WIRELESS COMMUNICATIONS AND MOBILE COMPUTING
Wirel. Commun. Mob. Comput. 2006; **6**:319–332
Published online in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/wcm.398

Cooperation enforcement schemes for MANETs: A survey

G. F. Marias[1]*,†, P. Georgiadis[1], D. Flitzanis[2] and K. Mandalas[2]

[1]*Department of Informatics and Telecommunications, University of Athens, 15784, Greece*
[2]*Department of Informatics, Athens University of Economics and Business, 10434, Greece*

Summary

The employment of adequate trust methods in mobile ad hoc networks (MANET) has been receiving increasing attention during the last few years, and several trust and security establishment solutions that rely on cryptographic and hashing schemes have been proposed. These schemes, although effective, produce significant processing and communication overheads and consume energy, and, hence, they do not take into account the idiosyncrasies of a MANET. More recently, cooperation enforcement methods have been proposed for trust establishment in MANET. These schemes, classified as reputation-based and credit-based, are considered suitable for ad hoc networks, where key or certificate distribution centers are absent or ephemerally present, and for networks that consist of devices with limited processing, battery, and memory resources. Cooperation enforcement methods do not provide strong authentication of entities. Instead, they contribute to the identification of the trustworthiness of peers and to the enforcement cooperation using mutual incentives. This paper surveys the most important cooperation enforcement methods that have been introduced, providing a comprehensive comparison between the different proposed schemes. Copyright © 2006 John Wiley & Sons, Ltd.

KEY WORDS: mobile ad hoc networks; trust establishment; selfishness; cooperation-enforcement

1. Introduction

A MANET is a self-organized wireless network, consisting of nodes responsible for its creation, operation and maintenance. Due to mobility, the number of nodes and the topology of the network vary with time. The nodes of a MANET follow their motivation to participate as a co-operation rule, if they behave rationally. A newcomer's incentive is to offer functions (e.g., routing and packet forwarding) to the other nodes, which, in their turn, return this by offering connectivity services. Such reciprocity principles can be used to establish trust among the nodes, which is essential for the steady-state operation of a MANET. Adjacent nodes (in the coverage area of each other) may build up trust with time, and provide this knowledge to the other nodes as a reputation. On the other hand the value of this trust diminishes when these nodes, due to their mobility, become distant. Thus, the trust established between two nodes might be lost with time, influencing network's performance. Moreover, it is a utopia to assume that all the nodes behave rationally, since passionate behaviors might occur. *Selfish*, *malicious*, and *hacker* nodes may easily follow the reciprocity principles in order to be connected on a MANET, but their intentions might be tainted. A *selfish* node disinclines to spend its resources (e.g., battery) for serving

*Correspondence to: G. F. Marias, Department of Informatics and Telecommunications, University of Athens, 15784, Greece.
†E-mail: marias@mm.di.uoa.gr

network's operations and maximizing the social welfare (e.g., forward packets not destined for it). Instead, it cooperates when the network tasks maximize its own profit. A *malicious node* attacks to damage network's operation, through denial of service (DoS) attacks, such as sinkhole, flooding, or sleep deprivation torture [1], or through packet dropping and misrouting. Selfish and malicious nodes misbehave, and, intentionally or unintentionally, attack on the robustness of the MANET and produce congestions. Finally, a *hacker node* might try to intercept the information exchanged between the nodes. Such violation is materialized through wormhole, impersonation, or Sybil attacks [2]. Selfish, malicious, and hacker nodes fabricate attacks against physical, link, network, and application-layer functionality.

In the literature several trust and security establishment solutions that rely on cryptographic (symmetric or public) methods, authentication codes and hashing chains have been proposed. Such schemes provide strong authentication of the end-entities, integrity and confidentiality of the messages, non-repudiation of the transactions and availability of resources. These security schemes apply to different layers of the OSI model. For the network layer, the proposed solutions aim to protect routing procedures from attacks (e.g., propagation of false routes) and packet forwarding from selfish dropping or misrouting. The disadvantage of these schemes, as far as their applicability to MANET is concerned, is their computation requirements. They are considered processing intensive and battery hungry. Additionally, some rely on *a-priori* trust between the nodes, which is not always the case, or they are based on centralized trusted third-parties, which are ephemerally present or absent. For the avoidance of the selfish nodes' effects on the cooperation functions of a MANET and the consolidation of the network robustness, a class of methods, referenced to as cooperation enforcement methods, is considered more appropriate. These, recently introduced, distributed, and lightweight methods contribute to the trust establishment between MANET's nodes without prior knowledge of the nodes' behavior. They apply to the network layer of a MANET, and their primary goal is to protect or enforce the two elementary functions of this layer: *routing* and *packet forwarding*.

This paper surveys the cooperation enforcement schemes that have been proposed, and is structured as follows: In Section 2, we address the threats and the attacks on the network layer of MANET and briefly describe the conventional security techniques that have been proposed. In Section 3, we introduce the cooperation enforcement schemes, their goals, and their taxonomy. In Section 4, we present the cooperation enforcement schemes that rely on reputations, and subsequently, in Section 5, we present those that are based on economic assets. Finally, we provide a comprehensive comparison of these methods in Section 6.

2. MANET, Attacks and Security

2.1. Attacks in the Network Layer Operations

The nodes of a MANET are actually mobile routers that build up routes dynamically. These routers can move randomly and insert themselves automatically into dynamic wireless topologies. They perform packet forwarding using the current routing information. A path form the source to the destination, that is, a route, can be established through well known routing protocols such as the ad hoc on-demand distance-vector routing (AODV, [3]), dynamic source routing (DSR, [4]), temporally ordered routing algorithm (TORA, [5]), zone routing protocol (ZRP, [6]), and destination-sequenced distance-vector (DSDV, [7]).

Selfish and malicious nodes take advantage of MANETs idiosyncrasies to misbehave, or attack. As far as the network layer of a MANET is concerned, the following types of attacks have been reported:

- Impersonation or spoofing. Such an attacker will try to spoof a node that resides in the route of the data flow of interest [8]. Such an attack can be materialized since the conventional routing protocols (e.g., AODV, DSR, TORA, ZRP) do not support authentication of IP addresses. A similar threat is called Sybil attack [2]. An attacker does not only impersonate one node, but it assumes the identity of several nodes, and, thus, undermines the redundancy of many routing protocols [9].
- Sinkhole, where an attacker tries to attract all the data sent by its neighbors. This attack is the basis for for example, eavesdropping [9]. Sinkhole attackers present themselves to adjacent nodes as the most attractive relay in a multi-hop route.
- Wormhole, where a malicious node uses a path outside the MANET (tunnel) to forward packets to another, colluding, node in the fixed network [10]. According to [10], the route discovery methods of on-demand routing protocols are violated by avoiding the normal route and by forwarding the RREQ packets directly to the destination.

- Routing 'fabrication,' where an attacker tampers with the normal routing procedures. It is achieved through alteration of the routing messages' fields (e.g., poisoning of DSR routing caches) or by the insertion of false routing messages (e.g., falsifying route error messages). Routing 'fabrication' produces denial-of-service (DoS) and partitioning of a MANET. In [11] several threats are identified, which are materialized through the modification of the routing messages' fields, such as modified sequence number, hop counts, or source route.
- DoS and flooding. They are considered as indirect results of the aforementioned attacks [9]. A direct DoS attack, introduced in [12], is the sleep deprivation torture. One node, or colluding nodes, continually request the services offered by the target node. This consumes the battery of the target, which goes into an idle or power preserving state.

2.2. Conventional Security and Authentication Methods

Several methods have been proposed that rely on cryptographic systems, symmetric or public key, and hash chains, to provide confidentiality, authentication, integrity and non-repudiation services to MANET.

Solutions of public keying incorporate a centralized or distributed certification authority (CA). Zhou and Haas in Reference [13] have proposed a distributed key management scheme, based on threshold cryptography [14]. In this (n, r) threshold scheme there is not a centralized CA, but n distributed CA servers. In Reference [15], the CA functions are distributed through a threshold secret sharing mechanism, in which each ad-hoc node holds a secret share. In [16] an online CA service in MANET that is based on threshold cryptography, called MOCA, is described. In the MOCA framework, n nodes provide the functionality of a single CA. In [17], GSM/GPRS technologies are proposed, enabling the ad hoc nodes to access CA services. An offline CA is considered in Reference [18] to decide which nodes can join the network, and to assign a unique identity to each one. Each node holds a copy of the CA's public key, so that it can validate the certificates of other nodes [18].

In a MANET, digital certificates are employed to protect both routing messages as well as packet forwarding. Secure routing protocols, such as ARAN [19], SAODV [20], and forwarding modules, such as TRM [21], involve CAs. ARAN secure routing protocol [19] requires the use of a CA whose public key is known to all the nodes. Each node keys are generated in advance, and exchanged through an out of band relationship with the CA server. Before entering the ad hoc network, each mobile node should obtain its certificate from the CA. Beyond public key solutions; existing routing protocols have been enhanced to incorporate symmetric cryptography and hash chains. The secure routing protocol (SRP, [22]) relies on pair wise key, distributed between all pairs of communicating nodes, for authentication of the nodes. SEAD [23] and Ariadne [24] make sure that all nodes on a route are authenticated and are based on DSDV and DSR routing protocols, respectively. SEAD uses symmetric keys for authentic distribution of the hash chain seed. It incorporates one-way hash chains to provide authentication of routing messages. Ariadne is based on symmetric keys for pair wise key distribution between all nodes, and on hash chains for node authentication. It uses a variant of the Tesla's public key infrastructure [25] for key and route discovery authentication.

3. Cooperation Enforcement Methods

Key-based schemes are considered computationally hard for MANET, due to complicated key management techniques, whilst *a-priori* knowledge of the identities is required for the initial key exchange. They efficiently support confidentiality services to prevent passive attacks (e.g., eavesdropping), authentication of nodes to establish end-to-end paths and integrity of messages to avoid fabrications. On the other hand, if the primary goal is the availability, the robustness of the network, and the overall throughput, then the cooperation enforcement techniques might fit better. These models face mainly the question of encouragement of collaboration between the nodes of a MANET so that the right implementation of routing and forwarding tasks is achieved. They can collaborate with the secure routing protocols to contribute to the creation of a comprehensive, but complicated, protection of a MANET. These models are categorized as reputation-based and credit-based. The first category is based on reputation building and the second is based on economic incentives (pricing or credit-based) to enforce cooperation.

The reputation-based models use the nodes' reputation to forward packets through the most reliable nodes. The reputation of a node increases when it carries out rightly the task of forwarding the packets that are dispatched by its neighbors, without altering

their fields. The models of this category support effective mechanisms to measure the reputation of other nodes of the network. They also incorporate techniques that isolate the misbehaving nodes, that is, those that show a low reputation value (RV). The reputation-based models can be further divided in two subclasses. The first subclass includes models according to which the nodes are only based in their personal observation of the reputation of their neighboring nodes (first-hand reputation information) to take a routing decision. The second category includes models according to which the nodes take into consideration the observations of other nodes in the network (second-hand reputation exchanges). The nodes in these models exchange information relative with the RV. If some node observes that another node does not behave rightly then it reports this observation to the remainder nodes of the MANET. The models that belong in this subclass deploy an effective mechanism to distribute this information.

For the credit-based models the packet-forwarding task is treated as a service which can be valuated and charged. These models incorporate a form of virtual currency to regulate the dealings between the various nodes for packet forwarding. They require the existence of tamper-resistant hardware or a virtual bank. The latter offers trusted-third party services to the nodes.

The categorization of the cooperation enforcement models is illustrated in Figure 1. It is worthwhile to mention that for the proposed reputation or credit-based schemes there is no common definition of the selfishness or malicious behavior. Additionally, a passive or active selfish behavior is not defined homogeneously in all the surveyed schemes. Thus, we use the term misbehavior to cover both phenomena, explaining the type of misbehavior each time.

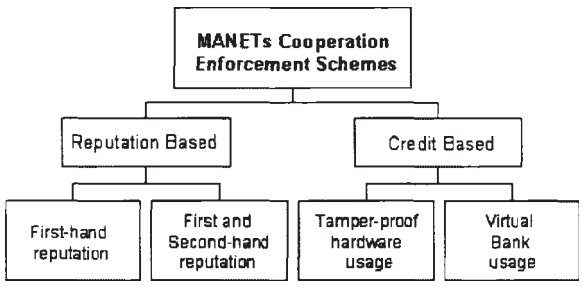

Fig. 1. The taxonomy of the cooperation enforcement models proposed for MANETs.

4. Reputation Based Models

4.1. CONFIDANT

Buchegger and Le Boudec proposed a scheme, called CONFIDANT [26], designed as an extension to an on-demand routing protocol, such as the DSR. CONFIDANT facilitates monitoring and reporting for a route establishment that avoids the misbehaving nodes. It is based on the assumption that the packets of misbehaving nodes are not forwarded by fair nodes. If, however, a node was incorrectly accused or turns out to be a repentant and no longer malicious, re-integration into the network is possible. CONFIDANT employs four functional components relying on each node, which include: (a) a monitor, (b) reputation records for first-hand and trusted second-hand observations about routing and forwarding function of other nodes, (c) trust records to control the trust that is given to received warnings, and, (d) a path manager to take routing decisions that avoid malicious nodes. The term reputation is used to evaluate routing and forwarding behavior according to the network protocol, whereas the term trust is used to evaluate participation in the protocol. Nodes monitor their neighbors and change reputations accordingly. Specifically, a node can detect selfish behavior of the next node in the source route either directly, by promiscuously sensing the transmission of the next node, or indirect, by observing routing protocol misbehavior. The *Monitor* component registers these deviations. As soon as a specific misbehavior occurs, the *Reputation System* is called, and ALARM messages are sent by the *Trust Manager*. Outgoing ALARMS are generated by the node itself after having experienced, observed, or received a report of malicious behavior of another node. They convey warnings of malicious nodes presence. The recipients of the ALARM messages, so-called friends, are maintained in a friends list. Incoming ALARMs that originated from 'strangers,' are checked for trustworthiness before triggering a reaction. The disadvantage here is the requirement of a pre-existed trust relationship. If there is sufficient evidence that the node reported in the ALARM is malicious, the information is sent to the *Reputation System*. This manages a table consisting of entries corresponding to nodes and their ratings. A rating is modified if two conditions coincide: (i) there is sufficient evidence of malicious behavior, and, (ii) a misbehavior occurs a number of times, exceeding a threshold to rule out coincidences. The ranking of a node is changed according to a rate function. This

function features the greatest weight for own experience, a smaller weight for observations captured in the neighborhood and the smallest weight to the reported, second-hand, experiences. If the ranking of a node has deteriorated so much as to fall out of a tolerable range, the *Path Manager* is activated. This component excludes routes containing misbehaving nodes and isolates them, ranks the paths in a cache, and forwards an ALARM about this node.

The first version of CONFIDANT was vulnerable to rumor spreading phenomena [27]. In a recent enhancement, this problem has been addressed through a *Bayesian* model [28,29] that classifies and excludes the liars. In this enhanced version, both positive and negative reputations are used to calculate a 'cooperation factor.' This factor consists of the frequency of misbehavior in relation to the cumulative activity of the node. The positive and negative experiences collected by a node should reveal the same sort of information for a node as what is gathered by the other nodes. Every node i keeps a cooperation factor of every other node j, expressed as Rij. This factor is expressed as a function of α and β, whereas, α and β is the number of misbehaviors and regular behaviors, respectively. These numbers are updated based on recent experiences. A recommendation is accepted if it is compatible (in the Bayesian model), that is, if the recommended *RV* is not completely different. This technique reduces the impact of the false accusations.

CONFIDANT does not use tamper-proof hardware. For a misbehaving node, it is hard to know the entries of its reputation in other nodes or to modify its reputations. However, it is still possible to alter the values of α and β or to change its identity. Only identities generated with cryptographic means can reduce this threat. The Bayesian approach reduces the impact of tampering with α and β. If values are not compatible with each other the algorithm will just ignore them. Evil nodes could only change the values with a small amount which is tolerable by the system.

4.2. CORE

This scheme, introduced by Michiardi and Molva in Reference [30], relies on the DSR routing protocol. It stimulates node collaboration through monitoring of the cooperativeness of nodes and a reputation mechanism. It uses first and second-hand experiences, combined by a specialized function. This function is used by the *Watchdog* mechanism for the evaluation of other nodes' behavior. If the observed behavior is different than the outcome of this function then the rating of the observed node is altered. Each node of the network monitors the behavior of its neighbors, with respect to the requested function, and collects observations about the execution of that function. These observations are recorded to the Reputation Table (RT), maintained by each node. Each row of the table corresponds to a neighbor node and consists of four entries, regarding the monitored function: the unique id of the node, a collection of recent (first-hand) observations made on the node's behavior, a list of the recent second-hand *RVs* provided by other nodes, and the *RV* evaluated for the monitored function. Thus, each node maintains one RT for each monitored function. Finally, a global RT is used to combine the different *RVs* calculated for the different functions.

CORE differentiates the *RVs* between *subjective reputation* ($[-1, 1]$), *indirect reputation* (positive reports by others), and *functional reputation* (e.g., when packet forwarding has greater effect than routing), which are weighted to provide a combined *RV*. The formula used to evaluate the *RV* avoids false detections by using an aging factor that gives more relevance to past observations. However, such an approach is vulnerable to an attack where a node can build up a good reputation before misbehaving. The *RVs* evaluated for each entry of the RT vary. A positive *RV* is decremented along time. So, if a node enters in an idle mode, its reputation has to be decreased, even if during the active time (i.e., when communicates) it cooperates to the network operation. Reputation is decreased until it reaches a null value, which corresponds to a neutral behavior. Furthermore, if the monitored function provides a reply message (e.g., the route reply of the DSR), reputation information can also be gathered about non-adjacent nodes. In this case, only positive ratings are assigned to the nodes that participate to this function.

The CORE scheme is immune to attacks performed using the mechanism itself: no negative ratings are spread, and, thus, it is impossible for a node to maliciously decrease another node's reputation. However, two or more nodes may collude (i.e., send positive rating messages) in order to increase their reputation. To prevent such phenomena, the CORE implicitly provides some protection, since subjective reputation has more impact (i.e., weight) than the indirect. CORE allows MANET's nodes to gradually isolate misbehaving nodes: when the reputation assigned to a neighboring node falls below a predefined threshold, the service provided to this node is interrupted. Misbehaving nodes can, however, be reintegrated in the network if

they increase on purpose their reputation, by cooperating to the network operation. CORE does not discriminate malfunction and misbehaving nodes. Additionally, a second chance mechanism is not consolidated, as in OCEAN in [31], and, hence, a malfunctioning node can't rebuild its reputation when it recovers from temporal problems. Finally, the CORE mechanism assumes that every node will use identical calculations of the *RV*, assigning the same weights to the same functions. This might not be the case, since in MANETs the devices are equipped with different resources and provide discrete services, and, hence they prefer to use difference levels of importance on functions.

4.3. SORI

The secure and objective reputation-based incentive scheme for ad-hoc networks, introduced in [31], focuses on the packet forwarding function. SORI, consists of three basic components: *Neighbor Monitoring*, *Reputation Propagation* and *Punishment*. A promiscuous mode is assumed, and a node is capable of overhearing the transmissions of its neighbors and to maintain a neighbor node list. Each neighbor's forwarding function is linked with two parameters. The $RF_N(X)$ (request-for-forwarding) is used to indicate the total number of packets that node N has transmitted to X for forwarding. The index $HF_N(X)$ (has-forwarded) corresponds to the total number of packets that have been forwarded by X and noticed by N. Given $RF_N(X)$ and $HF_N(X)$, N creates a record, called local evaluation record. This record, denoted by $LER_N(X)$) for the neighbor X, contains a confidence metric that is used to depict how confident the node N is for its judgment on the reputation of X. The more the packets transmitted to X for forwarding, the higher the confidence about the trustworthiness of X.

SORI combines features of the fist-hand schemes and those that use reputation spreading. In SORI the nodes exchange reputation information only with their neighbors. This way a no-cooperative node will be punished by all of its neighbors (who share the reputation information about its misbehavior), instead of just the ones who are directly affected by this node. Each node N periodically updates its $LER_N(X)$ for each neighbor node X based on the current values of $RF_N(X)$ and $HF_N(X)$. The updated record is broadcast to the neighborhood if the ratio $RF_N(X)/HF_N(X)$ has been significantly changed. Node N uses his own $LER_N(X)$ and the respective values of its neighbors to calculate its overall evaluation record of X, denoted as $OER_N(X)$. To do so, it takes into account the credibility of the nodes which contribute to the calculation of the reputation. This makes it difficult for an attacker to test multiple identities, trying to impersonate one identity in order to improve its reputation. If the $OER_N(X)$ is lower than a predefined threshold, node N takes a punishment action by probabilistically dropping the packets originated from X. This mechanism, as mentioned in [32], is designed to treat generously the nodes that do not intentionally drop packets.

In [32], a complementary security mechanism is proposed to deal with a node that uses the following attacks: (1) impersonation of an adjacent node's id, ranked with a good reputation, in order to send more packets, and, (2) impersonating a distant node's id, ranked with a good reputation, to broadcast fake observation information in order to boost its reputation. This mechanism is based on a one-way hash chain and message authentication codes (MACs). Finally, SORI takes no countermeasures to prevent collusion.

Liu and Issarny introduce a reputation model that incorporates *time* and *context*, along with mechanisms to support reputation formation, evolution and propagation [33]. The scheme is not focused only on the network-level functions, but on various types of services, such as a web service (e.g., ad-hoc discussion forums), and, thus, it applies to software agents, as well. It provides defense measures against the following types of attacks: (1) *Inactivity*: This refers to the free-ride attack, where an agent denies sharing the reputation information with its peers, (2) *Defame*: This refers to the propagation of a victim's reputation that is lowered on purpose, and, (3) *Collusion*.

In this model the agents maintain separate *RVs* for other agents that provide a service or a recommendation. These are denoted as *service reputation* (*SRep*) and *recommendation reputation* (*RRep*), respectively. The reputation of an agent builds over time, and, thus, to consolidate time-sensitivity into the scheme, a fading factor is introduced. The higher the value of this factor, the more emphasis is given to the recent behavior of a node. The scheme is context aware, as well. An agent estimates the trustworthiness of the service C provided by another node, based on his own observations and the observations of the other agents that have used C.

In some cases there is insufficient reputation information in the context of C, and adequate reputation information in the context of a relative service, say C'. To measure the relevance between the two different contexts, the authors incorporate concepts borrowed

from ontology trees and DAML (DARPA Agent Markup Language). Additionally, a parameter is used to reflect the agent's reliance on the context related reputations.

The scheme utilizes three functional components (managers) running on each agent: an *experience*, a *recommendation* and a *reputation* manager. The *experience manager* records the previous direct experiences of a service. After each interaction, agents can give a score of satisfaction. This score is usually subjective, and depends on multiple factors. The model scores the satisfaction based on the quality of service that an agent receives from its peer. The *recommendation manager* stores the recommendations collected from its peers, exchanges reputation information with peers, and manages a table of *RReps*. Reputations are exchanged periodically throughout the ad hoc network. Each agent communicates only with those nodes that have a high *RRep* value assigned to them. With a new experience available, an agent updates the *RRep* of the recommender of the newly interacted peer. In the scheme, the reputation is considered subjective. Thus, any deviation between the experiences obtained by an agent X and the recommendation that was made for the same service by another agent Y is accepted. A deviation is reflected in the procedure that updates the *RRep* values. When considerable deviations occur, Y is ranked as unreliable recommender. If an agent never recommends, its *RRep* will be ranked as ignored, and the others will hesitate to exchange reputation information with it. Thus, for a node, there is only one way to maintain its *RReps* to a decent level. That is to recommend actively and honestly. The *reputation manager* takes inputs from the other components to calculate the *SRep* of a node. It assigns a greater weight for its own experience and a lower for the collected recommendations.

4.4. OCEAN

The observation-based cooperation enforcement in *ad hoc* networks, proposed in [31], introduces an intermediate layer that resides between the network and the MAC layers. This layer helps the nodes to make intelligent routing and forwarding decisions. It is designed on the DSR level, but its principles can be applied to other routing protocols, as well. OCEAN relies only on first-hand observations. Every node maintains ratings for each neighboring node and monitors their behaviors through promiscuous observations. Positive or negative events are recorded through the reaction of a neighbor that is expected to forward a packet. Rating is initialized to a neutral value. Due to empirical studies, the absolute value of a decrement is chosen to be bigger than the value of an increment. When the rating of a node drops below a threshold, called *faulty threshold* the node is added to a faulty list. This list is constructed using the node's personal experiences and is attached (as a field called *avoid-list*) to the route request (RREQ) message of the DSR protocol in order to be flooded. A route is rated good or bad, based on whether the next hop in the route belongs to the *avoid-list*. The receiver of an RREQ decides to drop it or to further process it (through relaying or a route reply), if the intersection of the *avoid-list* and the DSR route in the RREQ packet is void. In this way, each node along a route, makes its own decision about the trustworthiness of other nodes, and has control only over routes that it belongs to.

Every node rejects the data packets arrived from the nodes belonging to its faulty list. Thus, misbehaving nodes are eventually isolated. However, a *second-chance* mechanism is used to allow nodes that misbehaved in the past to become operational again. After a certain period, a misbehaved node is excluded from the faulty list and assigned with a neutral rating.

OCEAN uses a different policy to deal with nodes that do not participate in the route discovery process. This policy, affected by the credit-based models, requires no tamper-proof hardware or a central server. Each node measures the behavior of its neighbors by directly interacting with them. Nodes track the forwarding balance with their neighbors by maintaining one counter, called *chip count*, per node. The counter increases when requesting a node to forward a packet and decreases with an incoming request from that node. Assume that a node B did not participate on the establishment of route with a source node A. If B demands from A to forward its packets, then, A will punish B and reject its requests, as long as the chip count for B is low. This policy is considered unfair for nodes belonging to the perimeter of the MANET, since they are not frequently required to forward messages on behalf of others. Penalizing these nodes might cause the network to shrink. To overcome such phenomena, the OCEAN introduces a *chip accumulation rate* (CAR) parameter, which expresses the rate at which all *chip count* in the network are increased per unit time. Thus, the forwarding of the packets sent by circumferential nodes is enforced, even at a reduced rate. CAR can't be adjusted easily and there no mechanism to prevent a node to change it at will.

Promiscuous mode of operation does not always provide sufficient evidence on the trustworthiness of a monitored node. A monitored node may not be able to relay the packet due to the low quality of the wireless link. Additionally, other reasons, such as network interface restart, or low battery, might affect the relay task. Thus, nodes should incorporate the logic to discriminate the cases where other nodes malfunction and misbehave, and not to faulty punish low capacity nodes. The introduced *second-chance* mechanism was designed to overcome the potential problems that might be observed due to the absence of such intelligence.

Hacker nodes might take advantage of the *avoid-lists* of the OCEAN, which are included on the RREQs, and tamper these lists to perform wormhole attacks. Simulations showed that OCEAN performs well under the presence of such attacks as long as the network topology is not static.

The *faulty threshold* reflects the speed and accuracy of misbehavior detection. A low value adds nodes quicker to a faulty list. High values might decrease the detection speed. Detection speed is important for the models that use first-hand observations, since the evaluation of a new joined node takes place from scratch. In contrast, schemes that use second-hand reputations obtain trust indexes for remote nodes that eventually will become adjacent, and thus, operate proactively. Simulations have showed that with a low *faulty threshold*, OCEAN performs better than a generic scheme that uses second-hand information. This is because OCEAN is more resilient to rumor spreading. However, OCEAN is sensitive to the tuning of the *faulty threshold* parameter; second-hand schemes perform better over a broader range of tunings. OCEAN, additionally, is not effective in reducing the throughput of misbehaving nodes. Finally, OCEAN, as SORI, takes no countermeasures to prevent collusion.

Dewan, Dasgupta, and Bhattacharya introduce a first-hand reputation information model, described in [34], which is based on the AODV routing protocol. It uses acknowledgements to observe the behavior of adjacent nodes, rather than promiscuous operations. The reputation of a node is based on its history of relaying packets, which is used by the neighbors to ensure that their packet will be forwarded. The source node finds a set of paths to a destination, using the routing protocol. Then the first hop node forwards the packet to the next hop with the highest reputation and the process is repeated till the packet reaches its destination. The destination acknowledges the packet to the source, which, in turn, updates the corresponding entry of the reputation table by rewarding (i.e., $+1$) the first hop. The intermediate nodes in the route reward their respective next hop in the route and update their reputation tables accordingly. If a no-cooperative node resides in the route, the data packet might not reach its destination. As a result, the source will not receive an acknowledgment for this data packet within a predefined interval. In such a case, the source penalized (i.e., -1) the first hop on the route. The intermediate nodes propagate this penalty in the route up to the node that dropped the packet. Thus, the reputation value of the nodes that are between the misbehaving node and the source, including the misbehaving one, gets a recommendation of -1.

Assume that during the path establishment process a misbehaving node claims that it maintains a path to the destination, in order to be part of the route. Assume, also, that the same node drops the data packets or does not broadcast route error (RERR) messages to inform about a broken path. Then, the proposed scheme will force the upstream node in the route to give a negative recommendation to this node. Once the reputation of the node falls below a threshold value, it will be considered as malicious and will eventually be ostracized. If a node avoids participating in the route discovery, his reputation will not increase and will experience significant delay when sending its own data packets. This is because all the nodes in the route to the destination will assign a lower routing priority to the packets of this node.

When using a scheme that works promiscuously, a node can be sure whether its neighbors forwarded its packets. On the other hand, it can not be sure if its packet reached the destination or even the next hop. The introduced mechanism has an advantage. It acknowledges the deliveries of packets. The price that is paid for this is the increased traffic volume. The destination nodes acknowledge all the packets they receive. These acknowledgements must reach a sender, following the reverse path. According to an alternative solution, proposed in Reference [34], the sender intercepts the TCP-layer acknowledgments to ensure that its previous packets have reached its destination. This approach reduces the traffic overheads considerably, even if it needs access to information across the layers of the OSI stack.

A drawback of the proposed scheme is the bottleneck introduced to the nodes with good reputation, since they are frequently preferred as the next-hop, no matter their distance. A load balancing method that

balances the load among the well-reputed nodes might overcome such phenomena. The scheme improves the throughput at the cost of a higher number of route discoveries with a relatively small increase in the average route length. It is claimed in Reference [34] that the scheme is resilient to the collusion of nodes. It does not include an explicit mechanism for giving a second chance to nodes that experience relay failures or have low recourses. However the authors propose two techniques that extend the basic scheme and handle these situations.

5. Credit-Based Models

5.1. Sprite

The simple, cheat-proof, credit-based system for mobile ah-hoc networks was proposed in Reference [35]. It does not require tamper-proof hardware to prevent the deviation of payment units, but incorporates a centralized credit clearance service (CCS). When receiving a packet, a node keeps the signed receipt of this packet, which was generated by the source node. Sprite assumes that each node has a public key certificate published by a CA. When the node has a fast connection to the CCS it reports the packets that it has received by uploading its collected and signed receipts. Sprite prevents any cheating by making it unattractive even in the case of collusion. When a node sends its own packets it loses a credit (virtual money), because other nodes incur a cost to forward these packets. In order to gain a credit and be able to send packets later, a node must forward packets on behalf of others. CCS charges the sender based on the number of receipts, the number of intermediate nodes left to reach the destination, if any, and whether the destination has sent a receipt. The mechanism is designed to be resilient against the following selfish actions: (1) after receiving a packet, the node saves a receipt but does not forward it, (2) the node has received a packet but does not report the receipt, and, (3) the node does not receive a packet but falsely claims that it has received it.

Receipts, used as a proof of forwarding, might produce a weak point. A receipt is generated by the source of a packet, signed with the source's secret key and appended to the packet that needs to be forwarded. Subsequent nodes on the path need that receipt that can be gathered both by receiving the packet (but not necessarily forwarding it) and by colluding with other nodes. The authors in Reference [35] take into account this possibility when designing the model such that this type of collusion will not pay. However, if the nodes had means of exchanging receipts, other than their radios, this collusion scenario could become attractive. Another critical issue is the rewarding of the fair nodes, that is, those that forward packets. Ideally, a node that tried to forward a packet should always be rewarded (no matter if the transmission was successful), because it has consumed recourses for this action. Even though some wireless systems provide link-layer acknowledgements, these are not universal. Moreover, any change on the basic network functions must be avoided. Given these, the CCS believes that a node has forwarded a packet if there is a successor of that node on the path reporting a valid receipt of the packet.

A potential disadvantage of sprite is the assumption that a fast connection to the CCS is needed for the reporting of the obtained receipts. An extension of the basic sprite provides integrity during packet exchanges, and is based on digital signatures [35]. Finally, a generalization of sprite that encourages the participation of nodes during the route discovery is also introduced.

5.2. Token-Based Cooperation Enforcement

This scheme, introduced in Reference [36], protects both routing and packet forwarding in the context of the AODV protocol. It is self-organized, without assuming any *a-priori* trust between the nodes, or the existence of any centralized trust entity. It isolates the misbehaving nodes and employs threshold cryptography to enhance the tolerance against these nodes. The scheme is fully localized (one hop), and its credit-based strategy produces overhead that is significantly decreased when the network is not harmed. It assumes that the nodes operate promiscuously. Multiple attackers may coexist, but it is assumed that they collude locally. However, the collusion impact is minimized, since it is assumed that each node has a unique id, and the underlying cryptography is strong.

The system's secret key is shared among the network nodes, and each node maintains only a limited portion of it. Each node carries a token, signed with the system's secret key as derived from the threshold cryptography process. The node's neighbors can verify this token. Nodes without a valid token are isolated because all the legitimate neighbors will not interact with them during routing and forwarding. The validity period of a token is time-bounded. Before the token expires, each node must renew it,

through its neighbors. Nodes collaboratively monitor others to detect any misbehavior. Once an attacker is detected, its token is revoked, and, thus, the attacker is blocked from network access. This mechanism allows a fair node to collect credits and to renew its token less frequently.

This token-based cooperation enforcement scheme includes the following four components: (1) *neighbor verification* which verifies whether neighbors are legitimate, (2) *neighbor monitoring* that monitors the behavior of each node in the network and detects attacks (3) *intrusion reaction* that generates alerts and isolates attackers, and, (4) *security enhanced routing protocol* that performs the routing protocol including its security extensions. The *neighbor verification* employs the RSA-based cryptographic primitives. A valid token is constructed using a group signature, using the polynomial secret sharing technique introduced in Reference [14] and used in Reference [15]. This assures that at least k neighbors agree to issue or renew the token.

The key setup complexity and the requirement for k nodes for the threshold cryptosystem are considered incompatible with high mobility MANETs, and call for a large and dense network. Furthermore, the validity period of a token increases proportionally with the duration of the node's fair behavior which calls for low mobility, as well.

In terms of energy efficiency, the scheme requires from each node to constantly sense the channel, a process that introduces energy consumption. A possible variation enforces each node to periodically monitor the channel. Finally, the computational complexity is mainly introduced by the asymmetric cryptography, and the storage overhead that is caused due to the monitoring mechanism. The authors mentioned that when employing light-weighted cryptography the computation complexity is decreased, whilst hashing techniques might decrease the storage overhead.

5.3. Ad hoc-VCG

Energy-efficiency is a parameter of high importance for the MANET routing protocols. It ensures that a packet gets routed along the most energy-efficient path. The total energy of a routing path is the sum on the emission energy levels used at the source and at each intermediate node. If a node is chosen as an intermediate in an energy efficient path, this eventually will drain its battery. When this node realizes that its battery will decrease with time, it might refuse to forward packets on purpose, acting selfishly.

The ad hoc-VCG scheme, proposed in Reference [37], is a credit-based model which deals with this issue and introduces a *second-best sealed* type of auction. A pricing question arises concerning the amount of the payment a node should ask to forward packets. One answer might be the cost it incurs when forwarding the packet. The ad hoc-VCG scheme estimates this cost through the *cost-of-energy* parameter, c. The value of c_i is expressed in, for example, Euros per watt, and is discrete for each node j in the network. If the forwarding of a unit-size packet requires an emission of a signal equal to P^{emit} watts, then node j asks for a payment of $c_i \cdot P^{emit}$ Euros to the forward. On the other hand, if a node does not get a payment that is sufficient to cover its forwarding costs, it refuses to forward. In that sense, participation is always voluntary. The *cost-of-energy* parameter is a time depended function, which takes into account each nodes' preferences, such as power autonomy. The nodes determine the energy emission levels to reach their neighbors using a signaling process and additional control fields on packets. They send a packet with high energy to indicate their emission signal strength in the header, and, on return, they receive a packet from neighbors which contains the signal strength at which they received the packet. Finally, they adjust their emission power accordingly. The nodes also communicate their *cost-of-energy* to their neighbors.

The ad hoc-VCG works on top of the DSR. It consists of two distinct phases. During the *route discovery* phase, a weighted graph is computed. The vertices represent network nodes; the weighted directed edges correspond to the payments a relaying node has to receive to forward a packet along this edge. A destination node collects all the weights of the edges, and then computes the shortest path in the graph from the source to destination, which corresponds to the most energy-efficient path. During the *data transmission* phase, packets are forwarded along the shortest path and payments are made to the intermediates.

However, there is a limitation in the route discovery phase: the nodes have to indicate the signal strength at which they emit and they also need to forward information regarding their neighbors' received signal strengths. This provides means for nodes to cheat. For instance, a node j that overstates its *cost-of-energy* c_i will receive a larger payment if it is chosen to be on the 'shortest path.' Similarly, a node may profit from sending false information regarding received signal strengths. To overcome these issues, the ad hoc-VCG

makes payments attractive enough such that the nodes will not try to cheat. Each node's incentive is to reveal its true cost. In Reference [37] it is proved that it is not profitable for a node to alter the *cost-of-energy* parameter, because the final cost is higher than the extra profit the node makes.

The ad hoc-VCG is robust when only one cheating node exists. It might fail in the presence of collusions of nodes who try to maximize their payments. An additional issue is the excessive overhead. It requires complete knowledge of the network topology to construct the graph, which creates significant overhead during the route discovery phase. Finally, it does not focus on the actual payment delivery, but only on the estimation of the payments. Existing payment delivery schemes, such as the sprite [35], might be combined with the ad hoc-VCG.

6. Discussion and Comparison

6.1. Discussion

6.1.1. Reputation-based models

The majority of the models assume that the *RV* can be trustworthily used for the forecast of future behaviors. Unfortunately, the past behavior can't always indicate future behavior. This is due to the fact that the end-systems are under the control of humans and are considered as passionate [12], showing a non-deterministic behavior.

Another issue is related with the so-called aggregated (global) *RV*. Each node maintains a unique *RV* for each other node with which it interacts. This value consolidates all the individual values for each networking service that a node provides. However, this aggregated value allows a node to hide his misbehavior with regard to an operation, and, hence, the aggregated value does not reveal the importance that is given to different tasks.

The second-hand models lead faster to a robust network. Malicious and selfish nodes are located faster and are punished more strictly. Additionally, second-hand models can identify and report remote misbehaviors before these effect the operations of a distant node. On the other hand, rumor spreading should be avoided. As stated in Reference [30], DoS attacks might occur if negative reputations spread around. Moreover, in Reference [27] the authors mentioned that the usage of only positive recommendations minimizes the effect of rumor spreading phenomena. Finally, the trustworthiness of the recommender (i.e., second hand) should be evaluated and examined, since recommendation is considered as a discrete function for second hand reputations. In Reference [38] the *RV* of a target is computed over a recommendation path. Second-hand indications received among different paths are then combined, through an averaging method originally introduced by Beth *et al.* in Reference [39]. The work in Reference [38] proposes a computation of the trust value of a target which takes into account nodes with a low trust value of the recommendation function. In such a case, it would be easy for a malicious node (i.e., with low *RV* for the recommendation function) to distribute forged information about other nodes, and to eventually generate a DoS attack. If the *RV* for one node falls under a specific threshold, then this node is considered as malicious. A question of fairness arises for specific types of nodes. If a node is equipped with low resources (e.g., CPU, memory), this node will lose progressively its reputation and, eventually, it will be considered as selfish. Additionally if a node is at the edge of the network it will not forward the total number of packets it senses, and this should be considered as a normal, rather than selfish, behavior. Finally, the promiscuous mode relies on omni-directional antennas and assumes symmetric bi-directional links. This mode is used by the reputation methods for the ranking of adjacent nodes, but it fails to capture transmission errors, and to distinguish those from potential misbehaviors.

6.1.2. Credit-based models

If the transmission of a packet from the node-sender to the node-destination it is considered as a deal, then it should be decided which node will be debited and which will be awarded with credits. In case the receiver is the only entity charged, then DoS attacks can be easily materialized through continual transmission of packets to the receiver node. Similar problems arise in case the cost is split between the sender and the destination. If the sender colludes with the intermediate nodes of the path in order to return spent credits to him, then the destination is the only entity that is charged. In case the sender is the only entity that is charged, then the intention of transmission of 'useless' messages will be revealed to the sender, which can lead to degradation of the network throughput. Finally, the use of specialized, tamper-proof, hardware, so each node cannot modify the number of credits that corresponds to him, is not suggested for open networks, such as MANET.

Table I. Comparison of the cooperation enforcement techniques.

	Payment or reputation	Robustness against misleading nodes	Robustness against collusion	1st-hand and 2nd-hand observations	Cryptographic authentication	Promiscuous Observations mechanism
OCEAN	Both	Selfish only	No	Global	No	Yes
Dewan *et al.* [34]	Reputation	Selfish only	Yes	Global	No	No
CONFIDANT	Reputation	Yes	Yes	Global	No	Yes
CORE	Reputation	Selfish only	No	Global	No	Yes
SORI	Reputation	Selfish only	No	Global	Yes	Yes
Liu and Issamy [32]	Reputation	Yes	Yes	Context	No	—
SPRITE	Payment	Yes	Yes	—	Yes	No
TOKEN	Payment	Yes	Yes	—	Yes	Yes
VCG	Payment	Selfish only	No	—	No	No

6.2. Comparison

For the comparisons presented in Table I we have made the following assumptions:

- A selfish node saves battery life for its own communications and jeopardizes network's stability and throughput, by simply not participating in the routing or forwarding tasks.
- A malicious node intentionally tampers with the execution of routing protocols. It perpetrates integrity attacks by simply altering the protocol fields in order to subvert traffic, and deny communication to legitimate nodes (DoS).

In Table I, the field 'Robustness against misleading nodes' has the following meaning:

- 'Selfish only.' The scheme deals with the selfish nodes only
- 'Yes.' The scheme deals with selfishness as well as additional misbehavior. For example, in Reference [32], the proposed scheme deals with *Inactivity* and *Defame*.

Additionally, 'Context' means discrete *RVs* per service (e.g., routing, packet forwarding, etc.), and 'Global' means an aggregated *RV.*

A feature, not included in Table I, concerns the second-chance mechanism. This mechanism is used to recover the reputation of a node that was wrongly punished or accused, and eventually isolated. For example, OCEAN [33] incorporates such a mechanism, whilst other schemes, such as CONFIDANT [26], implicitly address this issue.

As opposed to the credit-based models, the reputation models do not assume that a node has to forward for others at least as many packets as it generates itself. On the other hand, the credit-based models encourage, but not enforce, cooperation, which is established only if nodes voluntarily collaborate.

The schemes that we have examined in our study do not assume the presence of tamper-proof hardware, as in Nuglets [40]. Although such hardware might be essential for authentication or messages' integrity, its applicability in MANET as complementary to the cooperation enforcement schemes is rhetorical [41], since the existence of such infrastructure achieves the avoidance of several malicious or selfish behaviors.

6.3. Conclusion and Future Work

Recent studies [42] have shown that the cooperation enforcement mechanisms increase the probability of a successful forward, and the performance for small networks (i.e., fairly short routes) is enhanced, as well. However, in small networks, with short routes, this benefit is more considerable than in medium to large scale networks [43].

An important issue for the cooperation enforcement models is that the identity of a node should be unique and remain permanent. A spoofing attack on a MANET that uses such a method will allow nodes to impersonate other's identities, and, hence, their *RVs*. The majority of the schemes do not support an identity management technique, whilst some assume the existence of an authority (i.e., CA) to support it.

A malicious node may reject incoming data packets during the forwarding function. Nevertheless, it can not reject the control packets of the routing protocol, because it will be self-isolated, and, eventually, will not be able to eavesdrop or construct wormholes. If it drops routing control packets, this will harm the performance of the network, since shortest routes will be rejected in purpose. Such a behavior is more difficult to be detected by the cooperation

enforcement models, especially by those that rely only on first-hand observations.

Currently we are analyzing the performance of the proposed reputation and credit-based models, in terms of throughput improvement and communication overheads. The goal is to evaluate these models using a common reference scenario. However, the identification of a common reference scenario arises many difficulties, due to radically different assumptions that are used for each scheme. Moreover, although simulation results are presented by each of the authors, the simulation configurations, the parameters that are measured, and the assumptions that are made, significantly vary. Further work is focused on issues related to fairness (e.g., handling the reputation of nodes that are in the edge of the MANET) and the time that the schemes require to converge on a final RV for a selfish node, as well. Finally we are further studying the proposed second-hand reputation models to identify how these models distribute the reputation information, how they mitigate attacks on the second hand reputation information, and what types of countermeasures (e.g., punishment) these mechanisms employ.

Acknowledgment

This work was partly undertaken and submitted as a study for the Wireless and Mobile Communication lesson of the Computer Science postgraduate studies, Department of Informatics, Athens University of Economics and Business, Greece. We thank Prof. G. C. Polyzos, E. C. Efstathiou for their advice and M. Chronopoulou for her contributions. We acknowledge the valuable contributions of J. Liu, P. Michiardi Y. R. Yang, and S. Eidenbenz.

References

1. Stajano F, Anderson R. The resurrecting duckling. In *Proceedings of 7th International Workshop on Security Protocols*, 1999.
2. Douceur J. The sybil attack. In *Proceedings of 1st International Workshop on Peer-to-Peer Systems (IPTPS02)*, March 2002.
3. Perkins CE, Royer EM. Ad-hoc on-demand distance vector routing. In *Proceedings of 2nd IEEE Workshop on Mobile Computer Systems and Applications*, February 1999.
4. Johnson DB, Maltz DA. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, Imielinski T, Korth H (eds). Kluwer Academic Publishers: Boston, 1996; 153–181.
5. Park VD, Corson MS. A highly adaptive distributed routing algorithm for mobile wireless networks. In *Proceedings of IEEE INFOCOM'97*, April 1997.
6. Haas ZJ. A new routing protocol for the reconfigurable wireless networks. In *Proceedings of IEEE 6th International Conference on Universal Personal Communication*, October 1997.
7. Perkins CE, Bhagwat P. Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers. In *ACM SIGCOMM'94 Conference on Communications Architectures, Protocols and Applications*, August 1994.
8. Karlof C, Wagner D. Secure routing in wireless sensor networks: Attacks and countermeasures. In *Proceedings of 1st IEEE International Workshop on Sensor Network Protocols and Applications*, May 2003.
9. Burg A. Ad hoc network specific attacks. In *Seminar Ad hoc networking: Concepts, Applications, and Security.* Technische Universität München, '03.
10. Hu YC, Perrig A, Johnson DB. Packet leashes: A defense against wormhole attacks in wireless ad hoc networks. Technical Report TR01-384, Department of Computer Science, Rice University, December 2001.
11. Michiardi P. Cooperation enforcement and network security mechanisms for mobile ad-hoc networks. Ph. D. thesis, Ecole nationale supérieure des telecommunications, December 2004.
12. Jøsang A. The right type of trust for distributed systems. In *Proceedings of ACM New Security Paradigms Workshop*, September 1996.
13. Zhou L, Haas Z. Securing ad hoc networks. *IEEE Network* 1999; **13**(6): 24–30.
14. Shamir A. How to share a secret. *Communications of the ACM* 1979; **22**(11): 612–613.
15. Kong J, Zerfos P, Luo H, *et al.* Providing robust and ubiquitous security support for mobile ad-hoc networks. In *Proceedings of International Conference on Network Protocols (ICNP)*, November 2001.
16. Yi S, Kravets R. MOCA: Mobile certificate authority for wireless ad hoc networks. In *Proceedings of 2nd Annual PKI Research Workshop (PKI03)*, April 2003.
17. Cheambe J, Tchouto JJ, Tittel C, *et al.* Security in wireless ad-hoc networks. In *Proceedings of 13th IST mobile and wireless communications*, June 2004.
18. Capkun S, Hubaux JP. BISS: building secure routing out of an incomplete set of security associations. In *Proceedings of ACM WiSe2003*, September 2003.
19. Sanzgiri K, Dahill B, Levine BN, *et al.* A secure routing protocol for ad hoc networks. In *Proceedings of 10th IEEE International Conference on Network Protocols (ICNP02)*, November 2002.
20. Zapata MG, Asokan N. Securing ad hoc routing protocols. In *Proceedings of ACM WiSe02*, September 2002.
21. Song JH, Wong H, Leung V, *et al.* Secure routing with tamper resistant module for mobile ad hoc networks. In *Proceedings of ACM MobiHoc2003*, June 2003.
22. Papadimitratos P, Haas ZJ, Samar P. The secure routing protocol (SRP) for ad hoc networks. *IETF Internet Draft, Work in progress*, December 2002.
23. Hu YC, Johnson DB, Perrig A. SEAD: secure efficient distance vector routing for mobile wireless ad hoc networks. In *Proceedings of 4th IEEE Workshop on Mobile Computing Systems and Applications*, June 2002.
24. Hu YC, Perrig A, Johnson DB. Ariadne: a secure on-demand routing protocol for ad hoc networks. In *Proceedings of 8th ACM International Conference on Mobile Computing and Networking*, September 2002.
25. Perrig A, Canetti R, Song D, *et al.* Efficient and secure source authentication for multicast. In *Proceedings of Network and Distributed System Security Symposium*, February 2001.
26. Buchegger S, Le Boudec JY. Performance analysis of the CONFIDANT protocol. In *Proceedings of 3rd ACM International Symposium, on Mobile Ad Hoc Networking and Computing*, June 2002.
27. Buchegger S, Le Boudec JY. The effect of rumour spreading in reputation systems for mobile ad-hoc networks. In *Proceedings of WiOpt03*, March 2003.
28. Buchegger S, Le Boudec JY. Coping with false accusations in misbehavior reputation systems for mobile ad-hoc networks.

Technical Report IC/2003/31, Ecole Polytechnique Federale de Lausanne, May 2003.
29. Buchegger S, Le Boudec JY. A robust reputation system for P2P and mobile ad-hoc networks. In *Proceedings of P2PEcon2004*, June 2004.
30. Michiardi P, Molva R. CORE: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Proceedings of 6th IFIP Communication and Multimedia Security Conference*, September 2002.
31. Bansal S, Baker M. Observation-based cooperation enforcement in ad-hoc networks. *Techical Report*, Stanford University, 2003.
32. He Q, Wu D, Khosla P. SORI: a secure and objective reputation-based incentive scheme for ad-hoc networks. In *Proceedings of IEEE WCNC2004*, March 2004.
33. Liu J, Issarny V. Enhanced reputation mechanism for mobile ad hoc networks. In *Proceedings of 2nd International Conference on Trust Management*, March 2004.
34. Dewan P, Dasgupta P, Bhattacharya A. On using reputations in *ad hoc* networks to counter malicious nodes. In *Proceedings of QoS and Dynamic Systems*, July 2004.
35. Zhong S, Chen J, Yang R. Sprite: a simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *Proceedings of IEEE INFOCOM2003*, April 2003.
36. Yang H, Meng X, Lu S. Self-organized network-layer security in mobile ad hoc networks. In *Proceedings of ACM WiSe02*, September 2002.
37. Anderegg L, Eidenbenz S. Ad-hoc-VCG: a truthful and cost-efficient routing protocol for mobile ad-hoc networks with selfish agents. In *Proceedings of 9th Annual International Conference on Mobile Computing and Networking*, September 2003.
38. Rahman A, Hailes S. A distributed trust model. In *Proceedings of New Security Paradigms Workshop*, September 1997.
39. Beth T, Borcherding M, Klein B. Valuation of trust in open networks. In *Proceedings of European Symposium on Research in Computer Security*, November 1994.
40. Buttyan L, Hubaux JP. Nuglets: a virtual currency to stimulate cooperation in self-organized mobile ad hoc networks. Technical Report No. DSC/2001/001, EPFL, January 2001.
41. Pfitzmann A, Pfitzmann B, Waidner M. Trusting mobile user devices and security modules. *IEEE Computer*, February 1997, 61–68.
42. Lamparter B, Plaggemeier M, Westhoff D. Analysis of cooperation approaches in ad hoc networks. In *Proceedings of WiOpt03 Poster Session*, March 2003.
43. Lamparter B, Plaggemeier M, Westhoff D. Estimating the value of co-operation approaches for multihop ad hoc networks. *Elsevier Journal of Ad Hoc Networks*, January 2004, 17–26.

Authors' Biographies

Giannis F. Marias received his diploma in Computer and Software Engineering in 1995, from the Department of Computer and Software Engineering, University of Patras, Greece and his Ph.D. in informatics and telecommunications from the University of Athens, Greece. He is a visiting lecturer and senior research assistant in the Department of Informatics and Telecommunications, University of Athens. He has participated in several projects realized in the context of EC frameworks (RACE, ACTS, and IST) and several national R&D initiatives. His research interests are in the fields of security, trust, and privacy in wireless, mobile and personal communications, multiple access protocols, spectrum agility, and mobile and pervasive computing. He has authored more than 50 scientific articles in the above areas in international journals and conferences. He has organized international workshops and participated on technical committees in several conferences and symposiums.

Panagiotis Georgiadis, associate professor in the Department of Informatics and Telecommunications of the University of Athens, holds a B.Sc. degree in Physics, an M.Sc. degree, and Ph.D. in Computer Science. He has been a regular member of the Senate of University of Athens (1992 -1994), Director of the Computer Systems & Applications Division of the Department of Informatics and Secretary General for Information Systems by the Greek Ministry of Finance (20/4/1997–16/1/2002). Associate Professor P. Georgiadis lectures on Operating Systems, Information Systems Analysis, Software Engineering, Simulation Systems, IT Security and Network Performance Evaluation at undergraduate and graduate courses. His research interests include Distributed Systems, Simulation and Management of Information Systems. He has authored more than 70 scientific articles in international journals and conferences, and contributed in national and European research projects.

Dimitrios Flitzanis received his diploma in Computer Engineering and Informatics in 2003, from the Department of Engineering and Informatics, University of Patras, Greece and his M.Sc. degree in Computer Science from Athens University of Economics & Business in 2005. His research interests are in the fields of Modern Broadband Services, Technoeconomic Analysis & Network Planning and in the area of security, trust and privacy in wireless communications.

Kyriakos Mandalas received his diploma in Informatics in 2004, from the Department of Informatics and Software Telecommunications, University of Athens, Greece and his M.Sc. degree in Computer Science from Athens University of Economics & Business in 2005. He has received Ericsson's (first) award of Excellence in Telecommunications. His research interests are in the fields of Modern Voice Services & Technologies (VUI design & programming, speech recognition) and in the area of security, trust, and privacy in wireless communications.