LT-OLSR: Attack-Tolerant OLSR Against Link Spoofing

Yuseok Jeon, Tae-Hyung Kim
The Attached Institute of ETRI
Daejeon, Korea
Email: jys0710@ensec.re.kr, kth80@ensec.re.kr

Yuna Kim, Jong Kim
Division of IT Convergence Engineering
Pohang University of Science and Technology
Pohang, Korea
Email: existion@postech.edu, jkim@postech.edu

***Abstract*—Optimized Link State Routing is a routing protocol that has been extensively studied for mobile ad-hoc networks. Link spoofing, which disturbs the routing service, is one of the critical security problems related to the OLSR protocol. Existing approaches against link spoofing attack have several drawbacks. In this paper, we propose an LT-OLSR protocol that broadcasts Hello messages to neighbors within two-hops to defend networks against link spoofing attacks. Simulation and analysis results show that the LT-OLSR protocol tolerates link spoofing attacks extensively. The contributions presented in this paper are as follows: (1) We design a mechanism to ensure the integrity of a routing table. (2) In addition, our approach against link spoofing attack is not only extensive but also compact. (3) Finally, it can be practically implemented under various types of MANET environments.**

I. Introduction

In MANETs (Mobile Ad-hoc Networks), there is no infrastructure to deliver communication messages. Thus, many studies have focused on routing protocols specifically optimized for MANETs. OLSR (Optimized Link State Routing) [1] has been known by the IETF (Internet Engineering Task Force) as one of four critical routing protocols. However, the OLSR protocol is susceptible to many security threats, such as link spoofing, identity spoofing, relay and wormhole attacks. Malicious attackers just need to generate forged control messages containing false link information to initiate link spoofing attack. These malicious control messages result in the contamination of routing tables, which causes a couple of problems such as unreachable nodes and link loops. Link spoofing attacks have become a major threat to the security of OLSR, and are thus the focus of this paper. Although a number of approaches to defend link spoofing attack have been proposed, they have several disadvantages including an excessive overhead, vulnerability of being contaminated routing table problem, impracticality, and limited number of defensible attacks. We propose a LT-OLSR protocol to defend the networks against link spoofing attacks over shortcomings of existing approaches. In our proposed approach, each node is required to communicate with every node within two-hops in order to exchange the information for the identification of link spoofing attacks. Through simulation and mathematical analysis later in this paper, we emphasize the effectiveness of our approach.

The remainder of this paper is organized as follows. In Section II, we introduce a brief background about the OLSR protocol and link spoofing attacks. In Section III, we discuss related work on OLSR security issue, and in Section IV, we elaborate our defense approach against link spoofing attacks. In Section V, we verified the effectiveness of our approach through simulation and analysis. Finally, Section VI concludes this study.

II. Background

A. Optimized Link State Routing Protocol

OLSR is a proactive link state routing protocol for MANETs. Every node using OLSR in the networks exchanges control messages periodically to notify topological information among themselves about the networks. Upon collecting messages, the source nodes calculate optimal routes to the destination nodes. We now examine the key aspects of the OLSR protocol in detail.

1) MPR (Multipoint Relay) node: Each node selects some of its one-hop neighbors as MPR nodes that relay its messages to two-hop neighbors. Each node collects information from one and two-hop neighbors based on Hello messages from one-hop neighbors. Each node then selects a set of MPR nodes, which is a minimum subset of one-hop neighbors to cover the nodes within two hops away neighbors, to transmit messages to all the two-hop neighbors via the MPR nodes.

2) Control messages: There are two types of control messages in OLSR: Hello and TC (Topology Control) messages. These messages are periodically broadcasted for routing tables of each node in a network. The Hello message every node transmits plays two roles. The first one is to identify its neighbors. The second role of the Hello message is to notify the MPR nodes that some nodes selected them as MPR nodes. Also, each MPR node periodically broadcasts TC messages to build all other nodes' topology tables and maintain the information on the set of MPR selector nodes in the TC messages.

B. Link Spoofing Attack

In this section, we describe that link spoofing attacks can be classified into two types: Hello and TC message link spoofing attacks.

978-1-4673-1564-7/12/$31.00 ©2012 IEEE

1) Hello message link spoofing attacks: There are three classes of Hello message spoofing attacks.

- **Adding non-existent node information** A malicious node can inject adversarial information of non-existent nodes using Hello messages.
- **Adding non-neighbor node information** A malicious node can insert false information about neighbors into Hello messages in order to claim that they are its one-hop neighbor nodes.
- **Deleting existent neighbor node information** A malicious node can remove information about its one-hop neighbor nodes by exploiting Hello messages.

2) TC message link spoofing attacks: TC message link spoofing attacks fall into two classification.

- **Adding fake MPR selector information** A malicious node can falsely claim non-neighbor nodes as its MPR selectors by inserting them in TC messages.
- **Deleting MPR selector information** An adversarial node can generate TC messages without the information related to its MPR selectors.

III. Previous Work and Problems

A considerable number of studies have been conducted against link spoofing attacks. They can be fallen into the following four types.

- Using feedback messages: Employing feedback messages from TC messages were introduced to defend link spoofing attacks in [2], [3]. The drawbacks of these two approaches are as follows: (a) contaminated routing tables (b) excessive overhead
- Using a signature: The authors in [4], [5] proposed a signature based defensive approach for OLSR against information forgery attacks. The disadvantages of these approaches are as follows: (a) excessive overhead (b) limited attack set coverage
- Using geographic information: Raffo et al. from [6] proposed an approach that utilizes geographic information calculated by GPS and directional antenna against link spoofing and wormhole attacks. The drawbacks of this approach are as follows: (a) impracticality (b) limited attack set coverage
- Using semantic properties: The authors of [7], [8] designed a countermeasure system employing semantic properties in the protocol definition which specifies correct OLSR behaviors. The disadvantages of these approaches are as follows: (a) limited attack set coverage (b) contaminated routing tables

IV. LT-OLSR

In this section, we propose an LT-OLSR approach that can tolerate link spoofing attacks as compared to the inherent disadvantages of existing approaches. We assume that the following basic attack prevention approaches are in place.

- Identity spoofing attacks can be prevented by a signature-based approach like [9]
- Replay attacks can be prevented by a time-stamp based approach in [10]
- Mis-relay behavior that disturbs the correctness of broadcasted Hello messages up to two-hop neighbors is detected by monitoring the traffic they generate like [11]

A. Overview

The main idea of our approach is to extend the broadcasting range of Hello messages to two-hop neighbors by relaying the messages to verify the legitimacy of the exchanged control messages. The following modification of the OLSR protocol is in need.

1) Modification of the Hello message's broadcasting range: First, we modify the broadcasting count of Hello messages in each node. In the original OLSR, each node periodically broadcasts its Hello messages to one-hop neighbors. However, in our approach, each node not only periodically broadcasts Hello messages to its one-hop neighbors, but also re-broadcasts Hello messages without modification so that Hello messages will reach their two-hop neighbors as well.

2) Trust flag: A new 'trust flag' column is added to the neighbor and topology tables. In our approach, we only consider two-hop neighbor tables due to the fact that one-hop neighbor tables are not affected by link spoofing attacks. If a node receives a two-hop Hello message from its two-hop neighbors, the node updates the trust flag entries of the neighbor tables with true flags after the verification. Each entry in the topology tables represents a symmetric connection between an MPR and its selector node. In our approach, an entry is generated when a node has received TC messages generated from their MPRs or received a Hello message from two-hop neighbors. Upon the creation of the entry by the reception of Hello messages from two-hop neighbors, the trust flag is initially set to false and will be changed to true after reception of the validated TC message.

B. Operations

In this subsection, we describe how this approach works against link spoofing attacks.

1) Tolerating Hello message link spoofing attacks: The OLSR protocol in our approach is tolerant of these all three types of attacks.

- **Adding non-existent or non-neighbor node information** When a one-hop neighbor node adds information about a non-existent neighbor node or non-neighbor node in its Hello messages, an entry is created in the two-hop neighbor table. The trust flag of that entry in the neighbor table will not be changed to true value due to the fact that a non-existent or non-neighbor nodes cannot send Hello messages.
- **Deleting existent neighbor node information** When an attacker remove neighbor information through its Hello messages, the one-hop neighbors of the attacker will receive unexpected two-hop Hello messages broadcasted from the excluded nodes. The trust flag for an excluded

node will therefore be shown as a false value in two-hop neighbor tables.

For the reasons above, our approach can detect three classes of Hello message link spoofing attacks. If the trust flag of an entry is false value, the entry will not be considered as MPR nodes in routing tables.

2) Tolerating TC message link spoofing attacks: There are two types of TC message link spoofing attacks. Our approach can defend both attack types.

- **Adding a fake MPR selector** When a node receives a TC message, the node checks if the originator of the message is a one-hop neighbor node. If it is so, the node validates the TC message by finding all corresponding matching entries in the topology table. Recall that a new entry is created in the topology table upon receipt of a two-hop Hello message. When a matching entry exists, the trust flag of the matching entry will be changed to true value. If there is no matching entry, the TC a message is assumed to be originated by a malicious MPR node.
- **Deleting MPR selector information** The trust flag of an entry created by two-hop Hello messages remains false value if a node has not received matching TC messages from its one-hop neighbor MPR nodes. This situation takes place when existing MPR selector information is removed by the TC message from a malicious MPR node.

For the reasons described above, our approach can defend MANETs against two types of TC message link spoofing attacks. The countermeasure against the forged TC message is that, once detected, forged messages would not be forwarded by the node who has detected and reselect other MPR nodes instead of malicious nodes.

V. Evaluation

In this section, we validate the security level of the proposed approach and evaluate it in terms of tolerable link spoofing attacks, overhead and the ability to maintain the integrity of routing table.

A. Security Analysis

Our approach utilizes two types of information in a two-hop broadcast Hello message. The first type is the originator address information which is used to deal with Hello message link spoofing attacks. The originator address can be verified through a signature-based approach in [9]. The second type is MPR selection information against TC message link spoofing attacks. However, MPR selection attacks attempting to modify this information are beyond the scope of our approach since TC link spoofing attacks occur after the MPR selection is finished and, as can be seen in Section 2, the Hello message link spoofing attack does not modify the MPR selection information. However, the semantic approach [7] can be used to cope with the MPR selection attack.

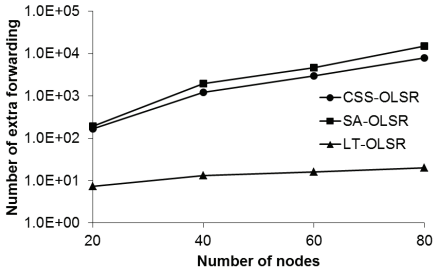

Fig. 1: Number of extra messages forwarded.

B. Overhead Analysis

We now analyze and compare the additional overhead caused by the proposed approach LT-OLSR, and the feedback-based approaches. In our proposed approach, forwarding Hello messages create only additional overhead. However, the feedback-based approaches cause two sets of additional overhead, feedback generation and feedback forwarding. where M is the average number of MPR nodes, N is the total number of nodes in a network, and Ngh_{2-hop} is the average number of two-hop neighbors of a MPR. In CSS-OLSR, each MPR node M receives feedback for a TC message from all nodes except itself N-1. In SA-OLSR, all two-hop neighbor nodes Ngh_{2-hop} of each MPR node M send a feedback message to the MPR nodes after receiving a TC message. Thus, the additional forwarding overhead of CSS-OLSR, SA-OLSR and LT-OLSR can be formalized as follows, respectively:

$$M * (N-1) * H \tag{1}$$

$$M * Ngh_{2-hop} * 2 \tag{2}$$

$$Ngh_{1-hop} \tag{3}$$

where H is the average hop-count between the source and destination nodes and Ngh_{1-hop} is the average number of one-hop neighbors of each node. In CSS-OLSR, a feedback message is forwarded as many as H times from all receivers up to each originator of the TC message. In SA-OLSR, the same number of feedback messages is forwarded to the originator. In LT-OLSR, each receiver of a Hello message rebroadcasts it once again. Fig. 1 shows the number of extra messages forwarded. As the number of nodes increases, the proposed approach becomes more efficient than the others. Because the OLSR protocol is usually used in dense networks for the efficiency of the MPR nodes, this result is significant.

C. Routing Table Integrity Analysis

We now describe the simulation results in terms of the capability to maintain the integrity of the routing tables. We simulated LT-OLSR and CSS-OLSR on PURE-OLSR in [12], which is an OLSR implementation for the NS-2 network simulator version 2.29. In this simulation, we employed Random Waypoint Mobility to construct a general network topology.

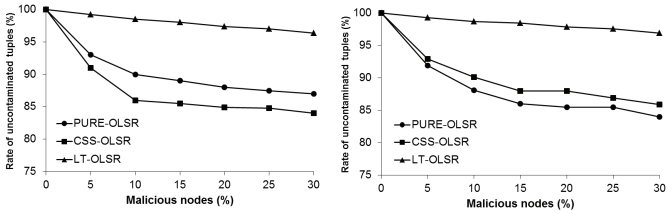

Fig. 2: The rate of correct routing entries in routing table (Left: Node speed = 1.5 m/s, Right: Node speed = 5.0 m/s)

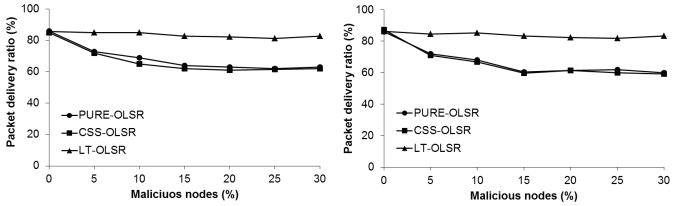

Fig. 3: Packet delivery ratio (Left: Node speed = 1.5 m/s, Right: Node speed = 5.0 m/s)

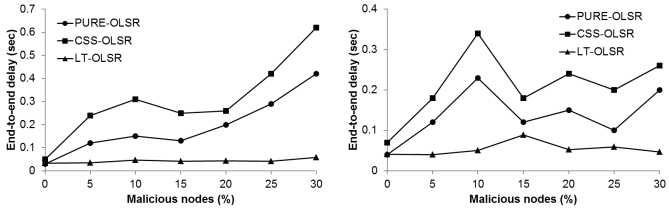

Fig. 4: The end-to-end delay (Left: Node speed = 1.5 m/s, Right: Node speed = 5.0 m/s)

The simulations were performed for 30 nodes with a transmission range of 250m, in a restricted 1000 * 1000m size network for 160s. To avoid bias, we performed 140 simulation runs in total for each OLSR protocol, using ten independent topologies with a set of two different mobility scenarios, 1.5m/s and 5.0m/s and seven different malicious node rate scenarios. In the simulation, the malicious nodes were randomly selected according to the malicious node ratio. A malicious node generates Hello and TC messages that contain false information for link spoofing attacks that we are mentioned above. As shown in Fig. 2, LT-OLSR has the highest rate of correct routing table entries. In CSS-OLSR, routing tables can have incorrect entries due to the late detection, and incorrect routing table entries can seriously degrade the entire network performance obviously. As we can see in Fig. 3, the packet delivery ratio under CSS-OLSR gradually decreases as the number of malicious nodes increases, while LT-OLSR maintains the same packet delivery ratio despite of the increasing number of malicious nodes. Also, as shown in Fig. 4, the end-to-end packet delay is relatively high under attack in CSS-OLSR, while LT-OLSR maintains the delay which is close to the level observed when there is no attack in spite of the increasing number of malicious nodes. From these simulation results, we can confirm that our approach has the capability to maintain better integrity in routing tables, even during link spoofing attacks. This ability can prevent the degradation of the entire network performance in terms of packet delivery ratio and end-to-end delay.

VI. Conclusion

In this paper, we presented a new approach to defend MANETs against many types of link spoofing attacks. Our approach is based on the two-hop broadcasting Hello messages through a process that enables every node to individually verify the legitimacy of received Hello and TC messages. Our evaluation results demonstrated that all nodes in the network can continually maintain uncontaminated routing tables with a low amount of additional overhead, even during a link spoofing attack, and we can therefore conclude that our approach effectively tolerates such attacks. In the future, we will consider a method that reduces the number of forwarded Hello messages, as well as other OLSR protocol issues that may be solved by our approach.

Acknowledgment

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2012-H0301-12-3002) and World Class University program funded by the Ministry of Education, Science and Technology through the National Research Foundation of Korea (R31-10100).

References

[1] T. Clausen and P. Jacquet, "Optimized link state routing protocol (OLSR)," October 2003, IETF RFC 3626.

[2] K. Bounpadith, N. Hidehisa, and A. Jamalipour, "Sa-olsr: Security aware optimized link state routing for mobile ad hoc networks," *Proceedings of ICC 2008*, pp. 1464–1468, 2008.

[3] J. P. Vilela and J. Barros, "A feedback reputation mechanism to secure the optimized link state routing protocol," *Proceedings of SecureComm 2007*, pp. 294–303, September 2007.

[4] D. Raffo, C. Adjih, T. Clausen, and P. Mühlethaler, "An advanced signature system for olsr," *Proceedings of 2nd ACM Workshop on Security of ad hoc and sensor networks*, pp. 10–16, 2004.

[5] S. Rana and A. Kapil, "Defending against node misbehavior to discover secure route in olsr," *Information Processing and Management*, vol. 70, pp. 430–436, 2010.

[6] D. Raffo, C. Adjih, T. Clausen, and P. Mühlethaler, "Securing OLSR using node location," *Proceedings of EW 2005*, April 2005.

[7] A. Adnane, R. T. de Sousa, Jr., C. Bidan, and L. Mé, "Autonomic trust reasoning enables misbehavior detection in olsr," *Proceedings of SAC 2008*, pp. 2006–2013, 2008.

[8] C. Tseng, T. Song, P. Balasubramanyam, C. Ko, and K. Levitt, "A specification-based intrusion detection model for olsr," *Proceedings of RAID 2005*, vol. 3858, pp. 330–350, 2006.

[9] D. Raffo, "Security schemes for the OLSR protocol for ad hoc networks," Ph.D. thesis, Universite Paris, 2005.

[10] C. Adjih, D. Raffo, and P. Mühlethaler, "Attacks against OLSR: Distributed key management for security," *Proceedings of Workshop on 2nd OLSR Interop and Workshop*, July 2005.

[11] S. Marti, T. J. Giuli, K. Lai, and M. Baker, "Mitigating routing misbehavior in mobile ad hoc networks," *Proceedings of MobicCom 2000*, pp. 255–265, 2000.

[12] "UM-OLSR," http://sourceforge.net/projects/um-olsr/.

[13] L. Boudec, J.-Y., and M. Vojnovic, "Perfect simulation and stationarity of a class of mobility models," *Proceedings of INFOCOM 2005*, vol. 4, pp. 2743–2754 vol. 4, March 2005.