QoS and security in Link State Routing protocols for MANETs

Gimer Cervera, Michel Barbeau, Joaquin Garcia-Alfaro, Evangelos Kranakis

QoS and Security in Link State Routing Protocols for MANETs

Gimer Cervera*, Michel Barbeau†, Joaquin Garcia-Alfaro‡ and Evangelos Kranakis†

* Universidad Tecnológica Metropolitana, 97279, Merida, Yuc., Mexico
Email: gimer.cervera@utmetropolitana.edu.mx

† School of Computer Science, Carleton University, K1S 5B6, Ottawa, Ontario, Canada
Email: {barbeau,kranakis}@scs.carleton.ca

‡ Telecom SudParis, 91000, Evry, France
Email: joaquin.garcia-alfaro@acm.org

***Abstract*—We study security issues in the Optimized Link State Routing (OLSR) protocol with Quality-of-Service (QoS). We propose the function *k*-robust-QANS, to construct a Quality Advertisement Neighbor Set (QANS). Given a node *v*, the one-hop nodes selected as part of its QANS generate routing information to advertise, when possible, a set with *k+1* links to reach any two-hop neighbor. Several approaches have been proposed to construct a QANS. However, none of them guarantees that the best links are advertised. A mechanism is presented for QANS construction with guarantee that the best links are advertised with respect to a given routing metric. We present the unadvertised quality links problem when QoS is considered. We also address the slanderer attack, i.e., a misbehaving node that advertises incomplete routing information. Our goal is to find a tradeoff between security and amount of information disseminated. We conduct simulations that confirm our claims.**

***Keywords*—Network Security, Wireless Security, Routing, OLSR, Ad Hoc.**

I. Introduction

We address vulnerabilities of the Optimized Link State Routing (OLSR) [1] protocol when Quality-of-Service (QoS) is considered. OLSR is a proactive link state routing protocol designed exclusively for Mobile Ad Hoc Networks (MANETs). OLSR is defined in RFC3626 [1] with no built-in security or QoS. The core of the protocol is the selection of Multipoint Relays (MPRs) [1] to flood the network with partial link state information. Hello and Topology Control (TC) messages are used to construct and maintain routing tables. Hello messages are not retransmitted. TC messages are forwarded exclusively by the MPRs. MPRs generate TC messages to advertise their MPR selectors. Every node obtains a partial view of the network topology and uses this information to compute the shortest path (in terms of hops), to reach any other node in the network.

A QoS routing scheme over the OLSR protocol, proposed by Badis et al., is called QOLSR [2]. The best routes in QOLSR are selected considering a given routing metric, i.e., a characteristic associated to a link or a node in a MANET [3]. Hello and TC messages are extended to advertise routing metric information. Several approaches refine the MPR selection according to different routing metrics [2]–[7]. However, they do not take into account the link advertisement process. Therefore, there is no guarantee that all the best links are advertised. When QoS is considered, the MPR set is known as the Quality Advertised Neighbor Set (QANS). The QANS may be different from the MPR set constructed by the original OLSR. In QOLSR, every node selects a QANS to cover every two-hop neighbor by at least one QAN. Every link between a node in the QANS and its selector is advertised in every TC message. However, the link between a QAN and the two-hop neighbor for which it was selected, is advertised only if that QAN selects the two-hop node as part of its own QANS or vice versa. The link advertisement process might be also affected by a misbehaving QAN generating incomplete TC messages. A node can detect if one of its QAN misbehaves by overhearing and analyzing its TC messages. To detect that a QAN fails to advertise a link requires additional information. The RFC3626 [1] defines two mechanisms to increase the amount of information to be disseminated: MPR selection with additional coverage and TC_Redundancy (TCR) parameter. The selection of MPRs with additional coverage specifies by how many MPRs any two-hop neighbor should be covered. The TCR parameter determines the amount of information that may be advertised in TC messages. These mechanisms increase the number and size of TC messages [8]. Besides, there is no routing metric support. The approaches presented in [2]–[7], select either a QANS or a MPR set with redundancy. Nevertheless, the link advertisement process remains as proposed in RFC3626 [1], which may fail to advertise important links.

We propose the function *k*-robust-QANS (i.e., Algorithm 1), that constructs a $QANS$ sensitive to the quality of the links. Given a node v, the nodes selected as part of $QANS_{k+1}(v)$ generate TC messages to advertise, when possible, $k+1$ links to reach any two-hop neighbor of v. A HELLO message accomplishes three independent tasks: link sensing, neighbor detection and MPR selection signaling. We propose an extension to Hello messages enabling the QANs to advertise the links for which they were selected, i.e., the links to the selectors and the links to the associated two-hop neighbors. We are aiming at a tradeoff between security and amount of information disseminated. As an alternative, we also propose an extension of the parameter TCR to address the case when the QANS and MPR sets are different. Nevertheless, the challenge is to decide when to update the TCR parameter. The

function k-robust-QANS reduces the size of the information included in the TC messages and guarantees that the best links are advertised, with respect to a given routing metric.

A. Network Model

Let us model a network as a graph $G = (V, E, R)$, where $V = \{v_1, v_2, \ldots, v_n\}$ is a set of nodes and $E \subseteq V \times V$ is a set of links. Every link $(v_i, v_j) \in E$ models the fact that nodes v_i and v_j are within wireless communication range. A path from a source s to a destination d is a sequence $p \equiv s = v_0, v_1, \ldots, v_{k-1}, v_k = d$ with $(v_i, v_{i+1}) \in E$ for $i = 0, 1, \ldots, k-1$. R is a routing metric used to evaluate links, e.g., delay $D(v_i, v_j)$. We assume that the links and routing metrics are bidirectional. Assume we are given a symmetric routing metric R, i.e., $R(v_i, v_j) = R(v_j, v_i)$ for all $i, j \in \{i, 2, ..., n\}$. $W(v_0, v_k)$ is used to weight a path, without loops, from v_0 to v_k, i.e., $W(v_0, v_k) = R(v_0, v_1) + R(v_1, v_2) + \ldots + R(v_{k-1}, v_k)$.

The most common measure of the length of a path is the hop count. However, the shortest path in terms of hops might not be the optimal path in terms of QoS. The optimal path p from a source to a destination, must meet the end-to-end QoS requirements. Therefore, several routing metrics are used to describe the state of a node (e.g., battery residual lifetime) or a link (e.g., bandwidth). The end-to-end delay of a path p is the sum of the delays of the individual links forming the path. The delay metric is an example of an **additive** metric, i.e., $D(p) = \sum_{i=0}^{k-1} D(v_i, v_{i+1})$. The bandwidth of a path p is the minimum bandwidth of all the individual links making the path, i.e., B(p) = $\min\{B(v_i, v_{i+1}) | i = 0, 1, \ldots, k-1\}$. The bandwidth is an example of a **concave** metric. The packet loss probability is an example of a **multiplicative** metric, i.e., $P(p) = \prod_{i=0}^{k} P(v_i)$.

Organization of the paper — Section II reviews the original OLSR and QOLSR protocols. The problem statement is presented in Section III. Section IV presents the related work. We present our countermeasures in Section V. In Section VI, we describe our experiments and results. Section VII closes the paper with our conclusions.

II. Optimized Link State Routing Protocol (OLSR)

In this section, we describe the original OLSR protocol. OLSR is a proactive table driven routing protocol designed exclusively for MANETs. OLSR is defined in RFC3626 [1]. Every node computes and maintains a routing table that includes the next hop to reach any other node in the network. To find the shortest path, every node uses the hop count metric. Every node exchanges Hello messages to discover its one and two-hop neighbors. Every node selects among its one-hop neighbors a set of nodes such that every two-hop neighbor is covered by at least one node in the MPR set. The MPRs generate and retransmit Topology Control (TC) messages. In every TC message, the MPRs advertise their selector nodes, i.e., the nodes that selected it as MPR. Every node computes its routing table using the information gathered from the Hello and TC messages. OLSR was designed with no built-in QoS and security. The core of the protocol is the selection of MPR nodes for the dissemination of control traffic information. Only partial link state information is advertised. In every TC message, MPRs include their Advertised Neighbor Set (ANS), i.e., the MPR selectors. OLSRv2 is presented as an Internet-Draft in [9] by Clausen et al. OLSRv2 retains the same basic mechanisms and algorithms for distributing control traffic (i.e., MPR-based flooding), but provides a more efficient signaling framework and implements some message simplification. OLSRv2 does not include security or quality metrics in its original design.

A. QoS OLSR

In this section, we describe the heuristics presented by Badis et al. [2], [4]. The authors presented two heuristics to select MPR sets considering two QoS metrics, i.e., bandwidth and delay. They proposed the *QOLSR-MPR1* and *QOLSR-MPR2* heuristics to select MPRs. These heuristics are based on the original MPR selection proposed in RFC3626 [1]. To explain the heuristics, consider $N_1(v)$ as a set that contains the one-hop neighbors of v, $N_2(v)$ is a set that contains the two-hop neighbors of v. The degree of a node v, d(v,v_i), is the number of nodes in $N_2(v)$ covered by $v_i \in N_1(v)$ such that $N_1(v_i) \cap N_2(v) \neq \emptyset$. M is the MPR set for a given node v, such that, every two-hop neighbors is covered by at least one node in $M \subseteq N_1(v)$. The reachability, $r(v, v_i, M)$, is the number of two-hop neighbors of v that are reachable through the one hop neighbor $v_i \in N_1(v) \setminus M$ and not reachable through the MPR set M. In *QOLSR-MPR1*, if there is more than one-hop neighbor covering the same uncovered two-hop neighbors, then the one-hop neighbor with highest reachability is selected as MPR. In case of a tie, the one with the maximum bandwidth link to the current node is selected as MPR. In case of a second tie, the node with the minimum delay is selected. *QOLSR-MPR1* does not guarantee computation of the optimal shortest path. To solve this problem, the authors proposed *QOLSR-MPR2*. In this heuristic, the one-hop neighbor that guarantees the maximum bandwidth and minimum delay is selected as MPR. In case of a tie, the node with highest reachability r is selected. The heuristic starts with an empty MPR set M. Select as MPRs the one-hop neighbors that provide unique paths to two-hop nodes. Remove the nodes in $N_2(v)$ that are covered by a node in the MPR set. While there exist nodes in $N_2(v)$ not yet covered by a node in M, select the one-hop neighbor that provides the best quality link. Remove the two-hop nodes now covered by the selected node. The QANS and the MPR set are the same. As a main drawback, both heuristics only consider the one-hop neighbors to select the MPRs.

B. QOLSR+

In [5], Munaretto and Fonseca presented three variants of the original QOLSR [2]. The authors proposed the heuristics: QOLSR1, QOLSR2 and QOLSR+. QOLSR1 selects as MPR the node in $N_1(v)$ with highest reachability. In case of a tie, select the one-hop neighbor with the minimum delay link. QOLSR2 selects the one-hop neighbor with the minimum delay link as MPR. In case of a tie, select as MPR the node in $N_1(v)$ with highest reachability. QOLSR+ selects as a MPR the neighbor node with minimum delay among the neighbors that are, at most, within two hops from the selector node,

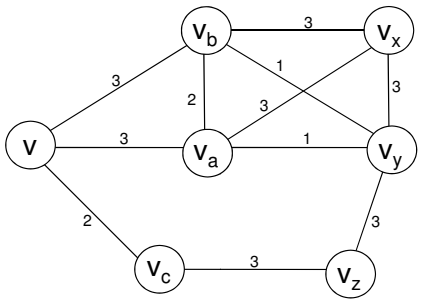


Fig. 1. Example of a network with links labelled with the bandwidth metric. Unadvertised quality links problem.

i.e., a node v selects as MPR, the node $v_i \in N_1(v)$ that has minimum delay path considering $N_1(v)$ and $N_2(v)$. In case of a tie, select as MPR the node in $N_1(v)$ with highest reachability. The authors demonstrate that the QOLSR+ heuristic finds the minimum delay path using only partial knowledge of the network topology. However, the QOLSR+ heuristic does not minimize the size of the MPR sets as proposed in RFC3626 [1]. Therefore, the number of TC messages increases due to the additional coverage in the selection of the MPRs. Therefore, $M = \{v_i, v_b, v_a\}$.

III. The Unadvertised Quality Links Problem and Slanderer Attack

In this section, we describe vulnerabilities of OLSR with QoS. OLSR was proposed with no built-in QoS in its original design. Several approaches have been proposed to improve the QANS selection considering the quality of the links. However, they do not consider the advertisement of link states. In this section, we present the unadvertised quality links problem and slanderer attack.

A. Unadvertised quality links problem

In this section, we describe the unadvertised quality links problem. When a QANS is selected, the QAN nodes with non empty tables generate TC messages advertising their selectors. Otherwise, the MPRs advertise their selectors. In both cases, the generator of a TC message only advertises the links to selectors. A link between a QAN and a two-hop neighbor is advertised only if that neighbor is also a selector of the QAN or vice versa. Connectivity is not affected. However, there is no guarantee that the best links are advertised. For instance, consider a network with links labelled with the bandwidth metric in Figure 1. Following the heuristic $QOLSR+$ presented in Section II-B, the QANS and MPR set are identical. First, nodes v_a and v_b have equal chances of being selected as a QAN by v. Suppose $QANS(v) = \{v_b, v_c\}$. From the perspective of v_x, it might select the $QANS(v_x) = \{v_a, v_y\}$. In this case, the link (v_b, v_x) is not advertised. As a consequence, v_z is unable to construct the best path to reach v_b, i.e., (v_z, v_y, v_x, v_b). Suppose that every node selects the node with the lowest id in case of a tie. In this case, the path $(v_z, v_y, v_x, v_a, v, v_b)$ can be constructed. However, this is not the optimal path, in terms of hop number. As an alternative, the node v_b may set its TCR parameter to advertise its one-hop neighbors. Nevertheless, the nodes need to analyze their local information to decide the amount of information that should be included in its TC messages. This strategy reduces the performance of the network.

B. Slanderer attack

According to RFC3626 (Section 9.3), the list of selectors in every TC message can be partial (e.g., due to size limitations). However, all selectors should be advertised within a certain period of time. A misbehaving node might take advantage of this. Let us consider a scenario with a misbehaving node w generating TC messages reporting an incomplete list of QANS selectors, i.e., there exists at least a node v in the QANS selector set of node w that is not included in its TC messages. As a result of the attack, the best link to reach node v might not be advertised. A node can detect this improper behavior by analyzing the TC messages generated by its QANs. In OLSR, every node must recompute its routing table when a change in the topology is detected. As a variation of the attack, the misbehaving node might randomly advertise a QANS selector. The receivers detect a change in the topology and recompute their routing tables. This creates unnecessary overhead and reduces the performance of the network.

IV. Related work

In this section, we present approaches that attempt to integrate QoS to OLSR. The heuristic presented in RFC3626 [1] uses *Willingness* as the principal criterion to select the MPRs. *Willingness* is the only routing metric defined in RFC3626 [1]. Several approaches use other routing metrics to improve the MPR selection in OLSR. Sondi [3] proposed, a generic heuristic that considers multiple routing metrics during the MPR selection. The author assumes that the selection of the MPRs is essential to support QoS in OLSR. Therefore, every node must choose its MPR set taking into account several routing metrics to select the best links to its one-hop neighbors. These links are advertised in every TC message. Approaches focus on improving the dissemination of the high quality links. Moraru and Simplot-Ryl presented in [6] a method for QoS path selection based on network complexity reduction. Every node selects one-hop neighbors that provide optimal links to two-hop neighbors. The selection is done considering a specific metric. First, every node eliminates from redundant paths, the worst performance links. The graph reduction is a variation of the Relative Neighborhood Graph (RNG). Then, every node selects and maintains a QANS. The QANS provides optimal connectivity to the two-hop neighbors. The QANS nodes advertise their selectors in every TC message. Besides, every node constructs a MPR set. The QANS and MPR are the same. In this approach, the QANS and MPR are different. In [7], Khadar et al. also proposed a QANS selection mechanism that takes into account QoS metrics of any kind in OLSR networks. The authors proposed First Node on Best Path (FNBP) based on QANS selection. In FNBP, a node constructs the QoS weighted shortest path towards its one-hop and two-hop neighbors. FNBP allows a node to construct a path whose length, in terms of hops, is greater than two for reaching a two-hop neighbor. It also reduces the number of advertised neighbors while selecting optimal paths. The authors proposed a heuristic to minimize the QANS. FNBP attempts to improve the QANS selection proposed in [6] by

Moraru and Simplot-Ryl. The authors also showed that their solution gets closer to the optimal route than QOLSR [2]. The main drawback with these approaches is that every time the topology changes, the QANS and MPR set must be recomputed. Hanzo and Tafazolli presented in [10], a survey of QoS routing solutions for MANETs. Santhi and Nachiappan presented in [11] a similar overview of QoS metrics, resources and factors affecting the performance of QoS routing protocols for MANETS. They compared and studied the strength, weakness and applicability of existing QoS routing protocols. The authors considered security as one of the main issues and challenges to be addressed to ensure QoS in MANETs. In [12], Gujral and Kapil proposed a secure QoS enabled on-demand link-state multipath routing mechanism. Their mechanism uses symmetric key cryptography and a one-way HMAC based key chain for broadcast node authentication to detect altering routes. In [13], Subramaniam and Baskaran proposed Energy AODV (EN-AODV). The nodes receive information about the energy levels of the intermediate nodes before computing a routing path. A node is considered for routing only if its energy level is above a predefined threshold value. The transmissions are made secure by introducing a message digest algorithm. In [14], Sedaghat et al. proposed a mechanism to obtain secure MANET QoS routing with regard to QoS considerations. The authors presented a new mechanism over on-demand multipath routing protocol (AOMDV) to provide reliability and security, while ensuring minimum data redundancy.

V. k-ROBUST-QANS

In this section, we present our proposed countermeasures to enhance the security and link advertisement process. We present the function *k*-robust-QANS and an extended interpretation of the TCR parameter to advertise redundant links.

A. QANS selection with additional coverage

First, we describe the function *k*-robust-QANS. Given a node v, function *k*-robust-QANS, selects a QANS considering $N_1(v)$ and $N_2(v)$. The QANS is computed, such that the path to reach a node v_j in $N_2(v)$, is advertised, when possible, $k+1$ times. Consider the following notation:

- $h(v, v_u)$: is the minimum number of hops between nodes v and v_u.
- PoorlyCovered: is a set of nodes $v_j \in N_2(v)$ such that $|N_1(v_j) \cap N_1(v)| < k$.
- required(v,A): is a set of nodes $b \in N_1(v)$. For which there is an $a \in A \subseteq N_2(v)$, such that $h(b, a) \leq 1$.
- linked(v, A): is a set of nodes $b \in N_2(v)$. For which there is a set $A \subseteq N_1(v)$ such that $N_1(b) \cap A \neq \emptyset$.
- nextNode(v, A, B): is a set of nodes $v_i \in A$ such that v_i provides the best link to reach a node v_j in B considering a specific routing metric. In case of a tie, we select the node with largest degree d.
- $E'(v)$: is a subset of E that includes the links connecting the nodes $N_1(v)$ and $N_2(v)$.
- $Cov(v, v_i)$: is a set of nodes $v_j \in N_2(v)$, for which node $v_i \in N_1(v)$ provides the best quality link.

Algorithm 1 The k-robust-QANS function.

```
1: function k − robust − QANS(n, k) → Q
2:     Q ← ∅;
3:     t ← 0;
4:     sLinks ← E'(v);
5:     PoorlyCovered ← N2(v) \ k-covered(v,2);
6:     Q_temp ← required(v, PoorlyCovered);
7:     N2(v) ← N2(v) \ PoorlyCovered
8:     repeat
9:         Q_aux ←QAN-set(v, sLinks, N2(v), t);
10:        Q_t ← Q_temp ∪ Q_aux;
11:        t ← t + 1;
12:    until (Q_aux = ∅ or t ≥ k or sLinks = ∅)
13:    Q ← ⋃_{t=0}^{k} Q_t;
14:    return Q;
15: end function
```

Algorithm 2 The k-covered function.

```
1: function k − covered(v , k) → S
2:     S ← ∅;
3:     for all (v_j ∈ N2(v)) do
4:         if |N1(v_j) ∩ N1(v)| ≥ k then
5:             S ← S ∪ {v_j};
6:         end if
7:     end for
8:     return S;
9: end function
```

Algorithm 3 The QAN-set function.

```
1: function QAN − set(v, sLinks, S, t) → Q_t
2:     Q_t ← ∅;
3:     for all (v_2 ∈ S) do
4:         v_1 ← nextNode(n, Remainder, Q_t);
5:         Cov(v, v_1) ← Cov(v, v_1) ∪ v_2
6:         Q_t ← Q_t ∪ {v_1};
7:         sLinks ← sLinks \ linked(v_1,v_2);
8:     end for
9:     return Q_t;
10: end function
```

k-robust-QANS has two phases. In the first phase, it obtains the nodes in $N_2(v)$ covered by only one node in $N_1(v)$. Such nodes in $N_2(v)$ are called poorly covered. It assigns to Q_{temp} the nodes in $N_1(v)$, which exclusively provide reachability to the poorly covered nodes. The link (v_i, v_j), such that $v_i \in N_1(v)$ and v_j is poorly covered should always be advertised by v_i. The second phase computes, when possible, a $QANS_{k+1}(v)$. It invokes the function QAN-set (i.e., Algorithm 3), to obtain subset Q_{aux} of $N_1(v)$ that covers all the nodes in $N_2(v) \setminus PoorlyCovered$. The number of subsets obtained by invoking the function QAN-set is counted by the variable t, such that, $t = 0, ..., k-1$. Q_t is equal to $Q_{temp} \cup Q_{aux}$. The function *k*-robust-QANS repeatedly invokes the function QAN-set until it is not possible to find a new subset Q_{aux} that covers all the nodes in $N_2(v) \setminus PoorlyCovered$ or a maximum of $k+1$ disjoint subsets has been found.

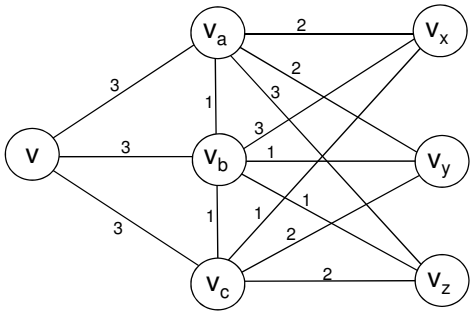


Fig. 2. A network with links labelled with the bandwidth metric.

Our goal is to advertise the best links to reach all nodes in $N_2(v)$. The core of our heuristic is function QAN-set. For instance, let us consider Figure 2 and execute function *k*-robust-QANS with k equal to one. In this example, consider the bandwidth as the routing metric and no poorly covered nodes. We analyze the second phase of our heuristic to construct $QANS_{k+1}(v)$. Q_1 is constructed by invoking function QAN-set with t = 0. Node v_b is the best node to cover v_x. $Q_1 = \{v_b\}$ and $Cov(v, v_b) = \{v_x\}$. In each step, remove the best link to reach a node in $N_2(v)$, i.e., remove the link (v_b, v_x), line 7. There exist two options to cover v_y. Select v_a. Thus, $Q_1 = \{v_b, v_a\}$ and $Cov(v, v_a) = \{v_y\}$. Remove the link (v_a, v_y). Finally, select v_a to cover v_z. Thus, $Q_1 = \{v_b, v_a\}$ and $Cov(v, v_a) = \{v_y, v_z\}$. Remove the link (v_a, v_z). Now compute Q_2 with t = 1. Node v_a is the best node to cover v_x. Then, $Q_2 = \{v_a\}$ and $Cov(v, v_a) = \{v_y, v_z, v_x\}$. Remove the link (v_a, v_x). Select v_c to cover v_y. $Q_2 = \{v_a, v_c\}$ and $Cov(v, v_c) = \{v_y\}$. Remove the link (v_c, v_y). Select v_c to cover v_z. Finally, $Q_2 = \{v_a, v_c\}$ and $Cov(v, v_c) = \{v_y, v_z\}$. Remove the link (v_c, v_z). We obtain a $\{v_a, v_b, v_c\}$ as a $QANS_{k+1}(v)$. We guarantee the advertisement of at least two links to reach a node in $N_2(v)$. The set $Cov(v, v_a)$ indicates that v_a should generate TC messages including the links to nodes: v, v_x, v_y and v_z. Our function reduces the size of the TC messages and improves the mechanism proposed in RFC3626 [1] to advertise redundant information, i.e., TCR parameter.

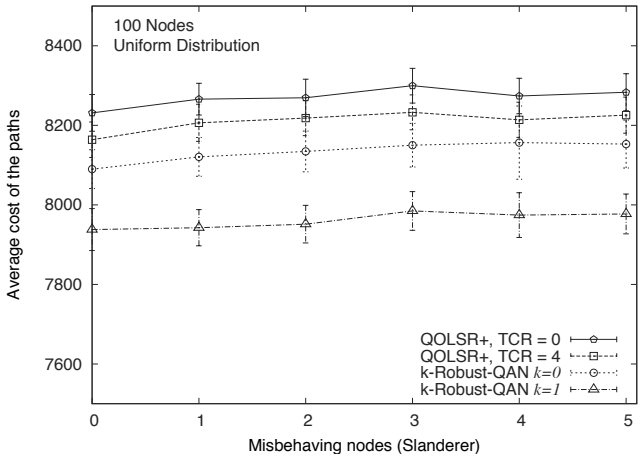
B. TC_Redundancy parameter

In OLSR, given a node with a non-empty selector set, the parameter TCR specifies the amount of information that may be included in the TC messages. The parameter is defined in RFC3626 (Section 15.1) [1] but does not consider the case where the MPR set and the QANS are different. We interpret the parameter considering different QAN and MPR sets. If TCR is equal to zero, then the advertised link set is limited to links to nodes in the QANS selector set. If TCR is equal to one, then the advertised link set includes the links to nodes in the QANS and QANS selector set. if TCR is equal to two, then the advertised link set also includes links to nodes in the MPR selectors set. If the TCR is equal to three, then the advertised link set also includes links to nodes in the MPR set. If TCR is equal to four, then the advertised link set also includes links to all one-hop neighbors. However, the amount of information in every TC messsage increases considerably. Following the example in Figure 2, the link (v_c, v_z) is advertised only if either v_z selects v_c as a QAN or v_c sets its TCR parameter equal to four. In this case, the $|TC_{v_c}|$ is always equal to five and $|TC_{v_b}|$ is equal to six. After executing the heuristic *k*-robust-QANS, v_c generates TC messages advertising only the links to the nodes: v, v_y and v_z. The node v_b only advertises the links to nodes v and v_x.

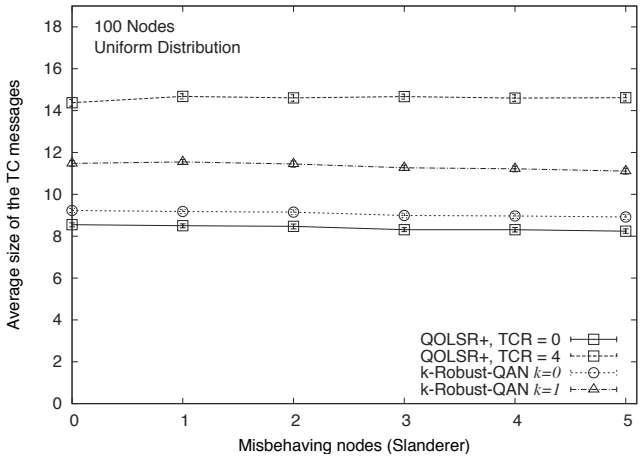
VI. Simulations and Results

In this section, we describe the experiments we conducted to measure the effectiveness of our proposed function *k*-robust-QANS and the results we obtained. We assume that the nodes have the same characteristics. Every node has just one interface. All the links are bidirectional. All the nodes have the same willingness to generate and forward traffic on behalf of other nodes, except for those that have been selected as misbehaving nodes. We conducted our experiments using the NS-3 simulator [15], version 3.9. We modified the original OLSR code developed by Ros and Carneiro to implement the functions described in Sections II-B and V-A. The routing table computation process is based on Dijkstra's algorithm. The complexity of function *k*-robust-QANS is $O(n^2)$. The nodes were distributed following a uniform distribution in an area of 1000 m by 1000 m. The malicious nodes are selected randomly. They do not collude. Every node has a transmission range of 250 m. No data traffic is generated. All scenarios are static.

Figs. 3(a) and 3(b), depict our results with 95% confidence intervals. We tested the function *k*-robust-QANS and QOLSR+ in 100 scenarios with 100 nodes each during 50 seconds. We selected an adversary node that generates partial link state information, i.e., a slanderer node. We use $|N_1(v_i)|$ as a routing metric. The value of a link $R(v_i, v_{i+1})$ is equal to the sum of the number of one hop neighbors of each end point, i.e., $R(v_i, v_{i+1}) = |N_1(v_i)| + |N_1(v_{i+1})|$. Every node uses Dijsktra's algorithm to obtain the cost to reach any other node in the network represented by the graph G. Given a node v_i, $W_n(v_i)$ is the sum of the costs of the paths to reach the other $n-1$ nodes in the network, i.e. $W_n(v_i) = \sum_{j=1}^{n-1} W(v_i, v_j)$. We take the average of the sum of $W_n(v_i)$ obtained by every node, i.e., $W_{avg}(G) = \sum_{i=1}^{n} W_n(v_i)/n$. We aim to minimize $W_{avg}(G)$ by adverting the best links in the network. Fig. 3(a) depicts the average of the sum of $W_n(v_i)$ obtained by every node in the network, i.e., $W_{avg}(G)$ We tested the function *k*-robust-QANS with k equal to zero and equal to one. QOLSR+ is also tested adding additional redundancy to the TC messages. A node with a non empty MPR selector set advertises links to all one-hop neighbors, i.e., TCR equal to four. In Fig. 3(a), the value of $W_{avg}(G)$ slightly grows when the number of misbehaving nodes increases. This is because the misbehaving nodes prevent other nodes from receiving important information about the best quality links in the network. Fig. 3(a) shows the effect of the misbehaving node when security is not considered, i.e., QOLSR+ with TCR equals 0. The function *k*-robust-QANS mitigates the effect of the attack and reduces the value of $W_{avg}(G)$ as depicted



(a) Total of the paths between any pair of nodes.



(b) Average size of the TC messages.

Fig. 3. Experiments results.

in Fig. 3(a). Fig. 3(b) shows the average size of the TC messages. The sizes of the messages are similar when the function *k*-robust-QANS (k equal to zero) and QOLSR+ (TCR equal to zero) are applied. However, the function *k*-robust-QANS reduces the value of $W_{avg}(G)$ as shown in Fig. 3(a). If QOLSR+ is applied with TCR equal to two, then the size of the messages and the amount of information processed increase considerably. According to our results in Fig. 3(a), there is no significantly improvement when TCR equals four. The average size of the messages also increases when the function *k*-robust-QANS is applied with k equal to one, however, our strategy minimizes $W_{avg}(G)$ more effectively.

VII. Conclusions

In this paper, we studied security issues in OLSR with QoS. We presented the function *k*-robust-QANS. Given a node v, our solution constructs the set $QANS_{k+1}(v)$ to advertise, when possible, $k+1$ sets of links to reach any two-hop neighbor. The nodes, selected as part of $QANS_{k+1}(v)$, generate TC messages including the links to the two-hop nodes for which they were selected. We also presented the unadvertised quality links problem when QoS is considered. Comparing to QOLSR+, our mechanism improves the network topology view and handles eventual attacks to the link advertisement process. A series of simulations show that the function *k*-robust-QANS reduces the total cost to reach any other node in the network and reduces the amount of information included in every TC message.

Acknowledgment

The authors graciously acknowledge the financial support received from the following organizations: Natural Sciences and Engineering Research Council of Canada (NSERC), the Spanish Ministry of Science and Innovation (projects TSI2007-65406-C03-03 E-AEGIS, TIN2011-27076-C03-02 CO-PRIVACY, CONSOLIDER INGENIO 2010 CSD2007-0004 ARES, and TIN2010-15764 N-KHRONOUS), Ministry of Education of Mexico (SEP-PROMEP, Program for Academic Improvement) and Universidad Tecnológica Metropolitana (UTM).

References

[1] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR), RFC3626," IETF Internet Draft, http://www.ietf.org/rfc/rfc3626.txt, United States, October 2003.

[2] H. Badis and K. Al Agha, "QOLSR multi-path routing for mobile ad hoc networks based on multiple metrics: bandwidth and delay," in *Vehicular Technology Conference, 2004. VTC 2004-Spring. 2004 IEEE 59th*, vol. 4, May 2004, pp. 2181–2184.

[3] P. Sondi, "Le Routage à qualité de service dans les réseaux mobiles ad hoc," Ph.D. dissertation, Université de Valenciennes et du Hainaut-Cambrésis, Décembre 2010.

[4] H. Badis, A. Munaretto, K. Al Aghal, and G. Pujolle, "Optimal path selection in a link state QoS routing protocol," in *Vehicular Technology Conference. VTC 2004-Spring. IEEE 59th*, vol. 5, 2004, pp. 2570–2574.

[5] A. Munaretto and M. Fonseca, "Routing and quality of service support for mobile ad hoc networks," *Computer Networks*, vol. 51, no. 11, pp. 3142–3156, 2007.

[6] L. Moraru and D. Simplot-Ryl, "QoS preserving topology advertising reduction for OLSR routing protocol for mobile ad hoc networks," in *WONS 2006 : Third Annual Conference on Wireless On-demand Network Systems and Services*. Les Ménuires (France): INRIA, INSA Lyon, IFIP, Alcatel, January 2006, pp. 196–202.

[7] F. Khadara, N. Mitton, and D. Simplot-Ryl, "Towards an efficient QoS based selection of neighbors in QOLSR," in *IEEE 30th International Conference on Distributed Computing Systems (ICDCS) Workshops*, Genoa, Italie, June 2010, pp. 152–160.

[8] G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis, "A multi-path routing strategy to prevent flooding disruption attacks in link state routing protocols for MANETs," *Journal of Network and Computer Applications (JNCA)*, vol. 36, no. 2, pp. 744 – 755, March 2013.

[9] T. Clausen, C. Dearlove, and P. Jacquet, "Optimized link state routing protocol version 2(OLSRv2), RFC3666 , Work in progress," Project Hipercom, INRIA, Internet Draft, http://bgp.potaroo.net/ietf/all-ids/draft-ietf-manet-olsrv2-11.txt, United States, October 2010.

[10] L. Hanzo and R. Tafazolli, "A survey of QoS routing solutions for mobile ad hoc networks," *IEEE Communications Surveys & Tutorials*, vol. 9, no. 2, pp. 50–70, 2007.

[11] G. Santhi and A. Nachiappan, "A survey of QoS routing protocols for mobile ad hoc networks," *International journal of computer science & information Technology (IJCSIT)*, vol. 2, no. 4, pp. 125–132, 2010.

[12] R. Gujral and A. Kapil, "Secure qos enabled on-demand link-state multipath routing in manets," in *Information Processing and Management*, ser. Communications in Computer and Information Science, V. Das, R. Vijayakumar, N. Debnath, J. Stephen, N. Meghanathan, S. Sankaranarayanan, P. Thankachan, F. Gaol, and N. Thankachan, Eds. Springer Berlin Heidelberg, 2010, vol. 70, pp. 250–257.

[13] S. Subramaniam and R. Baskaran, "Secured and energy based qos routing in manets," *International Journal of Computer Science and Business Informatics*, vol. 1, no. 1, p. online, June 2013.

[14] S. Sedaghat, F. Adibniya, and V. Derhami, "A secure mechanism for QoS routing in mobile ad hoc networks with QoS requirements consideration," *Computational Intelligence and Communication Networks, International Conference on*, vol. 0, pp. 320–324, 2010.

[15] T. Henderson et. al., "The NS-3 network simulator." Software package retrieved from http://www.nsnam.org/, 2012.