Validation of the OLSR routing table based on trust reasoning

Asmaa Adnane and Christophe Bidan and Rafael T. de Sousa Jr.

Abstract Self organization characteristics of Mobile ad hoc networks (Manet) make traditional security solutions inapplicable. In Manet, any node can be router, and can perturb the routing operation by broadcasting incorrect topological information, making the construction of routing table, which is the primary goal of routing protocols, the more vunerable operation. Thus, in such environments, there is no guarantee that a path between two nodes would be free of misbehavior nodes, and many attacks against the routing table has the objective to perturb the network topology by making legitimate nodes store incorrect routes, so that traffic flows through a specific node (attacker). In this paper, we prove that each OLSR (Optimized link state routing protocol) node is able to validate the topology information based on trust reasonings. In our approach, when an inconsistency is detected, we guarantee that the reasoning node is able to mistrust/reject the compromised route (which constitutes a set of nodes that include those that initiate the attacks), and to identify the remote nodes that are subject to attacks.

1 Introduction

Mobile ad hoc networks (Manet) introduce specific security problems for routing protocols. Several research studies were conducted the last few years aiming at developing robust and secured routing protocols for these networks. However, despite the existence of well-known security mechanisms, the vulnerabilities and features relating to this type of networks might render traditional solutions inapplicable. Es-

Asmaa Adnane, Christophe Bidan
SUPELEC, SSIR team (EA 4039), 5 Av de la Boulaie, 35510 Cesson-Sévigné- France e-mail: firstname.lastname@supelec.fr

Rafael T. de Sousa Jr.
University of Brasília - University of Braslia - AQUARELA team, Av L3 Norte - FT - ENE, 70910-900 Brasília - Brazil e-mail: desousa@unb.br

pecially, the absence of a some centralized unit and fixed infrastructure, Manet are self-organized and any node can be router and can disturb the routing protocol by broadcasting incorrect information.

In MANET, all nodes are required to cooperate in network operation (route discovery), but the absence of authority and infrastructure does not allow the verification of their behavior, separating them to trusted and untrusted entities. Consequently, it is difficult to guarantee the correcteness of received information, and so to have a clear view of the network topology. The presence of one misbehavior node could generate compromised routes, and, as a result, the networks nodes would have to rely on wrong routes to communicate.

In such environment trust is very important; we know that the data is correct when it comes from a person we trust.The concepts of trust have been the object of several recent research projects. Trust is recognized as an important aspect for decision-making in distributed and auto-organized applications [3] [2]. In spite of that, there is no consensus in the literature on the definition of trust and what trust management encompasses. Many authors propose their own definitions of trust, each one concerning a specific research domain [4]. In this paper, we use the trust definition and a language to express trust proposed by [2], which permit to formalize and clarify trust aspects present in communication protocols.

In this paper, we are intertested by securing OLSR protocol [8], and we propose the integration of trust reasonings into each node, to allow a self-organized trust-based control to verify consistency of the network topology (routing table information). An analysis of OLSR brings out the trust rules that characterize this protocol and allows us to express formally the trust-related properties that can be verified by each node to assess the expected correct behavior of the other nodes.

Analysis of trust assumption was proposed in our previous work [17], this analysis highlights possible measures to render OLSR more reliable and this by means of operations and information already existing in the protocol, we conclude that a mistrust-based control can be set up to detect suspect behavior using the correlation between information provided in the received messages. Indeed, in [19] we prove that neighbors discovery and MPR selection can be strengthened and validated by using trust properties and relations. This result motivates extending the approach for validating the routing table of OLSR nodes, which is the main goal of this paper.

The objectives of this paper is first to discuss the OLSR routing table properties, and second to show that trust reasoning on the routing table can be used to validate the topology information, and identify the remote nodes that are subject to attacks as well as a set of mistrusted nodes that include those that initiate the attacks.

1.1 Organization of the Paper

The paper is organized as follows. section 2 surveys related research and our previous works. In section 3, we present notations and the trust specification language. Section 4 proves that, given realistic assumptions, topology information can be val-

idated by each node, and attackers and victims nodes can be identified. Finally, the conclusion summarizes the results and indicates the interest of using trust as one of the means for securing OLSR.

2 Related works

Several studies have been proposed to secure routing protocols for mobile ad-hoc networks. Some approaches, based on cryptography mechanisms, propose to secure existing protocol such as OLSR [14], AODV [15] and DSR [6], or propose a new secured protocol [7]. The security of data forwarding was the main goal of such efforts, disregarding the topology informations. However, these approaches allow all routing data exchanges to be protected from forgery, but do not check the consistency of the received control protocol information (e.g. HELLO and TC messages for OLSR), which does not prevent any malicious router to disseminate incorrect topological information.

Another topic of research use intrusion detection system to secure routing operation in ad hoc networks. Most of this research are based on a generic distributed architecture, each node having its own local intrusion detection system (LIDS) and global detection being performed thanks to a module that allows the cooperation between the LIDS [9, 12, 13]. Other studies treat the problem of cooperation (one of the concepts related to trust) in ad hoc networks and to constrain the selfish nodes to cooperate [10],[5]. These techniques are interested by the detection of misbehavior/selfish nodes but do not allow the verification of the network topology and the detection of incorrect information.

Few works was conducted for securing route discovery based on trust. [18] proposes trust based extension for securing DSR, where trust between nodes is established using certificate mechanism. Authors in [11] propose a secure routing protocol based on AODV in order to find a secure end-to-end route free of any malicious entity.

The OLSR specification [8] does not establish any special security measures, but recognizes that, as a proactive protocol, OLSR is a target for attacks against the periodic broadcast of topological information. To our knowledge, only Wang *& al.* [16] propose a specific intrusion detection approach based on OLSR protocol semantics checking. The semantic properties, that are implied by the protocol definition, are used by every node for conflict checking regarding the correct OLSR behavior.

Our work can be considered as an extension of [16]. Indeed, after an analysis of the semantics properties of OLSR in term of trust [17], we have identified trust-related properties [19], then we have focused on the detection of attacks on MPR selection, where the attacker abuses the properties of the selection algorithm (HELLO message contents and scheduling) to be selected as MPR. Simulation results of these works have demonstrated the effectiveness of such verification in the attacks detection. Finally, our approach allows us to detect more misbehaviors than Wang *& al.* [19] and to validate the network topology. These previous results motivate us to

extend our approach for validating the routing table of OLSR nodes, as explain in section 4.

3 Notations

In OLSR, the node collects information about link configuration and routing topology from the exchanges of HELLO and TC messages, respectively. For these messages, we note:

- $X \xleftarrow{HELLO_Y} Y$, $X \xleftarrow{TC_Y} Y$: respectively, the reception by node X of HELLO and TC messages from node Y,
- $X \xleftarrow{(TC_Z)_Y} Y$: the reception by X of a TC message originated in Z and forwarded by Y,
- $X \xrightarrow{TC_X} *, X \xrightarrow{DATA_X} *$: The broadcast by X of a TC or respectively a data message to be forwarded by its MPRs.
- $X \overset{TC_Y}{\nleftarrow} Y$: absence of an awaited TC message from node Y, which is detected by expiration of a timer held by X,
- $X \overset{(DATA_X)_Y}{\nleftarrow} Y$: supposing that Y is MPR of X, this notation indicates the absence of an awaited DATA message generated by X and forwarded by node Y, which is detected by expiration of a timer held by X.

The node collects and records the received information so as to maintain its vision of the network. This vision, that represents OLSR implicit trust rules and that we use to integrate to OLSR the concept of mistrust towards choices of MPR and routes, is notated as follows:

- $MANET$: the set of the whole MANET nodes,
- LS_x (Link Set): the link set of the node x,
- NS_x (Neighbor Set): the set of symmetric neighbors of the node x ($NS_X \subseteq LS_X$),
- $2HNS_x$ (2-Hop Neighbor Set): the set of 2-hop neighbors of the node x,
- $MPRS_x$: the set of nodes selected as MPR by the node x ($MPR_x \subseteq NS_x$),
- $MPRSS_x$ (MPR Selection Set): the set of symmetric neighbors which have selected the node x as MPR ($MPRSS_X \subseteq NS_X$),
- TS_x (Topology Set): the set containing the network topology as seen by the node x,
- RT_x (Routing Table): the routing table of the node x, consisting of tuples (dad, nad, dis, if) asserting that the node identified by dad is located dis hops away from the local node, that the symmetric neighbor node with interface address nad is the next hop node in the route to dad, and that this symmetric neighbor node is reachable through the local interface if,
- $D(x,y) : MANET^2 \rightarrow \aleph$: the function which provides the distance, expressed as the number of hops, between two network nodes.

- $route_{x \to y}$: sequence of nodes which constitutes the route between x and y in the form of the predicate: $route_{Y_1 \to Y_n} = Y_1, ..., Y_n$ *with* $Y_{i+1} \in MPRS_{Y_i}$

For specifying the clauses concerning trust in the protocol, we use the language proposed by [2] which allows to express trust by the fact that if an entity A trusts an entity B in some respect, informally means that A believes that B will behave in a certain way and will perform some action in certain specific circumstances. With this language, the clauses relating to trust in routing operations are expressed with the following notations:

- the expression $A\ trusts_{fw}(Nodes)$ means that A trusts B $(B \in Nodes)$ to forward its messages. Otherwise, A not trusting B $(\forall B \in Nodes)$ is noted $A \neg trusts(Nodes)$,

4 Trust validation of the routing table

4.1 The routing table in OLSR

The routing table is the result of the OLSR protocol. Each node creates its point of view of the network topology, and calculates the shortest path to any destination using the Dijkstra shortest path algorithm [1]. The routing table RT is described by the following formula:

$$\forall Z \in MANET, \exists Y \in MPRS_X \Rightarrow \exists T \in RT_X, T = (Z, Y, N, I)$$

Each entry in RT consists of: (Z, Y, N, I), and specifies that the node identified by Z is located N hops away from the local node, that the symmetric neighbor node with identified by Y is the next hop node in the route to Z, and that this symmetric neighbor node is reachable through the local interface I. From the point of view of trust, the routing table specifies that X trusts only Y for routing towards Z because it provides the shortest path to Z. According to Dijkstra's definition, the shortest path has the two following properties :

1. A subpath of a shortest path is itself a shortest path.

$$T = (A, Y, N, I) \in RT_X, \forall a_i \in route_{X \to A} \Rightarrow \exists t_i \in RT_X, t_i = (a_i, Y, n_i, I), n_i < N \quad (1)$$

2. In weighted graph, if $P_{X,A}$ is the weight of the shortest path between X and A, for all node B we have : $P_{X,A} \leq P_{X,B} + P_{B,A}$. In the manet the weight of each edge is equal to 1, and the weight function can be replaced by the distance function. Thus, if $D_{X,A}$ denotes the number of hops in the shortest path between X and A, we have:

$$\exists T = (A, Y, N, I) \in RT_X \Rightarrow D(X, A) = N, \forall B \in MANET : D_{X,A} \leq D_{X,B} + D_{B,A} \quad (2)$$

In OLSR, these properties mean that the selected MPR to reach some destination with N hops, must select an MPR which provides a path to the same destination with $N-1$ hops. This property is presented in the figure (1). A selects B as MPR to

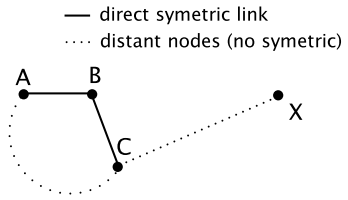

Fig. 1 second property of the shortest path

reach X (shortest path): $(B,X,N,I) \in RT_A$. B selects C as the next MPR to reach the same destination X: $(C,X,N-1,I') \in RT_B$. This implies that :

1. C must not be neighbor of A ($C \notin NS_A$). Indeed, the distance between C and the destination is $N-2$, so, if C is neighbor of A, it means that A made a mistake and should select C as MPR to reach this destination because it provide a shorter path than the one provided by B.
2. When B sends a data packet to destination X, then this packet must not be forwarded by neighbors of A. because the data packets are forwarded only by the node which provides the shortest path (B). Even if the packet is provided by node $C \in NS_A \cap NS_B$, this mean that C provide a shorter path, and that A made a mistake and should select him as MPR to reach a destination of the data packet.

If one of this two situations happens, it means that A did not calculate the correct shortest path. In §4.3 we will show that based on the shortest path property we are able to detect these situations.

4.2 Limitations of previous works

The integration of the trust properties presented previously mitigate certain vulnerabilities of OLSR protocol. Each node is able to verify direct received neighbor information, and if any inconsistency is detected, this neighbor will be mistrusted. When the correlation between a set of neighbors reveals an inconsistency, the set nodes will be mistrusted.

However, we show now that such this is not enough to detect all attacks, and there is at least an attack that can not be detected by previous rules. To present this attack, we first have to prove that according to OLSR specification, each node is able to calculate the routing table of its neighbors :

Theorem 1. *According to OLSR specification, every node is able to calculate the routing table of its neighbors.*

$$\forall X \in NS_A : A \textit{ can calculate } RT_X$$

Proof. Suppose that A and X are symmetric neighbor ($X \in NS_A$), the routing table of any node X is calculated with the following information :

- $MPRS_X$: as X has to advertise all its symmetric neighbors with the status of each link (MPR neighbor, symmetric neighbor) in its HELLO messages, so A can deduce the MPR set of X after the reception of $HELLO_X$.
- TS_X : the topology set of the node X is calculated with the received TC messages from any node. As TC messages are broadcasted in the network, A receives the same TC messages, and is able to generate the same topology set as X.

Thus, A can deduce the MPR and topology set of any neighbor X, and calculate the routing table of X, and so the distance between X and any other node in the network. □

Given this theorem, the routing attack consists in the following steps (figure 2):

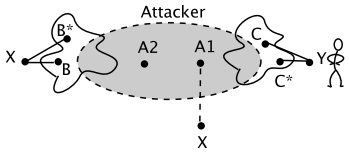

Fig. 2 Model of the routing table attack

1. Let the node $A \in MANET$ be an attacker. First, the attacker behaves correctly to collect information about network and builds a view of network topology.
2. The attacker selects the target node X where $D(A,X) > 3$, such that the two group of nodes $G1$ and $G2$ have the following characteristics :
   - According to the theorem 1, the attacker can calculate the distance between its neighbors and the target X. Nodes in $G1$ are neighbors farthest from the target:

$$G1 = \{Z \in NS_A / D(Z,X) \geqslant D(A,X)\}$$

   - Let $G2$ be the groups defined by:

$$G2 = NS_A - G1$$
$$G1 \cap G2 = \emptyset \textit{ and } G1 \cup G2 = NS_A$$

3. The attacker selects the second target Y between the following set of nodes:

$$G3 = \bigcap_{z \in G2} \{y / D(z,y) > D(y,A)\}$$

   if $G3 = \emptyset$, the attacker chooses another target X.
4. The attacker takes two different identities $A1$ and $A2$ with two different interfaces.

5. Each virtual attacker node advertises the other one as a symmetric and MPR neighbor, so as to construct a chain of MPR nodes, and $A1$ advertises being MPR of X :

$$NS_{A1} = G1 \cup \{A2, X\}, \;\; X, A2 \in TC_{A1}, \; \forall Z \in G2, \; (Z, A2, N, I1) \in RT_{A1}$$
$$NS_{A2} = G2 \cup \{A1, Y\}, \;\; Y, A1 \in TC_{A2}, \; \forall Z' \in G1, \; (Z', A1, N, I2) \in RT_{A2}$$

6. Consequently, when the TC_{A1} message is broadcasted (with $X \in TC_{A1}$):
   - When $A2$ receives this message, it generates another message without the node X, and broadcasts it as a TC message of $A1$. At the reception of the modified TC_{A1}, X will not detect any inconsistency.
   - When nodes included in $G1$ receives TC_{A1}, then all nodes in $G1$ broadcast the message and select $A1$ as MPR, as it seems to be the shortest path to X, it will be chosen in the routing table as next hop to reach X :

$$\forall Z \in G1, (X, A1, N, I) \in RT_Z \tag{3}$$

   - Finally, when Y receives TC_{A1}, it updates its routing table with the new detected distance.
7. Node $A2$ will forward all packets coming from $A1$ and $G1$ nodes with destination to X, so attackers can not be detected as deny of service attack.

Notice that, in this attack all packets reach the destination but not necessary following the shortest path. The network will function correctly, but attackers will give a wrong view of the network topology and gain the trust of their neighbors since they have selected it as MPR. This situation constitute an attack to the trust properties, it is important to point out that the scenario is not easily applicable, and the attacker must be well positioned to achieve it. In the following section, we present a trust-based reasoning to detect the attack.

4.3 The shortest path theorem

In this section we present a theorem based on the shortest path properties, and its utilization for the detection of routing attacks. We assume that any node sends a response at the reception of DATA packet. Thus, deny of service attack can not be executed, or at least it will be detected.

Following the description of MPR selection and routing table calculation in [8], we have deduced a property derived from the shortest path properties (1 and 2). Our property specifies the unicity of the shortest path length, i.e., if the node X calculates the shortest path to a target Y with N hops, Y should have the same length of shortest path to reach X.

Theorem 2. *Let X and Y two distant nodes, if X calculates the shortest path to a target Y with N hops, then Y should also have the same length of shortest path to*

reach X.

$$\exists t = (Y, M_X, N, I_X) \in RT_X,\ D(X,A) = N$$
$$\textit{and } \exists t' = (X, M_Y, N, I_Y) \in RT_A,\ D(A,X) = N' \Rightarrow N = N' \quad (4)$$

Proof. $N \in \aleph^*$
for $N = 1$, X and Y are symmetric neighbors $N = N' = 1$.
for $N = 2$, X and Y are 2-hop neighbors $N = N' = 2$.
for $N \geqslant 3$: let $route^N_{X\to Y}$ the shortest path calculated by X to reach Y with N hops. As the path between X and Y is a succession of symmetric neighbors, there is at least one path between X and Y with N hops, which is the opposite direction of $route^N_{X\to Y}$. If Y has selected another shortest path $route*^{N'}_{Y\to X}$, then :

- $N' < N$: it means that the path calculated by Y ($route*^{N'}_{Y\to X}$) is shorter then the path calculated by X, and the opposite direction of $route*^{N'}_{Y\to X}$ beginning by X is shorter then the shorter path selected by X to reach Y, Contradiction, $route^N_{X\to Y}$ is not the shortertest path.
- $N' > N$: it means that the path calculated by X ($route^N_{X\to Y}$) is shorter then the path calculated by Y, and the opposite direction of $route^N_{X\to Y}$ beginning by Y is shorter then the shorter path selected by Y to reach X, Contradiction, $route*^{N'}_{Y\to X}$ is not the shortertest path.

So, $N = N'$, the both shortest path has the same length. □

Notice that, it is not necessary the same route, $route_{X\to Y}$ can be different from $route_{Y\to X}$, but they must have the same number of hops.

4.4 Attack detection based on shortest path theorem

Let us suppose that node X selects the MPR B as next hop to the distant node Y ($(Y,B,N,I) \in RT_X$), and that node Y selects the MPR C as next hop to reach X ($(X,C,N,I') \in RT_Y$). When the routes are calculated correctly, if the node X sends data packet to Y, according to the theorem 2, Y should receive the data packet through a node that provides a path to X having the same length than the calculated shortest path.

Let us now suppose that the attack described in the figure 2 is used. Y calculates a path to reach X that is not the real shortest path ($(X,C',N',I) \in RT_Y$). When Y sends a data packet to X, the packet will not follow N' hops as it was calculated, but it will follow $D(Y,A) + D(A,X)$ hops, where A is the attacker. When the destination node X receives the Y data packet, it verifies the hop count N_p provided in the packet as specified in OLSR [8]. If $N_p = N$ we have two possibilities:

- The data packet is received through the neighbor B' : X will calculate the length of the shortest path between B' and Y, if not equal to N, the attack is detected.

- The data packet is received through the neighbor B : the attack can not be detected,

If $N_p = N$ and X can not detect the attack, this means that the attacker $A2$ correctly calculates the hop count and modifies it in the data packet, received from $A1$, before forwarding it to the $G2$ nodes, so the new hop count is $N - D(A2, X)$. But it is almost difficult for the attacker $A2$ to calculate the length of the shortest path between X and Y. Even if, it can calculate it, it should be located in the network in the manner to verify that $N > D(A2, X)$, which make the attack more and more difficult.

In the case that $N_p \neq N$, the attack is detected. In term of trust, X has to mistrust the node Y because it may be subject of attack, and mistrust all the nodes included into the route provided by the node which forward the packet (B or B'). We can resume this deduction by the following formula:

$$T = (Y, B, N, I) \in RT_X, \exists Z \in NS_X, X \overset{DATA_{Y-X}}{\longleftarrow} Z : hop\ count\ of(DATA_{Y-X}) \neq N \\ \Rightarrow \forall W \in route_{Z \to Y}, X \neg trusts(W \cup \{Y\}) \quad (5)$$

This expression presents that X sends a packet to destination Y using its MPR B which provide the shortest path to this destination (N hops), but when X receives a packet from Y provided by it neighbor Z, X verifies if the hop count of this packet ($N_p = hop\ count(DATA_{Y-X})$) is equal to the distance N of the shortest path calculated to reach Y: $T = (Y, B, N, I) \in RT_X$, If the two distances are not equal ($N_p \neq N$),X has to mistrust the path proposed by Y, and thus mistrust any node W constituting this path $W \in route_{Z \to Y}$.

Such a situation can not allows X to decide which is the misbehavior nodes. So it is worth to point out that the mistrust reasoning does not every time allow the precise identification of the misbehaving node, but allow the detection of a behavior anomaly related to a group of nodes which includes the attacker, as indicated in last expression 6. In the other side, when Y receives data packets from X, or a response to its previous request, it can detect this attack with the same reasoning presented by the following expression :

$$T' = (X, C, N*, I') \in RT_Y, \exists Z \in NSY, Y \overset{DATA_{X-Y}}{\longleftarrow} Z : hop\ count\ of(DATA_{X-Y}) \neq N* \\ \Rightarrow \forall W \in route_{Z \to X}, Y \neg trusts(W) \quad (6)$$

This expression presents the behavior of Y in the expression 5, when it receives a DATA message from X, Y verifies if the hop count of the received message is equal to the length of the shortest path $N*$ calculated in the routing table ($(X, C, N*, I') \in RT_Y$), If the two distances are not equal ($N* \neq hop\ count\ of(DATA_{X-Y})$, Y has to mistrust the path proposed by X, and thus mistrust any node W constituting this path $W \in route_{Z \to X}$.

5 Conclusion

In this paper, we have proposed a trust based reasoning for OLSR that allows each node to correlate information provided by HELLO, TC messages and DATA packets information so as to valide its local view of the global network topology. In our approach, when an inconsistency is detected between any received messages and its local view, the reasoning node is able to identify the compromised route. Notice that our approach does not require any modification of the bare OLSR, but only the integration of trust reasoning on each node.

In futur work, we plan to use this result to implement a trust based detection system. When a node has identified a compromised route, a first approach consists in locally changing the MPR selection so as to *black-list* the neighbor node that is the next-hop of the compromised route. Such approach can be enhance by allowing nodes to share their trust information. However, it is worth to point out that second-hand information can be subject to false accusations. To mitigate this problem, we plan to set up a mechanism that allows each node to give a proof of its mistrust opinion to participate in the propagation of mistrust towards the network. Such a mechanism could be used to enforce a reputation systems by establishing trust relationships before cooperating with the other nodes. Brown B, Aaron M (2001) The politics of nature. In: Smith J (ed) The rise of modern genomics, 3rd edn. Wiley, New York Dod J (1999) Effective Substances. In: The dictionary of substances and their effects. Royal Society of Chemistry. Available via DIALOG.

References

1. Johnson D B (1973) A note on Dijkstra's shortest path algorithm. In: Journale A.C.M. 385–388.
2. Yahalom R, Klein B, Beth T (1993) Trust Relationships in Secure Systems – A Distributed Authentication Perspective. In: RSP: IEEE Computer Society Symposium on Research in Security and Privacy.
3. Marsh S (1994) Formalising Trust as a Computational Concept. PhD thesis in the university of Stirling.
4. Grandison T, Sloman M (2000) Survey of Trust in Internet Applications. In: IEEE Communications Surveys and Tutorials.
5. Buchegger A, Le BoudecJ-Y (2002) Performance Analysis of the CONFIDANT Protocol (Cooperation Of Nodes - Fairness In Dynamic Ad-hoc NeTworks). In: MobiHoc 2002: The Third ACM International Symposium on Mobile Ad Hoc Networking and Computing. Lausanne,
6. Hu Y, Perrig A Johnson D (2002) Ariadne: A secure on-demand routing protocol for ad hoc networks. In: Proceedings of MobiCom. 21–38.
7. Sanzgiri K, Dahill B, Levine B, Royer E, Shields C (2002) A Secure Routing Protocol for Ad Hoc Networks. In: International Conference on Network Protocols (ICNP). Paris, France. IEEE Computer Society. 78–87.
8. Clausen T and Jacquet P (2003) IETF RFC 3626: Optimized Link State Routing Protocol OLSR.

9. Zhang Y, Lee W, Huang Y (2003) Intrusion detection techniques for mobile wireless networks. In: ACM Wireless Networks Journal, 9(5):545-556.
10. Michiardi P (2004) Mécanismes de sécurité et de coopération entre noeuds d'un réseau mobile ad hoc. Ecole nationale supérieur des télécommunications, Paris.
11. Pissinou N, Ghosh T, Makki K (2004) Collaborative Trust-Based Secure Routing in Multihop Ad Hoc Networks. In: Networking Technologies, Services, and Protocols; Performance of Computer and Communication Networks; Mobile and Wireless Communication, Third International IFIP-TC6 Networking Conference. Athens, Greece. (3042):1446–1451.
12. Puttini R S, Mé L, De Sousa R T (2004) On the Vulnerabilities and Protection of Mobile Ad Hoc Network Routing Protocols. In: Proceedings of the 3rd International Conference on Networking ICN'2004.
13. Puttini R S, Percher J-M, Mé L, De Sousa R T (2004) A fully distributed IDS for MANET. In: IEEE Computer Society, ISCC, 331–338.
14. Raffo D (2005) Security Schemes for the OLSR Protocol for Ad Hoc Networks. PhD thesis in the University of Paris 6.
15. Zapata M G (2005) Secure Ad hoc On-Demand Distance Vector (SAODV) Routing. In: IETF MANET Internet Draft.
16. Wang M, Lamont L, Mason P, Gorlatova M (2005) An Effective Intrusion Detection Approach for OLSR MANET Protocol. In: First Workshop on Secure Network Protocols (NPSec). Boston, Massachusetts, USA. 55–60.
17. Adnane A, De Sousa R T, Bidan C, Mé L (2007) Analysis of the implicit trust within the OLSR protocol. In: Joint iTrust and PST Conferences on Privacy, Trust Management and Security (IFIPTM-2007). Moncton, New Brunswick, Canada.
18. Gilaberte R L, HerreroL P (2007) A secure routing protocol for ad hoc networks based on trust. In: Proceedings of the Third International Conference on Networking and Services (ICNS).IEEE Computer Society.
19. Adnane A, De Sousa R T, Bidan C, Mé L (2008) Autonomic trust reasoning enables misbehavior detection in OLSR. In: 23rd Annual ACM Symposium on Applied Computing (SAC'2008). Fortaleza, Ceará, Brazil.