ABSTRACT

INTRODUCTION ON INTRUSION DETECTION SYSTEMS: FOCUS ON HIERARCHICAL ANALYSIS

by

Ratna Bajaj

In today's fast paced computing world security is a main concern. Intrusion detection systems are an important component of defensive measures protecting computer systems and networks from abuse. This paper will examine various intrusion detection systems. The task of intrusion detection is to monitor usage of a system and detect and malicious activity, therefore, the architecture is a key component when studying intrusion detection systems. This thesis will also analyze various neural networks for statistical anomaly intrusion detection systems. The thesis will focus on the Hierarchical Intrusion Detection system (HIDE) architecture. The HIDE system detects network based attack as anomalies using statistical preprocessing and neural network classification. The thesis will conclude with studies conducted on the HIDE architecture. The studies conducted on the HIDE architecture indicate how the hierarchical multi-tier anomaly intrusion detection system is an effective one.

INTRODUCTION ON INTRUSION DETECTION SYSTEMS:

FOCUS ON HIERARCHICAL ANALYSIS

by

Ratna Bajaj

A Thesis
Submitted to the Faculty of
New Jersey Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
Master of Science in Computer Engineering

Department of Electrical and Computer Engineering

January 2002

APPROVAL PAGE

INTRODUCTION ON INTRUSION DETECTION SYSTEMS: FOCUS ON HIERARCHICAL ANALYSIS

Ratna Bajaj

Dr. Constantine N. Manikopoulos, Thesis Advisor — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. Yun-Qing Shi, Committee Member — Date
Associate Professor of Electrical and Computer Engineering, NJIT

Dr. George E. Antoniou, Committee Member — Date
Professor of Computer Science, MSU

BIOGRAPHICAL SKETCH

Author : Ratna Bajaj

Degree: Master of Science

Date: January 2002

Undergraduate and Graduate Education

- Master of Science in Computer Engineering
  New Jersey Institute of Technology Newark, NJ, 2002

- Bachelor of Science in Computer Engineering
  New Jersey Institute of Technology Newark, NJ, 1998

Major: Computer Engineering

To my dear fiancé

who gave me the motivation

to complete this project and

to my parents for all their support

ACKNOWLEDGEMENT

I would like to thank my professor, Dr. Dinos Manikopoulos, who served as my supervisor and to the teaching assistant, Jun Li, who guided me throughout the project and served as the project lead. Special thanks are also given to Professor Yun-Qing Shi from NJIT and Professor George Antoniou from Montclair University for participating in my committee.

I would also like to recognize my fellow graduate and undergraduate students who worked hard to complete their part in the project.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figure **Page**

CHAPTER I

INTRODUCTION

An **intrusion** is an act of breaking into or misusing a system. Somebody attempting this act is known as a "hacker" or "cracker". Intrusion detection systems gather information from the networks to detect intruders or system abuse. Intrusion detection is needed in today's computing environment because it is impossible to keep pace with the current and potential threats and vulnerabilities in our computing systems. The environment is constantly evolving and changing fueled by new technology and the Internet. To make matters worse, threats and vulnerabilities in this environment are also constantly evolving. Intrusion detection systems are tools designed to assist in managing threats and vulnerabilities in this changing environment.

Threats can be people or groups who have the potential to compromise your computer system. These may be a curious teenager, a disgruntled employee, or espionage from a rival company or a foreign government. The hacker has become a nemesis to many companies.

Vulnerabilities are weaknesses in the systems. Vulnerabilities can be exploited and used to compromise your system. New vulnerabilities are discovered all of the time. Every new technology, product, or system brings with it a new generation of bugs and unintended conflicts or flaws. Also, the possible impacts from exploiting these vulnerabilities are constantly evolving. In a worst-case scenario, an intrusion may cause production downtime, sabotage of critical information, theft of confidential information, cash, or other assets, or even negative public relations that may affect a company's stock price.

Intrusion detection products are tools that can assist in protecting a company from intrusion by expanding the options available to manage the risk from threats and vulnerabilities. Intrusion detection capabilities can help a company secure its information. The tool could be used to detect an intruder, identify and stop the intruder, support investigations to find out how the intruder got in, and stop the exploit from use by future intruders. The correction should be applied across the enterprise to all similar platforms. Intrusion detection architecture and products can become a very powerful tool in the information security practitioner's tool kit.

CHAPTER 2

CATALOG OF INTRUSION DETECTION SYSTEMS

The following are the analyzed intrusion detection systems.

2.1 Network Intrusion Detection Systems (NIDS)

Network Intrusion Detection Systems (NIDS) analyze network traffic for attacks that exploit the connections between computers and the data that can be accessed via a network connection by examining the individual packets flowing through a network. Unlike firewalls, which typically only look at IP addresses, ports and ICMP types, network based intrusion detection systems (NIDS) are able to understand all the different flags and options that can exist within a network packet. The role of the network Intrusion Detection Systems (IDS) is to flag and sometimes stop an attack before it gets to information assets or causes damage. A NIDS can therefore detect maliciously crafted packets that are designed to be overlooked by a firewall's relatively simplistic filtering rules. Hackers often craft such traffic in order to "map out" a network, as a form of pre-attack reconnaissance.

NIDS are also able to look at the "payload" within a packet, i.e. to see which particular web server program is being accessed, and with what options, and to raise alerts when an attacker tries to exploit a bug in such code. Most firewalls are unable to do this. NIDS can detect the broadest range of attacks on corporate information assets. NIDS are effective for monitoring both inbound and outbound network traffic.

Network sniffers are an effective means for gathering information about events that occur on the network architecture. Capturing packets before they enter the server is

an effective means of monitoring data on the network. If the analysis is on the lower level of analyzing the content of the TCP or IP packet, then the system can perform quickly. However, if the system analyses each packet with respect to the application or protocol, this could be time consuming and raises several issues. Detection of network specific attacks cannot be determined in a timely fashion. Also, is it difficult to identify the user how submitted the packets. Another problem that arises is that the encryption makes it hard to analyze the payload of the packets. Therefore, a skillful attacker could still get past these intrusion detection systems.

2.2 Host-based Intrusion Detection Systems (HIDS)

Host-Based Intrusion Detection Systems (HIDS) monitor specific files, logs and registry settings on a single individual computer or "host" and can alert on any access, modification, deletion and copying of the monitored object. The role of a HIDS is to flag any tampering with a specific host and can automatically replace the altered files when changed to ensure data integrity. They are able to detect such things as repeated failed access attempts or changes to critical system files

A derivation of HIDS is centralized-host-based intrusion detection (CHIDS) that serves the same purpose but does the analysis centrally by sending monitored files, logs and registry settings to the manager for analysis. The primary difference between these systems is as follows.

- CHIDS is more secure because it sends all the needed information off the host so that if the host is compromised, the alerting and forensic analysis can still take place. The tradeoff is that centralized analysis requires substantially more network bandwidth to move the data to the manager.

- HIDS makes policy compliance decisions locally and only sends alerts to the manager when warranted. This uses substantially less network bandwidth. The shortcoming of HIDS is that if the host is compromised there is no alert or forensic data to determine what happened or what was lost.

Information about the activities are gathered by the host audit. However, while the host based intrusion detection system is processing the audit trail and setting off alarms, the attacker can sabotage the audit trail or the entire system.

There are other audit sources that are used in host based intrusion detection systems to gather information. The operating system can obtain a snapshot of the information about the events occurring. Accounting, another method, provides information on the consumption of the shared resources, such a processor time, memory, etc. The drawbacks with this are that there is a lack of parameterization, it is either on or off always for all users. Furthermore, there is lack of precise time stamp of when the events occurred which can be crucial to examine. Syslog is another audit service that is provided by the operating system to the applications. The service receives a text string from the application and prefixes it with a time stamp and the name of the system, then it is archived. C2 security audits are required on all computers systems. It records the crossing of instructions executed by the processor in the user space and instructions executed in the kernel. This contains information about the events, and user identification. There are many advantages to this because it identifies the user and login information. It repartitions the audit events into classes to facilitate the configuration of the audit system. One of the main drawbacks is that because of the high use of system resources the processor performance can decrease up to twenty percent.

2.3 Hybrid Intrusion Detection Systems

Hybrid Intrusion Detection Systems complement HIDS technology with the ability to monitor the network traffic coming in or out of a specific host. This is very different than NIDS technology that monitors all network traffic. Management and alert notification from network and host based intrusion detection devices can be done with Hybrid Intrusion Detection Systems.

CHAPTER 3

CLASSIFICATION OF INTRUSION DETECTION SYSTEMS

The task of intrusion detection systems is to monitor and detect any misuse of the system. In today's computerized world, commercial tools for intrusion detection systems are becoming easily available.

A generic intrusion detection system is essentially a detector that processors information received from a system (server, mainframe, firewall etc.). The detector processes the following information: knowledge of attacks, configuration information of the current state of the system, and audit information (information about the inner workings and behavior of the system). The detector evaluates all this data to determine if there is indeed an intrusion.

Evaluating efficiency is essential in intrusion detection systems. Different measures need to be considered such as accuracy, performance, such as the rate at which audit events are processed. Completeness is also important as well as fault tolerance. The intrusion detection system to should be resistant to attacks. Finally, timeliness to perform and propagate the analysis as soon as possible so that it can be handled.

There are many characteristics of intrusion detection systems. The analyzer has two functionality's: it can be behavior based, using information about the normal behavior, or knowledge based where the system uses information about the attacks. Another functional characteristic is the behavior on detection, how it reacts. It can react passively, generating alarms, or actively for example logging out attackers. (Described in detail earlier.) The audit source location distinguishes amongst systems based on the kind of information they analyze (logs, packets etc.). Finally, there is a usage frequency

concept, either running real-time or periodically. In real-time analysis information about the environment is acquired immediately after an event occurs, but with a static intrusion detection tool a snapshot of the environment is taken periodically.

3.1 Misuse vs. Anomaly Detection Mechanisms

A second level of categorization of intrusion detection systems is between those based on the detection of misuse and those based on the detection of anomalous use.

Misuse detection or knowledge-based detection within a network-based IDS involves checking for illegal types of network traffic, e.g. combinations of options within a network packet that should never legitimately occur. Misuse detection by host-based IDS would include attempts by a user to execute programs for which they have no legitimate need. The techniques used apply the knowledge about attacks and system weaknesses. When the system notices these attacks, an alarm is triggered.

There are advantages and disadvantages to this approach. Some advantages are that the potential for false alarm rates are low and the contextual analysis proposed is detailed so it becomes easier to take preventive or corrective action. However, there are disadvantages as well such as gathering and updating the environment with the known attacks. Maintenance requires careful analysis. Also, this has to be closely tied to the environment platform it is running on. Lastly, attacks involving insiders would be more difficult to detect.

Different systems can be used by Misuse detection techniques. One is the expert system, which contains a set of rules that describes attacks. The audit events are translated and a semantic is attaches to them. Rule-based knowledge is a common tool for

knowledge based detection systems. With this approach, the audit trail is searched for evidence of attacks. However, sometimes it is difficult to extract the knowledge about the attacks, and sometimes the information is not available. Also, the speed the information in the audits is not always efficient because of the importing and processing time involved. It is seen that these expert systems are used in prototypes because it is not the most efficient approach.

Signature analysis is an approach similar to the expert system. The data is exploited in a different manner, which makes it more efficient. The attack scenarios are translated into sequences of audit events or patterns of data, which decreases the semantic level of the attack description. Although with this approach the problem for frequently updating to keep up with new vulnerabilities remain, it is an efficient approach.

Detection of anomalous (behavior based) activity relies on the system knowing what is normal network traffic, and thus what isn't. Anomalous traffic to a host-based IDS might be interactive accesses outside of normal office hours. An example of anomalous traffic on a network-based IDS is repeated attempted access by one remote machine to many diverse services on one or more of your internal systems, all in quick succession.

The model of a normal behavior is extracted from reference information collected. The system then compares this model with current activities. If something abnormal is seen, an alarm is generated. Advantages of this system are that new and unforeseen vulnerabilities can be captured. These systems are not closely tied to the platform, which they run on. The disadvantage is that there tends to be high false alarm rate with an

anomaly system, mostly because the normal behavior can change frequently and be detected as an attack.

Statistics plays the most important role in anomaly systems. System behavior is measured by a number of variables sampled over a period of time, such as login and logout data, amount of resources consumed during the session. Then the system can monitor whether thresholds are exceed, to determine abnormal behavior. Expert systems as described above can also be used for anomaly detection.

Neural networks (described in more detail later), are a common algorithm used to learn about the relationship between input (output) vectors and generalize them to obtain new input (output) vectors in a reasonable way. Neural networks are used to learn the behavior of the components in the system by using a simple way to express nonlinear relationships between variables, which is why they have an advantage over the statistics.

Another technique used is the user intention identification, which models the normal behavior of users by the set of high level tasks they have to perform on the system. The analyzer keeps track of the tasks a user can perform.

Computer immunology differ from the previous techniques in that it builds a model of the UNIX network services rather than the user. This technique first collects references audits and checks whether the sequences generated are in the reference model or not.

Many modern systems use a combination of both misuse and anomalous detection engines.

3.2 Passive vs. Reactive

Another method of categorizing intrusion detection systems is by their passive or reactive nature. Passive systems simply detect the potential security flaw, log the information and raise an alert. Reactive systems, on the other hand, are designed to respond to the illegal activity, for example by logging off a user or by reprogramming the firewall to disallow network traffic from a suspected hostile source. While a reactive system might seem like an ideal solution there are serious drawbacks to such systems. Consider the following situation. An attacker crafts malicious network traffic aimed at your Internet mail system. The traffic is crafted so that it appears to come from your Internet Service Provider's mail system. Your network-based IDS detects this anomalous traffic, and reprograms your firewall to disallow all traffic from that system. Your company is now unable to receive any email via your ISP. A properly trained Intrusion Detection Analyst should be able to identify fake traffic, or, where this is not technically feasible, he or she would be able to work with your ISP to establish the source of the problem.

CHAPTER 4

INTRUSION DETECTION ARCHITECTURE

Some intrusion detection systems are based on a multi-tier architecture of a detection technology, a data analysis and configuration management layer and the user console or graphical user interface (GUI). When used by an individual on a single host, all the layers of the system may reside on the same host. In enterprise or managed service deployments, each layer of the intrusion detection system is generally deployed separately to facilitate operations, ensure performance and support organizational workflow.

Detection technologies vary by the different types of intrusion detection systems.

- **Sensors** (sometimes called engines or probes) are deployable software or appliance-based technologies that allow network intrusion detection systems to monitor the mass of traffic on high-speed networks. Sensors are placed in specific locations at the network perimeter or within the network fabric. Sensors are processor-intensive devices and generally require their own host or appliance to function correctly. The sensor analyzes all network traffic, looking for evidence of intrusion, and then reports the information to a centrally located manager following the parameters of the network IDS policy.

- **Agents** are deployable software installed on a particular host in a host-based intrusion detection system. Agent software generally has a small footprint and uses very little processing power. The agent's function is to monitor specific files or logs on the host and reporting to a central manager if and when these particular files are accessed, modified, deleted or copied according to the host-based security policy. Agents are considered intelligent software as they determine policy compliance on the host and then only report breaks in the security policy.

- **Hybrid agents** combine the functionality of a host-based agent with network based sensor technology that is limited to analyzing only the network traffic addressed to the specific host where the hybrid agent is installed. A hybrid agent's footprint is generally larger because of the additional functionality. The processor utilization of the hybrid agent is much greater than a host-based agent because of the continual processing of network traffic for the host.

- **Collectors** are like agents in that they are lightweight software applications that reside on the host, similar to agents. The primary difference is that collectors are considered dumb devices because they do not make any decision at the host level. A collector's function is to harvest log, registry and file information from the host and to forward all of it to a central manager as soon as the entry occurs. The central manager does all analysis and decision-making for policy compliance. Most applications that use collectors are considered centralized, host-based intrusion detection systems.

The manager layer is responsible for accepting inputs from the deployed detection technologies and storing, analyzing and correlating the data for higher level intrusion detection. The manager is also the configuration and policy repository for the intrusion detection system. The manager uses some type of data and configuration store and applies ease-of-use, data mining and system management features to make the large amount of data provided by intrusion detection systems usable information for security policy enforcement and IT policy decision support. The manager is generally installed in a data center or server room with other server platforms that warrant physical protection like automated back-up, fire-protection and un-interruptible power supplies (UPS).

The operator of the intrusion detection system will interface with the system via a console (sometimes called GUI or UI) that is generally installed on a host in the network operations center (NOC) or on the security professional's primary host. The console's primary function is to make the monitoring and reporting of the system as intuitive and flexible as possible, thereby increasing the value of the information provided by the system.

CHAPTER 5

STUDY OF DIFFERENT TYPES OF NEURAL NETWORKS

When building anomaly intrusion detection systems, statistical modeling and neural networks are applied. These schemes construct statistical models of the typical "normal" behavior and observe for actions that digress from the normal behavior. Statistics in anomaly intrusion detection systems measure the means and the variances of some variables and detect whether certain thresholds are exceeded. Neural Networks require high computational intensity and long training cycles. Five different types of neural networks: Perception, BP, PBH, Fuzzy ART MAP and RBF are examined in this chapter. [1]

5.1 Perception Architecture

The simplest neural network is the perception architecture. It consists of a single neuron with adjustable synapses and threshold.

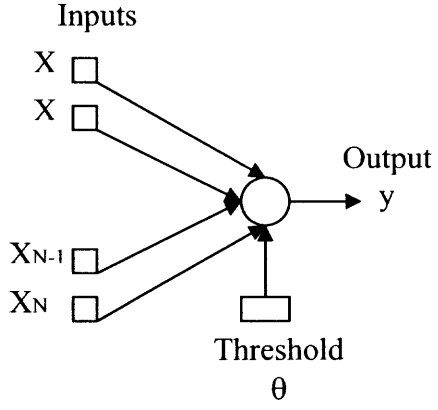

Figure 5.1 Perceptron architecture. [1]

5.2 Backpropagation Network (BP)

This Backpropagation network (BP) is made up of an input layer, one or more hidden layers, and an output layer. This network has high generalization capabilities and can solve diverse problems.

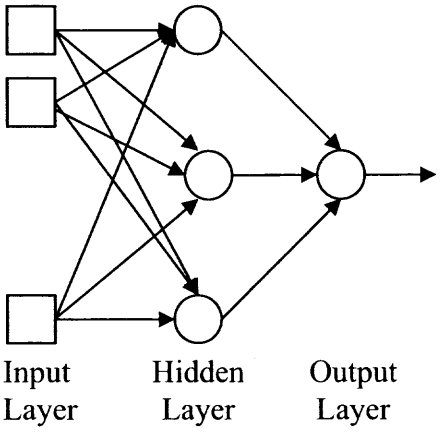

Figure 5.2 BP architecture. [1]

5.3 Perceptron -Backpropagation Hybrid Network

The Perceptron-backpropagation hybrid network (PBH) is a superposition of a perceptron and a small Backpropagation network. They are capable of exploring linear and non-linear correlation between the input stimulus vectors and the output values.

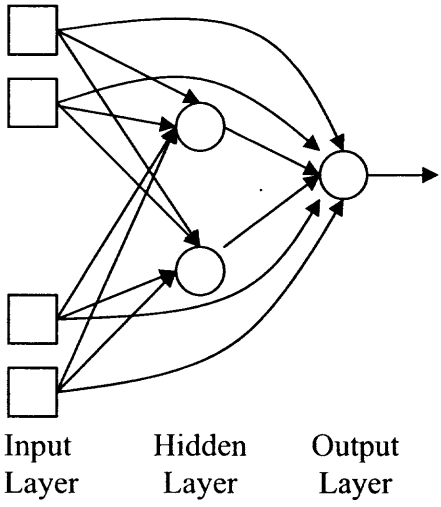

Figure 5.3 PBH architecture. [1]

5.4 Fuzzy Artmap

Layers in the Fuzzy ARTMAP network are connected by a subsystem referred to as a "match tracking system"

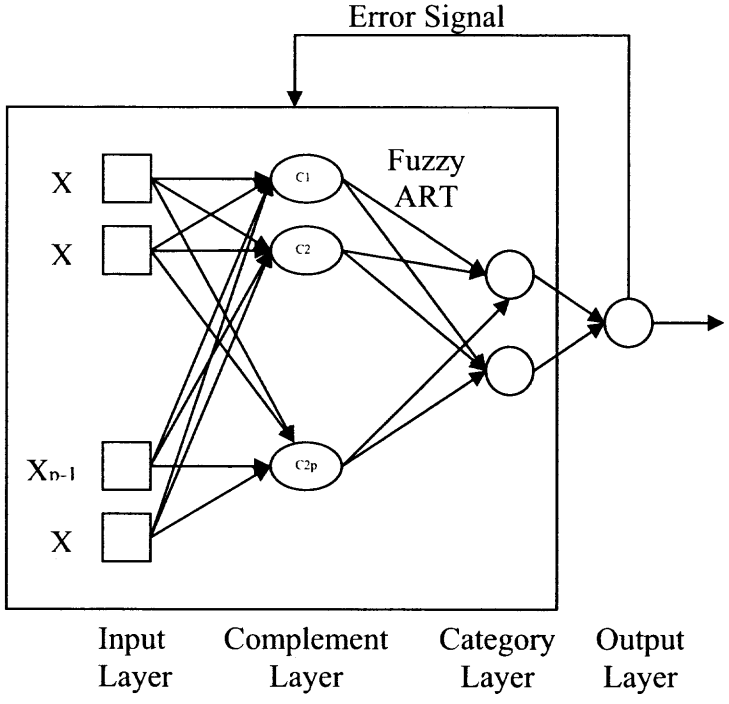

Figure 5.4 Fuzzy ARTMAP architecture. [1]

5.5 Radial-Basis Function Network

The Radial-basis function network (RBF) consists of three different layers. The first layer, input layer, is made up of source nodes. The second layer is a hidden layer of high dimension, which serves a different purpose from that in BP network. The third layer, output layer, supplies the response of the network to the activation patterns applied to the input layer.

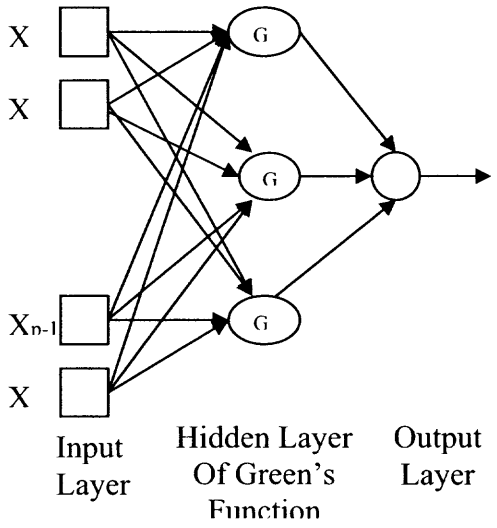

Figure 5.5 RBF architecture. [1]

When tests at the New Jersey Institute of Technology were performed, a virtual network was used to generate attack scenarios. The UDP flooding attack was simulated on the test bed. Four different scenarios with normal different traffic loads and attack traffic was used. Each simulation scenario collected 10,000 record of network traffic. It was concluded from these tests that the classification capabilities of the BP and PHB networks are for desired for statistical anomaly intrusion detection systems because they out performed the Perceptron, Fuzzy ARTMAP, and RBF networks. [1]

CHAPTER 6

A HIERARCHICAL ANOMALY NETWORK INTRUSION DETECTION SYSTEM

In the research conducted at New Jersey Institute of Technology, the prototype of a hierarchical anomaly network intrusion detection system that uses statistical models and neural networks to detect attacks was used. The Hierarchical Intrusion Detection System (HIDE) used is a multi tier, multi window, anomaly intrusion detection system that may operate automatically, adaptively, and proactively and be applied to networks that are wireless and wired. By stress testing this system, it was proven that HIDE could reliably detect UDP flooding attacks with attack traffic intensity as low as five to ten percent of the background traffic.

In the HIDE system, intrusion is analyzed and detected based on performance of Intrusion Detection Agents (IDA's) on more than one tier. This hierarchical multi tier architecture allows us to study the performance of the IDA's on different tiers. The IDA's monitor the activities of the hosts or the network they are attached to. This system monitors different parameters and analyses abnormal activity.

The architecture works simply that it set up with many tiers, each containing an IDA. For example, the system can be set into three tiers, where Tier 1 monitors the system activities of the servers and bridges within a subnet. Tier 1 generated reports about its host's activities to Tier 2 agents. These Tier 2 agents then gather and observe their network status and reports as well as the ones from Tier 1. This is then sent to Tier 3. Tier 3 also directly communicates with Tier 1 agents.

In the HIDE system architecture each IDA contains the same components, similar to the generic intrusion detection architecture described earlier. Each IDA contains a Probe, which collects the network traffic of the components on the network. The probe takes this traffic data and creates a set of statistical variables to of the network status and relays the reports to the event preprocessor. The event preprocessor takes these reports from the probe and converts the information for the statistical model to process. The statistical processor keeps the reference models of the normal network activities and compares the reports received from the event preprocessor and forms a stimulus vector to feed into the neural network classifiers. The Neural Network Classifier analyses the stimulus vector from the statistical model and determines if the traffic is normal. The Post Processor generates reports for the agents and displays the results through a user interface. Figure 6.1 is the diagram if the IDA's.

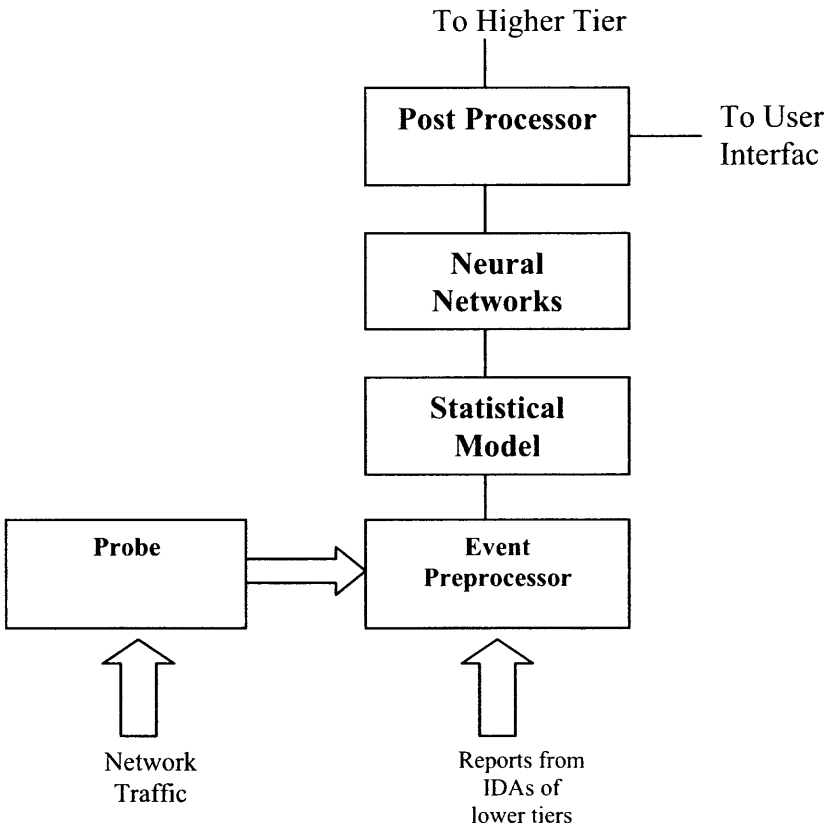

Figure 6.1 Intrusion detection agent (IDA). [2]

The statistical processor builds and analyses real-time probability density functions (PDF's) of the monitored network parameters and compares the measured PDF's to the normal preset PDF's models of the normal activity. Network based attacks can have different time duration that needed to be monitored with different time windows. This study uses a geometrically increasing detection time slice corresponding with each layer window. The events are feed into the event buffer of Layer 1 and reference model builder. The stored events will be compared with the reference model of

that layer and the results are sent into the neural network classifier to a certain the network status during that window. The event buffers then empties out once it is full. The stored events are then averaged and forwarded to the event buffer Layer 2. This process is repeated until the stored events arrive at the top level. Once they have reached the top level they are dropped after being processed [3]. This process is illustrated in the Figure 6.2.

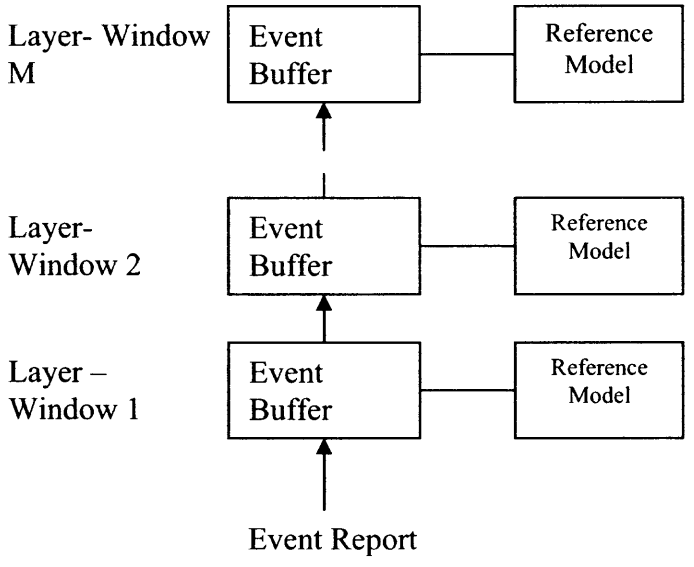

Figure 6.2 Statistical model. [2]

6.1 Simulation Test

This section describes the simulation test that was run to evaluate the hierarchical multi tier design of the HIDE architecture at New Jersey Institute of Technology. The Backpropagation (BP) network architecture was used in this study with two hidden layers

(neurons). The main objective of this study is to evaluate the performance of hierarchical multi-tier architecture to determine if this is an effective solution for intrusion detection systems.

Test data was generated by running a network simulation on a Optimized Network Engineering (OPNET) Modeler. OPNET is a tool for event driven modeling and simulation of communications networks, devices and protocols. The performances of Tier 1 and Tier 2 were examined. Performances of the Tiers are compared by evaluating the mean square root (MSR) errors and misclassification rates. Three scenarios were executed with different background and attack traffic (See Table 6.1). The experimental test bed used is shown below in Figure 6.3:

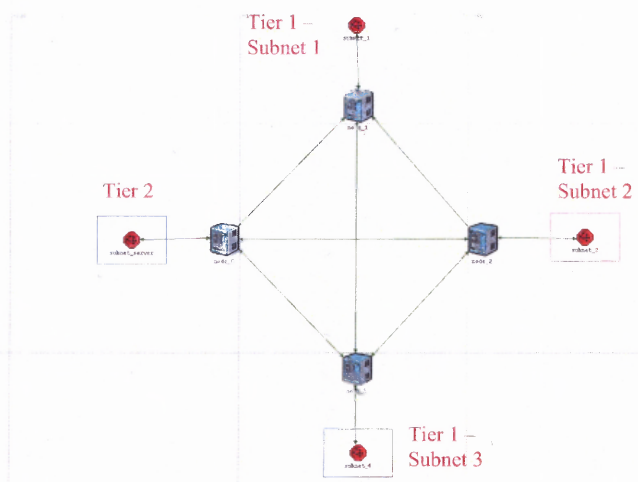

Figure 6.3 Simulation test bed.

There were three subnets (representing Tier 1) with 11 hosts and an IDA on each subnet. (Subnet 1 – Ethercoax, Subnet 2 – Ethernet, Subnet 3 – Token Ring). The three subnets each had an IDA, which monitored the activities and reported them to the monitor in the Tier 2. The fourth subnet (Tier 2) contained a ftp and telnet server and an IDA (monitor). The monitor on the Tier 2, monitors the status of the entire network and detects intrusion. Routers for all four subnets were connected via T1 link (see Figure 6.1).

The background traffic used in this experiment was HTTP (TCP) and Email (UDP) traffic. Each scenario ran for six hours and the network traffic was collected every 4 seconds (observation window size). The total data sets that were collected were 5400 records, for each scenario. This was divided into 3600 records for training set and 1800 records for testing. The number of epochs used to trail the neural networks for this study was 100. In each epoch, the neural network is trained by using the training record data and then tested by the testing record data. To collect the results, the training record data and testing record data remain the same during the epochs. The test data is analyzed to determine the overall performance and convergence speed of the neural network classifier.

Table 6.1 Scenario Traffic Information

Scenario 1	
Type of Traffic	**bps**
Background Traffic – (Total Traffic)	400,950
HTTP (TCP) Traffic (95 % of total background traffic)	308, 902.5
EMAIL (UDP) Traffic (5 % of total background traffic)	20,047.5
Attack Traffic (5 % of HTTP traffic) Packet Length = 500 bytes Packet Rate = 0.21 seconds	2,382
Scenario 2	
Type of Traffic	**bps**
Background Traffic – (Total Traffic)	3,142,084
HTTP (TCP) Traffic (99 % of total background traffic)	3,110,663.16
EMAIL (UDP) Traffic (1 % of total background traffic)	31,420.84
Attack Traffic (5 % of HTTP traffic) Packet Length = 1000 bytes Packet Rate = 0.05 seconds	19,546
Scenario 3	
Type of Traffic	**bps**
Background Traffic – (Total Traffic)	1,248,151
HTTP (TCP) Traffic (98 % of total background traffic)	1,223,187.98
EMAIL (UDP) Traffic (2 % of total background traffic)	24,963.02
Attack Traffic (5 % of HTTP traffic) Packet Length = 1500 bytes Packet Rate = 0.19 seconds	7,710

6.2 Simulation Results

The Table 6.1 and Figures 6.4 – 6.9 below outline the performance of the IDA 's for Tier 1 and Tier 2 based on the mean square root (MSR) errors and the misclassification ratio of the output. The Misclassification Ratio is the percentage of the network traffic that is misclassified by the neural network intrusion detection classifier during one epoch. This included false negative and positive misclassification.

Table 6.2 Average of MSR and Misclassification Rates

Scenario 1		
Tier IDA (Subnet)	**MSR Error**	**Misclassification Rates**
Tier 1 IDA (Subnet_1)	0.008337	0.002383
Tier 1 IDA (Subnet_2)	0.004665	0.001472
Tier 1 IDA (Subnet_3)	0.003375	0.000789
Tier 2 IDA	0.000923	0.000011
Scenario 2		
Tier IDA (Subnet)	**MSR Error**	**Misclassification Rates**
Tier 1 IDA (Subnet_1)	0.002178	0.000556
Tier 1 IDA (Subnet_2)	0.001922	0.000556
Tier 1 IDA (Subnet_3)	0.002049	0.000556
Tier 2 IDA	0.004045	0.001112
Scenario 3		
Tier IDA (Subnet)	**MSR Error**	**Misclassification Rates**
Tier 1 IDA (Subnet_1)	0.002648	0.000628
Tier 1 IDA (Subnet_2)	0.001818	0.00002223

Tier 1 IDA (Subnet_3)	0.003764	0.001406
Tier 2 IDA	0.001398	0.00000

From Table 6.2, it can be seen that when the attack and background traffic was low (Scenario 1) the MSR and Misclassification Rates for the Tier 2 are significantly lower then the ones for the Tier 1 data, but as the attack and background traffic increased (Scenario 3, and Scenario 2), the values became higher for the Tier 2 as compared to Tier 1.

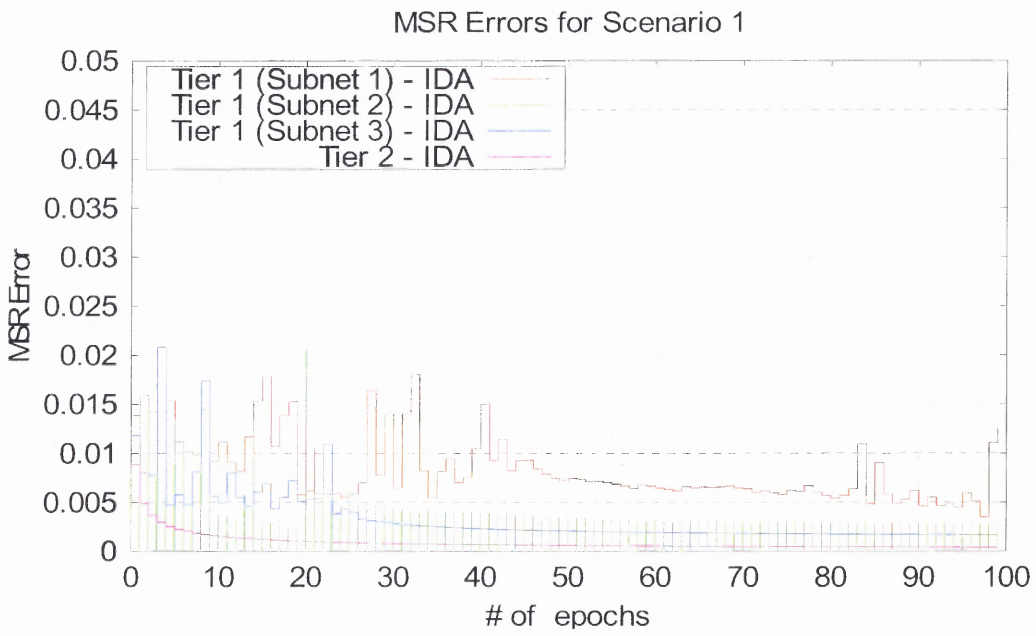

Figure 6.4 MSR error of Scenario 1.

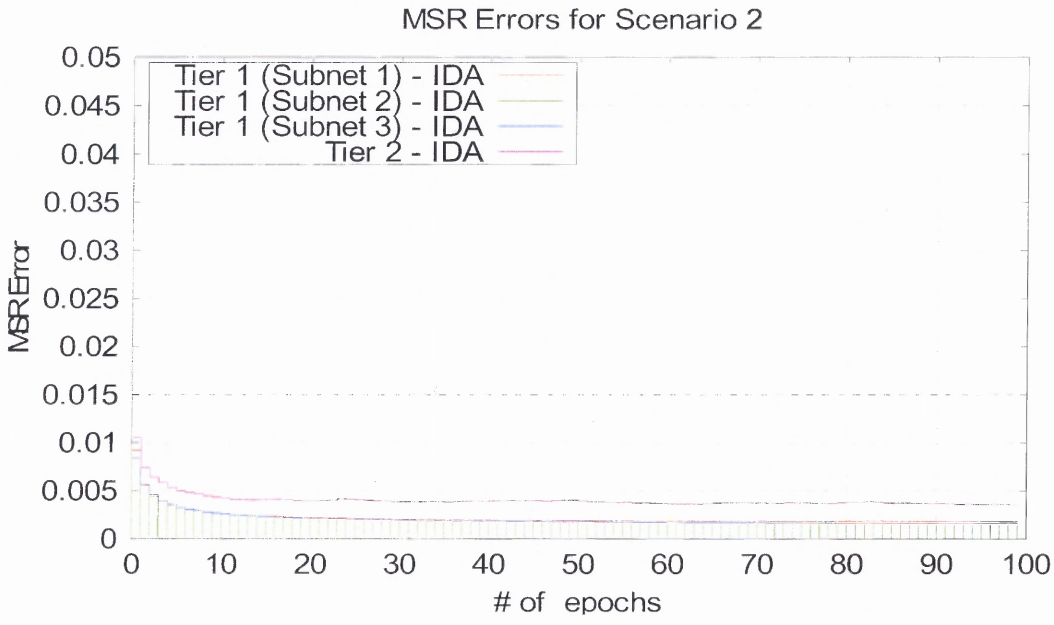

Figure 6.5 MSR error of Scenario 2.

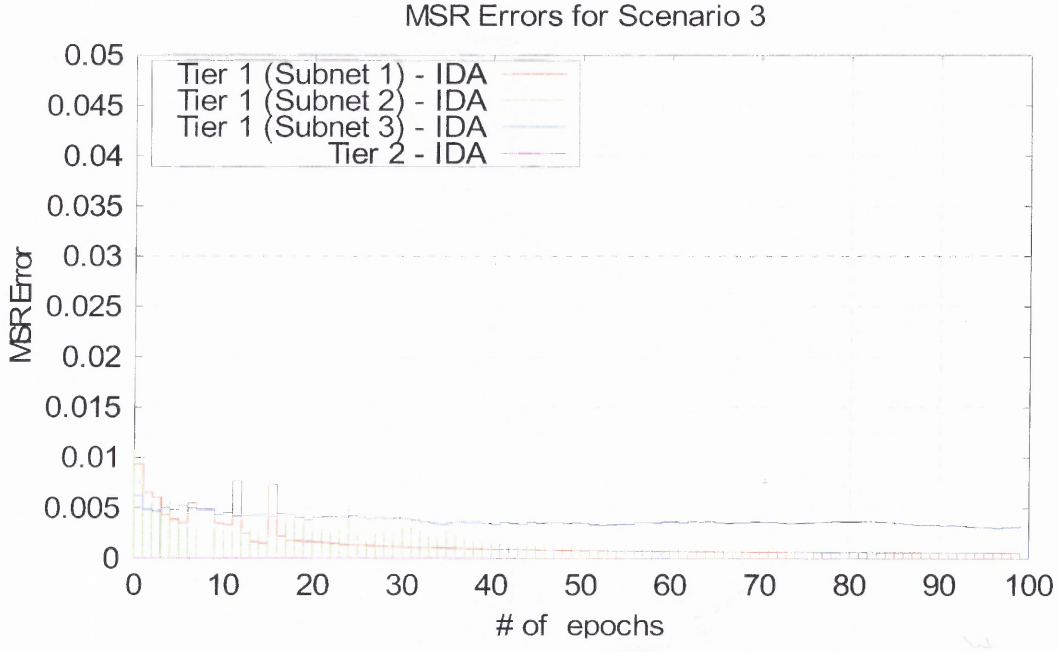

Figure 6.6 MSR error of Scenario 3.

The results in Figure 6.6 show the Mean Square Root (MSR) for each scenario. As seen in the graph initially the MSR is high but then decreases and remains somewhat constant while converging for the Tier 1 and Tier 2.

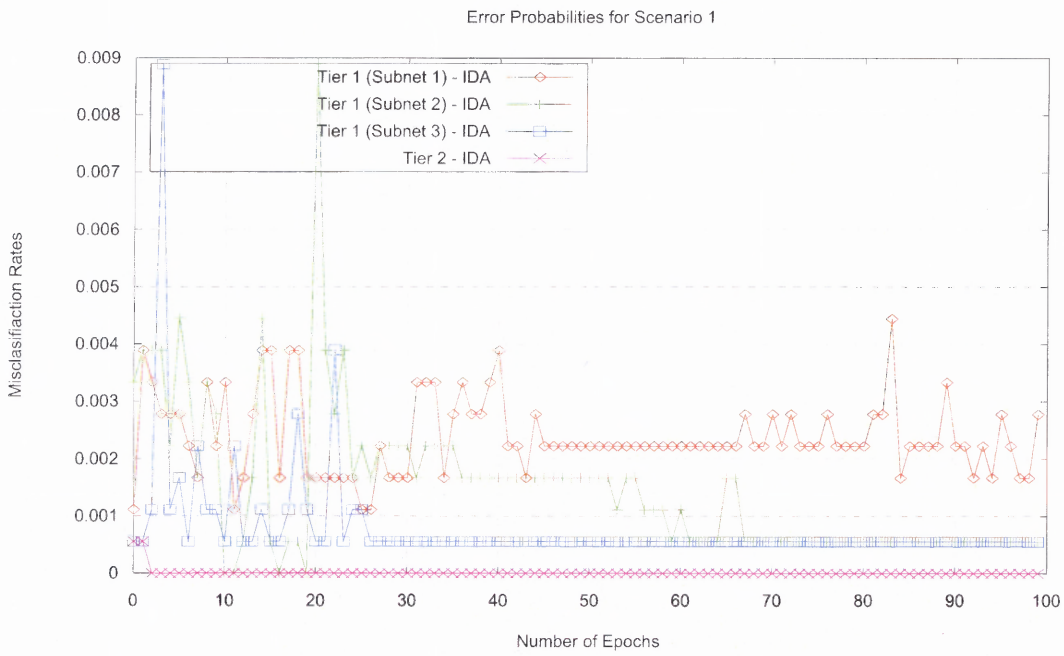

Figure 6.7 Error probabilities of Scenario 1.

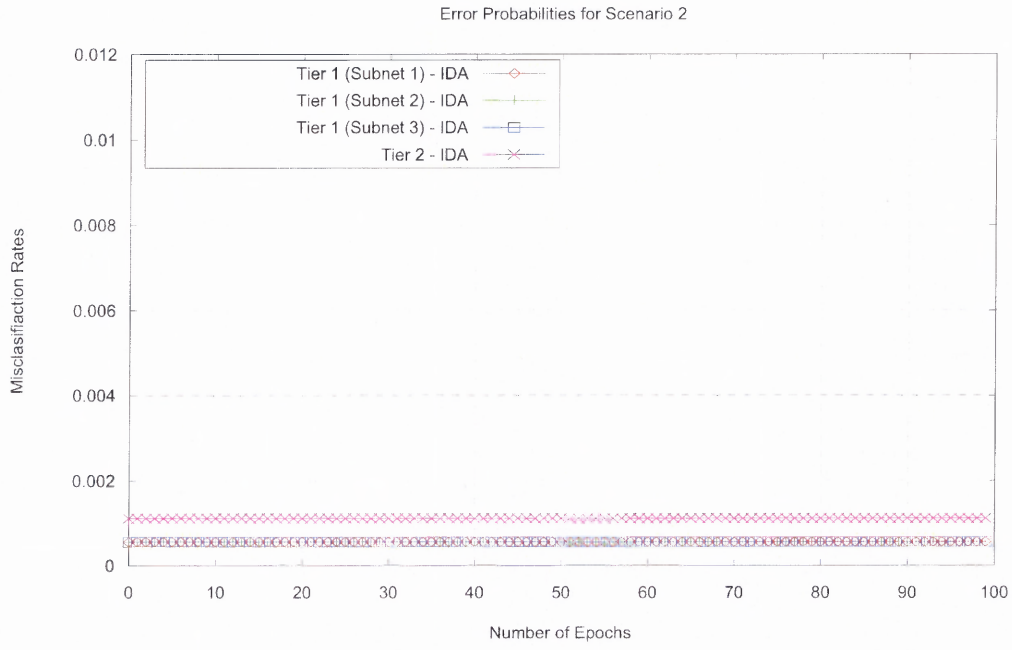

Figure 6.8 Error probabilities of Scenario 2.

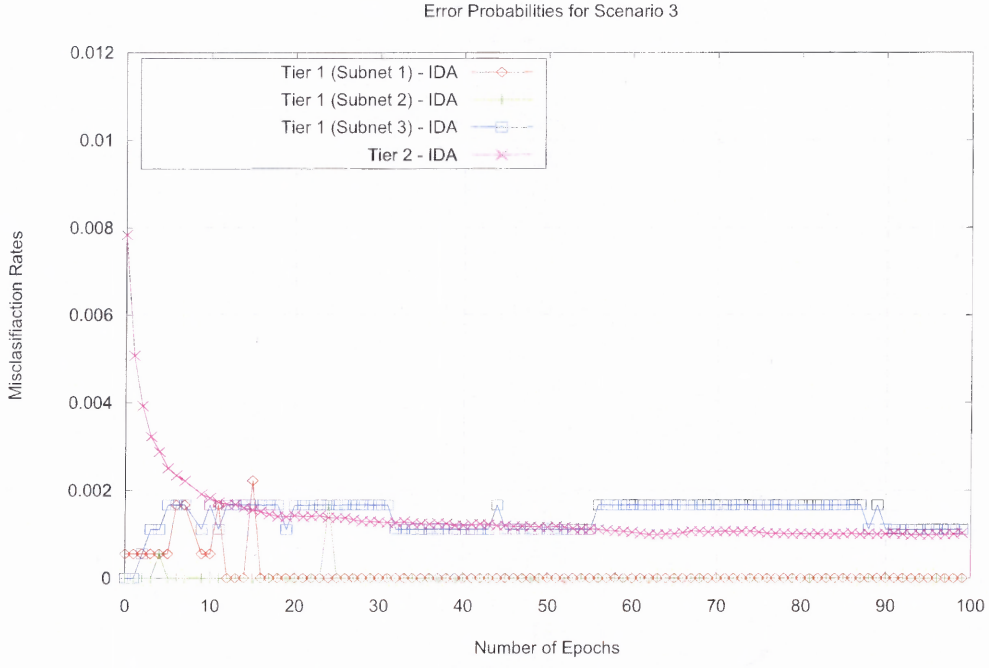

Figure 6.9 Error probabilities of Scenario 3.

The results in Figure 6.9 show the Misclassification Rates. From these results, it can be seen once again, although the MSR tends to be higher for Tier 2 in Scenario 2, that early on the Misclassification rates is high but as the epochs increase it converges and remains constant.

Figures 6.10 - 6.12 illustrate the Receiving Operating Characteristics (ROC) curves for all the scenarios examined with each scenario it can be seen the breakdown between the subnets of Tier 1 and Tier 2. By examining these graphs, it can be determined what the most excellent performance the neural classifier can achieve. The rate of normal traffic mistaken for attack (intrusion) traffic is donated on the X-axis, (False Alarm Rate). The ratio between the number of correctly identified attacks and total number of intrusions is donated by the Y-axis, (Detection Rate).

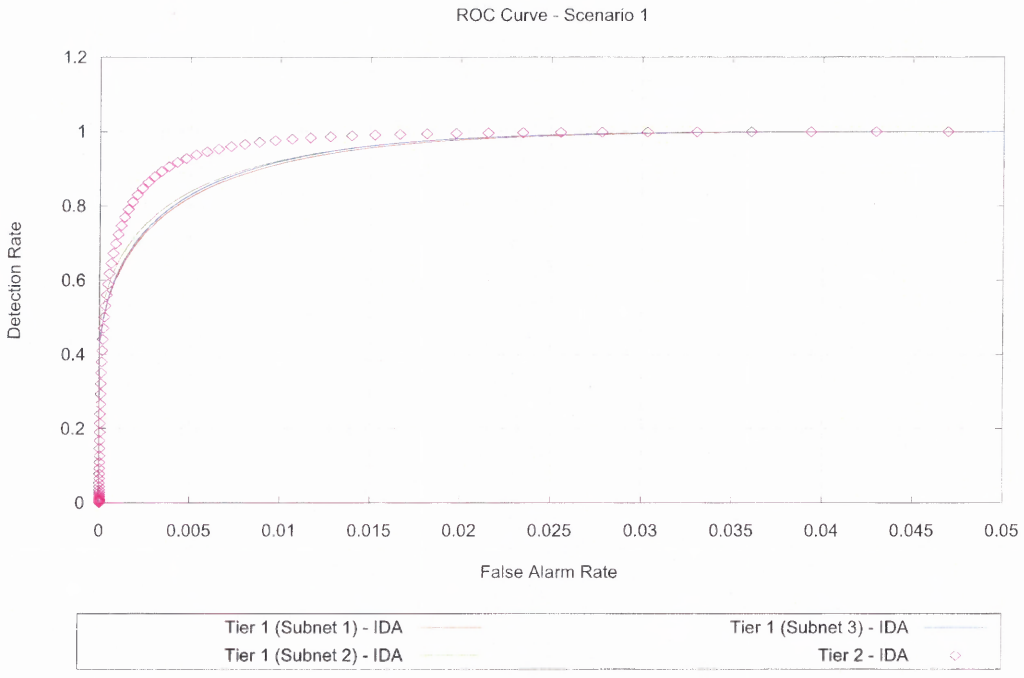

Figure 6.10 ROC curve for Scenario 1.

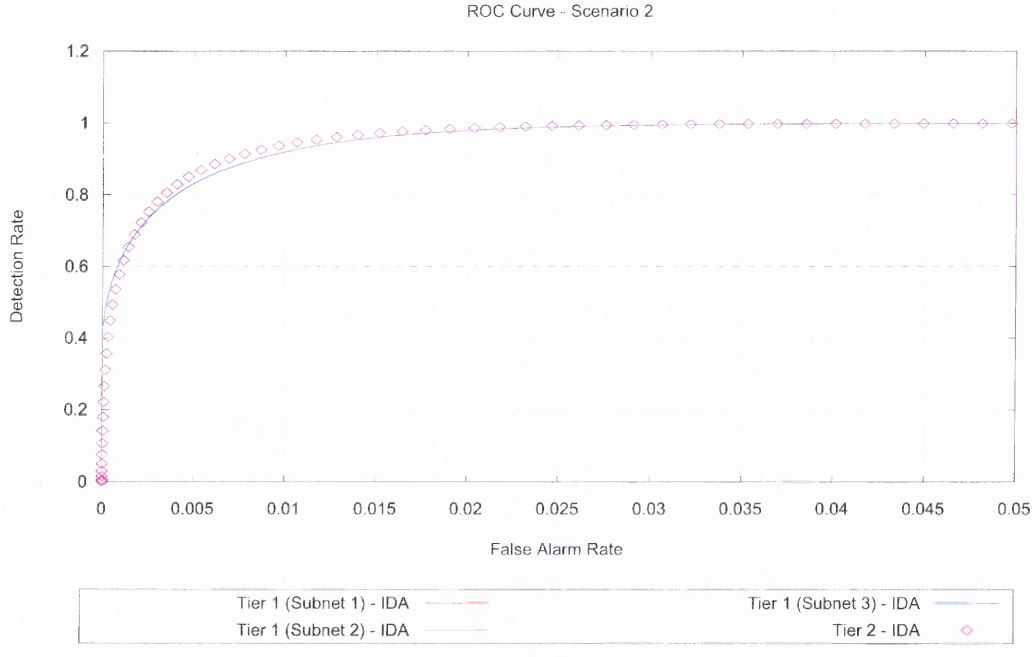

Figure 6.11 ROC curve for Scenario 2.

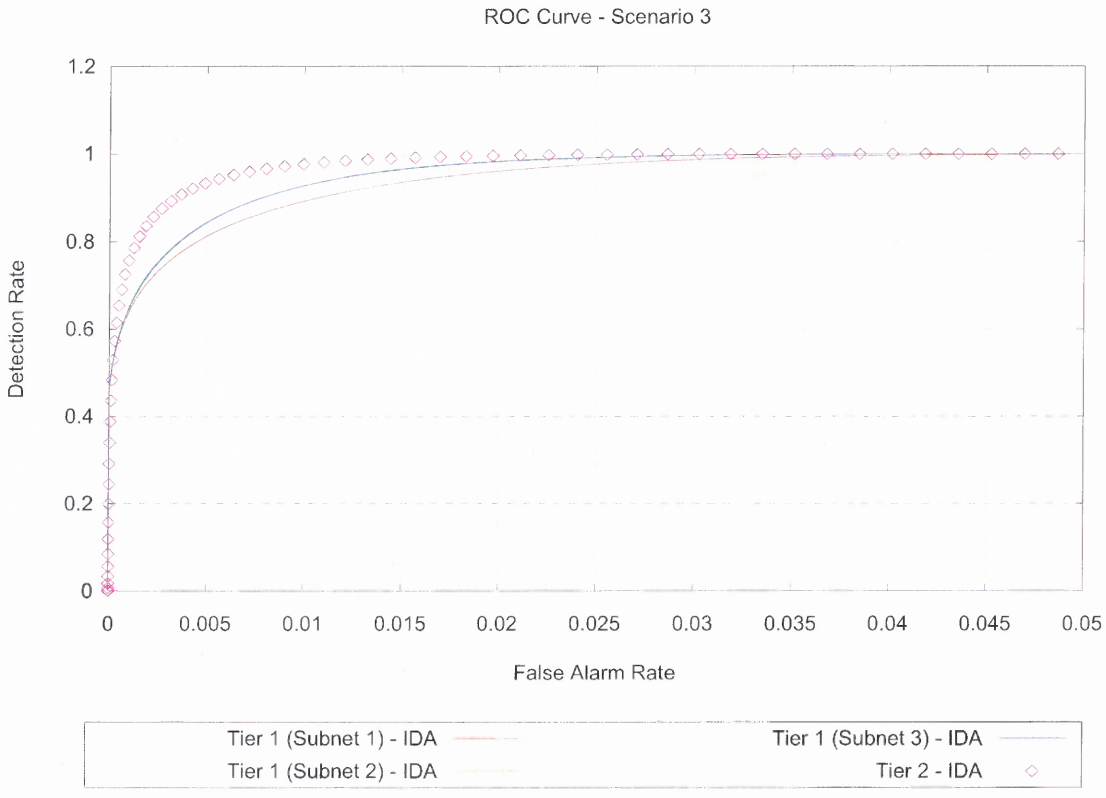

Figure 6.12 ROC curve for Scenario 3.

From Figures 6.10 - 6.12, it can be seen the optimal detection with high detection rate and low false alarm rate (upper left corner of the curve). The detection performance does not seem to decrease as the attack intensity increases. Consider the scenarios above: Figure 6.10 had attack traffic of 2,382 bps per subnet, Figure 6.11 had attack traffic of 19,546 bps and Figure 6.12 had attack traffic equal to 7,710 bps. However, it can clearly be seen the detection performance for Tier 1 and Tier 2 was not affected by the amount of attack traffic, (whether the traffic was low or high). If the performance of Tier 1 are

compared against Tier 2 it can be clearly seen that Tier 2 outperforms the Tier 1 IDA's. This proves the efficiency of the system.

From the MSR and Misclassification Rate graphs, optimistic results were achieved. The convergence and low classification rates prove that the system would operate in real-time fashion, monitoring intrusion accurately. From the ROC Curves, it can be seen that it can reliably detect HTTP flooding attacks with traffic intensity as low as five percent of the background traffic and even if the attack traffic intensity increases this architecture is still able to maintain high detection rate and low false alarm rate.

CHAPTER 7

CONCLUSION: STUDY OF INTRUSION DETECTION SYSTEM

In today's world almost every company is dependent on the Internet to survive, so it is not surprising that the role of network intrusion detection has grown so rapidly. While there may still be some argument as to what is the best way to protect a companies networks (i.e., firewalls, patches, intrusion detection, training), it is certain that the intrusion detection system (IDS) will likely maintain an important role in providing for a secure network architecture. However, recently the traditional intrusion detection systems have not been meeting the needs of the workload of today's networks. With high speed emerging computing technology, in order for intrusion detection systems to survive, they must keep up with the evolving new age networks.

REFERENCES

1. Z. Zhang, J. Li, C. Manikopoulos, J. Jorgeson. "Neural Networks in Statistical Anomaly Intrusion Detection," *Neural Network World,* vol. 11 no. 3, pp. 305-316, 2000.

2. Z. Zhang, J. Li, C. Manikopoulos, J. Jorgeson. "A Hierarchical Anomaly Network Intrusion Detection System Using Neural Network Classification," *CD-ROM Proceedings of 2001 WSES International Conference on: Neural Networks and Applications (NNA '01)*, February 2001.

3. J. Li, C. Manikopoulos. "Anomaly Intrusion Detection for Hierarchical Network Architecture," 2000.

4. Hervé Debar , Marc Dacier , Andreas Wespi, "Towards a taxonomy of intrusion-detection systems," *Computer Networks: The International Journal of Computer and Telecommunications Networking*, vol. 31 no. 9, pp. 805-822, April 23, 1999.

5. Z. Zhang, J. Li, C.N Manikopoulos, J. Jorgeson, J. Ucles, "HIDE: a hierarchical network intrusion detection system using statistical preprocessing and neural network classification," *Proceedings IEEE Workshop on Information Assurance and Security*, pp. 85-90, 2001.

6. Richard P. Lippmann, & David J. Fried. "Evaluating Intrusion Detection Systems: The 1998 DARPA Off-line Intrusion Detection Evaluation," *IEEE*, 1999.

7. Susan C. Lee, David V. Heinbuch, "Training a neural-network based intrusion detector to recognize novel attacks," *IEEE Transactions on Systems, MAN, and cybernetics –Part A: Systems and Humans,* vol. 31, no. 4, 2001.

8. Richard Barber, "The Evolution of Intrusion Detection Systems – The next step," *Computers and Society*, vol. 20, no 2, pp. 132-145, 2001.

9. Stephen Northcutt. *Network Intrusion Detection – An Analyst's Handbook Second Edition.* Indianapolis: New Riders Publishing, 2001.

10. Seth E. Webster, "The Development and Analysis of Intrusion Detection Algorithms," M.S. thesis, Massachusetts Institute of Technology, Cambridge, MA, 1998.

11. Jonathan Korba, "Windows NT Attacks for the Evaluation of Intrusion Detection Systems," M.S. thesis, Massachusetts Institute of Technology Department of Electrical Engineering and Computer Science, Cambridge, MA, June 2000.

12. Kristopher Kendall, "A database of Computer Attacks for the Evaluation of Intrusion Detection Systems," M.S. thesis, Massachusetts Institute of Technology, Cambridge, MA, June 1999.

13. John McHugh, Alan Christie, "Defending Yourself: The role of intrusion detection systems." *IEEE Software,* September/October 2000.