure credit-based incentive scheme.

*Mobicent*. In [53], the authors proposed a credit-based system to support Internet access service for heterogeneous wireless network environments. Two modes of operation are supported by mobile devices in such environments, namely: (1) a long-range low-bandwidth link, e.g., cellular interface, to maintain an always-on connection, used in particular by the source and destination nodes; (2) a short-range high-bandwidth link, e.g., Wi-Fi, to opportunistically exchange large amounts of data with neighboring nodes (which is used by all nodes), since due to node mobility these links tend to be intermittent.

The Mobicent network architecture consists of three components: (*i*) a TTP, that stores key information for all nodes, providing also verification and remuneration services; (*ii*) Helpers, which are mobile or static nodes which help in data relaying using the short-range high-bandwidth link; (*iii*) Mobile clients, that are the destination nodes.

The payment mechanism works as follows: the data payload is encrypted by each relay with a one-time symmetric key before being forwarded. When a client receives the encrypted data, and intends to access the decrypted data, it must make a payment to the TTP in exchange for the encrypted keys. This happens since the key is sent along with the data in encrypted form, and the TTP is the only means to recover them. Only relays involved in data forwarding receive payment.

In [53], the authors dealt with two forms of selfish actions (or attacks), namely: edge insertion attacks and edge hiding attacks. Let $G = (V, E)$ be a contact graph. Each vertex (or node) $v \in V$ can be identified by an integer value $i = 1,2, \ldots, |V|$. Each edge $e \in E$, identified by a pair $\{v_1, v_2\}$, denotes the opportunistic contact between two nodes at time $t(e)$. Thus, $(\{v_1, v_2), t_1)$ means that $v_1$ meets $v_2$ at time $t_1$. The former, i.e., an *edge insertion attack* of a node $v$ consists in creating a Sybil $v'$ so that $G \rightarrow G' = (V', E')$, where $V' = V \cup \{v'\}$ and $E' = E^{v \rightarrow (v,v')} \cup \{(v, v', t)\}$. The latter, i.e., an *edge hiding attack* for a node $v$ consists in modifying $G \rightarrow G' = (V, E - e)$, where $e \in E(v)$.

According to the authors, due to network's dynamics, edge insertion and hiding attacks are extremely difficult to be detected in DTNs. As the proposed scheme provides incentives for selfish nodes to honestly behave (that is, by setting the client's payments and relay's rewards so that nodes behave truthfully), mechanisms to detect selfish actions are not required. And also, by working on top of the DTN routing layer, this scheme ensures that selfish actions do not result in a large reward, not requiring pre-determined routing paths either.

*3) Game-based mechanisms*

*Tit-for-Tat*. In [54], the authors propose a pair-wise tit-for-tat (TFT) – a simple, robust and practical cooperation enforcement scheme for DTNs – which incorporates generosity and contrition to tackle bootstrapping or exploitation problems common in basic TFT mechanisms [29] [63] [64].

Upon the first encounter between two nodes, as they have not previously relayed packets successfully among them, the basic TFT mechanisms would prevent relaying. Generosity enables bootstrapping by allowing an initial cooperation between the nodes up to $\varepsilon$, that is, a node is allowed to send $\varepsilon$ packets more than it should according to what it had previously relayed. It also handles asymmetric traffic demands by absorbing traffic imbalance up to a $\varepsilon$ amount. But any imbalance exceeding $\varepsilon$ could lead to lengthy retaliation among neighbors. As a result, generosity is insufficient by itself. Contrition addresses the previous situation by refraining from reacting to a valid retaliation to its own mistake, i.e., preventing mistakes from causing endless retaliation. With contrition, a node realizes that the other node's action in the current interval was due to its own action in the previous interval, and so does not lower service in the future interval. Similarly, contrition cannot work by itself, since it only provides a way to return to stability after perturbation, not providing a way to reach stability.

The authors also proposed an incentive-aware routing protocol in which selfish nodes are allowed to maximize their individual utilities taking into account the TFT constraints. For a given pair of nodes, the TFT constraints state that the total amount of traffic through a link is equal to the total amount of traffic in the opposite direction.

The proposed routing protocol consists of the following components: (*i*) link state (i.e., link capacity, mean and variance of the waiting time on links) that are periodically exchanged by every node, similarly to many link state protocols (e.g., OSPF)[5]; (*ii*) with link state, each source node computes forwarding paths, and uses source routing to send traffic; (*iii*) each destination node sends an ACK via flooding, upon receiving a packet. Then, the source node uses it to update its TFT constraints for the subsequent interval.

*Barter*. The authors of [55] proposed a mechanism, which is modeled using game-theory, to discourage selfish behavior based on the principles of barter[6]. In the context of the proposed mechanism, exchanges are made in messages. Hence, when two mobile nodes are in

[5] It is assumed in [54] that link state is disseminated faithfully. Thus, the authors focused on making the data-plane incentive compatible. Security of the control plane was left for future work.

[6] According to [20], barter means "exchange (goods or services) for other goods or services without using money".

communication range with each other, (1) they send messages' descriptions that they currently store to each other, (2) and then they reach a consensus on the subset of messages they want to exchange. Fairness is ensured (*a*) by guaranteeing that the selected subsets have the same size, and (*b*) by using a preference order, in which messages are exchanged in a message-by-message manner. Notice that the number of exchanged messages depends on the length of the shorter list or the duration of the connection. The exchange can be interrupted, if any party cheats. So, any major disadvantage is not experienced by the honest party, since it at most downloaded one less message in comparison to the misbehaving party.

In this scheme, the mobile nodes decide which messages they want to download from each other. If nodes behave selfishly, that is, they only download messages that are of primary interest (or destined) to them, in the long run, and according to the principle of barter, they will not have other messages to exchange for the ones they are interested in. Therefore, messages that are secondary (or not destined) for a given mobile node may still have a barter value for the mobile node. Thus, it can be perceived as an investment to acquire new primary messages.

Two assumptions were made by the authors: (*i*) mobile nodes offer all their valid and only valid messages to download, and (*ii*) two mechanisms are present in the system to prevent the injection of fake messages, specifically: digital signatures, where only nodes with a digital signature, supplied by an authority, can exchange messages among themselves, and a reputation mechanism, that is based on the quality of the message contents.

Table 2 presents a summary and comparison of cooperation enforcement schemes applied to DTN Routing protocols based on the security mechanism used, the main idea of the routing protocol, the attacks considered, and the overhead.

## VI. Delay-Tolerant Mobile Cyber-Physical Applications

In this section, some example delay-tolerant mobile cyber-physical applications prone to security issues are presented.

### *A. Health monitoring*

With relatively affordable onboard sensors or connected external sensors, mobile cyber-physical applications can be leveraged to *monitor patient's health*. By means of wireless connectivity (e.g., Cellular, Wi-Fi, Bluetooth), standard IP networking can be used to send data back to Internet services that aggregate information for doctors. Doctors can use this service to more precisely adjust medication dosages, based on trends in symptoms over the course of a day. Onboard smartphone sensors' data collection is relatively easy for a mobile cyber-physical application, but processing and disseminating data is much more challenging for applications that use multiple external sensors networked through USB, Bluetooth, or other means. Therefore, appropriate architectures to buffer data when cellular connections are unavailable are necessary (thus involving delay-tolerance), hence not overrunning the device's memory. Indeed, through onboard phone processing, the amount of data to be transmitted from the phone to the Internet service or buffered can be reduced.

An example of a health monitoring system is the mobile electrocardiogram (ECG) system [65] that uses smartphones as base stations for ECG measurement and analysis. The main idea behind mobile ECG was, on the one hand, to reduce the workload of the medical personnel, and, on the other hand, to accelerate the measurement to analysis cycle. The smartphone that received the transmitted data from the measurement device stored it in a memory card. So, under normal circumstances the analysis is performed automatically near the patient, and only in critical cases

Table 2
A summary and comparison of incentive mechanisms applied to DTNs

| *Publication* | *Security mechanism* | *Main idea* | *Attack(s) considered* | *Overhead* |
|---|---|---|---|---|
| RCAR [51] | Reputation-based | Every node keeps locally the reputation of every forwarding node it comes in contact with. Nodes with the highest reputation are selected as message forwarders. | Black-hole attack | The overhead is reduced, since RCAR reputation mechanism is integrated in the routing protocol (data and ACK messages). Thus, broadcast/multicast is not used by RCAR. |
| Give2Get [52] | Reputation-based | Every node keeps locally evidences of their correct behavior as a relay. If no evidences are presented in future encounters, nodes are removed from the network. | Individual selfishness (message droppers, liars and cheaters) Social selfishness (collusion*) | Storage and communication overhead due to the use of PORs. The probabilities of detection of selfish behavior are of more than 90% and 60% for G2G Epidemic and Delegation forwarding, respectively. |
| SMART [50] | Remuneration-based | Secure multilayer credit-based incentive scheme to stimulate bundle forwarding cooperation among selfish nodes in a DTN environment. | Credit forgery attack Nodular tontine attack Submission refusal attack | Additional cryptographic (computational and communication) overhead due to the use of layered coins. The performance of the underlying routing protocol, with or without SMART, are very close. |
| Mobicent [53] | Remuneration-based | Runs on top of DTN routing layer and provides incentives for selfish nodes to behave honestly, thus not requiring a selfish actions' detection mechanisms. | Edge insertion attack Edge hiding attack | No overhead analysis mentioned in the paper. |
| Tit-For-Tat [54] | Game-based | Simple, robust and practical incentive mechanism for DTNs which incorporates generosity and contrition. | Individual selfishness | Additional control overhead caused by the dissemination of link state and ACK messages. |
| Barter [55] | Game-based | Incentive mechanism to discourage selfish behavior based on the principles of barter. | Individual selfishness | No overhead analysis mentioned in the paper. |

* Despite not being the target attack addressed in [52], the authors proposed two mechanisms, namely random checks of conformity and rewarding traitors, to mitigate the presence of colluding nodes or to limit their possible harm.

(i.e., if some abnormities are detected) the data is sent to medical personnel for further analysis. In previous mobile measurement systems like the one proposed in [66], smartphones, that were used as gateways, were used to send measurement data continuously to a separate server, where data analysis happened, incurring in high financial costs as continuous data communication on mobile networks is charged by the amount of data sent [67]. Another feature of this system was the possibility of the patient sending an alarm in the case of occurrence of one of the following symptoms: anxiety, faintness or other distress.

As previously stated, mobile measurement systems that rely on a mobile network for data communication to a server- or cloud-based infrastructure might incur in a financial penalty because of the amount of collected data by wearable and wireless medical sensors, which monitors for example heart rate, oxygen level, blood flow, respiratory rate, muscle activities, movement pattern, body inclination, and oxygen uptake. Since the measurement data is stored locally on the smartphone, a possible approach could be the use of data mules to carry the data to the hospital provided that encryption schemes were in place to safeguard the data.

In [68], a study of the potential of DTNs to support health care services, such as email access, notification of lab results, backup of Electronic Health Records (EHR) and tele-consultation in a low resource setting, is presented. As each service has its own requirements, the frequency of data delivery is context dependent. For example, for services like notification of lab results and/or ordering of medical supplies, the physical transportation of digital data at a frequency of less than once per week is tolerable.

### *B. CarTel*

The CarTel project [69] designed a multipurpose distributed sensor computing system to collect, process, deliver, and visualize data from remote and intermittently connected sensors located on mobile units (e.g., cars), which in comparison to static sensor networks can sense at much better fidelity and higher scale the environment, in particular over large areas. It is a multipurpose project as its application areas are diverse such as traffic monitoring, route planning, environmental monitoring, civil infrastructure monitoring, automotive diagnostics, augmented reality and data muling.

A CarTel node consists of a mobile embedded computer with a set of sensors, which collects and processes sensor readings locally before sending them to a central portal, for further processing.

Cartel is composed of three main components:

- The portal, which is a central location that hosts CarTel applications and functions, being the point of control and configuration for the distributed system. All data collected by the mobile nodes is sent to the portal.
- The intermittently connected database (ICEDB), which is a delay-tolerant continuous query processor. ICEDB distributes query execution and results between the ICEDB server running on the portal and the remote nodes. It supports heterogeneous data types and its queries are written in SQL with several extensions for continuous queries and prioritization.
- The carry-and-forward network (CafNet), which is a general propose network stack for delay-tolerant communication, which can be used by applications to send messages across an intermittently connected network. Two kinds of intermittency are envisioned:
  - The first one, through opportunistic networking (e.g., Wi-Fi, Bluetooth), since end-to-end connectivity is available but intermittent. Mobile nodes can access Wi-Fi access points to communicate with the portal.
  - The second one, through a best effort approach, by using other mobile nodes storage devices, such as USB keys and flash memory as data mules, relying on them to deliver data to the portal.

ICEDB and CafNet specify how nodes collect, process and deliver sensor data. Generally, CarTel applications use three main components of the portal environment, namely: (1) the portal framework, (2) the ICEDB server used to retrieve sensor data, and (3) a data visualization library used to display geo-coded attributes. The portal framework provides a platform for building applications that share a mutual user authentication mechanism and look-and-feel. And, in order to lessen privacy concerns, users are only allowed to view collected sensor data from remote nodes hosted by them.

For example, in the traffic monitoring CarTel application, each car is instrumented with a GPS sensor to opportunistically gather information, such as traffic delays observed as cars move, allowing to infer congestion hop spots. Furthermore, equipping cars with cameras allows building applications that can help users to navigate unfamiliar territories.

## VII. Conclusion

A discussion of the main security concerns in delay-tolerant cyber-physical applications was presented. Precisely, issues related to authentication, confidentiality, integrity, availability, privacy, trust, and cooperation enforcement were analyzed. A survey of cooperation enforcement schemes available for DTNs was also presented.

The design of delay-tolerant mobile cyber-physical applications poses significant security challenges. The increased interaction between the physical and cyber systems exposes physical systems to cyber systems' security threats. Mobility, self-organization and delay tolerance require

finding a suitable node to forward messages, which may incur in the consumption of nodes' limited resources, along with additional concerns such as trust establishment and fairness of contributions among the nodes.

Using cooperation enforcement (reputation-, remuneration-, or game-based) and encryption schemes, security in delay-tolerant mobile cyber-physical applications can be leveraged, providing confidentiality, integrity and availability. Indeed, such schemes may incur in additional computational costs and/or communication overheads. For example, additional data may be transferred between nodes to establish trust, or the sensitive application's data traversing the network needs to be encrypted, in a hop-by-hop or end-to-end manner, thus incurring in additional computation costs. Additionally, a trusted third party may be used to help providing such services, which also increases the complexity of the architecture and protocols.

## REFERENCES


[1] Janos Sztipanovits, "Composition of Cyber-Physical Systems," in *Engineering of Computer-Based Systems*, 2007, pp. 3-6.

[2] Edward A Lee, "Cyber physical systems: Design challenges," in *IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC)*, 2008, pp. 363-369.

[3] Jules White et al., "R&D challenges and solutions for mobile cyber-physical applications and supporting Internet services," *Internet Services and Applications*, vol. 1, no. 1, pp. 45-56, 2010.

[4] Maurice J Khabbaz, Chadi M Assi, and Wissam F Fawaz, "Disruption-tolerant networking: A comprehensive survey on recent developments and persisting challenges," *IEEE Communications Surveys & Tutorials*, vol. 14, no. 2, pp. 607-640, 2012.

[5] Kaimin Wei, Xiao Liang, and Ke Xu, "A survey of social-aware routing protocols in delay tolerant networks: applications, taxonomy and design-related issues," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 556-578, 2014.

[6] Pan Hui et al., "Selfishness, altruism and message spreading in mobile social networks," in *IEEE INFOCOM*, 2009, pp. 1-6.

[7] Naércio Magaia, Paulo Rogério Pereira, and Miguel P Correia, "Selfish and malicious behavior in Delay-Tolerant Networks," in *Future Network and Mobile Summit*, 2013, pp. 1-10.

[8] Zhaoyu Liu, Anthony W Joy, and Robert A Thompson, "A dynamic trust model for mobile ad hoc networks," in *Future Trends of Distributed Computing Systems*, 2004, pp. 80-85.

[9] Ying Zhu, Bin Xu, Xinghua Shi, and Yu Wang, "A survey of social-based routing in delay tolerant networks: positive and negative social effects," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 1, pp. 387-401, 2013.

[10] Eric Ke Wang et al., "Security issues and challenges for cyber physical system," in *IEEE/ACM Int'l Conference on Green Computing and Communications & Int'l Conference on Cyber, Physical and Social Computing*, 2010, pp. 733-738.

[11] Elinor Mills, "Hackers broke into FAA air traffic Networks," *The Wall Street Journal*, p. A6, May 2009. [Online]. http://online.wsj.com/articles/SB124165272826193727

[12] N Leavitt, "Researchers fight to keep implanted medical devices safe from hackers," *Computer*, vol. 43, no. 8, pp. 11-14, 2010.

[13] Kelly O'Connell, "CIA Report: Cyber Extortionists Attacked Foreign Power Grid, Disrupting Delivery," *Internet Business Law Services*, 2008. [Online]. http://www.ibls.com/internet_law_news_portal_view.aspx?id=1963&s=latestnews

[14] Karl Koscher et al., "Experimental Security Analysis of a Modern Automobile," in *Security and Privacy*, 2010, pp. 447-462.

[15] William Stallings, *Network security essentials: applications and standards*.: Pearson Education India, 2007.

[16] Abdullatif Shikfa, Melek Onen, and Refik Molva, "Privacy and confidentiality in context-based and epidemic forwarding," *Computer Communications*, vol. 33, no. 13, pp. 1493-1504, 2010.

[17] Zhengyi Le, Gauri Vakde, and Matthew Wright, "PEON: privacy-enhanced opportunistic networks with applications in assistive environments," in *PErvasive Technologies Related to Assistive Environments*, 2009, p. 76.

[18] Naércio Magaia, Paulo Rogério Pereira, and Miguel Pupo Correia, "Nodes' misbehavior in Vehicular Delay-Tolerant Networks," in *Conference on Future Internet Communications (CFIC)*, Lisbon, 2013, pp. 1-9.

[19] Ari Keränen, Jörg Ott, and Teemu Kärkkäinen, "The ONE simulator for DTN protocol evaluation," in *Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, Belgium, 2009, p. 55.

[20] Angus Stevenson, *Oxford dictionary of English*.: Oxford University Press, 2010.

[21] Karen Cook, *Trust in society*.: Russell Sage Foundation, 2001.

[22] Jin-Hee Cho, Ananthram Swami, and Ray Chen, "A survey on trust management for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 4, pp. 562-583, 2011.

[23] Matt Blaze, Joan Feigenbaum, and Jack Lacy, "Decentralized trust management," in *Security and Privacy*, 1996, pp. 164-173.

[24] Lorenzo Strigini et al., "Resilience-building technologies: State of knowledge," 2007.

[25] Levente Buttyán and Jean-Pierre Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *Mobile Networks and Applications*, vol. 8, no. 5, pp. 579-592, 2003.

[26] Markus Jakobsson, Jean-Pierre Hubaux, and Levente Buttyán, "A micro-payment scheme encouraging collaboration in multi-hop cellular networks," in *Financial Cryptography*, 2003, pp. 15-33.

[27] Ljubica Blazevic et al., "Self organization in mobile ad hoc networks: the approach of Terminodes," *IEEE Communications Magazine*, vol. 39, no. 6, pp. 166-174, 2001.

[28] Levente Buttyán and Jean-Pierre Hubaux, "Enforcing service availability in mobile ad-hoc WANs," in *Mobile Ad Hoc Networking & Computing*, 2000, pp. 87-96.

[29] Vikram Srinivasan, Pavan Nuggehalli, Carla-Fabiana Chiasserini, and Ramesh R Rao, "Cooperation in wireless ad hoc networks," in *IEEE INFOCOM*, vol. 2, 2003, pp. 808-817.

[30] Luzi Anderegg and Stephan Eidenbenz, "Ad hoc-VCG: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents," in *Mobile Computing and Networking*, 2003, pp. 245-259.

[31] Weizhao Wang, Stephan Eidenbenz, Yu Wang, and Xiang-Yang Li, "OURS: optimal unicast routing systems in non-cooperative wireless networks," in *Mobile Computing and Networking*, 2006, pp. 402-413.

[32] WeiZhao Wang, Xiang-Yang Li, and Yu Wang, "Truthful multicast routing in selfish wireless networks," in *Mobile Computing and Networking*, 2004, pp. 245-259.

[33] Sheng Zhong, Li Erran Li, Yanbin Grace Liu, and Yang Richard Yang, "On designing incentive-compatible routing and forwarding protocols in wireless ad-hoc networks," *Wireless networks*, vol. 13, no. 6, pp. 799-816, 2007.

[34] Pietro Michiardi and Refik Molva, "Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Advanced Communications and Multimedia Security*.: Springer, 2002, pp. 107-121.

[35] Sergio Marti, Thomas J Giuli, Kevin Lai, and Mary Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *Mobile Computing and Networking*, 2000, pp. 255-265.

[36] Jaydip Sen, "A survey on reputation and trust-based systems for wireless communication networks," *arXiv preprint arXiv:1012.2529*, 2010.

[37] Martin J Osborne and Ariel Rubinstein, *A course in game theory*.: MIT press, 1994.

[38] Theodore L Turocy, "Texas a&m university," *Bernhard von Stengel, London School of Economics "Game Theory" CDAM Research Report (October 2001)*, 2001.

[39] Shih-Fen Cheng, Daniel M Reeves, Yevgeniy Vorobeychik, and Michael P Wellman, "Notes on equilibria in symmetric games," in *Game Theory and Decision Theory*, 2004.

[40] Gyorgy Szabó and Gabor Fath, "Evolutionary games on graphs," *Physics Reports*, vol. 446, no. 4, pp. 97-216, 2007.

[41] David Kraines and Vivian Kraines, "Pavlov and the prisoner's dilemma," *Theory and decision*, vol. 26, no. 1, pp. 47-79, 1989.

[42] Trusted Computing Group, Trusted Platform Module Library Specification, Family "2.0", Level 00, Revision 01.07, March 2004.

[43] Tiago Alves and Don Felton, "TrustZone: Integrated hardware and software security," *ARM white paper*, vol. 3, no. 4.

[44] Jordan Robertson, "The Trials of a Diabetic Hacker," *BusinessWeek*, February 2012. [Online]. http://www.businessweek.com/articles/2012-02-23/the-trials-of-a-diabetic-hacker

[45] Jeanne Meserve, "Sources: Staged cyber attack reveals vulnerability in power grid," *CNN*, September 2007. [Online]. http://edition.cnn.com/2007/US/09/26/power.at.risk/

[46] Stephen Farrell, Howard Weiss, Susan Symington, and Peter Lovell, "Bundle Security Protocol Specification," *RFC*, no. 6257, 2011. [Online]. https://tools.ietf.org/html/rfc6257

[47] Karim El Defrawy, John Solis, and Gene Tsudik, "Leveraging Social Contacts for Message Confidentiality in Delay Tolerant Networks," in *IEEE International Computer Software and Applications Conference*, vol. 1, 2009, pp. 271-279.

[48] Iain Parris and Tristan Henderson, "Privacy-enhanced social-network routing," *Computer Communications*, vol. 35, no. 1, pp. 62-74, 2012.

[49] Qinghua Li, Sencun Zhu, and Guohong Cao, "Routing in Socially Selfish Delay Tolerant Networks," in *IEEE INFOCOM*, 2010, pp. 1-9.

[50] Haojin Zhu, Xiaodong Lin, Rongxing Lu, Yanfei Fan, and Xuemin Shen, "Smart: A secure multilayer credit-based incentive scheme for delay-tolerant networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 8, pp. 4628-4639, 2009.

[51] Gianluca Dini and Angelica Lo Duca, "Towards a reputation-based routing protocol to contrast blackholes in a delay tolerant network," *Ad Hoc Networks*, vol. 10, no. 7, pp. 1167-1178, sep 2012.

[52] Alessandro Mei and Julinda Stefa, "Give2Get: Forwarding in Social Mobile Wireless Networks of Selfish Individuals," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 4, pp. 569-582, jul 2012.

[53] Bin Bin Chen and Mun Choon Chan, "MobiCent: a Credit-Based Incentive System for Disruption Tolerant Network," in *IEEE INFOCOM*, 2010, pp. 1-9.

[54] Upendra Shevade and Yin Zhang, "Incentive-aware routing in DTNs," in *IEEE International Conference on Network Protocols*, 2008, pp. 238-247.

[55] Levente Buttyán, László Dóra, Márk Félegyházi, and István Vajda, "Barter trade improves message delivery in opportunistic networks," *Ad Hoc Networks*, vol. 8, no. 1, pp. 1-14, jan 2010.

[56] Mirco Musolesi and Cecilia Mascolo, "CAR: Context-Aware Adaptive Routing for Delay-Tolerant Mobile Networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 2, pp. 246-260, feb 2009.

[57] Rudolph Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Basic Engineering*, vol. 82, no. 1, p. 35, mar 1960.

[58] Pan Hui et al., "Pocket switched networks and human mobility in conference environments," in *ACM SIGCOMM workshop on Delay-tolerant networking - WDTN*, New York, New York, USA, 2005, pp. 244-251.

[59] Amin Vahdat and David Becker, "Epidemic routing for partially connected ad hoc networks," Duke University, Tech. rep. 2000.

[60] Vijay Erramilli, Augustin Chaintreau, Mark Crovella, and Christophe Diot, "Delegation forwarding," in *Mobile Ad Hoc Networking and Computing - MobiHoc*, New York, New York, USA, 2008, p. 251.

[61] Suk-Bok Lee, Gabriel Pan, Joon-Sang Park, Mario Gerla, and Songwu Lu, "Secure incentives for commercial ad dissemination in vehicular networks," in *Mobile Ad Hoc Networking and Computing - MobiHoc*, New York, New York, USA, 2007, p. 150.

[62] Anargyros Garyfalos and Kevin C. Almeroth., "Coupons: A Multilevel Incentive Scheme for Information Dissemination in Mobile Networks," *IEEE Transactions on Mobile Computing*, vol. 7, no. 6, pp. 792-804, jun 2008.

[63] Juan Jose Jaramillo and R. Srikant, "DARWIN," in *Mobile Computing and Networking - MobiCom*, New York, New York, USA, 2007, p. 87.

[64] Fabio Milan, Juan José Jaramillo, and R. Srikant, "Achieving cooperation in multihop wireless networks of selfish nodes," in *Game Theory for Communications and Networks - GameNets*, New York, New York, USA, 2006, p. 3.

[65] Harri Kailanto, Esko Hyvarinen, and Jari Hyttinen, "Mobile ECG measurement and analysis system using mobile phone as the base station," in *Pervasive Computing Technologies for Healthcare*, jan 2008, pp. 12-14.

[66] Wenxi Chen et al., "A mobile phone-based wearable vital signs monitoring system," in *Computer and Information Technology*, 2005, pp. 950-955.

[67] Jimena Rodriguez, Lacramioara Dranca, Alfredo Goñi, and Arantza Illarramendi, "Web access to data in a mobile ECG monitoring system," in *Transformation of Healthcare with Information Technologies*.: IOS Press, 2004, ch. Web access to data in a mobile ECG monitoring system, pp. 100-111.

[68] Shabbir Syed-Abdul et al., "Study on the potential for delay tolerant networks by health workers in low resource settings," *Computer Methods and Programs in Biomedicine*, vol. 107, no. 3, pp. 557-64, sep 2012.

[69] Bret Hull et al., "CarTel: a distributed mobile sensor computing system," in *Embedded Networked Sensor Systems*, 2006, pp. 125-138.

[70] Vikram Srinivasan, Pavan Nuggehalli, Carla-Fabiana Chiasserini, and Ramesh R Rao, "Cooperation in wireless ad hoc networks," in *IEEE INFOCOM*, vol. 2, 2003, pp. 808-817.