J. King Saud Univ., Vol. 18, *Comp. & Info. Sci*., pp. 27-45 (A.H. 1426/2006)

DoS Attacks Intelligent Detection using Neural Networks

Abdulkader A. Alfantookh
Department of Computer Science, College of Computer & Information Sciences
King Saud University, P.O. Box 51178, Riyadh11543, Saudi Arabia
fantookh@ksu.edu.sa

(Received 15 August 2004; accepted for publication 04 July 2005)

Abstract. The potential damage to computer networks keeps increasing due to a growing reliance on the Internet and more extensive connectivity. Intrusion detection systems (IDSs) have become an essential component of computer security to detect attacks that occur despite the best preventative measures. A problem with current intrusion detection systems is that they have many false positive and false negative events. Most of the existing Intrusion detection systems implemented nowadays depend on rule-based expert systems where new attacks are not detectable.

In this paper, a possible application of Neural Networks is presented as a component of an intrusion detection system. An intrusion detection system called Denial of Service Intelligent Detection (DoSID) is developed. The type of Neural Network used to implement DoSID is feed forward which uses the backpropagation learning algorithm. The data used in training and testing is the data collected by Lincoln Labs at MIT for an intrusion detection system evaluation sponsored by the U.S. Defense Advanced Research Projects Agency (DARPA). Special features of connection records have been identified to be used in DoS (Denial-of-Service) attacks. Several experiments have been conducted to test the ability of the neural network to distinguish known and unknown attacks from normal traffic. Results show that normal traffic and know attacks are discovered 91% and 100% respectively. Also it has been shown in the final experiment that the false negative of the system has been reduced considerably.

Keywords: Intrusion detection, Neural Network, anomaly detection, Network-Based detection, Denial-of-Service

1. Introduction

The potential damage that can be inflicted by attacks launched over the Internet keeps increasing due to growing reliance on the Internet and more widespread connectivity. Intrusion detection systems (IDSs) have now become an essential component of computer security: to detect attacks that occur despite the best preventive measures. Some approaches detect attacks in real-time and can be used to monitor and, possibly, stop an attack in progress. Others provide after-the-fact forensic information about attacks and can help repair damage, understand the attack mechanism, and reduce the possibility of future attacks of the same type. More advanced IDSs detect never-before-seen (new) attacks, while the more typical systems detect previously seen (known) attacks.

A set of attempts to compromise a computer or a computer network resource security is regarded as an intrusion. In addition, to security services (e.g. data confidentiality, integrity, authentication, etc.), intrusion detection (ID) techniques are used to strengthen the system security and increase its resistance to internal and external attacks. These techniques are implemented through an IDS. Generally, the main task of IDS is to detect an intrusion and, if necessary or possible, to undertake some measures to eliminate further intrusions.

In this paper, a Denial of Service Intelligent Detection (DoSID) System is presented. The system is developed using the feed-forward Neural Network (NN) and related improvements such as Gray Area and Distribution to improve the detection accuracy of Denial of Service (DoS) attacks.

In the following section, a brief introduction to IDSs and Neural Network concepts is given. In section 3, previous works related to IDS using NN have been discussed. DoSID framework and related improvements are explained in section 4. Finally, results of our experiments are shown in section 5.

2. IDS and Neural Network

An IDS is a security system that monitors computer systems and network traffics and analyzes that traffic for possible hostile attacks originating from outside the organization and also for attacks originating from inside the organization.

There are two general approaches to ID namely: misuse detection and anomaly detection. Methods of the first group operate with prior prepared patterns, also called signatures, of known attacks that are used to detect intrusions by pattern matching on audit information. Methods of the second group deal with profiling user behavior. In other words, they define a certain model of a normal user activity. Any deviation from this model is regarded as anomalous. DoSID uses the anomaly detection method.

Also IDSs are classified according to the kind of input information they analyze. These classes are: “application-based" IDS, "host-based" IDS and "network-based" IDS. This classification will be detailed in the coming section. DoSID is a network-based IDS.

2.1 Intrusion detection system hierarchy

The types of data examined by a particular IDS may vary significantly. IDS can be classified into one of the following categories based on the types of data they examine [2]:

- **Application-based**
  An application-based IDS examines the behaviour of an application program, generally in the form of log files.
- **Host-based**
  A host-based IDS examines data such as log files, processes accounting information, user behaviour, or outputs from application-based IDSs operating on the host. This type of system is limited in scope since it is only able to see its own host's environment, and cannot detect simultaneous attacks against multiple hosts.
- **Network-based**
  A network-based IDS examines network traffic. This type of IDS is a dedicated computer, or special-purpose hardware, with detection software installed. It is placed at a strategic point on a network (like a gateway or sub network) to analyze all network traffic on that particular segment. It scans data traffic for attacks. Also, it determines Internet Protocol (IP) addresses that originate outside its subnet.

2.2 Efficiency of an intrusion detection system

The following three measures have been used commonly to evaluate the efficiency of an IDS [3]:

- **Accuracy**
  Accuracy deals with the proper detection of attacks and the absence of false alarms. Inaccuracy occurs when an IDS flags a legitimate action in the environment as anomalous or intrusive.
- **Performance**
  The performance of an IDS is measured by the rate at which audit events are processed. If the performance of the IDS is poor, real-time detection is not possible.
- **Completeness**
  Completeness is the property of an IDS to detect all attacks. Incompleteness occurs when the IDS fails to detect an attack. This measure is much more difficult to evaluate than the others in real time because, for example, it is

impossible to have a global knowledge about every single type of attack or abuse of privileges.

2.3 Neural Network Concepts

A Neural Network is a structure which is composed of a number of simple elements or nodes called neurons as shown in Fig. 1. These elements are always operating in parallel. The function of the Neural Network is determined largely by the connection between the neurons. These neurons are connected by links and each link is adjusted by values called weights. The process of updating the weights is called learning.

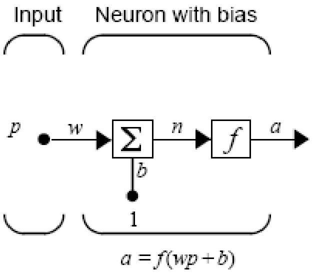

Fig. 1. Simple neuron.

Neuron showed in Fig. 1 is composed of: input *p* associated with weight *w* and there is a scalar bias *b*. The equation ***n=wp+b*** forms an input to the second main component which is the *transfer function*. The output of the neuron is the output of the transfer function.

The general equation is

$$\boldsymbol{a = f\,(wp+b)}$$

Here *f* is a transfer function which takes the argument *n* and produces the output *a*. The Neural Network will exhibit the desired or interested behavior by adjusting its parameters. That means, the Neural Network can be trained to a particular job by adjusting the weight or bias parameters or perhaps the network itself will adjust these parameter to achieve some desired results.

The input p to the neuron can be expanded to R-elements input and each input is multiplied by weight. Their sum is simply (W●P) which is the dot product of the matrix W and the vector P. Fig. 2 shows the neuron with the vector input. The argument n which is the input to the neuron transfer function will be:

$$\mathbf{n = w_{1,1}\, p_1 + w_{1,2}\, p_2 + w_{1,3}\, p_3 + \ldots + w_{1,R}\, p_R + b}$$

A one-layer network with R input vector and S neurons is shown in Fig. 3.

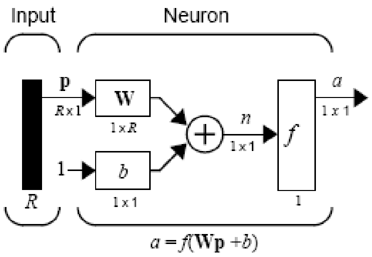

Fig. 2. Single neuron with input vector.

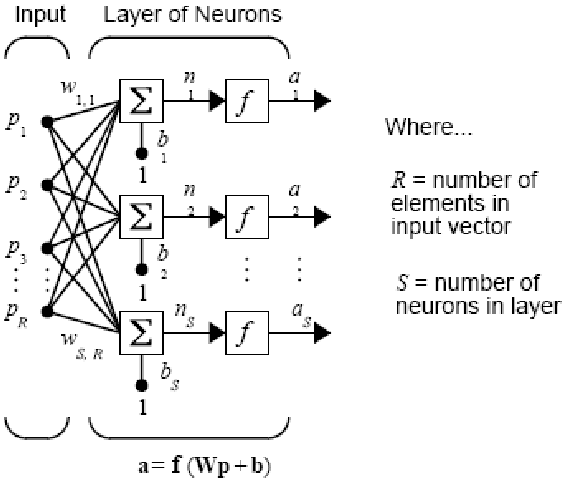

Fig. 3. Layer of neurons.

In this Neural Network, each input is connected to each neuron through the weight matrix W.

$$\mathbf{W} = \begin{bmatrix} w_{1,1} & w_{1,2} & \cdots & w_{1,R} \\ w_{2,1} & w_{2,2} & \cdots & w_{2,R} \\ & & & \\ w_{S,1} & w_{S,2} & \cdots & w_{S,R} \end{bmatrix}$$

The row indices indicate the destination neuron and the column indices indicate the source input.

One of the most commonly used Neural Networks is the multilayer feed-forward network. It falls under the category called "Networks for Classification and Prediction". The DoSID is built using this specific type of Neural Network.

Feed-forward networks usually consist of two to three layers in which the neurons are logically arranged. The last layer is the output layer and there are usually one or more hidden layers before the output layer. The DoSID Neural Network as shown in Fig.4 is composed of two layers (the hidden and the output layer), a variable number of neurons in the hidden layer and there is one neuron in the output layer. Each output vector element value is in the range [-1,1]. The transfer functions of neurons on both layers are "tan-sigmoid" function. This function takes the input, which may have any value between plus and minus infinity, and squashes the output into the range [-1,1]. The input vector contains 31 elements. These elements are the result of converting the 18 features in the DARPA dataset to Neural Network format.

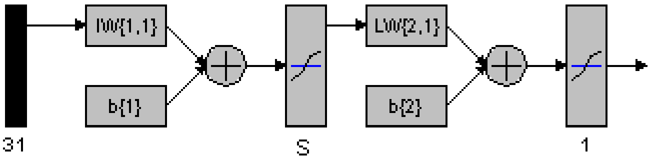

Fig. 4. Neural network architecture.

The most common and widely used learning algorithm for multilayer feed forward Neural Networks is the backpropagation algorithm. It is based on the *Delta Rule* that basically states that if the difference (delta error) between the user's desired output (target) and the network's actual output is to be minimized, the weights must be continually modified. The result of the transfer function changes the delta error in the output layer. The error in the output layer has been adjusted, and therefore it can be used to change the input connection weights so that the desired output may be achieved. This is why feed-forward networks are also often called "backpropagation feed-forward networks". The learning mechanism is illustrated in Fig. 5.

The input connection weights are adjusted in such a way that the delta error will be minimized. This process is repeated several times (Iterations). The training stops if: the number of iterations exceeds a certain number of iterations, the training performance function drops below certain threshold of MSE, or the training time is longer than certain threshold of seconds. The mean squared error (MSE) is computed by "summing the squared differences between the target and the network's actual output, and then dividing the sum by the number of components (input vector elements) that went into the sum."

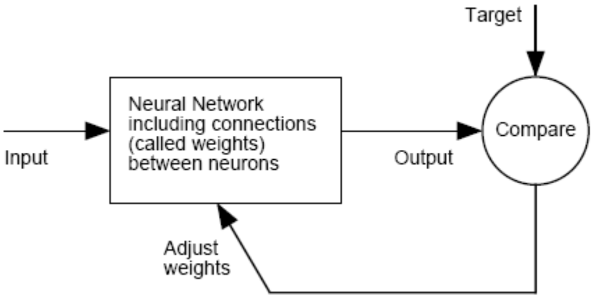
Target
Neural Network including connections (called weights) between neurons
Input
Output
Compare
Adjust weights

Fig. 5. Neural network learning mechanism.

3. Related Research

Since 1995, there are many researches that are based strongly on Neural Networks approaches to build various structures of IDS for either anomaly detection or misuse detection. Through this section, brief summaries of several researches and projects that apply Neural Networks approaches to IDS are given.

One of the recent research projects that focus on Multi Layer Perceptron (MLP) and statistical analysis engine is the work done by Jorgenson, Manikopoulos, Li, and Zhang [4]. It is a network-based IDS which consists of several tiers, with each tier containing Intrusion Detection Agents (IDAs). IDAs are IDS components that monitor the activities of a host or a network (See Figure 1). Each IDA consists of:

- **Probe**
  Collect the network traffic of a host or a network, abstract the traffic into a set of statistical variables and generate reports to the event preprocessor.
- **Event preprocessor**
  Convert the information into the format required by the statistical processor.
- **Statistical processor**
  Compare the data to reference models previously compiled describing the normal state of the system. Then, it forms the stimulus vector reports which are forwarded to the Neural Network.
- **Neural network**
  Analyzes the vector and decides if it is anomalous or normal.

One of the earliest attempts to apply Neural Networks in IDS is the work done by Cannady [5]. He used a network-based IDS responsible for monitoring and collecting information from the network packets. This information is then forwarded to the MLP Neural Network for analysis. The project's main idea is to make the data go through three levels of preprocessing, each level extracting certain features of the packet. Then, normalize and group them and convert the result to the Neural Network format in order to indicate whether it is an attack.

One of the host-based IDSs produced in 1998 which monitors the applications at process level is the result of work done by Charon, Ghosh and Wanken [6]. Their project is based on anomaly detection method. The process states and input/output combinations were used as input to the MLP Neural Network. A training set was generated using simulated normal data input and unknown attacks to the monitored application.

5. Denial of Service Intelligent Detection System Description

The DoSID is described in details in this section. A feed forward Neural Network is used. Also, the datasets from the 1998 DARPA Intrusion Detection Evaluation for training and testing our system are used.

4.1 Evaluation dataset

The 1998 DARPA Intrusion Detection Evaluation Program [7] was prepared and managed by MIT Lincoln Labs. The objective was to survey and evaluate researches in IDSs. A standard dataset to be audited was provided and called "DARPA dataset". This dataset includes a wide variety of intrusions simulated in a military network environment. Lincoln Labs set up an environment to acquire nine weeks of raw TCP dump data from a local area network (LAN) simulating a typical U.S. Air Force LAN. They operated the LAN as if it were a true Air Force environment, but peppered it with multiple attacks.

DARPA dataset is separated into two categories. These are: testing dataset and training dataset. The raw testing dataset was TCP dump data from two weeks of network traffic. This was processed into about two million connection records. These connection records are not labeled.

The raw training dataset was about four gigabytes of compressed binary TCP dump data from seven weeks of network traffic. This was processed into about five million connection records. Each connection record is labeled as either normal, or as an attack, with exactly one specific attack type. In the training dataset there are 22 different attack types.

In the training dataset, the attacks fall into four main categories: denial-of-service (DoS), unauthorized access from a remote machine (R2L), unauthorized access to local super user (root) privileges (U2R), and surveillance and other probing (Probing).In this paper, the focus is on DoS attacks. There are six DoS attacks: Back, Land, Neptune, Pod, Smurf and Teardrop.

Each record in DARPA training dataset is a connection called connection record. Each connection record consists of 41 features and a label. The label indicates either normal, or an attack, with exactly one specific attack name. Some of these features are immediately getting data from raw TCP dump data, and others required some statistical operations to get them. Out of 41 features, 18 features that are useful to detect DoS attack are used. So, each connection is filtered to contain only these features.

4.2 DoSID framework

The DoSID framework as shown in Fig. 7 reflects the sequence of input set transformation of connection records.

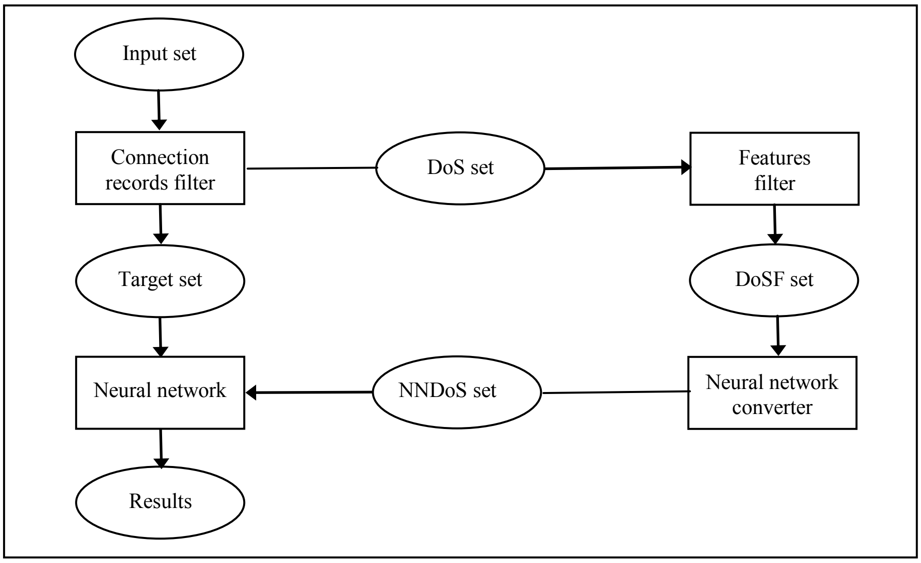

Fig. 2. DoSID architecture.

DoSID is dedicated to DoS attacks. Therefore, the types of connection records needed in our experiments are only normal traffic and any DoS attacks. The role of "Connection Records Filter" module is to filter the "Input set" to contain only normal and DoS attacks. The filtered set is called "DoS Set". For each connection record in

"DoS Set", the "Connection Record Filter" module prints in separate set the class of that record. It prints either 1 for normal or -1 for DoS attacks in separate line. This set is called "Target set".

Each connection record contains 41 features. Only 18 features that are useful for detecting DoS attacks are used. "Features Filter" module extracts the needed features (18 features) from each connection record in "DoS Set" and stores this record in "DoSF Set". The "DoSF Set" is converted to Neural Network format to be readable by the Neural Network and this is the role of "Neural Network Converter" module. The result of this module will be in “NNDoS Set”. The last set is used as input to “Neural Network” module. As described in the previous section, this module is a Neural Network composed of two layers (the hidden and the output layer), with a variable number of neurons in the hidden layer and one neuron in the output layer as depicted in Fig. 4.

Two improvements to DoSID are added by developing and implementing different techniques in order to enhance detection accuracy, decision making, and Neural Network performance (MSE) for testing phase.

4.3 Improvement 1: gray area

Neural Network predicts the type of each connection record. Its output is a vector that consists of one element which falls in the range [-1, 1]. The connection record is classified as normal when the vector value is around 1, where values around -1 indicate a DoS attack connection record.

Wherever the output value is closer to 1 or -1, the Neural Network decision becomes more accurate. The further the value from 1 and -1 toward 0 indicates non-accurate decision. Therefore a Gray Area concept is proposed to improve the accuracy of Neural Network as depicted Fig. 7.

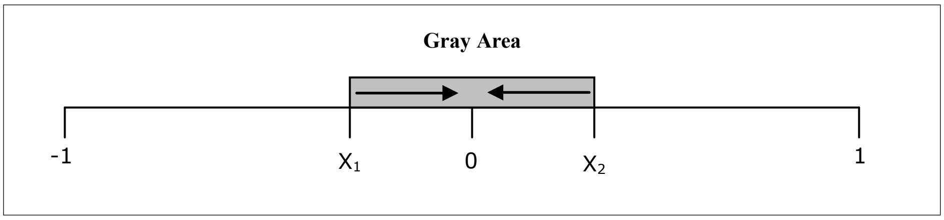

Fig. 3. Gray area concept.

The Gray Area is an area inside the range of the output value. The value that gets in this area [x1, x2] is not accurate because it is far from 1 or -1. So, the value is changed to zero which means that connection record is unrecognized.

The most critical issue in Gray Area concept is its boundaries x1 and x2. The values of boundaries are selected based on the desired objective of the gray area. For example, in a strict environment where any possible intrusion is to be reported regardless of the high false positive warnings, the value of x1 is increased.

The size of Gray Area depends on overall Neural Network results or decisions for each connection record in the training set. In order to specify the boundaries of the Gray Area, the Distribution concept is introduced.

4.4 Improvement 2: distribution

In experiment two, the Neural Network gave highly accurate decisions for connection records that were used in the training phase. The output value of each connection record during the training phase is distributed over the range of output value.

To do this distribution, first, the range is divided into small intervals. The length of each interval is 0.1. A counter is assigned for each interval to count the number of connection records that the corresponding Neural Network vector output value lies in.

The Gray Area will include each interval that has small counter value. For example, in one of the experiments, the distribution of 9979 connection records is as in Table 1. So, the range of the Gray Area will be [-0.8, 0.9)

Table 1. Distribution training data with 24 neurons and 1000 iterations

Intervals	[-1,-0.9)	[-0.9,-0.8)	[-0.8,-0.7)	[-0.7,-0.6)	[-0.6,-0.5)
Records No.	5974	6	2	0	0
Intervals	[-0.5,-0.4)	[-0.4,-0.3)	[-0.3,-0.2)	[-0.2,-0.1)	[-0.1,0)
Records No.	0	0	0	2	0
Intervals	[0,0.1)	[0.1,0.2)	[0.2,0.3)	[0.3,0.4)	[0.4,0.5)
Records No.	0	0	0	0	0
Intervals	[0.5,0.6)	[0.6,0.7)	[0.7,0.8)	[0.8,0.9)	[0.9,1]
Records No.	0	2	0	1	3999

5. Experimental Analysis

In this section, experiments are conducted and their results are presented along with the different improvements proposed in the previous section.

5.1 Introduction

For training and testing the Neural Network, connection records are collected randomly from the DARPA training dataset. Therefore, all connection records are labeled. They are used in the training phase to inform the Neural Network about the type of the connection (attack or normal), so the Neural Network will learn from each connection record (i.e. by adjusting its weights).

On the other hand, the labeled connection records are useful in the testing phase for measuring the system accuracy. This is done by feeding each connection record to the Neural Network. The normal and attack labeled records are fed separately. By computing the average output, the accuracy of the Neural Network decision is obtained.

In order to test the Neural Network against known and unknown attacks, it is trained with specific attacks. Some other attacks are left for testing unknown attacks. The six DoS attacks are categorized as follows:

1. **Training Attacks**
   Four attacks are used for training the Neural Network. These attacks are Back, Neptun, Smurf and Teardrop. The labeled connection records using these attacks are used to train the Neural Network.
2. **Testing Attacks**
   Two training attacks are used for testing the Neural Network. These attacks are Back and Smurf, which have been recognized by the Neural Network in the training phase. These are called *Known Attacks*. Also, there are two attacks which are not seen by the Neural Network in the training phase. These attacks are Land and Pod which are called *unknown attacks*.

In the following sections, the training performance (MSE) of various Neural Networks that have been trained using the training set is shown. Also the results of testing the Neural Network that has the best training performance are shown.

5.2 Training experiments

In the experiments conducted, various Neural Networks are trained using the training set. This set contains about 10000 connection records. 4000 connection records are labeled with Normal and 6000 connection records are labeled with one of training attacks namely Back, Neptun, Smurf or Teardrop.

In the training phase, diverse methods are used to train the Neural Networks in order to achieve good performance. Actually, there are many factors affecting the Neural Network. Some of the factors that are considered as the most significant factors affecting the neural network decision making are:

- **Number of neurons per layer**
  The Neural Networks is built using one hidden layer that contains 24 or 64 neurons.
- **Number of iterations (epochs)**

Each Neural Network is trained twice, using 1000 iterations and 5000 iterations.

- **The initial weights and bias**
  The initial weights of each Neural Network training session are selected based on the following methods:
  i. **Zeros initial:** The initial weights are zeros.
  ii. **Training initial:** The initial weights are the resultant weights from a Neural Network that has been trained using 200,000 iterations.
  iii. **Random initial:** The initial weights are generated randomly.

To show the effects of these factors, the experiments are conducted using all combinations. the training performance is measured using the mean square error (MSE). As mentioned before, the MSE is the difference between the target and the Neural Network's actual output. So, the best MSE is the closest to 0. If MSE is 0, this indicates Neural Network's output is equal to the target which is the best situation.

5.2.1 experimental results of training phase

Table 2 shows the training performance (MSE) resulted from training some Neural Networks using the training set.

Table 2. The MSE for training sessions using the training set

	Neural Network with 24 neuron		Neural Network with 64 neuron	
	1000 iterations	**5000 iterations**	**1000 iterations**	**5000 iterations**
Zeros initial	0.0021588	0.0016041	0.0019362	0.0015045
Training initial	0.0001617	**0.00015721**	0.00041351	0.00041306
Random initial	0.0019256	0.0016443	0.00306	0.00050973

It has been noticed that, the MSE for training using Training Initial Weights is better than with Zeros or Random Initial Weights. This is attributed to the fact that the training initial weights are generated using 200.000 iterations.

The shaded MSE in table 2 is for a Neural Network with 24 neurons and has been trained using 5000 iterations. This Neural Network has the best MSE. This particular Neural Network is used in the testing phase. The next section shows the results of the testing phase.

5.3 Testing experiments

To show the accuracy of the Neural Network decisions with each type of connection records, connection records are collected randomly from DARPA training dataset where these records did not exist in the training set. The connection records are divided into three separate sets. Table 3 lists these sets.

Table 3. Testing phase data sets

Set Name	Connection records	Possible label(s)
Normal Set	70 records	normal
Known Set	60 records	back, smurf
Unknown Set	50 records	land, pod

A Neural Network that has one hidden layer which contains 24 neurons is tested. This Neural Network has been trained using the training set, training initial weights and 5000 iterations. The training performance was 0.00015721.

The average output of the Neural Network is used as measurement of the Neural Network accuracy. In case of using the Normal set of connection records, the best average will be 1. If Known set or Unknown set are used, the best average will be -1.

To show the effect of Gray Area and Distributions improvements, the Neural Network is tested two times. One test is conducted without applying the Gray Area and the second is done with the application of Gray Area. The next sections show the results.

5.3.1 experimental results for regular testing

The Neural Network is tested without using the Gray Area concept. The Neural Network decision for each connection record must be normal or attack. There are no undefined connection records because there is no gray area to indicate uncertainty. Any connection record is considered normal if its output lies in the positive side. If the connection record output lies in the negative side, it will be considered an attack as depicted in Fig. 9.

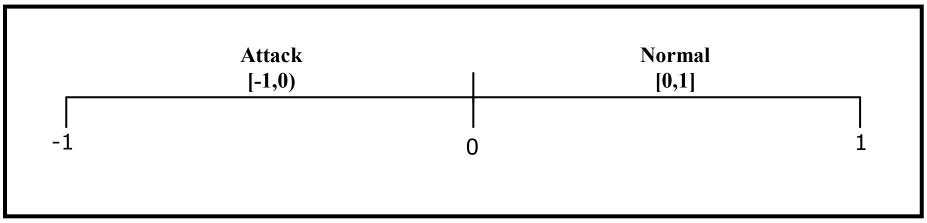

Fig. 9. The ranges of Attack and Normal decisions without Gray Area.

The results of testing the Neural Network using the testing sets are shown in Table 4.

Table 4. Results for testing a neural network with 24 neurons and 5000 iterations without gray area

Without Gray Area	Detection Rate	False Alarm		Connection records		Average	MSE
		False Positive	False Negative	DoS	Normal		
Normal (70)	91.42%	8.57%		6	64	0.835604	0.3103

Known (60)	100%		0%	60	0	-0.9936	7.47E-05
Unknown (50)	60%		40%	30	20	-0.38181	0.91189

As shown in Table 4, the neural network correctly detected 100% of the known attacks. Testing the normal traffic, the false positive indicator is low which is less than 9%. It has been noticed that the Neural Network can detect 60% of the attacks that it did not see in the training phase, which proves that the Neural Network can detect new attacks, but the false negative indicator is still high.

5.3.2 experimental results for testing with gray area

To see the effect of Gray Area, the Neural Network is tested again using the same testing sets that are used in the previous tests but using the Gray Area. The Distribution is used to determine the boundaries of gray area. Table 5 shows the distribution of simulation of the training set using the Neural Network under testing.

Table 5. The distribution of simulation of the training set

Intervals	[-1,-0.9)	[-0.9,-0.8)	[-0.8,-0.7)	[-0.7,-0.6)	[-0.6,-0.5)
Records No.	5975	4	0	0	0
Intervals	[-0.5,-0.4)	[-0.4,-0.3)	[-0.3,-0.2)	[-0.2,-0.1)	[-0.1,0)
Records No.	0	0	0	0	0
Intervals	[0,0.1)	[0.1,0.2)	[0.2,0.3)	[0.3,0.4)	[0.4,0.5)
Records No.	1	0	0	0	0
Intervals	[0.5,0.6)	[0.6,0.7)	[0.7,0.8)	[0.8,0.9)	[0.9,1]
Records No.	0	0	0	2	3997

From table 5, the density of records exists in intervals: [-1,0.8) which means that 5979 records are attacks, and [0.8,1] which means that 3999 records are normal. From this distribution, the range of the gray area is **[-0.8,0.8)**. The connection records lying in this range as “Unrecognized” records are considered. The connection records lying in the positive side are “Normal” and in the negative side are “Attack”. (See Fig. 10)

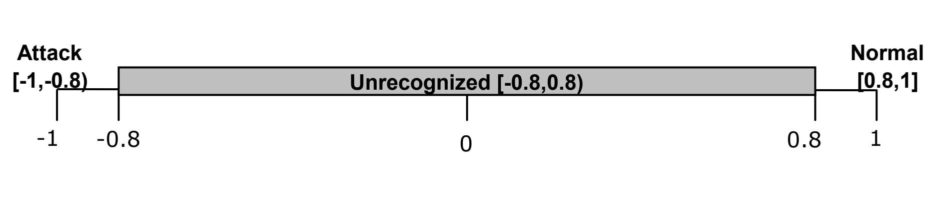

Fig. 6. The ranges of attack, normal and unrecognized decisions with gray area.

Table 6 shows the result of testing the Neural Network using the Gray Area and testing sets.

Table 6. Results for testing a neural network with 64 neurons and 5000 iterations with gray area

With Gray Area [-0.9,0.9)	Detection Rate	False Alarm		Connection records			Average	MSE
		False Positive	False Negative	DoS	Normal	Unrecognized		
Normal (70)	91.42%	5.71%		4	64	2	0.845042	0.285189
Known (60)	100%		0%	60	0	0	-0.9936	7.47E-05
Unknown (50)	58%		8%	29	4	17	-0.5074	0.608

By applying the Gray Area concept, there is considerable improvement in the results in two aspects. First, it minimizes the false negative indicator from 40% to 8%. Second, it shows the high level of accuracy of the Neural Network's decision where most of the output fell very close to 1 in case of Normal connection records and very close to -1 in case of Attack connection records.

6. Conclusion

The Neural Networks provide a number of advantages in the detection of new attacks. In this paper, the DoSID system as a network-based IDS is introduced using Neural Network to detect Denial of Service attacks. The training dataset from DARPA is used to train and test our Neural Network.

The ability of a feed-forward Neural Network is tested to classify normal traffic correctly and to detect attacks. It has been found that the Neural Network detects the known attacks which have been used in the training of the Neural Network. Also, it has been found that the Neural Network can detect unknown attacks which have never been used in the training phase. These results mean that the Neural Networks are a significant technique to detect new attacks.

The Gray Area improvement is proposed which uses the distribution concept to determine the boundaries of Gray Area. The experiment using the gray area resulted in improving the false negative indicator from 40% to only 8%, and it increased the accuracy of Neural Network decisions.

In the experiments, various Neural Networks are trained using different combinations of factors. Also it has been found that the training using initial weights resulting from a previous training has the best training performance (MSE). The Neural Network that has the best MSE is tested using three data sets, namely, Normal data set, Known data set, and Unknown data set. It has been shown that the Neural Network can detect the unknown attacks. These attacks are not seen by the Neural Network in the training phase.

References

[1] Verwoerd, T. and Hunt, R. “Intrusion Detection Techniques and Approaches”, Computer Communications, Vol. 25, No. 15, September 2002. pp. 1356- 1365.

[2] Julia, A., Christie, A., Fithen, W., McHugh, J., Pickel, J. and Stoner, E. “State of the Practice of Intrusion Detection Technologies (CMU/SEI-99/TR-028)”. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University, 2000.

[3] Debar, H., Dacier, M. and Wespi. A. “Towards a Taxonomy of Intrusion-Detection Systems”. Computer Networks, 31(8):805--822, April 1999.

[4] Zhang, Z., Li, J., Manikopoulos, C., Jorgenson, J. and Ucles, J. “A Hierarchical Anomaly Network Intrusion Detection System Using Neural Network Classification”, Proceedings of 2001 WSES International Conference on: Neural Networks and Applications (NNA’01), Feb. 2001

[5] Cannady, J. “Artificial Neural Networks for Misuse Detection.” Proceedings of the 1998 National Information Systems Security conference (NISSC'98) October 5-8 1998. Arlington, VA.

[6] Ghosh, A.K., Wanken, J. and Charron, F. “Detecting Anomalous and Unknown Intrusions Against Programs” . *Proceedings of the 1998 Annual Computer Security Applications Conference (ACSAC'98)*, December 1998.

[7] Hettich, S. and Bay, S. D. (1999). The UCI KDD Archive [http://kdd.ics.uci.edu]. Irvine, CA: University of California, Department of Information and Computer Science.

[8] Crosbie, M and Spafford, G. “Active Defense of a Computer System Using Autonomous Agents.” Technical Report 95-008, COAST Group, Department of Computer Sciences, Purdue University, West Lafayette, Indiana, February 1995.

[9] Axelsson S. "*Intrusion Detection Systems: A Survey and Taxonomy*". Technical report 99-15, Department of Computer Engineering, Chalmers University of Technology, Goteborg, Sweden, March 2000.

الاكتشاف الذكي لهجمات تعطيل الخدمة بإستخدام الشبكة العصبية

د. عبدالقادر بن عبدالله الفنتوخ

قسم علوم حاسب، كلية علوم الحاسب والمعلومات، جامعة الملك سعود
ص.ب. ٣٠١٣٤ الرياض، ١١٣٧٢ المملكة العربية السعودية
Fantookh@ksu.edu.sa

(قدّم للنشر في ٢٠٠٤/٠٨/١٥م؛ وقبل للنشر في ٢٠٠٥/٠٧/٠٤م)

ملخص البحث. ملخص البحث. في الآونة الأخيرة ازدادت حوادث الاعتداء على الشبكات وتعطيلها وذلك عائد لزيادة الاعتمادية عليها والرغبة في الترابط مع الشبكات الأخرى وشبكة الإنترنت. لقد أصبحت أنظمة اكتشاف الاختراق عنصر رئيس في منظومة الحماية بغض النظر عن قوة الحماية والدفاعات الخارجية. وكما هو معروف فإن معظم أنظمة كشف الاختراق للأجهزة والشبكات تعتمد على أنظمة الخبرة التي تحتوي على قواعد محددة لا تستطيع اكتشاف الهجمات الجديدة. بل إنها تتسم في الغالب في التسبب في إصدار نسبة كبيرة من البلاغات الكاذبة بهجمات، بالإضافة إلى إخفاقها في اكتشاف بعض الهجمات.

في هذا البحث تم تقديم تطبيق الشبكة العصبية كأحد مكونات نظام كشف الاختراقات الشبكية وبالتحديد لكشف هجمات تعطيل الخدمة. لقد تم بناء نظام شبكة عصبية لكشف الاختراقات المعروفة وغير المعروفة، ولقد تم استخدام خوارزمية الانتشار العكسي " Back Propagation Algorithm " التي تستخدم في تدريب الشبكات العصبية كاملة الارتباط وذات التغذية الأمامية "Feed Forward" ومتعددة الطبقات. في مرحلة تدريب الشبكة على الاختراقات

تم استخدام البيانات التي قدمها معمل لينكلن في معهد ماساشوستس للتقنية في المشروع الذي تم رعايته عن طريق وكالة داربا التابعة لوزارة الدفاع الأمريكية.

أثبتت التجارب التي أجريت على الشبكة العصبية أنها قادرة وبنسبة كبيرة على التمييز بين الاتصالات الاعتيادية والهجمات. كما تم تحسين الشبكة العصبية لتكون أكثر دقة. لقد قامت الشبكة العصبية باكتشاف ما نسبته ٦٠% من الهجمات غير المعروفة مسبقا. وبتقديم مفهوم المنطقة الرمادية تم إثبات أنه بالإمكان تقليل نسبة اعتبار بأن بعض الهجمات هي اتصالات اعتيادية من ٤٠% إلى ٨%.