http://ansinet.com/itj

ITJ

ISSN 1812-5638

INFORMATION TECHNOLOGY JOURNAL

ANSInet
Asian Network for Scientific Information
308 Lasani Town, Sargodha Road, Faisalabad - Pakistan

Information Technology Journal 13 (13): 2065-2078, 2014
ISSN 1812-5638 / DOI: 10.3923/itj.2014.2065.2078

Emerging Trends in Intrusion Detection in Mobile *ad hoc* Networks (MANETs)

[1]Mohammad Alnabhan, [1]Ahmad Haboush, [2]Binod Kumar Pattanayak,
[1]Mohammad Alnawayseh and [1]Anas Bassam Al-Badareen
[1]Department of Computer Science, Jerash University, Jerash, Jordan
[2]Department of Computer Science and Engineering, Institute of Technical Education and Research,
Siksha 'O' Anusandhan University, Bhubaneswar, Odisha, India

Abstract: Looking at the broad spectrum of Mobile *ad hoc* Networks (MANETs) applications, security challenges imposed during its deployment have triggered researchers to analyze the vulnerabilities of such networks and deduce newer reliable solutions. In this study, an attempt is made to address the challenges imposed by security impairments in Intrusion Detection Systems (IDS) in the context of MANETs. This study covers a broad analysis of IDS architectures and algorithms and measures advantages and disadvantages associated with these architectures. This study contributes significantly to IDS successful implementation and triggers researchers in this field, to develop more robust solutions encountering the security drawbacks and challenges currently experienced.

Key words: MANET, IDS, IDS architecture, IDS algorithms

INTRODUCTION

IEEE 802.11 based networks have found wide implementation over the years, in deployment of localized wireless data communication environments in the form of *ad hoc* networks. In a property devised network, the wireless hosts are well-equipped and well-positioned to support a fairly distributed connectivity among the nodes of the network. In a random way point mobility model, nodes in the network may possess a completely unpredictable mobility pattern, however within the coverage area of network. QoS aware routing across an *ad hoc* network can be successfully achieved with confirmation to a set of routing QoS parameters such as bandwidth management, delay management, congestion control, security support (Conti, 2003).

This research work mostly concentrates around the major issues related to security support for *ad hoc* networks related to intrusion detection. Here, a broad analysis of intrusions evolving around mobile *ad hoc* networks is carried out along with a variety of measures to overcome them. Several IDS architectures are addressed in the following sections along with a set of intrusion detection algorithms.

SECURITY ISSUES IN *ad hoc* NETWORKS

Ubiquitous computing has become extensively popular among the researchers of late. It requires wireless channels, where participating nodes are capable of accessing the information in the network as per requirement using these channels (Weiser, 1999). Mobile *ad hoc* Network (MANET) represents a category of such networks, where mobile nodes communicate among themselves in a self-organizing and dynamic pattern. Real world communication can be efficiently implemented using such networks even in places without any existing infrastructure (Corson *et al*., 1999). Nodes can communicate directly with neighboring nodes that lie within its communication range (single hop) or indirectly via other nodes (multi hop) if the destination node lies beyond its communication range. MANET can be characterized by the following properties (Mishra and Nadkarni, 2003).

Lack of reliability of wireless channels: Communication across the wireless channels becomes unreliable due to node mobility and limited power at the nodes.

Dynamic topology: Due to node mobility, the network topology changes frequently in time leading to frequent updates of routing information in routing tables at nodes.

Absence of security features in routing protocols: Dynamics of MANET and lack of security mechanism in *ad hoc* routing protocols makes the network prone to external attacks, which needs to be resolved for productivity of such networks.

Security vulnerabilities in *ad hoc* networks: MANETs inherently differ from conventional networks in their properties. A centralized control over security features

Corresponding Author: Mohammad Alnabhan, Department of Computer Science, Jerash University, Jerash, Jordan

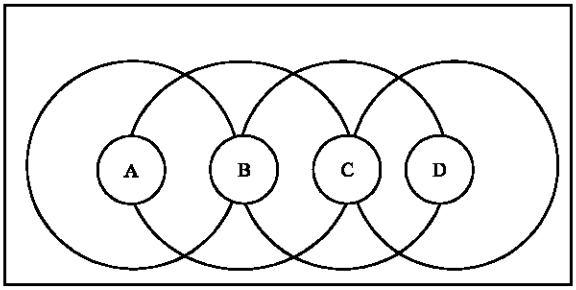

Fig. 1: Nodes in the network and their transmission ranges

cannot be achieved in them due to absence fixed infrastructure. At the same time, third party controlled mechanisms may not be useful too. Dynamic topology of MANET needs scalable security measures. Communications in a wireless environment are limited to fixed channel bandwidth. Real time applications are bounded by a strict time constraint that needs to be addressed at the level of the routing protocol. Another aspect is jitter that presents huge challenges for efficient implementation of multimedia applications in a MANET. In addition, limited power of nodes presents additional challenges for networks with heterogeneous nodes, where resource intensive nature adversely affects security measures especially for *ad hoc* environments. Current research is heading towards incorporation and maintenance of *ad hoc* infrastructure implementation using *ad hoc* routing protocols.

Intrusion detection in *ad hoc* networks: MANETs, unlike wired networks, need more security mechanisms. Attackers may intrude into the network through the subverted nodes as demonstrated in Fig. 1.

Intrusion can be defined as a set of activities that may bring harm to the confidentiality, integrity, or availability of a resource and an Intrusion Detection System (IDS) is a collection of measures in order for identification of such activities (Phatak and Mishra, 2012). It monitors the system activities on a regular basis, identifies the deviations with respect to incorporated security agreements and responds to them. IDS become an indispensable part of a security system, as intrusion prevention strategies can never be enough for the reason that every time new intrusions may arise. Possible deviations in security policy incorporated across a system can be detected by continually monitoring the activities on the system and subsequently responding to those. If an attack is detected in the network, a response can be generated in order for prevention or minimization of the damage incurred by the system (Sen, 2010).

The specific features of MANETs present a challenge to security solutions. The solutions to intrusion detection in wired networks, proposed in the literature, do not adhere to simple application to MANETs. The traditional approach to detect attacks at the traffic concentration points can be no longer suitable for this distributed environment. In addition, a large spectrum of solutions for conventional networks appears to be ineffective and inefficient for this resource constrained environment. A set of issues should be addressed while designing IDS for MANETs. The vital issues that make applying existing solutions impractical are: Dynamic nature of MANETs, the absence of fixed infrastructure and resource constrained nodes. However, adaptation of existing solutions to MANETs, is the challenge to modern researchers.

Three main intrusion detection approaches are proposed in the literature: Anomaly-based, misuse-based and specification based (Kheyri and Karami, 2012). All intrusion techniques have their own strengths and weaknesses. Specification-based intrusion detection represents one of the most widely implemented by researchers. In which, intrusion detection techniques detects intrusion as a result of deviations during runtime from the specifications defined at the level of routing protocols. This technique is incorporated in a variety of routing protocols in MANETs. However, denial of service (DoS) attacks cannot be detected by such protocols. Anomaly-based techniques detect intrusions as deviations from the normal behavioural patterns of a system. Defining normal behavioural patterns imposes the principal challenge to this technique. Normal behaviour can change over time and IDS systems need to adapt accordingly, otherwise, it may result in the system exhibiting a high false positive rate. On the contrary, it can successfully detect unknown attacks. In MANETS, it is vital as new attacks and new vulnerabilities can be perceived throughout the lifetime of the network. A comparison is made between known attack signatures and current system activities in Misuse-based IDS. It is incapable of detecting new attacks. However, this technique is commonly preferred by commercial IDSs in the literature due to its efficiency and its low false positive rate and very little attention is given to research on signatures of new attacks against MANETs.

Issues related to intrusion detection in MANETs: A large variety of IDSs have been proposed for wired networks by authors (Imella and McMillan, 2001). However, their implementation in MANETs makes it ineffective and inefficient for this new environment due to its specific features. Thus, it motivates researchers to development of

newer solutions for intrusion detection in MANETs or adaptation of the existing IDSs for wired networks to MANETs. There are a set of issues that should be addressed while designing a new IDS for MANETs. This includes the following.

Lack of central points: MANETs unlike wired networks do not support any devices like gateways, routers, which are implemented in wired networks so as to monitor the entire network traffic entering into it. Only a portion of the network is visible to a node, i.e., only the packets it sends or receives along with other packets moving within its transmission range. Due to distributed and cooperative nature of MANET, the intrusion detection and response systems incorporated in it are required to be distributed and cooperative in nature, which consequently imposes some difficulties (Zhang and Lee, 2000). Distribution and cooperativeness is difficult to achieve in IDS due to the resource constraints like limited bandwidth, fixed processor speed and limited power at the nodes. In addition, for such a dynamic environment, storing the record of possible attacks in a centralized database and their distribution across various IDS agents becomes a tedious task for misuse-based intrusion detection systems.

Mobility: Nodes in a MANET are free to enter or leave the network and move independently which leads to a frequent change in the network topology and thus, highly dynamic nature of MANET makes the traditional IDS unreliable. For instance, anomaly-based methods are unable to distinguish if a node that generates obsolete information, has been compromised or that the node is yet to receive duly updated information (Huang *et al*., 2003). Another factor of mobility effect on IDS, is that IDS architecture may change with changes in the network topology.

Wireless links: Wireless links in a MANET are constrained with limited bandwidth and link failures occur very often. In the process of communication, IDS agents must keep track of the state of the links in order to exchange data as well as alerts. IDS agents must limit their data transfer rates to a minimum so as to avoid any congestion across the network. Otherwise, limitations in channel bandwidth may make the IDS operation ineffective (Vishwakarma and Chopra, 2012). An IDS may not operate effectively due to bandwidth constraints. For instance, a real-time response to an attack may not be viable on the part of an ID because of communication delay. Further, failure of wireless links may lead the IDSs to become disconnected. An IDS needs to tolerate lost messages while maintaining required detection accuracy (Tseng *et al*., 2003).

Limited resources: Mobile nodes operate on battery power and different nodes may have different capacities depending on the types of mobile devices used such as laptops, smart phones and so on with varying storage and computing capacities. Effectiveness and efficiency of the IDS agents are strongly affected by the variety of nodes, generally with scarce resources. For instance, tendency of nodes to drop packets in order to save battery power makes it difficult to distinguish between attackers and selfish nodes and at the same time, huge number of alerts coming from neighbouring nodes cannot be effectively processed by an IDS agent due to limited memory space at the node that the intrusion detection algorithm must consider. For example, memory constraints related to signatures should be taken into account in case of misuse based algorithm whereas usage of resources should be reduced for optimization of anomaly-based detection algorithm.

Absence of secure communication and line of defense: MANETs do not possess a clear line of defense. In such environment, IDS traffic across a MANET needs to be encrypted in order to prevent attackers from getting acquainted with its functionalities (Vishwakarma and Chopra, 2012). However, since ample amount of resources are required for encryption and authentication, such operations are difficult to implement in a mobile wireless environment. Significant amount of risk is involved with most IDS agents in a mobile and wireless environment that may lead to undesired consequences. In such cases, they may tend to generate false alerts those make the IDS totally ineffective.

Cooperativeness: As the routing protocols in MANETs are cooperative, they can easily be targeted by new attackers. For example, a malicious node may masquerade to be a neighbour with the other nodes and take active part in decision making, consequently affecting significant portions of the network.

Intrusion detection architectures: An ample amount of work is reported in the literature by variety of authors as elaborated below.

Stand alone IDS architectures: The stand-alone IDS architectures aim at detecting malicious activities in a MANET using a self-contained approach (Xenakis *et al*., 2011). Authors briefly present and evaluate the most

recent stand-alone IDS architectures for MANET (i.e., battery-based, threshold-based and two-stage IDS architecture) taking into consideration the strengths and weaknesses of each one. Jacoby and Davis (2007) have devised a stand-alone architecture for detecting malicious activities in MANETs, by monitoring power consumption in every node's battery on a regular basis.

Here, an intrusion is detected by comparing a node's power consumption using smart battery technology with that induced by known attacks. As the simulation results depict, the proposed IDS could detect 99% of the attacks in cases that only one type of them occurred. It also detected multiple attacks, but only when the nodes were idle and did not possess any activities. This architecture is more reliable (i.e., since it supports hardware operation) as compared to other IDSs that rely on audit data and anomaly-based detection and these IDSs can be easily manipulated by intruders. However, it is capable of detecting only attacks that lead to power consumption irregularities and only when the nodes are idle, which is a rare event in real systems.

A stand-alone IDS proposed by Nadkarni and Mishra (2004) uses compound detection in order for reducing the amount of false positive alerts, which specifically occurs during anomaly detection. It implements adjustment of thresholds in order to determine malicious behaviors. In the process of initialization, the intrusion detection system at each node creates the normal profile of the network traffic. It computes threshold values based on this profile beyond which possible attacks can be identified. Once a known attack is detected a counter called mis-incident is created, the node responsible for it, is marked as suspicious and incremented with every time this attack being detected. If the attack is repeated and the mis-incident counter exceeds the threshold value for this specific attack, the suspicious node is labeled as malicious. If no malicious behaviors are detected after a predefined period of time, the threshold is increased, decreased otherwise. The principal strength of this architecture is adaptability to network changes, since it uses variable thresholds. For example, periodic symptoms of suspicious behaviors resulting from network topology changes will remain under the detection thresholds, whereas malicious behaviors will subsequently exceed the thresholds indicating the occurrence of attacks. On the contrary, the weakness of this approach of adjustment of thresholds imposes new security challenges as malicious nodes can easily exploit this mechanism. To be specific, a malicious node may tend to increase the threshold values thereby performing legitimately for a certain period of time. Consequently, if the threshold values become high enough, it may generate an attack considering not exceeding the threshold values and subsequently raising alarms. On the other hand, nodes that might not take part in the routing process or generate invalid routing updates due to outdated routing information (i.e., caused due to high mobility) might be falsely labeled as malicious ones. Furthermore, coordinated attacks like byzantine attacks cannot be detected, since nodes do not cooperate.

A two-stage, stand-alone IDS architecture proposed by Lauf *et al.* (2010) can be used in resource-constrained environments like MANETs. It incorporates two different detection engines in every node, where the first one, i.e. the maxima detection system (MDS), is used to rapidly identify a potential threat and activate the second engine, i.e. the cross-correlative detection system (CCDS). MDS is meant for identifying unusual symptoms in the observed interactions at the application layer. These unusual symptoms can be identified by maintaining the history of interactions at the application layer and comparing them with a normal profile created offline. In case of a possible attack, MDS activates CCDS that calibrates a threshold value as per the known attack, calculates average values of the application behavior of each node and compares them with the threshold value. Behaviors exceeding the threshold value are marked as malicious. With the deployment of two detection engines at each node, the presented IDS increases detection accuracy as compared to other single engine IDSs since one engine supplements the other. However, CCDS is prone to false positives and negatives, since it calibrates the threshold value only once during startup. Thus, dynamic changes of the network, caused due to nodes mobility, are not supported by CCDS.

Cooperative IDS architectures: An intrusion detection engine is installed in each node in the cooperative IDS architectures, with a provision of local audit data monitoring and intrusion detection. In order to resolve inconclusive intrusion detections and detect advanced types of attacks more accurately, detection engines may cooperate with engines of neighboring nodes through the exchange of audit data or detection outcomes.

A cooperative IDS architecture based on social network analysis: A cooperative IDS architecture proposed by Wang *et al.* (2009) relies on a detection engine that utilizes social network analysis methods. Here, each node incorporates an intrusion detection engine that performs detections using audit data received from its "ego" network, consisting of a hosting node ("ego") and the nodes ("alters") that are directly connected to it. The deployed engines operate in the same manner as anomaly detection but they utilize social relations as metrics of

interest that require less computational overhead as compared to standard anomaly detection engines. In this case, a training phase is required to create normal profiles like in anomaly detection and the detection engines monitor the Medium Access Control (MAC) as well as network layers. The proposed IDS comprises of three modules: (1) The data pre-processing module that for collecting and pre-processing audit data, (2) The social analysis module for performing intrusion detection and (3) The response module for integrating local and global intrusion alerts, gathered from neighboring nodes. In the process of IDS operation, the data pre-processing module collects audit data from its neighboring nodes with an interval of five seconds each. Following, the social analysis module processes the collected data in order for realizing social relations among the "ego" network nodes that represent the behavior of these nodes at a specific point of time. Then, a comparison is made between realized relations and the normal profile of expected behaviors and if any variation is concluded from this comparison, then an intrusion can be inferred. After an intrusion is detected, the response module sends notifications to the neighboring nodes regarding the inferred intrusion. Less computation complexity incurred by the employed detection engines as compared to conventional anomaly detection engines, which represents the main strength of this architecture.

The weaknesses of this architecture as presented in Wang *et al.* (2009) are detailed below:

- The detection accuracy may drop in cases of high mobility of nodes as a consequence of increased rate of false positives. A node would only have a limited amount of time to create social relations with neighboring nodes before it changes its location in high mobility. Consequently, enough information would not be available for social analysis module to distinguish between normal and malicious behaviors
- The communication load among nodes may be increased due to audit data exchange leading to degradation in the performance of the network. The authors have arbitrarily selected a 5 sec interval for audit data exchange within each "ego" network, without any evaluation of the impact of this parameter on the network performance
- The exchange of audit data may lead to new security challenges, since a malicious node may tend to either transmit false audit data or avoid transmitting any of them, in order for hindering or misleading the detection process

A multilayer cooperative detection architecture: A cooperative IDS architecture proposed by Bose *et al.* (2007) uses three parallel anomaly detection engines, referred to as MAC layer detection engine, routing detection engine and application layer detection engine, installed in each node. Multi-layer detection initiates increasing detection accuracy for the reason that attacks targeting upper-layer protocols, can be seen as legitimate events at lower-layers and vice versa. The MAC layer detection engine monitors both access control as well as addressing in the data link layer. The routing detection engine is meant for monitoring the network layer and keeping track of the packet delivery as well as routing information. The application layer engine is used for monitoring the application layer. The task of each engine is to collect the appropriate audit data, process them and look for malicious behaviors within them. In each node, a local integration module is used to combine the results from the three different detection engines and a global integration module combines the results received from the neighboring nodes. The effectiveness of the proposed architecture is evaluated by a set of simulations performed using the GloMoSim (Zeng *et al.*, 1998).

The multi-layer IDS possess the following strengths:

- As the multiple detection engines supplement each other, this architecture provides more accurate detection as compared to other single engine detection solutions. As inferred from the simulation results, integration of the results of all three engines can increase the detection accuracy up to 20% as compared to the results that each detection engine could yield (Bose *et al.*, 2007)
- Since only the detection results are exchanged but not the audit data, it induces comparatively low communication overhead even though it operates on the cooperation among the neighboring nodes

The multi-layer IDS architecture has the following weaknesses:

- The processing overhead in each node is increased in this architecture as compared to other single engine solutions, because the IDS implements three detection engines instead of one. However, the processing overhead is not estimated and reported by the authors
- High packet loss and/or high nodes' mobility affect the ratio of false positives and the detection accuracy negatively. This results as a consequence of the routing detection engine relying on packet delivery and routing information in order to detect attacks.

The inaccuracy in detection results strongly influences the global integration modules of the neighboring nodes
- New security risks arise from the functionality of cooperation as a result of a malicious node transmitting false detection results ("blackmail" attack) or modifying detection results originating from another cooperating node ("man in the middle attack") in order for hindering or misleading the detection process in a node or a set of nodes

A friend assisted intrusion detection architecture for MANETs: Cooperative two-tier, one for local and one for global, detection IDS architecture for MANETs, is proposed by Razak *et al.* (2008), including two detection engines. The first-tier that uses a local-level detection mechanism, is responsible for collecting local audit data and processing them using a signature-based detection engine. If a suspicious activity is detected and it cannot determine accurately a specific attack, then a second engine is activated (also located in the first-tier) in order to perform anomaly detection. If both engines at the first-tier are unable to determine if the suspicious activity is malicious, then the second-tier of the architecture is triggered. The second-tier implements a global detection mechanism. In the process of detection, it collects audit data from the neighboring nodes, performs a signature-based detection followed by an anomaly-based detection in the same manner as in the first-tier. The second-tier also maintains a list of friends, i.e., each node builds and maintains a list of trustful nodes, which is used to ensure that the nodes sharing their audit data with it are trustful.

The strengths of the friend-assisted IDS architecture are listed below:

- Since each node in this architecture incorporates a two-tier module including two detection engines that complement each other, it possesses high detection accuracy
- Since only trustful nodes can send audit data to the second-tier of a node in the process of global detection, it is not susceptible to blackmail attacks. Due to this reason, a malicious node is unable to provide false audit data in order for misleading the IDS or falsely characterizing legitimate nodes as malicious

This architecture has the following weaknesses:

- Due to multiple detection procedures and the employment of trust management mechanism, this architecture imposes a considerable complexity and processing load
- The lack of trust relationships among nodes and nodes' mobility negatively affect the rate of false positives and the detection accuracy of the IDS. The IDS might not be able to find enough trustful nodes in a network with limited trust relationships for collecting a sufficient amount of audit data to determine if an event is legitimate. This can also result from continuous movement of trusted nodes
- This architecture incurs extra communication overhead due to three possible reasons: (1) The second tier detection needs the exchange of audit data, (2) Nodes need to exchange trust information in order to build lists of friends and (3) The implementation of signature-based detection needs the existence of a signature distribution authority that periodically transmits new signatures to each node in the network

Fork: A two pronged intrusion detection scheme for MANETs: Ramachandran *et al.* (2008) propose a cooperative IDS architecture that uses lightweight modules (agents), in order to perform different detection tasks and aim at reducing battery consumption. Each node in the network contains all the modules required to perform the intrusion detection procedure and is assigned with a reputation value that increases when the node successfully contributes to intrusion detection procedure and decreases if the node's performance during intrusion detection is unreasonable. However, there is no clarification by the authors regarding the conditions under which the performance of a node in detection procedure is defined to be unsatisfactory. The intrusion detection engine, deployed at every node in this architecture, mostly relies on anomaly detection. When this engine identifies a suspicious behavior, it triggers an auction scheme to select a set of nodes that are most suitable in fruitfully contributing to intrusion detection. Nodes with the highest amount of battery resources and reputation values are selected for the purpose and specific tasks are assigned to them. These tasks include: (1) The execution of host for network monitoring, (2) The decision making with the help of a given set of audit data and (3) The activation of defensive actions after a malicious behaviors is detected. However the authors neither clarify the procedure of nodes' cooperation nor have evaluated the communication overhead induced from this cooperation strategy. Furthermore, the node's mobility is not considered during the simulations. For this reason, the impact of mobility on the detection accuracy, the rate of false positives and the communication overhead are left undetermined.

The principal advantage of this architecture is the distribution of detection procedures among a set of nodes that reduces the processing load for the initiating node and conserves its battery power. The selection of assisting nodes also considers the available battery resources and so, nodes with lower battery power are relieved from intrusion detection responsibilities. On the other hand, this architecture has the following weaknesses:

- The communication overhead imposed by this IDS architecture may increase as a result of high nodes' mobility. A mobile node assigned with a detection task may move away from the initiating node and needs to route the results of detection through other nodes. However, this extra communication overhead has not been reported by the authors
- This architecture is vulnerable to man in the middle attacks, since a malicious node, exploiting the task allocation mechanism, may eventually capture and modify intrusion detection task messages. A malicious node might also cause blackmail attacks, by transmitting false detection results to the node that has initiated detection procedure. In addition, a malicious node may generate sleep deprivation attacks, by triggering fake tasks to other nodes in order for consuming their battery power

Routing anomaly detection architecture: A cooperative IDS architecture, proposed by Sun *et al.* (2003) focuses on routing disruption attacks. Every node in a MANET stores a routing table holding shortest possible paths to all possible destinations that triggers each node participating in the routing process. Frequent changes in this table may indicate to malicious behaviors that attempt to disrupt the routing process. In this ID, two following routing features are used for identification of malicious behaviors: (1) The percentage of changes in the route entries (PCR) and (2) The percentage of changes in the number of hops (PCH). PCR is estimated from the added/deleted route entries in a certain time interval, where PCH represents the change in the sum of hops of all route entries taken together during the considered time interval. Every node in this ID, incorporates one or several intrusion detection engines that rely on anomaly detection. These engines collect and process routing information for detecting possible intrusions, using a modified Markov Chain anomaly detection technique (Jha *et al.*, 2001). When more than one detection engines are employed in a node, alerts and reports from all local engines are combined together.

Furthermore, data reports and alerts from neighboring nodes are also interrelated in order to achieve more accurate detections. From the simulation results in Sun *et al.* (2003), the authors conclude that this IDS is capable of detecting more than 90% of the routing disruption attacks, in scenarios with comparatively low mobility of nodes (i.e., nodes speed ranges from 3-5 m sec^{-1}).

Increased detection accuracy due to the deployment of multiple detection engines at every node represents the main advantage of this architecture as compared to other single engine solutions. On the other hand, this architecture possesses a set of drawbacks:

- Since it monitors only routing attacks, it cannot be used to detect all the types of possible attacks
- Since detection engines deployed on neighboring nodes need to frequently exchange detection reports and alerts in order to achieve more accurate decisions, it incurs extra communication overhead
- Nodes' mobility negatively affects the detection accuracy and the ratio of false positives for two reasons: (1) A node in a high mobility scenario can only notice a few falsified routing changes before changing its location and (2) The changes in routing tables in such scenarios are rapid and inconsistent. Hence, there can be no enough information for the detector in order to distinguish between normal behaviors demonstrated by nodes' mobility and abnormal behaviors provoked by malicious nodes
- As a malicious node might transmit false detection reports or alerts in order to make the intrusion detection process complex and falsely accuse a legitimate node(s) as malicious, it is vulnerable to blackmail attacks

Afterwards, improved routing anomaly IDS architecture was proposed by Sun *et al.* (2007). This architecture incorporates a new intrusion detection engine with regulative thresholds. In addition, this architecture addresses most important drawback encountered by anomaly IDS architecture including negative impacts of nodes' mobility on the detection accuracy and the ratio of false positives. The technique of regulative thresholds ensures that the periodical changes in routing information due to nodes' mobility, remains under the detection threshold; while malicious behaviors that are persistent would exceed the thresholds indicating the presence of attacks. The authors in Sun *et al.* (2007) compare the initial routing anomaly detection architecture with the enhanced one based on simulation results. The enhanced architecture preserves the advantages of the initial and

reduces the negative impact of high nodes' mobility on the detection ratio and the rate of false positives. On the contrary, the method of regulative thresholds leads to new security risks. To be more specific, in a case if a malicious node observes high mobility, it might act maliciously and may remain undetected.

LIDF: Layered intrusion detection framework for *ad hoc* networks: Bose *et al.* (2007) presents a cooperative IDS architecture utilizing multilayered detection strategy, in order to capture malicious behaviors. Every host in this architecture maintains an intrusion detection system comprising of three modules: (1) The collection module, (2) The detection module and (3) The alert module. The collection module collects audit data from both the data link and the network layers. The IDS captures a close view of the networking activities by monitoring these two layers (i.e., nodes' connectivity and routing). The detection module is responsible for performing anomaly-based detection on the collected audit data, in order to preserve the host's resources and battery. It processes only most recent local audit data. If data is insufficient to achieve an accurate decision regarding a suspicious behavior; more audit data are requested from neighboring nodes via secure communication channels. Nevertheless, authors do not specify the instances when nodes decide to request for neighbors' cooperation and how this cooperation is achieved (i.e., exchange of audit data or detection results). For this reason, the communication overhead incurred by nodes' cooperation cannot be determined. In addition, if a malicious behavior is detected, the alert module needs to notify the neighboring nodes. The strengths of this layered IDS architecture are:

- It is capable of detecting attacks at both the network and data link layers using multiple layers of detection
- Man in the middle attacks are resolved by the use of secure communication channels for nodes' cooperation

On the other hand, this architecture has the following weaknesses:

- It focuses only on attacks that target the network and data link layers. Attacks at the transport layer like a SYN flooding, where a malicious node sends a large number of SYN packets, or a session hijacking attack, where a malicious node takes control over a session between two nodes remain undetected
- The detection accuracy of the IDS is reduced and the ratio of false positives is increased due to Nodes' mobility, since it makes the cooperation complex as the nodes move away from each other
- Since a malicious node that cooperates might transmit modified audit data in order to complicate the intrusion detection process, hide malicious activities or falsely accuse legitimate nodes as malicious, it is vulnerable to blackmail attacks

Strengths and weaknesses of the cooperative IDS architectures: The strengths of the cooperative IDS architectures can be summarized as: (1) In order to provide increased detection accuracy and detect a wide range of possible attacks, most of these architectures employ multiple detection engines, (2) Few of them attempt to minimize the involved processing and communication overheads through task distribution or the exchange of detection results rather than reducing the voluminous audit data exchanged among neighboring nodes and (3) Some of them attempt to resolve certain attacks using trust or secure communication channels. On the contrary, the weaknesses of these architectures can be concluded as: (1) High nodes' mobility negatively affects the ratio of false positives and detection accuracy in the entire set of these architectures, (2) Extra processing and communication overhead is incurred in all of them (3) Most of them are vulnerable to attacks like man in the middle and blackmail, etc. and (4) Audit data exchanges lead to new security risks.

Hierarchical IDS architectures: In a MANET using a hierarchical IDS architecture, the nodes are divided into two categories: cluster-heads and cluster members. The cluster members run a lightweight local intrusion detection engine, while the cluster-head runs a comprehensive detection engine that processes pre-processed audit data from all the cluster members.

A cluster-based intrusion detection architecture with adaptive selection event triggering: Ma and Fang (2008) propose a hierarchical IDS architecture that implements a modular approach to design of IDS based on clusters. The intention behind this approach is to provide a clustered structure where nodes with the highest battery power are always hosted as cluster-heads. In the process of network initialization, each node needs to report its battery power to its neighbors. Following it, the node with the highest available battery power is elected as cluster-head. The re-election process of cluster-head is triggered as soon as one the following event takes place: (1) A new node joins the network, (2) The elected cluster-head leaves the network, or (3) The battery power of the cluster-head goes below a predefined threshold. The moment a new node joins the network, it should first notify all of its neighboring nodes. Accordingly, if a cluster-head leaves

the network, it needs to broadcast a packet to notify its cluster-member nodes for initiating the cluster-head re-election process. In this IDS architecture, each network node contains four different modules, as detailed below.

Network detection module: It provides a network packet monitoring within a cluster that is activated only when the hosting node is elected as cluster-head.

Local detection module: It monitors the hosting node and generates local alerts in case malicious activities are detected. This module is always active at every node.

Resource management module: It monitors the battery power of a node acting as cluster-head. When the battery power goes below a predefined threshold, the module first notifies the monitoring state managing module and then triggers the cluster-head reelection process.

Monitoring state managing module: It monitors whether the network detection module is active (i.e., the hosting node is elected as cluster-head).

The strengths of this IDS architecture are:

- The nodes with the highest battery powers are elected to be the cluster-heads
- It supports two layers of detection (local and network) thereby increasing detection accuracy
- The cluster-head monitors the network packets exchanged and hence, there is no extra communication overhead between the cluster-head and the cluster members

On the contrary, it has the following weaknesses:

- Since the cluster-heads are responsible for running both local and network detection modules, they are unfairly overloaded
- Detection accuracy of the architecture may be reduced and the ratio of false positives may be increased due to High nodes' mobility, since a number of nodes may move out of the range of a cluster-head. This feature limits the information which the network detection module needs to perform detection procedure
- The creation and maintenance of clusters and the election of cluster-heads induce extra processing and communication overhead
- With a few nodes carrying the responsibility of intrusion detection, it may create points of failure, at least locally in a cluster. If a cluster-head crashes, is attacked, or leaves the cluster or the network without initiating the re-election procedure, only the local detection modules will protect the nodes
- Since the communication channels between the network nodes are not protected, it is vulnerable to man in the middle and blackmail attacks. Hence, a malicious node can be able to modify the transmitted messages in order to mislead the cluster-head
- In order to be elected as cluster head, a malicious node may exploit the election procedure by reporting false values of battery power. Likewise, a selfish node may avoid becoming a cluster-head

A hierarchical IDS architecture that uses a game theoretic detection mechanism: A hierarchical approach to IDS design is described in Otrok *et al.* (2008) attempts to balance the consumption of resources among the nodes of a cluster resulting from intrusion detection tasks. It facilitates network nodes to participate in the election process of cluster-heads and attempts to prevent elected cluster-heads from misbehaving. In this proposed architecture, nodes can act as: (1) Cluster-members that carry no intrusion detection responsibilities, (2) Cluster-heads, which are responsible for intrusion detection within a cluster; or (3) Checkers that are cluster-members selected randomly in order to monitor the cluster-head for selfish or malicious behavior. In the process of initialization, the network nodes report the power of their batteries to their neighboring nodes, following which every node creates a list composed of its neighbors' battery power levels. Then, each node votes the node with the highest battery power, Based on this list, to be elected as cluster-head. The elected cluster-head then, deploys a detection engine that is based on a zero sum, non-cooperative game, where the cluster-head and a possible intruder are the players.

The cluster-head monitors only those nodes that participated in the election process. The election process is repeated (after a time period elapses) and a new cluster-head is elected depending on the battery power of the elected cluster-head. The randomly selected checkers are responsible for partially monitoring the cluster-head for selfish or malicious behavior. In case a checker acquires some indications of cluster-head misbehavior, it cooperates with other checkers in order to conclude to a decision. A simulation of the proposed architecture is carried out by the authors with 20 nodes and record their power levels at three distinct time moments (i.e., 0 sec, after 1500 sec and after 3000 sec) (Otrok *et al.*, 2008). At the beginning of the simulation (0 sec), 8 nodes possess power levels between 100- 80% and 12 nodes between 80 and 60%. After 1500 sec, only 3 of them maintain

power level between 100-80%, 4 between 80 and 60%, 7 between 60 and 40%, 3 between 40 and 20% and 3 below 20%, none of them runs out of battery power. After 3000 sec, 2 nodes preserve power level between 100 and 80%, 2 of them have power level between 80 and 60%, none between 60 and 40%, 2 between 40 and 20%, 7 between 20 and 0% and 7 run out of battery. Thus, it can be concluded that this architecture imposes unfair power consumption among the network nodes. Furthermore, since the authors do not take into consideration the nodes' mobility in the carried simulations and thus, its impact on the detection accuracy and the rate of false positives of the architecture cannot be determined.

The operational strengths of this architecture are summarized below:

- The nodes with the highest battery power are elected to act as cluster-heads
- Misbehaving cluster-heads can be detected from the randomly selected checkers those monitor them

On the other hand, the main weaknesses of the architecture can be inferred as:

- Cluster-heads/checker nodes are unfairly overloaded with intrusion detection responsibilities
- Extra processing and communication overhead are induced by it due to: (1) The formation and maintenance of clusters and (2) The operation of checker nodes
- Since the communication channels among the network nodes are not protected, it is vulnerable to man in the middle and blackmail attacks. A malicious node may tend to capture, modify the message and re-transmit modified messages in order for misleading the cluster-head
- A cluster-head may pose a single point of failure in each cluster. An attack or malfunction of the cluster-head complicates the intrusion detection procedure in the corresponding cluster
- Selfish nodes may tend to exploit the employed election process by reporting false battery power values in order for participating in the process but avoid being cluster-heads
- If a malicious node is selected to be a checker, it may falsely accuse a cluster-head for misbehaving

A clustered architecture that uses collective decision for intrusion detection: Two intrusion detection architectures, proposed by Marchang and Datta (2008), rely on a voting scheme to carry out intrusion detection procedure rather than employing an anomaly or signature based intrusion detection engine. The difference between the two proposed architectures is: (1) The first one called algorithms for detection in a clique (ADCLI), divides the network into cliques and (2) The second, called algorithm for detection in a cluster (ADCLU), divides the network into clusters. A clique is similar to a cluster, but with the only difference that each member of a clique is a neighbor with all the others members. In each cluster or clique, where intrusion takes place independently, a monitoring node is elected using various strategies that is rotated periodically. On receiving any suspicious or modified message from a member of its clique/cluster, the monitoring node asks the other clique/cluster members to trigger the intrusion detection procedure. In the process of this procedure, the monitoring node sends a message to all the other clique/cluster members, which subsequently forward this message to their neighboring clique/cluster members. If any of the clique/cluster members receives a modified message (or no message at all), it marks the corresponding node that transmitted the modified message (or did not transmit anything) as suspicious. In addition, there is a voting stage where every clique/cluster member notifies the monitoring node which nodes it believes to be suspicious. Then, the monitoring node decides which nodes are malicious, based on the votes received from the clique/cluster members and a threshold value. It should be noted that the authors assumed that a monitoring node can never be malicious and it is changed periodically in order to prevent unfair use of its resources and battery depletion. Low processing and communication overhead of these architectures is their principal advantage. The reason behind this is that both of them avoid using bandwidth or computation intensive operations, such as sharing audit data or employing anomaly detection algorithms. Te only traffic exchanged among the clique/cluster members are the monitoring and voting messages of the detection procedure.

On the contrary, both architectures present the following weaknesses:

- As reported by the authors from performed simulations, the ratio of false positives increased substantially when packet loss reached or exceeded 9% for the ADCLI and 12% for the ADCLU, respectively. Thus, in an environment that is characterized by high packet loss (e.g., due to high nodes' mobility or the presence of selfish nodes that drop packets), both architectures are assumed to be ineffective
- As the monitoring node poses a single point of failure in the respective clique/cluster, in case an

attack occurs against the monitoring node or it fails, the intrusion detection process is disabled subsequently
- These architectures are unable to detect any type of attack that does not modify or drop packets (such as man in the middle, replay, flooding, session hijacking, etc.)
- Malicious nodes may tend to exploit the detection scheme by voting legitimate nodes as malicious

An optimal hierarchical intrusion detection architecture: A hierarchical IDS architecture, proposed by Manousakis *et al*. (2008), uses a dynamic tree based structure, where detection data are aggregated upwards, i.e., from leaf nodes to authoritative nodes at the level of the root of the hierarchy (i.e., upper layer nodes) and the latter needs to dispatch directives down to the former (i.e., lower-level nodes). There are main objectives of this architecture: (1) In order to achieve a tree-based structure which is robust to network changes and triggers the rapid aggregation of detection data and (2) In order to detect attacks at a particular level of the hierarchy where enough aggregated detection data are available to infer an accurate decision. The tree-based structure is formed and two algorithms, namely, the initial solution generation procedure and the state transition mechanism, are used to maintain this tree-based structure. The initial tree-based structure is created by the first algorithm in two steps. In the first step, a network node is randomly selected to be a cluster-head and its neighbors are assigned as cluster members of the cluster.

The selected cluster-head exists at the highest level of hierarchy in the tree-based structure. In the second step, a cluster-member of the previously created cluster(s) is selected to be the cluster-head and its neighbors that have not been previously assigned to any other cluster, are assigned to it as cluster-members. The second step is repeated until all the network nodes become members of the tree-based hierarchical structure. The state transition mechanism modifies the initially created tree-based hierarchy using some permutations, in order to be robust to network changes and trigger the rapid aggregation of detection data. More specifically, it reassigns some of the branches of the tree-based structure (i.e., relationships between a cluster-head and cluster members), in order to achieve two goals: (1) The modified tree should possess the shorter possible height, and (2) It is estimated that the modified structure has a better longevity than any other, considering location of the node in the network topology, the speed of node and the range of transmission of the node. Intrusion detection takes place at the lowest possible level of the hierarchical structure, at which there are enough aggregated data used for an accurate decision. In case the responsible cluster-head in a cluster is not capable of detecting an attack accurately, it needs to forward all the related detection data to a higher-level cluster-head, which in turn attempts to accurately detect the attack.

The proposed IDS architecture has the following strengths:

- Since clusters are selected with the objective of "higher longevity", it tends to be more robust under high nodes' mobility
- Since it supports multiple levels of detection, it provides increased detection accuracy as compared to other single level detection architectures. The collected data are forwarded upwards until they reach a certain level where intrusion decision can be achieved

This architecture also possesses some weaknesses:

- Since lower level cluster-heads deal with detections constantly, they are unfairly overloaded, while higher-level cluster-heads perform detections only in case when lower level cluster-heads cannot resolve a malicious behavior
- It imposes extra processing and communication overhead for creating and maintaining the hierarchical structure. Furthermore, during permutations in the state transition mechanism, the several required calculations are carried out at each iteration
- Since the communication channels between the network nodes are not protected, it is vulnerable to man in the middle and blackmail attacks. A malicious node may tend to capture and re-transmit modified messages in order for misleading the cluster-head
- In case a cluster-head is compromised, it may trigger false alarms to the lower-level nodes and falsely characterize legitimate nodes as malicious that impose damage to the network. If the randomly selected node at the highest level of the hierarchy is malicious, it can complicate intrusion detection procedure throughout the entire lifetime of the network
- A malicious node or set of malicious nodes may exploit the tree optimization procedure in order to elect a malicious node as cluster-head by reporting false parameters to the state transmission mechanism. Similarly, a selfish node may tend to avoid becoming a cluster-head

Clustered anomaly detection architecture: A clustered IDS architecture presented in Deng *et al.* (2006), allows only the cluster-heads carry out intrusion detection process. It focuses on detecting attacks targeting at the routing infrastructure of a network and creates clusters using the "Distributed Efficient Clustering Approach" (DECA) protocol. In this protocol, each node votes for its neighboring node that has the highest number of connections and residual energy, as cluster-head. The nodes with the most number of votes become cluster-heads. Cluster-heads are re-elected after a predefined period of time. Each cluster head deploys an anomaly detection engine for monitoring: (1) The propagation of protocol specific routing packets (i.e., hello, error, request, reply, etc.), (2) The modifications in routing tables and (3) The transmission of data packets. These features are monitored either randomly by selecting a cluster member which transmits its own set of features to the cluster head, or actively by configuring the cluster head to listen to the traffic generated in the cluster.

The operational strengths of the clustered anomaly detection architecture are:

- Processing workload is fairly distributed among the nodes as the cluster-heads rotate after a specific period of time
- Considering nodes' connectivity in cluster-heads election infers that the elected cluster-heads monitor large spectrum of network activities that facilitates the IDS to achieve more accurate detections

The distinguishing weaknesses of this IDS architecture can be summarized as:

- The employed detection engine can detect only routing attacks
- The basic weaknesses that emerge from previously analyzed hierarchical architectures also hold: (1) Cluster-heads may represent points of failure, (2) Malicious or selfish nodes that do not cooperate may complicate or mislead intrusion detection process, (3) Malicious nodes may falsely accuse other legitimate nodes as malicious, (4) Malicious nodes may exploit the scheme of electing cluster-heads and (5) The employed election schemes do not take into consideration the processing capabilities of nodes

Strengths and weaknesses of the hierarchical IDS architectures: This section outlines the basic strengths and weaknesses of the discussed earlier hierarchical IDS architectures emerging from the carried analysis and evaluation, allowing their comparison. The strengths of hierarchical IDS architectures can be summarized as:

- The majority of them focuses on increasing the detection accuracy either by employing multiple layers of detection method or by employing one cluster-head to monitor large portions of a network or by monitoring the elected cluster-heads
- Some of them aim at the fair distribution of the processing workload among nodes either by considering nodes battery power, or by rotating cluster heads
- A few of them attempt to eliminate the imposed processing and communication overhead either by employing a detection mechanism based on voting or by selecting cluster heads with the objective of "higher longevity"

On the other hand, the weaknesses of this architecture are:

- The entire set of the studied hierarchical IDSs is vulnerable to a variety of attacks (i.e., man in the middle, blackmail, exploitation of the employed election scheme, malicious nodes may hinder or mislead detection, etc.)
- In most of such architectures, cluster heads are the points of failure
- Many of them impose extra processing and communication overhead during the formation and maintenance of clustered structures
- In few of these architectures, elected cluster-heads appear to be unfairly overloaded
- Some of them detect only specific types of attacks and are negatively affected by high nodes' mobility

Intrusion detection algorithms: Manikandan and Manimegalai (2011) provide a set of intrusion detection classification algorithms on MANET. Mitrokotsa *et al.* (2008) use the linear classifier, Gaussian mixture model and support vector machine approaches in order to evaluate the performance of intrusion detection systems for MANET. In which, classification algorithm was evaluated in varying mobility patterns and different traffic conditions. Authors take into consideration the attacks like Black Hole, Forging, Packet Dropping and Flooding attacks during their investigation.

Ahmed *et al.* (2007), propose a modified version of the Dynamic Source Routing (DSR) protocol in order to enhance the security features and introduce intrusion detection into it. This architecture comprises of fixed wired network with multi-hop wireless nodes like PDA,

Laptop or smart phones. A novel anomaly detection system consisting of detection subsystems at the MAC layer, IP layer and application layer is addressed in Bose *et al*. (2007). In which, the detection framework residing at each layer is provided with the data collected from network traffic. On identification of a violation from the normal behavior, an anomaly can be detected depending on predefined thresholds. A local integration module integrates the detection results obtained from detection subsystems residing at all the three layers and the final result is sent to the global integration module. Shrestha *et al*. (2010) propose a novel cross layer intrusion detection architecture in order to discover the malicious nodes and different types of denial of service attacks with the exploitation of the information available across different layers of protocol stack for improving the accuracy of detection. In addition, this study describes cooperative anomaly intrusion detection with data mining technique.

CONCLUSION AND FUTURE WORK

Implementation of *ad hoc* networks have become increasingly popular among the researchers, industry personnel, strategic organizations due to its simplicity of deployment and cost effectiveness. However, since it does not support any fixed infrastructure and operates on a resource constrained wireless channel, its effective and secured operation is a huge challenge. Intrusion detection has been a major concern for efficient implementation of variety of applications on MANETs from security point of view. IDS solutions proposed for wired networks do not sufficiently comply with the challenges imposed by MENETS due to its diversified dynamics and mobility of nodes in it. It motivates researchers to come up with newer solutions that can take into account the diversities of applications on MANETs running in wireless environment. This study covers a thorough investigation on Intrusion detection in MANETs. A set of IDS architectures such as stand alone, cooperative, hierarchical and clustered architectures along with their implementation details, advantages and limitations, is discussed. The advantages and disadvantages of IDS architectures as a whole are summarized. It is assumed that this work will contribute significantly to researchers in the field of IDS and trigger them to come up with more robust solution in highly dynamic environment such as MANET. As an extension to our endavour in this context we intend to carry our research work forward to develop an agent based IDS mechanism that would successfully address most of the challenges related to intrusion detection in MANETs.

REFERENCES

Ahmed, T., S. Ahmed, M.N. Haque and M. Masum, 2007. Modification of DSR and its implementation in *ad hoc* city. Proceedings of the 10th International Conference on Computer and Information Technology, December 27-29, 2007, Dahaka, pp: 1-6.

Bose, S. S. Bharathimurugan and A. Kannan, 2007. Multi-layer integrated anomaly intrusion detection system for mobile *ad hoc* networks. Proceedings of the International Conference on Signal Processing, Communications and Networking, February 22-24, 2007, Chennai, pp: 360-365.

Conti, M., 2003. Body, Personal and Local *ad hoc* Wireless Networks. In: The Handbook of *ad hoc* Wireless Networks, Ilyas, M. and R.C. Dorf (Eds.). CRC Press, Boca Raton, FL., ISBN: 9780849313325, pp: 3-24.

Corson, M.S., J.P. Marker and G.H. Cirincione, 1999. Internet based mobile *ad hoc* networking. IEEE Internet Compt., 3: 63-70.

Deng, H., R. Xu, J. Li, F. Zhang, R. Levy and W. Lee, 2006. Agent-based cooperative anomaly detection for wireless *ad hoc* networks. Proceedings of the 12th Conference on Parallel and Distributed Systems, July 12-15, 2006, Minneapolis, MN., pp: 613-620.

Huang, Y., W. Fan, W. Lee and P.S. Yu, 2003. Cross-feature analysis for detection *ad hoc* routing anomalies. Proceedings of the 23rd International Conference on Distributed Computing Systems, May 19-22, 2003, USA., pp: 478-487.

Imella, P. and O. McMillan, 2001. An introduction to intrusion detection system. Tetrad Digital Integrity, LLC. http://www.symantec.com/connect/articles /introduction-ids.

Jacoby, G.A. and N.J. Davis, 2007. Mobile host-based intrusion detection and attack identification. IEEE Wireless Commun., 14: 53-60.

Jha, S., K. Tan and R.A. Maxion, 2001. Markov chains, classifiers and intrusion detection. Proceedings of 14th IEEE Computer Security Foundations Workshop, June 11-13, 2001, Cape Breton, Nova Scotia, Canada, pp: 206-219.

Kheyri, D. and M. Karami, 2012. A comprehensive survey on anomaly-based intrusion detection in MANET. Comput. Inform. Sci., 5: 132-139.

Lauf, A.P., R.A. Peters and W.H. Robinson, 2010. A distributed intrusion detection system for resource-constrained devices in *ad hoc* networks. J. *ad hoc* Networks, 8: 253-266.

Ma, C.X. and Z.M. Fang, 2008. A novel intrusion detection architecture based on adaptive selection event triggering for mobile *ad hoc* networks. Proceedings of the IEEE 2nd International Symposium on Intelligent Information Technology and Security Informatics, January 23-25, 2008, Moscow, pp: 198-201.

Manikandan, S.P. and R. Manimegalai, 2011. Evaluation of intrusion detection algorithms for interoperability gateways in *ad hoc* networks. Int. J. Comput. Sci. Eng., 3: 3243-3249.

Manousakis, K., D. Sterne, N. Ivanic, G. Lawler and A. McAuley, 2008. A stochastic approximation approach for improving intrusion detection data fusion structures. Procedings of the IEEE Military Communications Conference, November 16-19, 2008, San Diego, CA., pp: 1-7.

Marchang, N. and R. Datta, 2008. Collaborative techniques for intrusion detection in mobile *ad hoc* networks. *ad hoc* Networks, 6: 508-523.

Mishra, A. and K.M. Nadkarni, 2003. Security in Wireless *ad hoc* Networks. In: The Handbook of *ad hoc* Wireless Networks, Ilyas, M. and R.C. Droft (Eds.). CRC Press, Boca Raton, FL., USA., ISBN: 9780849313325, pp: 499-549.

Mitrokotsa, A., M. Tsagkaris and C. Douligeris, 2008. Intrusion detection in mobile *ad hoc* networks using classification algorithms. Proceedings of the 7th Annual Mediterranean *ad hoc* Networking Workshop, June 25-27, 2008, Palma de Mallorca, Spain, pp: 133-144.

Nadkarni, K. and A. Mishra, 2004. A novel intrusion detection approach for wireless *ad hoc* networks. Proceedings of the IEEE Wireless Communications and Networking Conference, Volume 2, March 21-25, 2004, Atlanta, Georgia, USA., pp: 831-836.

Otrok, H., N. Mohamm, L. Wang, M. Debbabi and P. Bhattacharya, 2008. A game-theoretic intrusion detection model for mobile *ad hoc* networks. Comput. Commun., 31: 708-721.

Phatak, R. and D.K. Mishra, 2012. Distributed intrusion detection scheme for wireless *ad hoc* networks: A review. Proceedings of the 6th International Conference on Software Engineering, September 5-7, 2012, Indore, pp: 1-6.

Ramachandran, C., S. Misra and M.S. Obaidat, 2008. FORK: A novel two-pronged strategy for an agent-based intrusion detection scheme in *ad hoc* networks. Comput. Commun., 31: 3855-3869.

Razak, S.A., S.M. Furnell, N.L. Clarke and P.J. Brooke, 2008. Friend-assisted intrusion detection and response mechanisms for mobile *ad hoc* networks. *ad hoc* Networks, 6: 1151-1167.

Sen, S., 2010. Evolutionary computation techniques for intrusion detection in mobile *ad hoc* networks. Ph.D. Thesis, University of York.

Shrestha, R., K.H. Han, D.Y. Choi and S.J. Han, 2010. A novel cross layer intrusion detection system in MANET. Proceedings of the 24th IEEE International Conference on Advanced Information Networking and Applications, April 20-23, 2010, Perth, WA., pp: 647-654.

Sun, B., K. Wu and U.W. Pooch, 2003. Routing anomaly detection in mobile *ad hoc* networks. Proceedings of the 12th IEEE International Conference on Computer Communications and Networks, October 20-22, 2003, Santa Clara, CA., USA., pp: 25-31.

Sun, B., K. Wu, Y. Xiao and R. Wang, 2007. Integration of mobility and intrusion detection for wireless *ad hoc* networks. Int. J. Commun. Syst., 20: 695-721.

Tseng, C.H., S.H. Wang, W. Lee, C. Ko and K. Lewitt, 2003. Demem: Distributed evidence driven message exchange intrusion detection model for MANET. Proceedings of the 9th International Symposium on Recent Advances in Intrusion Detection, September 20-22, 2006, Germany, pp: 249-271.

Vishwakarma, D. and D. Chopra, 2012. An efficient attack detection system for mobile *ad hoc* network. Int. J. Eng. Adv. Technol., 1: 21-27.

Wang, W., H. Man and Y. Liu, 2009. A framework for intrusion detection systems by social network analysis methods in *ad hoc* networks. Secur. Commun. Networks, 2: 669-685.

Weiser, M., 1999. The computer for the 21st century. ACM SIGMOBILE Mobile Comput. Commun. Rev., 3: 3-11.

Xenakis, C., C. Panos and I. Stavrakakis, 2011. A comparative evaluation of intrusion detection architectures for mobile *ad hoc* networks. Comput. Security, 30: 63-80.

Zeng, X., R. Bagrodia and M. Gerla, 1998. GloMoSim: A library for parallel simulation of large-scale wireless networks. Proceedings of the 12th Workshop on Parallel and Distributed Simulation, May 26-29, 1998, Banff, Alta, Canada, USA., pp: 154-161.

Zhang, G.Y. and W. Lee, 2000. Intrusion detection in wireless *ad hoc* networks. Proceedings of the 6th International Conference on Mobile Computing and Networking, August 6-11, 2000, Boston, MA., USA., pp: 275-283.