

ANALYZING DNS QUERY AND RESPONSE

AHMAD FAIZ BIN MOHD

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

BORANG PENGESAHAN STATUS TESIS

JUDUL: ANALYZING DNS QUERY AND RESPONSE

SESI PENGAJIAN: 2009/2010

Saya AHMAD FAIZ BIN MOHD

(HURUF BESAR)

mengaku membenarkan tesis (PSM/Sarjana/Doktor Falsafah) ini disimpan di Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan Fakulti Teknologi Maklumat dan Komunikasi dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. ** Sila tandakan (/)

_____ SULIT

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

_____ TERHAD

(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/ badan di mana penyelidikan dijalankan)

___/___ TIDAK TERHAD


(TANDATANGAN PENULIS)


(TANDATANGAN PENYELIA)

Alamat tetap: 15151, Desa Fitri,
Jln Teras Jernang,
Sungai Merab,
43650, Bdr Baru Bangi,
Selangor Darul Ehsan.

Dr Mohd Faizal bin Abdollah

Tarikh : 25 / 6 / 10

Tarikh : 25 / 6 / 10

CATATAN: * Tesis dimaksudkan sebagai Laporan Akhir Projek Sarjana Muda (PSM)
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa.

ANALYZING DNS QUERY AND RESPONSE

AHMAD FAIZ BIN MOHD


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

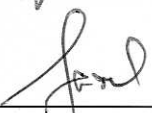
**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY
UNIVERSITI TEKNIKAL MALAYSIA MELAKA
2010**

DECLARATION

I hereby declare that this project report entitled
ANALYZING DNS QUERY AND RESPONSE

is written by me and is my own effort and that not part has been plagiarized
without citations.

STUDENT :  _____ Date: 25/6/10
(AHMAD FAIZ BIN MOHD)

SUPERVISOR :  _____ Date: 25/6/10
(DR MOHD FAIZAL BIN ABDOLLAH)

DEDICATION

To my beloved parents,
You're my place to pray for blessing,
To all my friends who were given me a moral support and also helped me,
A lot of appreciation to everyone involved,
To my supervisor and lecturers that helped me a lot,
Your good deeds will be remember,
Finally, thank you so much for everything,
May Allah repay and give all of us a blessing,
Thank you.

ACKNOWLEDGEMENTS

Alhamdulillah...

First of all, thank you to Allah because with award and permission, finally I had completed this project after going through various challenges and test.

In this opportunity, firstly I want to said a million of thanks and unlimited appreciation to lecturers especially Dr Mohd Faizal bin Abdollah as my supervisor that had give so much guidance, criticizing which built, give useful advice and never bored to aid me in every provision and compilation process about project. Without attention, support and guidance from him, it is impossible I can produce and completed this project nicely.

At the same time, I also want to record this much appreciation to my friends which already given so much solid moral support and already help a lot especially from energy aspect and spend time for me to complete this project.

Lastly, to my beloved family that had given much support too and always pray for my success. Without their helping, inducement and cooperation from various parties, impossible for me to finish this project in smoothly it ways. Every helping from all involved parties so much I appreciate.

ABSTRACT

This project is all about analyzing how DNS query and response back the requested domain name. The response type can be in many types. It is called Response Code (RCODE). For this project, three types of RCODE used which are non-existence domain (NXDomain), server fail (ServFail), and no error (NoErr). When the DNS server replied with NoErr RCODE, it means that the DNS request and response transaction successfully resolved because the name of domain name requested really exists. When the DNS server replied with NXDomain and ServFail RCODE, it means that the DNS request and response transaction fail to the requested domain name because the domain name really not exist or anything happen to their DNS server like internal error or server down. This project also developed one Java program that is using to execute the packet. When the packet executed, it will show what the behavior activity for each network is. The result got for behavior activity is the successful of this project because this project objective is to study and defining what the behavior activity for each network is.

ABSTRAK

Projek ini adalah berkaitan analisis tentang bagaimana DNS diminta dan direspons semula kepada sesuatu domain yang diminta. Terdapat beberapa jenis respons yang diberikan. Ia dikenali sebagai Kod Respons (RCODE). Bagi projek ini, tiga jenis respons digunakan iaitu non-existence domain (NXDomain), server fail (ServFail), dan no error (NoErr). Apabila server DNS memberi respons No Err RCODE, ia bermaksud suatu transaksi DNS telah berjaya dilakukan kerana nama domain yang diminta wujud. Apabila server DNS memberi respons NXDomain dan ServFail RCODE, ia bermaksud suatu transaksi DNS gagal dilakukan kerana nama domain yang diminta tidak wujud atau berlaku sebarang masalah pada server DNS. Projek ini juga dibangunkan dengan satu program Java yang digunakan untuk memproses paket yang dihasilkan. Apabila sesuatu paket telah diproses, keputusan akan diperolehi di mana akan menunjukkan apakah kelakuan aktiviti sesuatu rangkaian. Keputusan yang diperolehi bagi kelakuan aktiviti sesuatu rangkaian adalah kejayaan bagi projek ini kerana objektif utama projek ini adalah untuk menganalisis dan mentakrifkan apakah kelakuan aktiviti bagi sesuatu rangkaian.

TABLE OF CONTENTS

CHAPTER	SUBJECT	PAGE
	DECLARATION	ii
	DEDICATION	iii
	ACKNOWLEDGEMENTS	iv
	ABSTRACT	v
	ABSTRAK	vi
	TABLE OF CONTENTS	vii
	LIST OF TABLES	xii
	LIST OF FIGURES	xiii
	LIST OF ABBREVIATIONS	xviii
CHAPTER I	INTRODUCTION	
	1.1 Project Background	1
	1.2 Problem Statements	2
	1.3 Objectives	3
	1.4 Scopes	4
	1.4.1 Project Scope	4
	1.4.2 User scope	4
	1.4.3 Network Scope	4

1.5	Project Significance	5
1.6	Expected Output	5
1.7	Conclusion	6
CHAPTER II	LITERATURE REVIEW AND PROJECT METHODOLOGY	
2.1	Introduction	7
2.2	Literature Review	8
	2.2.1 Domain	8
	2.2.2 Keyword	9
	2.2.3 Previous Research	10
	2.2.3.1 DNS Query and Response	10
	2.2.3.2 DDoS attack using Open Resolver	13
	2.2.3.3 Tools used to detect or filter raw data	15
2.3	Proposed Solution	18
	2.3.1 Project Methodology	18
	2.3.1.1 Project Planning and Initiation	19
	2.3.1.2 Requirement Study and Analysis	20
	2.3.1.3 Design	20
	2.3.1.4 Coding and Testing	21
	2.3.1.5 Project Testing	21
2.4	Project Schedule and Milestones	23

	2.5	Conclusion	23
CHAPTER III		ANALYSIS	
	3.1	Introduction	24
	3.2	Problem Analysis	25
	3.2.1	Network Architecture	27
	3.2.2	Logical and Physical Design	28
	3.2.2.1	Logical Network Design	29
	3.2.2.2	Physical Network Design	31
	3.3	Requirement Analysis	34
	3.3.1	Hardware	35
	3.3.2	Software	37
	3.3.3	Quality of Data	38
	3.3.3.1	Journal	38
	3.3.3.2	TCPDump	39
	3.3.3.3	NetBeans IDE 6.7.1	40
	3.3.3.4	Testing Result	42
	3.4	Conclusion	45
CHAPTER IV		DESIGN	
	4.1	Introduction	46
	4.2	Possible Scenarios	46
	4.2.1	Abnormal Behavior	48
	4.2.2	Normal Behavior	49
	4.2.3	Overall Project Flowchart	50

	4.3	Conclusion	55
CHAPTER V		IMPLEMENTATION	
	5.1	Introduction	56
	5.2	Network Configuration Management	57
		5.2.1 Configuration Environment Setup	57
		5.2.2 Version Control Procedure	60
	5.3	Hardware Configuration Management	65
		5.3.1 Hardware Setup	65
	5.4	Development Status	66
		5.4.1 Normal Behavior Activity Results	67
		5.4.2 Abnormal Behavior Activity Results	79
		5.4.3 Results Comparison	92
	5.5	Conclusion	93
CHAPTER VI		TESTING	
	6.1	Introduction	94
	6.2	Test Plan	95
		6.2.1 Test Organization	98
		6.2.2 Test Environment	98
		6.2.3 Test Schedule	98
	6.3	Test Strategy	99
		6.3.1 Classes of tests	99
	6.4	Test Design	100

6.4.1	Test Data	100
6.5	Test Result and Analysis	101
6.5.1	Test Results	101
6.5.2	Analysis	109
6.6	Conclusion	110
CHAPTER VII	PROJECT CONCLUSION	
7.1	Observation on Weaknesses and Strengths	111
7.1.1	Weaknesses	111
7.1.2	Strengths	112
7.2	Propositions for Improvement	113
7.3	Contribution	114
7.4	Conclusion	115
REFERENCES		117
APPENDICES		119

LIST OF TABLES

TABLE	TITLE	PAGE
2.1	Keyword exist in this project	9
2.2	Detailed of query process	16
2.3	Detailed of response process	17
3.1	Abnormal behavior testing information	44
5.1	Project version 1	60
5.2	Project version 2	61
5.3	Project version 3	61
5.4	Hardware setup	65
6.1	Project version 3 details	97
6.2	Analysis result for all project versions	109

LIST OF FIGURES

DIAGRAM	TITLE	PAGE
2.1	Type of query message data	11
2.2	DNS query and response	13
2.3	TCPDump output for non-existence domain name	15
2.4	TCPDump output for ServFail responded	17
2.5	Slightly Modify Waterfall Methodology	18
2.6	Main task of this project	19
2.7	Waterfall methodology	22
3.1	Old approach of using TCPDump to read packet captured	25
3.2	New approach of using TCPDump to read packet captured	26
3.3	Abnormal behavior logical network design	29
3.4	Normal behavior logical network design	30
3.5	Abnormal behavior physical network design	31
3.6	Normal behavior physical network design	32
3.7	SPAN port	33
3.8	Cisco 2800 Series Integrated Services Routers	35
3.9	Cisco Catalyst 2950 Switch	35
3.10	HP Proliant DL160 G5 Server	36
3.11	D-Link DSL-520B Modem	36

3.12	HP 4400 Workstation	37
3.13	Executing file using zcat command	39
3.14	Read and convert file *.tcpdump to abnormal.pcap file	40
3.15	Java code created using NetBeans IDE 6.7.1	40
3.16	Output displayed	41
3.17	Main of Java Program	42
3.18	List of DNS transaction status	42
3.19	Status of all transaction	43
4.1	Abnormal behavior network diagram	49
4.2	Normal behavior network diagram	50
4.3	Overall project flowchart	51
4.4	Task of get packet to be analyze	52
4.5	Task of using TCPDump tools to filter the packet captured	53
4.6	Task of using Java program to execute the filtered packet	54
5.1	Abnormal Behavior Configuration Environment Setup	57
5.2	Normal Behavior Configuration Environment Setup	59
5.3	Context Diagram	62
5.4	Data Flow Diagram for Setup Abnormal Behavior Network	63
5.5	Data Flow Diagram for Setup Normal Behavior Network	63
5.6	Data Flow Diagram for Capture Traffic Packet	64
5.7	Data Flow Diagram for Check DNS Transaction	

	Status	64
5.8	Result from normal behavior packet captured on 3/2/2010	67
5.9	Web browser successfully open http://centos.maulvi.net	68
5.10	Result from normal behavior packet captured on 4/2/2010	69
5.11	Web browser successfully open http://apt.sw.be	70
5.12	Result from normal behavior packet captured on 5/2/2010	71
5.13	Web browser successfully open http://fr2.rpmfind.net	72
5.14	Result from normal behavior packet captured on 6/2/2010	73
5.15	Web browser successfully open http://mirror.aarnet.edu.au	74
5.16	Result from normal behavior packet captured on 7/2/2010	75
5.17	Web browser successfully open ftp://ftp.swin.edu.au	76
5.18	Result from normal behavior packet captured on 8/2/2010	77
5.19	Web browser successfully open Microsoft update website	78
5.20	Result from abnormal behavior packet captured on 27/1/2010	79
5.21	Address not found of edsaqmp.yi.org	80
5.22	Result from abnormal behavior packet captured on	

	28/1/2010	81
5.23	Failed to connect with domain DNS server	82
5.24	Web browser cannot found the http://www.fekhfvsfmf.net	83
5.25	Result from abnormal behavior packet captured on 29/1/2010	84
5.26	Proof of Kraken botnet	85
5.27	Result from abnormal behavior packet captured on 1/2/2010	86
5.28	Web Browser successfully opens the http://www.youtube.com	87
5.29	Result from abnormal behavior packet captured on 2/2/2010	88
5.30	Proof of botnet ontqxzy.dyndns.org domain name	89
5.31	Result from abnormal behavior packet captured on 3/2/2010	90
5.32	Proof of botnet zoipmnr.net domain name	91
5.33	Result comparison for normal behavior	92
5.34	Result comparison for abnormal behavior	92
6.1	Network diagram for normal and abnormal behavior	96
6.2	Process to produce dumpfile.pcap file	101
6.3	Process to produce dumpfile_baru.pcap file	101
6.4	Result for abnormal behavior activity happened	102
6.5	Request for xxqkofrft.yi.org domain name	103
6.6	Respond for xxqkofrft.yi.org domain name requested	103

6.7	Web browser cannot found the xxqkofrftte.yi.org domain name	104
6.8	Result for normal behavior activity happened	105
6.9	Request for lanset2007.com domain name	105
6.10	Respond for lanset2007.com domain name requested	106
6.11	Web browser successful open the lanset2007.com	106
6.12	Windows automatic updates function well	107
6.13	Request for download.windowsupdate.com domain name	107
6.14	Respond for download.windowsupdate.com domain name requested	108
6.15	Web site of windows automatic updates	108

LIST OF ABBREVIATIONS

ACRONYM		WORD
Botnet	–	Robot Network
DDoS	–	Distributed Denial of Service
DFD	–	Data Flow Diagram
DNS	–	Domain Name System
DSL	–	Digital Subscriber Line
DSP	–	Digital Signal Processor
FTMK	–	Fakulti Teknologi Maklumat dan Komunikasi
FQDN	–	Fully Qualified Domain Name
IN	–	Internet
IP	–	Internet Protocol
IPS	–	Intrusion Prevention System
LAN	–	Local Area Network
NoErr	–	No Error
NXDomain	–	Non-existence Domain
OS	–	Operating System
RCODE	–	Response Code
ServFail	–	Server Fail
SPAN	–	Switched Port Analyzer
TTL	–	Time to Live
QoS	–	Quality of Service
WAN	–	Wide Area Network

CHAPTER I

INTRODUCTION

1.1 Project Background

This project is concern about Domain Name System (DNS) query and response. It is about analyzing and capturing the DNS packets look when users send a query or request name resolution for the DNS server. When users send a request, the DNS server will receive the query and try to response the query immediately. If the DNS server does not have a response cached, it will refer to the root DNS server and consecutive name server until it receives a response. Then, the DNS server will cache the response until the Time to Live (TTL) expired and passes to the users. So, the point of successful DNS query and response is at its Response Codes (RCODE) and DNS Transaction ID.

For the RCODE, this project is basically looking for only three RCODE. The first RCODE is NXDomain. NXDomain is stands for non-existence domain. This RCODE will return fail result if it exist in the data that want to be analyze. The second RCODE is ServFail. ServFail is stands for Server Fail. These RCODE will also return fail results if it exists in the data that want to be analyze. The last RCODE is NoErr. NoErr is stands for No Error which means that the transaction request is successfully resolve by the DNS server.

DNS Transaction ID is the number that is representing each DNS query and response process. If the number of DNS Transaction ID is same, it means that the process of query and response are successful as one complete transaction. For example, if the DNS Transaction ID of query is 501010, then the DNS server will respond the same number. So, to get this DNS Transaction ID, TCPDump program is use to read the packet that contains the source IP address, destination IP address, port number, response time, and so on.

Next, one Java program has been developed to display all query made by users. The output contains of source IP address, destination IP address, domain name request, DNS Transaction ID, status of each transaction, and the total status of all transaction happen. Other than that, the flow of DNS packets using the proof of DNS Transaction ID is elaborate.

1.2 Problem Statements

This project is to prove whether there is abnormal behavior and normal behavior activity happen. Abnormal behavior happen if there is abnormal activity occurred that will bring the NXDomain and ServFail Response Codes (RCODE). If there is many NXDomain and ServFail replied, then it will assume that the abnormal behavior is happened. Normal behavior happens if there is only normal activity occurred that will bring the NoErr Response Codes (RCODE). If there is many NoErr replied, then it will assume that the normal behavior is happened.

Other than that, the problem to read the original packet captured also will be improved in this project. When the packet captured produced, there will be a problem to read the content of the packet captured. All users are difficult to read the original data because the data is not in sequence.

So, to solve this problem, one Java program will be developed to execute the data in the packet captured and produce the result. The result produced hopefully will help all users read the packet captured.

1.3 Objectives

- Able to analyze the behavior of DNS activity. This objective would be achieved with some explanation about DNS data flow. It will cover how the DNS flow from user made a query and DNS server resolve back the query.
- To develop a program using Java application. The program is use to execute the packet captured and display the status of each transaction. The status can be either success transaction or fail transaction.
- Able to identify the status of DNS query using a simple code. The simple code is a set of program using Java programming. It will display the output like source IP address, destination IP address, domain name request, DNS Transaction ID, status of each transaction, and the total status of all transaction happen.

1.4 Scopes

1.4.1 Project Scope

For this project, it focuses only for Domain Name System (DNS). It is use TCPDump tool to capture the DNS packet. Then, the DNS packet will be analyze using simple Java program and execute it to show the status of DNS query and response.

1.4.2 User Scope

For this project, all people that want to analyze the DNS query and response will be this project main scope. They will be introduced how to know the status of DNS packet more details.

1.4.3 Network Scope

This project will be using specific platform of Operating System (OS). This project will used Ubuntu Linux 9.0 as a platform to run the TCPdump program and execute a Java program.