Chapter 6
Security Issues in Link State Routing Protocols for MANETs

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

Abstract In link state routing networks, every node has to construct a topological map through the generation and exchange of routing information. Nevertheless, if a node misbehaves then the connectivity in the network is compromised. The proactive Optimized Link State Routing (OLSR) protocol has been designed exclusively for Mobile Ad Hoc Networks (MANETs). The core of the protocol is the selection of Multipoint Relays (MPRs) as an improved flooding mechanism for distributing link state information. This mechanism limits the size and number of control traffic messages. As for several other routing protocols for MANETs, OLSR does not include security measures in its original design. Besides, OLSR has been extended to address a number of problems in MANETs. For example, Hierarchical OLSR (HOLSR) has been proposed to address scalability and Multipath OLSR (MP-OLSR) to address fault tolerance. However, these OLSR extensions can be affected either by inheriting or adding new security threats. In this chapter, we present a review of security issues and countermeasures in link state routing protocols for MANETs.

6.1 Introduction

The design of a secure and efficient routing protocol for Mobile Adhoc Networks (MANETs) is a challenging problem. Routing protocols proposed for MANETs assume a trusted and cooperative environment. Therefore, several mechanisms to

G. Cervera (✉) • M. Barbeau • E. Kranakis
School of Computer Science, Carleton University, Ottawa, ON, Canada, K1S 5B6
e-mail: gcevia@scs.carleton.ca; barbeau@scs.carleton.ca; kranakis@scs.carleton.ca

J. Garcia-Alfaro
Institut Telecom, Telecom Bretagne Cesson-Sevigne, 35576, France
e-mail: joaquin.garcia-alfaro@acm.org

E. Kranakis (ed.), *Advances in Network Analysis and its Applications*, Mathematics in Industry 18, DOI 10.1007/978-3-642-30904-5_6,

enhance security in MANETs have been proposed. The proactive Optimized Link State Routing (OLSR) [12] protocol has been designed exclusively for MANETs. The core of the protocol is the concept of Multipoint Relay (MPR). A valid MPR set, is defined as a subset of one-hop neighbors, such that all two-hop neighbors are covered through at least one node in the MPR set. In OLSR, every node has to select a valid MPR set. This mechanism allows to flood the network with control traffic information. OLSR comprises Hello and Topology Control (TC) messages. Every node periodically generates Hello messages. Within each Hello message a node reports its one-hop neighbors. Receiver nodes learn about its one and two hop neighbors. TC messages are used to discover nodes at more than two hops away. TC messages are generated and retransmitted exclusively by the MPRs. Unlike other link state routing protocols (e.g., OSPF [28]), the MPRs report partial link state information. Therefore, the MPR mechanism reduces the size and amount of control traffic information flooded in the network.

OLSR is defined in RFC 3626 [12]. A second version of the protocol, i.e., OLSRv2, is presented by Clausen et al. as an Internet-Draft in [13]. OLSRv2 implements the same basic mechanisms and algorithms for distributing control traffic (i.e., MPR-based flooding). As many other routing protocols for MANETs, OLSR and OLSRv2 are not secure by design. The selection of the MPRs and exchange of topology control information are important vulnerability targets. In this context, a malicious node is defined as a node that interrupts the flooding of control traffic information or does not obey the rules of the protocol. The terms: malicious, misbehaving, attacker and intruder are equivalent. Therefore, several authors proposed countermeasures to prevent or mitigate security threats in link state routing protocols for MANETs. For instance, in [2, 29, 30], Raffo et al., reviewed vulnerabilities in OLSR. In [18, 19], Clausen et al., studied security risk in OLSRv2. The authors proposed cryptographic mechanisms to enhance: integrity, confidentiality, reliability and service availability (fault-tolerance). Countermeasures to secure OLSR can be classified in two categories: cryptographic mechanisms to avoid impersonation or replay attacks, and Intrusion Detection Systems (IDS) [2] to prevent altered information from an authenticated node. Nevertheless, cryptographic models are challenging because in MANETs there is no centralized authority. The network performance drops due to additional computation. Reputation models or IDS mechanisms are designed to detect malicious behavior. Nevertheless, they increase the network traffic and need time to detect misbehaving nodes. Additionally, when a malicious behavior is detected, an efficient method to report untrusted nodes is needed. Moreover, *flooding disruption* [10] attacks can be perpetrated in networks with cryptographic capabilities. For instance, if a node refuses to retransmit TC messages on behalf of other nodes (e.g., to save energy), then the connectivity is disrupted.

In this chapter, we present a review of security issues in OLSR networks, existing solutions and our proposed countermeasures. In addition to OLSR, we review the Hierarchical OLSR (HOLSR) [34] protocol proposed by Villasenor et al. to address scalability and the Multipath OLSR (MP-OLSR) [37–40], proposed by Yi et al., to address security, fault tolerance and reliability. This chapter is based on the work presented in [9–11]. In [9], we analyzed the effect of control traffic attacks in OLSR networks and the selection of MPR sets with additional coverage to mitigate their effect. The MPR selection with additional coverage is presented in RFC 3626 [12], we name it k-Covered-MPR selection. However, additional coverage reduces the performance of the network due to additional control traffic information (i.e., TC messages). We proposed a k-Robust-MPR selection. In a k-Robust-MPR selection a node selects, when possible, $k+1$ disjoint MPR sets to guarantee that even if k of the selected MPR sets become invalid, the remaining set is still a valid MPR set. Our proposed MPR selection offers equivalent protection against control traffic attacks but reducing the overhead generated by additional control traffic information.

In [10], we presented a taxonomy of flooding disruption attacks and their effect in HOLSR networks. HOLSR uses TC messages for intra-cluster communications and implements Hierarchical TC (HTC) messages for inter-cluster communications. HOLSR implements the MPR flooding mechanism for distributing control traffic information. HTC messages are flooded exclusively by the MPRs. Therefore, the inter-cluster communications are also affected by flooding disruption attacks. In [10], we proposed to mitigate the effect of the attacks against HTC messages by selecting MPR sets with additional coverage (i.e., k-Robust-MPR and k-Covered-MPR selections). Additionally, the cluster formation phase in hierarchical OLSR networks can be disturbed. In [11], we presented an algorithm based on hash chains to enforce the cluster formation phase in HOLSR networks. In HOLSR, Cluster ID Announcement (CID) messages are implemented to organize the network in clusters. A misbehaving node may maliciously alter mutable fields (e.g., hop count) in CID messages to unbalance the distribution of nodes in clusters. Our solution allows a node to detect and discard invalid CID messages. Our algorithm can be implemented in other hierarchical approaches that use messages with mutable fields to organize the network in clusters. Finally, we analyze vulnerabilities in multipath OLSR-based networks. MP-OLSR is based on the MPR flooding mechanism to distribute control traffic information in the network. The construction of multiple paths in MP-OLSR has two phases: topology discovery and route computation. In the first phase, the nodes obtain information about the network topology through the exchange of Hello and TC messages. In the second phase, the nodes compute multiple paths to a particular destination in the network based on the information gathered during the first phase. These two phases are affected by flooding disruption attacks. Additionally, MPRs report partial link state information. Therefore, MP-OLSR nodes only acquire a partial view of the network. We analyze how the construction of multiple paths in MP-OLSR networks is affected by flooding disruption attacks and incomplete view of the network topology.

We describe different link state routing protocols for MANETs, their specific vulnerabilities and proposed countermeasures. The chapter is organized as follows: in Sect. 6.2, we review the OLSR protocol, flooding disruption attacks and related work. HOLSR, other OLSR-based hierarchical approaches and their vulnerabilities are described in Sect. 6.3, MP-OLSR and its security risks are presented in Sect. 6.4 and finally, Sect. 6.5 concludes the chapter.

6.2 Optimized Link State Routing (OLSR)

This section presents an overview of the OLSR protocol and its vulnerabilities. OLSR is a proactive routing protocol designed for MANETs. The core of the protocol is the selection, by every node, of MPRs among their one-hop neighbors. The MPR set is selected such that all two-hop neighbors are reachable through at least one MPR. Figure 6.1 compares the MPR mechanism and classical flooding. In Fig. 6.1a, control traffic information is retransmitted by all the one-hop neighbors. In Fig. 6.1b, control traffic information is retransmitted exclusively by the MPRs. This optimization improves the network performance by reducing the size and number of control traffic messages in the network. OLSR is defined in RFC3626 [12]. A second version of the protocol, i.e., OLSRv2, is presented by Clausen et al. in an Internet-Draft [13]. OLSRv2 uses and extends: the MANET Neighbor Discovery Protocol (NHDP) [16], RFC5444 – Generalized MANET Packet/Message Format [17], RFC5497 – Representing Multi-Value Time in MANETs [14] and RFC5148 – Jitter Considerations in MANETs [15] (optional). These protocols were all originally created as parts of OLSRv2, but have been specified separately for wider use. 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 simplifications.

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 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 based on hop by hop routing, i.e., each routing table lists, for each reachable destination, the address of the next node along the path to that destination. To construct a topology map, every node implements a topology discovery mechanism leveraging the periodic exchange of control traffic messages. Topology discovery includes: link sensing, neighbor detection and topology sensing. In the first phase, every node populates its local link information base (link set) and establishes communication with their symmetric neighbors, i.e., nodes with bidirectional communication. This phase is exclusively concerned with the OLSR interface addresses and ability to exchange packets between such OLSR interfaces. During the neighbor detection phase, every node

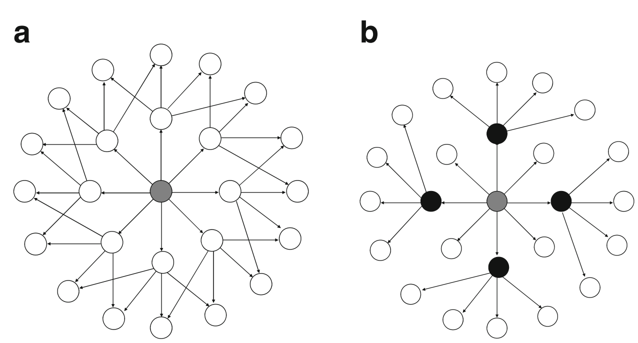


Fig. 6.1 MPR based mechanism against the classical flooding. Consider *gray nodes* as the originators of a TC message and *black nodes* as MPRs. (**a**) Classical flooding. (**b**) MPR mechanism

populates its neighborhood information base (i.e., one-hop and two-hop neighbor set). The link sensing and neighbor detection phases are based on the periodic exchange of Hello messages. Hello messages are solely transmitted to one-hop neighbors. In every Hello message, the nodes report their one-hop neighbors. This information allows every node to construct and maintain neighbor tables, as well as to select its MPR set. In the neighbor table, each node records the information about the one-hop neighbor link status (i.e., unidirectional, bidirectional or MPR), with this information every node builds its MPR selector set, i.e., the neighbors that selected that node as their MPR. OLSR detects and eliminates duplicate messages. OLSR keeps track of recently received messages by using a duplicate table. Therefore, when a message has been received and included in the duplicate table, the payload is not examined and the message is automatically discarded.

Topology sensing is achieved through the exchange of TC messages. TC messages are generated and retransmitted exclusively by the MPRs. TC messages have a Time-to-Live (TTL) field that is decremented every time an MPR retransmits the message. These messages allow each node to construct its topology table and to declare its MPR selector set. A TC message contains the MPR selector set of its originator. A node that has an empty MPR selector set does not send or retransmit any TC message. An MPR forwards a message only if it comes from a node in its MPR selector set (i.e., a source-dependant mechanism). This forwarding algorithm is defined in RFC 3626 [12]. Using the information from TC messages, each node maintains a topology table where each entry consists of:

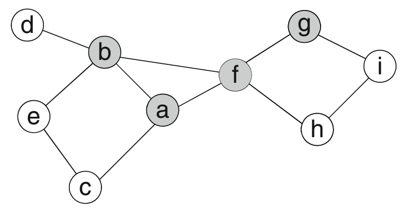
- An identifier of a possible destination, i.e., an MPR selector in a TC message,
- An identifier of a last-hop node to that destination, i.e., the originator of the TC message, and
- An MPR selector set sequence number [24].

It implies that a possible destination (i.e., an MPR selector) can be reached through the originator of the TC message. If there is an entry in the topology table whose last-hop address corresponds to the originator of a new TC message and the MPR selector set sequence number is greater than the sequence number in the received message, then the new message is discarded. Routing tables are constructed using the information from the one-hop neighbor, two-hop neighbor and topology tables.

OLSR implements two optional messages: Multiple Interface Declaration (MID) and Host and Network Association (HNA). They are exclusively retransmitted by the MPRs following the default forwarding algorithm defined in RFC 3626 [12]. MID messages are used to declare the presence of multiple interfaces on a node. HNA messages are employed to inject external routing information into an OLSR network and provide connectivity to nodes with non-OLSR interfaces (e.g., Internet). MID messages are implemented in a network with multiple interface nodes. Additional information is necessary in order to map interface addresses to main addresses. In OLSR, the main address is defined as the OLSR interface address. A node with multiple interfaces must generate periodically MID messages announcing all its interfaces to other nodes in the network. Thus, every node in an

Table 6.1 Summary of control traffic messages in OLSR networks. MID and HNA messages are optional

Messages	Generated by	Retransmitted by	Reported information
Hello	Every node	N/A	One-hop neighbors
TC	MPRs	MPRs	MPR selector set
MID	Nodes with more than one interface	MPRs	All available interfaces
HNA	Nodes with external access	MPRs	External routing information



MPRs	Selector Set
a	b,c,f
b	a,e,f,d
f	a,b,h,g
g	f,i,

Fig. 6.2 Example of an OLSR network

OLSR network will associate multiple interfaces to a node's main address. Nodes with just one interface do not generate MID messages and their main address is the OLSR interface address. A node with several interfaces, where only one of them is participating in an OLSR network must not generate MID messages. Upon receiving a MID message, the information is stored in an Interface Association table. This information is used to construct the routing tables. When a node misbehaves and does not retransmit TC, HNA or MID messages, the proper construction of the routing tables is compromised. Table 6.1 presents all the messages implemented in OLSR. In summary, the network topology discovery process is performed as follows:

1. First, every node periodically generates Hello messages to advertise itself and establish bidirectional links with its one-hop neighbors. Hello messages are not retransmitted. Figure 6.2 shows an example of an OLSR network. Node a includes nodes b, c and f in its one-hop neighbor set after exchanging Hello messages and establishing bidirectional links.
2. In subsequent Hello messages, every node reports its one-hop neighborhood. Receiver nodes identify their two-hop neighbors and compute their MPR set. In Fig. 6.2, nodes d, e, g and h are included in node a's two-hop neighbor table. Node a selects nodes b and f as its MPRs. Nodes a, b, f and g are selected as MPRs.
3. Nodes report their MPR set within their following Hello messages. If the receiver node was selected as an MPR, then it includes the sender node in its selector set, e.g., node b includes a in its selector set.
4. Nodes with a non empty selector set periodically generate TC messages advertising all nodes within their selector set. TC messages are retransmitted exclusively by the MPRs. To reach nodes more than two hops away, node a depends on the TC messages generated by all the MPRs. For instance, node g must periodically generate TC messages advertising its selector set, i.e., nodes f and i. TC messages generated by node g are retransmitted exclusively by nodes f, a and b.

5. When a node receives a TC message, it includes the contained information in its topology table. In Fig. 6.2, after receiving TC messages from node g, node a identifies node g as the last hop to reach node i. Note that node b receives TC messages from nodes a and f. However, node b stores the recently received TC messages in its duplicate table and discards future copies of the same message.
6. Finally, routing tables are constructed using information from the one-hop and two-hop neighbors and the topology table. Every node executes the Dijkstra's algorithm to obtain the shortest path to every other node more than two hops away. For instance, to reach node i, node a constructs a path trough nodes f and g. The shortest path to reach every other node in the network is always composed by MPRs. For example, to reach node d, node i constructs a path composed by nodes g, f and b.
7. Routing tables include the next node and number of hops to reach every other node in the network. Node i stores in its routing table only the next hop to reach node d (i.e., node g) and the number of hops (i.e., four hops). Thanks to the MPR mechanism, the nodes are aware of every other node in the network but some links are never advertised. For instance, node a never receives information about the link between nodes h and i, or between nodes e and c.
8. Optionally, a node with more than one interface generates MID messages. A node with access to an external network generates HNA messages. Information contained in MID and HNA messages is loaded in routing tables.

6.2.1 Related Work

As many other routing protocols for MANETs, OLSR is not secure by design. Vulnerabilities in OLSR have been studied extensively. For instance, in [2], Adjih et al. present security risks in the OLSR protocol and countermeasures based on cryptographic mechanisms to secure the protocol with or without compromised nodes in the network. The authors claim that an efficient securing mechanism should ensure the network integrity even when the network is subject to attacks that interrupt the connectivity. In [18, 22] Clausen and Herberg review security issues in OLSRv2. The authors analyze the basic algorithms that constitute the OLSRv2, and identify possible vulnerabilities and attacks.

Several authors have contributed with cryptographic mechanisms to secure OLSR. Cryptographic mechanism are proposed to enforce: integrity, authentication and confidentiality. Thus, public-key encryption is used for confidentiality, digital signature for integrity of the messages and digital certificates for authentication. However, the implementation of a Public Key Infrastructure (PKI) in MANETs is difficult due to the lack of a central authority (CA). Additionally, the efficient distribution of public and private keys is a challenging problem. Timestamps are implemented with digital signatures to assure the freshness of the message. However, time synchronization is difficult to achieve particularly in MANETs.

According to Adjih et al. [2], a *cryptographic capable* node is a node that has received valid keys to sign and verify messages. A misbehaving node can be also a cryptographic capable node. For example, in Fig. 6.2, node g may decide not to forward TC messages to node i or refuse to select an MPR set. In both cases, the connectivity of the network is compromised. Intrusion Detection Systems (IDS) are implemented to analyze malicious behavior in the network. However, once a misbehaving node has been detected, an efficient reputation model is needed to convey to other nodes the results observed by the IDS. In this chapter, we focus on attacks that prevent a node to acquire a complete network topology map. These attacks can be launched even in networks with cryptographic capabilities. In Sect. 6.2.2, we review them more precisely. In the following, we present some contributions to secure the OLSR protocol. We classify them in cryptographic mechanisms and IDS systems.

6.2.1.1 Cryptographic Mechanisms

In this section, we describe proposed solution based on cryptographic mechanisms. In [19], Clausen et al. present a digital signature mechanism for authentication and authorization in OLSRv2. The authors introduce the concept of admittance control for OLSRv2 networks and suggest a security extension based on digital signatures. They compare several standard digital signature algorithms such as: RSA, DSA, ECDSA and HMAC. The goal is to enable trusted nodes and to disable non-trusted nodes from participating in the control message exchange between routers, thereby providing a mode-of-operation similar to traditional mechanism employed for preserving network integrity in routed networks. Additionally, a performance study of the propose extension is presented to quantify the impact of increased control traffic overhead and increased message generation as well as processing time. The authors observed that HMAC requires significantly less time than ECDSA, DSA and RSA for generating a message signature. For the verification of a message signature, HMAC likewise spends substantially less time than ECDSA and DSA, whereas RSA is close to HMAC. Verification of RSA signatures has much greater overhead but is faster than both ECDSA and DSA.

In [30], Raffo et al., examined security issues related to the OLSR protocol, and enumerate a number of possible attacks against the integrity of the OLSR routing infrastructure. In particular, authors study attacks when a mechanism of digitally signed routing messages is deployed and an attacker may have taken control over trusted nodes. Their solution is based on inclusion of the geographical position of the sending node in control messages and on evaluation of reliability of links; this is accomplished using a GPS device and a directional antenna embedded in each node. Signatures with timestamps are sufficient to thwart attacks such as incorrect traffic generation and incorrect traffic relaying, when only legitimate nodes can sign control packets. Adding the node location in signature messages allows the network to avoid wormhole attacks and false messages generated by misbehaving nodes.

Raffo also presented in his Ph.D. thesis [29], a classification of possible attacks in OLSR networks. The author proposed a security architecture based on digital signatures. Additionally, the author proposed other techniques such as: reuse of previous topology information to validate the actual link state, cross-check of advertised routing control data with the node's geographical position, and intra-network misbehavior detection and elimination via flow coherence control or passive listening. Countermeasures in case of compromised nodes are also considered. Furthermore, the author assesses practical problems concerning the choice of a suitable symmetric or asymmetric cipher, alternatives for the algorithm of cryptographic key distribution, and the selection of a method for signature time stamping. In summary, the author presented an outline of different signature algorithms. The author suggested the study and design of better cryptographic algorithms, i.e., algorithms that use a smaller signature size to reduce computation complexity would increase the suitability of his proposed OLSR security architectures.

In [25], Khakpour et al., aboarded the access control problem in MANETs. The authors proposed a hierarchical distributed AAA (Authentication, Authorization, and Accounting) architecture for proactive link state routing protocols. This proposal contains a lightweight and secure design of an overlay authentication and authorization paradigm for mobile nodes as well as a reliable accounting system to enable operators to charge nodes based on their connection time. The authors also suggest a hierarchical distributed AAA server architecture with a resource and location aware election mechanism. Moreover, this proposal mitigates the OLSR security issues and eventually defines a node priority-based quality of service. The design of the architecture targets a minimum signaling overhead as well as calculation cost. In fact, different tasks are fairly distributed among distributed AAA servers. The calculation cost and overhead signaling is trivial compared to OLSR signaling and routing computations.

6.2.1.2 Intrusion Detection Systems

In this section we describe proposed solutions based on Intrusion Detection Systems (IDS). In [1], Abdellaoui and Robert, proposed the SU-OLSR protocol (SU for suspicious) to prevent attacks against OLSR-based routing protocols. In SU-OLSR the MPR selection is based on the trustworthiness of nodes. A malicious node might force its neighbors to choose it as an MPR node. Hence, a node should never select a neighbor as an MPR node if it behaves suspiciously and shows specific characteristics which would influence the MPR selection. Authors also show that to compute optimal paths, the optimality should not depend only on the length of a path but also whether or not it goes through fully or partially trusted MPR nodes. In [3], Adnane et al., 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 validate its local view of the global network topology. In their approach, when an inconsistency is detected between any received messages and its local view, the reasoning node is able to identify the compromised route. Their approach does not require any modification of the bare OLSR, but only the integration of the trust reasoning model on each node. Wu et al. present in [36] an overview of attacks according to the protocol layers, security attributes and mechanisms. Additionally, they present preventive approaches following the order of the layered protocol layers and an overview of reactive approaches based on IDS mechanism for MANET as a second line of defense to thwart attacks.

Vilela et al., present in [33] a feedback reputation mechanism which assesses the integrity of routing control traffic by correlating local routing data with feedback messages sent by the receivers of control traffic. Based on this assessment, misbehaving nodes are shown to be reliably detected and can be adequately punished in terms of their ability to communicate through the network. In [20], Cuppens et al. investigate the use of Aspect-Oriented Programming (AOP) in MANETs to provide availability issues in proactive routing protocols. Their approach is based on a detection-reaction process. Authors formally describe normal and incorrect node behaviors to derive security properties using AOP. The proposed algorithm verifies if those security properties are violated. If they are, then the detector node sends to its neighborhood the detection information to avoid choosing the intruder as part of valid paths to be constructed. A node chooses valid paths based on the reputation of other nodes.

6.2.2 Security Issues in OLSR Networks

In this section, we describe security attacks against the topology map acquisition process in OLSR networks. According to Herberg and Clausen [22], in OLSR networks every node must acquire and maintain a routing table that effectively reflects the network topology. Additionally, the routing tables constructed by every node must converge, i.e., all nodes must have an identical topology map. Therefore, the target of a misbehaving node may be that the nodes in the network (a) build inconsistent routing tables that do not reflect the accurate network topology, or (b) acquire an incomplete topology map. In link state routing protocols, some attacks can be launched even in networks with either cryptographic capabilities or IDS mechanisms implemented, e.g., a misbehaving node refuses to compute a valid MPR set. The exchange of control traffic information and the MPR selection process are important vulnerability targets. In this chapter, we focus on *flooding disruption attacks* [10], Fig. 6.3. In this kind of attacks, the target of an attacker is to disrupt the topology map acquisition process by disturbing the flooding of valid control traffic information. In [10], we presented a taxonomy of these attacks and countermeasures based on the selection of the MPR sets with additional coverage. The taxonomy we presented in [10] divides the attacks in two categories:

- Incorrect MPR Selection: in this category, the malicious node either selects an incomplete MPR set or forces other nodes to compute an incorrect MPR set. To launch the attack, the malicious node may either generate control traffic information with a false identity (i.e., identity spoofing) or report inexistent links

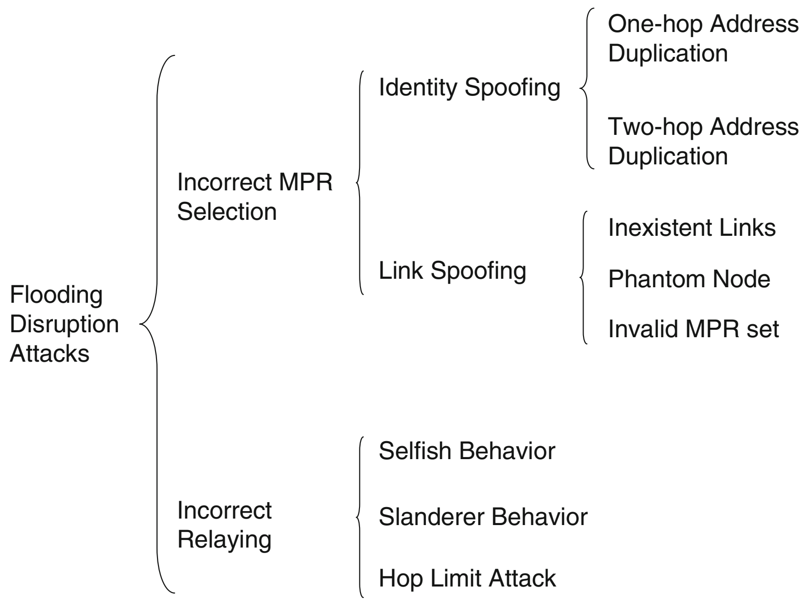


Fig. 6.3 Taxonomy of flooding disruption attacks [10]

to other nodes (i.e., link spoofing). As a consequence, the affected node computes an invalid MPR set, i.e., some of its two-hop neighbors are not covered through at least one node in its MPR set.

- Incorrect Relaying: in this category, the malicious node does not generate control traffic information (i.e., TC, MID or HNA messages) or does not forward valid messages on behalf of other nodes, e.g., selfish attack. In a variation of the attack, a malicious node may report incomplete information or eliminate some information reported by other nodes, e.g., slanderer behavior. Additionally, the misbehaving node can maliciously alter mutable fields in the messages before forwarding them, e.g., hop limit attack.

Figure 6.3 summarizes flooding disruption attacks in OLSR networks and the mechanisms used to perform them. In the sequel, we present these security threats in more detail. In Sect. 6.2.3 we present countermeasures to mitigate the effect of the attacks.

6.2.2.1 Incorrect MPR Selection

In this section, we describe vulnerabilities against the MPR selection process and some techniques to launch the attacks, i.e., link or identity spoofing.

Identity Spoofing. The identity spoofing attack [22] is performed by a malicious node pretending to be a different node in the network. The goal of the attack is to report false information about nodes one or two-hops away in order to maliciously affect the MPR selection process. Figure 6.4a illustrates an example where node x spoofs the identity of node d and broadcasts Hello message advertising a valid

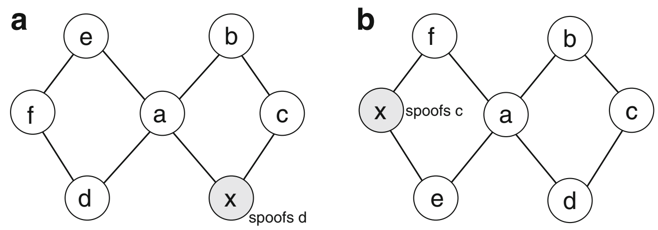


Fig. 6.4 Flooding disruption due to identity spoofing attacks. In Fig. 6.4a node *x* spoofs *d* and reports an incorrect link between nodes *c* and *d* (one-hop address duplication). In Fig. 6.4b, node *x* spoofs *c* and affects node *a*'s MPR selection (two-hop address duplication)

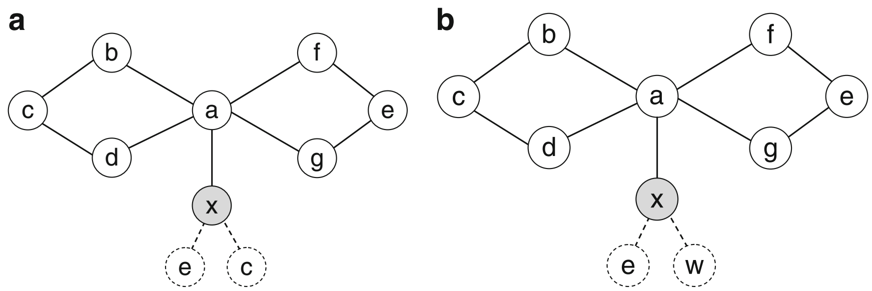


Fig. 6.5 Flooding disruption due to link spoofing attacks. In Fig. 6.5a, node *x* spoofs links to nodes *e* and *c*. In Fig. 6.5b, node *x* spoofs links to nodes *e* and the inexistent node *w*

link with node *c*. Then, node *a* receives Hello messages from node *x* indicating that node *d* has links with nodes *c* and *f*. In this case, node *a* selects incorrectly node *d* as the only element in its MPR set. In consequence, node *c* is unreachable through the MPR set and never receives TC messages. Figure 6.4b, presents an example where the attacker affects the MPR selection of a node at distance two hops. The malicious node *x* spoofs the identity of node *c*, i.e., nodes *f* and *e* generate Hello messages advertising node *c* as a one-hop neighbor. From the point of view of node *a*, nodes *b, e, f* and *d* have node *c* as a one-hop neighbor. As a result of the attack, node *a* can select incorrectly nodes *f* or *e* as an MPR. In this case, nodes *b* and *d* do not forward control traffic information to node *c* because they are not included in the MPR set.

Link Spoofing. The link spoofing attack [22] is performed by a malicious node that reports an inexistent link to other nodes in the network. The objective of the attacker is to manipulate the information about the nodes one or two hops away and be selected as part of the MPR set. Once the malicious node has been selected as an MPR, it neither generates nor forwards control traffic information. The flooding disruption attack due to link spoofing is illustrated in Fig. 6.5a. In this example, node *x* spoofs links to nodes *e* and *c*. Node *x* sends Hello messages and looks like the best option to be selected as an MPR for node *a*. Node *a* receives the Hello messages from node *x* and computes incorrectly its MPR set by selecting node *x* as the only element to reach nodes *e* and *c*. Thus, all routing information do not reach nodes two hops away from node *a*.

A variant of the attack can be performed by a misbehaving node either reporting a link to an inexistent node (i.e. a *phantom* node) or selecting an invalid MPR set.

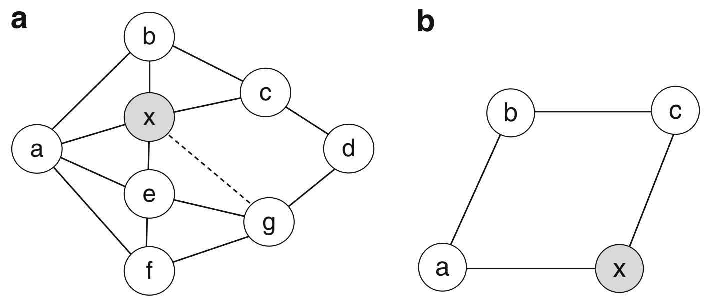


Fig. 6.6 Flooding disruption due to protocol disobedience. In Fig. 6.6a, node *x* never selects a valid MPR set. In Fig. 6.6b, node *x* modifies and forwards incorrectly TC messages

For instance, in Fig. 6.5b, node *a* is forced to select node *x* as an MPR because is the only node to reach the inexistent node *w*. In the second case, a malicious node may disrupt the flooding of topology control information by misbehaving during the MPR selection process. Figure 6.6a illustrates the attack. Node *x* wants to be selected as the only MPR of node *a*. Then, it spoofs a link to node *g* and generates Hello messages announcing node *g* as a one-hop neighbor and its only MPR. From the perspective of node *a*, nodes *c* and *g* can be reached through node *x*. Then, node *x* is the best candidate to be selected as an MPR for node *a*. Thus, node *x* receives and forwards TC messages from node *a*. However, those messages never reach node *d* because any one-hop neighbor of node *x* retransmits the messages. This attack exploits the *source dependent* requirement in OLSR to forward control traffic information. In this case, for nodes *a, b, c* and *e*, node *x* is not included in their selector table and they never forward any message from node *x*.

6.2.2.2 Incorrect Relaying

A misbehaving node can disrupt the integrity of the network by either incorrectly generating or relaying control traffic information on behalf of other nodes. Consider *x* in Fig. 6.6a as a misbehaving node. Node *x* wants to be selected as the only MPR of node *a*. Then, it spoofs a link to node *g* and generates Hello messages announcing node *g* as a one-hop neighbor. From the perspective of node *a*, nodes *c* and *g* can be reached through node *x*. Thus, node *x* is selected by node *a* as its only MPR and might perform the following incorrect behaviors:

- **Selfish behavior**. The attack is performed by a node that misbehaves and neither generates nor forwards TC messages. To increase the effectiveness of the attack, the malicious node might establish false links to other nodes in the network and force its one-hop neighbors to select it as their MPR. Figure 6.6a illustrates an example where node *x* has been selected by node *a* as an MPR but it does not relay control traffic on behalf of other nodes. As a result, node *d* does not receive control traffic information from node *a*. Note that in an OLSR network, the attacker can choose not to forward any particular message, i.e., TC, MID or HNA messages.

- **Slanderer behavior**. Due to message size limitations, an MPR may report only a partial list of elements in its selector set, i.e., an MPR may generate more than one TC message to report its entire selector set. A receiver can not know if an MPR reports its entire selector set in more than one TC message. The information gathered from the TC messages is accumulated in its topology table and is only eliminated when the validity time has expired. Thus, a misbehaving node can always generate TC messages without reporting all nodes in its selector table claiming that the size of the messages is not enough to include all nodes in its selector table. As a result, if node *x* generates TC messages without including node *a*, node *d* is not able to compute a path to node *a*.
- **Hop Limit attack**. A malicious node *x* may drastically decrease the hop limit (TTL value) when forwarding a TC message, e.g., setting the hop limit equal to zero. This reduces the scope of retransmitting the message. The attack can be performed by a malicious node that has not been selected as an MPR. For instance, in Fig. 6.6b, a control message is forwarded by node *a* and received by both nodes *x* and *b*. Previously node *b* was selected by node *a* as its MPR. However node *x* forwards the message without any delay or jitter such that its retransmission is received before the valid message from *b* arrives. Before forwarding, it reduces the hop limit of the message. The affected node, node *c*, processes the message and mark it as already received, ignoring future valid copies from *b*. Thus, the message with a very low hop limit will not reach the whole network.

6.2.3 Countermeasures

In an OLSR network, the MPR selection reduces at minimum the overhead generated by control traffic messages, if every node selects its MPR set with the following conditions:

- The MPR set is kept at minimum,
- An MPR retransmits control traffic messages if and only if the sender node is included in its selector table, and
- Only partial link state information is transmitted, i.e., an MPR reports only links with its selector nodes.

Nevertheless, we can loosen up the previous restriction in order to offer a higher level of security while maintaining a tradeoff between security and performance. In [10], we present strategies based on the selection of MPRs with additional coverage, a non source-dependent forwarding mechanism and redundant information. The selection of MPRs with additional coverage is defined in RFC3626 [12], we named it in [9] the *k*-Covered-MPR selection. In this approach, every node selects its MPR set such that any two-hop neighbor is covered by *k* one-hop neighbors, whenever possible. However, the overhead generated by the excessive number of TC messages reduces the performance of the network. This problem is addressed with the *k*-Robust-MPR selection presented in [9], which balances security and traffic overhead. In the

k-Robust-MPR selection, every node computes an MPR set that is composed of, at most, $k + 1$ disjoint sets, i.e., every two-hop node is covered, when possible, by $k + 1$ disjoint sets of one-hop neighbors. In a k-Robust-MPR selection, it is possible to discard a maximum of k invalid MPR sets and all nodes two hops away are still covered by the remaining elements in the MPR set. In a non source-dependant mechanism the MPRs retransmit all TC messages even if the sender node is not part of their selector set. Redundant information is possible by tunning the TC_redundancy parameter. This parameter is defined in the RFC3626 [12] and has three options:

- MPRs report their selector table when TC_redundancy is equal to zero,
- MPRs report their selector table and MPRs when TC_redundancy is equal to one, and
- MPRs report their one-hop neighbors when TC_redundancy is equal to two.

Advertising redundant information increases the size of the TC messages, but more links are advertised. In [9], we compared both k-Covered-MPR and k-Robust-MPR selections 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 showed that our k-Robust-MPR selection reduces the amount of traffic generated by the k-Covered-MPR selection, and offered equivalent protection against control traffic attacks. Our k-Robust-MPR selection increased the performance ratio of the number of nodes with complete routing tables over the number of topology control messages.

6.3 Hierarchical OLSR

In this section, we present the Hierarchical OLSR (HOLSR) protocol and its vulnerabilities. By nature, MANETs are formed of heterogeneous nodes that can join the network following an unpredictable pattern. Furthermore, scalability is a problem in MANETs. In [34], Villasenor-Gonzalez et al. define scalability as the capacity of the network to adjust or to maintain its performance even if the number of nodes increases. OLSR is a *flat* routing protocol and its performance degrades when the number of nodes increases due to a higher number of topology control messages propagated through the network. The MPR mechanism is local and therefore very scalable. However, the diffusion of link state information by all the nodes is less scalable. Hence, OLSR's performance decreases in large ad hoc networks. Additionally, OLSR does not differentiate the capabilities of the nodes and, in consequence, does not exploit nodes with higher capabilities. HOLSR is an approach designed to improve the scalability of the OLSR protocol in large-scale heterogeneous networks.

The main improvements are a reduction of topology control traffic and an efficient use of high capacity nodes. HOLSR organizes the network in hierarchical clusters. This architecture reduces the routing complexity, i.e., in case a link is broken only nodes inside the same cluster have to recalculate their routing table while nodes in other clusters are not affected. Nodes are organized according to

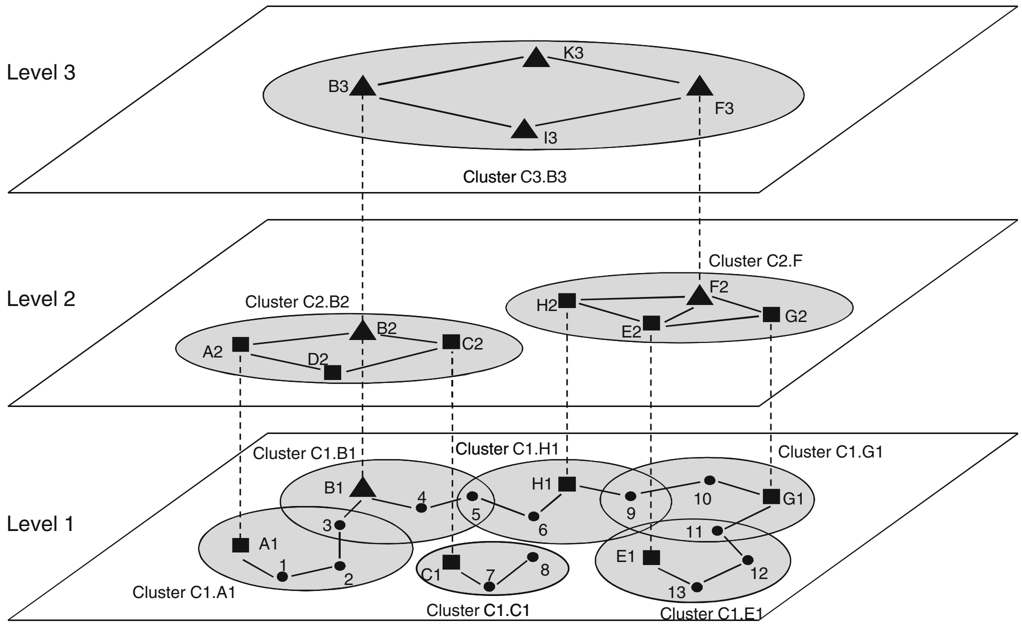


Fig. 6.7 Example of a hierarchical architecture with heterogeneous nodes

their capacities. The network architecture is illustrated in Fig. 6.7. At level 1, we have low-capability nodes with a single interface, represented by circles. Nodes at the topology level 2 are equipped with up to two wireless interfaces, designated by squares. Nodes at level 2 employ one interface to communicate with nodes at level 1. Nodes at level 3, designated by triangles, represent high-capacity nodes with up to three wireless interfaces to communicate with nodes at every level. Thus, in Fig. 6.7, node F3 represents node F's interface at level 3. The only restriction for every node at levels 2 and 3 is that they have at least one interface to communicate with nodes at its levels. For instance, in Fig. 6.7 nodes F2 and F3 represent node F's interfaces at levels 2 and 3 respectively. Nodes A1 and A2 represent node A's interfaces to establishes communication with nodes at levels 1 and 2 respectively. Node D2 has only one interface and can just communicate with nodes at level 2. In the example, the notation used to name the clusters reflects the level of the cluster and cluster head, e.g., C1.A1 means that the cluster is at level 1 and cluster head is node A1, which is node A's interface at level 1. HOLSR allows formation of multiple clusters. Unlike OLSR, HOLSR nodes can exchange Hello and TC messages exclusively within each cluster. This constraint reduces the broadcast traffic.

Across cluster topology control information is exchanged via specialized HOLSR nodes designated as cluster heads. Cluster heads are selected and nodes are classified according to their capabilities at the startup of the HOLSR process. A cluster is formed by a group of same-level mobile nodes that have selected a common cluster head. Nodes can move from one cluster to another and associate with the nearest cluster head. Any node participating in multiple topology levels automatically becomes the cluster head of the lower-level cluster. In HOLSR, a cluster head declares its status and invites other nodes to join in by periodically sending

Table 6.2 Summary of control traffic messages in HOLSR networks

Messages	Generated by	Retransmitted by	Reported information
Hello	Every node	N/A	One-hop neighbors
TC	MPRs	MPRs	MPR selector set
CID	Cluster heads	N/A	Cluster head identification
HTC	Cluster heads	MPRs	Nodes within a cluster

out Cluster ID Announcement (CID) messages. These and Hello messages are transmitted in the same packet using a grouping technique. This reduces the number of packet transmissions. A CID message contains two fields: *cluster head*, that represents the interface address of the originator of the message, and *distance*, which is the distance in hops to the cluster head generating the message. Every time the cluster head generates a CID message, the field *distance* is set to zero. A receiver node joins the cluster head and sends a new CID message. The new CID message increments the value of the distance. It invites other nodes to join the same cluster. The cluster formation process is described in more detail in [34].

The hierarchical architecture must support the exchange of topology control information between clusters without introducing additional overhead. Thus, Hierarchical TC (HTC) messages are generated by the cluster heads and used to transmit the membership information of a cluster to higher level nodes. HTC forwarding is enabled by the MPRs and restricted within a cluster. Nodes at the highest topology level have full knowledge of all nodes in the network. Their routing tables are as large as they would be in an OLSR network. However, in lower levels, the size of the routing table of every node is restricted by the size of the cluster and it is smaller than in OLSR. For instance, in Fig. 6.7 the cluster head A2 generates a HTC message at level 2 announcing that nodes 1, 2 and A1 are members of its cluster at level 1. The message is relayed to all nodes at the same level. Node B3 generates HTC messages at level 3 advertising that nodes 1, 2, 3, 4, 5, 7, 8, A1, B1, C1 (at level 1) and A2, B2, C2, D2 (at level 2) are members of its cluster. Table 6.2 presents a summary of the messages implemented in HOLSR networks.

Control messages are generated and propagated exclusively within each cluster unless a node is located in the overlapping zone of several clusters, i.e., a border node. For example, in Fig. 6.7 node 2 is within the border of cluster C1.A1 and may accept a TC or a HTC message from node 3 located in cluster C1.B1 (i.e., nodes 2 and 3 are border nodes). However, node 2 does not retransmit. Thus, except for the border nodes, knowledge of member nodes is restricted to the cluster itself. Data transfer between nodes in the same cluster is achieved directly using the routing tables. However, when transmitting data to destinations outside the local scope of a cluster, the cluster head is used as a gateway. A different strategy might be used, when transmitting data between border nodes in different clusters at the same level. Border nodes in different clusters at the same topology level can communicate directly without having to follow the strict clustering hierarchy. Therefore, HOLSR offers two main advantages (a) the traffic control reflecting local movements is restricted to each cluster (thus, reducing the routing table computation overhead), and (b) an efficient use of high-capacity nodes without overloading them.

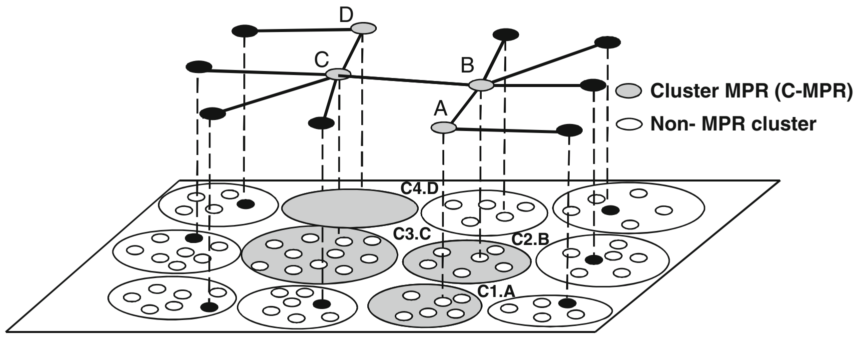


Fig. 6.8 Example of a Cluster OLSR network. Consider *gray* clusters as C-MPRs

6.3.1 Related Work

In this section, we review other hierarchical models based on OLSR to improve scalability in MANETs.

6.3.1.1 Cluster OLSR

In [31], Ros et al. present the Cluster OLSR (C-OLSR) protocol. Unlike HOLSR, C-OLSR does not assume any particular cluster formation algorithm nor existence of higher capacity nodes. C-OLSR implements OLSR inside every cluster and uses the MPR mechanism for distributing control traffic at both inter-cluster and intra-cluster levels. C-OLSR limits the forwarding of TC messages inside every cluster to minimize the number control traffic messages. Every node can compute routes to any other node inside its cluster. To reach nodes in other clusters, nodes create routes to every cluster and not to every node. When a data packet arrives to a destination cluster, every node has enough information to deliver the packet to its final destination. This mechanism reduces the size of the routing tables.

For inter-cluster communications, Cluster Hello (C-Hello) and Cluster Topology Control (C-TC) messages are defined. C-Hello messages are used to sense neighboring clusters and to compute the Cluster MPR (C-MPR) set. C-Hello messages are flooded within the receiver cluster but not retransmitted to neighbor clusters. A C-MPR is a cluster selected to reach other clusters and mitigate the overhead of distributing C-TC messages for inter-cluster communications. C-TC messages advertise the nodes within a cluster to all the network. Figure 6.8, shows an example of a C-OLSR network. At the first level, nodes are organized in clusters. The second level, shows how clusters are linked. Gray clusters are C-MPRs, e.g., C1.A is a C-MPR and node *A* is the cluster head. When a node in a cluster needs to send a data packet to a node inside another cluster, it computes a path through the clusters selected as C-MPRs, i.e., $C1.A$, $C2.B$, $C3.C$ and $C4.D$.

When a C-Hello or C-TC messages arrive to a cluster, they are relayed to every node in the cluster. This allows nodes to learn about clusters topological

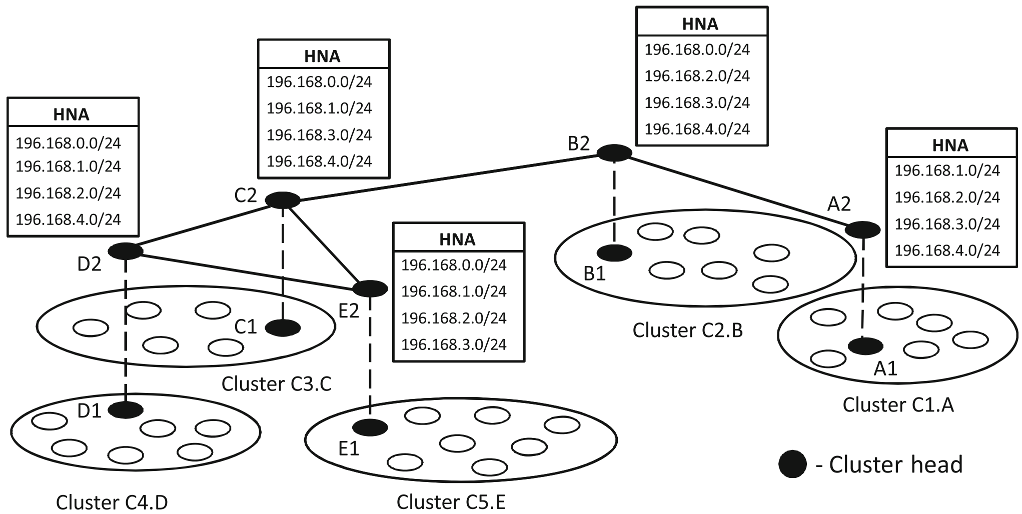


Fig. 6.9 Example of a MORHE network. Consider *black nodes* as backbone nodes

information. C-TC messages must be relayed to adjacent clusters, only if the sender of the message has selected the receiver node as an C-MPR. To support this hierarchical architecture, every C-OLSR node has additional information repositories: one-hop neighbor cluster set, two-hop neighbor cluster set, cluster topology set, cluster MPR set and cluster MPR selector set. The information in these repositories supports inter-cluster communications. In C-OLSR, not every node has to generate inter-cluster information. The generation of C-Hello and C-TC messages can be done according to three different algorithms: a cluster head-based algorithm, a distributed algorithm or a hybrid approach. In the former case, only cluster heads generate control information. In the second algorithm, topology information is generated exclusively by border nodes. Finally, in the hybrid approach, C-Hello messages are generated by border nodes and C-TC messages are generated by the cluster heads. In all cases, the selected C-MPRs are responsible for forwarding C-TC messages.

6.3.1.2 The *Multi-level OLSR Routing Using the HNA Extension*

In [35], Voorhaen et al. present a multi-level routing scheme for ad hoc networks based on OLSR. The *Multi-level OLSR Routing using the HNA Extension* (MORHE) protocol improves scalability by exploiting high capability nodes. Using HNA messages and hierarchical addressing, MORHE constructs an overlay network formed by nodes with higher capabilities. Nodes with higher capabilities are selected as cluster heads. A cluster head is called a *backbone* node. Backbone nodes are chosen before network deployment and have more than one interface. Nodes are organized into clusters around every backbone node. Figure 6.9, shows an example of a two-levels MORHE network. Nodes A, B, C, D and E are backbone nodes. Backbone nodes use one interface to communicate with the nodes inside their cluster and the second interface for inter-cluster communications. For instance, backbone node A, communicates with the nodes at the first level through the interface $A1$ and uses interface $A2$ to communicate with other backbone nodes. OLSR is implemented at each level.

MORHE is similar to HOLSR, nonetheless it only uses HNA messages already defined in the RFC 3626 [12]. Each backbone node periodically sends HNA messages informing other backbone nodes that it can reach all the nodes in the subnet that it is connected to. When a backbone node receives a HNA message, it updates its association database. Every backbone node uses HNA messages to inform all the nodes in its cluster about other clusters that can be reached. HNA messages are distributed using the MPR mechanism as defined in OLSR. Nodes can communicate directly with every node inside its cluster. Backbone nodes enable communication between nodes in different clusters. When a packet arrives at a backbone node, it attempts to find a route to the destination in its cluster. If this fails, then the backbone node retransmits the message to another backbone node. If the receiver finds a route, then it forwards the packet inside its cluster. In a MORHE network, every cluster is identified as a subnetwork. For instance, in Fig. 6.9, the network is divided in five subnetworks. Every backbone node has the IP addresses of every subnetwork in its association table. For example, 192.168.1.0/24 is the prefix of an IPv4 subnetwork, having 24 bits allocated for the network prefix, the remaining 8 bits are reserved for host addressing. If a node inside the subnetwork $192.168.0.0/24$ needs to communicate with a node in the subnetwork $192.168.2.0/24$, then it sends the packet to its backbone node which retransmits the packet to its final destination.

6.3.1.3 Tree Clustering

In [6, 7], Baccelli proposed a *Tree Clustering* mechanism to enable hierarchical routing within an OLSR network. Each cluster is a *tree*. Their head is the *root*. To organize the network in trees, every node selects as its parent the adjacent node with the maximum number of one-hop neighbors. The parent of a node is called a node's preferred neighbor. A node with maximum degree, i.e., maximum number of neighbors, is selected as the root of the three. The network is then viewed as a *forest*, i.e., a collection of logical trees. To form and maintain trees, OLSR nodes periodically exchange *Branch* messages. These messages are piggy-backed with Hello messages. Branch messages are not retransmitted. Within a Branch message, a node specifies its identity, the tree it belongs to, its parent in the tree and its distance in hops to the root. Roots can choose to limit the size of their three by imposing a *maximum depth* value. The organization in trees is dynamic. A mechanism allows to switch between a traditional flat networking, i.e., flat mode or a hierarchical networking, i.e., tree mode. The mechanism to transit between the flat mode and the tree mode is explained in detail in [6].

Within a tree, OLSR nodes operate as if there was no tree, except that messages originated by a node in a different tree are not considered and not forwarded, the root is responsible for the communication between the tree and the rest of the network, and a node in contact with another tree i.e., a leaf node, must inform its entire tree (specially its root), of the distance to reach other roots. A leaf node must generate a *Leaf* message for each other tree it reaches. In a Leaf message, the node specifies its ID, the root of the neighbor tree and the estimated distance between the roots, i.e., the sum between its depth in its tree, and the distance to the root of the neighbor tree. With this information, every root is able to compute the shortest path to reach its neighbor roots.

This protocol employs Hello and TC messages within every tree, but implements Super-Hello (S-Hello), Super-TC (S-TC) and Super-HNA (S-HNA) messages for inter-cluster communications. Super messages are generated exclusively by the roots. These messages are identical to regular messages except for an additional field that includes the IP address of the next root to reach. Unlike regular messages, Super-messages are routed using the constructed paths instead of being flooded. Super-messages are unicasted using the shortest root-to-root path advertised by Leaf messages. Super-messages are the only messages to be forwarded outside a tree. MPR selection is performed as if there were no trees. When a tree mode is activated, the scope of TC messages is limited to the tree they were generated. However, Super-messages are forwarded between clusters following the MPR flooding mechanism.

To allow hierarchical routing, routes exchange Super-messages in order to identify other roots and construct a Super-topology. S-Hello and S-TC messages allow the roots to construct a super-topology formed by roots. The roots periodically exchange S-Hello messages to learn about other roots in neighbor trees (i.e., one-super-hop neighbors). As in OLSR, every root computes its super-MPR set formed by other roots. A super set of MPRs is used for distributing S-TC messages among clusters. S-Hello messages are not forwarded. S-TC messages are forwarded by the S-MPRs. S-TC messages include the super-selector set, i.e., the roots that have selected the sender as a S-MPR. Finally, every root generates S-HNA messages to inform other roots about the link state information within its cluster. Therefore, every root is aware of the link state information of other threes. Routing among clusters is achieved using the information between S-TC and S-HNA messages. Traffic outside the tree scope is achieved via the root nodes. Figure 6.10 shows an example of a tree clustering hierarchical architecture. Nodes A, B, C, D and E are selected as roots. These nodes have the maximum degree. Root node A selects B as its MPR to reach root trees C, D and E. When a node inside cluster C1.A needs to communicate with a node inside cluster C5.E, it sends the data traffic to its root node A which retransmits the traffic to its final destination trough B and E.

Table 6.3 presents a summary of the features of each hierarchical approach that we reviewed. Unlike MORHE and C-OLSR, HOLSR and the Tree clustering approaches include a cluster formation mechanism. MORHE and HOLSR were designed for heterogeneous networks and multiple hierarchical levels. C-OLSR and Tree clustering were designed for homogeneous networks and two hierarchical levels. Nevertheless, these approaches might be implemented in networks with heterogeneous capabilities. All approaches implement the MPR mechanism for distributing control traffic messages.

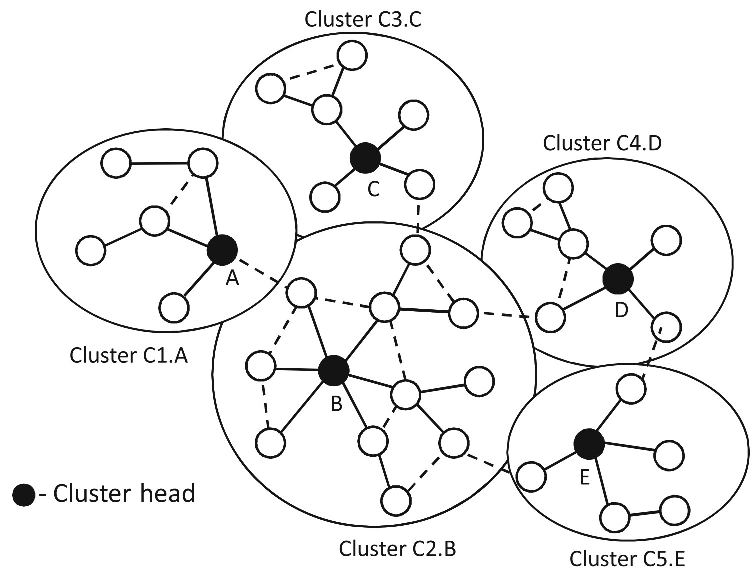


Fig. 6.10 Tree clustering. *Black nodes* represents the roots of the tree. Branches of the trees are shown with *solid lines* between nodes. Links that are not branches are *dashed*

Table 6.3 Comparison of OLSR-based hierarchical approaches. All approaches implement Hello and TC message for intra-cluster communications

Routing protocol	Network	Logical levels	Messages	Cluster formation Alg.
HOLSR	Heterogeneous	n	CID and HTC	Yes
MORHE	Heterogeneous	n	HNA	No
C-OLSR	Homogeneous	2	C-Hello and C-TC	No
Tree	Homogeneous	2	Leaf, Branch, S-Hello, S-TC and S-HNA	Yes

6.3.2 Security Issues in HOLSR Networks

Note that in all described approaches, the exchange of control traffic at both intra-cluster and inter-cluster levels is performed by using the MPR mechanism. Security is no addressed. Therefore, they are vulnerable to the flooding disruption attacks described in Sect. 6.2.2. The cluster formation phase is vulnerable to malicious behavior. In [10, 11], we describe in detail security threats to both the *cluster formation* and *topology map acquisition* phases.

In HOLSR, the flow of CID messages is an important vulnerability target. The *hop count* has to be updated every time a new message is retransmitted. Thus, a malicious node might alter this field to unsettle the cluster formation process. The attack, has a bigger impact when a malicious node drastically reduces the *hop count* field. Because receivers accept the CID message with the lowest *hop count* value. Thus, when an attacker increases drastically the value, receivers automatically discard the altered message and accept valid messages from other nodes. When a

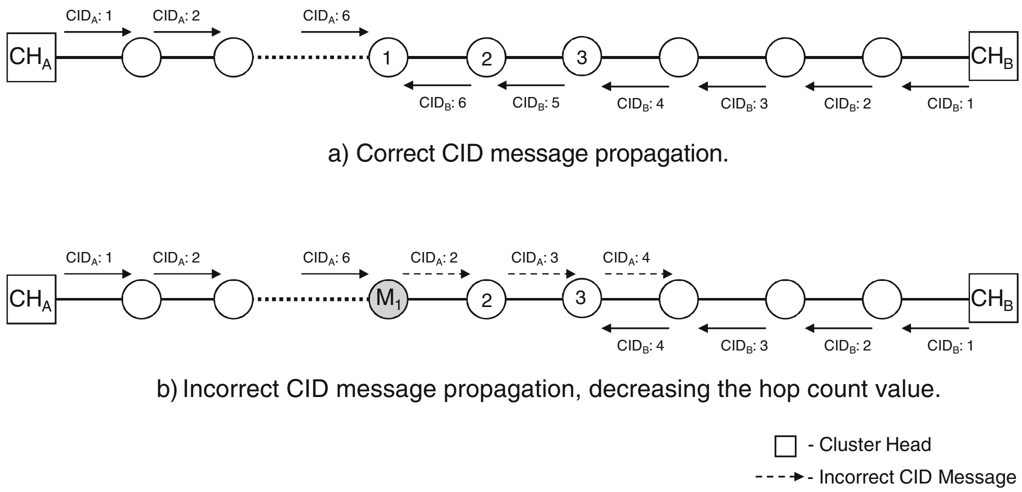


Fig. 6.11 Cluster formation attack in HOLSR networks. (**a**) Correct CID message propagation. (**b**) Incorrect CID message propagation, decreasing the hop count value

node that generates a CID message reinitializes the value of the field *hop count*, the receiver nodes may join a farther cluster head and discard valid CID messages from closer cluster heads. We address the case where the *hop count* field is maliciously reduced. For instance, Fig. 6.11a shows the correct propagation of CID messages. Figure 6.11b shows an example of the attack. In Fig. 6.11b, M_1 is a malicious node at distance six hops from cluster head CH_B. M_1 receives CID messages from CH_B, and generates a new CID message assigning the incorrect value two to the field *hop count*. Thus, all nodes at distance from CH_B, greater or equal than four hops (nodes 2 and 3) process the message and incorrectly join CH_A. Note that the lowest value that can be used to reinitialize the field *hop count* is two because CID messages with a field *hop count* equal to one are generated exclusively by the cluster heads. We assume that the attacker has only one interface. It can not impersonate a cluster head. It only modifies the *hop count* value. This attack can affect other OLSR-based hierarchical approaches. For instance, a misbehaving node may alter the field distance in *Branch* messages in the Tree Clustering approach proposed by Baccelli, reviewed in Sect. 6.3.1.3.

6.3.3 Countermeasures

In [10, 11], we describe in detail security threats in both the *cluster formation* and *topology map acquisition* phases. Countermeasures to mitigate the effect of the attacks are also presented. In the former case, in [10], we analyze the effect of flooding disruption attacks in HOLSR networks to interrupt the propagation of HTC messages. We proposed additional coverage in the selection of MPRs at any hierarchical level. We analyze the effect of flooding disruption attacks. Unlikely flat OLSR networks, when a malicious nodes attempts to interrupt the propagation of

HTC messages the inter-cluster communication is affected. Our proposed solution is based on the selection of MPRs with additional coverage, i.e., *k*-Covered-MPR and *k*-Robust-MPR selections. Our results showed that it is possible to mitigate the effect of the attack by adding additional coverage. The *k*-Covered-MPR selection increased the chances of mitigate the attack but the performance of the network reduces due to an increased number of TC and HTC messages. Our proposed *k*-Covered-MPR selection offers an equivalent level of protection but reducing the amount of TC and HTC messages flooded in the network.

In [11], we presented a solution based on *hash chains* to protect mutable fields in HOLSR networks. Our algorithm Hash-Chained_CID_Dissemination (HCCD) allows to detect and discard invalid CID messages. A valid cluster head (CH_j) generates a random number s_j, i.e., a nonce that is only known by the originator of the message. After, it initializes the hop count field i equal to one and computes the Max_j value by applying t times the hash function $h(x)$ to the nonce s_j, such that Max_j is equal to $h^t(s_j)$. We assume that Max_j and the value of t are known by all the nodes in the network during the execution of the protocol. Additionally, CH_j applies i times the hash function to s_j, to obtain $h^i(s_j)$. Then, CH_j generates a *CID* message with the fields: $< Max_j, h^i(s_j), i >$. The receiver node verifies that the CID message is valid by applying $t-i$ times the hash function to $h^i(s_j)$ and comparing the result with Max_j. Therefore, if Max_j is equal to $h^{t-i}(h^i(s_j))$, then the hop count value i has not been altered and the received CID message is valid. Finally, the receiver node joins CH_j until it receives a CID message from a different cluster head with a lower hop count value. In the mean time, the receiver node generates periodically CID messages announcing its cluster head and the hop count distance to reach it, i.e., $< Max_j, h(h^i(s_j)), i+1 >$. Our solution is based on the work presented by Hong et al. in [23]. The authors presented a wormhole detective mechanism and an authentication protocol to strengthen the neighbor relationship establishment in standard OLSR. The authors used digital signatures to ensure the non-mutable fields and hash chains to secure the Hop Count and TTL fields. Their solution is similar to our proposed algorithm, however it is implemented in flat OLSR to protect only standard control traffic messages. We address a different kind of attack in HOLSR networks. Our mechanism protects the integrity of CID messages and enforces the proper distribution of nodes in every cluster. In [11], our experiments showed that the distribution of nodes is less balanced when the hop count in CID messages is maliciously altered. We also showed that we can prevent this kind of attacks by applying our proposed algorithm. Note that our mechanism, can be also applied in other hierarchical routing protocols for MANETs that utilize mutable information to organize the network in clusters.

6.4 Multipath OLSR-Based Routing

In this section, we analyze a multipath routing strategy based on OLSR that takes advantage of the MPR flooding mechanism. In [37–40], Yi et al. proposed the

Multipath OLSR (MP-OLSR) routing protocol aiming to enhance load-balancing, energy-conservation, Quality-of-Service (QoS) and security. MP-OLSR is a hybrid multipath routing protocol. In MP-OLSR, the OLSR proactive behavior is changed for on-demand route computation. MP-OLSR becomes a source routing protocol. There are two phases: *topology discovery* and *routes computation*. During *topology discovery*, nodes obtain a partial topology map just like in OLSR. However, MP-OLSR nodes do not construct routing tables. During *routes computation*, nodes calculate multiple paths to reach any other node in the network following an on-demand scheme. MP-OLSR implements Multiple Description Coding (MDC) for data transfer. MDC adds redundancy to information streams and split them up into several sub-streams to improve the integrity of data. These sub-streams are sent along multiple paths from the source to the destination. MP-OLSR implements source routing with route recovery and loop detection to adapt to the changes in the network topology. Thus, when data transfer is required, route recovery and loop detection allow every node to detect if a path is not valid anymore and to find a new path to reach the final destination. MP-OLSR uses the Dijkstra's algorithm to discover routes. The routes that are obtained can be grouped in two categories:

1. Disjoint: In this category we have two types of disjoint paths: node-disjoint and link-disjoint. Node-disjoint paths type do not share nodes except for the source and destination nodes. Link-disjoint paths can share some nodes but all the links are different.
2. Inter-twisted: In this case, the paths may share several links.

To construct disjoint paths, MP-OLSR defines cost functions to obtain new paths that tend to be node-disjoint or link-disjoint. Once a path is computed, a function f_p is used to increase the costs c of the links that belong to the computed path, e.g., $f_p(c) = 3c$. A function f_e is defined to increase the cost of the links of the nodes included in the path previously obtained. In MP-OLSR, neither nodes nor links used in computed paths are eliminated. This strategy allows MP-OLSR to construct multiple paths in sparse networks where is not always possible to find strictly node-disjoint paths. In addition, to increase the chances of constructing node-disjoint paths, the MPRs report all their one-hop neighbors (i.e., the TC_redundancy parameter is equal to two). Consider f_{id} as the identity function, i.e., $f_{id}(c) = c$. Therefore, to construct disjoint paths, there are three possibilities:

- If $f_{id} = f_e < f_p$, then paths tend to be link-disjoint;
- If $f_{id} < f_e = f_p$, then paths tend to be node-disjoint;
- If $f_{id} < f_e < f_p$, then paths also tend to be node-disjoint, but when not possible they tend to be link-disjoint.

For example, in Fig. 6.12a, node s attempts to construct multiple paths to node d. MP-OLSR implements a Multipath Dijkstra's algorithm to obtain the shortest paths. Consider initial cost c of each link equal to one and $f_p(c) = 3c$ and $f_e(c) = c$, i.e., a penalty is only applied to the used links. The first time the Dijkstra's algorithm is applied, the computed path is $s \to c \to d$. Thus, the cost of the links (s,c) and (c,d) is changed from one to three using f_p, see Fig. 6.12b. The second path we obtain is: $s \to b \to c \to h \to d$. The cost of the links (s,b), (b,c), (c,h) and (h,d) is set

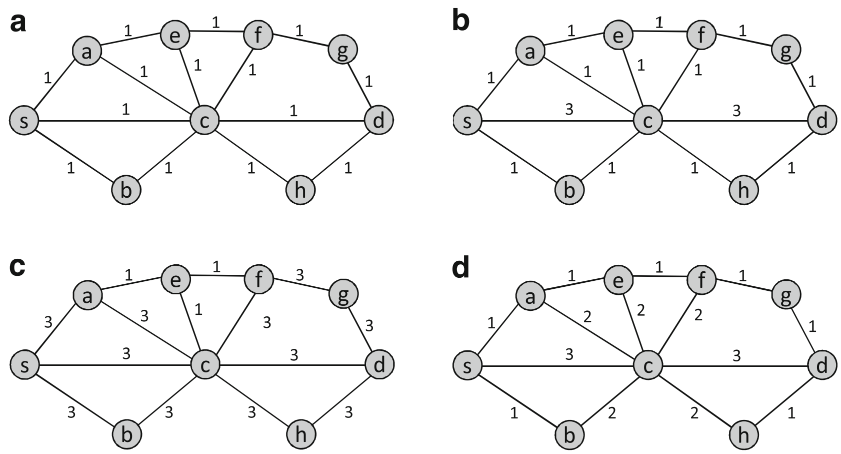


Fig. 6.12 OLSR network. In Fig. 6.12a, consider the cost of all links equal to one

to three. Finally, the third computed path is: $s \rightarrow a \rightarrow c \rightarrow f \rightarrow g \rightarrow d$. The cost of all used links is set to three, see Fig. 6.12c. These three paths are link-disjoint. To obtain paths that tend to be node-disjoint, we define functions $f_p(c) = 3c$ and $f_e(c) = 2c$. In this case, the penalty is also applied to the used nodes. First, the path $s \rightarrow c \rightarrow d$ is computed and the cost of the links is updated. The links that include a node in the computed path -except for the source s and the destination d- are set to two, see Fig. 6.12d. Then, the next path we obtain is: $s \rightarrow a \rightarrow e \rightarrow f \rightarrow g \rightarrow d$. These two paths are node-disjoint. The path: $s \rightarrow a \rightarrow c \rightarrow h \rightarrow d$, is an example of an inter-twisted path.

6.4.1 Related Work

In this section, we present other multipath routing strategies based on OLSR. Several multipath routing approaches take advantage of the proactive behavior and MPR flooding mechanism proposed in OLSR. The strategies proposed, attempt to improve security, QoS, load balancing or energy consumption. However, all strategies proposed are not secure by design. For instance, in [26], Kun et al., proposed a different version of multipath OLSR using IP-source routing. Based on the Dijkstra's algorithm, nodes calculate multiple node-disjoint paths. Additionally, the authors introduce an algorithm of load-assigned to transmit data through the paths based on the congestion information of all intermediate nodes on each path. Badis and Al Agha [8], also proposed a path selection criteria and multi-path calculation based on bandwidth and delay to improve QoS in OLSR networks (QOLSR). The resulting protocol, computes multiple loop-free and node-disjoint paths. The authors implement the shortest-widest path algorithm to guarantee loop-free routes. Additionally, they evaluated and compared QOLSR multipath routing versus a QOLSR single-path routing using a scalable simulation model. In [32], Srinivas and Modiano proposed algorithms for finding minimum energy disjoint paths in wireless networks. Their main contribution is a polynomial time algorithm for the minimum energy k node-disjoint problem. Node-disjoint paths are more resilient to failures. However, the authors showed that link-disjoint paths save more energy. Zhou et al. proposed in [41] the Source Routing based Multi-Path OLSR (SR-MPOLSR) protocol. The protocol implements the Dijkstra's algorithm to calculate multiple disjoint routes. Data transmission at the source is carried out through predetermined multiple paths (i.e., source routing). The loads are distributed in a weighted round-robin fashion. These strategies proposed attempt to construct multiple link-disjoint or node-disjoint paths. However, all approaches are affected by the flooding disruption attacks described in Sect. 6.2.2. Nodes in OLSR-based multipath routing protocols only acquire a partial view of the topology network. These problems are described in the following section.

6.4.2 Security Issues in Multipath OLSR-Based Networks

Multipath OLSR-based approaches are vulnerable to the flooding disruption attacks [10] attacks presented in Sect. 6.2.2 during the *topology discover* and *route computation* phases. An attacker may refuse to retransmit control traffic or may select an invalid MPR set to prevent other nodes from calculating disjoint paths to reach other nodes in the network. MP-OLSR constructs non disjoint multiple paths. The protocol computes several routes, but it is impossible to know how many of them are disjoint. When a node part of several paths misbehaves, all paths are affected. All OLSR-based multipath strategies use the MPR mechanism to flood the network with control traffic. However, only partial topology information is generated by the MPRs. We identify two vulnerabilities in all OLSR-based multipath routing strategies: the nodes in an OLSR network only obtain a partial view of the network topology and they are affected by the security threats presented in Sect. 6.2.2. The MPRs generate and forward TC messages to advertise their selector set to other nodes at more than two hops away. However, with this information nodes only obtain a partial view of the topology. This is because TC messages only report partial link state information. For instance, Fig. 6.13a shows the complete topology of an MP-OLSR network. Gray nodes represent MPRs. Figure 6.13b shows the perspective of node s after the topology discovery phase. The links (g, j), (i, l), (j, d), (l, d), (j, k) and (l, k) are not reported in TC messages. Thus, the link between node g and j is not reported because neither g nor j are MPRs. Node k is an MPR but it does not report links to nodes j and l because they are not included in its selector set. From the perspective of node s, k is the only node that reaches node d. Hence, it is not possible to compute multiple disjoint paths. To increase the chances of finding disjoint paths, the MPRs in an MP-OLSR networks report more information in their TC messages by tunning their TC_redundancy parameter. The TC_redundancy parameter is defined locally by every node. Nodes with different TC_redundancy values can coexist. MP-OLSR nodes set their TC_redundancy

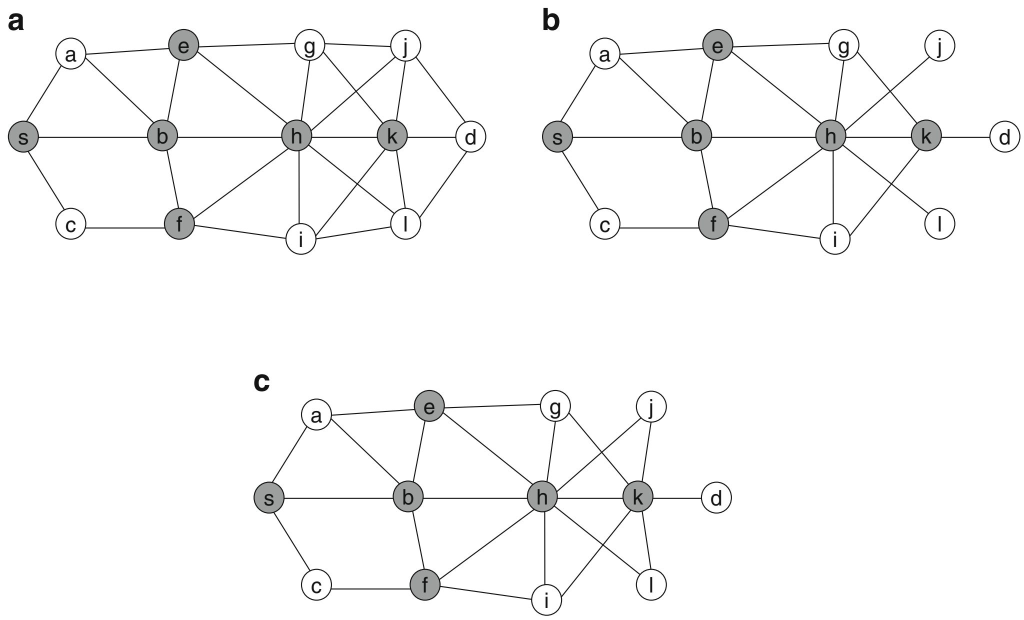


Fig. 6.13 Network topology perspective of node *s*. *Gray nodes* represent MPRs. (**a**) Complete network topology. (**b**) Node s perspective of the network. TC_redundancy equal to 0. (**c**) Node s perspective of the network. TC_redundancy equal to 2

parameter to two. However, the size of the TC messages increases and in some situation it is not enough to report important links. For example, Fig. 6.13c shows the network perspective of node s if the MPRs report their one-hop neighbors, i.e., TC_redundancy parameter equal to two. Hence, node s is aware of the links (j,k) and (l,k). However, the links (g,j), (i,l), (j,d) and (l,d) remain unreported. Figure 6.13c also shows that all the possible routes to reach node d include node k. When node k misbehaves, all the computed paths are compromised.

6.4.3 Countermeasures

The MPR selection with additional coverage (i.e., k-Robust-MPR or k-Covered-MPR) helps to mitigate the attacks against the construction of disjoint paths. Additional coverage helps to advertise more links and construct multiple node-disjoint paths without increasing the size of the messages. In OLSR networks, the MPRs form a Connected Dominating Set (CDS). A CDS is a subset of connected nodes such that if a node in the network is not part of the CDS, then it has a link to a node in the CDS. Every node must be able to construct a CDS of the network with the information gathered during the topology discovery phase. We define an MPRCDS as a CDS such that every node in the CDS has been selected as an MPR. When the nodes select their MPRs following a k-Covered-MPR selection we obtain a k-CCDS. When the nodes compute their MPRs following a k-Robust-MPR selection we obtain a k-RCDS. Therefore, if a node obtains a more complete view of the network (i.e., k-CCDS or k-RCDS), then it is able to find alternative routes to compute disjoint paths.

6.5 Conclusion and Future Work

In link state routing protocols for MANETs, the generation and exchange of control traffic messages are important vulnerability targets. A malicious node may perpetrate an attack by flooding the network with incorrect information or by preventing other nodes from acquiring a complete network topology map. We presented security threats in link state routing protocols based on OLSR. Particularly, we addressed flooding disruption attacks in OLSR networks. This kind of attacks can be carried out in networks with cryptographic capabilities. Additionally, a review of related work and proposed countermeasures is also presented. In addition, we reviewed security threats in other link state routing protocols based on OLSR. We presented vulnerabilities and countermeasures specific to HOLSR and MP-OLSR.

6.5.1 Future Work

The k-Robust-MPR selection may be affected either by a malicious node, that generates false links to avoid the selection of k+1 disjoint MPR sets or due to the network topology. As part of future work, we consider an extended k-Robust-MPR selection to address the cases when is not possible to select multiple disjoint MPR sets. Countermeasures against more complex attacks during the cluster formation phase in hierarchical OLSR-based networks is also part of further research. A mechanism to improve the selection of multiple disjoint routes in OLSR-based networks is required. To improve load balancing, nodes with the smallest number of nodes in their selector set should be privileged to be included in the computed paths. Clearly, in sparse networks is not always possible to compute disjoint paths. Nevertheless, multipath routing takes advantage of large and dense networks. Then, the cases where the construction of multiple node-disjoint paths is affected either by an incomplete view of the network topology or by the presence of a misbehaving node should be addressed.

Acknowledgements The authors graciously acknowledge the financial support received from the following organizations: Natural Sciences and Engineering Research Council of Canada (NSERC), Mathematics of Information Technology and Complex Systems (MITACS), Institut Telecom, Spanish Ministry of Science and Innovation (grants TSI2007-65406-C03-03 E-AEGIS and CONSOLIDER-INGENIO CSD2007-00004 ARES), National Council of Science and Technology (CONACYT), and Ministry of Education of Mexico (SEP, Program for Academic Improvement).

References

1. R. Abdellaoui and J.-M. Robert. SU-OLSR: A new solution to thwart attacks against the OLSR protocol. In *4th Conference on Security in Network Architectures and Information Systems (SAR-SSI)*, pages 239–245, Luchon, France, June 22–26, 2009.
2. 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.
3. A. Adnane, R.T. de Sousa Jr., C. Bidan, and L. Me. 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. H. Aiache, F. Haettel, L. Lebrun, and C. Tavernier. Improving security and performance of an ad hoc network through a multipath routing strategy. *Journal in Computer Virology*, vol. 4(4), pages 267–278, 2008.
5. P. Arce, J.C. Guerri, A. Pajares, and O. Lázaro. Performance evaluation of video streaming over ad hoc networks using flat and hierarchical routing protocols. In *Mobile Networks and Applications*, vol. 13(3–4), pages 324–336, 2008.
6. E. Baccelli. OLSR scaling with hierarchical routing and dynamic tree clustering. In *IASTED International Conference on Networks and Communication Systems (NCS)*, Chiang Mai, Thailand, March 2006.
7. E. Baccelli. OLSR trees: A simple clustering mechanism for OLSR. In *Challenges in Ad Hoc Networking, IFIP International Federation for Information Processing*, vol. 197, pages 265–274, 2006.
8. 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, pages 2181–2184, May 2004.
9. G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis. Mitigation of topology control attacks in OLSR networks. In *5th International Conference on Risks and Security of Internet and Systems (CRISIS 2010)*, Jean-Marc Robert, editor, pages 81–88, Montreal, Canada, October 10–13, 2010.
10. G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis. Mitigation of flooding disruption attacks in HOLSR networks. *In 9th Annual Conference on Communication Networks and Services Research Conference (CNSR 2011)*, pages 167–174, 10.1109/CNSR.2011.32. Ottawa, ON, Canada, May 2–5 2011.
11. G. Cervera, M. Barbeau, J. Garcia-Alfaro, and E. Kranakis. Preventing the Cluster Formation Attack Against the Hierarchical OLSR Protocol: Invited Talk. In proceedings of *4th Canada-France MITACS Workshop on Foundations & Practice of Security (FPS 2011)*, Paris, France, May 12–13 2011. Springer LNCS.
12. T. Clausen and P. Jacquet. Optimized link state routing protocol (OLSR), RFC3626. IETF Internet Draft, Available via http://www.ietf.org/rfc/rfc3626.txt, October 2003.
13. 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-olsrv2-13.txt, November 2011.
14. T. Clausen and C. Dearlove. RFC5497: Representing Multi-Value Time in Mobile Ad Hoc Networks (MANETs), std. track, http://www.ietf.org/rfc/rfc5497.txt.
15. T. Clausen, C. Dearlove, and B. Adamson. RFC5148: Jitter Considerations in Mobile Ad Hoc Networks (MANETs), informational, http://www.ietf.org/rfc/rfc5148.txt.
16. T. Clausen, C. Dearlove, and J. Dean. I-D: MANET Neighborhood Discovery Protocol (NHDP), work in progress.
17. T. Clausen, C. Dearlove, J. Dean, and C. Adjih. RFC5444: Generalized mobile ad hoc network (manet) packet/message format, std. track, http://www.ietf.org/rfc/rfc5444.txt.

18. T. Clausen and U. Herberg. Vulnerability analysis of the optimized link state routing protocol version 2 (OLSRv2). In *Wireless Communications, Networking and Information Security (WCNIS)*, 2010 IEEE International Conference on, pages 628–633, 2010.
19. T. Clausen, U. Herberg, and J. Milan. Digital signatures for admittance control in the optimized link state routing protocol version 2. Research Report RR-7216, INRIA, February 2010.
20. 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, pages 52–59, Washington, DC, USA, 2007. IEEE Computer Society.
21. A. Hajami, K. Oudidi, and M. Elkoutbi. An enhanced algorithm for MANET clustering based on multi hops and network density. In *New Technologies of Distributed Systems (NOTERE)*, 2010 10th Annual International Conference on, pages 181–188. IEEE, 2010.
22. U. Herberg and T. Clausen. Security Issues in the Optimized Link State Routing Protocol version 2 (OLSRv2). *International Journal of Network Security & Its Applications (IJNSA)*, vol. 2(2), 2010.
23. F. Hong, L. Hong, and C. Fu. Secure OLSR. International Conference on Advanced Information Networking and Applications (AINA 2005), vol. 1, pages 713–718, Taipei, Taiwan, March 2005.
24. 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.
25. A. R. Khakpour, M. Laurent-Maknavicius, and H. Chaouchi. WATCHMAN: An overlay distributed AAA architecture for mobile ad hoc networks. In *ARES '08: Proceedings of the 2008 Third International Conference on Availability, Reliability and Security*, pages 144–152, Washington, DC, USA, March 2008. IEEE Computer Society.
26. M. Kun, Y. Jingdong, and R. Zhi. The research and simulation of multipath-OLSR for mobile ad hoc network. In *Communications and Information Technology, 2005*. ISCIT 2005. IEEE International Symposium on, vol. 1, pages 540–543, October 2005.
27. J. Liu, X. Hong, J. Kong, Q. Zheng, N. Hu, and P. Bradford. A hierarchical anonymous routing scheme for mobile ad-hoc networks. In *Proceedings of the Military Communications Conference (MILCOM 2006)*, IEEE, pages 1–7, 23–25 October, 2006. Washington, DC.
28. J. Moy. Open Shortest Path First (OSPF) version 2, RFC2328. IETF Internet Draft, http://www.ietf.org/rfc/rfc2328.txt, April 1998.
29. D. Raffo. Security schemes for the OLSR protocol for ad hoc networks. PhD thesis, LUniversité Paris 6 - Pierre et Marie Curie, INRIA Roquencourt, September 2005.
30. 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.
31. F.J. Ros and P.M. Ruiz. Cluster-based OLSR extensions to reduce control overhead in mobile ad hoc networks. In *Proceedings of the 2007 international conference on Wireless communications and mobile computing*, pages 202–207. ACM, 2007.
32. A. Srinivas and E. Modiano. Minimum energy disjoint path routing in wireless ad-hoc networks. In *Proceedings of the 9th annual international conference on Mobile computing and networking (MobiCom 03)*, pages 122–133, New York, NY, USA, 2003. ACM.
33. 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.
34. L. Villasenor-Gonzalez, Y. Ge, and L. Lamont. HOLSR: A hierarchical proactive routing mechanism for mobile ad hoc networks. *IEEE Communications Magazine*, vol. 43(7), pages 118–125, July 2005.

35. M. Voorhaen, E. Van de Velde, and C. Blondia. MORHE: A transparent multi-level routing scheme for ad hoc networks. In *Challenges in Ad Hoc Networking*, K. Al Agha, I. Guérin Lassous, and G. Pujolle, editors, In *IFIP International Federation for Information Processing*, vol. 197, pages 139–148. Springer Boston, 2006.
36. B. Wu, J. Chen, J. Wu, and M. Cardei. A survey of attacks and countermeasures in mobile ad hoc networks. In *Wireless Network Security, Signals and Communication Technology*, Y. Xiao and X. S. Shen and D. Du, editors, pages 103–135. Springer US, 2007.
37. 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.
38. J. Yi, E. Cizeron, S. Hamma, B. Parrein, and P. Lesage. Implementation of multipath and multiple description coding in OLSR. CoRR, abs/0902.4781, 2009.
39. J. Yi, S. David, H. Adnane, B. Parrein, and X. Lecourtier. Multipath OLSR: Simulation and Testbed. In *5th OLSR Interop/Workshop*, Vienna Autriche, 10 2009.
40. J. Yi, A. Adnane, S. David, and B. Parrein. Multipath optimized link state routing for mobile ad hoc networks. In *Ad Hoc Networks*, vol. 9(1), pages 28–47, 2011.
41. X. Zhou, Y. Lu, and B. Xi. A novel routing protocol for ad hoc sensor networks using multiple disjoint paths. In *Broadband Networks, 2005. BroadNets 2005. 2nd International Conference on*, vol. 2, pages 944–948, Boston, MA, October 2005.