A Cooperative Security Scheme for Optimized Link State Routing in Mobile Ad-hoc Networks

João P. Vilela João Barros

***Abstract*— We consider the security of control traffic generated by pro-active or table-driven link state protocols in mobile ad-hoc networks. Focusing on the Optimized Link State Routing (OLSR) protocol, we propose a security solution that rewards nodes depending on their cooperation in the exchange of routing information. The proposed scheme, which correlates direct observation of transmissions with path information from successfully delivered packets, is shown to mitigate a relevant set of security issues.**

I. Introduction

The successful operation of a mobile ad-hoc network (MANET), i.e. a self-organizing collection of devices communicating over the wireless medium, requires a minimum amount of cooperation between the nodes in the network. This requirement is particularly striking with respect to the discovery and establishment of routes for reliable and secured data delivery. It is now widely accepted that the specific cooperation mechanisms of MANETs are a source of additional vulnerabilities thus requiring novel security solutions beyond those of the infrastructured/wired paradigm. In the absence of a fixed infrastructure that establishes a line of defense by identifying and isolating non-trusted nodes, it is possible that control messages generated by routing protocols, e.g. neighbor advertisements or link state data, are corrupted or compromised thus jeopardizing the communication within the network.

Among the numerous proposals for routing protocols in MANETs, the Optimized Link State Routing (OLSR) protocol [2], [3] is arguably one that offers promising performance in terms of bandwidth, required overhead and delivered traffic albeit at the cost of a wide range of security challenges, mostly with respect to the required exchange of topology information and the underlying design assumption that all nodes are benign.

The goal of this paper is to provide the OLSR protocol with a security solution that defends the network against malicious nodes by rewarding proper routing behavior and thus assuring effective cooperation between communicating parties. The main novelty of our contribution is the ability to combine two sources of traffic information: (1) the (unreliable) monitoring of whether neighbors relay packets sent to them (as in *watchdog* [4]) and (2) the paths traversed by successfully delivered packets. We argue that the latter increases the network's ability to detect misbehaving nodes. Although our analysis of these security issues, which includes a thorough review of related work, is mainly focused on the OLSR protocol, the described problems and the proposed solutions are equally applicable to other common routing protocols for MANETs.

The rest of the paper is organized as follows. We delay a commented overview of previous work until Section III and present first, in Section II, the basic characteristics of the OLSR protocol, which simplifies the understanding of the following sections. Our main contribution is presented in Section IV, which describes a mechanism capable of solving part of the open security problems. The paper concludes with Section V, which discusses the main features of the proposed solution and offers some directions for future work.

II. Optimized Link State Routing (OLSR)

OLSR can be classified as a proactive link state routing protocol. As a proactive routing protocol, it has the advantage of making the routes immediately available when needed, and as a link state protocol, it uses flooded information about the network topology to calculate the best next-hop for every possible destination in the network.

OLSR offers, in fact, more than a pure link state protocol, because it provides the following features:

- *reduction of the size of control packets* by declaring only a subset of links with its neighbors who are its *multipoint relay selectors* (MPR selectors);
- *minimization of flooding* by using only a set of selected nodes, called *multipoint relays* (MPRs), to diffuse its messages to the network (only the multipoint relays of a node retransmit its broadcast messages).

The use of MPRs for message transmission results in a scoped flooding instead of full node-to-node flooding thus inducing a reduction of the amount of exchanged control traffic. The protocol is particularly suitable for large and dense networks, because the optimization procedure based on multipoint relays works best in those cases.

There are two types of control messages in OLSR:

1) HELLO messages are periodically broadcasted by each node, containing its own address, neighbor lists and the corresponding link state for each of them (uni-directional, bi-directional or MPR). These messages are only exchanged between neighboring nodes but they allow each node to have information about one and two-hop neighbors which is later used in the selection of the MPR set.

2) TC messages are also emitted periodically by nodes in the network. These messages are used for diffusing topological information to the entire network. A TC message contains the list of neighbors who have selected the sender node as a MPR (MPR selector set) and a sequence number associated to the MPR selector set.

The intent of multipoint relays is to minimize the flooding of the network with broadcasted packets by reducing duplicate retransmissions in the same region. Each node selects

The authors are with the Department of Computer Science and the Laboratory of Artificial Intelligence and Computer Science, Faculdade de Ciências da Universidade do Porto, Portugal. Email: {joaovilela, barros}@ncc.up.pt. Work partially funded by the IST project DAIDALOS (Designing Advanced Interfaces for the Delivery and Administration of Location independent Optimised personal Services) [1].

a set of its neighbor nodes that will retransmit its packets. This set of nodes is called the *multipoint relay set* of that node and can change over time, as indicated by the selector nodes in their HELLO messages. The node which chooses the multipoint relay set is a *multipoint relay selector* for each node in the set.

Each node selects its MPR set in a way such that it contains a subset of one-hop neighbors covering all the two-hop neighbors. Additionally, all two hop neighbors must have a bi-directional link to the selected MPR set. The smaller the multipoint relay set, the more efficient the routing protocol.

OLSR determines the routes to all destinations through these nodes, i.e. MPR nodes are selected as intermediate nodes in the path. The scheme is implemented by having each node periodically broadcast traffic control information about the one-hop neighbors that selected it as a multipoint relay (or, equivalently, its multipoint relay selectors). Upon receiving information about the MPR selectors, each node calculates and updates its routes to each known destination. Consequently, the route is a sequence of hops through multipoint relays from the source to the destination. The neighbors of any node which are not in its MPR set receive and process the control traffic but do not retransmit it.

In summary, the OLSR protocol can be specified as follows.

1) Each node periodically broadcasts its HELLO messages;
2) These are received by all one-hop neighbors but are not relayed;
3) HELLO messages provide each node with knowledge about one and two-hop neighbors;
4) Using the information from HELLOs each node performs the selection of their MPR set;
5) The selected MPRs are declared in subsequent HELLO messages;
6) Using this information each node can construct its MPR selector table, with the nodes that selected it as a multipoint relay;
7) A TC message is sent periodically by each node and flooded in the network, declaring its MPR selector set;
8) Using the information of the various TC messages received, each node maintains a topology table which consists of entries with an address of a possible destination (a MPR selector in the TC message), an address of a last-hop node to that destination (the originator of the TC message) and a MPR selector set sequence number;
9) The topology table is then used by the routing table calculation algorithm to calculate the routing table at each node. Details about this procedure may be found in [3] and [2].

For the rest of this paper we assume that the data link layer is able to provide reliable transmission between neighboring nodes, i.e. if a message is sent and no collision occurs, the message is delivered the intended recipient. Naturally, this assumption does not lead to reliable end-to-end communication, because one or more nodes may not behave according to the expectation of the underlying protocols.

In a proactive routing protocol, each node has two tasks to accomplish [5]: (1) correctly generate the routing protocol control traffic (this way giving correct information to the other nodes on the network) and (2) correctly relay the routing protocol traffic on behalf of other nodes (this way allowing for the control traffic to reach every node in the network). Thus, an attack on the routing protocol must result as the corruption of one of this tasks by some node. This can be accomplished by four main actions:

1) *Fabrication of false routing messages:* A node generates regular routing control traffic messages containing false information or omitting information of the current state of the network.
2) *Refuse of control traffic generation/relay:* A node refuses to generate its own routing control traffic or refuses to forward other nodes control traffic (as he is expected).
3) *Modification of routing control traffic:* A node does relay other nodes traffic but modifies it to insert wrong information or omit information from the network.
4) *Replay attacks:* A node listens to routing control traffic transmissions on the network and later on injects possibly wrong and outdated information in the network.

III. Related Work

Recently, several contributions have appeared, aimed at securing OLSR [5], [6], [7], [8]. In the following, we provide an overview of their main features, identifying the underlying assumptions and unsolved issues.

The proposal in [5] is based on a mechanism for key distribution and establishes a line of defense in which (i) nodes are either trusted or untrusted and (ii) trusted nodes are not compromised. It entails a timestamp and a signature associated with each routing control message: the signature is used to identify messages from trusted nodes, and timestamps are used to prevent the replay of older messages. The approach does not contemplate the following issues: (a) trusted nodes may behave incorrectly because of malfunctioning, unconsciously corrupting the routing protocol; (b) nodes in MANETs typically get in and out very often, thus it is hard to separate nodes into trusted and untrusted; (c) the signing mechanism is not detailed (a possibility is [8]).

The contribution in [8], [9] considers the compromise of trusted nodes. It is assumed that a public key infrastructure (PKI) and a timestamp algorithm (e.g. the one in [5]) are in place. An additional message (ADVSIG) is sent in conjunction with routing control traffic. This message contains timestamp and signature information. Each node has a so called *Certiproof* table where information received in ADVSIGs is kept. This information is then reused as a proof of correctness of the link state information in subsequent messages. The procedure ensures that a lone attacker node is not able to send wrong link state information to the network. Its drawbacks are as follows: (a) it does not protect against denial of service or wormhole attacks (two nodes exchange encapsulated packets creating an unexisting connection which results e.g. in false routes) and (b) it imposes a large overhead to the network in terms of additional traffic and computation of signatures.

The focus of [6] is on distributed key management techniques, providing a brief overview of methods to prevent wormhole and message replay attacks. The technique to prevent wormhole attacks is based on a variant of the counting technique [7] in which nodes advertise a set of hashes of the packets received over each of the last k intervals. This way it is possible to check if packet losses cross a certain threshold, in which case a node is assumed to be compromised. Replay attacks are prevented by the use of timestamps.

The security mechanism proposed in [7] uses signature and timestamp schemes to ensure authentication and protection against replay attacks. The signature technique is based on sending a signature with each routing control message as in [8]. Also proposed is a scheme to counter relay attacks based on the geographical position of nodes and a scheme that deals with compromised nodes based on *network flow conservation*, where misbehavior in traffic relaying is detected based upon the number of packets sent and received by each node. The drawbacks of this proposal are as follows: (a) the weak assumption that forwarding the correct number of packets by a node proves that the packets were sent properly; and (b) a centralized security authority that manages misbehavior detection and remedy is difficult to implement in a MANET.

In [10] a fully distributed certificate authority (CA) based on threshold cryptography is described. The CA is distributed in the way that a node requests a certificate from any coalition of k nodes (shareholders) of the network. Upon the certificate request, each of the shareholders determines if he wants to serve the request based on whether the requesting node is well behaving. Upon receiving k "partial certificates" they are manipulated to generate a valid certificate as if it was signed by a regular CA. A monitoring system used to determine behavior of network nodes is not incorporated in the proposal.

With respect to the cooperation between nodes, in [4] a *watchdog* mechanism identifies misbehaving nodes based on direct observation, and a *pathrater* mechanism constructs routes avoiding these nodes. [11], [12] assumes a tamper resistant security module and use a virtual currency called *nuggets* to charge and reward packet forwarding activities. The proposal in [13] is based on message *receipts* that are sent to a central authority that charges and rewards nodes involved in the transmission of a message. In the context of reputation based systems, CONFIDANT [14] entails a scheme which detects and isolates uncooperative nodes by direct observation and reputation dissemination; CORE [15] consists of two basic components: the already mentioned *watchdog* mechanism and a reputation table which comprises a sophisticated reputation mechanism that differentiates between three types of reputation which are used to specify the resources available for each node.

In summary, current security extensions to OLSR cover a sizeable number of distinct problems. Consensus seems to have been reached in the use of signature and key management systems to ensure the integrity and authenticate the sender of routing control traffic. Similarly, timestamps have found full acceptance in the referred proposals dealing with the replay of old messages. For the remaining issues, however, different techniques have been proposed. In the case of link spoofing by compromised nodes, the techniques presented vary from establishing a line of defense (between trusted and untrusted nodes), to the transmission of a cryptographic message in conjunction with routing control traffic. For incorrect traffic relaying, proposals are based on detecting misbehavior based upon the number of packets sent and received by each node or by the usage of geographical positioning.

Although these proposals solve some of the key security issues, it is our belief that improvements can be made by scrutinizing the underlying assumptions and the aforementioned technical drawbacks. Thus, while adopting some of the generally accepted schemes for tasks such as avoiding replay attacks or guaranteeing integrity and authentication, we will propose a scheme based on rewarding nodes that cooperate with the routing protocol to tackle two of the remaining security issues (fabrication of false routing messages and traffic relay refusal) and avoid the problems mentioned above.

IV. A Cooperative Security Scheme for OLSR

The fundamental concern behind the proposed Cooperative Security Scheme for OLSR (CSS-OLSR) is that of assuring that nodes correctly generate and relay OLSR control traffic. To achieve this goal, the guiding principles will be to reward well behaved nodes and to strongly penalize damaging behavior, as suggested e.g. in [14], [11], [12], [4]. Recall that a well behaved node is a node that: (1) correctly generates routing protocol control traffic and (2) correctly relays routing protocol traffic on behalf of other nodes. Our objective is, thus, to reward nodes that comply with this definition of good behavior. For this purpose, we add three new elements to regular OLSR operation:

- *Complete path message (CPM):* A CPM is used to convey the path traversed by another message through the network. Upon receipt of a TC message, according to the rules specified below, each node sends a CPM back to the originator with the path traversed by the original TC message which, therefore, must have recorded the path traversed by itself (e.g. by setting the record route flag in the IP header or keeping the information on the payload of a TC message).
- *Rating table:* Each node of the network keeps a rating table which holds information about the behavior of its one and two-hop neighbors. Each entry in the rating table has a node ID, a primary and a secondary rating. The node ID uniquely identifies a node, the secondary rating is a classification of a node based on the direct observation, and the primary rating is a more mature classification of a node based on its secondary rating and the matching of the information provided by CPMs with the information announced by a node. The information maintained on this table enables the nodes to decide how to handle misbehaving nodes.
- *Warning message:* Another type of messages called potential misbehavior warning message is used to notify neighbor nodes of potential misbehavior of nodes.

Since CSS-OLSR requires the ability to identify each node and the exact origin of each packet, it relies on the use of a distributed CA's that conforms with the MANET paradigm such as those presented in [10], [6] and [9].

A. Protocol Specification

A security extension to the OLSR protocol that employs the proposed scheme can be defined as follows.

i) At the formation of the network, a distributed CA is employed guarantying the proper authentication of each node;
ii) Each time a new node enters the network, the distributed CA is used to ensure the node's authenticity;
iii) During the broadcast of HELLO messages to ensure knowledge of one and two-hop neighbors, only properly authenticated nodes are considered;
iv) For each authenticated node found, a new entry in the rating table is added with value α for the secondary rating and ρ for the primary rating;
v) The same as items 4, 5, 6 and 7 of the original OLSR protocol (as described in Sec.II);
vi) Upon receipt of a TC message, a CPM containing the path traversed by the TC message may be sent back to the origin depending on the rate λ of CPM transmission;
vii) The same as items 8 and 9 of the original OLSR protocol (as described in Sec.II).

B. Detection of misbehavior through direct observation

The detection of misbehaving through direct observation is done by having each node to listen promiscuously to its MPR transmissions. If the source node of a communication, *S*, detects that a MPR did not relay its message, it decreases the MPR secondary rating by τ_1 and sends a potential misbehaving message to all one-hop neighbors. Upon receipt of this message, each neighbor of *S* decrements the MPR secondary rating by τ_2. Otherwise, if the MPR is detected to relay the message, its secondary rating is increased by γ, but only by node *S*.

To encourage cooperation, the punishment should be greater than the reward, i.e. $\tau_1 > \gamma$. Additionally, the fact that only the source node *S* increases the secondary rating by direct observation and all of its one-hop neighbors decrease it if the node misbehaves makes it harder for a node to keep a good reputation and misbehave often.

In order to motivate nodes to behave well, a node A relays node B's traffic based on the primary rating of B in A, specifically the primary rating controls the rate at which node A relays node B traffic.

C. Detection of misbehavior through analysis of the CPMs

Although OLSR assumes a bidirectional connection between a node and its MPRs, in the following scenarios a node may not detect misbehavior through direct observation of its neighbors: packet collisions, limited transmission power, nodes collusion and partial packet dropping. Therefore the secondary rating (obtained through direct observation of other node's packet forwarding) is only used as an unreliable node status. To classify nodes as misbehaving the primary rating is used. The primary rating is obtained through correlation of the secondary rating and information gained from the CPMs.

To prevent redundant information to be used, upon the reception of a CPM by a node, say node *A*, if the CPM has a path that *A* has sent to his neighbors within a certain period of time β, or a packet generated by the same node has been received within the same period of time, *A* discards it. Otherwise, the processing is as specified in Algorithm 1.

Basically, Algorithm 1 states that if node *A* is the intended receiver of the CPM and has sent a TC message within a period of time δ (step 3), *A* finds the MPR to which he forwarded the packet, say M_1, and checks (a) if the hop after M_1 in the path contained in the CPM belongs to the MPRs of M_1 and (b) if that hop is the one expected by the current routing table of *A*.

If so, and if the secondary rating of M_1 is bigger than the primary rating of M_1 (which corresponds to the node being well behaving), the primary rating of M_1 gets the value of the secondary rating of the same node (step 6). If the secondary rating is lower than the primary rating (the node has been reported as misbehaving) the information of the secondary rating might be corrupted (because direct observation of nodes forwarding is error-prone) and the secondary rating is increased by γ (step 9).

Otherwise, if the information in the CPM is not consistent with what M_1 advertises (step 11) and the secondary rating of M_1 is lower than the primary (misbehaving node), the primary rating of M_1 is set to the value of the secondary rating (step 13). If the secondary rating is bigger than the primary M_1 seems to be well behaving, but because the (more important) CPM information shows the opposite, M_1 secondary rating is decreased by τ_1 (step 15). Afterwards, *A* forwards the packet to all one-hop neighbors for the same processing.

At each node we only verify if its own MPRs are behaving correctly (generating correct traffic and relaying traffic that is sent to them). Although, as the proposed changes in CSS-OLSR are distributed in the sense that every node in the network executes them, the tampering of a message somewhere along a path will also be detected and punished,

Algorithm 1 CPM processing

```
1: SR$_{MPR}$ ←secondary rating of the MPR in A's rating table
2: PR$_{MPR}$ ←primary rating of the MPR in A's rating table
3: if A is the intended receiver of the CPM and A has sent a TC message to the network within a short period of time $\delta$ then
4:     if the information in the CPM is consistent with the information obtained from the MPR by A then
5:         if SR$_{MPR}$ > PR$_{MPR}$ then
6:             PR$_{MPR}$←SR$_{MPR}$
7:         else
8:             SR$_{MPR}$ ←SR$_{MPR}$ + $\gamma$
9:         end if
10:    else
11:        if SR$_{MPR}$ < PR$_{MPR}$ then
12:            PR$_{MPR}$ ←SR$_{MPR}$
13:        else
14:            SR$_{MPR}$ ←SR$_{MPR}$ - $\tau_1$
15:        end if
16:    end if
17:    A forwards the CPM to all one-hop neighbors.
18: else
19:    Forward the CPM as usual.
20: end if
```

eventually not by the source node of the message, but by closer nodes in the path.

V. Discussion

Clearly, CSS-OLSR inherits the benefits of distributed certificate authorities enabling it to identify each node and the exact origin of each packet without a centralized approach. This way, identity spoofing attacks are addressed and countered, whereas to defend against replay attacks the traditional usage of timestamp mechanisms can be relied upon. Beyond these well-understood aspects, our scheme, which correlates error-prone information obtained through direct observation of node transmissions with information obtained from the paths traversed by successfully delivered packets, allows us to resolve the following pending issues:

- *Fabrication of false routing messages* will cause the malicious node to be penalized – by sending incorrect link state information (either from HELLO or TC messages), the paths received in the CPMs will be inconsistent with the information provived by the malicious node, decreasing its primary rating and, consequently, reducing its ability to communicate. The case in which a node simply does not generate any control traffic is not addressed;
- *Traffic relay refusal* can be detected by a correlation of the number of CPMs received, the rate of CPM transmission and the density of the network.

Moreover, our scheme also presents a simple way of solving typical problems (see e.g. [14], [11], [12], [4]) related to stimulation of cooperation among nodes : (a) a simple feedback mechanism avoids the classification of nodes using solely the error-prone detection of neighbors retransmissions; (b) the technique used for traffic relay refusal also detects the situation in which power control causes the source to assume that a packet was sent when in reality it does not reach the next node; (c) reputation information is not disseminated through the network, only among neighboring nodes; and (d) nodes are not able to falsely accuse or praise other nodes because either they would have to generate more CPMs than all other nodes put together (and there is a timeout in which repeated CPMs from the same node are not accepted) or they would have to generate many potential misbehavior messages and collude with other nodes to send accusing CPMs.

It is also worth mentioning that CSS-OLSR is highly configurable in terms of security requirements and traffic overhead. The following variables allow us to fine-tune the protocol according to the desired level of security.

- *Rate of CPM transmission:* the higher the rate, the faster the convergence to a correct detection of the misbehaving nodes, albeit at the price of more traffic;
- *Interval between sending the TC and receiving the CPM:* should be configured according to the bandwidth and the dimension of the network (in a large network it should be increased, otherwise only a small amount of CPMs, originated in close vicinity, will be considered);
- *Timeout between CPMs from the same source:* its value can be configured according to the level of confidence placed on the nodes in the network (if the expected number of malicious nodes is high, the timeout should have a higher value that avoids repetitive malicious CPMs);
- *Initial primary and secondary rating of nodes:* these values may be altered based on the trust placed on the nodes. If they are assumed to be malicious in general, a low primary rating will force them to behave correctly, otherwise communication is impossible.

Due to the use of multipoint relays for conveying traffic information, the overhead incurred by CPMs is much lower than in classical link state routing. As part of our ongoing work we are studying how to optimize the aforementioned parameters (α, ρ, τ_1, τ_2, γ and λ) and how to further integrate the MPR selection process with information from both the rating table and the distributed certification. The results of the compreensive experiments that we are currently carrying out based on an implementation of the CSS-OLSR protocol with different parameter configurations will be reported in a subsequent paper. At a more conceptual level, we are currently also aiming at a game theoretical analysis of the proposed approach.

References

[1] "Daidalos IST Project: Designing advanced interfaces for the delivery and administration of location independent optimised personal services," (FP6-2002-IST-1-506997). http://www.ist-daidalos.org.

[2] T. Clausen and P. Jacquet, "Optimized link state routing protocol (olsr), rfc 3626," October 2003.

[3] P. Jacquet, P. Mühlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Vennot, "Optimized link state routing protocol for ad hoc networks," in *Proc. of IEEE International Multitopic Conference (INMIC 2001)*, 2001.

[4] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," in *MobiCom '00: Proceedings of the 6th annual international conference on mobile computing and networking*. New York, NY, USA: ACM Press, 2000, pp. 255–265.

[5] C. Adjih, T. Clausen, P. Jacquet, A. Laouiti, P. Mühlethaler, and D. Raffo, "Securing the OLSR protocol," in *Proceedings of Med-Hoc-Net*, Mahdia, Tunisia, June 2003.

[6] C. Adjih, D. Raffo, and P. Mülethaler, "Attacks against OLSR: Distributed key management for security," in *2005 OLSR Interop and Workshop*, Ecole Polytechnique, Palaiseau, France, July 28–29 2005.

[7] C. Adjih, T. Clausen, A. Laouiti, P. Mühlethaler, and D. Raffo, "Securing the OLSR routing protocol with or without compromised nodes in the network," HIPERCOM Project, INRIA Rocquencourt, Tech. Rep. INRIA RR-5494, February 2005.

[8] D. Raffo, C. Adjih, T. Clausen, and P. Mühlethaler, "An advanced signature system for OLSR," in *SASN '04: Proceedings of the 2nd ACM Workshop on security of ad hoc and sensor networks*. New York, NY, USA: ACM Press, 2004, pp. 10–16.

[9] D. Raffo, "Security schemes for the OLSR protocol for ad hoc networks," Ph.D. dissertation, Université Paris, 2005.

[10] D. Dhillon, T. S. Randhawa, M. Wang, and L. Lamont, "Implementing a fully distributed Certificate Autorithy in an OLSR MANET," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC 2004)*, Atlanta, Georgia, USA, March 21–25 2004.

[11] L. Buttyan and J.-P. Hubaux, "Enforcing service availability in mobile ad-hoc wans," in *MobiHoc '00: Proceedings of the 1st ACM international symposium on mobile ad hoc networking & computing*. Piscataway, NJ, USA: IEEE Press, 2000, pp. 87–96.

[12] ——, "Stimulating cooperation in self-organizing mobile ad hoc networks," *Mob. Netw. Appl.*, vol. 8, no. 5, pp. 579–592, 2003.

[13] S. Zhong, J. Chen, and Y. R. Yang, "Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks." in *Proceedings of INFOCOM*, San Francisco, USA, March 2003.

[14] S. Buchegger and J.-Y. L. Boudec, "Performance analysis of the confidant protocol," in *MobiHoc '02: Proceedings of the 3rd ACM international symposium on mobile ad hoc networking & computing*. New York, NY, USA: ACM Press, 2002, pp. 226–236.

[15] P. Michiardi and R. Molva, "Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proc. of the IFIP-Communication and Multimedia Security Conference*, Copenhagen, June 2002.