*J. Basic. Appl. Sci. Res.*, 2(5)4667-4671, 2012

© 2012, TextRoad Publication

ISSN 2090-4304
**Journal of Basic and Applied**
**Scientific Research**
*www.textroad.com*

# Intrusion Detection System Using New Synthetic Neural Networks

**Adel Jahanbani [1], Manijeh Keshtgari [2], S Amirhassan Monadjemi[3]**

[1]Department of Computer Engineering Islamic Azad University, Lamerd Branch, Lamerd, Iran
[2]Department of Computer Engineering and Information Technology, Shiraz University of Technology, Shiraz, Iran
[3]Department of Computer Engineering, Esfahan University, Iran, Esfahan

## ABSTRACT

This paper propose a new training paradigm in which we use the data representing abnormal behavior (in contrast to the conventional use of the data representing normal behavior) in computer networks to train a neural network based anomaly detection system in computer networks. We apply our proposed paradigm to an anomaly detection system that is constructed using a Self Organizing Features Map (SOFM) and a Generalized Feed Forward (GFF) neural network. The new training paradigm in this system yields the same performance level or better as compared to other existing systems, but with about 70% reduction in its Computational complexity.

**Key words:** Intrusion Detection System, Anomaly Detection, Self Organizing Features Map, Generalized Feed Forward, Neural Networks.

## 1. INTRODUCTION

Computer networks are increasingly subjected to security threats. In the past two decades, extensive research on devising various techniques to improve data confidentiality, information integrity, and service availability have resulted in systems that are more robust. Nevertheless, we are witnessing novel attacks on a daily basis, which require continued efforts to deal with anomalies in computer networks that emanate from such threats. However, it is evident that one cannot deal with attacks only by relying on conventional tools such as cryptography, security policies, firewalls, or other available means. This is because operating systems as well as application software usually contain bugs or other unavoidable weaknesses that enable potential attackers to exploit certain weaknesses of networking protocols to initiate attacks.

An Intrusion Detection System (IDS) is an effective tool that can help to prevent unauthorized access to network resources. The methods used in IDSs can be categorized as misuse detection or anomaly detection. Misuse detection utilizes predefined patterns of known attacks, while anomaly detection is based on abnormal events on hosts or in the network. The underlying belief in anomaly detection is that such abnormal events are usually followed by intrusion attempts.

Anomaly detection does not require predefined patterns of attacks, but needs extensive training to extract patterns of normal behavior. It is for this reason that development of anomaly detection systems is more complicated than misuse detection systems.

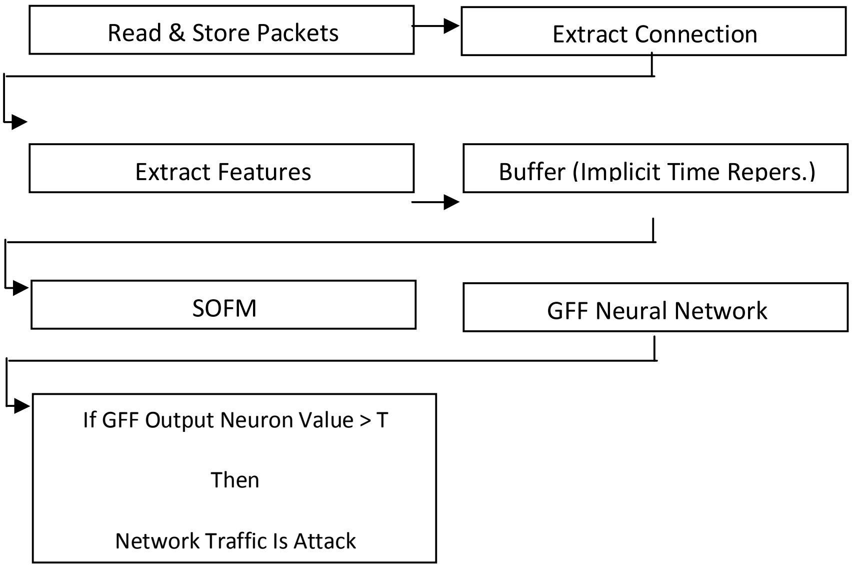

Read & Store Packets
Extract Connection
Extract Features
Buffer (Implicit Time Repers.)
SOFM
GFF Neural Network
If GFF Output Neuron Value > T
Then
Network Traffic Is Attack

Figure 1. The Block Diagram o the proposed Intrusion Detection System

*Corresponding Author:** Adel Jahanbani, Department of Computer Engineering Islamic Azad University, Lamerd Branch, Lamerd, Iran. Email: jahanbani_adel@yahoo.com

Rule based anomaly detection systems look for specific signatures in the monitoring data, and require frequent updates to add new rules for covering new cases. In contrast, statistical anomaly detection systems look for abnormal activities in the behavior of the users. Anomaly detection systems suffer from two types of false alarms. A false negative alarm (FN) that identifies a situation in which no abnormal activity is detected, but an intrusion has occurred. In contrast, a false positive alarm (FP) detects an abnormal activity that is not related to an intrusion attempt. As false alarms are not desirable, one must avoid such indications. Usually, a reduction in false negative alarms results in an increase in false positive alarm, and vice versa. In practice, one has to seek a trade-off between the two. As an intrusion alarm would initiate certain defensive actions by the network manager, in this research we aim to reduce false positive alarms to improve the credibility of alarms. We recognize that such a policy will inevitably result in failure to detect SOFM new anomalies.

Neural networks have been employed in intrusion detection systems [2]-[3], in which SOFMs are utilized to develop the IDSs. In this paper, we develop an anomaly detection system that consists of a preprocessing block and a neural network block. We extract certain features in the preprocessing block, and then use them in a combined SOFM and GFF neural networks to detect anomalies in the computer network. In all intrusion detection systems, the amount of monitoring data is quite significant, which results in a proportional processing overhead. Our choice of a small set of distinguishing features reduces the processing overhead, and enables us to detect anomalies in real time. Also, the way in which we combine the capabilities of SOFM and GFF neural networks substantially improves the performance in comparison with existing systems as presented in the Results Section.

## 2. METHOD

As shown in Fig-1, two SOFM and GFF neural networks are used. At first network packets, consisting of 41 features, were read and saved for processing. Then we processed the saved packets to extract information and connection features to be saved at the final of a buffer. n×m elements (the number of features is called by n and the number of connections by m) are applied to the neural network inputs which are designed with SOFM and GFF. In order to distinguish the occurrence or non occurrence of attack. Moreover this paper provides the related argument, based on explaining the procedure that contains 2 steps; one the way of extracting and preparing data, and the other the method of designing combined neural network.

### 2-1. Data Set

The information technology group at MIT Lincoln Laboratories (with DARPA support) has provided a collection of 5 million label connections for IDS training, and another 2 million for IDS test and evaluation [7]. Each connection packet that is associated with an attack is labeled in one of the following attack categories: Remote-to-Local, User-to-Root, Denial-of-Service, and Probing. This data base was further processed [8] for clustering the packets into the following 4 main groups: Basic TCP, Content and the presumption that certain distinct features are different in normal and abnormal cases. In this research, we exploit only the basic TCP category with 6 features as identified in *Table 1* and show that these features can distinguish normal from abnormal cases.

Since attacks are initiated sequentially, and as the SOFM and GFF neural networks are not capable of recalling temporal information, we use an implicit scheme to represent time by inputting the extracted features into the SOFM and GFF in a sequential manner that corresponds to their arrival times. In this scheme, *m successive connection features (m six-feature vectors) form one input vector for the SOFM neural network. We chose the arbitrary value of 6 for m in this paper; hence the number of input neurons in the SOFM is 36.*

### *2-1. Neural Network Architecture and Training*

We introduce a novel combination of SOFM and GFF and show that it reduces the processing overhead and detects anomalies accurately. We began with 64000

Connections from DARPA data set to train the neural networks. We designed and trained the SOFM separate from the GFF in the following manner using ***Neurosolution*** software: First of all we designed two SOFM neural networks, parameters of which are shown in Table 2. Each of which is composed of 36 input (m=6, n=6) and an output one. So totally there would be 2 outputs in these two networks. The first SOFM network is trained just by connective features with unattack label and the second with attack label.

Now, some connections which have both attack and unattack labels are applied one time to the first neural network and then to the second one. The output rates of the 2 networks would be the real rates instead of 0, 1 (unattack, attack). Consequently, by using these 2 outputs, the designed GFF neural network with Table 2 parameters would be trained. It contains 0, 1 outputs that 1 represents attack label and 0 represents unattack label.

## 3. RESULTS

In this paper, to assess the designed intrusion detection system over 200000 varied connections were used, 40%of which. were attack connections and 60% unattack ones, testing the neural network. 80% of connections were used in training and 20% in testing the IDS.

The output rate of GFF neural network changes between range of [0, 1], it changes to 0 or 1 by choosing the appropriate Threshold to account the minimum false positive (FP) and the false negative (FN).False Negative (FN) and False Positive (FP) alarm rates are defined as:

$$\text{FN}=\frac{\text{No. of attack patterns having score smaller then T (detected as normal)}}{\text{Total number of attack patterns}}\times 100$$

$$\text{FP}=\frac{\text{No. of normal patterns having score smaller then T (detected as attack)}}{\text{Total number of normal patterns}}\times 100 \qquad (1)$$

After several varied examinations, we found that by choosing the Threshold which is 0.01 for GFF the FN would be reduced to 0.049% and 0% respectively. (Table 3). on the other hand, compromise between FP and FN, would happen by choosing varied Thresholds. the selection is dependant to different applications but the best status is minimizing the FN.

Comparing the results with the results in [4] in which FP is 22% and FN is 33%. in [14] shows that FP is 76% and FN 0% .also the result of [2] is nearly equal to [4]. The result of [13] FP is 21.9% and FN is 20.5%.[15] shown that FP is %8.8 and FN %13.The result of [16] FP is %1.5 and FN is %0.

Consequently on contrasting to the other's, in other method considerable reductions has happened in both FP and FN. And, it is when there are reductions in complication of processing. The complication of this method is 303 according to Eq.2 while in [2] is1518 (=36×*36+36*×*6*) and in [13] is 1050(=36×*25+25*×*6*).[15] shown complication is 1027 .

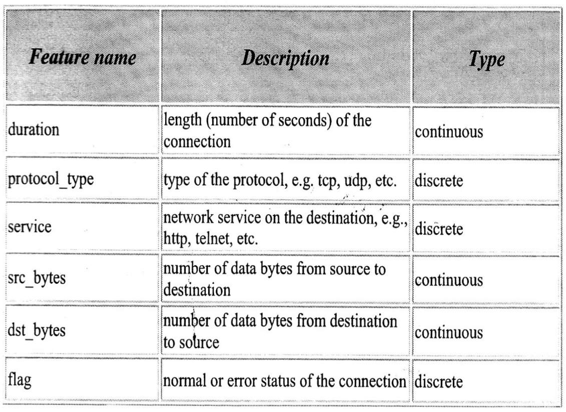
Our method result is preferred to [3] and [4] result about the complication, the FN and FP. the complication reduction in this method is related to the few output neurons of tow SOFM neural networks.

**Complexity**= *No of SOFM Out put neurons × (2× No. of input neurons -1) +No of GFF hidden layer neurons× (2× No. of SOFM output neurons -1) +No of output GFF neurons× (2× No. of GFF hidden layer neurons -1)* (2)

Table 1-Bassic feature in TCP Connections

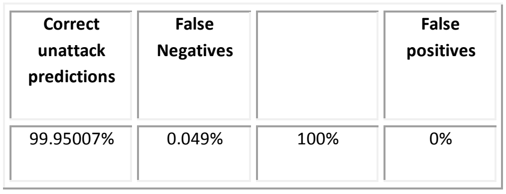
| Function/Parameter Name | Designation/Value |
|---|---|
| Transfer function | Liner Sigmoid Axon |
| Number of Hidden Layers | 1 |
| Number of Output Layers | 1 |
| Learning Rule | Momentum |
| Step Size For Learning | 1 |
| Momentum | 0.7 |
| Neighborhood shape | Square kohonen Full |
| Number of Epochs | 1100 |
| Rows | 6 |
| Columns | 6 |
| Output PEs | 1 |
| NO of *SOFM hidden layer neurons*(PEs) | 6 |
| NO of *GFF hidden layer neurons*(PEs | 4 |

**Table 2-**Desigen parameters for the SOFM and GFFMneural networks



| *Feature name* | *Description* | *Type* |
|---|---|---|
| duration | length (number of seconds) of the connection | continuous |
| protocol_type | type of the protocol, e.g. tcp, udp, etc. | discrete |
| service | network service on the destination, e.g., http, telnet, etc. | discrete |
| src_bytes | number of data bytes from source to destination | continuous |
| dst_bytes | number of data bytes from destination to source | continuous |
| flag | normal or error status of the connection | discrete |

**Table.3** Current results



| Correct unattack predictions | False Negatives | | False positives |
|---|---|---|---|
| 99.95007% | 0.049% | 100% | 0% |

## 4. CONCLUSION

We used our previously proposed intrusion detection system in [2] and proved that training the system with abnormal behavior (in contrast to the conventional practice of training the system with normal behavior) would enhance even further its superior performance as compared to other existing intrusion detection systems reported in the literature [2], [3], [4], [13],[14],[15] and [16].

## 5. REFERENCES

[1] H. Debar, M. Dacier, and A. Wespi, Towards a Taxonomy *of Intrusion-Detection Systems, IBM Research Report, RZ 3030 (#93076), 06/01/98, Zurich, 1998.*

[2] A. R. Sharafat and M. Rasti, "real time anomaly detection in computer networks using self organizing maps and back propagation neural networks", to be published in the *proceedings of the IST 2003.*

[3] P. Lichodzijewski, A. N. Zincir-Heywood, and M. I. Heywood, "Dynamic intrusion detection using self-organizing maps", Proceedings of the 14 Annual Canadian Information Technology Security Symposium – CITSS 2002, pp. 127-131, 2002.

[4] W. Lee, S. J. Stolfo, and K. W. Mok, "Mining in a data- flow environment: experience in network intrusion detection, in Knowledge Discovery and Data Mining Journal, pp. 114-124, 1999.

[5] H. Javitz and A. Valdes, "The SRI IDES statistical anomaly detector", Proceedings of the IEEE Symposium in Security and Privacy, Oakland, CA, May 1991.

[6] P. Lichodzijewski, A. N. Zincir-Heywood, and M. I. Heywood, "Host-based intrusion detection using self organizing maps", Proceedings of the 2002 IEEE World Congress on Computational Intelligence, pp. 1714-1719, 2002.

[7] A. K. Ghosh, J. Wanken, and F. Charron, "Detecting anomalous and unknown intrusion against programs", *Proceedings of the IEEE Conf. on Security Applications, pp. 259-267, December 1998.*

[8] K. Labib and R. Venmuri, *"NSOM: a real-time networked based intrusion detection system using self-organizing maps*", Available [online]: http://www.cs.ucdavis.edu/~vemuri/papers/som-ids.pdf.

[9] B. Rahodes, J. Mahaffey, and J. Canady, *"Multiple self organizing maps for intrusion Detection System, Proceedings", of the NISSC Conference, 2000.*

[10] B. V. Nguyen, "Self organizing map (SOM) for anomaly detection", Available [Online]: http://132.235.28.162/bnguyen/papers/IDS_SOM.pdf.

[11] DARPA Off-Line Intrusion Detection Evaluation Schedule, Available [online]: http://www.ll.mit.edu/IST/ideval/docs/1998/schedule.html

[12] The Third International Knowledge Discovery and Data *Mining Tools Competition, Available [online]:* http://kdd.ics.uci.edu/databa ses/kddcup99.kddcup99.html

[13]A. R. Sharafat and M. Rasti *"Neural network based anomaly detection in computer networks: a novel training program"*, to be published in the *proceedings of the IST 2006.*

[14] A. Bivens and C. Palagiris, "Network-based intrusion detection using *neural networks"*, Rensselaer polytechnic Institute, Tory, New York 12180-3590, 2006 .

[15] A. Abrahamaand R.Jainb and J. Thomas and S. Yong Hana"Distributed soft computing intrusion detection system",doi:10.1016/j.jnca.2005.06.001, Available [online]:. www.elsevier.com/locate/jnca,2007.

[16]V.Venkatachalam and S.Selvan"Intrusion Detection using an Improved Competitive Learning Lamstar Neural Network"IJCSNS International Journal of Computer Science and Network Security", VOL.7 No.2, February,2007.