JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

ARCHITECTURE FOR COMBINING INTRUSION DETECTION SYSTEMS TO FIND INTRUSIONS

[1]Dr. Riyad.A.M, [2]Dr. R.L. Raheemaa Khan

[1]Assistant Professor, [2]Assistant Professor
[1]Department of Computer Science, EMEA College of Arts and Science, Kondotty, Malappuram, Kerala, India
[2]Department of Computer Science, Karpagam Academy of Higher Education, Coimbatore, Tamilnadu, India

Abstract : In this era of modern computing and internet world, data and knowledge sharing is done through various networks all around the world. As sharing increases, the attack through the networks also increases simultaneously. There are various techniques in which intrusion detection systems (IDS) plays vital role. Various types of IDS are available which use various techniques. Each IDS has its own merits and demerits. So, in order to utilize the advantages of various IDSs, different IDSs with different techniques are combined to make the final decision. Majority voting is done for making final decision. Pre processed NSL KDD data set is used for training and testing purposes.

IndexTerms **- Intrusion detection system, multiple IDS, ANN, K-NN, OneR, J48, NSL-KDD, feature selection**

I. INTRODUCTION

An intrusion detection system (IDS) is a device or software application that monitors network or system activities for malicious activities and produces reports to an authority. If finding the intrusion is challenging, avoiding a normal activity misjudged as intrusion will be more challenging.

Intrusion detection is the act of detecting the intrusions through various techniques. Intrusions can be found by tracing the anomalous network activities. Different methods are devised to identify intrusions. We can broadly classify them into misuse detection and anomaly detection. Misuse detection techniques trace the abnormalities through matching anomalous signatures of previous attacks with current patterns. This is more similar to antivirus logic. On the other hand, Anomaly detection techniques catch hold on all activities other than normal ones [1]. Hence, here the normal profiles are well identified first to observe deviations of current activities if any. It is the responsibility of this technique to make sure that the deviations identified is well enough to label as 'intrusion'. The quality of the approaches towards detection of anomalous activities can be measured by observing the false positives and false negatives. False positive is the scenario where an event is incorrectly identified by the system as being an intrusion when none has occurred. False negative is the situation where no intrusion has been identified by the system when one has in fact occurred.

Signature based detection systems are well suited for detecting formerly known attacks. On the other hand, anomaly based detection systems are tuned for unknown intrusion scenarios. Certain classifiers are well known for identifying certain classes of attacks. Hence, utilization of multiple signature based and anomaly based classifiers will definitely produce good detection results. Hence, the objective of this paper is to design an intrusion detection system architecture with multiple intrusion detection modules.

The rest of the paper is organized as follows: Section 2 discusses the related works in the area. Section 3 discusses the proposed work. Section 4 discusses the performance evaluation. Section 5 discusses experimental platform and results. The paper is concluded in section 6. In the end, the references are listed.

II. RELATED WORKS

The concept of IDS was first proposed by James P Anderson in his technical report in 1980[2]. He introduced the notion that audit trail calls, application logs, file system modifications and other host activities related to the machine. After seven years, Dr. Dorothy Denning published a model which revealed the necessary information for commercial IDS development [3]. Hansen and Salamon [4] used different ANNs to improve accuracy of classification. Weak learners can be combined to obtain high accuracy. Recently, there are many papers published regarding ensemble approaches [5][6][7]. There are various approaches published for IDS fusions. Ciza Thomas amd N Balakrishnan [8] proposed a fusion of multiple IDS. They used old DARPA dataset which is out

dated and have duplicate records. Categorization to signature based and anomaly based study was not made. Parikh and Chan [9] proposed a data fusion method with minimal cost. Here, the rate of error was not reduced. Siraj and Vaughn [10] proposed and intelligent IDS, which uses various detection sensors. They used artificial intelligence. Improvement in anomaly based detection had to be introduced. Other IDS fusion methods were also proposed [11][12][13][14] in which most of them used old network data leading to less detection accuracy and more error rate when applied to new scenarios. Hence, intrusion detection that can detect new styles of attacks by considering fresh network traffic dataset will definitely contribute to the modern network security systems. This is one of the inspirations for this paper

III. PROPOSED MODEL

3.1 Basic architecture of a normal intrusion detection system

The generic architecture of a normal intrusion detection system is shown below.

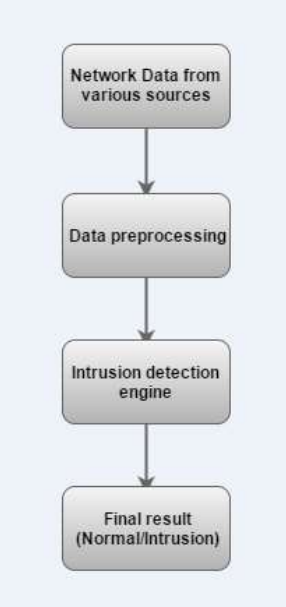

Figure 1

First of all we need the network traffic data. This can be collected from the real time network traffic or can be obtained from various standardized data sets such as DARPA and KDD99. The network data has to be preprocessed for selecting the relevant features by eliminating the un- wanted features that can reduce the detection rate as well as increasing the processing time. This can be achieved through various feature reduction and feature selection techniques. After selecting the relevant features, the data set is trained and tested for better detection accuracy. Good IDS should also be capable of finding novel attacks while reducing the false positives to minimum.

Internet services have increased tremendously by all means. Almost all software are online, shared and even distributed. Information sharing has taken a big leap in explosive manner which has not seen before. This is going up in an exponential manner. Now, various types of novel attacks have been seen in these days due to the entry of different domains to the network information sharing model. It is high time to device new techniques to find novel attacks for the sustainability of the whole system.

Current individual intrusion detection systems concentrates highly on particular types of attacks as they use particular type of classifiers. More over the existing IDSs heavily depend on old data sets which is now almost obsolete and useless for raining new network attacks.

Hardware cost has been tremendously decreased while processing power has increased in a big fashion. So, the opportunity of combining the advantages of various IDS in reaching the final result is largely viable. It can improve the overall performance of detection and accuracy.

We propose a model which combines various signature based and anomaly based IDS. Each type has its own capabilities of detecting particular types of attacks. We use NSL KDD data sets for our work as this dataset is more optimized and redundancy eliminated when compared to old DARPA and KDD cup data sets.

For preprocessing and feature selection, we use a correlation based feature selection algorithm. Here, we experiment with anomaly based classifiers such as SVM, ANN and K-NN classifiers. We also use signature based classifiers such as OneR and J48. Even we can use n number of classifiers with this model for predicting the classes of attacks. Final decision is how ever made by using majority voting technique.

3.2 Dimensionality reduction and feature selection

Dimensionality reduction is the process of minimizing the size of data while maintaining the needed characters. Increase in dimensionality will reduce the effectiveness of various algorithms. Hence, a set of high dimensional vectors are projected to a lower dimensionality space. The dimensionality reduction can be efficiently achieved by feature extraction and feature selection.

We use a correlation based feature selection algorithm for our experiments which is proposed by M.Hall. The algorithm is an excellent one that can remove redundant features and at the same time maintain classification accuracy. The algorithm is depicted below [15].

Algorithm 1

1. // Remove irrelevant features
2. Input original data set D that includes features X and target class Y
3. For each feature Xi, calculate mutual information SU(Y; Xi)
4. Sort SU(Y; Xi) in descending order
5. Put X_j whose SU(Y; Xi) > 0 into relevant feature set R_{XY}
6. //Remove redundant features
7. Input relevant feature set R_{XY}
8. For each feature X_j
   Calculate pair wise mutual information
   SU (X_j ; X_k) ∀j ≠ k
9. S_{XX} = Σ (SU(X_j; X_k))
10. Calculate means μ R and μ S of R_{XY} and S_{XX}, respectively. w = μ S /μ R
11. R = w.R_{XY} – S_{XX}
12. Select X_j whose R > 0 into final set F

3.3 The proposed system architecture

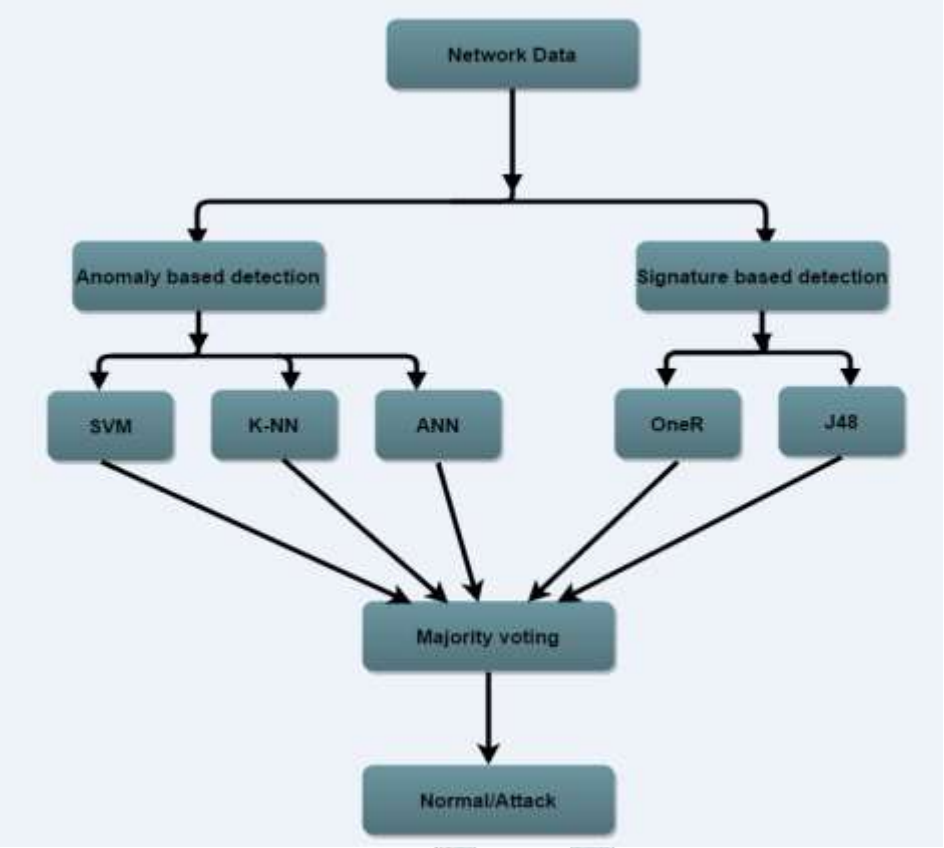

Figure 2

3.4 Anomaly based and signature based classifiers

We use various anomaly based and signature based classifiers for detecting the attack classes. We use both categories because each category has its own special capabilities of detecting attacks. Signature based classifiers are well suited for identifying known attacks while anomaly based classifiers are more efficient in identifying abnormalities. The figure 2 shows the proposed system architecture.

3.5 Support vector machine

Support vector machines are supervised learning models that analyze data and recognize patterns. Support vector machine constructs a hyper plane in multi-dimensional space, which can be used for classification. SVM was proposed by vapnik [16]. It takes input from and built a model which assigns new input data into one of the classes. SVM can perform both linear and non-linear classification. In the non-linear problem, the algorithm solves by extending the original set of variable x in a high dimensional feature space with map Φ. If input vector x∈ $R^d$is transformed to feature vector Φ(x) by a map Φ: $R^d \rightarrow H$, then a function can be found, K $(R', R') \rightarrow R$ that satisfies condition K(x_i, x_j)= Φ(x_i).Φ(x_j) and problem leads to the following quadratic optimization problem,

Minimize $\sum_{i=1}^{k} \alpha_i - \frac{1}{2} \sum_{i=1}^{k} \sum_{j=1}^{k} \alpha_i \alpha_j y_i y_j (x_i x_j)$

Subject to $\sum_{i=1}^{k} y_i \alpha_i = \forall i: 0 \leq \alpha_i \leq C$

3.6 Artificial Neural Network

ANN is modeled on the basis of nervous system of human which is presented as interconnection of neurons. Neurons are processing elements or units. This network of simple processing elements shows complex global behavior determined by the connection between the neurons and their parameters. The back propagation method is used in training multilayer neural networks in supervised manner. The technique is based on error collection learning [17]. It consists of forward and backward pass. In forward pass, an activity is given to the input nodes and its effect propagates through network through each layer. In the end, a set of output is produced as the response of network. In forward pass, synaptic weights are fixed and during backward pass synaptic weights are adjusted using error correction rule. Difference between actual response and desired response is found for producing error signal and propagated back through network.

The neuron activation functions are given below.
Sigmoidal : f(x) = $\frac{1}{1+\exp(-ax)}$, a > 0
Tansig: f(x) = a tanh(bx) , a & b > 0

3.7 K-Nearest neighbor (k-NN)

K-nearest neighbor classifies objects according to its nearest distanced training set in the space. Distance between different objects in the input vector is calculated and the labeling is done according to the k-nearest neighbor. The distance measure is done by calculating the distance between each features of the object and its neighbors. Here, k is small integer. If it is 1, the neighbor's class is given to that object. Otherwise the object is classified according to the majority of votes given by its neighbors; i.e. the object will be assigned in the class in which most of its neighbors are included [18] [19] [20].

3.8 J48 classifier

C4.5 decision tree is the most popular tree classifier which is developed by Quinlan [21]. J48 is a Java implementation of C4.5 in Weka environment [22]. C4.5 builds decision trees from a set of training data using the concept of information entropy. The training data is a set $S = s_1, s_2, \ldots$ of already classified samples. Each sample s_i consists of a p-dimensional vector $(x_{1,i}, x_{2,i}, \ldots, x_{p,i})$, where x_j represent attributes.

Pseudocode:

1. Check for base cases
2. For each attribute *a*
    1. Find the normalized information gain ratio from splitting on *a*
3. Let *a_best* be the attribute with the highest normalized information gain
4. Create a decision *node* that splits on *a_best*
5. Recur on the sublists obtained by splitting on *a_best*, and add those nodes as children of node [23].

3.9 OneR Classifier

OneR is a rule based classification algorithm. Classification is done in a straight forward and simple manner by generating a one level decision tree. Training set is considered and each feature is given one rule. One rule is the rule with least error rate. Most frequent class for every attributes is found.

Pseudocode:

1. Consider an attribute 'Attri'
2. For each value 'AttriVal', of attribute, rule is created and find the frequent class by keeping the count.
3. Calculate the rate of error of the rules.

Choose best rule with least error [24].

3.9 Proposed algorithm for intrusion detection with multiple intrusion detection modules

Algorithm 2
Input: Network traffic with set of features F{}
Output: Find whether attack or normal

1. Relevant features are selected by omitting irrelevant ones using a correlation based feature selection algorithm.
2. Feed the network traffic data to first classifier with features F{} and find out the attack category/normal.
3. Repeat 2 for all other anomaly based and signature based classifiers.
4. All individual classifier decisions are obtained such as D_1, D_2, D_3….Dn.
5. Individual decision 'd' is labeled as seither attack (ATK) or normal (NORM)

If DOS||PROBE||U2R||R2L then
D_1=ATK
Else
D_1= NORM
6. Final decision is taken according to the majority voting technique. Total ATK/NORM count is taken.
If ATK>3
Result=ATK
Else if ATK<3
Result=NORM
Else

Result = decision of the best classifier that has superior capability over others in that particular scenario is selected.

3.10 Combining intrusion detection modules (majority voting/fusion technique)

Various classifiers are used to detect various types of attacks. Each of them has their own capabilities for finding various forms of attacks. The system becomes highly powerful when we combine these classifiers using various techniques. Combining will help to utilize the advantages of each classifier simultaneously. This will increase the detection capability and accuracy. Various combinational methods are available in which we used majority voting technique for combining different classifiers.

In simple words, majority voting algorithm will consider the output of the classifier and feed to other classifiers and select the class of attack which is recommended by majority of classifiers. Incase all classifiers differ in their opinion, the opinion of the first classifier which has superior capability over others in that particular scenario, will be selected.

IV. Performance Evaluation

We have used NSL-KDD data set. When compared to KDD cup99 data set, NSL-KDD data set doesn't contain repetition of records in training and test sets. Performance of any intrusion detection system is verified using the confusion matrix. It depicts accuracy, false alarm rate and detection rate. Table 1 below shows the confusion matrix.

		Predicted class	
Actual class		Attack	Normal
	Attack	TP	FN
	Normal	FP	TN

Table 1

Accuracy= TP+TN/ (TP+TN+FP+FN)
Detection rate = TP/ (TP+FN)
False alarm rate= FP/ (TN+FP)

Where TP = True positive, TN = True negative, FP = False positive, FN = False negative.
The true positives (TP) and true negatives (TN) are correct classifications. A false positive (FP) occurs when the outcome is incorrectly predicted as yes (or positive) when it is actually no (negative). A false negative (FN) occurs when the outcome is incorrectly predicted as negative when it is actually positive.

V. Experimental Platform and Results

Experiments are done on windows OS platform with 2.8 GHz Intel core i5 processor and 4 GB RAM. The software used are Java and Weka (Waikato Environment for Knowledge Analysis) 3.5.7 designed by machine learning group at University of Waikato. NSL-KDD with reduced features is used here. It contains one type of normal and 22 different attack types. All attacks come under any one of the four groups such as DoS, Probe, U2R and R2L. Table 2 below shows the attack category, number of records and percentage out of total.

Attack category	Number of records	Percentage out of total
Normal	34821	53.14%
DoS	24029	36.66%
R2L	528	0.84%
Probe	6108	9.32%
U2R	28	0.04%
Total	**65534**	**100%**

Table 2

Out of 42 features in NSL-KDD data set, 12 features are selected using the correlation based future selection algorithm. The selected features are given below.

count, dst_host_count, dst_host_same_srv_rate, same_srv_rate, dst_host_srv_count, dst_host_same_src_port_rate, protocol_type, serror_rate, dst_host_srv_serror_rate, dst_host_serror_rate, srv_ serror_rate and logged_in.

5.1 Duration for training and testing results for all features and reduced features

Table 3 and table 4 shows the duration (in seconds) taken by each classifiers for training and testing phases. It is obvious that less amount of time is taken with reduced set of features.

Training set			
Classifier	**All features**	**12 features**	**Decrease in training time**
SVM	10.1	8.2	1.9
KNN	8.6	7.3	1.3
ANN	9.3	9.1	0.2
OneR	0.11	0.07	0.04
J48	1.92	1.34	0.58

Table 3

Testing set			
Classifier	**All features**	**12 features**	**Decrease in training time**
SVM	9.6	7.1	2.5
KNN	7.7	7.1	0.6
ANN	8.4	8.1	0.3
OneR	0.092	0.089	0.003
J48	1.69	0.88	0.81

Table 4

5.2 Detection and false alarm rate for individual classifiers

Attack detection rate and false alarm rate of individual classifiers are shown in the table 5 and table 6 respectively. From the table it can be easily known that each type of classifiers have their own capabilities for identifying different attacks. Hence, combinational detection techniques will definitely out perform the individual intrusion detection techniques. Table 5 shows detection rate and Table 6 shows False alarm rate.

Attack type	Individual classifiers				
	SVM	**ANN**	**K-NN**	**OneR**	**J48**
DoS	95.8	99.2	98.6	96.2	99.7
Probe	98.2	96.6	97.2	97.1	98.2
U2R	87	82.5	83.2	81.3	81.1
R2L	95.2	95.5	94.2	95.1	94.6
Normal	95.6	98.2	97.1	96.8	92.2

Table 6

Attack type	Individual classifiers				
	SVM	**ANN**	**K-NN**	**OneR**	**J48**
DoS	0.6	0.2	0.7	0.2	0.1
Probe	0.8	0.1	0.2	0.3	0.2
U2R	1.2	1.1	1.6	0.6	0.8
R2L	0.2	0.3	0.3	0.2	0.1
Normal	3.8	2.1	3.2	1.9	1.2

Table 7

5.3 Detection rate with multiple IDS modules

When multiple IDS modules are utilized for arriving in final decision of attack class, very good performance was shown in attack detection and accuracy. The figure 3 and figure 4 depicts the final outcome.

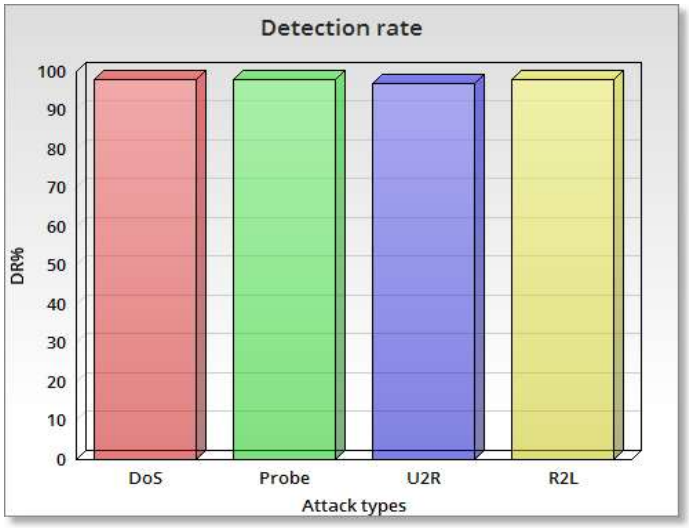

Figure 3

5.4 False alarm rate with multiple detection modules

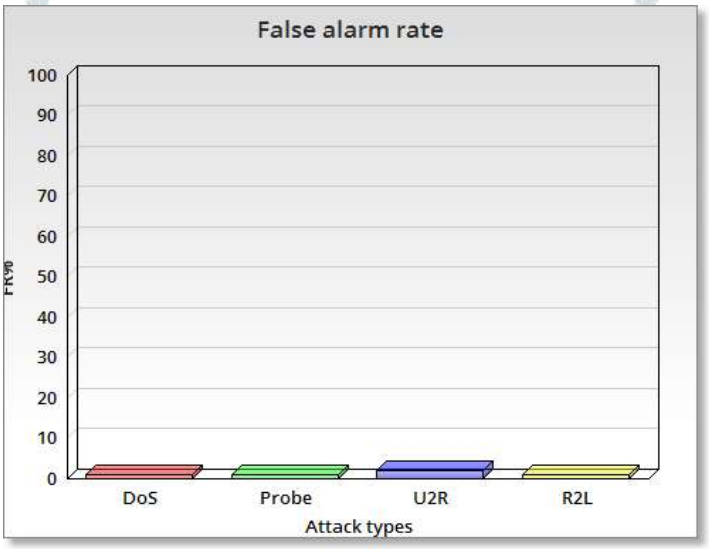

Figure 4

VI. CONCLUSION

Different classifiers perform differently with different attack classes. Signature based detection techniques are highly accurate in detecting known attacks while they are weak in finding novel attacks. At the same time, anomaly detection classifiers are very good in identifying abnormalities (probably new attacks) and categorizing them to an attack class. By taking this into consideration we combined various signature based and anomaly based IDSs for classifying the attacks and used majority voting for reaching into final decision. We proposed an architecture and algorithm. We used NSL-KDD data set with reduced features by utilizing feature selection algorithm. It reduced the training and testing time tremendously.

We evaluated the performance of combined multiple IDS modules. When compared to individual classifiers, combination of multiple IDSs easily outperformed with respect to increase in detection accuracy as well as decrease in false alarm rate.

REFERENCES

[1] Gogoi P, Borah B, Bhattacharyya D, Anomaly detection analysis of intrusion data using supervised & unsupervised approach, Journal of Convergence Information Technology 2010.

[2] J. P. Anderson, Computer security threat monitoring and surveillance, Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, April 1980.

[3] Dorothy E. Denning, An intrusion-detection model, IEEE Trans. Software Eng., 1987.

[4] Lars Kai Hansen and Peter Salamon, Neural Network Ensembles, IEEE Transactions on Pattern Analysis and Machine Intelligence, October 1990.

[5] Emna Bahri, Nouria Harbi, and Hoa Nguyen Huu, Approach Based Ensemble Methods for Better and Faster Intrusion Detectio, In Proceedings of the 4th International Conference on Computational Intelligence in Security for Information Systems, June 2011Springer.
[6] Silvia Gonz´alez, Javier Sedano, Alvaro Herrero, Bruno Baruque, and Emilio Corchado, Testing ensembles for intrusion detection: On the identification of mutated network scans, In Proceedings of the 4th international conference on Computational intelligence in security for information systems, CISIS'11, Torremolinos-Malaga, Spain, June 2011. Springer-Verlag.
[7] Peng Zhang, Xingquan Zhu, Yong Shi, Li Guo, and Xindong Wu, Robust ensemble learning for mining noisy data streams, Decision Support Systems, January 2011.
[8] Ciza Thomas and N Balakrishnan, Improvement in Intrusion detection with advances in sensor Fusion, IEEE Transaction on Information Forensics and security September 2009.
[9] Parikh D., Chen T. Data fusion and cost minimization for intrusion detection. *IEEE Transactions on Information Forensics and Security*. 2008;3(3):381–389. doi: 10.1109/tifs.2008.928539. [Cross Ref]
[10] Siraj A., Vaughn R. B., Bridges S. M, Intrusion sensor data fusion in an intelligent intrusion detection system architecture, Proceedings of the Hawaii International Conference on System Sciences; January 2004
[11] Kamran Shafi Hussain A Abbass, An adaptive genetic based signature learning system for Intrusion detection‖ , Elsevier Experts system with application 2009
[12] Su-Yun Wua, Ester Yen b, Data mining-based intrusion detectors, Elsevier machine learning methods in intrusion detection system 2008
[13] Y. Wang, H. Yang, X. Wang, and R. Zhang, Distributed intrusion detection system based on data fusion method, in Intelligent Control and Automation (WCICA), Hangzhou, China, June. 2004.
[14] G. Giacinto, F. Roli, and L. Didaci, ―Fusion of multiple classifiers for intrusion detection in computer networks, Pattern Recognit. Lett., vol.24, August 2003
[15] T. S. Chou, K. K. Yen, and J. Luo, Network Intrusion Detection design Using Feature Selection of Soft Computing Paradigms, World Academy of Science, Engineering and Technolog 2008
[16] V. Vapnik,. Statistical Learning Theory. Wiley, New York, 1998
[17] J. Principe, N. Euliano, W. Lefebvre, Neural and Adaptive System – Fundamentals Through Simulations, Wiley, 2000.
[18] Hossein M.Shirazi, Anomaly Intrusion Detection using Information Theory,k-NN and KMC Algorithms, Australian Journal of Basic and Applied Sciences 2009.
[19] Benjamin Thirey and Christopher Eastburg, Increasing Accuracy Through Class Detection: Ensemble Creation Using Optimized Binary Knn Classifiers, IJCSEA, Vol.1, No.2, April 2011.
[20]Manocha, S. and Girolami, M, An empirical analysis of the probabilistic K-nearest Neighbor Classifierm, Pattern Recognition Letters, Vol. 28, 2007
[21] Quinlan, J, C4.5: Programs for Machine Learning. Morgan Kaufmann, San Mateo (1993)
[22] Weka – Data Mining Machine Learning Software, http://www.cs.waikato.ac.nz/ml/weka/
[23] Http://en.wikipedia.org/wiki/C4.5_algorithm
[24] Witten, I.H., Frank, E, Data Mining: Practical Machine Learning Tools and Techniques, 2nd Morgan Kaufmann, San Francisco (2005).