ISSN:2249-5789

DESIGN OF INTRUSION DETECTION SYSTEM BASED ON ARTIFICIAL NEURAL NETWORK AND APPLICATION OF ROUGH SET

Dilip Kumar Barman
Sr. Manager, ERNET India
6, CGO Complex, New Delhi
barman@eis.ernet.in

Dr.Guruprasad Khataniar
Department of Computer Engineering
Assam Engineering Institute Guwahati
drkhataniar@gmail.com

Abstract

Securing data in a networked environment has been a major concern for Network Administrator as intruders may get access and steal the information available in the Computer network. As absolute security is not possible in a network, detecting intrusion is very important from the standpoint of protection of the information as well as the network. The paper intends to cover the development of an Intrusion Detection System based on Neural Network Systems. As the Intrusion Detection Systems (IDS) have to depend on known signatures, we have to train the IDS about the signatures. KDD99 is a freely available dataset for intrusion signatures and we intend to use KDD99 dataset for both training and testing our IDS. As the number of input attributes for the signatures to the IDS (for detection of the intrusion of the network) is quite high(41 in all together), minimization of the inputs to the network is very important as processing time for the inputs to be kept minimized for real time detection. Selection of the most relevant features is very important for this and concept of Rough Set has been applied for selection of the most relevant features. Effects of minimization of input features for the signatures, through use of Rough Set for Detection of Intrusions in a network, have been studied in this research paper.

Keywords: KDD99, DOS, Rough Set (RST).

1 INTRODUCTION

Securing data in a networked environment has been a major concern for network administrator and security personnel responsible for the network. Networks are under threat from intruders. The intruders may get access to the network and in turn may steal valuable data, causing immense damage to the network and the application running based on the data base. Detection of Intrusion is very much essential as it is almost impossible to make a network completely secure. Many methods and techniques have been in place for detection of intrusion. However, almost all intrusion detection methods/techniques will require some kind of signatures for training the Intrusion Detection System (IDS). Once, the system has been trained with some kind of signatures, IDS is ready for detection of that type intrusion. In this paper, we intend to use Artificial Neural Network (ANN) with back propagation as IDS. The ANN based IDS system will use the attributes (total of 41) of the intrusion signatures of the dataset KDD99. The output of the ANN will be either 1 or 0 based on the fact that the packet is infected or not with intrusion. However, the inputs of total 41 features [1] for each signature will make the performance of the IDS slower. Descriptions of the 41 features of KDD99 are available in [1] for reference. Concepts of Rough Set [2][3] based Dependency Ratios for determining relevance of features on the decision attribute (output in this case), have been applied in this paper. Based on Dependency Ratios, relevant features are selected (on the decision attribute) and the selected features are then used as inputs to the IDS. Performances of IDS, based on selected features, for detection of intrusions are being examined

2 METHODS

The design of the proposed IDS system is based on Neural Network with back propagation for visualizing intrusion. The ANN implemented in [3] has been taken as ANN tool for detecting intrusion [16]. The architecture of the IDS system based on Artificial Neural Network and implemented in [3] will consist of a three layers: one input, one hidden, and one output layers with one feedback layer from output to the hidden layer. The ANN has been adjusted as per the learning process, based on the selected sample (signature) of attacks which can be seen in *MAPWIGHTS.WTS* file created by default in the working directory of the ANN program.

The proposed system will be trained [8] by using the KDD99 based training dataset [4] and its performance will be evaluated [9] by KDD99 test dataset. The dataset contains a total of 22 training attack types, with an additional 14 types of attacks in the test data only.

As there are total of 41features/attributes [4] for each signature, the exiting IDS consider 41 input neurons as input for each type of intrusion and either 1 or 0 as output for each type of attack/intrusion. However, while analyzing [10], we have observed that concepts of Rough Set [2][5] may be utilized for minimization of the attributes/features to be used both for training and testing of the Neural Network based Intrusion Detection Systems (IDS). Implementation of Rough Set will enable us to use only the relevant fields [6] for our design. The processing time will subsequently be reduced [12] substantially for the IDS.

Our proposed system has been tested against the KDD 99 test data [4](“*corrected.gz*” Test data with corrected labels) and performances for the systems have been evaluated and compared to the existing systems [7]. The block diagrams for our proposed system are given below:

2.1 THE BLOCK DIAGRAM OF THE PROPOSED INTRUSION DETECTION SYSTEM DESIGNED FOR DETECTION OF INTRUSION

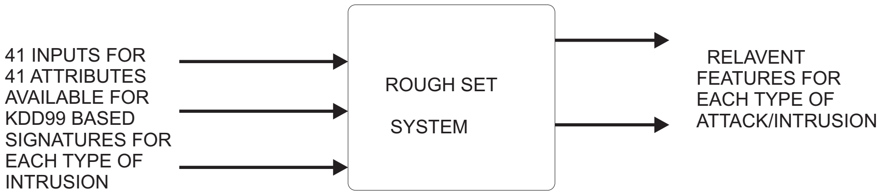

Figure 1: Application of Rough Set for selection of relevant features for a class of intrusion

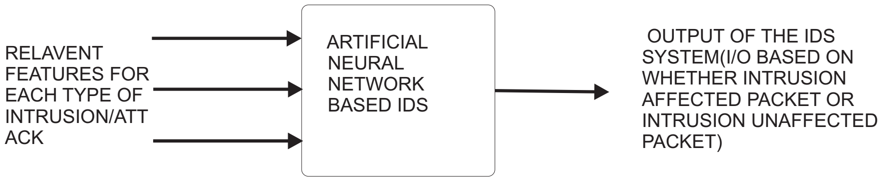

Figure 2: Application of ANN for detection of intrusion for a class of intrusion

The Rough Set based system Figure 1 will take the input signatures of attacks with all the 41 attributes. Based on the dependency ratios [14], the system will output the signatures with only the most relevant features. The signatures with the most relevant attributes are then fed into the Neural Network part of the IDS for training the IDS with [15], and then for testing for detection of Intrusions.

3 RESULTS AND DISCUSSIONS

The typical signatures from KDD99 training dataset [4] are as follows:

0,tcp,http,SF,181,5450,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.00,0.00,9,9,1.00, 0.00,0.11,0.00,0.00,0.00,0.00,0.00,normal.

```
0,tcp,http,SF,239,486,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,8,8,0.00,0.00,0.00,0.00,1.00,0.00,0.00,19,19,1.00
,0.00,0.05,0.00,0.00,0.00,0.00,0.00,normal.

......

0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.50,1.00,0.00,1.00,1,1,1.0
0,0.00,1.00,0.00,0.00,0.00,0.00,0.00,back.

0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,2,3,0.00,0.00,0.00,0.33,1.00,0.00,0.67,2,2,1.0
0,0.00,0.50,0.00,0.00,0.00,0.00,0.00,back.

......

23,tcp,telnet,SF,104,276,0,0,0,0,5,0,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,1.00,0.00,0.00,1,2,1.0
0,0.00,1.00,1.00,0.00,0.00,0.00,0.00,guess_passwd.

......

0,icmp,ecr_i,SF,1480,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.00,0.00,0.00,86,2,0.0
2,0.05,0.02,0.00,0.00,0.00,0.00,0.00,pod.

......

0,tcp,http,SF,291,1200,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,6,12,0.00,0.00,0.00,0.00,1.00,0.00,0.17,26,255,
1.00,0.00,0.04,0.05,0.04,0.01,0.00,0.00,normal.

......

0,tcp,http,SF,219,1234,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,6,35,0.00,0.00,0.00,0.00,1.00,0.00,0.14,6,255,
1.00,0.00,0.17,0.05,0.00,0.01,0.00,0.00,normal.
```

Here signatures of all the attacks are available in a file. In our proposed design for the IDS, signatures of a particular attack are extracted from the training dataset by the code *SEPARATOR.C*. The separated signatures for a particular type of attack are saved in a file named *ABC.TXT* where ABC is the attack type. As for example, the contents of the file *BACK.TXT* which contains the signatures of *back* attack, which is a Denial of Service (DOS) attack, will be as

```
0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,1,2,0.00,0.00,0.00,0.50,1.00,0.00,1.00,1,1,
1.00,0.00,1.00,0.00,0.00,0.00,0.00,0.00,back.

......

0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.00,0.00,99,99,
1.00,0.00,0.01,0.00,0.01,0.01,0.01,0.01,back.

0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,3,3,0.00,0.00,0.00,0.00,1.00,0.00,0.00,100,100,
1.00,0.00,0.01,0.00,0.01,0.01,0.01,0.01,back.
```

Thus *BACK.TXT* will have signatures of only *back* attack. Similarly for all other attacks, signatures will be saved in their respective files.

The input signatures of a particular attack, contained in a file are then input to the Attribute Extractor part of the IDS. Attribute Extractor part of the IDS will extract the most relevant attributes based on dependency ratios, which is a concept from Rough Set. As for example, the relevant attributes/fields for *back* attack are 5 and 6 with dependency ratios on the decision attribute (which is the output represented mathematically as 1 or 0 depending on whether the signature is for *back* attack or not) for the signatures. This part of the IDS which has applied the concepts of Rough Set [2][3] for determining the most relevant field for each type of attack will produce the most relevant fields for a particular type of attack. Block diagram of this part of IDS is shown in Figure 2.

The most relevant fields for particular attacks are then fed into the ANN for training the ANN as IDS for that particular attack. As for example, the relevant fields (fields no 5 and 6 which source bytes and destination bytes) for *back*, which is a typical Denial of Service (DOS) attack are fed into the neural network for training for *back* attack. The formats in which the input data for the *back* attack are fed input to the Neural Network are in the format below:

```
54540 8314
1

...

54540 8314
1

...

54540 8314
1

...

20440 1460
1
46720 8314
1

...

46720 8314
1
```

The first line for each signature with reduced fields,(with only 2 nos of attributes separated by a space) is the input line for the ANN and the 2nd line for each signature(with value of 1) is the output for the training of the ANN. The 1s have been inserted in the second line for each signature to the file *FEABACK.TXT* (feature of back), for tallying the input formats for the ANN program of [3].

The same processes right from separating the signatures for a particular attack to minimizations of the input fields based on dependency ratios on and inputting the attributes to the ANN as per the format required for inputting to the ANN, are repeated for the testing dataset [4] also. Observed Detection Rates are tallied with the available detection rates from other IDS systems based on KDD99 dataset. The detection rates for our IDS systems closely resemble with those available from other IDS systems. As a matter of fact, we have achieved 100% detection for the test data available in the test dataset for KDD99. However, our system is at least 20.5(41/2) times faster in detection for *back* attack. Hence, our system may be used effectively for online detection of intrusion.

4 Conclusion

The detection rates of our system (IDS) closely tally with those of the results available from other kinds of IDS systems based on KDD99 dataset. However, our system is minimum 20.5 times faster in detection for "*back*" which is a Denial of Service (DOS) attack. The exact Time Complexity for the system developed by us is further to be studied and will be taken in our future studies. Moreover, in this paper we considered the performance for detection of intrusions for only a specific numbers of attacks/intrusion. Studies are to be carried out for all other classes of attacks for making the system to be utilized for online network and applications. We expect to carry out full-fledged studies in the near future, covering all the possible attacks and making the system applicable for live network.

References

[1] H. Gunes, Kayacik, A. Nur Zincir-Heywood, Malcolm I. Heywood, "*Selecting Features for Intrusion Detection: A Feature Relevance on KDD99 Intrusion Dataset*", Third Annual Conference on Privacy, Security and Trust, St. Andrews, New Brunswick, Canada,October 2005. .

[2] Zhang Lian-hua, Zhang Guan-hua, Yu Lang, Zhang Jie, Bai Ying-cai,*Intrusion detection using rough set classification*, Journal of Zhejiang University Science, ISSN 1009-3095, Shanghai, China.

[3] Dr.Manry, Dr.Xun Cai, Kanishka Tyagi, *MLP-Approximation source code*, IPNN Lab, UT Arlington, July, 2010,

[4] "*KDD 99 Dataset*", Website Reference: http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.

[5] Shusaku Tsumoto, *Statistical Evidence for Rough Set Analysis*, Proceedings of the IEEE International Conference on Fuzzy Systems, 2002. FUZZ-IEEE'02, Vol 1, pp 757 - 762.

[6] Adetunmbi A. Olusola., Adeola S.Oladele, Daramola O. Abosede, *Analysis of KDD99 Intrusion Detection Dataset for Selection of Relevance Features*, Proceedings of the World Congress on Engineering and Computer Science, 2010, Vol I, San Francisco, USA.

[7] P. Ganesh Kumar and D. Devraj, *Network Intrusion Detection using Hybrid Neural Networks*, IEEE-ICSCN 2007, MIT Campus, Anna University, Chennai, India, 2007, pp 563 - 569.

[8] Xin Xu, *Adaptive Intrusion Detection Based on Machine Learning: Features Extraction, Classifier Construction and Sequential Pattern Prediction*, International Journal for Web Services Practices, Vol.2, No1-2(2006), pp 49-58.

[9] Mahbod Tavellaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani, *A Detailed Analysis of KDD CUP 99 Dataset*, Proceedings of the IEEE Symposium on Computational Intelligence in security and Defense Applications (CISDA 2009), pp 1-6.

[10] Mostafa A. Salama,Heba F. Eid,Rabie A. Ramadan,Ashraf Darwish,and Aboul Ella Hassanien, *Hybrid Intelligent Intrusion Detection System*, Soft Computing in Industrial Applications, Advances in Intelligent and Soft ComputingVolume 96,2011,pp 293-303.

[11] T. Subbulakshmi1, George Mathew2, Dr. S. Mercy Shalinie, *Real Time Classification and Clustering of IDS Alerts Using Machine Learning Algorithms*, International Journal of Artificial Intelligence & Applications (IJAIA), Vol. 1, No.1, January 2010.

[12] H. Gune Kayack, A. Nur Zincir -Heywood, Malcolm I, *Selecting Features for Intrusion Detection, A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets*, PST, DBLP:conf/pst/2005, 2005,

[13] Jonatan Gomez and Dipankar Dasgupta, *Evolving Fuzzy Classifiers for Intrusion Detection*, Proceedings of the 2002 IEEE Workshop on Information Assurance United States Military Academy, West Point, NY June 2001.

[14] Mouhcine Guennoun, Zine E.A Guennoun and Khalil El-Khatib, *Impact of Feature Reduction on the Efficiency of Wireless Intrusion Detection Systems*, International Conference on Computer Engineering and Applications IPCSIT vol.2 (2011) IACSIT Press, Singapore pp. 270-274.

[15] Dong Song, Malcolm I. Heywood, A. Nur Zincir-Heywood, *Training Genetic Programming on Half a Million Patterns: An Example from Anomaly Detection*, Trans. Evol. Comp9, 3 (June 2005), 225-239.

[16] Maheshkumar Sabhnani, Gursel Serpen, *Application of Machine Learning Algorithms to KDD Intrusion Detection Dataset with Misuse Detection Context*, EECS Dept, University of Toledo, Ohio, MLMTA 2003:209-215.