Multi-class Intrusion Detection System for MANETs

Konagala Pavani and Auvula Damodaram

***Abstract*—As MANETs change their topology dynamically, intrusion detection in these networks is a challenging task. These networks are more liable to the security attacks because of the properties such as node mobility, lack of concentration points where aggregated traffic can be analyzed, intermittent wireless communications and limited band width. We present a multi-class intrusion detection system that addresses these challenges. In this paper we propose a neural network method based on MLP (multi-layer perceptron) for detecting normal and attacked behavior of the system. The method was tested for Black Hole and Gray Hole attacks. We have implemented these attacks using NS2 simulator. The method successfully detected these attacks. We compared the results with KNN (K-Nearest Neighborhood) which is another classifier used for classification. Finally, Re sampling methods were also applied to assess the performance of classifier. This paper presents a graphical representation of the results.**

***Index Terms*—Intrusion detection system, Black Hole attack, Gray Hole attack, multi-layer perceptron, K-nearest neighborhood.**

I. Introduction

A mobile Ad hoc network (MANET) is a collection of mobile nodes which are connected to each other without any fixed or pre defined infrastructure. MANETs have gained increased popularity because of low cost equipment. In MANETs, mobile nodes communicate with each other without any centralized administration. MANETs are used in various applications such as military & civilian areas, rescue operations, personal area, and campus as they provide reliable routing services without any infrastructure. Flexibility and adaptability of MANETs have made it more susceptible to the attacks. So there is a need of strong security system. MANETs doesn't have the facility of installing firewalls at routers and switches as in wired networks. So, to enhance the security level of MANETs Intrusion detection systems can be used. Intrusion detection system is a software or hardware system which monitors the intrusions [1] occurring in a computer system or in a network IDS detects the behavior of users that conflict the intended use of the computer system. Misuse and Anomaly detection are two general approaches for intrusion detection [2]. Misuse detection or Signature based detection generates an alarm when a known attack signature is matched. Anomaly detection identifies activities that deviate from the normal behavior of the monitored system and thus has the potential to detect an attack. In fact, anomaly detection can be regarded as a classification problem which classifies normal or attacked behavior. This paper elaborates the ongoing research on Anomaly Detection systems for detecting network layer attacks in MANET. AODV routing protocol is used to analyze the vulnerabilities in network layer. A neural network method based MLP is proposed as a solution for detecting vulnerabilities in the network.

The rest of this paper is organized as follows. In next section we review some related work. In Section III we described Black Hole and Gray Hole attacks. Section IV describes the architecture of proposed system. Section V explains multi-class classification.MLP & KNN modeling techniques are described in the sections VI. Section VII describes experiments. The results are shown in Section VIII. Finally, we summarize our work in Section IX.

Manuscript received October 20, 2014; revised March 20, 2015.

The authors are with Vaagdevi College of Engineering/CSE, Warangal, India (e-mail: bandaripavani@gmail.com).

II. Related Work

In recent years, researchers have proposed different methods to improve security level of MANET. Y. Zhang *et al*. [2], proposed an architecture for a distributed and cooperative intrusion detection system based on statistical anomaly detection techniques. Rizivi *et al*. used watchdog and pathrater approaches to show the effects of routing misbehavior in DSR protocol [3]. Loo [4] used a clustering algorithm for detecting routing attacks in sensor networks. Huang [5] proposed an anomaly detection scheme that gives solution for routing anomalies. Rakesh Shrestha *et al*. [6] proposed a novel technique for cross layer intrusion detection system for MANETs. In [7] Zahra Moradi *et al*. proposed a neural network technique for detecting denial of service attacks. Daniel [8] has proposed multi-class cancer classification using gene expression profiling and probabilistic neural networks. Sujatha *et al*. proposed [9] genetic algorithm based IDS for MANETs.

III. Attack in MANET

Spite the fact of popularity of MANET, these networks are very much exposed to attacks. In this paper we have implemented Black Hole and Gray Hole attacks using AODV routing protocol [10] to evaluate the performance of proposed multi-class intrusion detection system.

A. Black Hole Attack

Black Hole attack [11] is an active type of attack which occurs in network layer. It is also known as packet drop attack. In this type of attack, the attacker displays least hop count and maximum sequence number to the requested source node. Sequence number work as a time stamp and let the nodes to determine how fresh their information on the other node is. Source node then routes the packets to the destination through this compromised node. The compromised node drops all the

data or control packets passing through it without forwarding them to the next node on the routing path. This type of attack results in very low packet delivery ratio.

B. Gray Hole Attack

The AODV routing protocol is vulnerable to Gray Hole attack [11]. It is also an active type of attack which occurs in network layer. It also leads to dropping of messages. Initially the attacker node behaves normally and replays true RREP messages to the source node. So the source node starts communicating with the destination through this path. But later Gray Hole node drops the packets without forwarding them to the next node on the routing path which leads to loss of information.

IV. Proposed System Architecture

The architecture of proposed system using multi-layer perceptron is shown in the Fig. 1. The components of the proposed system includes MANET network which is implemented using NS2 with different node combinations like 20, 21 through 80.This network includes attack and normal nodes. In Data Collection module the data is collected by analyzing the log files generated by NS2 simulation. This data consists of network information such as routing updates and attack information which is collected from physical, network and MAC layers. The network related parameters like Route Request Packets (RREQ), Route Reply Packets (RREP), variance (variance of the delay), sent (number of packets sent), received (number of packets received by receiver), Delay (delay in sending the packet), Drop (number of packets dropped during transmission), Packet Delivery Ratio (number of packets delivered in a second) are being collected by feature selection module. Then the classification models are applied and performance of these models is analyzed using different metrics which will be explained in the forthcoming sections.

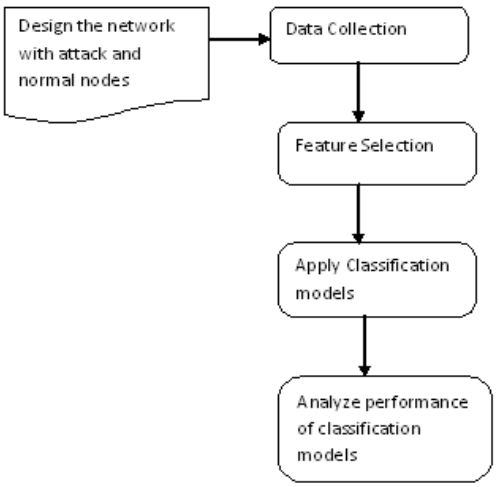

Fig. 1. Architecture of proposed system.

V. Multi-class Classification

Multi-class classification is a process of categorizing the instances into more than two classes. The goal of this paper is to build a model which categorizes each instance in dataset into one of the three classes (normal/Gray Hole attack/Black Hole attack). In this paper we used MLP & KNN models for multi-class classification.

A. MLP Modeling

Artificial neural network (ANN) [12] is one of the artificial intelligence methods, which is inspired by human nervous system. It consists of a large number of highly interconnected processing elements called as neurons. Neurons work with each other to solve specific problem. ANNs are utilized to detect the node under attacks. It is designed with summing element followed by an activation function. In this, output of each neuron is given as input to the neurons in the next layer. A multi-layer perceptron (MLP) is a feed forward ANN model which maps set of input data on to a set of output data. The model of each neuron in the network includes a non linear activation function. The network contains one or more layers of hidden neurons which are not part of input or output of the network. These hidden neurons enable the network to learn complex tasks.MLP can classify data which is not linearly separable. MLP uses a supervised learning technique called as back propagation algorithm for training the network.

Architecture of MLP consists of 3 types of layers. They are an input layer, one or more hidden layer, and an output layer. Each node in one layer connects with certain weight to every node in the next layer. The network is trained by changing connection weights. We have used TANAGRA software to model the network. MLP is trained by applying the audit data as input. The designed network model consists of 9 input neurons (selected features), 10 hidden layer neurons and three output layer neurons. The output layer consist of three neurons i.e. Black Hole attack/Gray Hole attack//normal. Fig. 2 illustrates the flow diagram of MLP modeling.

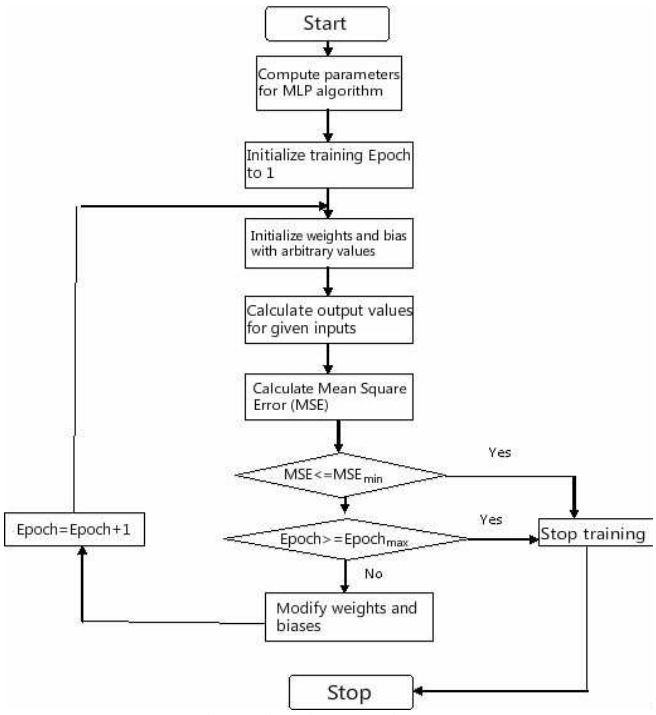

Fig. 2. Flowchart to train MLP.

The algorithm of the proposed technique is as follows.

1) Compute the parameters for MLP algorithm. This step includes getting the input parameters like packet drop,

delay, node id etc. These parameters are evaluated in Feature selection module shown in Fig. 1.

2) Initialization of MSE, EPOCH: Mean Square Error is the error rate calculated during learning/training process. EPOCH is the number of iterations for training. These parameters are used to stop the learning parameters. Initialize minimum MSE and maximum EPOCH.
3) Initialize bias and synaptic weights with random values.
4) Load the input pattern and Calculate output values.
5) Calculate MSE.
6) If ((MSE <= MSEmin) or (EPOCH >= EPOCHmax)) then (stop training) else:
   - Modify weights and bias.
   - Increment EPOCH by 1.
   - Repeat steps 4 to 6.
7) Stop.

B. KNN Modeling

K-nearest neighbor algorithm [13] is the simplest of all machine learning algorithms. It is used to find a group of K objects in the training set that are closest to test object. Similar to other classification techniques it also has training and testing phase. The training process of KNN is simple that other models. Because during training it simply loads sample set into data base. To test the new sample (S) it follows the following steps.

- Calculate distances of all training samples to test sample S.
- Find the K instances in training sample set which are closest to S.
- These K instances then vote to determine the class of S.

VI. Experiments

A. Simulation Environment for MANET

MANET environment is simulated using Network Simulator NS-2 [14], [15] with parameters listed in Table I. As it is noticed, in simulated network, AODV routing protocol is used with Mac layer IEEE 802.11. The Black Hole and Gray Hole nodes are introduces in the network by making appropriate changes into routing protocol. By analyzing log files generated from NS2 simulation, parameters are extracted. Then sample set is constructed using these parameters.

TABLE I: Simulation Parameters

Parameter	Definition
Protocol	AODV
Mac Layer	IEEE 802.11
Simulation time	500 s
Connection time	450 s
Node Placement	Random
Simulation Area	1000×1000
Size of data packet	512
Traffic Sources	CBR
Number of nodes	20 to100
Version NS2	NS-2.29 (under windows, cygwin)
Data rate	10 bits

B. Modeling for Detection of Node under Attacks

The present research deals with anomaly based IDS which is characterized into training and testing phase. The training phase contains the data with normal and abnormal data. However testing is done to test how much data set is matching with the model. We have used TANAGRA software to model the network. Modeling is done in two phases namely training & testing to detect node under Black and Gray Hole attack. To model MLP & KNN, 64% of data are randomly selected for training and remaining data are used for testing.

1) MLP modeling

For MLP, the training was performed for 100 times (epochs) and MSE_{min} is set to 0.01. Fig. 3 shows the graph between mean square error and training epochs. The training error (MSE) is reduced during learning process to an outstanding value. As error is reduced to zero; it may give good classification results. Then testing is done to evaluate the performance of the model.

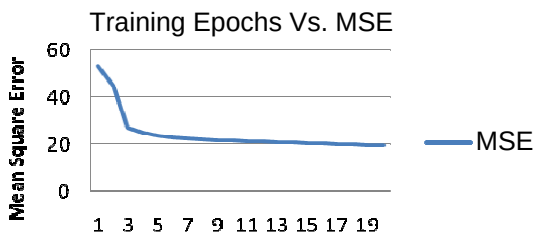

Fig. 3. Mean square error drop during learning process.

2) KNN modeling

To use KNN modeling the parameters used are distance and neighborhood size. In our experiments we used Euclidean distance method to measures the distance between the samples and neighborhood size as 5.

VII. Experimental Results

A series of experiments were conducted to evaluate the performance of classification algorithms. The most important criteria to evaluate the performance of classifier is the ability to predict correctly. The preferred indicators are error rate and accuracy. The most widely used other evaluation parameters for testing the performance of classifier were sensitivity, specificity, recall and ROC Curves [16]. The classification model is learned (trained) on finite training dataset. Then the classifier has to be tested on different test data (unseen data). In this paper we used random sampling method to divide the sample set into training and testing i.e. 64% of sample set is used for training and remaining data is used for testing.

A. Error Rate

The instance's class which is predicted incorrectly is called as an error. The error rate is the proportion of the errors made over the available sample set. Error rate is used to measures the overall performance of classifier. In our experiments we have calculated testing error rates for both types' methods. The error rates are been shown as a graph in Fig. 4. The lowest testing error rate is recorded for MLP when compared to KNN.

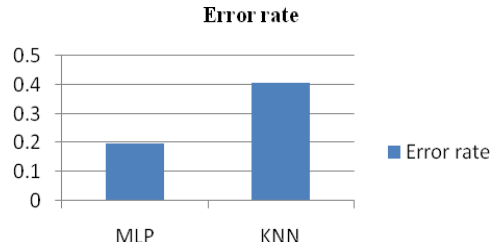

Fig. 4. Test error rate.

B. Accuracy

Accuracy refers to the percentage of the correctly classified instances over the whole set of instances. Mathematically it is defined as follows:

$$\text{Accuracy} = \text{No of correctly classified instances}/\text{Total no of instances} \times 100$$

Fig. 5 shows the accuracy percentages resulted after applying MLP & KNN algorithms. The MLP classifier has achieved highest accuracy values when compared to KNN.

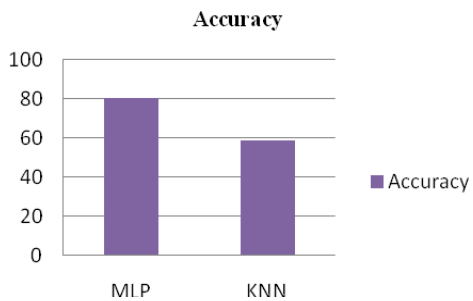

Fig. 5. Test accuracy.

C. Confusion Matrix

There are other measures to evaluate the performance of classifier. These are derived from confusion matrix. This matrix is also called as contingency table. Confusion matrix gives full picture of the errors made by classification model. i.e. it shows the predictions made by a classification model. The rows correspond to the known class data i.e. labels in the data and columns correspond to predicted class labels. The diagonal elements show the number of correct classifications TP (true positives) made for each class and the off-diagonal elements show the errors made for each class. The notations used in abstract confusion matrix are shown in the Table II.

TABLE II: STRUCTURE OF CONFUSION MATRIX

	A	B	C
A	TP_A	e_{AB}	e_{AC}
B	e_{BA}	TP_B	e_{BC}
C	e_{CA}	e_{CB}	TP_C

Performance measures calculated from this matrix are

- Precision:

It is the measure of the accuracy provided that a specific class has been predicted. It is defined as follows:

$$\text{Precision} = TP/(TP+FP)$$

where TP is true positive predictions and FP (false positives) is false positive predictions for a class. If precision is more, then it is assumed that the classifier is good at classifying the records as it is generating less false positives.

- Recall:

It is also called as sensitivity or true positive rate (TPR). It is a measure of accuracy of a classification model to select instances of a certain class from sample set. It is defined as follows:

$$\text{Recall}=\text{Sensitivity}=TPR=TP/(TP+FN)$$

If recall is more, then the classifier is good at classification and produces less false negatives (FN).

- Specificity:

Specificity is also known as True Negative Rate (TNR) which is defined as follows:

$$\text{Specificity}=TNR = TN/(TN+FP) \times 100$$

If specificity is high, then the classifier is good at classifying records of negative class i.e. it produces few false positives. Fig. 6 and Fig. 7 represent the graphical representation of the above mentioned metrics. From the experiments it is observed that the MLP classifier has achieved high sensitivity, specificity and precession values when compared to KNN method for both types of attacks.

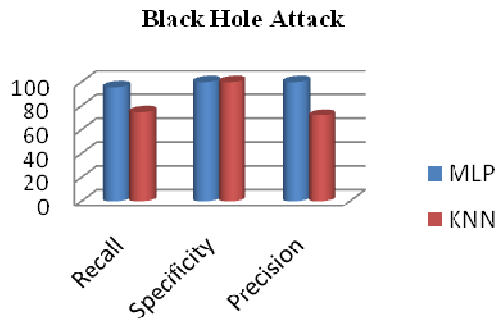

Fig. 6. Comparison of metrics for Black Hole attack.

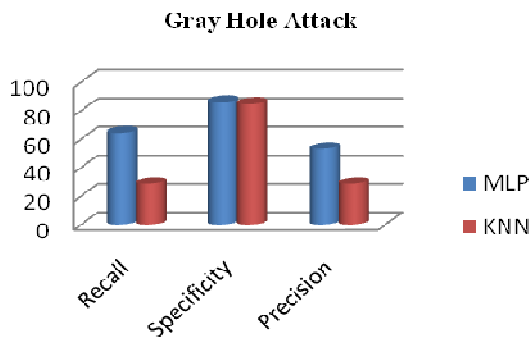

Fig. 7. Comparison of metrics for Gray Hole attack.

- ROC curves:

ROC graphs are the best tools for assessing performance of classifiers. These are the two dimensional graphs constructed by plotting FPR (false positive rate) Vs TPR (true positive rate). Fig. 8 represents the ROC Curve of random classifier. The diagonal line from (0, 0) to (1, 1) shows random classifier performance. The classification model mapped onto this diagonal line produces same number of false positives & true positives. ROC graphs are symmetric along random

performance line. Classifiers that fall below the region of random performance line give worst performance than random classifier. The point (0, 1) on the top left corner shows perfect classification i.e. 100% TPR & 0% FPR.

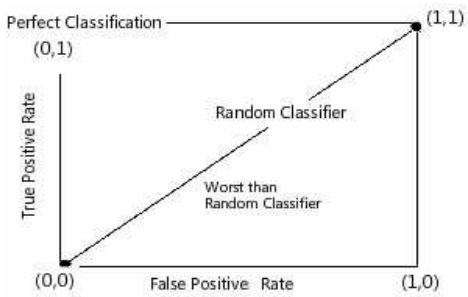

Fig. 8. Performance of random classifier.

The point (0, 0) on bottom of the random classifier line, will not produce any false positives or true positives. However the point (1, 1) on the right top of random classifier line produces large number of true positives and false positives. AUC (area under curve) is a variable which gives the area below the ROC curve. If AUC is high, then the classifier is good at classifying records. Fig. 9 and Fig. 10 represent the ROC Curves for both types of attacks using MLP. Fig. 11 and Fig. 12 represent the ROC Curves for both types of attacks using KNN. However Fig. 13 shows comparison of AUC Values for both methods. The MLP classifier has achieved more AUC values ranging from 88% to 99%. However KNN has achieved lowest AUC values ranging from 64% to 89%.

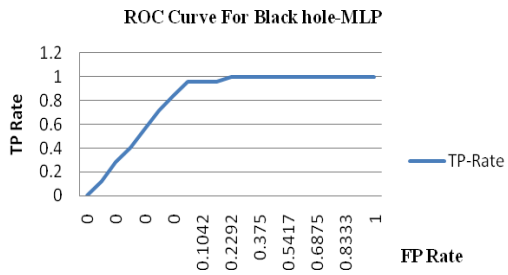

Fig. 9. ROC Curve for Black Hole attack using MLP.

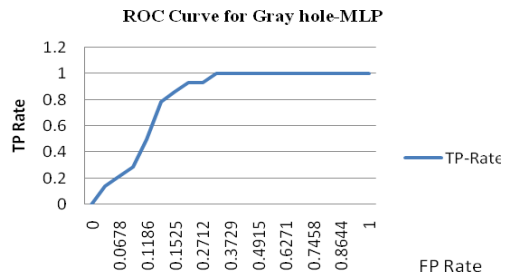

Fig. 10. ROC Curve for Gray Hole attack using MLP.

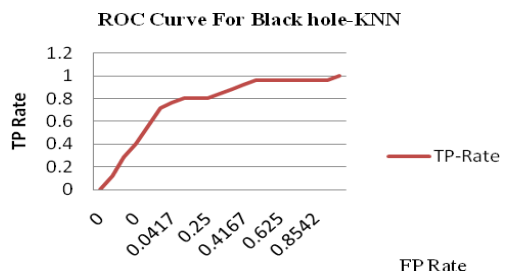

Fig. 11. ROC Curve for Black Hole attack using KNN.

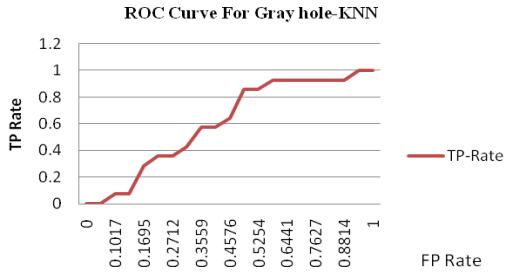

Fig. 12. ROC Curve for Gray Hole attack using KNN.

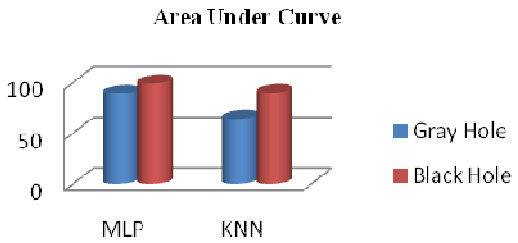

Fig. 13. Comparison of AUC values.

D. Re Sampling Methods

Learning the training data too precisely usually leads to poor classification results on test data. So a classifier has to have the ability to generalize. The available sample dataset has to be used for training and testing. If training data is more it gives better generalization. But if test data is more it gives better estimation of classification of errors. Also when we deal with a small size sample set, dedicating a part of dataset for testing penalizes the learning phase, and the error estimation is unreliable because the test sample size is small. Thus in small sample context, it is preferable to implement re-sampling approaches for error rate estimation. In this paper we used cross validation and leave one out methods to evaluate the error rate of the classifier on the whole sample set.

1) Cross validation

Cross validation is an estimation technique used to assess the performance of a classification model. The main goal of cross validation is to define a dataset to test the model in training phase so that the problems like over fitting are reduced. In K-fold cross validation, the given sample set is randomly divided into K equal size sub samples. From these sub samples a single sub sample is used to test the performance of classification algorithm, and remaining K-1 subsamples are used for training. This process then repeated for K times. Then the results are averaged to produce single estimation. The size of k depends on the size of sample set. In this paper we used 5 fold cross validation. The cross validation error rates obtained from the experiments are represented as a graph in the Fig. 14. The MLP classifier has achieved less error rates when compared to KNN.

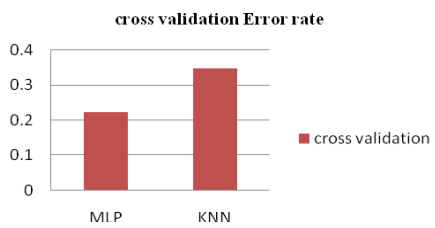

Fig. 14. Cross validation error rate.

2) Leave one out method

In this method if the sample set consists of N instances, then it performs N rounds. In each round it uses N-1 instances for training and remaining instances for testing. This process is repeated for N times and the error rate is evaluated as the average of error rates obtained on test instances. These error rates obtained from our experiments are represented in the Fig. 15. The KNN method has achieved more error rates when compared to MLP.

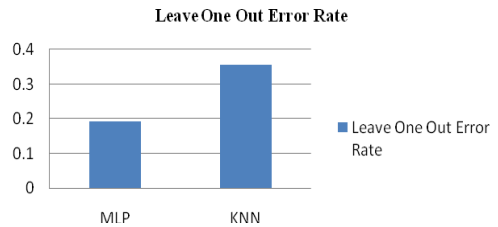

Fig. 15. Leave one out method error rate.

VIII. Conclusion

In this paper we have implemented Black Hole and Gray Hole attacks for AODV routing protocol using NS2.Then a multi-class intrusion detection system is implemented using MLP. Also this research has conducted a series of experiments to assess the performance of the proposed method with other classification method called KNN. Re sampling methods were also applied to estimate the true generalization error rate of the classifier. Finally we conclude that MLP is the best classification algorithm for detecting Black Hole and Gray Hole attacks in MANETs.

References

[1] K. Bounpadith, H. Nakayama, Y. Nemoto, and A. Jamalpour, "A survey of routing attacks in mobile Ad-hoc networks," *IEEE Wireless Communication*, vol. 14, no. 5, pp. 85-91, 2007.

[2] Y. Zhang, W. Lee, and Y. A. Huang, "Intrusion detection techniques for mobile wireless networks," *ACM J. Wireless Networks*, pp. 545-556, 2003.

[3] S. S. Rizvi, S. Poudyal, V. Edla, and R. Nepal, "A novel approach for creating trust to reduce malicious behavior in MANET," in *Proc. the ACM Conference on Emerging Network Experiment and Technology*, 2007.

[4] C. Loo, M. Ng, C. Leckie, and M. Palaniswami, "Intrusion detection for routing attacks in sensor networks," *International Journal of Distributed Sensor Network*s, pp. 313-332, 2006.

[5] Y. Huang, W. Fan, W. Lee, and P. S. Yu, "Cross-feature analysis for detecting ad-hoc routing anomalies," in *Proc. the 23rd International Conference on Distributed Computing Systems (ICDCS) Providence*, 2003, pp. 478-487.

[6] R. Shrestha, K.-H. Han, D.-Y. Choi, and S.-J. Han, "A novel cross layer intrusion detection system in MANE," in *Proc. the 24th IEEE International Conference on Advanced Information Networking and Application*s, 2000.

[7] Zahra and M. Teshnehlab, "Implementation of neural networks for intrusion detection in MANET," in *Proc. IEEE ICETECT*, 2011.

[8] D. P. Berrar, C. S. Downes, and W. Dubitzky, "Multi-class classification using gene expression profiling and probabilistic neural networks," in *Proc. the 8th Pacific Symposium on Biocomputing*, 2003.

[9] K. S. Sujatha and R. S Bhuvaneshwaran, "Design of genetic algorithm based IDS for MANET," in *Proc. the IEEE Conference ICRTIT*, 2012.

[10] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999.

[11] K. Pavani and A. Damodaram, "Performance of mobile Ad-hoc networks in presence of attacks," in *Proc. the 3rd International Conference on Information Security and Artificial Intelligence*, 2012, vol. 56.

[12] A. Bivens and C. Palagir, "Network based intrusion detection using neural network," in *Proc. the Intelligent Engineering Systems through Artificial Neural Networks*, 2002.

[13] O. Sutton, *Introduction to K-nearest Neighbor Classification and Condensed Nearest Neighbor Data Reduction*, February 2012.

[14] NS by Example. (May 14, 2006). [Online]. Available: http://nile.wpi.edu/NS/overview.html

[15] K. Fall and Vardhan. (2000). The NS Manual. [Online]. Available: http://www.isi.edu.nsnam/ns/

[16] F. Zulkhairi and Shaidah, "Dynamic intrusion detection method for mobile Ad-hoc network using CPDOD algorithm," *IJCA Special Issue on Mobile Ad-hoc Networks*, 2010.

Pavani Konagala is an associate professor at Vaagdevi College of Engineering (Autonomous), Bollikunta, Warangal. She had completed her B. Tech degree from Kakatiya Institute of Technological Sciences, Warangal and the M. Tech degree from Jawaharlal Nehru Technological University, Hyderabad. She has 8 years of experience and has performed as a project co-coordinator for technical symposium and guided B. Tech and M. Tech students for their dissertation projects. She is a prime member of various committees in the department for organizing events in the college. She is presently pursuing her PhD degree from JNTU Hyderabad and had presented 4 papers in international conferences and she had published 2 papers in international journals.

Avula Damodaram was a recipient of distinguished academician awarded by Pentagram Research Centre, India, in January 2010. Dr. Damodaram has more than two decades of dedicated service in the Department of Computer Science & Engineering and performed distinguished services to the University as a professor, the head of the Department, the vice principal and director of UGC-Academic Staff College, the director of School of Continuing & Distance Education and the director of University Academic Audit Cell. He has successfully guided 10 Ph.D. scholars and currently guiding 9 scholars for Ph.D. program. Dr. Damodaram is on the editorial board of 2 international journals and a number of course materials. Dr. Damodaram successfully executed an AICTE research project at a cost of 7 Lakhs. He has been a UGC nominee for a number of expert and advisory committees of various Indian Universities. Dr. Damodaram has published 45 well researched papers in national and international journals. He has also presented 59 papers at different national and international conferences in United States of America, Austria and the United Kingdom etc.