International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:11, No:6, 2017

A Distributed Mobile Agent Based on Intrusion Detection System for MANET

Maad Kamal Al-Anni

Abstract**—**This study is about an algorithmic dependence of Artificial Neural Network on Multilayer Perceptron (MPL) pertaining to the classification and clustering presentations for Mobile Adhoc Network vulnerabilities. Moreover, mobile ad hoc network (MANET) is ubiquitous intelligent internetworking devices in which it has the ability to detect their environment using an autonomous system of mobile nodes that are connected via wireless links. Security affairs are the most important subject in MANET due to the easy penetrative scenarios occurred in such an auto configuration network. One of the powerful techniques used for inspecting the network packets is Intrusion Detection System (IDS); in this article, we are going to show the effectiveness of artificial neural networks used as a machine learning along with stochastic approach (information gain) to classify the malicious behaviors in simulated network with respect to different IDS techniques. The monitoring agent is responsible for detection inference engine, the audit data is collected from collecting agent by simulating the node attack and contrasted outputs with normal behaviors of the framework, whenever. In the event that there is any deviation from the ordinary behaviors then the monitoring agent is considered this event as an attack , in this article we are going to demonstrate the signature-based IDS approach in a MANET by implementing the back propagation algorithm over ensemble-based Traffic Table (TT), thus the signature of malicious behaviors or undesirable activities are often significantly prognosticated and efficiently figured out, by increasing the parametric set-up of Back propagation algorithm during the experimental results which empirically shown its effectiveness for the ratio of detection index up to 98.6 percentage. Consequently it is proved in empirical results in this article, the performance matrices are also being included in this article with Xgraph screen show by different through puts like Packet Delivery Ratio (PDR), Through Put(TP), and Average Delay(AD).

Keywords**—**Mobile ad hoc network, MANET, intrusion detection system, back propagation algorithm, neural networks, traffic table, multilayer perceptron, feed-forward back-propagation, network simulator 2.

I. Introduction

A MANET is a self-configuring network that is formed automatically likewise the mobile nodes are free to move randomly and no base station can restrict its topological changes [1], [2]. Let us have a look to the normal functionality of MANETs, Each node is equipped with a wireless transmitter and receiver, in an environment like MANET environment needs to all nodes in the network collaborating for network sustainability, moreover those nodes sometimes behave like the host of in-coming packets and while being a router at the same time, due to the shared nature of wireless channels, noise within the channels, and instability caused by mobility, MANET is much more vulnerable to attacks than wired networks [2]. Threats in MANETs Security are an important issue for MANETs wherever either it is leaked to network-based access control mechanisms as firewalls or cryptographic systems, MANETs are especially used in tactical networks and emergency services. ad hoc networks useful as infrastructure not available or moreover no fix centralized management, and thus wireless communication is much more vulnerable to attacks than wired networks [4], An IDS is a computer system that dynamically monitors the system user actions in the network and computer systems in order to detect intrusions and report unauthorized or malicious network activity [5], IDS falls into three approaches first the signature-based IDS uses pre-known attack scenarios (or signatures) and compare them with incoming packets traffic, secondly the anomaly-based IDS attempts to detect activities that differ from the normal expected system behavior, ultimately.

The Specification based IDS is mingled between both the misused and the anomalous detection approaches. Machine learning is a computer software relying on IDS approaches to mechanize the scenarios of network attacks and building a digitalized software to detect and warn the administrator about its risks, in spite of BP has utilized for a considerable length of time and a capable machine learning calculation, Feed-Forward Back-Propagation (FF-BP) Frameworks are still the most normally powerful ANN topology to achieve the mathematically sophisticated relationships. FF-BP ANNs are connected in a broad scope of applications, including speech recognition, remote sensing and image classification, handwritten character/digits recognition, data mining, information retrieval, traveling sales man, VLCI placement and routing, and last not the least the IDS Multi-layered ANN performance depends mainly upon the Network Topology, when the MLP technique is parameterized with different values which yields to obtain the different information about the performance of ANN throughputs and asymptotic. There are many algorithms for ANN learning algorithm such as Adaline, Hebbian, Perceptron Learning rule, Back propagation, Artificial Bee Colony.

II. Related Works

A vast majority researches for a MANET are namely either distributed approach (i.e., local - node based IDS) or synergistic

Maad Kamal Al-Anni is with N.I. Lobachevsky Institute OF Computer Mathematics and Informational Technologies, Kazan (Volga Region) Federal University, 18 Kremlyovskaya st., Kazan 420008, Russia (e-mail: maadk-anni@live.com).

approach (i.e., clustered-node based IDS), hereabout we will have a short glance about the most highlighted research works.

S. Madhavi et al [9] proposed MIDS (Mobile Intrusion Detection System) through a scrupulous study of diverse vulnerabilities in MANET, they imbedded the IDS as the security architecture for adhoc wireless networks, which detects nodes misbehavior, anomalies in packet forwarding, such as intermediate nodes dropping or delaying packets, eavesdropping and masquerading are also applicable, thus implementation with multilayer security protections.

Angelo Rossi et al [10] have proposed IDS which is effectively consummated the conniving perilous element into the count of the path reliability, which considers the distance and the popularity concerning nodes, so be able examining both the source profile and the retransmitted one. Also, theirs prolonged structure efficiently detects malicious and conniving nodes within the order after isolating them or shield the network. The simulations launched within a variety of MANETs containing such a quite proportions concerning malicious as well as conniving nodes, it shows as their proposed solution in which offers a considerable throughput gain compared to current solutions.

Depren et al [11] has described a hybrid IDS architecture comprising three models, i.e., an anomaly analyzer, misuse analyzer and decision support system (DSS). The anomaly analyzer uses an unsupervised Self-Organizing Map(SOM) to model normal network traffic, whereas the misuse analyzer employs Decision Tree to classify attacks. The DSS aims at interpreting the combined results of anomaly and misuse analyzers. The anomaly analyzer is further specialized into three sub-analyzers based on three different types, i.e., TCP, UDP, and ICMP. Given a test instance, both anomaly and misuse analyzers are operated concurrently to identify the instance's types. Then, DSS assigns a class label for the test instance based on the results from the two analyzers.

Bo Sun et al [12] have presented a non-overlapping Zone-Based Intrusion Detection System (ZBIDS) that fits the requirement of MANETs. On the local detection part, they have presented a general intrusion detection agent model and propose a Markov Chain based anomaly detection algorithm. They have focused on the protection of MANET routing protocols and present the details regarding feature selection, data collection, data pre-process, Markov Chain construction, classifier construction and parameter tuning.

Finally, *Power and He* [13] describe a hybrid approach that integrates the advantages of both Artificial Immune System (AIS) and Kohonen SOMs, in which the output of the AIS is taken as input of the SOM. The AIS analyzes network connection records to tell whether they are normal or anomalous. Subsequently, the records that the AIS flags as anomalous are passed to the SOM. then, the SOM is in charge of clustering these anomalous records based on the specific characteristics of attacks.

III. System Architecture

IDS monitors packet data streaming in the networks in order to discover any abnormal behaviors hereafter it is its responsibility to immediately alerts the network administrator for a such detecting alarms. The class imbalance problem is one of the most important factor that sometimes led to the deterioration of overall performance, hence the reduction in terms of data set sizable is needed by redundant of some of the features that contribute little to the detection process. To handle the reduction in the size of data set we use an oversampling technique that is semantically similar to [8], the basic idea is to amplify the density of attack instances by generating synthetic attack instances rather than simply replicating real attack.

The proposed IDS applied in MANET is distributed in nature so it consisted of a collecting agent and a mobile agent equipped with an IDS. System architecture of proposed IDS comprises four components as shown in Fig. 1.

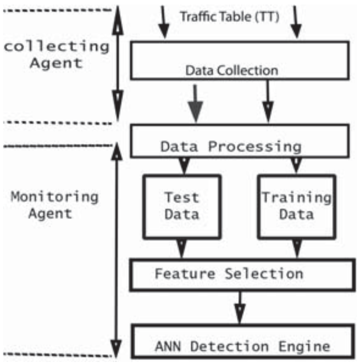

Fig. 1 A proposed System for Distributed Mobile IDS Agent

A. Data Collection

The collecting agent is responsible for data collection, therefore the proposed system is highly dependent about predominating normal instances over attack instances while both instances are created during the simulation time, likewise this dataset was derived from real life space network with simulating the normal scenario (normal packet follow) as well as the attack scenario (black hole and flooding attacks).

B. Data Processing

Audit data is transformed into an appropriate format which be utter understandable in the computerization process. Ultimately processed datum fall into three parts (training set, testing set, validated set), training data set is created to train the machine learning classifier. Training data consists of labeling of events whether it is a normal event or an attack. Test and validated data is collected under a simulated attack environment and it is verified to pinpoint an event whether it is an attack or normal with real life circumstances, while the overall architecture for training, validation, and testing data set have the same network topology and Architecture (Input-Hidden Units-Outputs), Numbers of hidden layers, type of activation function used, as shown in Table I.

TABLE I
The Back-Propagation Architecture

Data Set	Architecture	Weights	Training	Validation	Testing
Flooding Attack	18-6-1	114	360	180	180
Block hole Attack	18-6-1	114	360	180	180

C. Feature Selection

It is important to illustrate that the feature selection process is concerning with the technique that groups the relevant features which effect the detection process. BP-based Feature selection algorithm is based on the wrapper model [14].

This technique first calculates the information gain(*IG)* of each feature, and then outputs the rank of each feature based on information gain values. The higher the information gain, the more important the feature is with respect to the class (target variable). The information gain *IG* (F_j) calculated by (1) of feature F_j, given class variable Y, is calculated by the formula entropy where (2) and (3) are the entropy of Y before and after observing F_J, respectively [14].

$$IG(F_j) = H(Y) - H(\frac{Y}{F_j}) \qquad (1)$$

$$H(Y) = -\sum_{y\in Y} P\,(Y) \log_2 P\,(y) \qquad (2)$$

$$H\left(\frac{Y}{F_j}\right) = \sum_{x\in F_j} P\,(X) \sum_{y\in Y} P\,(\frac{y}{x}) \log_2 P\,(\frac{y}{x}) \qquad (3)$$

D. Artificial Back Propagation

An Artificial Neural Network(ANN) is an information processing paradigm that is inspired by biological nervous system. ANN can fall into too many categories that depend on different criteria like (number of layers, fully or partially connected network, forward or backward learning process, etc.), when the network is used to identify the input pattern and associating with the output pattern, this process is widely used for classification and clustering, MLP is an machine learning approach that has been used for classification the Mobile ad hoc threats (1 used for True Positive rate and 0 used for true Negative Rate) and likewise it will be able to learn with a supervised learning algorithm, i.e., Back propagation is a currently significant research area in the following specific paradigm: hand-writing word recognition, biomedical system, Network Security and etc. [6], Back propagation algorithm is a set of inputs and outputs along with partially or fully connected neurons, in which certainly contained hidden layers for its topology as shown in Fig. 2, the goal of back propagation searches for weight values that minimize the total error of the network over the set of training patterns. It is comprised of two stages:

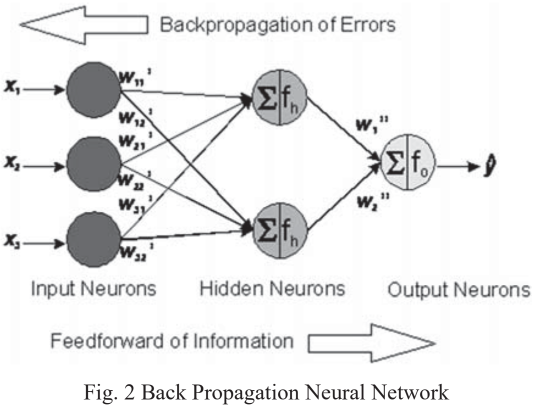

Fig. 2 Back Propagation Neural Network

- **Forward Pass:** The total net input to each hidden layer neuron figured by (4), cascade the aggregate net information utilizing an enactment work (here we utilize the calculated capacity), then rehash the procedure with the output layer neurons.

$$net_{h_i} = \omega_1 * i_1 + \cdots\cdots\cdots + \omega_n * i_n + b_1 * 1 \qquad (4)$$

We then squash it using the logistic function shown in (5) to get the output of h_i.

$$out_i = \frac{1}{1+e^{-net_{hi}}} \qquad (5)$$

Carrying out the same process for all hidden layer neurons calculated by (6) and (7), we repeat this process for the output layer neurons, using the output from the hidden layer neurons as inputs.

$$net_{o_i} = \omega_1 * out_{h_i} + \cdots\cdots\cdots + \omega_n * out_{h_n} + b_2 * 1 \qquad (6)$$

$$out_{o_i} = \frac{1}{1+e^{-net_i}} \qquad (7)$$

And carrying out the same process for all output neurons:

- **Stopping Criterion:** One of these stopping criterion is the fitness value. Since the BA algorithm is chosen to be a supervised learning algorithm, then there are observed values of (out) and desired output values of (target), the closer the value of both target and out values, the better the fitness function, otherwise the algorithm carry on the updating weight until no changes is been occurred or the number of iterations is over, mean square error is one of these stopping criteria denoted by MSE is calculated by (8), it tells us how close regression line is to a set of points.

$$MSE = E_{total} = \sum \frac{1}{2}(target - output)^2 \qquad (8)$$

The total error as calculated by (9) for the neural network is the sum of these errors:

$$E_{total} = E_1 + \cdots\cdots\cdots + E_n \qquad (9)$$

- **The Backwards Pass**: The goal with back propagation at the backwards pass diminishes the error producing during the forwards pass, consequently error is appertaining in accordance with change the weights within such a way that the error will arrive smaller, the method is repeated once more or again till the optimal function is minimal, that is computed by using steps backwards.
- **Output-hidden layer update:** Back propagation algorithm is based on delta learning rule in which the weights modification is done through mean square error of the output mapping layer to the input mapping layer which is modulated the layer adjustment by (10) and (11), sometimes called **Chain Rule** as $\frac{\partial \mathrm{Etotal}}{\partial \mathrm{Wi}}$ is a partial derviation of E_{total} with respect to W_i.

To find out how much adequate modification of error change that must be occurred to reach the global minima with respect of the output, so we need to figure out each piece in this question and evaluate it relying on the differential derivative.

$$E_{total} = \tfrac{1}{2}(target_{o_i} - out_{o_i})^2 + \tfrac{1}{2}(target_{o_i} - out_{o_i})^2 \qquad (10)$$

$$\frac{\partial E_{total}}{\partial out_{o_i}} = 2 * \tfrac{1}{2}(target_{o_i} - out_{o_i})^{2-1} * -1 + 0 \qquad (11)$$

The partial derivative by (13) of the logistic function (12) is the output multiplied by 1 minus the output:

$$out_{o_i} = \frac{1}{1+e^{-net_{oi}}} \qquad (12)$$

$$\frac{\partial out_{o_i}}{\partial net_{o_i}} = out_{o_i}(1 - out_{o_i}) \qquad (13)$$

Finally, how much does the total net input of Oi change with the respect to $\mathcal{W}i$ seen in (14) and (15):

$$net_{o_i} = \omega_i * out_{h_i} + \cdots\cdots\cdots + \omega_n * out_{h_n} + b_2 * 1 \qquad (14)$$

$$\frac{\partial net_{o_i}}{\partial \omega_i} = 1 * out_{h_i} * \omega_i^{(1-1)} + 0 + 0 = out_{h_i} \qquad (15)$$

This calculation combined (11), (13) and (15) in the form of the delta rule, thus it is obtained (16):

$$\frac{\partial E_{total}}{\partial \omega_i} = -(target_{o_i} - out_{o_i}) * out_{o_i}(1 - out_{o_i}) * out_{h_i} \qquad (16)$$

Alternatively, we have $\frac{\partial \mathrm{Etotal}}{\partial \mathrm{O}\mathbb{UT}\mathrm{oi}}$ and $\frac{\partial \mathrm{O}\mathbb{UT}\mathrm{oi}}{\partial \mathbb{NET}\mathrm{oi}}$ which can be written as $\frac{\partial \mathrm{Etotal}}{\partial \mathbb{NET}\mathrm{oi}}$, aka $\mathcal{S}_{oi}$ (the Greek letter Delta) aka the ***Node Delta,*** We can use (16) to rewrite the calculation δ $_{oi}$ by calculating (17):

$$\delta_{o_i} = -(target_{o_i} - out_{o_i}) * out_{o_i}(1 - out_{o_i}) \qquad (17)$$

Therefore $\Delta\omega$ calculated by (18):

$$\frac{\partial E_{total}}{\partial \omega_i} = -\delta_{o_i} out_{h_i} \qquad (18)$$

We function the real weight updates in the neural network then we bear the consecutive weight changes into the the hidden layer neurons by (19):

$$\omega_i^+ = \omega_i + \eta \frac{\partial E_{total}}{\partial \omega_i} \qquad (19)$$

where η (eta) is learning rate.

- **Hidden-input Layer updation:** Next, continue the backwards pass by calculating new values for all weights from hidden neurons to inputs layer. It utilizes a same procedure as had for the output layer, yet a bit dissimilarity to represent the way that the yield of each hidden layer neuron adds to the output (and consequently error of numerous output neurons). We realize that OUT hey influences both OUT$_1$ use and *OUT$_2$*, therefore, the $\frac{\partial \mathrm{E}total}{\partial \mathrm{O}\mathbb{UT}hi}$ needs to take into consideration its effect on the both output neurons:

The partial deviation of hidden to input updating might also be seen written by (20) and (21):

$$\frac{\partial E_{total}}{\partial \omega_i} = (\textstyle\sum_0 \delta_0 * \ \omega_{h_0}) * \ out_{h_i}(1 - out_{h_i}) * \ l_i \qquad (20)$$

$$\frac{\partial E_{total}}{\partial \omega_i} = \delta_{h_i} l_i \qquad (21)$$

We can now update weights calculated by (22):

$$\omega_i^+ = \omega_i + \eta \frac{\partial E_{total}}{\partial \omega_i} \qquad (22)$$

Finally, we have updated all of weights and continue passing through the two phases until the desired output occurred. Thus this technique would be very advantaged classification strategy to learning the model in order to detect the malicious activities of mobile ad hoc intruders by monitoring its malicious signatures or misbehaviors.

IV. System Implementation

The proposed system uses NS-2 versions 2.3x, Ubuntu 12.4, C++ and Object-oriented Tool Command Language (OTcl) script for simulating the attacks in MANET and its simulation environment are given in table II. It is utilized for the reproduction of system protocol with various system topologies, it is implicit C++ and gives the manipulation interface through OTcl. Additionally, it is an open-source event-driven simulator designed specifically for researches into computer communication networks. It provides a widespread assistant in conformity with the simulated environment over protocols as TCP, FTP, UDP, HTTP then DSR which makes use of OTcl in imitation of propagation along with configure a network or C++ according to lead simulation, therefore C++ codes need to be compiled and linked after create an executable file. We have to use 50 nodes to form the MANET using ADOV routing protocol. Each of which are all mobile nodes, Then the source node will send the second route request RREQ to all nearest node to reach the destination and also send the route response RREP to all nearest node to reach the source, during the RREQ and RREP the attacker node will be get the data's by exploiting the masquerade scenario, but not transfer to next to node that we called as the fraudulence event, then detect the attacker node relying on the BBP, Additionally we could be able to use the Xgraph screen to evaluate the path between the source and destination, For route selection we use DSR routing protocol. This selects the best and also shortest path between the source and destination.

- ❖ ***Simulated Attack***: In this project the black hole attack and flooding attack are simulated.
- ❖ ***Flooding Attack:*** it concering with the pentratiive phenomenon when a malicious node C separates the illicit approach for RREQ flooding and data flooding, namely called Distributed Denial of Service Attack (DDoS). The malicious node C can launch fake RREQs or data packets by flooding the network with so many requests,

consequently leads to the congestion of the network and reduces the probability of data transmission of the genuine nodes. Eventually leads to lose the potential information transmitted, the flooding attack illustrated in Fig. 3.

TABLE II
NS2 SIMULATION ENVIRONMENT

Sno	Parameters	Value
1	Simulator Duration	100 seconds
2	Topology	1000m*1000m
3	Number of Mobile Nodes	50
4	Transition Range	250m
5	Node Movement Model	Random Waypoint Model
6	Traffic Type	CBR(UDP)
7	Data Payload	512 bytes

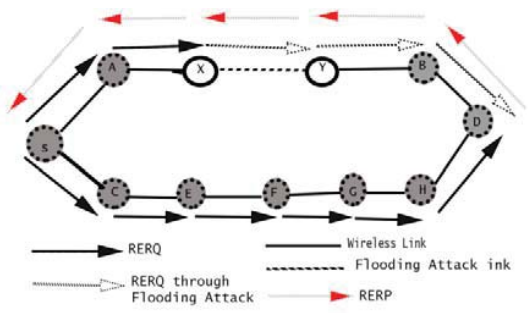

Fig. 3 Flooding Attack

❖ ***Black Hole Attack:*** Even though there are so many senarios about black hole attack but still all of them shares the inspire nature of this cosmological term as the matter of fact where the energy sinks down in dark point over the immerse space, Let us assuming that an node be a malicious node called H, and S be a source node want to send a Route Request to a destination node over the MANETS, this initiates the route discovery process, immediately H response to A by claiming to have the shortest path to the destination and receives the RREQ from A, it will then send a response to A before other nodes. Hence, node A will start to route all data through H ignoring other responses from the neighboring nodes. This node H then will drop the packets upon receiving them. The attack is caused to massive exhaustion of information transmission as illustrated in the Fig. 4.

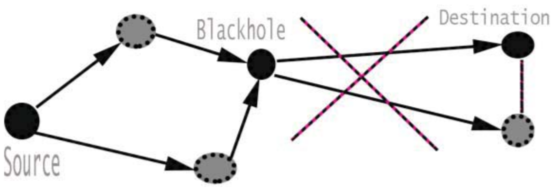

Fig. 4 Black-Hole Attack

Normal profile is accumulated together with the non-existence of attacks. Whereas the attack profile is constructed through simulating the black hole and flooding attacks. Various traffic related features are accumulated [3] as listed in Table III.

TABLE III
TRAFIC RELATED FEATURES

Packet Type	Data, Route Request, Route Reply, Route Error and Hello
Flow Direction	Send, Receive, Forwarded, Dropped

The orginal data set extracted from a traffic table in which it consist of the packet type and the flow type, first one has five components each one of them shows a specfic role in the discovery process like the Route Request RREQ searching the route from Source Node to Destination Node, latter contains four types obviously descripted by its flow direction like the Dropped action means discharging the packets in a specfic node in which those dropped packets have no effect any more, hence the raw data set passes through the preprocess to formulate these data set into an appropriate form readable by the machine learning, as well the detailed features capture involving the actual encoding carrys out within OTcl, likewise Hello pakets encod by the OTcl script as *NBHelloSend and NBHelloRecv*. Therefore, totally (5*4-2) =18 features are considered for intrusion detection [3] as shown in Table IV.

TABLE IV
COLLECTED FEATURES

Data Packets	*NBDataSend, NBDataRecv, NBDataDrop, NBDataFwd*
RREQ Packets	*NBRREQSend, NBRREQRecv, NBRREQDrop, NBRREQFwd*
RREP Packets	*NBRREPSend, NBRREPRecv, NBRREPDrop, NBRREPFwd*
RERR Packets	*NBRERRSend, NBRERRRecv, NBRERRDrop, NBRERRFwd*
Hello Packets	*NBHelloSend, NBHelloRecv*

❖ ***Training Data:*** so far we have explained the shaffling process for gathering the raw data set as well as the preprocessed stage, hence the data set is ready for computerizing and presenting to the machine learning, the synthetic approach tunes up with normal and labeled profiles, namely two classes, normal and abnormal. This data set is used a stochastic feature selection and classifier, each feature is represented by a packet type and flow direction.

❖ ***Testing and validation data :*** The accumulated trained data will be separated as validation data and Test data, Test data are accumulated by using the simulated black hole and flooding attacks through varying the attackers. It will be put forward as like input to the classifier in order to identify whether the particular event is an attack or normal. Once the data is accumulated it will be given for the feature selection module. The accumulated features transformed in accordance with computer readable form, then the stochastic classifier is constructed. Constructed classifier is trained with training data and validated through.

V. PROBLEM IDENTIFICATION AND SOLUTION

Classification and clustering of Intrusion Detection for MANET has been defined as the conversion of Traffic Table minutiae into machine readable codes. Though they collect data set from distributed nodes, they analyze them centrally (identifying the attack signature of malicious attacks and benign TT). Several problems with the deployment of current distributed IDSs, are the high rate of false positive rates, insufficient protection against compromised nodes, etc. So we have to enhance the distributed Intrusion detection with a layer of active, vigilant, monitoring defense mechanism [7].

In this approach we view our agents as autonomous, mobile, proactive and cooperative entities. Agents are equipped with BP in which responsible for alerting system/network administrator with any possible attack-related activities.

Our approach consists of two types of agents: Collection Agent and Monitoring Agent.

Collecting Agent: It gets the traffic features of the neighboring nodes and creates a Traffic Table (TT) such as:

- Packet type
- Flow direction
- Sampling periods
- Statistic measures.

Monitoring Agent: It is not present in all the nodes but it is present only in a subset of nodes. By using the TT information feeding to BP to classify threats among simulated networks, it monitors its neighbor nodes behavior. When an abnormal behavior or any deviation in the traffic pattern of the nodes occurs, then it raises an alarm for intrusion.

The kernel of BP implementation supportive of Monitoring Agent Work is compromised of 5 experiments as shown in Table V as well as Figs. 5 and 6. We need to used the following symbols in the table of BP implementation.

- Experiment Number.
- Number of Iterations.
- Least Fitness (Mean Square Error).
- Accuracy Level of the Learning.
- {ALg}: Algorithm.

The multi-layer perceptrons established for relevant purposes are trained with BP algorithm, respectively, through the selected training set.

In order to establish a fair start-up state for BP-based perceptrons, the training processes of BP-based perceptrons always start with random solution parametric gradually attempting to get at optimal result, for BP-based perceptrons, such "evolution time" directly equals the times of its updated iterations. The implementation is done for 5 experiments for Data Set divided to training set and testing set by 70-30 respectively. It's obvious that experiment 5 is much better than others due to the minimal error correction and the Accuracy Level.

TABLE V
BP EXPERMENTAL RESULTS

EXP	ALg	NoI	MSE	AL %
1	BP	142	0.101776	53.75%
2	BP	159	0.090614	60.2%
3	BP	172	0.070087	66.2%
4	BP	327	0.031445	95.1%
5	BP	625	0.000298	98.6%

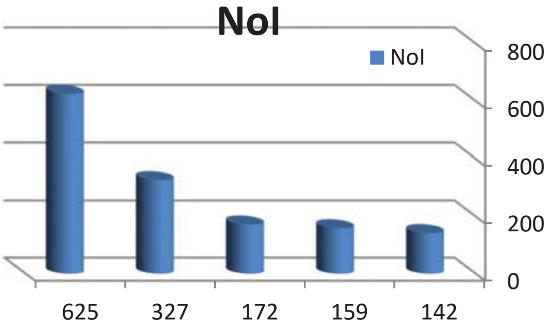

Fig. 5 Skew Graph for no of iteration with the respect of MSE

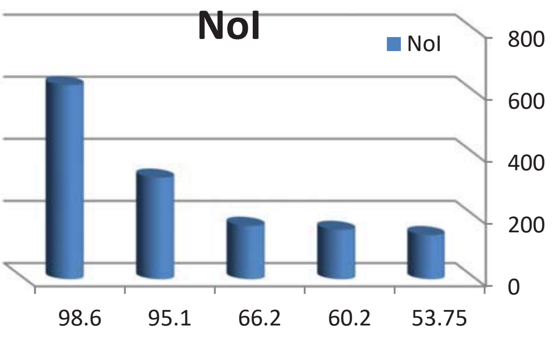

Fig. 6 Skew Graph for the number of iteration with respect to percentage of detection rate

VI. Performance Metrics

Packet delivery ratio (PDR): PDR defines the ratio over the number of packets acquired through the destination mobile nodeto the number of packets despatched through the source mobile node, as it is illustrated by Fig. 7.

$$\text{PDR}=\frac{\sum(aquisted\ packets\ at\ destination\ node)}{\sum(send\ packets\ by\ the\ intial\ node)}$$

Throughput(TP): It is described as like the average rate of effectively acquired message is delivery on a communication channel. All malicious mobile nodes to send out bogus misbehavior report to the source node on every occasion it is possible. This type of scenario placing is designed in accordance with check the IDS's performance under the bogus misbehavior report, in which is illustrated by Fig. 8.

Average End to End Delay (AED): The common end-to-end delay upon the whole efficiently acquired packets at the destination. It is calculated for every data packet be subtracting the sending time of the packet from the acquired time at final destination. Then the average represents the AED, it is illustrated by Fig. 9.

$$\text{AED}=\frac{\sum_1^N(time\ recieved-time\ sent)}{N}$$

VII. Conclusion

In this article, the anomaly detection approach is utilized for

MANET to recognize the intrusions. This approach utilizes the network layer data to represent the behavior of mobile nodes. The audit data is accumulated from all the mobile nodes under various scenarios to group the incoming events.

During the collecting phase it was involving into two approaches in order to simulate the real life senario by creating the normal profile under the absence of attacks 80-70% and the attack profile is created by simulating attacks such as black hole and flooding attacks 20-30%.

After the computerized form and feature selection technique is applied, respectivly, since the BP algorithm is used. The selected features are used during the discovery process, the attack profile is in contrast to the normal profile. If there are some deviation from the normal behavior, then the event is labeled as like an attack.

Finally, the overall performance of BP is evaluated primarily based on discovery rate (AL Percentage), number of iterations (NoI), and mean square error (MSE).

This approach makes use of early method, therefore., it will give practical enhancement in performance.

In future, the feature selection is evaluated with various exclusive classifiers, it will be combined to produce a desired result for different type of attacks.

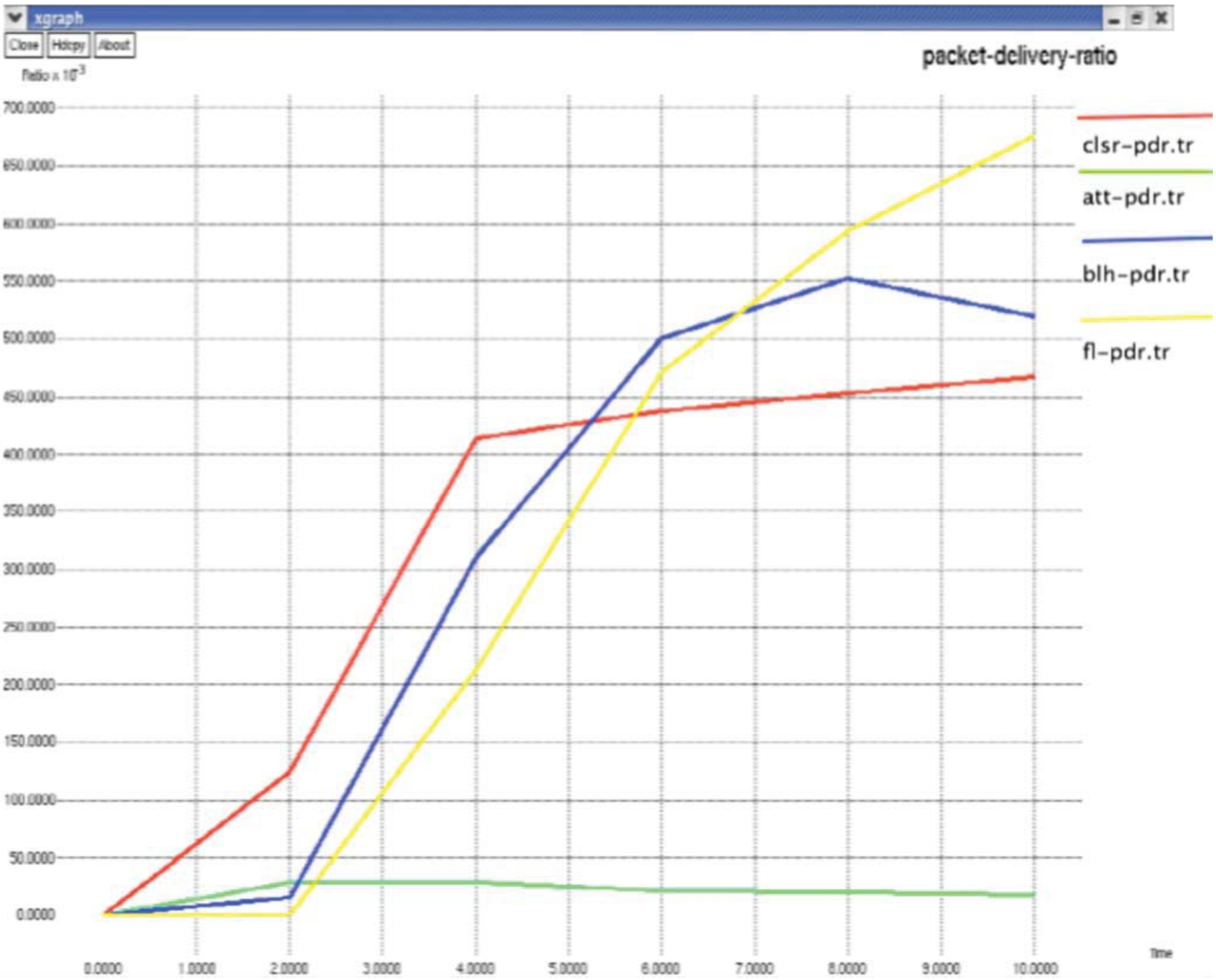

Fig. 7 Packet Delivery Ratio

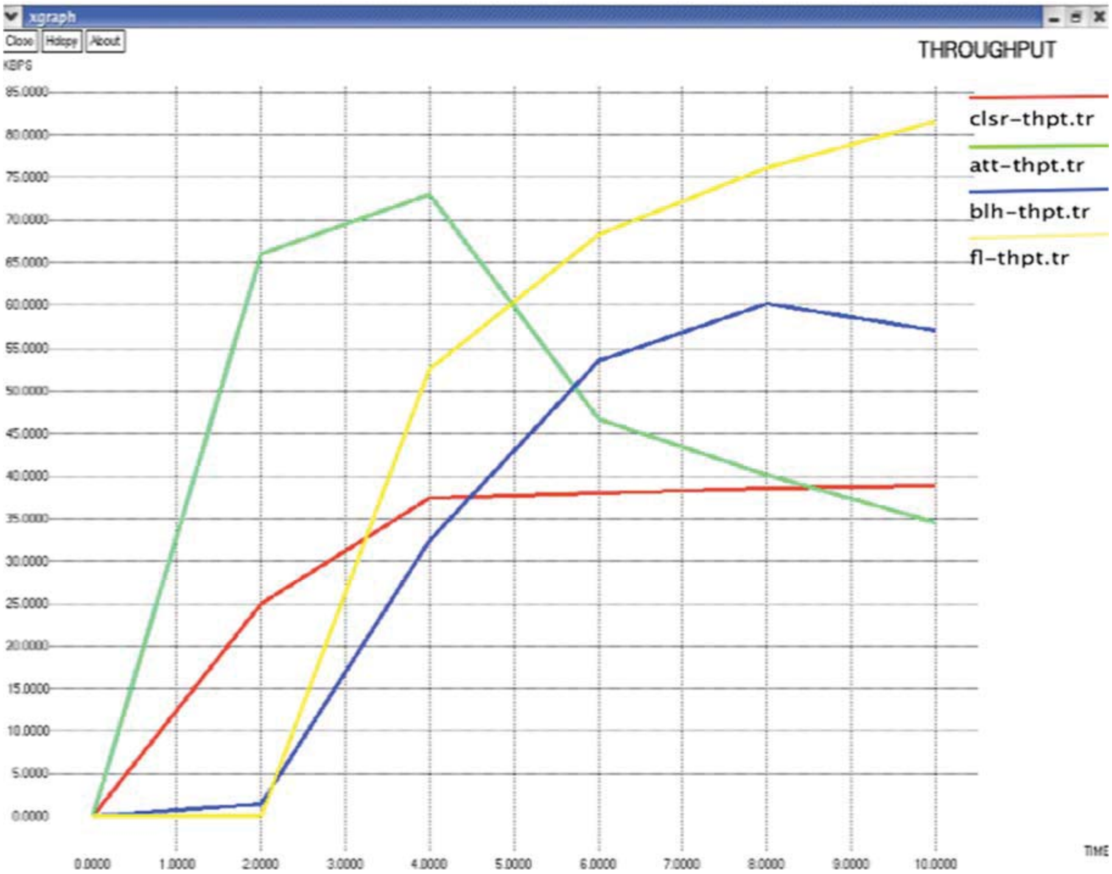

Fig. 8 ThroughPut

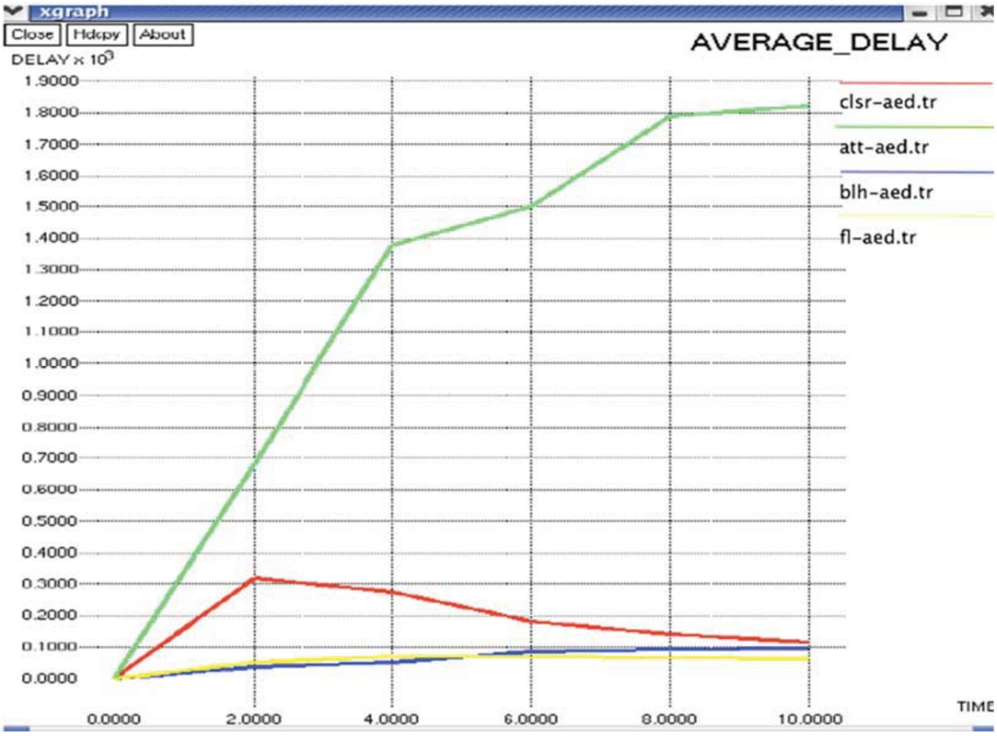

Fig. 9 Average Delay

REFERENCES

[1] Tiranuch. Anantvalee. and Jie. Wu, “*A Survey on Intrusion Detection in Mobile Ad Hoc Networks, Wireless/Mobile Network Security,*” Springer, 2006.

[2] Ovais. Ahmad. Khan, "*A Survey of Secure Routing Techniques for* MANET*",*http://ovais.khan.tripod.com/papers/Secure_Routing_MANET.pdf, 2010.

[3] Ernesto. Jiménez. Caballero,*"Vulnerabilities of Intrusion Detection Systems in Mobile Ad-hoc Networks -The routing problem},"* http://www.tml.tkk.fi/Publications/C/22/papers/Jimenez_final.pdf, 2006.

[4] Yanchao. Zhang, Wenjing. Louy, Wei. Liu and Yuguang. Fang, “*A Secure Incentive Protocol for Mobile Ad Hoc Networks}, Wireless Networks,*” springer 2006.

[5] Satria. Mandala, Md. Asri. Ngadi and A. Hanan. Abdullah, “*A Survey on MANET Intrusion Detection, International Journal of Computer Science and Security,*” August 2007.

[6] Mirza. Cilimkovic, “*Neural Networks and Back Propagation Algorithm, Institute of Technology Blanchardstown*,” Ireland, 2014.

[7] Arjita. Ghosh and Sandip. Sen, "*Agent-Based Distributed Intrusion Alert System*," Springer, 2005.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic Minority Over - Sampling Technique," Journal of Artificial Intelligent Research, Vol. 16, pp. 321-357, 2002. Cited on page 126.

[9] S. Madhavi and Tai. Hoon Kim, "*An Intrusion Detection System in Mobile Adhoc Networks,” International* Journal of Security and Its Applications, Vol. 2, No.3, July, 2008.

[10] Angelo. Rossi and Samuel. Pierre, “*Collusion-resistant reputation-based intrusion detection system for MANETs"* International Journal of Computer Science and Network Security, VOL.9 No.11, November 2009.

[11] O. Depren, M. Topallar, E. Anarim, and M. k. Ciliz, “*An Intelligent intrusion detection system (IDS) for anomaly and misuse detection in computer networks,*" Expert Syst. Appl, vol. 29, no. 4, pp. 713-722, 2005. Cited on pages 47 and 123.

[12] Bo. Sun, Kui. Wu and Udo. W. Pooch, “*Zone-Based Intrusion Detection for Mobile Ad Hoc Networks*," http://webhome.cs.uvic.ca/~wkui/research/IDS.pdf, 2010,

[13] S. T. Powers and J. He, " *A hybrid artificial immune system and self organizing map for network intrusion detection,” Information* Sciences, vol. 178, no. 15, pp. 3024-3042, 2008. Cited on page 48.

[14] M. Hall and G. Holmes, *"Benchmarking attribute selection techniques for discrete class data mining," IEEE Transaction on Knowledge and Data Engineering, vol. 15, no.6, pp.1437-1447, 2003. Cited on page 88 and 93.*