A Guide to Trust in Mobile Ad Hoc Networks

Shane Balfe, Po-Wah Yau and Kenneth G. Paterson,
Information Security Group, Royal Holloway, University of London,
Egham, Surrey, TW20 0EX, United Kingdom.
{s.balfe, p.yau, kenny.paterson}@rhul.ac.uk

***Abstract*—In this paper we examine issues of trust and reputation in Mobile Ad hoc Networks. We look at a number of the trust and reputation models that have been proposed and we highlight open problems in this area.**

I. Introduction

A Mobile Ad hoc NETwork (MANET), as described by the Internet Engineering Task Force (IETF) MANET working group, is a (temporary or permanent) autonomous network comprised of free roaming nodes (wireless communication devices) [21]. Nodes within these networks are typically ascribed the following characteristics: nodes move arbitrarily resulting in a dynamic network topology; communication links between nodes may be bandwidth-constrained; messages are typically routed in a multi-hop fashion; nodes may be powered by an exhaustable energy source and nodes may have limited physical security [21]. In addition to this, dedicated infrastructure elements within a MANET may be not present, ephemerally available, or need to be built from the ground up.

MANETs are attractive in military and emergency response settings as they may form dynamically in response to some immediate operational requirement. However, with this immediacy comes the problem that pre-established agreements dictating the terms in which nodes will collaborate may not be fully specified. It has been suggested that trust can play a role in mitigating this issues by helping to address node uncertainty [39]. Trust has also been proposed as a means of aiding service selection in the presence of multiple competing offers, enforcing node cooperation, assuaging worries over the leakage of sensitive information (such as personally identifiable information) and in establishing node identity [16].

Unfortunately, trust as a concept, whilst intuitively simple, is notoriously difficult to specify, particularly in terms of computer networks. One of the principle problems with trust is the variety of meanings that have come to be associated with it [51]. For example, in [37], Jøsang defines trust as a belief that one entity holds about another entity, based on past experiences, knowledge of entity behaviour and/or recommendations from trusted entities. McKnight and Chervany define trust as the situation where one is willing to depend, or intends to depend, on another party with a feeling of relative security, in spite of lack of control over that party, and even though negative consequences may arise [50]. However, both these definitions predominately focus on aspects of human-mediated trust relations, it is not immediately obvious how such a definition translates to autonomous computer networks. Compounding this issue are the problems with the related concepts of trusted and trustworthy which are often used, but rarely clearly defined. In the context of distributed systems, Anderson in [4] defines a trusted component as one whose failure can break the security policy of the system, while a trustworthy component is one that won't fail. This differs to the prevailing usage of these terms in the MANET literature in which a trusted node is one in which sufficient trust has been established, while a trustworthy node is one that will behave as expected [16].

This notion of behaviour, and in particular the detection and mitigation of undesirable behaviour, has received much attention in recent years [5], [16], [20]. The goal of this work has been to either prevent undesirable behaviour, or react to it when it occurs. In the main, undesirable behaviour has been categorised as either selfish, malicious or Byzantine.

A node is considered selfish if it endeavors to protect its resources above all else, possibly to the detriment of network connectivity. A node's utility to the network is affected by its willingness to honestly participate in the

The first and third authors were sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and research was accomplished under Agreement Number W911NF-06-3-0001. The second author was sponsored by the Engineering and Physical Sciences Research Council (EPSRC) UK e-Science programme of research (EP/D053269). The views and conclusions contained in this document are those of the author(s) and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

correct functioning of the network's protocols. However, given the cost associated with both the reception and forwarding of packets in MANETs and the limited battery capacity of nodes, conforming to network protocols may be at odds with the best interests of a node. A selfish node's strategy is to attempt to consume the resources of others whilst limiting its own resource expenditure. By contrast, a node is considered malicious if it tries to do harm to other nodes in the network [16]. Malicious behaviour tends to have a specific interpretation and is typically associated with a node preforming a particular attack, as we shall see in Section III. Byzantine behavior occurs when a node deviates arbitrarily from its protocols, by selectively displaying good, selfish or malicious behaviour.

Whilst proposed definitions such as these allow us to neatly categorise node (mis)behaviour, in reality detecting and distinguishing between selfish, malicious and Byzantine nodes is an exceedingly difficult task. Compounding this problem is that nodes that are simply overloaded or misconfigured may be incorrectly labeled as selfish or malicious. As we will see in Section II, the detection of undesirable behaviour is largely dependent on what a node can physically observe and to a lesser extent on what a node has been told indirectly by one of its neighbours. In this paper we will look at trust, and in particular reputation models, that have been proposed in the literature. We examine the operation of a number of protocols and highlight a number of areas for future research.

This paper is laid out as follows. In Section II, we look at the issue of trust in MANETs, with a particular emphasis on how trust is established in routing protocols. In Section III, we examine a number of threats posed to MANETs. In Section IV, we look at what happens when trust needs to be revoked and review a number of proposals for handling revocation in MANETs. Finally, we conclude with Section V.

II. Trust and Reputation

Trust models are an attempt to formalise trust definitions [1] and are often tied to the establishment of a Public Key Infrastructure (PKI) in MANETs [75]. For example, Hubaux, Buttyán, and Capkun bootstrap trust relationships from Pretty Good Privacy (PGP) like certificates in [35]. In [2], Abdul-Rahman also propose a trust management and recommendation protocol built upon PGP. In [37], [38], [41], Jøsang describes methods for computing authenticity based on certificates, key bindings, and on trust relationships in which an opinion and evidence driven models are used to represent trust. Kagal et al. give a security approach based on trust for pervasive computing [42] in which a security agent (fixed device) in each domain is responsible for trust management, authentication and authorization. A European project, SECURE [28], [66], presents trust and risk frameworks for enabling secure collaboration between ubiquitous computer systems. Establishing trust by physical contact between devices is presented in [67] and extended (to include the use of location-limited channels) in [7].

In addition to this work, the modelling of trust at the network layer has received much attention. Various authors have proposed methods for nodes to establish trust in one another. In this section we provide an overview of these proposals, many of which are designed to tackle the problem of packet forwarding selfishness using preventative and/or reactive measures. The solutions described below can be classed as follows:

- Routing protocol mechanisms,
- Currency systems, and
- Reputation systems.

A. Trust in Routing Protocols

Awerbuch et al. [6] requires that destination nodes receiving data packets respond to the originator node with a signed acknowledgement. This acknowledgement consists of the packet's unique identifier, concatenated with the destination node's address. If the number of unacknowledged packets exceeds a threshold, then a fault detection protocol is used, similar to the Secure Traceroute protocol proposed by Padmanabhan and Simon [59]. In any subsequent data packets the originator sends on the same route, it includes a 'probe list'; this contains the addresses of the intermediate nodes from which the originator wants an acknowledgement, with the destination node's address as the last entry. The list includes a HMAC which is recursively produced, each round using a secret key shared between the originator node and the intermediate node being 'probed' (this is a technique also known as 'onion encryption' [63]).

When a probed intermediate node receives the probe list, it decrypts a layer of the onion encryption and verifies the HMAC before forwarding the packet, so that the next intermediate node in the probe list can verify that it belongs to the list.

After forwarding a probe list, an intermediate node waits for an acknowledgement from the next node on the probe list. If one is not received with a specific timeout interval, then the node must initiate an acknowledgement chain by creating an acknowledgement. The timeouts are calculated in such a way that the last probed node which successfully receives the packet will always initiate the acknowledgment chain. Thus, when the route is working, this will be the destination node. This acknowledgement chain is forwarded towards the originator node.

The originator node decrypts each layer of the final acknowledgement packet, verifying the HMAC within each layer to confirm that the corresponding intermediate node received the packet. The originator can use this probe protocol to perform a binary search by adding the intermediate node in the middle of the route to the probe list of successive data packets, halving the route being searched after every iteration. A faulty link is discovered when an expected acknowledgement is not received from an intermediate node at position i in the probe list, but an acknowledgement is received from the intermediate node at position $i-1$.

B. Currency Systems

Currency systems have been proposed for use in ad hoc networks to create a form of 'economy' in which either, the nodes exchange tokens for forwarding each others' packets, or the node owners themselves receive some form of gain. These systems require either a trusted security module in each node, or some kind of central authority. Both approaches are now discussed.

1) Using Nuglets: The most widely-cited currency-based scheme is due to Buttyán and Hubaux [14]. They introduce a currency called *nuggets* or *nuglets*, for which they propose two models: the Packet Purse Model and the Packet Trade Model. They have also suggested [15] an extension to the Packet Purse Model.

In the Packet Purse Model, an originator node loads its packet with a 'purse' of nuglets, so that intermediate nodes can boost their own stores of nuglets by forwarding the packet and taking nuglets in return for providing service. The Packet Trade Model creates a trading route, where intermediate nodes 'buy' packets to 'sell' to the next hop, except the next-hop of the originator node who receives the packet at no cost. The implementation of both models is similar, and thus we only discuss the Packet Purse Model.

The exchange of nuglets relies on a tamper-resistant security subsystem being present in every node, which runs the routing protocol and acts independently of the node to which it is connected. Whenever a node needs to send a packet, a request is sent to the security module which will respond with a 'purse', i.e. a data structure in the packet header, containing n of the nuglets from its supply, and data to identify the 'purse'.

Each intermediate node will forward the packet if the number, p, of nuglets left in the 'purse' is enough to satisfy the node's cost of forwarding, c, i.e. if $p \geq c$. If this is the case, then the intermediate node's security module will compute a header for its owner to add to the packet. This header consists of the old 'purse' containing p nuglets, and a new 'purse' containing $p-c$ nuglets. If there are enough nuglets in the new 'purse' to satisfy the next-hop node, then it will repeat the same process and forward the packet. The old 'purse' is included to act as a token for passive observation for the previous hop — if an acknowledgement is received then the security module will increase the nuglet counter for its owner. This gives an incentive for a node to forward a packet after it has been processed by its security module.

The security module also maintains and increments a 'fine' on a neighbour if it does not receive an acknowledgement. The 'fine' is attached to the 'purse' of the next packet which is sent to the neighbour. When the neighbour's security module receives the packet, it will decrease its owner's nuglet counter. This provides an additional incentive for a node to forward a packet, to ensure that the previous hop receives the passive acknowledgment.

In an extension to the Packet Purse Model (proposed in [15]), nuglets are not sent in a purse. Instead each security module maintains a nuglet counter for each of its owner's neighbours, and the corresponding counter is increased when a forwarded packet is received from a neighbour. A protocol is periodically used by the security modules to update each others' nuglet counters they hold for their owner. This removes the need for the acknowledgement and fining mechanisms in the original model.

2) The Sprite System: The Sprite system [74] makes use of a public key infrastructure to deal with the problem of selfishness in networks which use source route based routing. Each packet is sent with a digital signature computed on a hash of the packet payload, source route and sequence number. Each intermediate node receiving the packet can compute the same hash digest and store this along with the signature, sequence number and source route as a receipt for the packet. The scheme also requires four parameters x, y, z and a, affecting payments made in various circumstances, to be chosen.

Nodes upload receipts to a Credit Clearance Service (CCS), a central authority which is available when the nodes are connected to the Internet. The CCS charges the sender x for every intermediate node for which the next-hop reports receipt of the packet, and y for every intermediate node which reports a receipt but for which the next-hop did not. These charges are transferred to the accounts of the various nodes. To prevent collusion in which the sender saves credit by providing external payments to colluding intermediate nodes, the CCS charges the sender an extra z if the destination does not report receipt of the message.

3) Other Proposals: Anderegg and Eidenbenz [3] propose a more accurate currency system in which nodes advertise the cost of forwarding. These costs are revealed during source route based route discovery, and nodes can select a lowest-cost route to send their packets. However, the issues surrounding currency systems, as discussed above, are still present.

The advertisement of node-specific forwarding costs is also used by a scheme proposed by Raghavan and Snoeren [62]. The key difference is that currency is not used to determine whether a packet is forwarded or dropped, but instead is used to gain priority forwarding. If a packet meets the cost advertised by an intermediate node, then the node can forward 'paid' packets prior to 'unpaid' packets. Passive observation is used to ensure that the next-hop adheres to the protocol. Unlike the previously discussed systems, this scheme does not place a node at the network boundary at a major disadvantage; that is, even if a node runs out of currency, it can still get its packets forwarded; instead such nodes will simply not be able to get priority service. Care must be taken to ensure that priority forwarding does not dominate normal forwarding to such an extent that it becomes a source of a denial of service. Unfortunately, no details are given about how the currency system is managed, so the related issues still remain.

C. Reputation Systems

Reputation systems have been proposed for use in ad hoc networks to address some of the threats arising from misbehaving network nodes. These mechanisms, explored in more detail in this section, are potentially of particular value in addressing the threats arising from selfish nodes. In the context of an ad hoc network, these mechanisms seek to dynamically assess the trustworthiness of neighbouring network nodes, with a view to excluding untrustworthy nodes. Although reputation can be seen as a particular trust metric there have been attempts to draw a distinction between the two. In [11], Buchegger et al. define reputation as representing how well a node behaves whilst a trust rating represents how honest a node is. Reputations are based on either direct or indirect evidence where direct implies first-hand interaction with a node while indirect is based on second-hand information coming from other nodes in the form of recommendations. A lot of recent research has been done to evaluate and manage reputation in mobile ad hoc networks, a survey of which can be found in [5].

The use of reputation systems in many different areas of information technology is increasing, not least because of their widely publicised use in online auctions and product reviews, see, for example, eBay and Amazon [65]. Mui et al. [57] give many examples of how reputation systems are used.

Reputation systems are used to decide who to trust, and to encourage trustworthy behaviour. Resnick and Zeckhauser [64] identify three goals for reputation systems:

1) To provide information to distinguish between a trustworthy principal and an untrustworthy principal,
2) To encourage principals to act in a trustworthy manner, and
3) To discourage untrustworthy principals from participating in the service which the reputation mechanism is present to protect.

Reputation systems can be managed either centrally or in a distributed manner [36]. As was the case for the discussion of currency systems in section II-B), we concentrate here on distributed reputation systems, which suit the properties of a stub ad hoc network. Reputation systems rely on principals monitoring sequences of transactions with other principals, and on communications between principals that are willing to take part in the reputation system. Each principal maintains a reputation value for some subset of the other principals in the system — these values can be shared between principals or they may be unique for each participant. The precise meaning of the reputation value, how it is calculated and updated, and how it is communicated between parties, are all system-dependent. However, it is generally true that this value is intended in some way to measure the trustworthiness of the principal, at least for the purposes of the system concerned.

One of the first proposals for a reputation-based scheme designed to address the problem of selfish nodes in ad hoc networks is due to Marti et al. [48]. While not presented as a reputation system, their solution incorporates the mechanisms used in such a system. Two widely cited reputation mechanisms for ad hoc network routing are the Cooperation Of Nodes: Fairness In Dynamic Ad-hoc NeTworks (CONFIDANT) protocol [9], [10], [11], [12], [13], and the Collaborative Reputation Mechanism (CORE) protocol [52], [53], [54], [55], which work in a similar way. The rest of this section uses these three schemes, along with some other less cited proposals, as examples in order to study the use of reputation systems to protect ad hoc networks.

1) The Watchdog and Pathrater Mechanisms: Marti et al. [48] were amongst the first to highlight the problem of non-forwarding behaviour in ad hoc networks. They proposed two mechanisms to deal with the issue of selfishness in an ad hoc network, namely the Watchdog and the Pathrater.

Like the DSR routing protocol, the Watchdog mechanism makes use of passive observations. Therefore, if a node maintains a buffer containing packets it has sent to a neighbouring node, then, using passive acknowledgements, the node can determine whether this neighbour has forwarded the packets. If a packet in the buffer remains unacknowledged for a certain period of time, i.e. it has not been forwarded, then a failure count for that neighbour is incremented. If the failure count exceeds a threshold, then the node sends a notification to the source of the packet identifying the selfish node.

The Pathrater mechanism operates only with source routing based protocols such as DSR, and is essentially a reputation system. A node assigns a null rating of 0.5 for each node connected to the network, derived from the source routes accumulated through route discovery. The ratings of nodes on actively used source routes are increased by 0.01 every 200 ms, up to a maximum of 0.8. When a link break occurs, the node upstream of the break can send a route error message back to the source. On receipt of a route error message, the ratings of the nodes downstream of the route error originator are decreased by 0.05, unless the rating is already 0 or less, in which case the rating is left unchanged[1]. If a notification of selfishness is received about a node, then the rating of that node is assigned a value of –100. All negative ratings are either increased slowly, or reset to zero after a specific period of time, in order to allow a selfish node to recover. When a node has multiple paths to the same destination, it can calculate the mean average of the ratings of each path, in order to determine which path is most likely to offer successful delivery of traffic.

2) An Overview of CORE: Michiardi and Molva [52], [53], [54], [55] define an ad hoc network as a community which uses a common resource, in which each member must contribute. Any member not contributing will find their reputation worsening until they are gradually excluded from the operation of the network because of their bad reputation.

CORE defines two types of reputation, namely subjective and indirect, both of which are calculated for each function being observed. A node maintains the reputation of each neighbour node for each of a range of functional behaviours. The two types of reputation values are computed as follows:

- **Subjective Reputation** is based on local observations. If an observed behaviour matches the expected behaviour, then the observation will be deemed positive; otherwise it is deemed negative. When updating a reputation value, greater weight is given to past behaviour than current behaviour; placing more weight on past observations prevents subjective reputation being influenced by sporadic behaviour.
  To be able to perform observations reliably is of extreme importance to the CORE scheme, and the authors have suggested the Watchdog mechanism based on promiscuous observation (see section II-C1). The expected result of the current operation is stored in a buffer until a matching observation is made. While the expected result is still present in the buffer, the reputation for the observed function is gradually decreased.
- **Indirect Reputation** information from other nodes can also be accepted. Only positive reputation values are used, to eliminate an attack in which a malicious node transmits negative reputation information to cause a denial of service.

For each function, the subjective and indirect reputations are combined to give a composite functional reputation, weighted in favour of the subjective reputation. All composite functional reputation values are combined to form an overall reputation value for a community member, weighted according to the importance of the function. For example, more weight will be given to data packet forwarding if it is deemed to be more important than forwarding control packets. A reputation value of 0 represents a neutral view, and this is assigned when there are not enough observations to make an accurate assessment of a node's reputation. A node should refuse to provide any network services to a node for which it has calculated a negative overall reputation.

As discussed above, each node maintains a reputation table for each function being monitored. A function is seen as having a request phase and a reply phase. The reputation table contains the reputations of other nodes; each entry consists of a unique ID for the node, recent subjective observations, recent indirect observations and the composite reputation for the given function. There are three ways in which a reputation table entry can be updated:

1) In the first case, a node A requests a service from node B, but node B refuses to perform the service. As a result, node A decreases its perceived reputation of node B by adjusting B's subjective reputation accordingly.

[1] Unless a node on that route is being successfully used in another source route.

2) In the second case, some form of reply is sent containing a list of entities which have successfully cooperated to provide the service, all of which must have positive reputations. For example, for the function of forwarding packet in the DSR protocol, the authors suggest using an end-to-end acknowledgement which includes the reversed source route taken from the corresponding packet.
3) The final case is when reputations are gradually decreased if there is no interaction with the observed node.

A node can refuse to cooperate when it is asked to perform a service by a node with a negative reputation.

3) An Overview of CONFIDANT: There are several versions of the CONFIDANT protocol [9], [10], [11], [12], [13], and we summarise the most recent version here. A study of older versions is given in [72].

As in the previous two systems, nodes using the CONFIDANT scheme rely on passive observation of all packets sent within a one-hop neighbourhood to detect non-forwarding behaviour. Each neighbour is initially allocated a null reputation value, and Bayesian theory is used to update the reputation values based on the node's own observations. More emphasis is placed on recent behaviour, and past behaviour is weighted less every time the reputation is recalculated. If the reputation value for a node drops below a threshold value, then that node is deemed to be misbehaving. A node also periodically 'fades' the reputation values it stores, so that reputation values always tend back towards the threshold value used for misbehaving nodes.

A node periodically broadcasts the observations it has made for each of its neighbours[2]. A node will reject a neighbour's observations of another node if the neighbour's observations differ from the node's own perception by more than a deviation value d. Of those received observations which do not differ by more than d, the node incorporates them into its own corresponding calculation of the reputation value, weighted so that they do not significantly alter the node's original calculation. In this case, the observations are weighted based on a *trust value* for the source of the message.

Trust values are maintained in a trust table. These values indicate how much the node can trust its neighbours to send accurate reputation information. If received reputation information corresponds to the node's own view, then the node can increase the trust value of the source of the message. Otherwise, the trust value is decreased.

CONFIDANT is designed to be used in conjunction with the DSR protocol. Nodes can rank routes according to the reputations of the intermediate nodes. All paths which contain a misbehaving node are deleted. Route requests and packets received from misbehaving nodes are dropped. It is unclear if all packets received from a misbehaving neighbour are dropped, or only those which are originated by a misbehaving node. A node marked as misbehaving must wait until its reputation 'fades' back above the threshold before its packets will be forwarded once more.

4) An Overview of SORI: The Secure and Objective Reputation-based Incentive (SORI) scheme [29], like CONFIDANT, restricts reputation reporting to the local 1-hop neighbourhood. Unlike the previous schemes, SORI concentrates solely on tackling non-forwarding behaviour.

Each node a calculates the reputation of each neighbour b as the ratio of the number of packets passively acknowledged by b to the number of packets sent to b for forwarding. Hence reputation values are always in the range [0,1]. The number of packets sent to the neighbour is also used as a estimation of a node's confidence in the reputation value it has calculated, i.e. the more packets sent, the more accurate the ratio calculation will be as a representation of reputation.

Each node periodically updates the reputation and confidence values of each neighbour, broadcasting the pair if a significant change has occurred. Thus, each node a calculates an overall reputation value known as the *overall evaluation record* $OER_a(b)$, for each neighbour b, based on both its own observations and the observations received from its neighbours. As in the other schemes, the observations from neighbours are weighted: in SORI the neighbour's reputation is used to weight the observations received from that neighbour.

If a neighbour's overall reputation value drops below a threshold, then the node reacts by probabilistically dropping packets that the neighbour sends for forwarding. The probability p that node a drops a packet received from b is:

$$p = \begin{cases} 1 - OER_a(b) - \delta & \text{if } 1 - OER_a(b) > \delta \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where δ is a system parameter used to introduce a margin of error to represent packets dropped because of reasons other than selfishness.

Of all the reputation mechanisms studied, SORI is the only one to propose additional security mechanisms to provide origin authentication of reputation messages.

[2] It is not clear whether a node broadcasts direct observations of only its current neighbours, or of all neighbours encountered during the node's lifetime.

This is achieved using the TESLA broadcast authentication mechanism, as used in the Ariadne protocol [31].

5) Using a Reliability Index: Conti, Gregori and Maselli [19] suggest a scheme in which each node maintains a reliability index for each neighbour. When a node needs to choose a route, it can use the reliability index in one of two ways. The first approach is simply to use the next-hop with the highest reliability index. The second approach is to use the totality of the reliability indexes of all possible next-hops to select the next-hop for a particular packet in a probabilistic way. The latter enables a better distribution of traffic over all available next hops.

To maintain the reliability index, end-to-end acknowledgements are required for every packet sent. The reliability index for a neighbour is decreased if an end-to-end acknowledgement is not received within a specific time. Otherwise, the index is increased according to a smoothing factor, which determines what percentage of the new reliability index should be computed from the previous index, i.e. the smoothing factor determines the degree of influence of past behaviour. How the smoothing factor is chosen is not specified. It is not stated in [19], but it is presumably the case that a node maintains a table in order to match the received acknowledgements with expected behaviour. Encryption is suggested for communication between a pair of nodes to prevent masquerade attacks.

To encourage cooperation, each node uses the reliability index to probabilistically determine whether it should forward a packet on behalf of a neighbour. A low reliability index for a particular neighbour will result in a lower probability that packets received from that neighbour are forwarded. It is not clear whether the reliability index applies to all packets received from a neighbour, or only those originated by the neighbour. This could have a significant effect on the effectiveness of this solution, and is an issue which needs resolution if this scheme is to deployed.

6) Other Reputation Systems: The ‘Friend or Foe’ reputation system proposed by Miranda and Rodrigues [56] differs from the previously described schemes in that each node periodically floods a reputation message throughout the network. This message contains:

1) The set of nodes the originator is willing to provide service for, i.e. *friends*,
2) The set of nodes the originator is not willing to provide service for, i.e. *foes*, and
3) The set of nodes the originator knows has reported it as a foe, i.e. *selfish*.

This allows the network to correlate all reputation information, so that nodes can send packets via routes involving more friends.

Paul and Westhoff [61] propose mechanisms for securing the DSR protocol, one of which is a context inference scheme for deciding if reputation messages are truthful. Only if at least three messages have been received reporting a particular misbehaviour by a node will the reports be deemed to be an indication of true events. The messages must each contain a copy of the same packet which caused the report, and matching details of the misbehaviour. If only a single report is received, then this is deemed as misbehaviour by the originator of that report.

D. Applicability to general distributed scenarios

Each of the three classes of solution can be applied to general distributed systems. In all three cases there is a distinction to be made between ‘security’ and ‘trust’, which are prerequisites of each other. Each class of solution requires security to ensure that the created trust, or mistrust, can be itself be considered reliable; for example, non-repudiation and origin authentication are two services that would be necessary. In turn, these services require a security architecture that allows for key management, which in itself requires notions of ‘trust’.

An alternative view is taken by Jøsang et al. [40], who makes the distinction between the user and the provider. Conventional security, including authorisation and authentication, focuses on the provider and protection of resources. On the other hand, trust management is used by users when selecting between providers. In many cases, an imbalance exists in which the volume of research is weighted in favour of conventional security; the three classes of solution presented above are an attempt to address this imbalance for MANETs.

The ideas from the routing protocol mechanism (section II-A) can be developed to detect problems in a linear sequential workflow that are distributed amongst different entities (perhaps making use of services that are exclusive to a particular entity). Conventional auditing may be too resource intensive in some scenarios, and reactive probing may present a more efficient solution.

Currency systems rely on creating an economy, by introducing a virtual finite resource for trading. While this encourages trust to develop, in much the same way as real world economics nurture trust, it also creates a marketplace. This has benefits and disadvantages — the benefits of competition improve quality-of-service and the ‘value’ of trust, since any loss of trust means loss of potential earnings.

However, there are several issues that will determine whether or not a currency based trust system can be considered. The method of determining whether a service has been correctly delivered must be robust. Additionally, a trusted authority is required either as an external third party or internal security module installed in all participating entities. Physical location and communication links could also be important; in all the currency systems reported above, nodes on the boundary of the ad hoc network will receive fewer packets to forward — certain entities could find themselves in an disadvantageous position to provide service, to receive currency, which could be a result of natural network convergence or active attacks on the system, or underlying communications networks. Finally, there is the problem of only providing enough service in order to make ends meet, and potentially encourages willful discrimination. This is a problem that is also common to reputation systems.

Of the reputation systems discussed in section II-C, CORE introduces a general model that can be applied to a variety of distributed systems. It is interesting to note that the systems focus on local reputation determined from local observations, rather than on global distributed values. This is mainly because two significant factors, highlighted by Yau and Mitchell [71]. Firstly, determining whether service has been delivered is much easier and resource efficient in the local neighbourhood. Secondly, unless the reputation systems themselves are secured, then they introduce additional vulnerabilities that can be exploited by malicious attackers. Global reputation calculations seem to be reliable only when the system which is being used it large, and a policing authority exists.

A final fundamental issue with all reputation systems is that they require time to establish and evolve trust relationships. In many situations, this requirement cannot be met: an ad hoc network, as a whole, can be temporary, as can the relationship between a pair of ad hoc network nodes. Furthermore, ephemeral relationships can also affect the ability for forgiveness, i.e. to repair trust. This is an area of research [46], [70] that requires more attention within MANETs.

III. Threats to Ad hoc Networks

The adversarial nature of environments in which MANETs may be expected to operate, place nodes at risk from a number of threats.

In a *black-hole* attack on routing protocols, an attacker attempts to interpose themselves in as many routing paths as possible by distributing forged routing information (claiming short distance to other nodes). This attracts traffic to the attack node who then drops packets in an attempt to disrupt network performance [34]. In a special case of a black-hole attack, an attacker may create a *gray-hole* in which it selectively drops some packets but not others, for example, by forwarding routing packets but not data packets.

In a *worm-hole* attack [32], an attacker records packets at one location in the network and (selectively) tunnels them to another location. For example, a wormhole attack would involve two distant nodes colluding to understate their (hop) distance from each other by relaying packets along an out-of-band channel. Like a black-hole this will establish false routes and poses a risk to data aggregation, clustering protocols and location-based services.

A rushing attack [33] occurs against reactive (on-demand) routing protocols in which route request messages are propagated on a first-come-first-served basis. If route request messages that reach an attacker's neighbours are those of the attacker, then any route discovered by an initiator will include the attacker.

A Sybil attack is one in which a malicious party claims multiple identities, all of which are controlled by the same entity [24], [58]. The ability to control an arbitrary fraction of the nodes in a network, allows a malicious adversary to effectively out-vote any honest nodes in collaborative tasks. It has been argued that in any peer-to-peer system without a centralised point of trust, such attacks on identity are endemic and can never be effectively combatted [24]. Node replication attacks [60] are the dual of the Sybil attack. By capturing a node an adversary can replicate that node's identity and distribute it throughout the network. The result of this attack is that several nodes will share the same identity which can result in routing protocols being led astray.

A further attack on ad hoc networks has been examined in [68] and [45]. In a resource exhaustion (sleep depravation) attack an adversary attempts to consume a nodes (limited) battery resource by continually sending messages to that node, forcing it to do work and preventing it from cycling into a power saving 'sleep' mode. Continued over a prolonged period, this attack can effectively remove a node from the network.

A overview of defences to the attacks outlined above can be found in [16]. However, like the various protocols outlined in Section II-A, each additional defence mechanism introduced to the network comes at a cost to network utility. Where possible, due care needs to preformed in deciding what attacks are practicable with such analysis possibly feeding into a larger risk calculation.

IV. Punishment

Revocation is perhaps the most serious outcome of any trust decision. Typically the decision to revoke it based upon the premise that malicious behaviour is detectable and malicious nodes are identifiable. In Section II, we saw how reputation mechanisms have been used to locally revoke a node by ignoring that nodes requests. In this section we briefly look at proposed mechanisms that aim to globally remove a node from the network. In the MANET literature, revocation is typically tied to invalidating a node's key material by informing other nodes in the network that they should no longer communicate with the owner of this key.

In [26], Eschenauer and Gligor propose a scheme in which a centralised base-station determines which keys are tied to a compromised node and instructs all nodes holding those keys to delete them. In [17], Chan et al. propose a distributed revocation mechanism where nodes sharing a pre-assigned pairwise key can vote to remove a misbehaving node. The use of certificate revocation list (CRL) as part of a larger PKI have been studied in [25], [27]. However, a dearth of research exists on issues concerning revocation and re-issuance/replacement of keys/certificates. Typically the issue of revocation is either not handled in the MANET literature or simply focuses on a particular revocation mechanism. The actual decision that goes into determining if a node should be revoked or not is often left to some undefined and largely unexamined policy "layer". In this respect, terminating trust in a key through global revocation may be somewhat problematic in ad hoc networks.

A. *Is Suicide the Answer?*

The advent of practical identity-based public key cryptographic schemes has resulted in a number of proposals for using Identity-based Public Key Cryptography (ID-PKC) as the underlying key management infrastructure for Mobile Ad-hoc Networks (MANETs) [30], [49], [73]. However, while there isn't the explicate need for certificate revocation found in traditional PKIs, ensuring validity periods for identifiers yields comparable implementation problems. Key revocation in ID-PKC requires revocation of a node's identity, and, if this identity is one that is inexpedient to change, the problem is exacerbated. Typically this issue is addressed by the reissuance of keys as originally suggested by Boneh and Franklin [8]. In this respect, key renewal is an important consideration in ID-PKC-based ad hoc networks. The frequency of renewal is an important parameter, as the higher the turnover, the less impact key compromise will have on the network and the more key revocation may become redundant [44]. Here expiry dates are included in deriving a node's identifer, which avoids the need for a revocation mechanism.

However, there may still be a requirement for revoking a key prior to the date specified in an identifer. Hoeper and Gong [30] introduce two ID-PKC schemes for MANET environments that attempt to address the key revocation/key renewal problem. The authors identify four parameters for revocation. Firstly, nodes should be able to revoke their own key, an approach they refer to as *harakiri*. Secondly, nodes should be able to revoke keys of suspicious or compromised nodes using an quorum-based accusation scheme. Thirdly, nodes should broadcast revocation information and finally, new nodes should receive a list of all previously revoked keys. Other schemes that rely on quorum-based voting for revocation include [22], [43]. However, many such voting schemes are vulnerable to Sybil attacks [23].

Clulow and Moore [18] and Anderson et al. [69] both introduce suicide schemes in which a node detecting malicious behaviour can instigate a suicide-bombing on the malicious node. Here instead of relying on a quorum-based decision, nodes can act unilaterally. A node commits suicide by broadcasting an instruction to remove the bad node and itself from the network. Such an approach has been shown to perform favourably in comparison to voting based protocols, however, this approach is based on the assumption that nodes value the network more than their own utility.

Current methods of global revocation pose a number of problems. Revocation introduces high communication overhead as revocation information is typically flooded throughout the network. Additionally, it is unclear under what circumstances a node should trust a message (from another node, or group of nodes) stating a node they have neither seen nor interacted with should be revoked? Finally, many of the mechanisms used to formulate a revocation message are vulnerable to abuse. In a threshold voting scheme, gangs of malicious nodes (whose number is greater than the threshold) can wander the network revoking nodes at will. In Moore et al's. suicide scheme [69], as all nodes are created equal, "high-value" nodes may be removed by "low" value nodes unless one factors node valuations into the revocation decision. This may be difficult to do as node valuations may change dynamically in response to changes in topology.

V. Conclusions

A node joining a MANET must transition from potentially having no trust relationships to a situation in which it can obtain service. Typically, this begins with a leap of faith on the part of the joining node by offering some function in return for connectivity. Such reciprocity allows nodes to iteratively build trust with their neighbours. As we saw in Section II, trust establishment and maintenance mechanisms are relatively well understood and have been the focus of much work in recent years. However, less well studied aspects of trust, such as distrust, mistrust and untrust [47] have only received cursory examination.

This has been particularly true in reference to revocation. In general, instigating a revocation procedure comes as a result of detecting a destructive action in the network. Reacting to this detection, the goal of the detector is to excommunicate the node responsible for committing the (negative) action. However, detection mechanisms may rarely yield non-repudiable evidence, as signing every message may be impractical. Additionally, distinguishing between a heavily loaded node forced to drop packets and a node which is actively being malicious is a non-trivial task and may require a large energy expenditure on the part of (multiple) nodes. Furthermore, it is important that nuisances of (mis)behaviour are taken into consideration when designing trust and reputation mechanisms. For example, selfishness is a rational response to attacks that aim to deplete an exhaustable resource, such behaviour may actually be desirable in protecting the resources of important nodes in the network.

The introduction of new mechanism to cope with threats typically introduces new attack vectors. Introducing a mechanism to cope with malicious behavior (assuming it can be defined and detected in the first place) introduces two problems. Firstly, detection is expensive and may rely on knowledge of MANET topology (which may be prone to frequent changes) as well as potentially introducing an unacceptable overhead. Secondly, as we saw in Section IV, it may also introduce new attack vectors. A MANET needs to be examine (either proactively, reactively or retroactively) as to whether to deploy a particular detection mechanism, possibly based in some larger risk calculation.

Finally, there has been much talk about the need for establishing key infrastructures that can adapt to the introduction of new coalition partners; however, the same has not been true of trust mechanisms. It may be unreasonable to assume that every node will share common trust/risk aggregation rules or that analysis of common events will result in the same conclusions being drawn. In this respect, trust mechanisms need to be designed so that they account for variance, and allow for multiple competing interpretations of an event to be easily composed into a single assessment.

References

[1] A. Abdul-Rahman and S. Hailes. A distributed trust model. In *NSPW '97: Proceedings of the 1997 workshop on New security paradigms*, pages 48–60, New York, NY, USA, 1997. ACM Press.

[2] Al. Abdul-Rahman and S. Hailes. Supporting trust in virtual communities. In *HICSS '00: Proceedings of the 33rd Hawaii International Conference on System Sciences-Volume 6*, page 6007, Washington, DC, USA, 2000. IEEE Computer Society.

[3] L. Anderegg and S. Eidenbenz. Ad hoc-VCG: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In Z. J. Haas, S. R. Das, and R. Jain, editors, *Proceedings of the 9th annual international conference on Mobile computing and networking, MobiCom '03, San Diego, CA, US, September 14-19, 2003*, pages 245–259. ACM Press, Sep 2003.

[4] R.J. Anderson. *Security Engineering: A Guide to Building Dependable Distributed Systems*. John Wiley & Sons, Inc., New York, NY, USA, 2001.

[5] G. Athanasiou, L. Tassiulas, and G. S. Yovanof. Overcoming misbehaviour in mobile ad hoc networks: An overview. *Crossroads The ACM Student Magazine*, (114):23–30, 2005.

[6] B. Awerbuch, D. Holmer, C. Nita-Rotaru, and H. Rubens. An on-demand secure routing protocol resilient to byzantine failures. In D. Maughan and N. Vaidya, editors, *Proceedings of the ACM Workshop on Wireless Security, Atlanta, Georgia, USA, September 28, 2002*, pages 21–30. ACM Press, 2002.

[7] D. Balfanz, D.K. Smetters, P. Stewart, and H.C. Wong. Talking to strangers: Authentication in ad-hoc wireless networks. In *Proceedings of Network and Distributed System Security Symposium 2002 (NDSS'02)*, San Diego, CA, February 2002.

[8] D. Boneh and M.K. Franklin. Identity-Based Encryption from the Weil Pairing. In *CRYPTO '01: Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology*, pages 213–229, London, UK, 2001. Springer-Verlag.

[9] S. Buchegger and J.-Y. Le Boudec. Nodes bearing grudges: Towards routing security, fairness, and robustness in mobile ad hoc networks. In *Proceedings of the 10th Euromicro Workshop on Parallel, Distributed and Network-based Processing PDP, Las Palmas de Gran Canaria, Canary Island, Spain, January 9-11, 2002*, pages 403–410. Euromicro, Jan 2002.

[10] S. Buchegger and J.-Y. Le Boudec. Performance analysis of the CONFIDANT protocol (cooperation of nodes: Fairness in dynamic ad-hoc networks). In J. Hubaux, J. J. Garcia-Luna-Aceves, and D. Johnson, editors, *Proceedings of the Third ACM International Symposium on Mobile Ad Hoc Networking and Computing, Lausanne, Switzerland, June 9-11, 2002*, pages 226–236. ACM Press, June 2002.

[11] S. Buchegger and J.-Y. Le Boudec. A robust reputation system for P2P and mobile ad hoc networks. In *Proceedings of the Second Workshop on the Economics of Peer-to-Peer Systems, P2PEcon 2004, Cambridge, MA, US, June 4-5, 2004*, pages 403–410. Harvard University Press, Jun 2004.

[12] S. Buchegger and J.-Y. Le Boudec. Self-policing mobile ad hoc networks. In Mohammad Ilyas and Imad Mahgoub, editors, *Mobile Computing Handbook*, pages 435–456. CRC Press, 2004.

[13] S. Buchegger and J.-Y. Le Boudec. Self-policing mobile ad hoc networks by reputation systems. *Communications Magazine, IEEE*, 43(7):101–107, Jul 2005.

[14] L. Buttyán and J.-P. Hubaux. Enforcing service availability in mobile ad-hoc wans. In *Proceedings of the 1st ACM international symposium on Mobile ad hoc networking & computing*

MobiHoc '00, Boston, MA, US, August 11, 2000, pages 87–96. ACM Press, Aug 2000.

[15] L. Buttyán and J.-P. Hubaux. Stimulating cooperation in self-organising mobile ad hoc networks. *ACM/Kluwer Mobile Networks and Applications (MONET)*, 8(5):579–592, October 2003.

[16] L. Buttyán and J.-P. Hubaux. *Security and Cooperation in Wireless Networks: Thwarting Malicious and Selfish Behavior in the Age of Ubiquitous Computing*. Cambridge University Press, 2007.

[17] H. Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *SP '03: Proceedings of the 2003 IEEE Symposium on Security and Privacy*, page 197, Washington, DC, USA, 2003. IEEE Computer Society.

[18] J. Clulow and T. Moore. Suicide for the common good: a new strategy for credential revocation in self-organizing systems. *SIGOPS Oper. Syst. Rev.*, 40(3):18–21, 2006.

[19] M. Conti, E. Gregori, and G. Maselli. Towards reliable forwarding in mobile ad hoc networks. In M. Conti, S. Giordano, E. Gregori, and S. Olariu, editors, *Personal Wireless Communications: IFIP-TC6 8th International Conference, PWC 2003, Venice, Italy, September 23-25, 2003*, pages 790–804. Springer-Verlag (LNCS 2775), Sep 2003.

[20] M. Conti, E. Gregori, and G. Maselli. Cooperation issues in mobile ad hoc networks. In *ICDCSW '04: Proceedings of the 24th International Conference on Distributed Computing Systems Workshops - W7: EC (ICDCSW'04)*, pages 803–808, Washington, DC, USA, 2004. IEEE Computer Society.

[21] S. Corson and J. Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations. RFC 2501 (Informational), January 1999.

[22] C. Crpeau and C.R. Davis. A certificate revocation scheme for wireless ad hoc networks. In *SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, pages 54–61, New York, NY, USA, 2003. ACM Press.

[23] John R. Douceur. The Sybil Attack, booktitle = IPTPS '01: Revised Papers from the First International Workshop on Peer-to-Peer Systems. pages 251–260, London, UK, 2002. Springer-Verlag.

[24] J.R. Douceur. The Sybil Attack. In P. Druschel, M. F. Kaashoek, and A. I. T. Rowstron, editors, *Peer-to-Peer systems, First International Workshop, IPTPS 2002*, volume 2429 of *LNCS*, pages 251–256. Springer, 2002.

[25] S. Eichler and B. Muller-Rathgeber. Performance analysis of scalable certificate revocation schemes for ad hoc networks. In *LCN '05: Proceedings of the The IEEE Conference on Local Computer Networks 30th Anniversary*, pages 382–391, Washington, DC, USA, 2005. IEEE Computer Society.

[26] L. Eschenauer and V.D. Gligor. A key-management scheme for distributed sensor networks. In *CCS '02: Proceedings of the 9th ACM conference on Computer and communications security*, pages 41–47, New York, NY, USA, 2002. ACM Press.

[27] H. W. Go, P. Y. Chan, Y. Dong, A. F. Sui, S. M. Yiu, Lucas C. K. Hui, and Victor O. K. Li. Performance evaluation on crl distribution using flooding in mobile ad hoc networks (manets). In *ACM-SE 43: Proceedings of the 43rd annual Southeast regional conference*, pages 75–80, New York, NY, USA, 2005. ACM Press.

[28] E. Gray, P. O'Connell, C. D. Jensen, S. Weber, J. M. Seigneur, , and Y. Chen. Towards a framework for assessing trust-based admission control in collaborative ad hoc applications. Technical Report TCD-CS-2002-66, Trinity College Dublin, 2002.

[29] Q. He, D. Wu, and P. Khosla. SORI: A secure and objective reputation-based incentive scheme for ad-hoc networks. In *Proceedings of the Third IEEE Wireless Communications and Networking Conference, WCNC 04, Atlanta, GA, US, March 21-25, 2004*, volume 2, pages 825–830. IEEE Press, Mar 2004.

[30] K. Hoeper and G. Gong. Bootstrapping Security in Mobile Ad Hoc Networks Using Identity-Based Schemes with Key Revocation. Technical Report CACR 2006-04, Centre for Applied Cryptographic Research (CACR) at the University of Waterloo, Canada, 2006.

[31] Y.-C. Hu, A. Perrig, and D. B. Johnson. Ariadne: a secure on-demand routing protocol for ad hoc networks. *Wireless Networks, Springer Science*, (11), 2005. 21–38.

[32] Y.-C. Hu, A. Perrig, and D.B. Johnson. Packet leashes: A defense against wormhole attacks in wireless networks. In *Proceedings of IEEE Infocomm 2003*, April 2003.

[33] Y.-C. Hu, A. Perrig, and D.B. Johnson. Rushing attacks and defense in wireless ad hoc network routing protocols. In *WiSe '03: Proceedings of the 2003 ACM workshop on Wireless security*, pages 30–40, New York, NY, USA, 2003. ACM Press.

[34] Y.C Hu and A. Perrig. A survey of secure wireless ad hoc routing. *IEEE Security and Privacy*, 2(3):28–39, 2004.

[35] J.-P. Hubaux, L. Buttyán, and S. Capkun. The quest for security in mobile ad hoc networks. In *MobiHoc '01: Proceedings of the 2nd ACM international symposium on Mobile ad hoc networking & computing*, pages 146–155, New York, NY, USA, 2001. ACM Press.

[36] R. Ismail, C. Boyd, A. Josang, and S. Russell. A security architecure for reputation systems. In K. Bauknecht, A. M. Tjoa, and G. Quirchmayr, editors, *E-Commerce and Web Technologies - 4th International Conference, EC-Web 2003, Prague, Czech Republic, September 2-5, 2003*, pages 176–185. Springer-Verlag (LNCS 2738), Sep 2003.

[37] A. Jøsang. The right type of trust for distributed systems. In *NSPW '96: Proceedings of the 1996 workshop on New security paradigms*, pages 119–131, New York, NY, USA, 1996. ACM Press.

[38] A. Jøsang. An algebra for assessing trust in certification chains. In *In Proceedings of the Network and Distributed Systems Security (NDSS'99)*, 1999.

[39] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decis. Support Syst.*, 43(2):618–644, 2007.

[40] A. Jøsang, C. Keser, and T. Dimitrakos. Can we manage trust? In Peter Herrmann, Valérie Issarny, and Simon Shiu, editors, *Trust Management: Third International Conference, iTrust 2005 Paris, France, May 23-26, 2005*, pages 93–107. Springer-Verlag (LNCS 3477), 2005.

[41] A. Jøsang and S. Knapskog. A metric for trusted systems. In *Proc. 21st National Security Conference*, pages 16–29, 1998.

[42] T. Finin L. Kagal and A. Joshi. Trust-based security in pervasive computing environments. pages 154–157, December 2001.

[43] H. Luo, P. Zerfos, J. Kong, S. Lu, and L. Zhang. Self-Securing Ad Hoc Wireless Networks. In *ISCC '02: Proceedings of the Seventh International Symposium on Computers and Communications (ISCC'02)*, pages 567–574, Washington, DC, USA, 2002. IEEE Computer Society.

[44] J. Luo, J.-P. Hubaux, and P.T. Eugster. DICTATE: DIstributed CerTification Authority with probabilisTic frEshness for Ad Hoc Networks. *IEEE Trans. Dependable Secur. Comput.*, 2(4):311–323, 2005.

[45] N. Vijaykrishnan P. McDaniel M. Kandemir M. Pirretti, S. Zhu and R. Brooks. The sleep deprivation attack in sensor networks: Analysis and methods of defense. In *Conference on Innovations and Commercial Applications of Distributed Sensor Networks*, page to appear, October 2005. Best Paper Award.

[46] S. Marsh and P. Briggs. Examining trust, forgiveness and regret as computational concepts. In Jennifer Goldbeck, editor,

Computing with Social Trust, chapter 2, pages 9–44. Springer London, 2006.

[47] S. Marsh and M.R. Dibben. Trust, untrust, distrust and mistrust - an exploration of the dark(er) side. In Peter Herrmann, Valérie Issarny, and Simon Shiu, editors, *Trust Management: Third International Conference, iTrust 2005 Paris, France, May 23-26, 2005*, pages 17–33. Springer-Verlag (LNCS 3477), 2005.

[48] S. Marti, T. J. Giuli, K. Lai, and M. Baker. Mitigating routing misbehavior in mobile ad hoc networks. In R. Pickholtz, S. Das, R. Caceres, and J. J. Garcia-Luna-Aceves, editors, *Proceedings of the 6th annual ACM/IEEE international conference on Mobile Computing and Networking, MobiCom '00, Boston, Massachusetts, US, August 6-11, 2000*, pages 255–265. ACM Press, 2000.

[49] B.J. Matt. Toward hierarchical identity-based cryptography for tactical networks. In *MILCOM '03: Proceedings of the 2004 Military Communications Conference*. IEEE Computer Society, 2004.

[50] D. Harrison McKnight and Norman L. Chervany. Trust and distrust definitions: One bite at a time. In *Trust in Cyber-societites, Integrating the Human and Artificial Perspectives*, pages 27–54, London, UK, 2000. Springer-Verlag.

[51] D.H. McKnight and N.L. Chervany. The meanings of trust. Working Paper 96-04, University of Minnesota, 1996.

[52] P. Michiardi and R. Molva. CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In B. Jerman-Blazic and T. Klobucar, editors, *Communications and Multimedia Security, IFIP TC6/TC11 Sixth Joint Working Conference on Communications and Multimedia Security, Portoroz, Slovenia, September 26-27, 2002*, volume 228 of *IFIP Conference Proceedings*, pages 107–121. Kluwer Academic, 2002.

[53] P. Michiardi and R. Molva. Game theoretical analysis of security in ad hoc networks. Technical Report RR-02-070, Institute Eurécom, Apr 2002.

[54] P. Michiardi and R. Molva. Making greed work in ad hoc networks. Technical Report RR-02-069, Institute Eurécom, Mar 2002.

[55] P. Michiardi and R. Molva. Prevention of denial of service attacks and selfishness in mobile ad hoc networks. Technical Report RR-02-063, Institute Eurécom, Jan 2002.

[56] H. Miranda and L. Rodrigues. Friends and foes: Preventing selfishness in open mobile ad hoc networks. In *Proceedings of the First International Workshop on Mobile Distributed Computing, MDC '03, Rhode Island, US, May 19, 2003*, pages 440–445. IEEE Press, May 2003.

[57] L. Mui, M. Mohtashemi, and A. Halberstadt. Notions of reputation in multi-agents systems: a review. In M. Gini, T. Ishida, C. Castelfranchi, and W. Johnson, editors, *Proceedings of the first international joint conference on Autonomous agents and multiagent systems, Bologna, Italy, July 15-19, 2002*, pages 280–287. ACM Press, Jul 2002.

[58] J. Newsome, E. Shi, D. Song, and A. Perrig. The sybil attack in sensor networks: analysis & defenses. In *IPSN '04: Proceedings of the third international symposium on Information processing in sensor networks*, pages 259–268, New York, NY, USA, 2004. ACM Press.

[59] V. N. Padmanabhan and D. R. Simon. Secure traceroute to detect faulty or malicious routing. *ACM SIGCOMM Computer Communications Review*, 33(1):77–82, Jan 2003.

[60] B. Parno, A. Perrig, and V. Gligor. Distributed detection of node replication attacks in sensor networks. In *SP '05: Proceedings of the 2005 IEEE Symposium on Security and Privacy*, pages 49–63, Washington, DC, USA, 2005. IEEE Computer Society.

[61] K. Paul and D. Westhoff. Context aware detection of selfish nodes in DSR based ad-hoc networks. In *Proceedings of the IEEE Global Telecommunications Conference, GLOBECOM '02, Taipei, Taiwan, November 17-21, 2002*, volume 1, pages 186–190. IEEE Press, Nov 2002.

[62] B. Raghavan and A. C. Snoeren. Priority forwarding in ad hoc networks with self-interested parties. In *Proceedings of the First Workshop on Economics of Peer-to-Peer Systems, Berkley, CA, June 5-6, 2003*, June 2003.

[63] M. G. Reed, P. F. Syverson, and D. M. Goldschlag. Anonymous connections and onion routing. *IEEE Journal on Selected Areas in Communications*, 16(4):482–494, May 1998.

[64] P. Resnick and R. Zeckhauser. Trust among strangers in internet transactions: Empirical analysis of ebay's reputation system. In M. Baye, editor, *Advances in Applied Microeconomics: The Economics of the Internet and E-Commerce*, volume 11, pages 127–157. Elsevier Science Ltd., November 2002.

[65] P. Resnick, R. Zeckhauser, E. Friedman, and K. Kuwabara. Reputation systems. *Communications of the ACM*, 43(12):45–48, 2000.

[66] B. Shand, N. Dimmock, and J. Bacon. Trust for ubiquitous, transparent collaboration. *Wirel. Netw.*, 10(6):711–721, 2004.

[67] F. Stajano. The resurrecting duckling - what next? In *Revised Papers from the 8th International Workshop on Security Protocols*, pages 204–214, London, UK, 2001. Springer-Verlag.

[68] F. Stajano and R.J. Anderson. The resurrecting duckling: Security issues for ad-hoc wireless networks. In *Proceedings of the 7th International Workshop on Security Protocols*, pages 172–194.

[69] R. Anderson T. Moore, J. Clulow and S. Nagaraja. New strategies for revocation in ad-hoc networks. In *ESAS '07: Proceedings of the Fourth European Workshop on Security and Privacy in Ad Hoc and Sensor Networks*, July 2007.

[70] A. Vasalou and J. Pitt. Reinventing forgiveness: A formal investigation of moral facilitation. In Peter Herrmann, Valérie Issarny, and Simon Shiu, editors, *Trust Management: Third International Conference, iTrust 2005 Paris, France, May 23-26, 2005*, pages 146–160. Springer-Verlag (LNCS 3477), 2005.

[71] P. Yau and C. J. Mitchell. Reputation methods for routing security for mobile ad hoc networks. In *Proceedings of SympoTIC '03, Joint IST Workshop on Mobile Future and Symposium on Trends in Communications, Bratislava, Slovakia, October 26-28, 2003*, pages 130–137. IEEE Press, Oct 2003.

[72] P. Yau and V. Sdralia. Towards the security of routing in ad hoc networks. In C. J. Mitchell, editor, *Security for Mobility*, chapter 10, pages 231–268. IEE Press, 2004.

[73] Y. Zhang, W. Liu, W. Lou, Y. Fang, , and Y. Kwon. AC-PKI: Anonymous and Certificateless Public Key Infrastructure for Mobile Ad Hoc Networks. In *ICC 2005: Proceedings of the International Conference on Communications*. IEEE Computer Society, 2005.

[74] S. Zhong, J. Chen, and Y. R. Yang. Sprite: A simple, cheat-proof, credit-based system for mobile ad-hoc networks. In *Proceedings of the Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003), San Francisco, CA, April 1-3, 2003*, volume 3, pages 1987–1997. IEEE Press, Apr 2003.

[75] L. Zhou and Z.J. Haas. Securing Ad Hoc Networks. *IEEE Network*, 13(6):24–30, 1999.