MITIGATING NODE ISOLATION ATTACK IN OLSR PROTOCOL USING DCFM TECHNIQUE

C. Selvaraj* & R. Shanthakumari**

* M.Tech Information and Cyber Warfare, Department of Information Technology, Kongu Engineering College, Perundurai, Tamilnadu

** Assistant Professor, Department of Information Technology, Kongu Engineering College, Perundurai, Tamilnadu

Cite This Article: C. Selvaraj & R. Shanthakumari, "Mitigating Node Isolation Attack in OLSR Protocol Using DCFM Technique", International Journal of Advanced Trends in Engineering and Technology, Page Number 75-82, Volume 1, Issue 2, 2016.

Abstract:

A Mobile Ad Hoc Network (MANET) is a collection of mobile devices which are connected by wireless links without the use of any fixed infrastructures or centralized access points. The Optimized Link State Routing (OLSR) protocol is an important proactive routing protocol designed for mobile ad hoc networks. It employs periodic exchange of messages to maintain topology information of the network at each node. Based on topology information, each node is able to calculate the optimal route to a destination. One major DoS attack against the Optimized Link State Routing protocol (OLSR) known as the node isolation attack occurs when topological knowledge of the network is exploited by an attacker who is able to isolate the victim from the rest of the network and subsequently deny communication services to the victim. The proposed method named Denial Contradictions with Fictitious Node Mechanism (DCFM) relies on the internal knowledge acquired by each node during routine routing, and augmentation of virtual (fictitious) nodes. Moreover, DCFM utilizes the same techniques used by the attack in order to prevent it. DCFM successfully prevents the attack, specifically in the realistic scenario in which all nodes in the network are mobile.

Key Words: MANET, OLSR, Fictitious Node & Node Isolation Attack

1. Introduction:

A Mobile Ad Hoc Network (MANET) is a collection of mobile devices which are connected by wireless links without the use of any fixed infrastructures or centralized access points. In MANET [3], each node acts not only as a host but also as a router to forward messages for other nodes that are not within the same direct wireless transmission range. Each device in a MANET is free to move independently in any direction, and will therefore change its links to other devices frequently. Routing protocols in MANET can be classified into two categories: Reactive protocol [4] and Proactive protocol [4]. In proactive routing protocols, all nodes need to maintain a consistent view of the network topology. When a network topology changes, respective updates must be propagated throughout the network to notify the change. In reactive routing protocols for mobile ad hoc networks, which are also called "on-demand" routing protocols, routing paths are searched for, when needed.

- ✓ Reactive routing protocols (e.g., AODV) and
- ✓ Proactive routing protocols (e.g., OLSR)

The Optimized Link State Routing (OLSR) protocol is an important proactive routing protocol designed for mobile ad hoc networks. It employs periodic exchange of messages to maintain topology information of the network at each node. Based on topology information, each node is able to calculate the optimal route to a destination. In OLSR, routes are immediately available when needed. The key concept of the protocol is the use of "MultiPoint Relays" (MPR) [7]. Each node selects a set of its neighbor nodes as MPR. Only nodes, selected as such MPRs, are responsible for generating and forwarding topology information, intended for diffusion into the entire network. MPRs provide an efficient mechanism for flooding topology information by reducing the number of transmissions required. The protocol is best suitable for large and dense network as the technique of MPRs work well in this context. The core functionality of OLSR includes neighbor sensing, multipoint relays selection, topology diffusion and routing table calculation. For neighbor sensing, the HELLO messages are broadcasted periodically. The HELLO [1] messages are broadcasted only one hop away and are not forwarded further. These messages are used to obtain the information about neighbors. The idea of multipoint relays is to minimize the overhead of flooding messages in the network by reducing redundant retransmissions in the same region. Each node in the network selects a set of nodes in its 1-hop neighbors which may forward its messages. In this paper we review a specific DOS attack called node isolation attack and propose a new mitigation method. Our solution called Denial Contradictions with Fictitious Node Mechanism (DCFM) relies on the internal knowledge acquired by each node during routine routing, and augmentation of virtual (fictitious) nodes. Moreover, DCFM utilizes the same techniques used by the attack in order to prevent it.

2. Background and Related Work:

A. OLSR Overview: Network overhead , Unicast routing and dynamic network topology are some of the challenges faced by Mobile Ad Hoc Networks (MANET's).Optimized Link State Routing (OLSR) [8] Protocol is one of the algorithms widely used today. Although OLSR is efficient in bandwidth utilization and in path calculation, it is vulnerable to various attacks. OLSR protocol optimize a pure link state protocol for mobile ad hoc network by reducing the size of control packets, instead of declaring all links, its declares only a subset of links with its neighbors and by minimizing flooding of control traffic by using only selected nodes, called Multi Point Relays (MPR). Only the MPR node retransmits its broadcast messages. As OLSR relies on the cooperation between network nodes, it is susceptible to a few colluding rogue nodes, and in some cases even a single malicious node can cause routing havoc. These attacks include link withholding attacks, link spoofing attacks, flooding attacks, wormhole attacks, replay attacks, black-hole attacks, colluding mis-relay attacks, and DOS attacks. Being a proactive protocol, OLSR periodically circulates topological information. In OLSR, Node isolation attack is a kind of Denial Of Service (DOS) attack whose goal is to isolate a node from communicating with other nodes within that network. In this attack, attacker prevents victim node or group of nodes route information by dropping TC message. Thus, other nodes could not able to receive route information of the effected node so those nodes become not reachable or isolated from network.

B. Node Isolation Attack: In Node Isolation attack [3], an attacker exploits the fact that the victim prefers a minimal MPR set in order to hide the existence of the victim in the network. The attacker, which must be located within broadcast distance of the victim, advertises a fake HELLO message claiming to be in close proximity to all of the victim's 2-hop neighbors. In addition, a fictitious node is advertised, giving the attacker an advantage over other possible legitimate candidates for MPR selection. Knowledge of the victim's 2-hop neighbors is readily available by analyzing TC messages of the victim's 1-hop neighbors, a list of which can be constructed directly from the HELLO message broadcast by the victim himself. MPR selection rules would cause the victim to exclusively select the attacker as its sole MPR, as it is the minimal set that allows for coverage of all of the victim's 2-hop neighbors (including the fictitious node). The attacker can isolate the victim simply by not including the victim in its TC message. In essence, the attacker refrains from notifying the network that the victim can be reached through it, and because no other node advertises a path to the victim, it is isolated. Other nodes, not seeing link information to the victim, would conclude that it has left the network, and remove its address from their routing tables. Although nodes 1- and 2-hops from the victim would continue to exchange information with it, they will not propagate that information further as they were not designated as its MPR.

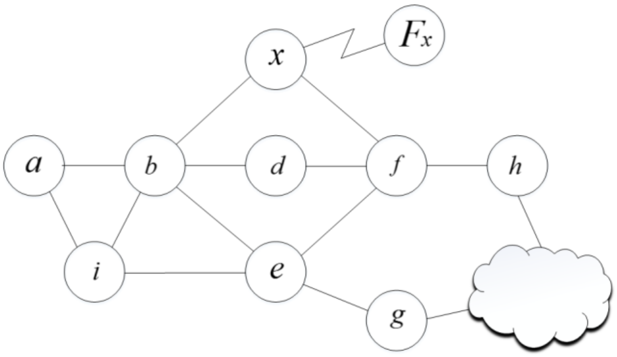

Figure 1: Example of Node Isolation Attack

Figure 1 [4] Assume all nodes within broadcast distance have an edge connecting them, that node x is the attacker, that Fx is a fictitious node, and that node b is the victim. The cloud in the Figure represents the rest of the network. OLSR rules state that x should have advertised a legitimate HELLO message containing {b, f}. Instead, it sends a fake HELLO message that contains {b, f, g, Fx}. This list contains all of b's 2-hop neighbors, as well as one non-existent node, Fx. b would now innocently select x as its sole MPR, setting the ground for node isolation. By not advertising b in its TC message, x effectively isolates b from the rest of the network.

C. Related Work: There have been a number of solutions proposed in the literature for mitigating node isolation attack. In [20], Kannhavong et al. attempt to mitigate the problem of colluding attackers. By modifying the HELLO message to include all 2-hop neighbors, a node can detect existing contradictions between messages, thus identifying an attack. Of course, it is difficult to distinguish between contradictions which occur due to an attack as opposed to those resulting from topology changes. In addition, such contradictions identify an attack but fail to identify the attacker. Raffo et al. [11] propose a mechanism to improve the security of the OLSR routing protocol against external attackers. In their solution, each node signs its HELLO and TC messages. These signatures are later used by others to prove their own HELLO and TC messages. The resulting solution prevents devices from declaring imaginary links with known nodes. This solution functions correctly but is expensive in terms of overhead; besides the usual overhead of OLSR, signing messages requires extensive computation, a cumulative factor that grows as the size of the network increases. Another problem is the fact that the network loses its spontaneity as all nodes are required to know each other in advance in order to share their public keys. This prevents the network from evolving naturally from the various nodes that appear at a certain place and time, a fundamental trait of MANETs. Another approach, based on local detection of link spoofing, is given by [16]. The authors provide a number of rules to identify abnormal behavior on the network. The solution includes a message sent in response to the detection of an intrusion, allowing for the exclusion of compromised nodes and preventing them from being included in network-wide routing tables. Besides the limited scope of the solution, as identification effectiveness is constrained to local nodes only, the ability of sending a warning message is disastrous in itself. Any malicious node can falsely advertise that some other node, local or remote, is malicious, causing for its immediate removal from routing tables all around. In a sense, the solution opens up an attack vector not present in the original problem. Dhillon et al. [10] present an Intrusion Detection System (IDS) in which each node evaluates non-conformances of TCs with respect to previously known HELLO messages. This solution is effective under the assumption that HELLO messages can be trusted. In node isolation attack, however, the HELLO message itself is the problem. Indeed, the authors themselves mention the works of [11] and [20] as a methods for preventing spoofing attacks in HELLO messages. But, as we already mentioned, [11] adds overhead to the network, as does [20] by using control messages for verifying the HELLO messages. [21] Attempts to validate every node mentioned in the HELLO message a node receives. This is accomplished by adding two new control messages which are used for node verification. Upon receiving a new HELLO message, the would-be victim sends a 2-hop verification request through pre-existing channels to every node claimed by the potential MPR (the attacker) to be its neighbor. In response, the queried nodes reply with their 1-hop neighbor list. If the sender is present in all the reply messages, the node deduces that it's legitimate and can appoint it as MPR if it wishes. Otherwise, an attacker has been identified, and the presence of a malicious node is broadcast to the network. The attacker is subsequently removed from the routing tables throughout the network. In [9], Suresh et al. investigate collusion attack in OLSR based MANETs. They propose a method called Forced MPR switching (FMS-OLSR) which requires that a node having a single MPR periodically change its MPR selection; thus, eliminating the necessary pre-condition for node isolation attack. This method might cause a legitimate network to temporarily fragment and is further limited because mitigation can only occur after the attack has commenced.

3. Denial Contractictions with Fictitous Node Mechanism:

The first requirement of the proposed method is that each node will only use information available to it, without relying on any centralized or local trusted authority. This technique does not actively verify the HELLO message [2], rather it checks its integrity by searching for contradictions between the HELLO message and the known topology. This allow for lone MPR nominations, provided that no contradictions are found. Even in the face of contradictions, an MPR can be nominated for all 2-hop neighbors for which it is the sole access point. It cannot, however, be nominated as sole MPR for 2-hop neighbors that can be reached through other paths.

A. Notations Used: The following notations are used

- ✓ V denote the set of all nodes in the network,
- ✓ v, x ∈ V are the victim (as well as/or the receiver) and attacker nodes, respectively,
- ✓ Fx is a fictitious node advertised by x,
- ✓ ADJ(v) ⊂ V is the set of all 1-hop neighbors of v,
- ✓ ADJ2 (v) ⊂ V is the set of all 2-hop neighbors of v,
- ✓ MPR (v) ⊆ ADJ (v) is the set of 1-hop nodes of v who appointed v as their MPR, and
- ✓ MPR' (v) ⊆ ADJ (v) is the set of 1-hop nodes who were selected by v as MPRs.

B. Contradiction Rules: In this section we describe the rules that must be satisfied in order for a node to deem a HELLO message's sender trustworthy.

Consider Figure 4.3 where ADJ (v) = {b, c, x} and ADJ2 (v) = {d, e}. Based on OLSR, v must select MPR (v) = {b, c} so that ADJ2 (v) is covered. Suppose x is interested in isolating victim v. x declares a fake HELLO message containing ADJ (x) = {v, d, e, Fx}. The Rules are,

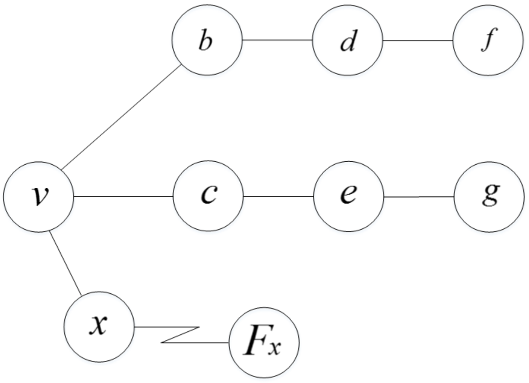

Figure 2: Identifying contradictions to prevent node isolation attack

- ✓ When node x advertises a HELLO message containing ADJ(x), v should confirm that all of the nodes declared by x are not among ADJ (v). As nodes b and c must exist in ADJ2(x), x must select MPRs that will allow it to reach these nodes. It might be the case, however, that x will pretend that it wants to choose v itself as MPR for covering b and c.
- ✓ For each node y mentioned in a HELLO message, v should examine whether there exists z ∈ ADJ(y), such that (a) it is not mentioned in the sender's HELLO message and (b) is located at least three hops away from v. If these conditions are fulfilled, another examination is needed: (c) has x appointed w ∈ ADJ(x) as MPR for covering z?

Examinations (a) and (b) could be done by searching within the TC table. If an entry containing the MPR that was appointed by x and allows it to reach z in only two hops does not exist, then there is a contradiction. Note that contradictions cannot be detected in cases in which either condition (a) or (b) is not fulfilled. In order to verify (c), v has to check each z 2 Z, where Z _ ADJ2(x) is based on x's message, and determine whether there is a TC message where either x has appointed z or z has appointed x as MPR. A test of the condition (2a-c) is represented in Algorithm 1, where the format of the TC message is {last (address), dest (address)}.

Algorithm 1 Testing-Condition-2

```
Testing-Condition-2 (TC,G,x,v)
Z ⟵ Φ
   for each r ∈ TC do
           if r.last ∈ ADJ(x) do
               Z ⟵ Z ∪ {r.dest}
           if r.dest ∈  ADJ(x) do
               Z ⟵ Z ∪ {r.last
   for each z ∈ Z do
           if z ∈ Z ∩ADJ(v) do
               Z ⟵ Z – {z}
   for each m ∈ MPR'(x) do
   for each z ∈ Z do
           if {m, x} ∈ TC or {x,m} ∈ TC such that z is covered by m do
               Z ⟵ Z – {z}
           if Z ≠ Φ do
   mark x as a suspected node
else
   mark x as a legitimate MPR
```

3) v must treat a HELLO message containing all ADJ(v) as an attack and take appropriate measures.

C. Using Fictitious Node: This detects contradictions between a HELLO message and the network topology as is known to v based on proceeding HELLO and TC messages. However, verify every node that was mentioned in the HELLO message. As a result, there are scenarios where node isolation attack is still feasible. Consider, Figure 4.4 in which x advertises that ADJ(x) = {v, e, c, g}, lying about the node c. ADJ2(x) = {b, c, d, i , h}, and MPR (x) = {e, g}.

v cannot identify any contradiction because:

- ✓ x doesn't claim to know any node, other than itself, contained in ADJ(v) (rule No. 1),
- ✓ x appointed MPRs for reaching all of ADJ2(x), namely, {b, c, d, i, h}. Thus, it is expected that x wouldn't appoint c as one of its MPRs, as d is already reachable by e (rule No. 2), and
- ✓ x doesn't claim to know all of ADJ(v), specifically {b} (rule No. 3).

Unfortunately, if every node in the network were to declare an additional fictitious node, the network would revert to LSR (Link-State Routing), as all nodes would be nominated as MPRs due to their fictitious advertisement. Therefore, a mechanism for limiting fictitious announcements must be crafted, balancing between the need for node (and MPR) minimization and protecting the network from isolation attack

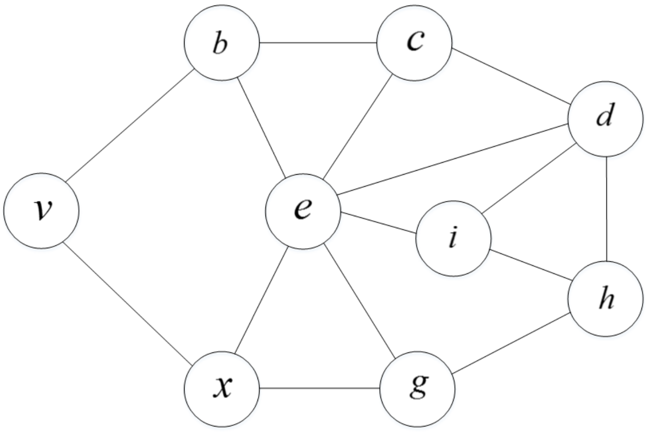

Figure 3: Example of node isolation attack with no contradictions

D. Fictitious Setting Mechanism: In order to prevent nodes in the network from disseminating false information about their connectivity to the others, we set up a mechanism requiring each node to check whether an attack can be made through it. If such a lie is possible, the node adds a fictitious node to the network, preventing anyone from claiming false connectivity to this fake node. That is, responsibility for correctness of the connectivity information is delegated to the nodes themselves, as they must inhibit others from using them falsely. The limitation mechanism for adding or removing fictitious nodes is given by:

- ✓ Each node v has to add a fictitious node when every node in ADJ2 (v) ∈ ADJ(v) such that the distance between y and z <3-hops.
- ✓ Fv ∉ ADJ(v)
- ✓ New node z, advertises Fz by default, and only then calculates rule 1.
- ✓ Removing the fictitious node is done when (1) is false.

- ✓ Examination must be performed periodically (every FICTITIOUS_CHECK_INTERVAL).

Nodes {x, i} ⊂ADJ2(c) in Figure 3 do not contain any node with a distance of 3 from each of them. Therefore, based on rule No. 1 of the fictitious setting mechanism, node c must add a fictitious node to the network. This counters the attack and protects node v, as node x must appoint c as an MPR in order to reach Fv. This will contradict rule No. 2 of the contradiction rules (section III-A) and will be flagged.

4. Result Analysis:

The first set of simulations was designed to test the effectiveness of DCFM against node isolation attacks. For this purpose we ran three different simulations: without movement, with movement using a single attacker, and with movement when a colluding attack is in progress. Each of the first two simulations used 30 nodes in random topology in an area of 750x1000m. In addition, three predefined nodes were used: the victim, the attacker, and a sender used for sending messages to the victim. Both the victim and the sender were randomly placed at a distance of at least 3-hops from each other. The reason for this restriction is because according to the OLSR RFC [4], packets sent from a distance of one or two hops do not use the TC table and are thus not affected during the isolation attack. The attacker, however, was designed to follow the victim. The transmission range was about 250 meters. The following are the parameters which are taken into account for comparison

- ✓ Number of required MPR
- ✓ Average TC message
- ✓ Amount of fictitious node

A. Number of Required MPR: Figure 4 presents the overhead costs as the number of nodes in the network grows. The X-axis represents the number of nodes in a random network topology, while the Y-axis represents the percentage of nodes, again, as a function of nodes in the network that were selected as MPRs. Overhead is any combination of excess or indirect computation time, memory, bandwidth, or other resources that are required to attain a particular goal. It is a special case of engineering overhead. The comparison take place normal OLSR and DCFM activate OLSR.

Density	Base	DCFM
50	0.4	0.71
100	0.28	0.59
150	0.21	0.52
200	0.18	0.48
250	0.15	0.39
300	0.12	0.27
350	0.11	0.14
400	0.1	0.11

Table 1: Network Density vs DCFM

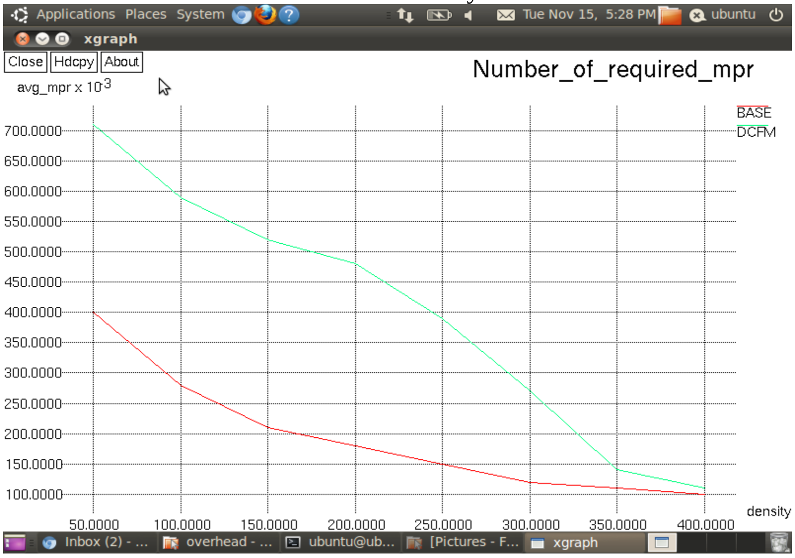

Figure 4: Number of required MPRs, depending on the network density.

B. Average TC Message: Figure5 shows the average TC message of Link state routing protocols, Normal OLSR, DCFM active protocols. TC represents the average size of a TC message when DCFM is active, and LSR is the average size of a TC message in Link-State Routing protocol. When there is no movement, fewer messages are naturally lost; increasing the success metrics. [18]This is the reason why in the third simulation, when no attack was carried out, the results are substantially better than the same simulation under attack with the protection of DCFM. It also explains why some of the messages get through even though the network is under attack and no protection is applied.

Table 2: Network Density vs DCFM vs LSR

Density	BASE	DCFM	LSR
50	3	8	9
100	4	10	15
150	5	12	23
200	4	12	30
250	6	11	38
300	7	11	44
350	5	11	52
400	8	10	59

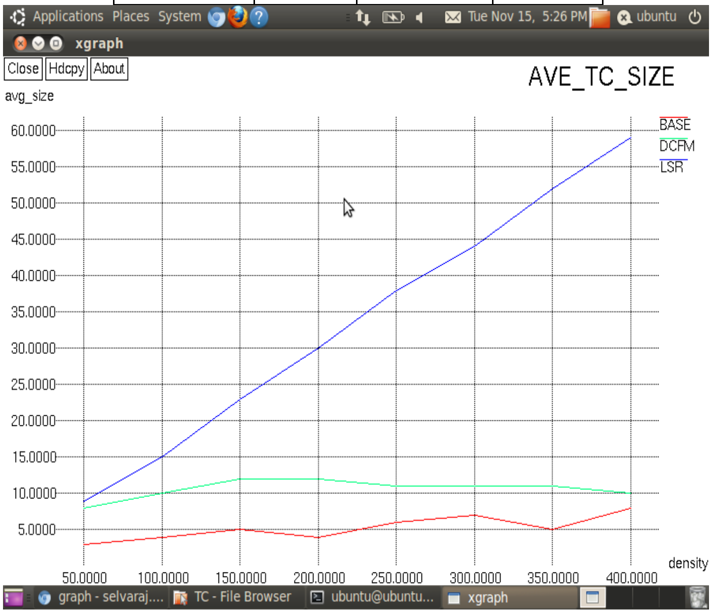

Figure 5: The average size of a TC message, depending on the network density

C. Amount of Fictitious Node: Figure 6 shows a number of fictitious nodes was estimated based on the network density. While the X-axis represents the number of nodes in a random network topology, the Y-axis represents the average percentage of fictitious nodes as a function of the number of nodes that were added to the network (the actual overhead).

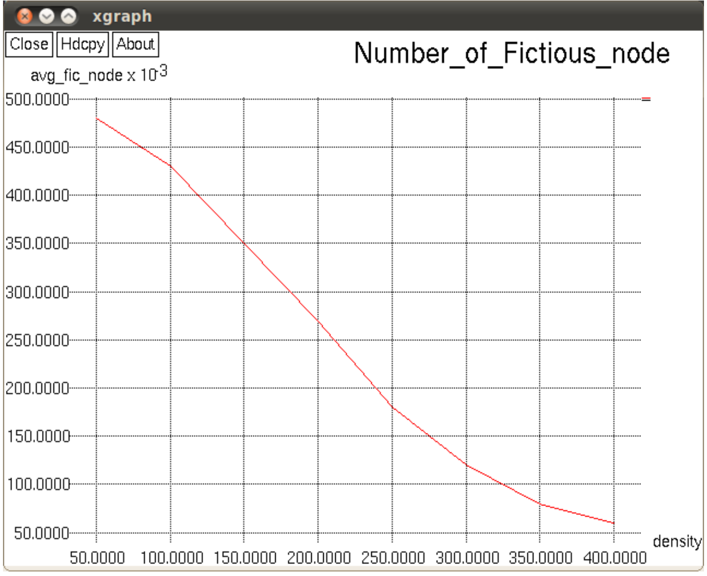

Figure 6: Number of required fictitious nodes, depending on the network density

According to the third simulation settings, attacking nodes are fixed without any movement. This explains why the values of the third row, both when attack is executed and when it isn't are better than the values in the second row (all nodes are moving). The discrepancy between when under attack and when attack isn't executed can be attributed to a real bottleneck in the network; a situation that is independent from the attack and defense. Thus, even with the protection of DCFM, the results are slightly worse than the same scenario without attack.

Table 3: Network Density vs Average fictitious node

Density	Average Fictitious Node
50	0.48
100	0.43
150	0.35
200	0.27
250	0.18
300	0.15
350	0.08
400	0.06

To conclude, our simulations show that

- ✓ DCFM effectively defends OLSR from node isolation attack, even when every node in the network is allowed to move.
- ✓ Basically, there are two approaches to protecting ad-hoc networks: prevention-based approaches and detection based approaches [7]. DCFM belongs to the first class since it prevents the attack by not appointing the attacker as MPR.
- ✓ Figure 4 clearly shows that as the population grows, the percent of overhead decreases. Moreover, as the population density increases, the difference between DCFM.MPR and R.MPR decreases.

5. Conclusion:

The proposed method named Denial Contradictions with Fictitious Node Mechanism (DCFM) whose function is to prevent a node isolation attack in which the attacker manipulates the victim into appointing the attacker as a sole MPR, giving the attacker control over the communication channel. DCFM utilizes the same techniques used by the attack in order to prevent it. Simulation shows that DCFM successfully prevents the attack. Node population increases in density and size, the closer DCFM overhead is to OLSR. Given that OLSR functions best in dense large networks, DCFM can function without real additional cost. In future, that with only minor adjustments, DCFM can protect OLSR from the family of attacks that centers around the falsification of HELLO messages with the intention of being appointed as sole MPR (e.g., black hole [7], gray hole [12], and wormhole attacks [7]).

6. References:

1. De Sousa R, Adnane A, Bidan C (2009), “Trust-based countermeasures for securing OLSR protocol”, Elsevier on Computer Communications, Vol. 2, pp. 745– 752.
2. Jamalipour A, Kannhavong B, Nakayama H, Nemoto Y and Kato N (2007), “A survey of routing attacks in Mobile Ad Hoc Networks”, IEEE Wireless Communications, vol. 14, no. 5, pp. 85–91.
3. Kannhavong B, Nakayama H, Kato N, Nemoto Y, and Jamalipour A (2006), “Analysis of the node isolation attack against OLSR-based Mobile Ad Hoc Networks”, in Proceedings of the Seventh IEEE International Symposium on Computer Networks, pp. 30–35.
4. Aschenbruck N, Gerhards-Padilla E, Martini P, Jahnke M and Tolle J (2007), “Detecting black hole attacks in tactical MANET using topology graphs”, IEEE Conference on Local Computer Networks, pp. 1043–1052.
5. Nadeem A and Howarth M (2013), “Protection of MANETs from a range of attacks using an intrusion detection and prevention system”, in Advances of Telecommunication System, Springer, Vol. 52, no. 4, pp. 2047–2058.
6. Tamil selvi K (2014), “The secured OLSR protocol for MANET”, IEEE Transactions on Information Communication and Embedded Systems, pp. 1 – 6.
7. Yang H, Luo H, Ye F, Lu S and Zhang L (2004), “Security in Mobile Ad Hoc Networks: challenges and solutions”, IEEE Wireless Communications, Vol. 11, no. 1, pp. 38–47.
8. Marimuthu M and Krishnamurthi I (2013), “Enhanced OLSR for Defense against DOS attack in Ad Hoc Networks”, in Proceedings of IEEE Journal of Communications and Networks, Vol. 15, pp. 31–37.
9. Suresh P, Kaur R, Gaur M, and Laxmi V (2010), “Collusion attack resistance through forced MPR switching in OLSR”, IEEE Potentials, pp. 1–5.
10. Malik D, Mahajan K and Rizvi M (2014), “Security for node isolation attack on OLSR by modifying MPR selection process”, IEEE Transactions on Networks Soft Computing, pp. 102–106.
11. Omar M, Challal Y and Bouabdallah A (2009), “Reliable and fully distributed trust model for Mobile Ad Hoc Networks”, Elsevier on Computers & Security, Vol. 28, pp. 199 – 214.
12. Jacquet P,Muhlethaler P,Clausen T,Laouiti A,Qayyum A, and Viennot L (2001), “Optimized link state routing protocol for ad hoc networks,” in Multi Topic Conference. Technology for the 21st Century. Proceedings. IEEE International, 2001, pp. 62–68.

13. Dhillon D, Randhawa T, Wang M, and Lamont L (2004), “Implementing a fully distributed certificate authority in an olsr manet,” in Wireless Communications and Networking Conference. WCNC. 2004 IEEE, vol. 2, March 2004, pp. 682–688 Vol.2.
14. Jeon Y, Kim T.-H, Kim Y, and Kim J (2012), “Lt-olsr: Attack-tolerant olsr against link spoofing,” in Proceedings of the 2012 IEEE 37th Conference on Local Computer Networks (LCN 2012), ser. LCN ’12. Washington, DC, USA: IEEE Computer Societ, pp. 216–219. [Online]. Available: http://dx.doi.org/10.1109/LCN.2012.6423612
15. Raffo D (2005), “Security schemes for the olsr protocol for ad hoc networks,” Ph.D. dissertation, INRIA.
16. Nadeem A and Howarth M (2013), “A survey of MANET intrusion detection & prevention approaches for network layer attacks,” Communications Surveys Tutorials, IEEE, vol. 15, no. 4, pp. 2027–2045.
17. Nadeem A and Howarth M. P (2014), “An intrusion detection & adaptive response mechanism for manets,” Ad Hoc Networks, vol. 13, Part B, no. 0, pp. 368–380.[Online].Available: http://www.sciencedirect.com/science/article/pii/S1570870513001959
18. Hong F, Hong L, and Fu C (2005), “Secure olsr,” in Advanced Information Networking and Applications. AINA 2005. 19th International Conference on, vol. 1, March 2005, pp. 713–718 vol.1.
19. Wang M, Lamont L, Mason P, and Gorlatova M (2005), “An effective intrusion detection approach for olsr manet protocol,” in Secure Network Protocols. (NPSec). 1st IEEE ICNP Workshop on, Nov 2005, pp. 55–60.
20. Kannhavong B, Nakayama H, and Jamalipour A (2006), “Nis01-2: A collusion attack against olsr-based mobile ad hoc networks,” in Global Telecommunications Conference. GLOBECOM ’06. IEEE, Nov 2006, pp. 1–5.
21. Raffo D, Adjih C, Clausen T, and Mühlethaler P (2004), “An advanced signature system for OLSR,” in Proceedings of the 2Nd ACM Workshop on Security of Ad Hoc and Sensor Networks, ser. SASN ’04. New York, NY, USA: ACM, pp. 10–16. [Online]. Available: http://doi.acm.org/10.1145/1029102.1029106