BORANG PENGESAHAN STATUS TESIS*

JUDUL: P	ACKET ANALYZER W	ITH MULTIPLE ARTIFICIAL NEURAL
<u>NETWOR</u>	K BACK-PROPAGATIO	ON TRAINING
SESI PEN	GAJIAN: <u>2008</u>	
Saya <u>LIM</u>	SIEW YUEN	
	1	(HURUF BESAR)
mengaku i Fakulti Te	membenarkan tesis(PSM/ eknologi Maklumat dan K	Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Komunikasi dengan syarat-syarat kegunaan seperti berikut:
2.3.	Perpustakaan Fakulti membuat salinan untuk Perpustakaan Fakulti	hakmilik Universiti Teknikal Malaysia Melaka. Teknologi Maklumat dan Komunikasi dibenarkan tujuan pengajian sahaja. Teknologi Maklumat dan Komunikasi dibenarkan ni sebagai bahan pertukaran antara institusi pengajian
	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub didalam AKTA RAHSIA RASMI 1972)
	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
	TIDAK TERH	IAD 1
7:1	Mer Car	Jan les
(TANDATANGAN PENULIS)		(TANDTANGAN PENYELIA)
Alamat te	tap: No 1, Lorong 3,	En. Nazrulazhar bin Hj Bahaman
Jalan PE 6	6, Taman Paya Emas,	Nama Penyelia
76450 Me	elaka.	_
Tarikh: 30	0/03/2008	Tarikh: 2 - 5 - 08

CATATAN: *Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda(PSM)

** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak
berkuasa

PACKET ANALYZER WITH MULTIPLE ARTIFICIAL NEURAL NETWORK BACK-PROPAGATION TRAINING

LIM SIEW YUEN

This report is submitted in partial fulfillment of the requirements for the Bachelor of Computer Science (Computer Network)

FACULTY OF INFORMATION AND COMMUNICATIONS TECHNOLOGY UNIVERSITI TEKNIKAL MALAYSIA MELAKA 2008

DECLARATION

I hereby declare that this project report entitled PACKET ANALYZER WITH MULTIPLE ARTIFICIAL NEURAL NETWORK BACK-PROPAGATION TRAINING

is written by me and is my own effort and that no part has been plagiarized without citations.

STUDENT	:	Date :02/08/
	(LIM SIEW YUEN)	
SUPERVISOR	: sold les-	Date : 2-5-08
	(FN NAZRIJI AZHAR BIN HI BAHAN	(AN)

DEDICATION

Specially dedicated to my beloved parents, Mr. Lim See Juan and Mrs. Choo An Na; and my belove husband, Mr. Tan Wei Tak. Thank You for Your fully supported.

For my supervisor, En. NazrulAzhar bin Hj.Bahaman (UteM)

And lastly to my entire friend who have encouraged, guided and inspired me throughout the journey for my project and also in my study.

ACKNOWLEDGEMENT

First and foremost, I would like to express my appreciation to Universiti Teknikal Malaysia Melaka (UTeM) for offering this courses, BITU 3973 Projek Sarjana Muda I (PSM I) and BITU 3983 Projek Sarjana Muda II (PSM II). The Projek Sarjana Muda (PSM) is compulsory for a UTeM student before being awarded the degree. The purposes of this PSM are to provide guidance and chances to students to develop a complete and individual project whereby students will encounter many new problems and challenges. Throughout this project, PSM will enhance the students' ability and skills in literature research, ability to analyse problems in various views and able to propose alternative solutions or models, ability to manage and utilize available resources in accomplishing the project and present the output effectively.

Besides, I would like to thank gracefully to Mr. NazrulAzhar Bin Hj.Bahaman, my supervisor who leads and guides me during this semester. The patience and generosity in guiding me through are much welcomed and appreciated. A note of thanks is especially dedicated to committees PSM on their kindness in organized briefing, seminar and talk to students who took this subject. Further more, I would like to thank all of the lecturers in UTeM that gives their cooperation during my PSM by providing me with the required needs. Without their cooperation, I would not be able to go through the phase smoothly.

Last but not least, I would like to thanks to my parents and my husband and also to all of my friends who are being kind and helpful throughout this special semester.

ABSTRACT

The packet analyzer with multiple artificial neural network back-propagation training is basically a system that integrated with the feed-forward backpropagation neural network to analyze the packet from the network. This system is proposed to overcome the intrusion that expands widely nowadays. This system using a system solution, which is the output of the training and using process using MATLAB, to integrated with the simple system developed using Visual Basic 6.0. The system can automatically identified and categorizes the packet captured into two categories, that is either normal or misuse packet. Neural networks are widely considered as an effective approach to classify patterns. In this project, the packet analyzer system will receive data packet from the network and the neural network will classify it into normal packet or misuse packet. For any occurrence of the possibility misuse packet, the system will alert the system administrator. Since neural network have a high learning rate, it will recognize most of the characteristics of attacks packet and by this, it can recognize, and block the packet before the misuse packet can get through the network. The advantage of using neural networks over statistics resides in having a simple way to express nonlinear relationships between variables, and in learning about relationships automatically. With few exceptions, behavior of most other users is also predictable. Neural networks are still a computationally intensive technique, and are not widely used in the packet analyzer community.

ABSTRAK

"Packet Analyzer with Multiple Artificial Neural Network Back-Propagation Training" adalah satu sistem yang mengesan penggemar computer yang melakukan "hacking process" ke atas rangkaian komputer. Sistem ini diperkenalkan dengan tujuan untuk mengesan penggemar komputer ini dan mengurangkan aktiviti "hacking" ke atas rangkaian komputer. Sistem ini menggunakan sistm penyelesaian yang didapati daripada pelajaran dan ujian yang dilakukan dengan menggunakan MATLAB. Sistem ini akan automatic mengenal dan mengkategorikan "packet" sama ada dalam keadan normal atau tidak normal. Keadaan adalah tidak normal dimana penggemar komputer melakukan "hacking" kepada rangkaian komputer. Dalam sistem ini, "packet analyzer system" akan menerima "packet" dari rangkaian dan "neural network" yang digabungkan dengan sistem tersebut akan menkategorikannya kepada biasa atau tidak biasa. Untuk keadaan di mana packet yang tidak biasa memasuki rangkaian sesebuah organisasi, system tersebut akan memberitahu pentadbir organisasi tersebut. "Neural network" mempunyai kadar belajar yang tinggi, oleh itu, system tersebut akan mengecam ciri-ciri "packet" yang disyaki daripada penggemar komputer.

TABLE OF CONTENTS

CHAPTER	SUBJECT	PAGE
	DECLARATION	i
	DEDICATION	ii
	ACKNOWLEDGEMENTS	iii
	ABSTRACT	iv
	ABSTRAK	\mathbf{v}
	TABLE OF CONTENTS	viii
	LIST OF TABLES	xi
	LIST OF FIGURES	XV
	LIST OF ABBREVIATION	xvii
	LIST OF APPENDICES	xviii
CHAPTER 1	INTRODUCTION	
	1.1 Project Background	1
	1.2 Problem Statements	3
	1.3 Objective	5
	1.4 Scopes	5
	1.5 Project Significance	6
	1.6 Expected Output	7
	1.7 Conclusion	8
CHAPTER II	LITERATURE REVIEW AND	
	PROJECT METHODOLOGY	
	2.1 Introduction	10

.2	Facts	And Findings	
	2.2.1	Artificial Neural Networks	11
	2.2.2	MATLAB	12
	2.2.3	Architecture of Generalized	14
		Network Service Anomaly	
		and Fault Thresholds and	
		Detection	
	2.2.4	Artificial Neural Networks for	16
		Misuse Detection	
	2.2.5	The 'Dark Ages' or the Birth	18
		of Artificial Intelligence	
		(1943-56)	
	2.2.6	How to Make a Machine	19
		Learn, or the Rebirth of	
		Neural Networks (1980-	
		onwards)	
	2.2.7	Technique	20
	2.2.7	'.1 Feedforward Back-	20
		propagation neural network	
	2.2.7	.2 Self Organizing Map (SOM)	23
	2.2.7	.3 Radial Basis Function	23
		Network (RBF)	
	2.2.7	.4 Hopfield Neural Network	23
	2.2.7	.5 Boltzmann Machine	24
	2.2.7	.6 Stohastic Neural Networks	24
	2.2.7	.7 Recurrent Network	24
	2.2.7	.8 Echo State Network (ESN)	25
	2.2.7	.9 Long Short Term Memory	25
	2.2.7	.10 Committee Machines	25
	2.2.7	.11 Associative Neural Network	25
	2.2.7	.12 Perceptron	26

2.3	Project Methodology	26
2.4	Project Requirements	30
	2.4.1 Software Requirements	30
	2.4.2 Hardware Requirements	32
	2.4.3 Other Requirements	32
2.5	Project Schedule And Milestones	32
2.6	Conclusion	37
CHAPTER III AN	ALYSIS	
3.1	Introduction	39
3.2	Problem Analysis	40
3.3	Requirement Analysis	42
	3.3.1 Data Requirements	43
	3.3.2 Functional Requirement	47
	3.3.2.1 Functional Requirement	52
	Use Case View	
	3.3.3 Non-functional Requirement	53
	3.3.4 Other Requirement	54
3.4	Quality of Data (Types of Data)	56
3.5	Conclusion	58
CHAPTER IV DE	SIGN	
4.1	Introduction	59
4.2	High Level Design	60
	4.2.1 System Architecture	60
	4.2.2 User Interface Design	63
	4.2.2.1 Navigation Design	67
	4.2.2.2 Input Design	69
	4.2.2.3 Output Design	70
4.3	Detailed Design	71
	4.3.1 Software Specification	71
	4.3.2 Software Specification	72

	4.4	Conclusion	74
CHAPTER V	IMI	PLEMENTATION	
	5.1	Introduction	76
	5.2	Software Development Environment	77
		Setup	
	5.3	Software Configuration Management	78
		5.3.1 Configuration Environment	79
		Setup	
		5.3.1.1 Hardware and Network	79
		Setup	
		5.3.1.2 Software Setup	80
		5.3.2 Version Control Procedure	81
	5.	4 Implementation Status	84
	5.	5 Conclusion	85
CHAPTER V	I TE		
	6.1	Introduction	87
	6.2	Test Plan	88
		6.2.1 Test Organization	88
		6.2.2 Test Environment	89
		6.2.3 Test Schedule	90
	6.3	Test Strategy	90
		6.3.1 Classes of Test	91
	6.4	Test Design	93
		6.4.1 Test Description	93
		6.4.2 Test Data	94
	6.5	Test Results and Analysis	97
	6.6	Conclusion	112

CHAPTER VII PROJECT CONCLUSION

7.1	Observation on Weakness and	113
	Strengths	
7.2	Propositions for Improvement	114
7.3	Contribution	116
7.4	Conclusion	117
RE	FERENCES	119
ΔĐΙ	PENDICES	122

LIST OF TABLES

TAB	LE TITLE	PAGE
2.1	Activity of each of the Waterfall Phases	29
2.2	Project Schedule and Milestones PSM I	33
2.3	Project Schedule and Milestones PSM II	35
3.1	Sample of pre-process Data	46
3.2	Elaboration of Sample data before pre-processing	46
3.3	Sample of pre-processed Data	46
3.4	Elaboration of sample data pre-processed	47
3.5	Artificial Neural Network Intrusion Detection	52
	System Use Case	
4.1	Input Designs for Intrusion Detection System	69
4.2	Browsing File Class Description	73
4.3	Option Menu Class Description	73
4.4	Detection Class Description	74
5.1	Version of Artificial Neural Network Intrusion	82
	Detection System	
5.2	Artificial Neural Network Intrusion Detection	83
	System Version 1.1	
5.3	Artificial Neural Network Intrusion Detection	84
	System Version 1.2	
5.4	Artificial Neural Networks Intrusion Detection	85
	System Implementation Status	

6.1	Summarize of Personnel's Responsibility	89
6.2	Test Schedule	90
6.3	Test Description	94
6.4	Test Input Data: Data from Real Environment	95
6.5	Test Input Data: DARPA Data (Not Used for	96
	Training Solution)	
6.6	Test Input Data: DARPA Data (Used to Train	97
	Solution)	
6.7	Test Result and Analysis: System Solution	98
6.8	Test Result and Analysis: Input Packet Data	99
6.9	Test Result and Analysis: Display Record	100
6 10	Test Desult and Analysis: Detection	101

LIST OF FIGURES

FIG	GURE TITLE	PAGE
2.1	Fault Management Agent	15
2.2	The Waterfall Model	27
3.1	Use-Case Models of Intrusion Detection System	40
3.2	Packet Structure	44
3.3	Flowchart on Artificial Neural Network Training Algorithm	48
3.4	Flowchart on Artificial Neural Network Testing	49
	Algorithm	
3.5	Flowchart on Step for Creating Intrusion	56
	Detection System using Artificial Neural Network	
3.6	Packet Analyzer Using Multiple Artificial Neural	58
	Network Back-Propagation Use Case View	59
4.1	Architecture for Packet Analyzer Using Multiple	59
	Artificial Neural Network Back-Propagation	59
	Training	60
4.2	Interfaces for Packet Analyzer Using Multiple	60
	Artificial Neural Network Back-Propagation	60
	Training	62
4.3	Interface for Packet Detail	
4.4	Interface for Packet Analyzer	65
4.5	Interface for Exit System	
4.6	Navigation Design of Packet Analyzer using	66
	Multiple Artificial Neural Networks.	
4.7	Flowcharts for Packet Analyzer Using Multiple	

	Artificial Neural Network Back-Propagation	71
	Training	88
4.8	Class Diagram of Packet Analyzer Using Multiple	
	Artificial Neural Network Back-Propagation	
	Training	
5.1	Software Development Environment	
6.1	Data Capture through the Network (Real Time)	95
6.2	1 st Training Data Process	102
6.3	2 nd Training Data Process	103
6.4	3 rd Training Data Process	104
6.5	4 th Training Data Process	105
6.6	5 th Training Data Process	106
6.7	1 st Testing Data Process	107
6.8	2 nd Testing Data Process	108
6.9	3 rd Testing Data Process	109
6.10	4 th Testing Data Process	110
6 11	5th Testing Data Process	111

CHAPTER I

INTRODUCTION

1.1 Project Background

Due to a growing number of intrusion events and also because the Internet and local networks have become so ubiquitous, organizations are increasingly implementing various systems that monitor IT security breaches. Unfortunately, computer and network intrusions have become more common and more complicated, challenging the intrusion detection systems. As a consequence, the amount of data to be processed by an intrusion detection system has been growing, making it difficult to efficiently detect intrusions online. This condition leads most of the system administrators and information security researchers' goal to solve the problem by develops more and more effective intrusion detection system.

In an information system, intrusions are the activities that violate the security policy if the system, and intrusion detection is the process used to identify intrusions. Commonly, intrusion detection techniques composed of one of two methodologies which is anomaly detection or misuse detection. Anomaly detection is based on the normal behavior of a subject (E.G., a user or a system); any action that significantly deviates from the normal behavior is considered intrusive. While misuse detection catches intrusion in terms of the characteristics of known attacks or system

vulnerabilities; any action that conforms to the pattern of a known attack or vulnerability is considered intrusive.

Intrusion Detection System (IDS) monitors network traffic and monitors for suspicious activity and alerts the system or network administrator. In some cases, the intrusion detection system may also respond to anomalous or malicious traffic by taking action such as blocking the user or source IP address from accessing the network. Intrusion detection system approaches the goal of detecting suspicious traffic in different ways. There are network based (NIDS) and host based (HIDS) intrusion detection systems. There is also intrusion detection systems that detect based on looking for specific signatures of known threats similar to the way antivirus software typically detects and protects against malware and there are intrusion detection system that based on comparing traffic patterns against a baseline and looking for anomalies.

Besides, intrusion detection system also differs in whether they are online or offline. Offline intrusion detection systems are run periodically and they detect intrusions after the fact based on system logs. While online systems are designed to detect intrusions while they are happening, thereby allowing for quicker intervention. Online intrusion detection systems are computationally very expensive because they require continuous monitoring. Decisions need to be made quickly with less data and therefore they are not as reliable.

Host based intrusion detection system (HIDS) can be used to determine if a system has been compromised and can warn administrator if that happens. There are four methods of host-based intrusion detection which includes file system monitoring (systems checking the integrity of files and directories), log file analysis (systems analyzing log files for patterns indicating suspicious activity), connection analysis (systems that monitor connection attempts to and from a host), and kernel-based intrusion detection (systems that detect malicious activity on a kernel level). Implementations of intrusion detection systems generally use one of the four methods to detect intrusions.

A network intrusion detection system (NIDS) is an intrusion detection system that tries to detect malicious activity such as denial of service attacks; port scans or even attempts to crack into computers by monitoring network traffic. The network intrusion detection system does this by reading all the incoming packets and trying to find suspicious patterns. Besides, a network intrusion detection system is not limited to inspecting incoming network traffic only. Network intrusion detection system work well with other system, such as update some firewall's blacklist with the IP address of computers used by suspected crackers.

The artificial neural networks provide a number of advantages in the detection of network intrusions. The application of the neural network techniques has been considered for both the misuse detection model and the anomaly detection model. An increasing amount of research has been conducted on the application of neural networks for detecting network intrusions. The artificial neural networks have the potential to resolve a number of problems encountered by the other current approaches in intrusion detection. The neural networks gain experience by training the system to correctly identify the pre-selected examples of the problem.

1.2 Problem Statements

A computer system should provide confidentiality, integrity and assurance against denial of service. However, due to increased connectivity (especially on the Internet), and the vast spectrum of financial possibilities that are opening up, more and more systems are subject to attack by intruders. These subversion attempts try to exploit flaws in the operating system as well as in application programs and have resulted in spectacular incidents like the Internet Worm incident of 1988.

However, intrusions can be divided into the main six types, which composed of attempted break-ins, which are detected by atypical behavior profiles or violations of

security constraints; masquerade attacks, which are detected by atypical behavior profiles or violations of security constraints; penetration of the security control system, which are detected by monitoring for specific patterns of activity; leakage, which is detected by atypical use of system resources; denial of service, which is detected by atypical use of resources; and last but not least, malicious use, which is detected by atypical behavior profiles, violations of security constraints, or use of special privileges.

Host-based intrusion detection system can analyze activities on the host it monitors at a high level of detail, and it can often determine which processes and users are involved in malicious activities. Though they may focus on a single host, many host-based intrusion detection systems use an agent-console model where agents run on individual hosts but report to a single centralized console so that a single console can configure, manage, and consolidate data from numerous hosts. Host based intrusion detection system can detect attacks undetectable to the network based intrusion detection system and can gauge attack effects quite accurately. Host based intrusion detection system can use host based encryption services to examine encrypted traffic, data, storage, and activity. Last but not lease, host based intrusion detection system have no difficulties operating on switch based networks. On the other hand, there are some disadvantages of host based intrusion detection system which is data collection occurs on a per-host basis, writing to logs or reporting activity requires network traffic and this can decrease network performance. In addition, a host based intrusion detection system does consume processing time, storage, memory, and other resources on the hosts where such systems operate.

With the growth of computer networking, electronic commerce, and web services, security of networking systems has become very important. Many companies now rely on web services as a major source of revenue. Computer hacking poses significant problems to these companies, as distributed attacks can render their cyber-storefront inoperable for long periods of time. As a result, intrusion detection system is devoted to detecting this activity by detecting this attack by a careful analysis of network data.

1.3 Objective

Project objective define target status at the end of the project, reaching of which is considered necessary for the achievement of planned benefits. There are several objectives in order to develop this project and this including: -

- ✓ To provide an accurate, extendable and adjustable to make sure the security of the computer system are confidentiality, integrity and able to prevent intrusion or attacking from the intruders.
- ✓ To propose and investigate a neural network based packet analyzer so that it can analyze the packet whether it is misuse or normal packet.
- ✓ To uses a feed-forward back-propagation neural network to learn users' behavior.
- ✓ To train and test the data by using feed-forward back-propagation neural network.

1.4 Scope

Project scope is the sum total of all projects products and their requirements or features. This project's scope including: -

✓ Developed the mathematical function using the software called Math Works MATLAB to create an algorithm which will train the data to be use in packet analyzer with multiple artificial neural network back-propagation training.

- ✓ Classified data from DARPA into two categories which is normal packet and misuse packet and train both of the packet.
- ✓ Convert the artificial neural network algorithm in MATLAB to VB language so that it can integrate in the intrusion detection system that have been created using VB so that the system can identify the normal and misuse packet.

1.5 Project Significance

This project is significance in nowadays computer's world because of the security incidents that occur over the network. These attacks may consist of otherwise authorized users who are disgruntled employees. The remainder comes from the outside, in the form of denial of service attacks or attempts to penetrate a network infrastructure. Intrusion detection systems remain the only proactive means of detecting and responding to threats that stem from both inside and outside a corporate network.

The aim of packet analyzer using multiple artificial neural network back-propagation training is to integrate the neural network techniques in the intrusion detection system. As a result, the system can automatically identified and categorizes the packet captured into two categories, that is either normal or misuse packet. Neural networks are widely considered as an effective approach to classify patterns. In this project, the intrusion detection system will receive data packet from the network and the neural network will classify it into normal packet or misuse packet. For any occurrence of the possibility misuse packet, the system will alert the system administrator. Since neural network have a high learning rate, it will recognize most of the characteristics of attacks packet and by this, it can recognize, and block the packet before the misuse packet can get through the network. The advantage of using neural networks over statistics resides in having a simple way to express nonlinear relationships between

variables, and in learning about relationships automatically. With few exceptions, behavior of most other users is also predictable. Neural networks are still a computationally intensive technique, and are not widely used in the intrusion detection community.

1.6 Expected Output

This project is design to suite the requirements in the intrusion detection system that needed by most of the organizations and corporate since the intrusion scenarios becomes more and more strictness. The packet analyzer using multiple artificial neural network back-propagation training will function in three steps which consist of packet capturing, packet preprocessing and intrusion or misuse detection.

First of all, third party software will be used to capture packets from network. In this project, Packet Sniffer and Wireshark will be used as the third party software. The TCP/IP streams of packet will be saving in a text file (.txt) and can be view using notepad or text editor.

The second step is preprocessing. Neural network training can be made more efficient by performing preprocessing steps on the network inputs and targets. Network-input processing functions transform inputs into a better form for the network use. Processing functions associated with a network output transform targets into a better form for network training, and reverse transformed outputs back to the characteristics of the original target data. In this project, the packet that has been captured will get through the preprocessing process and the packet will be analyzed to characterize the normal packet and the misuse packet. Normal packet is the packets with the normal data and information while misuse packet is the packet that containing suspicious intrusive information.

In order to prepare the data for training and testing process in neural network, the data will get through three levels of preprocessing. First, ten of the event record data elements will be selected in the data set. The elements are selected because its can supply a complete description of the information that transmitted by the packet. Next, a few elements in the network data packets will be converts into a standardized numeric representation. The data types and the sequential numbers that assigned to each element are categorized in the relational tables. Last, the results of the query will be converting into an ASCII comma delimited format that is needed in order to train the neural network.

The training process of artificial neural network will produced the output in binary form that is 0 and 1. The 0 represents the normal packets while 1 represents the misuse packets. The diverse connection weights were frozen and the network was interrogated after the training and testing process of the neural network. Sample data from DARPA with one normal packet and one misuse packet will be used to test the neural network. As a result of the testing process, a simulate graph will be shown regarding the normal and misuse network. For the result below 0.1, the packet is considered as normal and if the result is above 0.1, the packet will be consider as misuse.

After the test is prove correct, the solution will be translated into Visual Basic language. It will be merging with a system interface that developed using Visual Basic to deliver an alert system that can identified misuse in network.

1.7 Conclusion

Overall, this chapter is mainly explaining on the project background of developing an artificial neural network for intrusion detection system. The objectives, project scopes, project significance also covered in this chapter. The aim of this project is to develop a system using artificial neural network which can detect the attacking of intruders into the