Integrating Intrusion Response Functionality into the MANET Specific Dynamic Intrusion Detection Hierarchy Architecture

Manpreet Kaur, Dale Lindskog(✉), and Pavol Zavarsky

Concordia University of Edmonton, Edmonton, Canada
mkaur2@student.concordia.ab.ca,
{dale.lindskog,pavol.zavarsky}@concordia.ab.ca

Abstract. In this paper, our interest is intrusion response in mobile ad hoc networks (MANET). All intrusion response systems (IRS) presuppose an underlying intrusion detection system (IDS). We propose improvements to an existing dynamic and hierarchical IDS architecture for MANETs, proposed by Sterne et al. Our improvements are designed to enhance its ability to form an underlying base IDS for an imagined IRS. The enhancements are chosen to overcome the lack of resiliency in the selected architecture, by adding backup cluster heads and a backup root node. Additionally, we also propose revisions designed to avoid giving the root node too much authority over intrusion response, by distributing that power among cluster heads. The root node acts, rather, as an attack information database. The cluster heads, we propose, would make use of a MANET specific intrusion response algorithm proposed and described by Kaur et al.

Keywords: Mobile ad hoc networks · Intrusion detection · Intrusion response
Clustering · OLSR

1 Introduction

A MANET is a wireless network which consists of some number of wireless mobile hosts that form a temporary network, a network without the presence of any centralized administration or infrastructure, such as access points, base stations, mobile switching centers, etc. Since there is this lack of centralized infrastructure, mobile nodes are required to cooperate with one another, in order to perform network related operations such as routing and packet forwarding. In a MANET, nodes have freedom to enter into, leave from, and move around within the network. Therefore, changes in network topology can occur at any time. Due to such characteristics, MANETs are more difficult to protect against intrusions [2, 3].

An IDS is used to detect, analyze and report intrusions, and has become an indispensable component of a defense-in-depth security approach for MANETs [4]. More recently, there has been interest in intrusion response in MANETs. But due to their peculiar characteristics, mentioned above, designing either an IDS or an IRS is considerably more complex for MANETs than for traditional networks. For MANETs, it is

Y. Zhou and T. Kunz (Eds.): AdHocNets 2017, LNICST 223, pp. 69–80, 2018.
https://doi.org/10.1007/978-3-319-74439-1_7

impossible to choose a specific set of nodes in the internetwork to perform the intrusion detection or intrusion response functions, since the ability of effectively perform such functions depends upon their placement in the internetwork, which, in a MANET, is subject to change, and potentially frequent change.

Much has been written about methods of detecting intrusions in MANETS, and differences between these methods depend to a great degree on the type of IDS architecture, which can be usefully categorized into three types. The simplest type is the Stand-alone IDS architecture, where each node of the network has a detection engine/IDS agent installed on it. Nodes are responsible themselves for detecting intrusions [5]. A more sophisticated architecture is the Distributed and Cooperative architecture, proposed by Zhang and Lee [6], which is designed so that neighboring IDS agents will cooperatively participate in MANET wide IDS. Both of these IDS architectures are suitable for a flat network infrastructure, but not for a multi-layered network divided into groups of clusters and organized into hierarchies [5, 7]. For a multi-layered infrastructure, a Hierarchical IDS architecture has been proposed, which is organized into levels, with a so-called root node present at the top of the hierarchy. The network is subdivided into groups called clusters. Each cluster has its 'cluster head' which acts as a control point and has more responsibility than other nodes for providing communication to other cluster heads and zones. In this architecture, local detection is carried out by a cluster, whereas global detection is carried out by cluster heads and inter-zone nodes [7].

It is often desirable to have mechanisms by which the network can respond against detected intrusions. Badie et al. [8] noted that, unlike a traditional fixed topology, in a MANET the node best suited to execute a response will often need to be determined at run time. Moreover, an attack detecting node might or might not be in a suitable position to execute the response instruction, in which case two problems arise: how is it determined which node is most suitable for executing a response, and how is a response instruction sent to that node? They argued that a Hierarchical IDS architecture would form the most suitable base for a MANET specific IRS, because of the presence of the root node at the top of the hierarchy. Being at the top of the hierarchy, and receiving aggregated information from below, the root node has, in theory, the most comprehensive picture of the attack and therefore is in the best position to make decisions about intrusion response. But to make informed choices about response, the root node needs a current and updating conception of the topology of the network, in order to know the current position of the attacker, the victim, and the identity of that node best placed to execute the response. They hinted that the MANET specific optimized link state routing protocol (OLSR) could be used to acquire the current network topology, and to quickly detect changes to it [8].

Kaur et al. [2] expanded on this these ideas, and developed an algorithm that could be implemented on the node responsible for determining intrusion response (the root node of the Hierarchical IDS architecture, in their example). This algorithm would rely upon the OLSR protocol for network topology related information. Their proposed algorithm works in a query-response mode, where the IRS function of the IDS root node queries the implemented algorithm to answer network topology related questions, which then returns a response, relying on the OLSR for its database of topology related

information. By querying the implemented algorithm, a node determines the positions of the attacker and victim in the overall MANET, and most importantly, determines other network topology related information that it needs to judge which node is best suited to execute the response.

We selected an existing IDS architecture, proposed by Sterne et al. [1]. Because of various features of their proposed IDS architecture, described in Sect. 2 of this paper, it forms a very suitable starting point on which to build the IRS related solutions proposed by Badie et al. and Kaur et al. However, we observed that Sterne et al.'s IDS architecture has also some limitations, discussed later. Therefore, we enhanced it with two revisions. First, we increase its resiliency, by introducing backup cluster heads and a backup root node, and additionally, we demote somewhat the authority of the root node, by proposing cluster heads as the executers of response instructions. Since the cluster heads will have the responsibility to respond to attacks, we imagine they will run the algorithm proposed by Kaur et al. [2] to facilitate the intrusion response functionality. The root node's primary responsibility will be to maintain a comprehensive IDS information database, allowing cluster heads to consult that database while making decisions about intrusion response.

Our concern in this paper is not to identify the structure and nature of the response instruction that a cluster head would send, nor methods to determine which nodes should respond, and how. Rather, our purpose is to describe an IDS architecture conducive to such a MANET specific intrusion response system, and one that will make use of Kaur et al.'s algorithm. The rest of this paper is organized as follows: Sect. 2 reviews types of MANET IDS architectures, OLSR, the related research on IRS in MANETs, and a brief explanation of the algorithm developed by Kaur et al. [2]. Section 3 is our justification for selecting a particular type of Hierarchical IDS architecture. The following section describes imperfections in the selected IDS architecture, proposes enhancements to that architecture to make it more resilient, and proposes revisions to address the danger of a single node, the root node, having too much authority over intrusion response. Finally, Sect. 5 concludes.

2 Review of Related Research

Mobile devices in MANETs, often simply referred to as nodes, are free to leave from, enter into, and move around within the network. Due to such characteristics, intrusion detection systems designed for traditional wired and even wireless networks cannot be applied to MANETs directly. We dedicate the first part of this section to an overview of various types of MANET specific IDS architectures. Existing IDS architectures for MANETs fall under three basic categories, and we discuss them below, along with their problems.

2.1 Types of MANET Specific IDS Architectures

Existing IDS architectures for MANETs fall under three basic categories; these are discussed below along with their problems [9]:

1. Stand-alone IDS Architecture
   In the Stand-alone IDS architecture, an IDS agent is installed on each node and runs independently to detect intrusions locally. This architecture is very limited, and amounts primarily to a mere host intrusion detection system.
   These types of IDS are inherently limited in their ability to detect attacks, because decisions about intrusion or attack must be based only on information available to individual nodes. That is to say, a stand-alone IDS architecture does not engage in cooperative detection. Chadli et al. [9] presented a study and analysis of the different proposed IDS architectures for MANETs, where they identified strengths and weaknesses of the different proposed IDS architectures. They describe various different stand-alone IDS architectures, including but not limited to: Battery-Based IDS, Threshold-based IDS, and Two-stage IDS. But according to their analysis, these architectures are prone to false-positives and negatives, and also introduce new security weaknesses [9].
2. Cooperative and Distributed IDS Architecture
   In this type, an IDS agent is again installed on each node of the MANET. However, the IDS agent's responsibilities involve not only collecting and analyzing evidence obtained locally, but furthermore, nodes share data with neighboring agents in an effort to detect attacks based on a wider range of information. Zhang et al. [6] proposed this type of architecture by considering the salient features of the MANET. IDS architectures should, they argued, be distributed and cooperative, because MANETs are distributed and network nodes cooperate with each other.
   But this architecture and various other Cooperative IDS architectures, such as the Friend assisted IDS architecture, the Cooperative IDS architecture based on social network analysis, etc., have limitations, studied and analyzed by Chadli et al. [9]. They argued that, for the entire set of architectures of this type that they studied, the rate of false positives and the detection accuracy is negatively affected by the high mobility of nodes. Moreover, the majority of them are vulnerable to various attacks, such as man in the middle, session hijacking, blackmailing, etc. Another weakness they found was that almost all cooperative IDS architectures impose extra processing and communication overhead [9].
3. Hierarchical IDS Architecture
   The Hierarchical IDS is an advanced version of the distributed and cooperative IDS architecture. This IDS architecture is organized into levels. This architecture divides the network into groups called clusters, where 'clustering' refers to the virtual partitioning of the network, and the arrangement of nodes into clusters, with each cluster having a 'cluster head', and with the other nodes of a cluster referred to as cluster members. There is a root node present at the top of the hierarchy that periodically gathers the aggregated intrusion related information from the lower level cluster heads. Cluster heads detect any malicious activity in their own cluster. Cluster heads report intrusions to higher level cluster heads, and this process continues until the information reaches the root node [1].
   If the IDS root node or any cluster head that performs IDS functionality becomes unavailable or is compromised, then the IDS may be compromised. For example, if an attacker takes control of a root node, then the detection system will not be able

to detect attacks. If any cluster head is down, then the root node will be unable to receive attack information from that particular cluster, and in such cases, the root node will be in a poorer position to detect or fully understand an intrusion, for its conclusions may be based on less complete information. Thus, this type of IDS architecture has one major drawback, i.e. a single point of failure, i.e. the root node, and also various 'single points of weakness', namely the cluster heads. This situation is even more serious if the IDS root node is also performing an IRS function. For if the root node is compromised, then an attacker may initiate or execute a fake intrusion response against the network. Therefore, it is important to have resiliency in this type of IDS architecture, and to mitigate the consequences of IRS compromise. Sterne et al. [1] proposed a specialized hierarchical IDS architecture for MANETs, which they termed the *cooperative intrusion detection* architecture. It is designed to facilitate accurate detection of MANET-specific attacks. The architecture is described as a dynamic hierarchy that can be structured into more than two levels, and organized into clusters. Each cluster has a cluster head that performs almost the same functions as discussed earlier in the hierarchical IDS architecture. Additionally, cluster heads also perform (1) data fusion/integration and data filtering, (2) computations of intrusion, and (3) security management. Every member node of a cluster monitors, logs, analyzes, responds, and alerts or reports to cluster heads [1]. The authors also explain the process by which intrusion detection information is propagated up the hierarchy. To maintain the hierarchical structure of the IDS, a clustering algorithm is run, and the process of clustering continues until all nodes in the network are part of the hierarchy.

Though Sterne et al.'s proposed IDS architecture is suitable for detecting a wide range of MANET-specific and conventional attacks, the question we are interested in is whether it is suitable as an IDS foundation for a MANET IRS. Our answer to this question is provided later in this paper, but before that, it is important to understand how intrusion response works in MANETs, and we dedicate the next subsection to this topic.

2.2 Intrusion Response System in MANETs

There are an indefinite number of different possible response actions, such as node isolation, session disruption, tarpitting, packet filtering, disabling portions of the network, blocking ports, etc. Regardless of the specific response, it is important to note that, because MANETs have no fixed network topology, the node (e.g. the root node of the Hierarchical IDS architecture) that detects the intrusion might not be in the best, or even in an appropriate position to execute the response. Therefore, it becomes necessary for the root node to send a response instruction to a more suitable node, and for that more suitable node to instead execute the response. In MANETs, according to Badie et al., "An effective IRS often depends on an accurate conception of network topology, and it requires some method for the IRS to discover which particular node is in the best position to execute the response" [8]. For MANETs, there are complexities involved in making that determination, because a MANET does not have a fixed network topology.

What is clearly required is a current conception of the network topology at the time the response instruction is to be sent.

Badie et al. further explained that an efficient manner in which to determine the current network topology is to rely on the topology database used in link state routing protocols, and they remarked in passing that the MANET specific optimized link state routing protocol (OLSR) is a link state protocol, and therefore appears to be a good choice. They imagined that "The root of the hierarchy, in the Hierarchical IDS architecture, receives aggregated information from the cluster heads. Then, the root node, having a complete picture of attack and the topology of the network, can choose the most appropriate node to execute the response" [8].

Kaur et al. expanded on this, and proposed an algorithm that relies on the OLSR database to compute answers to relevant network topology related questions.

The proposed algorithm operates in a query-response mode, and would be employed by the root node of the IDS/IRS. The root node sends a query as input to the implemented algorithm, and it outputs the response, formatted as an unordered list of nodes satisfying the query. In the hierarchical IDS architecture, the root is placed at the top of the hierarchy and receives consolidated intrusion detection information. After examining the received information and detecting the attack, the root node may decide to perform intrusion response functions based on that consolidated information. At that time, the IDS root node becomes an IRS root node. When the IRS root node decides to send one or more response instructions, it must have current information about the topology of the network. To determine the relative position, in the network, of the attacker, the victim, and of the most suitable node(s) to respond, the IRS root node asks certain kind of questions of the implemented algorithm. According to Kaur et al., these are the three general questions that IRS root node may ask:

- Which nodes are currently in the network?
- Which nodes are n hops away from X?
- Which nodes are in-line between X and Y?

(where 'n' ranges over positive integers, and 'X' and 'Y' range over nodes in the MANET).

After receiving the input from the IRS root node, the implemented algorithm returns output which helps the IRS root node to determine the most suitable node to respond to the attack or intrusion. To generate results and give output, the algorithm relies on the OLSR protocol. The next subsection briefly explains how the OLSR protocol operates in MANET.

2.3 Optimized Link State Routing (OLSR) Protocol

OLSR, defined in RFC 3626 [10], is a proactive routing protocol developed specifically for MANETs, and optimized for networks with rapidly changing topologies. OLSR uses two types of messages, 'HELLO' and 'TC' (Topology Control), to maintain current topological information about the network at every node. HELLO messages are used to discover the immediate neighbors (one-hop neighbors), and then two hop neighbors are discovered from the answers given by one-hop neighbors. The information about a one

hop neighbor's link status, and information about its two hop neighbors, is contained in a neighbor table that is maintained by each node. Topology control messages are broadcast by MPRs (Multipoint Relays) to advertise links or topological information throughout the network. Based on this topological information, every node calculates its topology table. This topology table contains information about the topology of the network obtained from the TC message, and information about the MPRs of the other nodes. In this way, OLSR maintains the topology database for the network at each node by periodically exchanging HELLO and TC messages with neighboring nodes. Topology and neighbor tables are maintained by OLSR at every node, and it is this which makes them able to calculate the best route to a destination node.

Kaur et al.'s [2] algorithm uses the OLSR protocol to determine facts about the topology of the network, such as the total number of nodes and positions of specific nodes in the network, e.g., the attacker and victim.

3 Selecting a Suitable IDS Architecture

One of the main concerns in choosing an IDS that will smoothly interoperate with an IRS is whether the IDS facilitates an IRS's need to determine which action to perform, and against which node and from which node that response action will be performed. The details surrounding this question are not in the scope of this paper, for to determine which particular response related actions need to be taken depends upon details such as the type of the attack detected by an underlying IDS, and the placement of the attacking and attacked nodes at the time of the attack.

However, these concerns do indicate, in a general way, a need to have mechanisms in place for effective decision making about the response instruction to be given, and about which node should execute that response.

We believe that the answer to this general question is that such decision making should be done by making use of the most attack- and network activity- related information available to the IDS, e.g., by selecting a node that has more consolidated/aggregated intrusion detection information as compared to the other nodes of the network, and for the following reasons. First, the more information that is known about the attack, the better the intrusion response will be. That is to say, the less complete the information about the attack, the greater the possibility of incorrect decisions, not only decreasing the effectiveness of the IRS, but significantly, increasing the likelihood of harmful responses, e.g. responses against innocent nodes or the network as a whole. This is especially true of attacks designed to cause the IRS to respond against innocent nodes, or that affect the functioning of routing protocols. Second, the rate of false positives would be significantly lower, that is, the IDS would be less likely to raise alarms in response to normal network behavior. This again is extremely important for any IDS underlying an IRS, since false positives are not just time wasting, as in the case of an IDS, but can in an IRS be very harmful, since, as noted, they may generate responses against innocent nodes.

Of course, these concerns are valid for IRSs generally speaking, but they are especially important in the case of a MANET specific IRS, because in MANETs it is

particularly difficult to avoid misleading attack or network related information, due to the fact that a MANET does not have a fixed network topology. This can be made more clear with the following example. Consider the topology depicted in the Fig. 1, and consider also the following, very simple attack scenario.

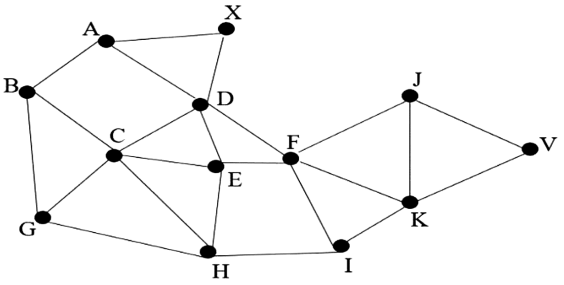

Fig. 1. MANET topology

Suppose node A attacks node V, but in doing so spoofs its IP address to make it appear that the attack has originated from node B. Suppose further that the victim node, V, detects this attack, but (incorrectly, and understandably) judges that it is coming from node B. With a stand-alone or distributed cooperative IDS architecture, this error in detection is a serious risk, and if such an architecture formed the base of an IRS, it is quite easy to imagine that a response instruction is sent to the neighboring nodes of B, e.g. nodes A, C and G. And that response instruction, depending on its details, could in effect perform a denial of service or other attack on the innocent node.

A. For example, such would be the effect if the response instruction was either to perform packet filtering for all the packets sourced from node B, or, e.g., to stop forwarding any traffic coming from and going to node B. Thus, node A would have exploited the IRS system itself, and node A might even continue the attack, as no intrusion or attack response was initiated against it. In this case, the IRS is not only ineffective but harmful.

Let us now discuss the same scenario by imagining the hierarchical IDS architecture underlying an IRS. In this case, there will be a root node at the top of the IDS hierarchy, and the network will be organized into levels in some way. Suppose, in Fig. 1 above, that node X is the root node and that the other nodes are organized into hierarchies. With this type of IDS architecture, the dissemination of intrusion or attack information is done in such a way that nodes at the lower level of the hierarchy send intrusion or attack information to the nodes stationed at their immediate upper level of the hierarchy, and this dissemination of information continues until the information, or an aggregate of it, reaches the root node (node X in our example).

In our example, referring to Fig. 1, it is the root node X that is receiving consolidated/aggregated intrusion and attack information. It is very easy to imagine an IDS so configured that information about spoofed IP packets is contained in the aggregate, detected, e.g., because of mismatches or suspicious pairings between MAC and IP addresses were noticed by adjacent nodes (e.g. nodes adjacent to the node A!). The root node would,

therefore, be in a much better position to judge that the attack, apparently sourced from B, is in fact sourced from node A. The root node would, of course, therefore be in a much better position to cause an intrusion response instruction that actually targets the guilty node, e.g. by determining all possible routes from node A, and sending a response instruction to all neighboring nodes of A, to stop forwarding A's traffic. In this case, the IRS is clearly more effective, and less dangerous, when compared with the previous case. In conclusion, the hierarchical IDS architecture is, other things being equal, a superior form of MANET specific IDS architecture on which to base an IRS. There are, however, many different sub-types of Hierarchical IDS architecture described in the literature. The best sub-type for our purposes is, as discussed earlier, that proposed by Sterne et al. They proposed a dynamic hierarchy architecture that acts as a foundation for all intrusion activities in mobile ad hoc networks.

We selected Sterne et al.'s IDS architecture for various reasons: It is a general IDS architecture, and thus not restricted to MANET-specific attacks. It uses a dynamic hierarchy designed to accommodate nodes and links which might appear and disappear rapidly, even under normal network activity. It explicitly identifies particular nodes as having consolidated/aggregated intrusion detection related information, and intrusion detection and correlation occurs at the lowest level in the hierarchy at which the aggregated data is sufficient to enable an accurate detection or correlation decision. Finally, the responsibilities of nodes and cluster heads are clearly defined, and the operation of the architecture is also depicted with scenarios that carefully illustrate its intended operation and features.

4 Enhancing the Proposed Enhanced Dynamic Intrusion Detection Hierarchy Architecture

Though Sterne et al.'s architecture has commendable features, there are serious problems of resilience. There is a single point of failure at the root node. The root node, so important because it stores the greatest amount of consolidated/aggregated intrusion detection information, can itself be under attack, or can otherwise fail. There is a similar problem with the cluster heads. The cluster head of each cluster is responsible for monitoring traffic and detecting intrusions local to its cluster. If a cluster head is compromised or otherwise fails, and if there is no node at a higher level that is in a position to perform IDS functionality in that specific circumstance, we again have a single point of failure, this time at the level of the cluster as opposed to the hierarchy as a whole. Thirdly, if there is a denial of service attack or network congestion, then the root node may be unable to receive intrusion detection data from nodes at lower levels of the hierarchy. For example, suppose a cluster head is under a denial of service attack and cannot forward intrusion detection information, or at least is delayed in forwarding such information to the cluster head one level above in the hierarchy. Then the root node will be unable to get information from one cluster, i.e., from one portion of the network. The root node will therefore be working with incomplete information. Finally, the architecture depends upon the reliability of upper level cluster heads to forward upward their aggregated information toward the root node, and if any fail to do so, this too constitutes

a group of single points of failure in the system, a group that grows commensurate with the size of the network.

To enhance resiliency in the dynamic intrusion detection hierarchy architecture, we propose two root nodes, and two cluster heads in each cluster, as suggested by Ishaq [11]. One of each will act as a primary, and the other as a backup. Consider the illustration shown in Fig. 2. The network is divided into five clusters. Out of five, three clusters (C1.A, C2.B, and C1.C) are at the first level of the hierarchy, and the remaining two clusters, namely C2.A and C3.A respectively, are the second and third level representatives of the hierarchical architecture. The first, second and third level cluster heads are annotated by '1', '2', and '3' respectively. The root node is at the top of the hierarchy. Each cluster has a primary (denoted by 1, 2 and 3) and a backup (green) cluster head. The arrows originating from the member nodes and backup cluster heads, and pointing to the first level cluster heads, represent the propagation of intrusion detection information. Further, the arrows pointing from the first level cluster heads, to the second level, depict the propagation of aggregated intrusion-related information. This process of consolidating intrusion detection data from the lower- to upper-level hierarchies continues until it reaches the root node, as suggested by Sterne et al. [1].

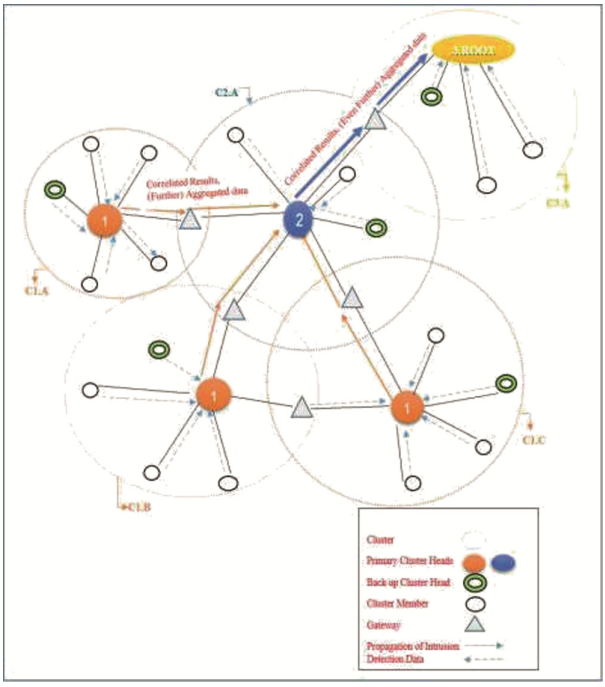

Fig. 2. Enhanced dynamic intrusion detection hierarchy architecture for MANETs (Color figure online)

To organize the architecture hierarchically, we prefer to use a clustering scheme that applies the Weight-Based Hierarchical clustering algorithm. Details of this clustering algorithm can be found in [12] but the general approach is that all nodes use some set of factors (such as mobility, transmission, battery power, bandwidth, etc.) to compute weight. Then, the selection of the root and the cluster heads is made fairly from the calculated weight. Two cluster heads would then be elected from each cluster. The criterion to choose the backup cluster head is large transmission range (LTR). A primary cluster head would select as its backup the node within its cluster that is in its best transmission range. Then, if a primary cluster head fails in a detectable way, the backup cluster head will take over its responsibilities, and thus the cluster head would not constitute a single point of failure.

Recall that the root node is at the top of the IDS hierarchy and receives aggregated/consolidated intrusion detection information from the entire network, and therefore in theory has the most complete information. It may seem natural, for this reason, to assign to it the function of determining and communicating response instructions. But assigning both IDS and IRS functionalities to a single entity is a major risk, as that node might misuse its power. For this reason; we propose that the root node does not perform IRS functions, but rather, that authority over IRS functions is distributed to the cluster heads collectively. This not to say that there would be a change in the mechanism of disseminating intrusion detection information from the lower level hierarchies to the root node. The root node will continue to receive consolidated/aggregated information but will not detect or respond to attacks. Rather, the root node will act as an attack information database, consulted by cluster heads in the process of detecting intrusions.

5 Conclusion

In MANETs as elsewhere, an IRS presupposes an underlying IDS. In MANETs, intrusion detection is complicated by the fact that the network topology is dynamic, and for this and other reasons, various MANET specific IDS architectures have been proposed in the literature. Intrusion response is further complicated by the fact that the best node to execute a response must often be determined at run time, again because MANET topologies are dynamic. We chose Sterne et al.'s DSt05 IDS architecture to form a base IDS architecture for the purpose of intrusion response in MANETs. However, their proposed IDS architecture lacks resiliency, and therefore this paper proposes an enhanced dynamic intrusion detection hierarchy architecture that we argue forms a better basis for an IRS. The proposed enhancements include removal of single points of failure, and distribution of power over intrusion response instruction execution.

References

1. Sterne, D., Balasubramanyam, P., Carman, D.: A general cooperative intrusion detection architecture for MANETs. In: Proceedings of the Third IEEE International Workshop on Information Assurance (2005)

2. Kaur, J., Lindskog, D., Zavarsky, P.: An algorithm to facilitate intrusion response in mobile ad hoc networks. In: Proceedings of the 9th International Conference on Security of Information and Networks (2016)
3. Gupta, P.: A literature survey of MANET. Int. Res. J. Eng. Technol. (IRJET) **03**(02), 95–99 (2016)
4. Hicham, Z., Ahmed, T., Rachid, L., Noureddin, I.: Evaluating and comparison of intrusion in mobile ad hoc network. Int. J. Distrib. Parallel Syst. (IJDPS) **3**(2), 243–259 (2012)
5. Anantvalee, T.: A survey on intrusion detection in mobile ad hoc network. In: Wireless/ Mobile Network Security, pp. 170–196 (2006)
6. Zhang, Y., Lee, W., Huang, Y.-A.: Intrusion detection techniques for mobile wireless networks. Wirel. Netw. **9**(5), 545–556 (2003)
7. Alattar, M.: Security supervision of mobile ad hoc network: a lightweight, robust and reliable intrusion detection system, Université de Franche-Comté (2013)
8. Badie, A.M., Lindskog, D., Zavarsky, P.: Responding to intrusions in mobile ad hoc networks. In: World Congress on Internet Security (WorldCIS-2013), pp. 30–34 (2013)
9. Chadli, S., Emharraf, M., Saber, M., Ziyyat, A.: Combination of hierarchical and cooperative models of an IDS for MANETs. In: Tenth International Conference on Signal-Image Technology & Internet-Based Systems (2014)
10. Clausen, T., Jacquet, P.: Optimized link state routing protocol (OLSR). October 2003. http://www.ietf.org/rfc/rfc3626.txt. Accessed 26 June 2016
11. Ishaq, Z.: Secure MANET using two head cluster in hierarchical cooperative IDS. Int. J. Comput. Appl. (0975–8887), **57**(3), 10–13 (2012)
12. Sahana, S., Saha, S., Das Gupta, S.: Weight based hierarchical clustering algorithm for mobile ad hoc networks. Procedia Eng. **38**, 1084–1093 (2012)