An IDS First Line of Defense for Ad Hoc Networks

Alia Fourati
LRI, Université Paris-SUD XI
Paris, France
alia@fourati.com

Khaldoun Al Agha
LRI, Université Paris-SUD XI
Paris, France
alagha@lri.fr

Abstract

Ad hoc networks arouse imminently an important interest within industrial and research communities. Actually, their salient features, essentially the absence of infrastructure and auto-management, promise huge applications and possibilities for wireless communication. However, the tremendous boom of these networks depends incontestably on their reliability in terms of security and quality of services (QoS). In this paper, we study the ad hoc security vulnerabilities for which cryptographic-based solutions are ineffective and which require IDS. Appropriate designed IDS services appear essential as countermeasures to those threats, where previous IDS proposals often seem to be too sophisticated in the ad hoc environment and merely ineffective. Our goal here is to provide an IDS mechanism dedicated to the OLSR protocol, fitting to its characteristics and operation, and designed to avoid its vulnerabilities. Hence, our proposed IDS represents an urgent and primary line of defense since it protects from the protocol flaws themselves avoiding a significant panoply of easily operated attacks.

Keywords: Ad hoc networks, OLSR protocol, vulnerabilities, security services, IDS.

1. Introduction

In a MANET (Mobile Ad hoc Network), the absence of infrastructure, added to the weak range of wireless transmission mediums, lead to the forwarding of messages through intermediate nodes in order to guarantee the routing function. Consequently, all nodes of an ad hoc network operate as routers, and routing protocols become the base of MANETs. However, such cooperation-based routing weaken the routing function because participating nodes can act dishonestly, or can be themselves compromised by external adversaries causing the disruption of the network communication and even its inhibition. Besides, routing protocols have defined neither prevention measures, nor security mechanisms in their specifications. In the OLSR protocol RFC [1] –and other ad hoc routing protocols RFCs [2][3]- this is clearly specified as follows: "currently, special OLSR does not specify any security measures". Securing ad hoc routing protocols appear then as an urgent requirement crucial to promote ad hoc network's deployment and to widen their application domains.

A survey of ad hoc routing vulnerabilities had leaded to classify possible attacks into two main categories. The first category, including attacks such impersonation, traffic sniffing, modification or replay, is due to classic attacks on wireless networks worsened by the absence of centralized entities. The latter require cryptographic mechanisms to provide authentication of nodes and integrity of routing messages. The second category of vulnerabilities is due to attacks inherent to ad hoc networks and which occur even when nodes are authenticated. For example, an authenticated node can announce false neighbors in its control messages inducing false topology tables, then false routing tables, and finally false routing of messages. Such attacks are countered by IDS (Intrusion Detection Systems) techniques.

Being aware of these weaknesses, researcher's community have made many efforts to provide security services especially challenging due to the constraining features of ad hoc environment, i.e. the absence of centralized entity, dynamic topology changing, auto-organization and open wireless communication medium. However, most recent ad hoc routing securing research has focused on providing security services while relying on assumptions and number of pre-setup restrictions which are not fitting to ad hoc operation principles.

In this paper, we propose an IDS scheme for ad hoc routing protocols, especially designed for the OLSR protocol. In the second section, we present a survey of IDS techniques and of most relevant proposals of literature. Section 3 exposes the OLSR protocol operation and presents threat scenarios exploiting flaws of the protocol. In section 4, we define our IDS scheme aiming to avoid attacks using the OLSR specification's weaknesses. This scheme represents an urgent and a first line of defense protecting from unpredictable external maneuvers. Finally, section 5 concludes the paper and releases strong and weak points of our IDS.

2. IDS overview and related works

Intrusion detection is defined as the method to identify "any set of actions that attempt to compromise the integrity, confidentiality, or availability of a resource" [7]. Hence, it concerns nodes that are outside or inside the network attempting to disturb the network functions through attacks that preventions measures such authentication of nodes or integrity of messages can not detect and counter. Considered as the complementary securing protection to prevention measures, IDS does not rely on specific mechanisms, but commonly on the continuous monitoring of the network for unusual activity [8]. The monitoring of the system is generally held in three phases: the collection of audit data, the analysis of collected data and finally the release of an alert when a threat is detected. The differences between existing IDS reside in designing those three phases, i.e. how and where collecting data, how searching intrusions in collected data, and how responding to intrusions. Yet, intrusion detection has been classified into two main categories: anomaly detection and misuse detection [7][8] [9].

Anomaly detection: Anomaly detection bases its mechanisms on the profile of user's normal behavior. It analyzes the user's current session and compares it to the profile representing the user's normal behavior statistically. More generally, since a "normal activity profile" can be established for a system, it is possible to detect anomalous states varying from the established profile. A typical anomaly detection system takes in audit data for analysis. This type of detection systems is well suited to detect unknown or previously not encountered attacks.

Misuse detection: Misuse detection bases its mechanisms on detecting previously known attacks. A typical misuse detection system takes in audit data for analyses and compares it to large databases of attack signatures. Attack signatures are specified as rules with respect to timing information and are also referred to as known attack patterns. This type of detection systems is useful in networks with highly dynamic behavioral patterns, but like a virus detection system, it is only as good as the database of attacks signatures that it uses to compares with.

Other categories of IDS had been defined in literature. A third category called "Specification-based detection" has been defined by Mishra *et al.* in [8] as a set of constraints that describe the correct operation of a program or protocol and monitoring its execution.

Another approach proposed in [9] classifies IDS by separating IDS techniques and IDS architectures. According to authors, IDS techniques refer to misuse and anomaly detection, whereas IDS architecture represents a larger concept. Actually, the latter must involve many modules necessary for the efficient operation of the IDS, including an IDS techniques module, a module on how nodes collaborate in intrusion detection decision making, etc.

With respect to these IDS principles, resulting IDS solutions for ad hoc networks belong to two main groups.

The first one represents IDS systems having modular architecture including at least a data collection module, a detection module and a response module. This is the case of solutions proposed in [7] [10][11] which differ in the number of additional modules and of their functionalities, in the way that modules interact between each other locally and cooperatively with the modules of other nodes, etc.

For ad hoc networks, proposed IDS aim to provide solutions being self-organized, collaborative and without centralized entity. Nevertheless, some requirements of these IDS seem to be inappropriate to ad hoc networks characteristics. For example, in [7] the normal behavior for each node is acquired by the trace data gathered for many simulated normal situations during a training process. We wonder how acquiring the normal behavior for nodes coming after the training process, and if it is acquired from nodes already present in the network, how to be sure that they are honest? Besides, these IDS generally require important databases to store collected audit data and normal behaviors patterns. In addition, implemented algorithms intended to detect intrusions in collected data are complex and involve important memory. Such material requirements are not suited with wireless terminals which are used to have limited resources (energy, memory, CPU power, etc).

The second group of solutions doesn't rely on a particular architecture, but on the general function of an IDS "detecting misbehaviors by observing the networks traffic" to respond to a specific flaw or threat in a MANET. An illustrative example of this approach is the well famous Watchdog and Pathrater of Marti *et al.* [12] which aim to tackle nodes that agree to forward packets but don't do so. This threat is caused by nodes that can be overloaded, selfish, malicious or broken. To mitigate these misbehaviors, authors propose for each node two mechanisms: a watchdog and a pathrater. The watchdog verifies that the next node in the path also forwards the packet by listening to the node's transmissions since it belongs to its radio range. When a node omits forwarding a packet, the watchdog increments a failure count which defines the node as misbehaving if it reaches a threshold. The pathrater combines the watchdog results with link reliability data to supply reliable routes to nodes.

In our work, we opt for the second approach to tackle specific problems within the OLSR routing. We propose indeed an IDS solution completely adapted to the OLSR operation and avoiding its specific flaws.

3. OLSR threat scenarios

3.1. OLSR overview and operation

OLSR (Optimized Link State Routing) protocol is a proactive, link state routing protocol, developed for ad hoc networks. Since its standardization in October 2003, the protocol arouses an increasing interest in research, commercial and military domains [4]. Its proactive nature implies that routes are continually maintained up-to-date, such that when a node requests for sending a message, an optimal route is already available. In order to decrease control messages number, OLSR determines for each node of the network, a minimal subset of neighbors, called MultiPoint Relays (MPR), which are able to reach all 2-hop neighbors of the node. A node has then to transmit its broadcast traffic, to only its MPRs. The OLSR robustness and popularity are mainly due to its MPR principle. Indeed, MPRs significantly minimizes the control traffic in term of packet length and control messages number. In parallel, it optimizes the bandwidth use. The OLSR operation relies on three mechanisms: neighbor sensing, optimized MPRs flooding, and topology diffusion [1].

Neighbour sensing. Due to the radio propagation environment, each node has to qualify links nature with its neighbors (symmetric and asymmetric). To achieve the neighbor sensing, each node broadcasts periodically to its 1-hop neighbors *Hello* messages, which have not to be forwarded. These messages contain the node neighbors list with their link status allowing the deduction of the totality of 2-hop neighbors and of their status. The MPR selection can then be made and the MPR list is included in *Hello* messages. Lastly, the MPR list allows the construction of the MPR selector list which contains the neighbors which have selected it as MPR. Thus, nodes forward only messages received from their MPR selectors.

MPR flooding. The objective of MultiPoint Relays (MPR) is optimizing the control traffic flooding. MPRs are chosen such that, any emitted flooding message, when relayed by the MPR set, must reach all 2-hop neighbors. The MPR set of a node n, denoted MPR(n), represents in other terms the smaller subset of symmetric 1-hop neighbors of n, having symmetric links with all 2-hop neighbors of n. MPR flooding conducts to the elimination of duplicate transmission and of the minimization of duplicate reception.

Topology diffusion. Topology diffusion aims to construct routing tables through periodic topology control messages (*TC* messages). To achieve this task, each node, with a non-empty MPR selector set, must diffuse *TC* messages to all network nodes, announcing at least links between itself and the nodes in its MPR selector set. *TC* messages provide sufficient information, enabling nodes to construct their topology table, and then to deduce their routing table. The routes are calculated with a shortest path algorithm, e.g. the *Dijkstra*'s shortest path algorithm, optimizing thus hops number.

3.2. OLSR threat scenarios

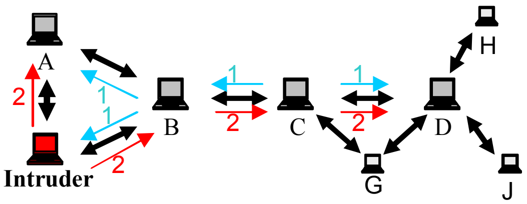
According to the OLSR operation, we note the important role of MPR nodes in the transmission of routing messages. Basically, they represent the only nodes of the network entitled to diffuse routing messages to the other nodes in order to optimize diffusion and thus traffic in the network. Because of this privileged position, malicious nodes tempt to acquire the status of MPR in order to get routing messages deviating towards them. Once intruders appear to the other nodes as an MPR, they can misbehave with received messages by modifying their content, omitting to forward them, etc.

As there exists two main control messages in OLSR (Hello and TC), we present here two threat scenarios targeting each of them. Threats are operated by nodes that are authenticated, and so belonging legitimately to the network. Nevertheless, they decide to act maliciously to appear for other nodes as MPRs without being revealed by the other nodes.

3.2.1 1st scenario: cheating through TC messages

In the first scenario, the intruder is not an MPR and do not have to send TC messages. In spite of this, he generates and diffuses forged TC messages presenting for his 1-hop neighbors sorter routes to reach node D As a result, the intruder will get deviated all messages passing through B to D by him.

The flaw within OLSR here is that when the node D receives the TC message of the intruder, he does not react about the erroneous announcement concerning him, and designing him as having selected the intruder as its MPR, which is false.



A is 3-hop distance from D through B and C
1. When C sends a TC message, the intruder identifies D with 3-hop distance
2. The intruder announces in a forged TC message that D belongs to its MPR Selector set.

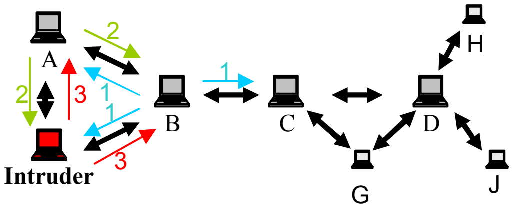
=>Consequences:

- The intruder presents for A a shortest route to reach D than B
- A stops sending traffic to D through B
- A sends traffic to D through the intruder
- The intruder misbehaves with received messages

Figure 1. 1st threat scenario

3.2.2 2nd scenario: cheating through Hello messages

In the second scenario, the intruder is not an MPR, but succeeds to be elected as MPR of the node A by cheating with Hello messages. In fact, he announces a false number of 1-hop neighbors (with whom he has symmetric links) letting know to its 1-hop neighbors that they can reach a more important number of 2-hop neighbors by passing only by him. Hence, they can reduce the number of their MPR and optimize more the traffic.



B is MPR of A. C is 2-hop distance from A
1. B sends a Hello message
2. A sends a Hello message
3. The intruder sends a Hello message announcing that he has symmetric links with A, B, C, D and X (X could be one of the other nodes)

=> Consequences:

- The intruder presents for A a shortest route to reach D than B, and also to reach other destinations (C, X)
- A selects the intruder as MPR
- A stops sending traffic to D through B
- A sends traffic to D through the intruder

Figure 2. 2nd threat scenario

The flaw within OLSR here is that the intruder sends a Hello message announcing its 1-hop neighbors list, to which the node C belongs. In the same way, the node C sends a Hello message announcing its 1-hop neighbors list, which does not include the intruder. This discordance of information is not observed by node B, which could have detected a problem, but does not.

4. An IDS first line of defense for the OLSR protocol

4.1. Security requirements

It is obvious that intrusion detection systems developed for wired networks are unsuitable for wireless networks and for ad hoc networks in particular. This is due to the following reasons:

In wired networks, data monitoring is done at points of data concentration as switches, routers or gateways. These centralized entities are inexistent in ad hoc networks. Besides, the non-clear frontier of ad hoc networks caused by the radio ranges makes the control of exchanges in the network more difficult, and the interception of messages easier.

Besides, the continuous change of topology and links in the network require distributing services through the different nodes forming of the network. The cooperation of nodes could be required too [7] [8] to strengthen the provided services in a spontaneous and autonomous way, such that nodes still free to leave and join the network at any time.

Finally, it is compulsory to consider the limitations of the ad hoc environment. First, the wireless medium implies limitations of bandwidth and transmission rates. Second, wireless terminals present limitations concerning energy, processing and memory.

Considering these characteristics it becomes compulsory to conceive suitable and realistic solutions in the ad hoc environment, where the management and the optimization of overhead appear as a major requirement.

4.2. The solution

The OLSR protocol is designed to only complete the routing function in ad hoc networks since it allows the automatic integration of received control messages information without any check of their veracity. Nevertheless, we have shown in section 3.2 that it is possible to easily divert the protocol operation and to disturb the routing function. Our idea is to avoid such basic and damaging threats by reinforcing the protocol through an Intrusion Detection System. Our IDS acts on control messages by checking the veracity of their content. We suppose here that authentication and integrity are assured and thus messages are not modified during their transmission.

As exposed in section 3.1, nodes operating the OLSR protocol maintain neighborhood information databases which are constructed through two control messages: Hello and TC messages. Hello messages are emitted by each node of the network announcing the list of its 1-hop neighbors. TC messages are flooded by only MPR nodes to all the nodes of the network announcing their MS set, i.e. the set of nodes that have selected them as MPRs. Hence, each node knows at least its (1) 1-hop neighbor set, (2) its 2-hop neighbor set, (3) its MPR set, and (4) its MPR Selector Set. This implies that each node knows a partial graph of the network which is complete at least until a 2-hop range.

To prevent from threats describes in section 3.2, our IDS verifies the veracity of control messages contents in a distributed and collaborative way. Actually, a group of nodes skim through their local information each time a node receives a TC or a Hello message in order to detect possible discordance and then potential threats. If so, a response is launched against intruders.

An intruder I emits a forged TC message if he announces nodes that have not chosen him as MPR, or if he omits to announce nodes that have chosen him as MPR. These nodes can then detect the maneuver since TC messages are diffused in all network.

In the example of the first threat scenario (section 3.2.1), the intruder sends a TC message announcing D belonging to its MS set, meaning that the intruder belongs to the MPR set of D. However, the TC message is flooded in the entire network and when the node D receives this message, he can pick up information about himself and verify in his MPR set the concordance of announcements received in the TC message. He can then observe that the intruder does not exist. As a response, he floods an alert message announcing a potential threat from the intruder. Nevertheless, this solution stills weak since it is based on the only targeted node as detector and which is supposed honest. Hence, new flaws could be introduced if a dishonest node launch forged alerts against honest nodes. To strengthen our approach, we propose that the detection of threats is done collaboratively by the 1-hop neighbors of the targeted node, in addition of the targeted node himself.

In the case of the example, it will be to the nodes H, G and J and D to detect the forged TC message against node D. Actually, this is possible since each node knows its 1-hop neighbors, 2-hops neighbors (and those chosen as MPR).

The same approach is now applied for forged Hello messages. In this case, the detection is operated through the neighbors of forged nodes in Hello messages. In the example of the second threat scenario (section 3.2.2), when the intruder sends a Hello message announcing that he has symmetric links with A, B, C, D and X, the node B can detect the maneuver because he knows that C haven't a symmetric link with the intruder. Node B can then flood an alert message announcing a potential threat from the intruder.

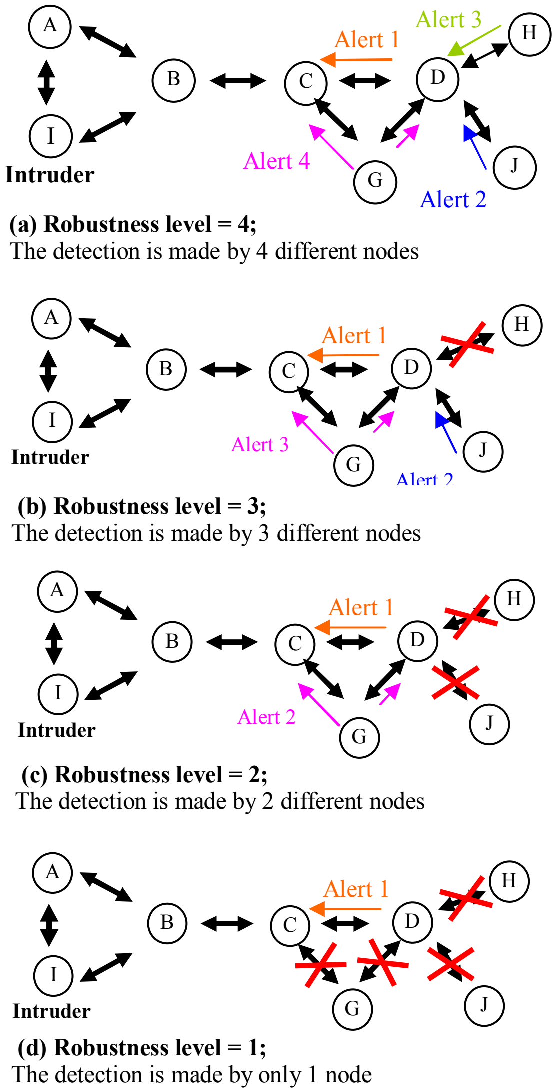
More generally, all nodes of the network participate in our IDS. Nevertheless, the detection is operated through 1-hop neighbors of the nodes which are forged in control messages, in addition of targeted nodes themselves. Consequently, the detection and the response of the intrusion are distributed and collaborative. Our IDS is implemented through the processing of control messages as the following:

Consider M(X) a control message (Hello or TC) emitted from node X.

```
For each node N,
  If Reception Control_Message M(X) do
    For i : 1 -> Size[M(X)] do
      If M(X)[i] ∈ Neighbors_List (N) Or M(X)[i] = N
        Then Verify M(X)[i] is 1-hop_neighbor of X,
          If True,
            Then
              Verify_Attributes (Sym/Asym links, etc)
                If Ok,
                  Update databases,
                If NOk,
                  Reject Control_Message,
                  Flood Alert,
          If Not True,
            Then
              Reject Control_Message,
              Flood Alert,
      Else
        Update Routing_Tables from M(X) unless
        Reception of an  Alert_about_X;
    End For;
  End if;
```

4.3. Robustness and discussion

In this work, we have considered that authentication and integrity are provided, selfish nodes detected, and thus messages are nor modified or dropped during transmission. The robustness of our IDS depends then on the number of nodes able to detect the intrusion, i.e. the number of 1-hop neighbors of the forged node in control messages and the forged node himself. Because dishonest nodes can emit forged alerts against innocent nodes, the higher the number of nodes detecting the intrusion is, the more the robustness is high. So, for the first threat we have presented (section 3.2.1), the robustness of the detection and of the response varies between these different topologies.



(a) Robustness level = 4;
The detection is made by 4 different nodes

(b) Robustness level = 3;
The detection is made by 3 different nodes

(c) Robustness level = 2;
The detection is made by 2 different nodes

(d) Robustness level = 1;
The detection is made by only 1 node

Figure 3. Different robustness levels according topologies

5. Conclusion and future works

In this paper, we propose an IDS solution to protect the OLSR routing protocol. This solution represents a first line of defense for the OLSR protocol since it mitigates threats exploiting flaws in the OLSR specifications to divert the normal routing operation. Our approach fits well to ad hoc networks characteristics and does not introduce constraints to the routing protocol operation. Currently, simulations are in progress to prove the effectiveness of our IDS by evaluating its response time and its false postivie and negative rates.

Nevertheless, enhancements have to be provided to our IDS specifications. For example, in case of discord between two nodes, cooperation of several nodes should be required to determine the threat source. Besides, considerations about mobility could be necessary to avoid false positives and negatives.

6. Acknowledgment

This research has been supported by France Telecom R&D Department.

7. References

[1] Clausen T., Jacquet P., Laouati A., Minet P., Muhltahler P., Qayyum A., and Viennot L., “Optimized Link State Routing Protocol”, IETF RFC 3626, 2003.
[2] Perkins C. E., and Royer E. M., “Ad hoc on-demand distance vector routing”, IETF RFC 3561, 2003.
[3] Johnson D. B., Maltz D. A., and Hu Y.C., “The Dynamic Source Routing Protocol for Mobile Ad Hoc Networks (DSR)”, IETF Draft 10, 2004.
[4] http://www.olsr.org
[5] Fourati.A., Badis H. and Al Agha K., “Security Vulnerabilities Analysis of the OLSR Routing Protocol”, *12th International Conference on Telecommunications*, *ICT 2005*, Cape Town, South Africa, May 3-6 2005.
[6] Baruch Awerbuch, Reza Curtmola, David Holmer, Cristina Nita-Rotaru, and Herbert Rubens, “Mitigating Byzantine Attacks in Ad Hoc Wireless Networks”, *Technical Report Version 1*, March 2004.
[7] Y. Zhang, W. Lee and Y. Huang, “Intrusion Detection Techniques for Mobile Wireless Networks”, *ACM/Kluwer Mobile Networks and Applications (MONET)*, 2003.
[8] A. Mishra, K. Nadkarni, A. Patcha, and V. Tech, “Intrusion detection in wireless ad hoc networks”, IEEE Wireless Communications, Vol. 11, Issue 1, pp. 48-60, February 2004.
[9] Y. Li, and J. Wei, “Guidelines on selecting intrusion detection methods in MANET”, The 21st Annual Conference for Information Systems Educators (ISECON), Rhode Island, USA, 4-7 November, 2004.
[10] I. Stamouli, P.G. Argyroudis, and H. Tewari, “Real-time intrusion detection for ad hoc networks”, The 6th IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks, Taormina, Italy, June 2005.
[11] B. Sun, K. Wu, and U.W. Pooch, “Zone-based intrusion detection for mobile ad hoc”, International Journal of Ad Hoc & Sensor Wireless Networks, September 2004.
[12] S. Marti, T. Giuli, K. Lai, and M. Baker, “Mitigating routing misbehavior in mobile ad hoc networks”, The Sixth ACM/IEEE International Conference on Mobile Computing and Networking (Mobicom), Boston, USA, 2000.