Mitigation of topology control traffic attacks in OLSR networks

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

Mitigation of Topology Control Traffic Attacks in OLSR Networks

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

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

†Institut Telecom, Telecom Bretagne, Cesson-Sevigne, 35576, France
Email: joaquin.garcia-alfaro@acm.org

***Abstract*—The core of the Optimized Link State Routing (OLSR) protocol is the selection of Multipoint Relays (MPRs) as a flooding mechanism for distributing control traffic messages. A node in an OLSR network, selects its MPR set such that all two-hop neighbors are reachable through, at least, one MPR. However, if an MPR misbehaves during the execution of the protocol, the connectivity of the network is compromised. Additional coverage in the selection of the MPRs helps to mitigate the effect of control traffic attacks. RFC3626 defines the selection of MPRs with additional coverage. Nevertheless, the overhead of the network increases due to the added number of control traffic messages. In this paper, we propose an improved MPR selection with additional coverage. Every node selects, if it is possible, $k+1$ disjoint MPR sets. The union of those sets, is a k-robust-MPR set. Thus, given a node, alternative paths are created to reach any destination two-hops away. We test both approaches against two kinds of adversaries misbehaving during the execution of the protocol. Our proposed MPR selection with additional coverage mitigates the effect of control traffic attacks by offering equivalent protection compared to the MPR selection with extra coverage presented in RFC3626, but reducing the overhead generated by redundant control information.**

I. Introduction

The Optimized Link State Routing (OLSR) [4] protocol, is a proactive link state routing protocol for Mobile Ad hoc networks (MANETs). As many other routing protocols, OLSR did not include security constrains in its original design. The core of the protocol is the selection, by every node, of Multipoint Relays (MPRs) among their one-hop symmetric neighbors. The nodes selected as MPRs are responsible of generating and forwarding control traffic messages and used to form optimal routes from a given node to any destination in the network. Thus, if an MPR fails or misbehaves sending or forwarding control traffic information, the connectivity of the network is compromised. The standard OLSR specification (RFC3626, Section 20), enumerates security considerations for confidentiality, integrity and interaction with external routing domains, but does not include security measures. Our research focus on mitigating control traffic attacks in OLSR.

Selecting the MPR sets as small as possible ensures that the overhead of the protocol is kept at minimum. However, adding extra coverage in the selection of the MPRs is an alternative solution to mitigate the effect of control traffic attacks. Additional MPR coverage, ensures that reachability for a node is advertised by more nodes. The RFC3626 [4], Section 16, defines an MPR coverage parameter (MPR_Coverage) to specify by how many MPRs any strict two-hop node should be covered. We present this approach in function k-covered-MPR. Nevertheless, the overhead of the protocol increases considerably due to excessive control traffic messages.

We propose function k-robust-MPR, in order to improve the selection of MPRs with additional coverage. Thus every node selects, when it is possible, $k+1$ disjoint MPRs sets. The union of the $k+1$ disjoint sets is a k-robust-MPR set, if we remove a maximum of k elements from the MPR set of a given node n, all nodes two hops away from node n are still covered by the remaining elements in the k-robust-MPR set. We tested the two functions under the presence of two types of misbehaving nodes. One adversary interrupts the proper flooding of the control traffic messages, and the second one, generates them incorrectly. Our results show that our improved MPR selection offers a better balance between the protection against control traffic attacks and the overhead generated by the increased number of messages.

A. Control Traffic Attacks to OLSR

The generation and exchange of critical information are important vulnerability targets. The control messages flood the network to allow every node to create optimal paths to any destination in the network. If a node misbehaves by generating or forwarding incorrect control traffic information the integrity of the network is compromised. During the execution of the protocol, each node broadcasts Hello messages to advertise their presence among their one-hop neighbors, to learn about their two-hop neighbors and to select its MPRs. The MPRs generate and retransmit Topology Control(TC) Messages. The information from Hello and TC messages allows every node to construct their routing tables. Thus, in an OLSR network, the nodes have two principal tasks to perform [15]: (1) to generate correctly routing information (i.e., Hello and TC messages), and (2) to correctly relay traffic on behalf of other nodes in the network.

A target of an attack can be that legitimate nodes never

receive correct control traffic messages or store incorrect information to affect negatively the network topology. The attacks at the routing level can be classified in two categories [6]: incorrect traffic generation and incorrect traffic relaying.

- Incorrect traffic generation: a node misbehaves by generating incorrect Hello or TC messages under a false identity or generating control traffic messages reporting an incorrect set of links. An attacker can either hide valid links or insert non-existing links. In all cases, the network connectivity is disrupted.
- Incorrect traffic relaying: if a node decides to drop valid packets, e.g., an MPR refuses to forward TC messages, the network will experience a degradation of communication. Equally, an attacker misbehaves by resending old valid control messages (not timestamped), or forwarding altered control messages. The nodes in the network will receive wrong information and update their routing tables with stale information.

Additionally, two or more nodes can collude to perform an attack. For instance, misbehaving nodes can establish invalid links and replay valid information in different regions of an OLSR network mounting a wormhole attack. This type of attack is particularly severe, because it is difficult to detect even in a network where security constraints (i.e., authentication, integrity and confidentiality) are implemented.

B. Contributions of the Paper

In this paper, we propose an improved MPR selection with additional coverage (cf., function k-robust-MPR) to mitigate the effect of traffic control message attacks. We compare our function against the MPR selection with additional coverage (cf., function k-covered-MPR) proposed in the RFC3626. We analyze the cost and benefit of functions k-covered-MPR and k-robust-MPR for the selection of MPR with extra coverage in the presence of misbehaving nodes in an OLSR network. We also present the correctness of both functions in Section IV-C. Our function k-robust-MPR, mitigates the effect of the adversaries performing control traffic attacks and affecting negatively the construction of the routing tables for every node in an OLSR network. We also compare functions k-covered-MPR and k-robust-MPR to measure the level of mitigation against two types of adversaries. The experiments show that it is possible to offer equivalent protection but reducing the overhead due to the excessive number of control traffic messages generated by function k-covered-MPR. Additionally, function k-robust-MPR increases the performance ratio between the number of nodes with complete routing tables and the increased number of TC messages.

C. Related Work

The vulnerabilities in the OLSR protocol have been studied extensively. In general, the aim of security mechanisms are integrity and service availability(fault-tolerance). We can classify measurements in two categories [1], [12], [15], [9], [3]: i) cryptographic mechanisms to avoid attacks such as impersonation, replay or modification attacks, and ii) Intrusion Detection Systems (IDS) to prevent altered information from an authenticated node. Nevertheless, cryptographic models are challenging because in MANETs they have no centralized authority and reputation models require additional computation, increase the traffic in the network and need certain period of time to detect a misbehaving node, and during that time, the integrity of the network is compromised. Our scheme focuses on service availability and security by implementing an improved MPR selection with additional coverage. In the literature, authors implement the MPR_Coverage parameter in terms of fault tolerance, load balancing, reliability and Quality-of-Service(QoS) [11], [18]. However, they do not analyze their approaches with a security perspective. An improvement of the OLSR flooding method presented in [2] proposes the transformation of MPRs into Connected Dominating Sets (CDS). The use of these CDS-based algorithms for the flooding of topological data as a replacement of the standard flooding procedure of OLSR is claimed by the authors as a promising trade-off between minimization and reliability. However, they do not address the use of an additional coverage in the selection of the MPR sets, for tuning the flooding process once the CDS has been built. In [16] authors provide a balance between selecting additional coverage using the MPR_Coverage parameter proposed in [4] and redundancy in the link-state information (TC_Redundancy parameter) to improve the QoS of the protocol. Nonetheless, they analyze their approach using the MPR with extra coverage proposed in [4] and without the presence of misbehaving nodes. In [14], authors prove that an MPR selection with additional coverage can be used to preserve k-connectivity in an OLSR network; based on that approach we improve the selection of MPRs with additional coverage to reduce the overhead generated for excessive control messages and to mitigate traffic control attacks.

Organization of the paper — Section II reviews the OLSR protocol and its vulnerabilities. Section III describes our adversaries model. Section IV presents the selection of MPRs with extra coverage described in the RFC3626 and our k-robust-MPR selection. Additionally, we present the correctness of functions k-covered-MPR and k-robust-MPR. Section V shows our experimental results.

II. OLSR AND ITS SECURITY FLAWS

This section presents an overview of the OLSR protocol and its security flaws. OLSR is a proactive routing protocol designed for MANETs. The core of the protocol is the selection, by every node, of Multipoint Relays (MPRs) among their one-hop symmetric neighbors. OLSR nodes flood the network with link-state information messages. The link-state information is constructed by every node and involves periodically sending Hello and Topology Control (TC) messages. This information is used to determine the best path to every destination in the network. Due to the proactive nature, the routes are immediately available when needed. The OLSR protocol is hop by hop routing, i.e., each routing table lists, for all

reachable destinations, the address of the next node along the path to that destination.

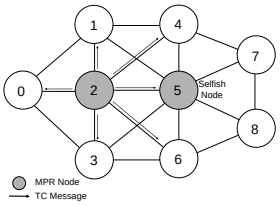
Hello messages are transmitted to one-hop neighbors. These messages are not retransmitted further. Every node uses the received information to learn about its one-hop and two-hop neighbors. That information allows to construct and to maintain neighbor tables. In the neighbor table, each node records the information about the one-hop neighbor link status (unidirectional, bidirectional or MPR). With this information every node builds its MPR selector set, i.e., the number of neighbors who have selected that node as their MPR. The MPRs are selected such that all two-hop neighbors are reachable through, at least, one MPR.

TC messages are broadcasted and retransmitted exclusively by the MPRs. These messages allow each node to construct its topology table and to declare its MPR Selector Set. The MPR Selector Set is the set of nodes that have selected a given node as an MPR. A node that has an empty MPR Selector Set does not send or retransmit any TC message. An MPR forwards a message if it comes from a node in its MPR Selector Set. Using the information of TC messages each node maintains a topology table where each entry consists of: (i) an identifier of a possible destination, i.e., an MPR selector in a TC message, (ii) an identifier of a last-hop node to that destination, i.e., the originator of the TC message, and (iii) an MPR Selector Set sequence number [8]. Routing tables are constructed using the information from the neighbor and topology table.

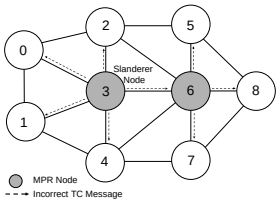
The selection and behavior of the MPRs are critical vulnerability targets. As many other routing protocols, OLSR did not include security measures in its design. A node has an incorrect behavior if it is either generating or relaying incorrect control messages in an OLSR network [6]. In the OLSR protocol, each node needs to construct and maintain routing tables. A valid entry in a routing table corresponds to a known path to another node in the network. MPRs are selected as intermediate nodes for all routes. Thus, all the destinations to routes partially constructed or broken are excluded from the routing table. Any failure during the control traffic relaying [5] affects the link-state information and integrity of the network. For instance, if a node selected as an MPR is misbehaving and decides not to retransmit TC messages (e.g., to save energy) compromises the proper construction of routing tables for the entire network.

III. Adversary Model

In this section, we describe the characteristics of our adversaries. Flooding the network with TC messages allows nodes to construct a path to all destinations. The TC messages are generated exclusively by the MPRs, each MPR includes in the messages all its selector nodes. Thus, the receptor of the message learns that the original sender is the last-hop to all the nodes included in the message. If an MPR misbehaves generating or forwarding TC messages, then the connectivity of the entire network is compromised. In [6], authors define node profiles according to their behavior in a network during the execution of the OLSR protocol. We select the *Selfish* and *Slanderer* node profiles as active attackers.



(a) Selfish attack.



(b) Slanderer attack.

Fig. 1. Example of a selfish and a slanderer attack. In (a), node 5 is a selfish node and has been selected by node 2 as an MPR. In (b), node 3 is a slanderer node and has been selected by node 1 as an MPR.

- A *Selfish* node, is an MPR that decides neither to generate nor to retransmit TC messages. Figure 1(a) is an example of a selfish attack. For instance, consider node 2 as an MPR of node 0, and node 5 an MPR of node 2. If node 2 broadcasts a TC message, then node 5 might be responsible to retransmit the message but may decide not to do so. In consequence, nodes 7 and 8 will never learn that the last-hop to reach node 0 is node 2.
- A *Slanderer* node, is an MPR that generates incorrect information, i.e., an MPR that does not declare a complete MPR Selector Set. Figure 1(b), is an example of a slanderer attack. Node 3 has been selected by nodes 0,1,2 and 4 as an MPR. Node 3 is an attacker and generates a TC message but without including node 1 in the message. Node 6 receives and forwards the message. In consequence, node 8 will never be aware of the presence of node 1.

These misbehaving nodes affect the integrity and the proper construction of routing tables for each node in the network. The nodes can be isolated and will not compute a complete view of the network topology. We consider that a node's routing table is affected if at least one element for a valid path in the network is missing. In both cases, alternative paths between nodes can help to mitigate the attacks. For instance, in Figure 1(a) node 2 can select node 5 or nodes 4 and 6 as MPRs. Thus, node 2 can reach nodes 7 and 8 trough nodes 4 and 6. We analyze in the sequel two approaches for additional coverage in the selection of MPRs: the function presented in the RFC3626 [4] (cf., function k-covered-MPR), and an alternative selection of disjoint MPR sets (cf., function k-robust-MPR). The objectives are to minimize the overhead

generated by an increased number of TC messages in the network and to offer equivalent protection in the presence of misbehaving nodes.

IV. MPR Computation with Additional Coverage

This section describes the selection of MPRs with extra coverage presented in the RFC3626 [4] and our proposed function k-robust-MPR choosing disjoint groups of MPRs. In [4], additional coverage is defined as the ability of a node to select redundant MPRs. The selection of MPRs must be as small as possible to reduce the overhead due to flooding the network with TC messages. Nevertheless, additional coverage allows a node to advertise its presence to more nodes in the network. In this manner, extra coverage helps to maintain the integrity of the network in spite of the presence of misbehaving nodes during the execution of the OLSR protocol. However, the overhead generated by the increased number of TC messages reduces the performance of the network. This problem is addressed with our improved function k-robust-MPR (cf. subsection IV-B), which balances additional coverage and traffic overhead.

A. RFC3626's MPR Coverage Parameter

The RFC3626 [4] defines the MPR_Coverage parameter to specify by how many one-hop nodes any two-hop neighbors must be covered. If MPR_Coverage is equal to one then the overhead is kept at minimum and the function is equivalent to the MPR selection without additional coverage specified in [4], Section 8.3.1. If MPR_Coverage is equal to k, a node selects its MPR set such as any two-hop neighbor is covered by k one-hop neighbors, whenever possible. A poorly covered node, is a node in the two-hop neighborhood that cannot be covered by at least k nodes in the one-hop neighborhood. The MPR_Coverage parameter is local to every node in the network. Nodes with different values of MPR_Coverage may operate in a same network. Function k-covered-MPR describes the MPR selection with coverage k defined in [4], Section 16.1. To explain functions k-covered-MPR and k-robust-MPR, we will use the following notation:

- $d(n, u)$: number of hops between nodes n and u.
- $N_1(n) := \{n_1 : d(n, n_1) \leq 1\}$.
- $N_{\leq 2}(n) := \{n_2 : d(n, n_2) \leq 2\}$.
- $N_2(n) := N_{\leq 2}(n) \setminus N_1(n)$.
- degree(n,n_1): returns the number of nodes in $N_2(n)$ such that $N_1(n_1) \cap N_2(n) \neq \emptyset$, assuming that $n_1 \in N_1(n)$.
- $M : M$ is an MPR set for node $n \Leftrightarrow M \subseteq N_1(n)$ such that for every node $n_2 \in N_2(n)$, $N_1(n_2) \cap M \neq \emptyset$.
- reachability(n, n_1, A): returns the number of nodes in $N_2(n)$ such that $d(n_1, n_2) \leq 1$ and $N_1(n_2) \cap A = \emptyset$, assuming that $A \subseteq N_1(n)$ and $n_1 \in N_1(n) \setminus A$.
- required(n,A): returns a set B, such that $B \subseteq N_1(n)$, and for every $b \in B$, $d(b, a) \leq 1$, assuming that $a \in A$ and $A \subseteq N_2(n)$.
- linked(n, A): returns a set B, such that $B \subseteq N_2(n)$, and for every node $b \in B$, $N_1(b) \cap A \neq \emptyset$, assuming that $A \subseteq N_1(n)$.
- nextNode(n, A, B) : returns a node $a \in A$ such that a provides the largest degree(n, a) and reachability(n, a, B). We assume that $A, B \subseteq N_1(n)$ and $A \cap B = \emptyset$.
- PoorlyCovered: subset of $N_2(n)$ formed by every node $n_2 \in N_2(n)$ such that $|N_1(n_2) \cap N_1(n)| < k$.
- WellCovered: subset of $N_2(n)$ formed by every node $n_2 \in N_2(n)$ such that $|N_1(n_2) \cap N_1(n)| \geq k$.

Function k-covered-MPR, with respect to a given node n works as follows [4]:

1) First, we obtain the poorly covered nodes in $N_2(n)$. Then, we include in the MPR set M, the nodes in $N_1(n)$ that poorly cover nodes in $N_2(n)$.
2) We remove the poorly covered nodes from $N_2(n)$.
3) While there exist nodes in $N_2(n)$ not yet covered by at least k nodes in the MPR set:
   - We add to M the node n_1 in $N_1(n)$ not in the MPR set, that provides the largest reachability(n, n_1, M) and degree(n,n_1).
   - We eliminate all the nodes in $N_2(n)$ now covered by at least, k nodes in the MPR set.

For instance, according to the example presented in Figure 2, the MPR sets selected by all the nodes, derived in polynomial time by function k-covered-MPR, corresponds to column four in Table I.

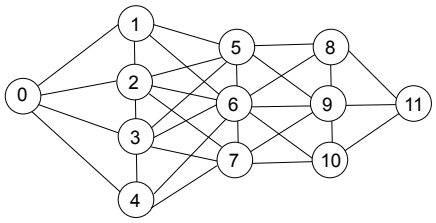


Fig. 2. Consider nodes 2, 3, 6 and 9 as MPRs, selected with no extra coverage.

Node	Possible disjoint MPR sets	No extra coverage	k-covered-MPR k = 2	k-robust-MPR k = 2
0	{{2},{3},{1,4}}	{2}	{2,3}	{2,3,1,4}
1	{{6}}	{6}	{6,5,2,0}	{6}
2	{{6},{7,5}}	{6}	{6,7,5}	{6,7,5}
3	{{6},{5,7}}	{6}	{6,5,7}	{6,7,5}
4	{{6}}	{6}	{6,7,3,0}	{6}
5	{{3,9},{6,2,8}}	{3,9}	{6,3,9,2,8}	{3,9,6,2,8}
6	{{2,9},{3,8},{1,10}}	{2,9}	{2,3,9,8}	{2,3,9,8}
7	{{2,9},{6,3,10}}	{2,9}	{6,2,9,3,10}	{2,9,6,3,10}
8	{{6}}	{6}	{6,5,9}	{6}
9	{{6}}	{6}	{6,5,7}	{6}
10	{{6}}	{6}	{6,7,9}	{6}
11	{{9}}	{9}	{9,8,10}	{9}

TABLE I
MPR computation examples in figure 2.

Function k-covered-MPR(n, k) $\rightarrow M$

1 $M \leftarrow \emptyset$;
2 WellCovered $\leftarrow$ k-covered(n,k);
3 PoorlyCovered $\leftarrow N_2(n) \setminus$ WellCovered;
4 $M \leftarrow$ required(n,PoorlyCovered);
5 **repeat**
6 **foreach** $n_2 \in$*WellCovered :* $|N_1(n_2) \cap M| \geq k$ **do**
7 WellCovered $\leftarrow$WellCovered $\setminus \{n_2\}$;
8 **if** *WellCovered* $\neq \emptyset$ **then**
9 $n_1 \leftarrow$ nextNode(n,$N_1(n) \setminus M$,M);
10 $M \leftarrow M \cup \{n_1\}$;
11 **until** *(WellCovered* = $\emptyset$*)*;
12 **return** M*;*

Function k-covered(n , k) $\rightarrow S$

1 $S \leftarrow \emptyset$;
2 **foreach** $(n_2 \in N_2(n))$ **do**
3 **if** $|N_1(n_2) \cap N_1(n)| \geq k$ **then**
4 $S \leftarrow S \cup \{n_2\}$;
5 **return** S;

Function k-robust-MPR(n, k) $\rightarrow M$

1 $M \leftarrow \emptyset$;
2 $i \leftarrow 0$;
3 $Remainder \leftarrow N_1(n)$;
4 **repeat**
5 $M_{aux} \leftarrow$MPR-set(n, $Remainder$, $N_2(n)$);
6 $M \leftarrow M \cup M_{aux}$;
7 $i \leftarrow i + 1$;
8 $Remainder \leftarrow Remainder \setminus M_{aux}$;
9 **until** $(M_{aux} = \emptyset$ ***or*** $i > k$ ***or*** $Remainder = \emptyset)$;
10 **return** M;

Function MPR-set(n, $Remainder$, S) $\rightarrow$ M_i

1 $M_i \leftarrow \emptyset$;
2 **if** $(|linked(n, Remainder)| = |S|)$ **then**
3 PoorlyCovered $\leftarrow N_2(n) \setminus$ k-covered(n,2);
 $M_i \leftarrow$ required(n, PoorlyCovered);
4 $S \leftarrow S \setminus$ linked(n,M_i);
5 **while** $(S \neq \emptyset)$ **do**
6 $Remainder \leftarrow Remainder \setminus M_i$;
7 $n_1 \leftarrow$ nextNode(n, $Remainder$, M_i);
8 $M_i \leftarrow M_i \cup \{n_1\}$;
9 $S \leftarrow S \setminus$ linked(n, M_i);
10 **return** M_i;

B. k-Robust-MPR Selection

In this section, we describe function k-robust-MPR that improves the selection of MPR sets with additional coverage. Function k-robust-MPR computes an MPR set that is composed of, at most, $k + 1$ disjoint groups, i.e., every two-hop node is covered, if possible, by $k + 1$ disjoint groups of one-hop neighbors. Function k-robust-MPR works as follows:

1) First, by invoking function MPR-set, we obtain a subset M_i such that M_i is subset of $N_1(n)$ and covers all the nodes in $N_2(n)$.
2) We repeat the function until it is not possible to find a new disjoint subset M_i that covers all the nodes in $N_2(n)$ or we have found a maximum of $k + 1$ disjoint subsets.
3) The MPR set is formed by the union, if it is possible, of k disjoint subsets M_i.

If we apply function k-robust-MPR to the example depicted by Figure 2, then we obtain the set of MPRs shown in Table I, column five. For example, the execution of function k-robust-MPR on node 0, with parameter k equal to two, returns the MPR set $\{2, 3, 1, 4\}$, which is 2-robust.

C. Correctness of the Functions

In this section, we demonstrate the correctness of functions k-covered-MPR and k-robust-MPR.

Lemma 1 *Let S be the set returned by applying function k-covered to node n. Every node in $N_2(n)$ covered by at least k nodes in $N_1(n)$, is in S.*

Proof: Suppose that if we apply the function k-covered to node n then it is possible to have a node n_2 in $N_2(n)$ such that $|N_1(n_2) \cap N_1(n)|$ is greater than or equal to k and n_2 is not in S. However, according to the definition of function k-covered, every node n_2 in $N_2(n)$ is inspected by the foreach-loop and, if $|N_1(n_2) \cap N_1(n)|$ is greater or equal to k, we add node n_2 to S in the body of the loop. Thus, S is equal to the union of every node in $N_2(n)$ covered by at least k nodes in $N_1(n)$. ■

Theorem 1 *Let M be the set obtained by applying function k-covered-MPR to node n. Every node in $N_2(n)$ with exactly k' neighbors in $N_1(n)$, such that k' is less than k, is covered by exactly k' nodes in M.*

Proof: Consider M as the set obtained by applying function k-covered-MPR to node n. Then, suppose that there exists a node n_2 in $N_2(n)$ with exactly k' neighbors in $N_1(n)$, k' less than k, that is not covered by k' nodes in M. However, by applying function k-covered-MPR, in line 2, and according to Lemma 1, we assign to the set WellCovered every node in $N_2(n)$ covered by at least k nodes in $N_1(n)$. In line 3, we assign to the PoorlyCovered set all the nodes in $N_2(n)$ not in the set WellCovered. Then in line 4, for every node n_2 in PoorlyCovered, we assign to M every node n_1 in $N_1(n)$ such that $N_1(n) \cap N_1(n_2)$ is not equal to the empty set. Therefore,

every node in $N_2(n)$ with k' neighbors in $N_1(n)$ is covered by exactly k' nodes in M. ■

Theorem 2 *Let M be the set obtained by applying function k-covered-MPR to node n. Then, every node in $N_2(n)$ with exactly k' neighbors in $N_1(n)$, such that k' is greater than or equal to k, is covered by, at least, k nodes in M.*

Proof: Consider M as the set with coverage k obtained by applying function k-covered-MPR to node n. Then, suppose that there exists a node n_2 in $N_2(n)$ with exactly k' neighbors in $N_1(n)$, k' greater than or equal to k, that is not covered by at least k nodes in M. However, in line 2, we assign to the set WellCovered all nodes in $N_2(n)$ covered by at least k nodes in $N_1(n)$. We assign to the PoorlyCovered set every node in $N_2(n)$ not in WellCovered. Thus, according to Theorem 1, we assign to M the nodes in $N_1(n)$ that poorly cover all nodes in the PoorlyCovered set. Then, we repeat the following procedure until every node in WellCovered is covered by, at least, k nodes in M:

- In lines 6 and 7, we eliminate every node in the set WellCovered covered by, at least, k nodes in M.
- If WellCovered is not equal to the empty set, then we add to M the node in $N_1(n)$ that covers the largest number of elements in $N_2(n)$.

Therefore, if k' is greater than or equal to k, every node in $N_2(n)$ with k' neighbors in $N_1(n)$, is covered by, at least, k nodes in M. ■

Lemma 2 *Let M_i be a set obtained by applying function MPR-set to node n, such that M_i is not equal to the empty set and $M_i \subseteq N_1(n)$. Then M_i successfully covers all nodes in $N_2(n)$ (i.e., M_i is a valid MPR candidate).*

Proof: Consider that after applying function MPR-set to node n with parameters *Remainder*, a subset of $N_1(n)$, and S, equal to $N_2(n)$, it is possible to obtain a set M_i not equal to the empty set, such that M_i does not cover all elements in $N_2(n)$. However, by applying function MPR-set to node n, if $|linked(n, Remainder)|$ is equal to $|S|$, then we can construct a set M_i, such that M_i covers all elements in S. In function MPR-set line 3, we add to M every node in $Remainder$ that is the only one to provide reachability to a node in S. After that, while there exist nodes in S that are not yet covered by, at least, one node in M_i, we select a new node n_1 from $Remainder$ not in M_i that covers the largest number of nodes in S. Thus, we obtain M_i such that $|linked(n, M_i)|$ is equal to $|S|$. Therefore, if M_i is not equal to the empty set, we can affirm that every node in S is covered by, at least, one node in M_i and so that every set M_i returned by function MPR-set is, indeed, a valid MPR set. ■

Theorem 3 *Let M be a set obtained by applying function k-robust-MPR to node n. Then M is a valid MPR set, i.e., M covers all the nodes in $N_2(n)$.*

Proof: Suppose that by applying function k-robust-MPR we obtain a set M, such that that M is not a valid MPR set. However, according to Lemma 2, if M_i is not equal to the empty set, then M_i is a valid MPR set. Notice that, function k-robust-MPR invokes function MPR-set $k'+1$ times, such that k' is greater than or equal to zero and less than or equal to k. Since M is the union of $k'+1$ disjoint valid MPR sets, we can affirm that M is a valid MPR set. ■

Corollary 1 *Let M be a valid MPR set obtained by applying function k-robust-MPR to node n. If a node in $N_2(n)$ is covered by k' nodes in $N_1(n)$, with k' less than or equal to k and there is no k'' in $N_1(n)$ such that k'' is less than k'. Then, it is possible to invoke a maximum of k' times function MPR-set and M is a k'-robust-MPR set.*

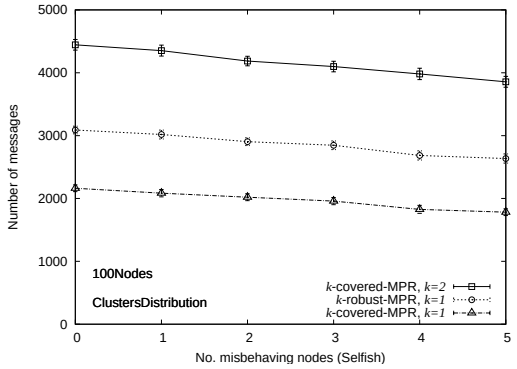
Proof: Suppose that there exists a node n_2 in $N_2(n)$ such that n_2 is covered by k' nodes in $N_1(n)$, then it is possible to invoke function MPR-set k times to obtain k-robust-MPR set. However, by applying function k-robust-MPR to node n, we invoke function MPR-set with parameters n, $Remainder$ and S. Initially, in line 3 we assign to $Remainder$ the set $N_1(n)$, and every time a valid M_i is obtained from function MPR-set, we subtract the set M_i from $Remainder$. Thus, if a node in $N_2(n)$ is covered by k' nodes in $N_1(n)$, after invoking function MPR-set k' times, then there exists a node n_2 in $N_2(n)$ that is not covered in $Remainder$ and $|linked(n, Remainder)|$ is less than $|N_2(n)|$. Then, function MPR-set returns the empty set and M is equal to the union of k' valid disjoint M_i sets. ■

Theorem 4 *Let M be the set obtained by iteratively applying k'+1 times function MPR-set from function k-robust-MPR with parameters n and k, such that k' is greater than zero and less than or equal to k. Then, for any S subset of M of size k', the nodes in M not in S still cover all the nodes in $N_2(n)$.*

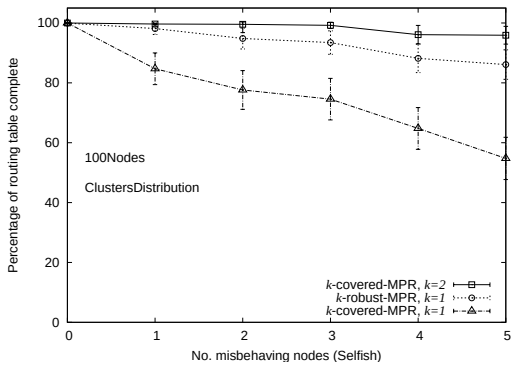
Proof: Suppose that M is a valid MPR set obtained by applying function k-robust-MPR to node n, and the elements in M not in S do not cover all nodes in $N_2(n)$. However, every M_i in M is obtained by applying k' times function MPR-set to node n. According to Corollary 1, we have $k'+1$ valid M_i sets in M. We consider the worst case when every element in S belongs to a different M_i and we have k' invalid M'_i sets in M. Consider M' equal to the union of k' invalid M'_i, i.e., M is equal to the union of $M_{k'+1}$ and M' such that, according to the definition of function MPR-set, $|linked(n, M_{k'+1})|$ is equal to $|N_2(n)|$. Then, $M_{k'+1}$ is a valid MPR set and the elements in M not in M' still cover all nodes in $N_2(n)$. Therefore, nodes in M not in S are a valid MPR set. ■

V. Experiments

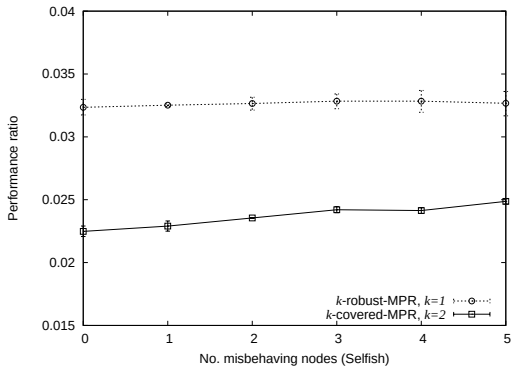
We conducted simulations to confirm that our k-robust-MPR set selection allows to minimize the effect of misbehaving nodes and helps to reduce the overhead generated by the k-covered-MPR function proposed in the standard OLSR protocol. To measure the effectiveness of our proposal, we count the number of nodes that were able to find a path to all nodes in the network after executing the two different MPR selection methods, for a certain period of time. Additionally, we take into account, the number of retransmissions during



(a) Number of Topology Control messages.

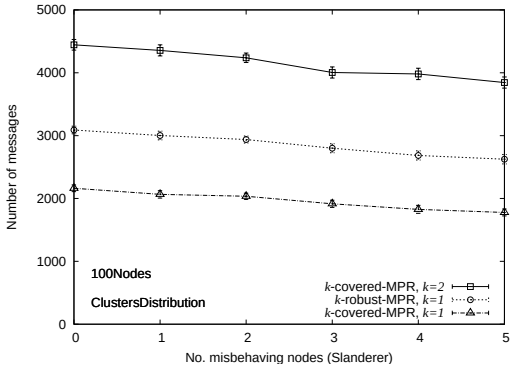


(b) Percentage of nodes with complete routing tables.

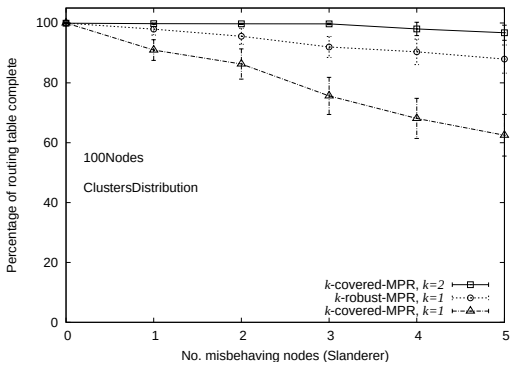


(c) Performance ratio.

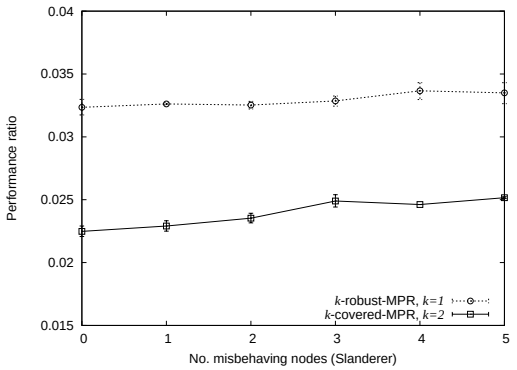
Fig. 3. Comparative of functions k-covered-MPR and k-robust-MPR under the presence of selfish nodes.



(a) Number of Topology Control messages.



(b) Percentage of nodes with complete routing tables.



(c) Performance ratio.

Fig. 4. Comparative of functions k-covered-MPR and k-robust-MPR under the presence of slanderer nodes.

the simulations. We conducted our simulations using the NS-2 simulator [10], version 2.29, with the UM-OLSR [13] package. We modified the original UM-OLSR code to implement the k-robust and k-covered MPR functions.

The RFC3626 considers that nodes with different characteristics can coexist in an OLSR network. For our experiments, we assume that all the nodes have the same characteristics, every node has just one interface and all the links between the nodes are bidirectional. Additionally, all the nodes have the same willingness to carry and forward traffic on behalf of other nodes, except for those that have been selected as misbehaving nodes. The misbehaving nodes do not collude to perform an attack. Table II describes the parameters for our simulations in NS-2. In our experiments, no data traffic is generated and all the scenarios are static.

We test functions k-robust-MPR and k-covered-MPR (with and without additional coverage), against one to five selfish or slanderer nodes. We test one hundred different topologies and one hundred nodes in each case. For each topology, we test all the approaches. In all the topologies, the misbehaving nodes are selected at random among all the MPRs. For our experiments, the nodes are distributed in ten clusters. In [17]

Simulator Parameters	
Propagation model	TwoRayGround
Network type	IEEE 802.11 (2Mbps)
Area	1000m x 1000m square
Transmission Range	200 mts
Number of nodes	50, 100
Nodes' Distribution	Clusters
% Misbehaving nodes	1 - 5
Coverage	k=1,2
Simulations Time	15 sec
Adversaries	Selfish and Slanderer nodes

TABLE II
NS-2 AND SCENARIO PARAMETERS.

authors organize an OLSR network in clusters in a hierarchical architecture. We can consider our topology as a particular set of clusters at the same level. The nodes in each cluster follow a Zipf [7] distribution. Then, following this distribution the nodes are located in the center of each cluster with higher probability. Thus, it is possible to obtain topologies where at least 40% of the nodes have k-robust MPR sets, with $k \geq 1$.

Figure 3 depicts the average number of nodes with complete routing tables and 95% confidence intervals. It shows how our strategy offers additional protection to mitigate the effect of misbehaving nodes in contrast with the selection of MPRs without additional coverage. We point out that is not always possible to find k-robust MPR sets for all the nodes in the network. In consequence, if the number of misbehaving nodes increase the level of protection decrease. Notice that our k-robust-MPR function mitigates the effect of misbehaving nodes with a better performance than the k-covered-MPR (cf. Figure 3(c)). Figure 4 depicts our results, but with one to five slanderer nodes. Again, we observe that our k-robust-MPR function mitigates the effect of misbehaving nodes with a better performance than the k-covered-MPR (cf. Figure 4(c)).

VI. CONCLUSION

In this paper, we proposed an improved MPR selection with additional coverage (cf., function k-robust-MPR) to mitigate control traffic attacks in an OLSR network. Our goal is to provide service availability and security. In our proposal, every node selects, if it is possible, $k+1$ disjoint MPR sets. As a result, we obtain a k-robust-MPR set. The RFC3626 defines an MPR selection with additional coverage (cf., function k-covered-MPR), however, the number of topology control messages increases considerably reducing the performance of the network. We compared functions k-covered-MPR and k-robust-MPR in the presence of misbehaving nodes. We measured the number of nodes with complete routing tables after the execution of the OLSR protocol. Our experiments show that our function k-robust-MPR reduces the amount of traffic generated by function k-covered-MPR, and offers equivalent protection against control traffic attacks. Additionally, our function k-robust-MPR increases the performance ratio of the number of nodes with complete routing tables over the number of topology control messages.

Acknowledgments — Research supported by: Natural Sciences and Engineering Research Council of Canada (NSERC), Mathematics of Information Technology and Complex Systems (MITACS), Institut TELECOM (programme Futur et Ruptures), Spanish Ministry of Science (grants TSI2007-65406-C03-03 E-AEGIS and CONSOLIDERINGENIO 2010 CSD2007-00004 ARES), National Council of Science and Technology (CONACYT) , and Ministry of Education of Mexico (SEP, Program for Academic Improvement).

REFERENCES

[1] C. Adjih, T. Clausen, A. Laouiti, P. Muhlethaler, and D. Raffo. Securing the OLSR routing protocol with or without compromised nodes in the network. Technical Report, INRIA RR-5494, HIPERCOM project, INRIA Rocquencourt, February 2005.

[2] C. Adjih and L. Viennot. Computing connected dominated sets with multipoint relays. *Ad Hoc and Wireless Sensors Networks*, 1:27 – 39, March 2005.

[3] A. Adnane, R.T. de Sousa Jr., C. Bidan, and L. Mé. Autonomic trust reasoning enables misbehavior detection in OLSR. In *SAC'08: Proceedings of the 2008 ACM symposium on Applied computing*, pages 2006–2013, New York, NY, USA, 2008. ACM.

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

[5] F. Cuppens, N. Cuppens-Boulahia, S. Nuon, and T. Ramard. Property based intrusion detection to secure OLSR. In *ICWMC '07: Proceedings of the Third International Conference on Wireless and Mobile Communications*, page 52, Washington, DC, USA, 2007. IEEE Computer Society.

[6] F. Cuppens, N. Cuppens-Boulahia, T. Ramard, and J. Thomas. Misbehaviors detection to ensure availability in OLSR. In *MSN, Mobile Sensor Networks*, volume 4864 of *Lecture Notes in Computer Science*, pages 799–813. Springer, 2007.

[7] L. Devroye. *Non-uniform random variate generation*. Springer-Verlag, New York, 1986.

[8] P. Jacquet, P. Muhlethaler, T. Clausen, A. Laouiti, A. Qayyum, and L. Viennot. Optimized link state routing protocol for ad hoc networks. In *IEEE International Multi Topic Conference, 2001. IEEE INMIC 2001. Technology for the 21st Century. Proceedings*, pages 62–68. Lahore University of Management Sciences, Pakistan, December 2001.

[9] B. Kannhavong, H. Nakayama, N. Kato, A. Jamalipour, and Y. Nemoto. A study of a routing attack in OLSR-based mobile ad hoc networks. *International Journal of Communication Systems*, 20(11):1245–1261, 2007.

[10] S. McCanne, S. Floyd, K. Fall, K. Varadhan, et al. The network simulator: NS-2. Software package retrieved from http://www.isi.edu/nsnam/ns/, 1997.

[11] S. Mueller, R. P. Tsang, and D. Ghosal. Multipath routing in mobile ad hoc networks: issues and challenges. In *Performance Tools and Applications to Networked Systems, volume 2965 of LNCS*, pages 209–234. Springer-Verlag, 2004.

[12] D. Raffo, C. Adjih, T. Clausen, and P. Muhlethaler. Securing OLSR using node locations. In *Proceedings of 2005 European Wireless (EW 2005)*, pages 437–443, Nicosia, Cyprus, April 10-13 2005.

[13] F.J. Ros. UM-OLSR, an implementation of the OLSR (optimized link state routing) protocol for the NS-2 network simulator. Software package retrieved from http://masimum. inf. um. es/um-olsr/html, 2007.

[14] L. Viennot and P. Jacquet. Bi-connectivity, k-Connectivity and multipoint relays. Research Report RR-6169, INRIA, 2007.

[15] J.P. Vilela and J. Barros. A feedback reputation mechanism to secure the optimized link state routing protocol. In *IEEE Communications International Conference on Security and Privacy for Emerging Areas in Communication Networks (Securecomm 2007). Los Alamitos (2007)*, pages 294–303, 2007.

[16] P.E. Villanueva, T. Kunz, and P. Dhakal. Extending network knowledge: making OLSR a quality of service conducive protocol. In *IWCMC '06: Proceedings of the 2006 International Conference on Wireless Communications and Mobile Computing*, pages 103–108, New York, NY, USA, 2006.

[17] L. Villasenor-Gonzalez, Y. Ge, and L. Lament. HOLSR: a hierarchical proactive routing mechanism for mobile ad hoc networks. *IEEE Communications Magazine*, 43(7):118–125, July 2005.

[18] J. Yi, E. Cizeron, S. Hamma, and B. Parrein. Simulation and performance analysis of MP-OLSR for mobile ad hoc networks. In *IEEE Wireless Communications and Networking Conference, IEEE WCNC*, Las Vegas, March 31-April 3 2008.