International Journal of Computer Applications Technology and Research
Volume 3– Issue 9, 579 - 584, 2014, ISSN: 2319–8656

The Use of Intelligent Algorithms to Detect Attacks In Intrusion Detection System

Faezeh Mozneb khodaie
Department of computer,
Shabestar branch, Islamic Azad University, Shabestar, Iran

Mohammad Ali Jabraeil Jamali
Department of computer,
Shabestar branch, Islamic Azad University, Shabestar, Iran

Ali Farzan
Department of computer,
Shabestar branch, Islamic Azad University, Shabestar, Iran

Abstract: More networks are connected to the Internet every day, which increases the amount of valuable data and the number of resources that can be attacked. Some systems have been designed and developed to secure these data and prevent attacks on resources. Unfortunately, new attacks are being created everyday, which makes the design of system that could catch these attacks harder. The need is not only for preventing the attack, but also to detect such an attack, if it happens. Intrusion Detection Systems is built to accomplish this task and complement other security systems. In this paper we build an Intrusion Detection System using Artificial neural networks (ANN) and Self-Organizing Map (SOM).

Keywords: intrusion detection systems; attacks; system security; artificial neural network; self-organizing map

1. INTRODUCTION

Heavy reliance on the internet and worldwide connectivity has greatly increased the potential damage that can be inflicted by remote attacks launched over the internet. It is difficult to prevent such attacks by security policies, firewalls, or other mechanisms because system and application software always contains unknown weaknesses or bugs, and because complex, often unforeseen, interactions between software components and/or network protocols are continually exploited by attackers. Intrusion detection systems are designed to detect attacks which inevitably occur despite security precautions [1]. Intrusion Detection Systems (IDS) is a piece of software or hardware that captures the inbound and outbound traffic, and analyzes it, in order to detect unusual flows. After detecting the abnormal flows, it notifies the system or the network administrator to take the appropriate action. IDS detects that a security breach happened, while firewall protects the system from security breaches. Hence they complement each other and should be used together [2].
The first concept of the IDS was introduced in 1980 by Anderson James P. [3]. In 1984 Fred Cohen mentioned that the percentage of detecting an attack will increase as the traffic increases [4]. Dorothy E. Denning introduced a model of IDS in 1986, which becomes the basic model of the current IDS models [5].

2. SECURITY OF COMPUTER SYSTEMS

Nowadays computer and Internet systems are used in almost all aspects of our lives. With the advent of personal computers and the growth of its use, Today all companies, universities and even small stores customer information, purchasing and sales and store in a computer database. One of the facilities, computer systems, computer networking systems is to establish a resource sharing among users. With the ability to connect multiple computers together, and create a computer network, protecting it from invaders came all this information and the machines. This information is critical for people trying to win others to use, alter, or destroy it.

Various measures to protect companies and home users computer resources available, But if you follow all the recommendations of the experts, the system will never be safe from attack. For this reason, users or the security of an organization, you should know the value of their and Risk analysis on it do to protect it [6].

A good security policy with a proper risk analysis by experts, the system is more resistant to the influence of many. Security is defined by three basic principles:

- Confidentiality: the attacker does not have access to confidential information.
- Integrity: Information may be altered or destroyed by the invaders.
- Access control: The system may be blocked so that it can not be normal.

One of the three major attempt to disturb the security of computer systems is called diffusion.

3. INTRUSION DETECTION SYSTEM

In order to combat computer systems and networks against hackers, Several methods have been established as a method for intrusion detection that The practice of monitoring the events occurring in a computer system or network play.
In general, an intrusion detection system to monitor the activities of the environment in which it operates and Eliminates unnecessary data from the data obtained, Usually a series of features to be extracted from the data collected, Then after assessment activities, the probability of an attack is considered that this procedure is done by recognizer. After identifying a suitable response system against invasions usually diagnosed offers. Most intrusion detection systems only detect attacks, and to warn the Nmayndv usually no

preventive action is not the issue. The most important part of an intrusion detection system, detection is the main task is to check the data collected. Figure 1. An overview of Intrusion Detection System based on the definitions provided by the show.

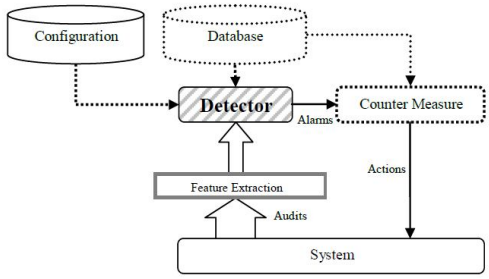

Figure. 1 Intrusion detection system

3.1 Types of IDS

There are two main types of IDS:

1) Host-based Intrusion Detection System (HIDS*):* HIDS is one of the first IDS types that were developed. Its main job is to monitor the information that flows to a computer by collecting the information that goes through and analyze it. Because of the nature of the HIDS, it has the ability to detect which process in the host computer is being under attack. This is its main advantage over other types.
2) Network-based Intrusion Detection System (NIDS):
Using the NIDS is more economical, which make it useful than any other types. The NIDS collects the packet that flows through the network to the different hosts of the network, then analyzes all the collected information and sends the results to a central system, in order to detect a possible attack. This is done by using different single purpose sensors that are placed in various points of the network [7].

3.2 Intrusion Detection Techniques

There are two basic techniques to detect an intruder, namely anomaly detection and misuse detection [2].

3.2.1 Anomaly Detection:

This technique has been developed to detect abnormal operations. It works by registering every activity in the system in a profile for hosts or network connection. If there is a sudden change in the profile, it will be treated as an abnormal activity. For example, if a normal user usually logs on to his account 2 times a day then, if in any one day he logs 20 times, the system will treat this as an abnormal and considers it as an attack.

3.2.2 Misuse Detection:

Misuse detection is also known as signature detection. It discovers any attempt to breach the Not every misuse is an attack, because some of them are just mistakes that were done by authorized ends, but every unauthorized attempt has to be taken seriously. Depending on the robustness and seriousness of a signature, some alarm, response, or notification should be sent to the proper authorities.

3.3 Types of network attacks

There are three main kinds of attacks that could be detected by the IDS: system scanning, denial of service (DoS) and system penetration. These attacks could be executed on the local machine or could be executed from a different remote machine. Every kind of these attacks should be treated differently.
1) Scanning Attacks: Before performing an attack, the attacker may search for a week point to use for attacking the system. This is performed by releasing a number of packets to some specific hosts, until vulnerable ports are discovered.
2) Denial of Service Attacks : Denial of Service (DoS) attacks is used to shut down a service that is being provided by a specific server, or to slow down the host network connection. This is done by sending infinite number of requests to the target host, until it will reach its limit and shut down.
3) Penetration Attacks: Penetration attacks target the system privileges, data and resources to alter them by an unauthorized party. This attacker could gain access to huge amount of information on the host machine and this makes it more dangerous than other attacks.

4. DATASETS KDD'99

Complex relationships exist between features, which are difficult for humans to discover. The IDS must therefore reduce the amount of data to be processed. This is very important if real-time detection is desired. The easiest way to do this is by doing an intelligent input feature selection. Certain features may contain false correlations, which hinder the process of detecting intrusions. Further, some features maybe redundant since the information they add is contained in other features. Extra features can increase computation time, and can impact the accuracy of IDS. Feature selection improves classification by searching for the subset of features, which best classifies the training data.

Feature selection is done based on the contribution the input variables made to the construction of the decision tree. Feature importance is determined by the role of each input variable either as a main splitter or as a surrogate. Surrogate splitters are defined as back-up rules that closely mimic the action of primary splitting rules. Suppose that, in a given model, the algorithm splits data according to variable 'protocol_type' and if a value for 'protocol_type' is not available, the algorithm might substitute 'flag' as a good surrogate. Variable importance, for a particular variable is the sum across all nodes in the tree of the improvement scores that the predictor has when it acts as a primary or surrogate (but not competitor) splitter.

The data for our experiments was prepared by the 1998 DARPA intrusion detection evaluation program by MIT Lincoln Labs MIT. The LAN was operated in a real environment, but was subjected to multiple attacks. For each TCP/IP connection, 41 various quantitative and qualitative features were extracted. The data set has 41 attributes for each connection record plus one class label. The data set contains 24 attack types that could be classified into four main categories.
1) DoS: Denial of service
Denial of service (DoS) is a class of attack where an attacker makes a computing or memory resource too busy or too full to handle legitimate requests, thus denying legitimate users access to a machine.

2) R2L: unauthorized access from a remote machine
A remote to user (R2L) attack is a class of attack where an attacker sends packets to a machine over a network, then exploits the machine's vulnerability to illegally gain local access as a user.
3) U2Su: unauthorized access to local super user (root)
User to root (U2Su) exploits are a class of attacks where an attacker starts out with access to a normal user account on the system and is able to exploit vulnerability to gain root access to the system.
Probing: surveillance and other probing
4) Probing is a class of attack where an attacker scans a network to gather information or find known vulnerabilities. An attacker with a map of machines and services that are available on a network can use the information to look for exploits. Table 1. attacks in the KDD dataset based on each of the four classes above shows.

Table 1. Attacks in the KDD dataset

DOS	back, land, neptune, pod, smurf, teardrop
U2R	buffer_overflow, loadmodule, multihope, perl, rootkit
R2L	fip_write, guess_password, imap, phf, spy, warezclient, warezmaster
PROBE	ipsweep, nmap, portsweep, satan

5. NEURAL NETWIRKS

Human brain, composed of many elements, is capable of processing very elaborate and complex tasks. The brain contains billions of neurons, which are basically regarded as the most essential brain processing units. The information process is achieved by exchange of electrical pulses between these units. Neurons process information in parallel and they are connected through synaptic weights to each input in order to generate an output.
Synaptic weight refers to the significance of the established connection between an input value and a neuron. Since neurons process information in a distributed way it is possible to achieve high processing rates [8].
Neural networks terminology refers to the cluster of neurons that function or act together to solve a particular task and process information. These networks are also capable of learning through supervision or independently. Artificial neural networks (ANN) as processing models are inspired by the way nervous system work and they attempt to implement in computer systems neuron like capabilities. Three layers are present in a typical ANN: input layer, hidden layer and output layer. Each layer is composed of one or more nodes (neurons) and communication paths between them [9]. All layers connected together form a network of nodes (or neurons). Typically information flows from the input to the output layer, although in some ANN architectures a feedback flow is present. The input layer represents the stimulus or information forwarded to the network, while the output layer is the final product of the neural processing. Input layer nodes often carry out hidden relationships amongst them producing "hidden" nodes. The hidden nodes and the interaction weight between input nodes compose the hidden layer. Figure 2. shows the neural netwok layers.

The performance of neural networks depends on the architecture, algorithms and learning model chosen to collect and process data.

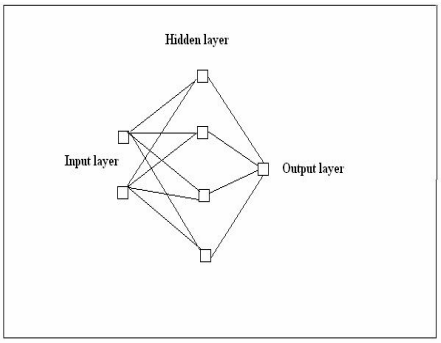

Figure. 2 ANN layers

Neural networks main features:

a) Architecture: Layer feed forward, multiple layer feed-forward, recurrent etc Single layer networks have only one layer of neurons connected individually to input points while multiple layers usually have several layers of neurons to process the data. In a single feed forward network the information move forward from input layer to output layer without backward feedback. Multiple layer models use algorithms such as back propagation to learn; output values are compared with the result values in order to correct errors. The acquired information is then forwarded back to the network for self correction. Recurrent networks use multiple layers and back propagation for learning [10].

b) Learning algorithms: There is a variety of algorithms used for learning including: error correction learning, Hebbian learning, competitive learning, self organizing maps, back propagation, snap-drift algorithm neocognition, feature map, competitive learning, adaptive resonance theory, principal component, perceptron, decision-based, multilayer perceptron, temporal dynamic model, hidden Markov model, Hamming net, Hopfield net, combinatorial optimization etc. Snapdrift in particular, performs well in frequently changing environments because of its ability to alter between minimalist learning when network performance is down and cautious learning when performance is up [11].

c) Learning model: Supervised or unsupervised. Supervised models have been the mainstream of neural development for some time. The training data consist of many pairs of input/output training patterns and the learning process relies on assistance (Kung,1993).While in the learning phase the neural network learn the desired output for a given input .Multiple layer perceptron (MLP) algorithm is used often with supervised models. In the case of unsupervised models, the network gain knowledge without specifying the required output during the learning phase. The self-organizing map (SOM) algorithm is associated frequently with unsupervised models [12].

6. SELF ORGANIZING MAP

The Self-Organizing Map is a neural network model for analyzing and visualizing high dimensional data. It belongs to the category of competitive learning network.The SOM figure 2. defines a mapping from high dimensional input data space onto a regular twodimensional array of neurons. It is a

competitive network where the goal is to transform an input data set of arbitrary dimension to a one- or two-dimensional topological map. The model was first described by the Finnish professor Teuvo Kohonen and is thus sometimes referred to as a Kohonen Map. The SOM aims to discover underlying structure, e.g. feature map, of the input data set by building a topology preserving map which describes neighborhood relations of the points in the data set [13].

The SOM is often used in the fields of data compression and pattern recognition. There are also some commercial intrusion detection products that use SOM to discover anomaly traffic in networks by classifying traffic into categories. The structure of the SOM is a single feed forward network, where each source node of the input layer is connected to all output neurons. The number of the input dimensions is usually higher than the output dimension.

The neurons of the Kohonen layer in the SOM are organized into a grid, see figure 3. and are in a space separate from the input space. The algorithm tries to find clusters such that two neighboring clusters in the grid have codebook vectors close to each other in the input space. Another way to look at this is that related data in the input data set are grouped in clusters in the grid. The training utilizes competitive learning, meaning that neuron with weight vector that is most similar to the input vector is adjusted towards the input vector.The neuron is said to be the 'winning neuron' or the Best Matching Unit (BMU). The weights of the neurons close to the winning neuron are also adjusted but the magnitude of the change depends on the physical distance from the winning neuron and it is also decreased with the time.

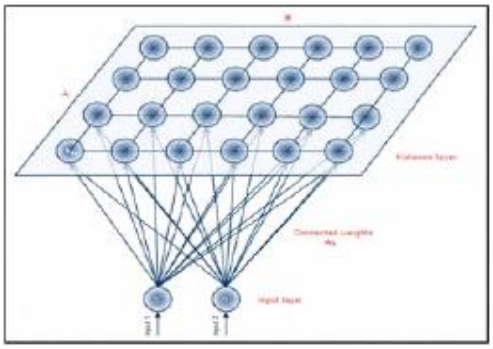

Figure. 3 Self-Organizing (Kohonen) Map

The learning process of the SaM goes as follows:

1) One sample vector x is randomly drawn from the input data set and its similarity (distance) to the codebook vectors is computed by using Euclidean distance measure [14]:

$$\| x - m_c \| = \min_i \{ \| x - m_i \| \} \quad (1)$$

2) After the BMU has been found, the codebook vectors are updated. The BMU itself as well as its topological neighbors are moved closer to the input vector in the input space l.e. the input vector attracts them. The magnitude of the attraction is governed by the learning rate. As the learning proceeds and new input vectors are given to the map, the learning rate gradually decreases to zero according to the specified learning rate function type. Along with the learning rate, the neighborhood radius decreases as well. The update rule for the reference vector of unit i is the following:

$$m_i(t+1) = m_i + a(t) h_{ci}(r(t))[x(t) - m_i(t)] \quad (2)$$

3) The steps 1 and 2 together constitute a single training step and they are repeated until the training ends. The number of training steps must be fixed prior to trainingthe SaM because the rate of convergence in the neighborhood function and the learning rate are calculated accordingly.

After the training is over, the map should be topologically ordered. This means that *n* topologically close input data vectors map to *n* adjacent map neurons or even to the same single neuron.

5.1 Mapping Precision

The mapping precision measure describes how
accurately the neurons respond to the given data set. If the reference vector of the BMU calculated for a given testing vector xi is exactly the same xi, the error in precision is then 0. Normally, the number of data vectors exceeds the number of neurons and the precision error is thus always different from 0. A common measure that calculates the precision of the mapping is the average quantization error over the entire data set:

$$E_q = \frac{1}{N} \sum_{i=1}^{N} \| x_{i+} m_c \| \quad (3)$$

5.2 Topology Preservation

The topology preservation measure describes how well the SOM preserves the topology of the studied data set. Unlike the mapping precision measure, it considers the structure of the map. For a strangely twisted map, the topographic error is big even if the mapping precision error is small. A simple method for calculating the topographic error:

$$E_q = \frac{1}{N} \sum_{i=1}^{N} u_x(x) \quad (4)$$

Where $u(x_k)$ is 1 if the first and second BMUs of x_k are not next to each other. Otherwise $u(x_k)$ is 0.

7. SYSTEM ARCHITECTURE

Architecture for intrusion detection system based on self-organizing map and artificial Neural Networks. Figure 4. shows the general view of the system. This system uses two detection layers used to detect and isolate attacks. The first layer of a self-organizing map And the next layer of the neural network 1 and 2 were separately taken. The task of separating the first layer attacks from normal traffic. self-organizing map layer, first taught by normal traffic data. In fact, in this episode we have a sample of intrusion detection system to detect anomalies. This means that if the vector has been recognized by the SOM is determined as part of the normal traffic Otherwise be regarded as an attack. The main task of

this layer is actually separating normal traffic from attack traffic. The next layer is the output layer, in general, two routes that one of the normal traffic that has been detected in the neural network together and other vectors in the direction of the attack has been detected in the neural network together.

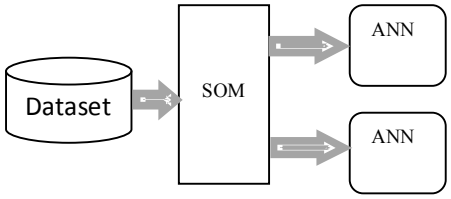

Figure. 4 System architecture

Firstly, using a dataset consisting of 13,472 normal vector is given, self-organizing map of the train. Then in the second step using a set of normalized data and attack that include Vector Data is 72,102, we label the neurons of the self-organizing map this means that each neuron is responsible for one type of attack. The third stage is the main test system, Test data including vector data is 18,794, First, we give to self-organizing map and after determining the direction (attack or normal) to one of the neural networks and this type of attack is detected by neural networks.

7.1 The proposed system simulation parameters

The first layer is a flat topology self-organizing map the dimensions of 50*40 interlocking hexagonal. Gaussian neighborhood function and the learning algorithm used is batch. The second layer of the neural network with separate entrance View of 41 neurons, 10 neurons in the middle and 1 output neuron is used. The size of the training data set includes 72,102 self-organizing map and neural network vector data. Table 2. Number of data vectors in the series to show each type of attack and the size of the testing data set included 18,794 data vectors.

Table 2. The number of data vectors in the training and testing data set self-organizing map and neural network the type of attack

Attack Type	Count (Train)	Count (Test)
Dos	45927	5741
U2R	52	37
R2L	995	2199
Probe	11656	1106
Normal	13472	9711

8. RESULTS AND EVALUATION CRITERIA

Evaluation criteria in the system is calculated as shown in Table 4. All these criteria are based on the accuracy is measured.

Table 4. Evaluation results

Criteria	Error Count	Percent %
Total Error	2735	85.45
False Positive	460	95.27
True Negative	1099	87.91
Attack Type DOS Error	272	95.27
Attack Type U2R Error	33	10.82
Attack Type R2L Error	1953	11.19
Attack Type PROBE Error	17	98.47

9. CONCLUSIONS

The new system has a very high accuracy and speed of detection compared to other methods of attack. The system is also able to detect and classify them by type of attacks.

10. REFERENCES

[1] Lippmann R., Haines J.W., Fried D. J.,Korba J., Das K., "Analysis and Results of the 1999 DARPA Off-Line Intrusion Detection Evaluation", . Recent Advances in Intrusion Detection 2000: 162-182, 2000.

[2] http://www.securityfocus.com/infocus/1520 - An introduction to IDS, (last checked 15/July/2009).

[3] Anderson, James P., "Computer Security Threat Monitoring and Surveillance," Washing, PA, James P. Anderson Co., 1980.

[4] Cohen, Fred, "Computer Viruses: Theory and Experiments," 7thDOD/NBS Computer Security Conference, Gaithersburg, MD, September 24-26, 1984.

[5] Denning, Dorothy E., "An Intrusion Detection Model," Proceedings of the Seventh IEEE Symposium on Security and Privacy, May 1986.

[6] Gollmann, D. (2002), "Computer Security", New Jersey, Wiley.

[7] Rebecca B., Peter M., "NIST Special Publication on Intrusion Detection System" http://danielowen.com /NIDS, (last checked 15/July/2009).

[8] Silva, L., Santos, A., Silva, J., Montes, A.: A neural network application for attack detection in computer networks (2004).

[9] Smith, S.: The Scientist & Engineer's Guide to Digital Signal Processing. California Technical Publishing, USA (1998).

[10] Hagan, T., Demuth, H., Beale, M.: Neural network design. PWS Publishing, USA (1996).

[11] Palmer-Brown, D., Lee, S.: Continuous reinforced snap-drift learning in a neural architecture for proxylet selection in active computer networks (2004).

[12] Planquet, J.: Application of neural networks to Intrusion Detection systems (2001).

[13] Kohonen, T, "Self-Organizing Maps", Springer Series in Information Sciences. Berlin, Heidelberg: Springer. 2006.

[14] P. Lichodzijewski, A. Zincir-Heywood, and M. Heywood. "Dynamic intrusion detection using self organizing maps", 2002.