KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY

DEPARTMENT OF COMPUTER SCIENCE

INVESTIGATING THE FIREWALL SECURITY AND NETWORK PERFORMANCE RELATIONSHIP IN A DISTRIBUTED SYSTEM

BY

FRANCIS KWADZO AGBENYEGAH (B.ED. COMPUTER SCIENCE)

A THESIS SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE, KWAME NKRUMAH UNIVERSITY OF SCIENCE AND TECHNOLOGY IN PARTIAL FUFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF M.PHIL INFORMATION TECHNOLOGY

APRIL, 2014.

DECLARATION

I hereby declare that this thesis is my own work towards the M.Phil. degree and that, to the best of my knowledge, it contains no material previously published by another person nor material which has been accepted for the award of any other degree of the University, except where due acknowledgment has been made in the text.

FRANCIS KWADZO AGBENYEGAH	……	…….
(PG8137612)	Signature	Date
Student Name & ID		

Certified by:

DR. M. ASANTE	……	…….
Supervisor's	Signature	Date
DR. M. ASANTE	……	…….
Head of Department	Signature	Date

ABSTRACT

This research work investigated the firewall security and performance relationship for distributed systems. Internet connectivity is growing with most enterprises shitting to the use of web based services for services provision. As enterprises take on the Internet as a new business tool whether to sell, to collaborate or to communicate - web applications have become the new weakest link in the organization's security strategy. Firewalls provide a mechanism for protecting these enterprises from the less secure internet over which customers or collaborating partners transfer packets destined for the corporate network.

The relation between the security and performance efficiency is presented through different scenarios and the relationship between security and performance in firewalls is evaluated.

We modeled networks with and without firewalls and different firewall functionality and simulated such networks with an eye on their performances. The simulation was done for 300 work stations and simulated in a way that all the 300 work stations access a database, ftp, email and web application under three different scenarios.

Emphasis is on the relationship between network security and performance; the effects of firewalls on network performance. Various scenarios were evaluated through simulations using OPNET IT Guru Academic Edition 9.1 to show the effects of firewalls on network performance. The result shows that the security is inversely related to network performance.

AKCNOWLEDGEMENT

I thank the Almighty God for granting me this opportunity and seeing me through the academic work.

I also thank my supervisor Dr. Michael Asante for his supervision and technical guidance throughout the study. May God reward him abundantly

I also express my profound gratitude to the staff and management of Kpando Mutual Health Insurance Scheme especially the IT department.

Last but not the least I would like to thank my brother Samuel Agbenyegah for his insightful, encouragement and dedication to the course of my studies. To my mum Grace Okyerewa Agbenyegah for her support and prayers and to my siblings Christine, Francisca and Richard for their prayers

DEDICATION

I DEDICATE THIS THESIS TO GOD ALMIGHTY. THE CREATOR AND GIVER OF LIFE AND KNOWLEDGE

TABLE OF CONTENT

DECLARATION.....ii
ABSTRACT..... iii
AKCNOWLEDGEMENT.....iv
DEDICATION.....v
TABLE OF CONTENT.....vi
LIST OF TABLE xi
LIST OF FIGURE xiii
LIST OF ABBREVIATIONS xv
CHAPTER 1.....1
1.1 Introduction.....1
1.2 Overview of Firewall Technologies.....3
1.2.1 Packet filter.....5
1.2.2 Stateful Inspection 7
1.2.3 Application Firewall 9
1.2.4 Application - Proxy Gateways.....10
1.2.5 Dedicated Proxy Servers.....12
1.2.6 Virtual Private Network.....12
1.2.7 Network Access Control.....14
1.2.8 Web Application Firewalls 15
1.2.9 Firewall for Virtual Infrastructures.....15
1.3 Firewall and Network Architecture 15
1.3.1 Network Layouts with Firewall 17
1.3.2 Firewall Acting as Network Address Translator 18
1.3.3 Architecture with Multiple Layers of Firewalls.....19
1.4 Firewall Policy.....21
1.4.1 Policies Based on IP Addresses and Protocols 21
1.4.2 IP Addresses and other Characteristics.....22
1.4.3 TCP AND UDP.....23
1.4.4 ICMP.....24
1.4.5 Policy Based on Network Activity 25
1.5 Problem Statement.....26
1.6 Objectives 26

1.7 Research Questions 27
1.8 Research Hypothesis 27
1.9 Significance of the study 28
1.10 Scope of the study 28
1.11 Research Methodology 28
1.12 Thesis Organization 29
CHAPTER 2 30
LITERATURE REVIEW 30
2.1 Introduction 30
2.2 Security Policy 30
2.2.1 Enforcing Security Policies 31
2.2.2 Policy Enforcement Problems 32
2.3 Network Security Threat 34
2.3.1 Worms, Viruses and Trojans 35
2.3.2 Network Scanning 35
2.3.3 User Privilege Gain 36
2.3.4 Denial of Service Attack 36
2.4 Summary 37
CHAPTER 3 38
METHODOLOGY 38
3.1 Introduction 38
3.2 OPNET IT Guru as a Simulation Tool 38
3.2.1 Project 39
3.2.2 Scenario 39
3.2.3 Object 40
3.2.4 Application Definition Config 41
3.2.5 Profile Definition Config 42
3.3 No Firewall Scenario 42
3.4 Firewall scenarios 45
3.5 Firewall: With Packet Filtering Capabilities scenarios 45
3.6 Simulation Procedure 45
3.6.1 Simulation of No Firewalls Scenario 45
3.6.2 Application Configuration 48

3.6.3 Profile Configuration 49
3.6.4 Cloud/Internet Configuration 51
3.6.5 Office Configuration 51
3.6.6 Server Configuration 52
3.6.7 Performance Metrics 54
3.6.8 Firewall Scenario 57
3.6.9 Firewall Blocking Scenario 59
3.7 Running the Simulation 59
3.8 Result of the Simulation Experiment 60
3.8.1 Result for Database Application 61
3.8.2 Database Query Response Time 61
3.8.3 Server Database Query Load 62
3.8.4 Result for E-mail Application 63
3.8.5 E-mail Download Response Time 63
3.8.6 E-mail Upload Response Time 65
3.8.7 Server E-mail Load 66
3.8.8 Result Web Application 68
3.8.9 Http Page Response Time 68
3.8.10 Http Server Load 69
3.8.11 Result for Ftp Application 71
3.8.12 Ftp Download Response Time 71
3.8.13 Ftp Upload Response Time 72
3.8.14 Server Ftp Load 74
3.8.15 Cloud Performance 75
3.9 Conclusion 76
CHAPTER 4 77
RESULTS AND EVALUATION 77
4.1 Introduction 77
4.2 Result for Database Application 77
4.2.1 Database Query Response Time - No Firewall Scenario 78
4.2.2 Database Query Response Time - Firewall Blocking Scenario 80
4.2.3 Server Database Query Load 81
4.2.4 Server Database Query Load - No Firewall Scenario 81

4.2.5 Server Database Query Load - Firewall Scenario....82
4.2.6 Server Database Query Load - Firewall Blocking....83
4.3 Result for E-mail Application....85
4.3.1 E-mail Download Response Time – No Firewall Scenario....85
4.3.2 E-mail Download Response Time – Firewall Scenario....86
4.3.3 E-mail Download Response Time – Firewall Blocking Scenario....88
4.3.4 E-mail Upload Response Time – Firewall Scenario....89
4.3.5 E-mail Upload Response Time – Firewall Blocking Scenario....90
4.3.6 Server E-mail Load....92
4.3.7 Server E-mail load – Firewall Scenario....93
4.3.8 Server E-mail load – Firewall Blocking Scenario....94
4.4 Result for Web Application....95
4.4.1 Http Page Response Time - No Firewall Scenario....95
4.4.2 Http Page Response Time - Firewall Scenario....96
4.4.3 Http Page Response Time - Blocking Firewall Scenario....97
4.4.4 Server Http Load....99
4.4.5 Server Http Load - No Firewall Scenario....99
4.4.6 Server Http Load - Firewall Scenario....100
4.4.7 Server Http Load - Firewall Blocking Scenario....101
4.5 Result for Ftp Application....102
4.5.1 Ftp Download Response Time – No firewall Scenario....102
4.5.2 Ftp Download Response Time - Firewall Scenario....103
4.5.3 Ftp Download Response Time – Firewall Blocking Scenario....104
4.5.4 Ftp Upload Response Time....106
4.5.5 Ftp Upload Response Time – No Firewall Scenario....106
4.5.6 Ftp Upload Response Time – Firewall Scenario....107
4.5.7 Ftp Upload Response Time – Firewall Blocking Scenario....108
4.5.8 Server Ftp Load....109
4.6 Cloud Performance....110
CHAPTER 5....112
FINDING, CONCLUSION, AND RECOMMENDATION....112
5.1 Findings....112
5.2 Conclusion....113

5.3 Recommendation 113
REFERENCES..... 114

LIST OF TABLE

Table 1.1: State Table 8

Table 3.1: database query response time with packet size of 32MB (Low) 61

Table 3.2: database query response time with packet size of 100MB (Medium) 61

Table 3.3: database query response time with packet size of 200MB (High) 62

Table 3.4: server database load with packet size of 32MB (Low) 62

Table 3.6: server database load with packet size of 200MB (High) 63

Table 3.7: E-mail downloads response time with packet size of 32MB (Low) 64

Table 3.8: E-mail downloads response time with packet size of 100MB (Medium) 64

Table 3.9: E-mail downloads response time with packet size of 200MB (High) 64

Table 3.10: E-mail Upload Response Time with packet size of 32MB (Low) 65

Table 3.11: E-mail Upload Response Time with packet size of 100MB (Medium) 65

Table 3.12: E-mail Upload Response Time with packet size of 200MB (High) 66

Table 3.13: E-mail server load with packet size of 32MB (Low) 66

Table 3.14: E-mail server load with packet size of 100MB (Medium) 67

Table 3.15: E-mail server load with packet size of 200MB (High) 67

Table 3.16: page response time with packet size of 32MB (Low) 68

Table 3.17: page response time with packet size of 100MB (Medium) 68

Table 3.18: page response time with packet size of 200MB (High) 69

Table 3.19: server http load with packet size of 32MB (Low) 69

Table 3.20: server http load with packet size of 100MB (Medium) 70

Table 3.21: server http load time with packet size of 200MB (High) 70

Table 3.22: Ftp downloads Response Time with packet size of 32MB (Low) 71

Table 3.23: Ftp downloads Response Time with packet size of 100MB (Medium) 71

Table 3.24: Ftp downloads Response Time with packet size of 200MB (High).....72

Table 3.25: Ftp uploads Response Time with packet size of 32MB (Low) 72

Table 3.26: Ftp uploads Response Time with packet size of 100MB (Medium) 73

Table 3.27: Ftp uploads Response Time with packet size of 200MB (High).....73

Table 3.28: Server Ftp load with packet size of 32MB (Low) 74

Table 3.29: Server Ftp load with packet size of 100MB (Medium).....74

Table 3.30: Server Ftp load packet size of 200MB (High).....75

Table 3.31: cloud utilization with packet size of 32MB (Low).....75

Table 3.32: cloud utilization with packet size of 100MB (Medium).....76

Table 3.33: cloud utilization with packet size of 200MB (High) 76

LIST OF FIGURE

Fig.1.1: Firewall Operation and Data Flow 5
Fig.1.2: Simple Routed Network with Firewall Device 17
Fig.3.1: Project Editor Window 39
Fig. 3.2: Scenario 40
Fig.3.3: Object Palette 41
Fig.3.4: Application Attribute 41
Fig. 3.6: OPNET Startup Screen 43
Fig. 3.7: New Project 43
Fig .3.8: No Firewall Scenario 46
Fig.3.9: Workspace for Network Definition 47
Fig.3.10: Network Layout 48
Fig.3.11: Application Configuration 49
Fig.3.12: Database Profile Config 50
Fig.3.13: Web Profile Config 50
Fig.3.14 Ftp Profile Config 50
Fig.3.15 Email Profile Config 50
Fig.3.16: Cloud/Internet Configuration 51
Fig.3.17: Office Configuration 52
Fig.3.18: Database Server Configuration 53
Fig.3.19: Web Server Configuration 53
Fig.3.20 File Server Configuration 53
Fig.3.21: Three Level Performance Metrics 54
Fig.3.22: Global Statistics Performance Metrics 55
Fig.3.23: Node Level Performance Statistics 56
Fig.3.24: Link Level performance Statistics 56
Fig.3.25: Duplicate Scenario 57
Fig.3.26: Firewall Configuration 58
Fig.3.27: Firewall Scenario Setup 58
Fig.3.28: Web Traffic Block 59
Fig.3.29: Manage Scenarios 60
Fig.4.1: Database Query Response Time- No Firewall Scenario 78
Fig.4.2: Database Query Response Time-Firewall Scenario 79

Fig.4.3: Database Query Response Time-Firewall Blocking Scenario 81
Fig.4.4: Server Database Query Load- No Firewall Scenario 82
Fig.4.5: Server Database Query Load- Firewall Scenario 83
Fig.4.6: Server Database Query Load- Firewall Blocking Scenario 84
Fig.4.7: Server Database Query Load.....85
Fig.4.8: E-mail download response time- No Firewall Scenario.....86
Fig.4.9: E-mail download response time- Firewall Scenario.....87
Fig.4.10: Email Upload Response Time – No Firewall Scenario.....88
Fig.4.11: E-mail upload response time- Firewall Scenario.....90
Fig.4.12: E-mail upload response time- Firewall Blocking Scenario.....91
Fig.4.13: E-mail upload response time 91
Fig.4.14: Server E-mail Load – No Firewall Scenario 92
Fig.4.15: Server E-mail Load – Firewall Scenario 93
Fig.4.16: Server E-mail Load.....94
Fig.4.17 Http Page Response Time – No firewall Scenario 96
Fig.4.18: Http Page Response Time – Firewall Scenario 97
Fig.4.19: Http Page Response Time- Firewall Blocking 98
Fig.4.20: Http Page Response Time 98
Fig.4.21: Http Server Load- No Firewall Scenario.....99
Fig.4.22 :Http Server load- Firewall Scenario 100
Fig.4.23: Http Server Load 101
Fig.4.24: Ftp Download Response Time – No Firewall Scenario 102
Fig.4.25: Ftp Download Response Time – Firewall Scenario 103
Fig.4.26: Ftp Download Response Time – Firewall Blocking Scenario 104
Fig.4.27: Ftp Download Response Time 105
Fig.4.29: ftp upload response time – Firewall scenario 107
Fig.4.30: ftp upload response time – Firewall Blocking scenario 108
Fig.4.31: ftp upload response time.....109
Fig.4.32: Cloud Point to Point utilization 110

LIST OF ABBREVIATIONS

AH	AUTHENTICATION HEADER
CA	CERTIFICATE AUTHORITY
CIDR	CLASSLESS INTER DOMAIN ROUTING
DB	DATABASE
DDOS	DISTRIBUTED DENIAL OF SERVICE
DHCP	DYNAMIC HOST CONFIGURATION PROTOCOL
DMZ	DELIMITARIZED ZONE
DNS	DOMAIN NAME SERVICE
DOS	DENIAL OF SERVICE
ESP	ENCAPSULATING SECURITY PAYLOAD
FTP	FILE TRANSFER PROTOCOL
HTTP	HYPERTEXT TRANSFER PROTOCOL
I T	INFORMATION TECHNOLOGY
ICMP	INTERNET CONTROL MESSAGE PROTOCOL
IDPS	INTRUSION DETENSION PREVENTION SYSTEM
IGMP	INTERNET GROUP MANAGEMENT PROTOCOL
IM	INSTANT MESSAGING
IMAP	INTERNET MESSAGE ACCESS PROTOCOL
IP	INTERNET PROTOCOL
IPS	INTRUSION PREVENTION SYSTEMS
LANS	LOCAL AREA NETWORKS
LDAP	LIGHTWEIGHT DIRECTORY ACCESS PROTOCOL
MAC	MEDIUM ACCESS CONTROL
NAC	NETWORK ACCESS CONTROL

NAP	NETWORK ACCESS PROTECTION
NAPT	NETWORK ADDRESS AND PORT TRANSLATION
NAT	NETWORK ADDRESS TRANSLATION
NHIA	NATIONAL HEALTH INSURANCE AUTHORITY
NTP	NETWORK TIME PROTOCOL
QoS	QUALITY OF SERVICE
OPNET	OPTIMUM NETWORK PERFORMANCE
OSI	OPEN SYSTEM INTERCONNECT
PAT	PORT ADDRESS TRANSLATION
PIX	PRIVATE INTERNET EXCHANGE
PMTU	PATH MAXIMUM TRANSMISSION UNIT
POP	POST OFFICE PROTOCOL
RADIUS	REMOTE AUTHENTICATION DIAL IN USER SERVICE
RFC	REQUEST FOR COMMENT
SMTP	SIMPLE MAIL TRANSFER PROTOCOL
SQL	STRUCTURED QUERY LANGUAGE
SSL	SECURED SOCKET LAYER
TCP	TRANSMISSION CONTROL PROTOCOL
TLS	TRANSPORT LAYER SECURITY
UDP	USER DATAGRAM PROTOCOL
US	UNITED STATE
VOIP	VOICE OVER IP
VPN	VIRTUAL PRIVATE NETWORK
WAN	WIDE AREA NETWORK
XML	EXTENSIBLE MARKUP LANGUAGE

CHAPTER 1

1.1 Introduction

Internet connectivity is growing with most enterprises shitting to the use of web based services for services provision (Hunt R et al., 2003). As enterprises take on the Internet as a new business tool whether to sell, to collaborate or to communicate – web applications have become the new weakest link in the organization's security strategy. Technological innovations are fundamentally changing the way people live, work, play, share information and communicate with each other (Shelth et al., 2011). This is seen to be sharpening their competitive edge as it gives them and their customers, rapid access to information. Firewalls provide a mechanism for protecting these enterprises from the less secure internet over which customers or collaborating partners transfer packets destined for the corporate network (Hamed H et al., 2006). Network Firewalls protect a trusted network from an untrusted network by filtering traffic according to a specified security policy. A firewall is often placed at the entrance of each private network in the Internet. The function of a firewall is to examine each packet that passes through the entrance and decide whether to accept the packet and allow it to proceed or to discard the packet. A firewall's basic task is to regulate some of the flow of traffic between computer networks of different trust levels. Typical examples are the Internet which is a zone with no trust and an internal network which is a zone of higher trust. A zone with an intermediate trust level, situated between the Internet and a trusted internal network, is often referred to as a "perimeter network" or Demilitarized zone (DMZ) (Shelth et al., 2009).

Serving as the first line of defense against malicious attacks and unauthorized traffic, firewalls are crucial elements in securing the private networks of most businesses, institutions, and home networks. A firewall is typically placed at the point of entry between a private network

and the outside internet such that all network traffic has to pass through it (Hwang et al., n.d) In a distributed system, messages are encapsulated into packets, which often pass through multiple access points in a network and firewalls are responsible for filtering, monitoring, and securing such packets (S.W Lodin and C.L.Shuba, 1998)

Firewalls are usually either appliance type devices or software systems that run over an underlying operating system. Whitman and Mattord (2005, p.241) list five major processing categories of firewalls as packet filtering firewalls, application gateways, circuit gateways, MAC layer firewalls and hybrids. Each different type typically operates at different layers in the Open System Interconnect (OSI) model. Firewalls can also offer additional services such as Network Address Translation (NAT), encryption functionality through a Virtual Private Network (VPN), Dynamic Host Configuration Protocol (DHCP) and application content filtering (NIST, 2002, p.4). Firewalls can be configured in a variety of network connection architectures. These include packet filtering servers (Whitman et.al. 2005, p256-260). Firewalls can be placed within the organizations intranet to separate LANs, or as a bastion host to the hostile internet. There are performance issues to consider, as well as operational and monitoring factors, fitness for purpose, operational costs, vulnerabilities, threats, business rules, and partnerships.

"Serious attention has to be given to rule relations and interactions in order to determine the proper rule ordering and guarantee correct security policy semantics" (Al-Shaer, Hamed , 2004, p.1). As the rule set increases, the addition of new rules or modification of existing rules must not conflict with the intent of policy. Anomalies may be introduced if the rule set is not optimized, leading to a less than effective firewall implementation, in terms of both performance and security. Firewalls have grown to take up more tasks than merely filtering traffic to managing bandwidth, routing control, packet forwarding (Shimonski, R.J et al., 2003).

1.2 Overview of Firewall Technologies

A firewall is a logical object (hardware and/or software) within a network infrastructure which prevents communications forbidden by the security policy of an organization from taking place, analogous to the function of firewalls in building construction. Often a firewall is also referred to as a packet filter. The basic task of a firewall is to control traffic between different zones of trust and/or administrative authorities. Typical zones of trust include the Internet (a zone with no trust) and an internal network (a zone with high trust). The ultimate goal is to provide controlled connectivity between zones of differing trust levels through the enforcement of a security policy and a connectivity model based on the least privilege principle (Niederberger et al., 2006). Firewalling is often combined with other technologies-most notably routing. For example, network address translation (NAT) is sometimes thought of as a firewall technology, but it is actually a routing technology. Many firewalls also include content filtering features to enforce organization policies not directly related to security. Some firewalls include intrusion prevention system (IPS) technologies, which can react to attacks that they detect to prevent damage to systems protected by the firewall (Hofman et al., 2009).

Firewalls are often placed at the perimeter of a network. Such a firewall can be said to have an external and internal interface, with the external interface being the one on the outside of the network. These two interfaces are sometimes referred to as unprotected and protected, respectively. A firewall's policies can work in both directions; for example, there might be a policy to prevent executable code from being sent from inside the perimeter to sites outside the perimeter.

A firewall's configuration contains a large set of access control rules, each specifying source addresses, destination addresses, source ports, destination ports, one or multiple protocol ids,

and an appropriate action. The action is typically “accept” or “deny.” Some firewalls can support other types of actions such as sending a log message, applying a proxy, and passing the matched packets into a VPN tunnel. For most firewalls, the rule set is order-sensitive. An incoming packet will be checked against the ordered list of rules. The rule that matches first decides how to process the packet. Due to the multidimensional nature of the rules (including source/destination addresses and ports), the performance of a firewall degrades as the number of rules increases. Commercially deployed firewalls often carry tens of thousands of rules, creating performance bottlenecks in the network (Shelth et al., 2009).

The firewall data flow model in figure 1.1 depicts the operations performed by firewall. When a packet is received by a firewall, it first undergoes link layer filtering. Then, it is checked against a dynamic rule set. The packet then undergoes packet legality checks, and IP and port filtering. Finally, network/port address translation is performed. Sophisticated firewalls also reassemble packets and perform application level analysis. After a routing decision is made on the packet, out-bound filtering may also be performed. Each of these operations is optional, and the order in which the packet traverses may also differ in different firewalls.

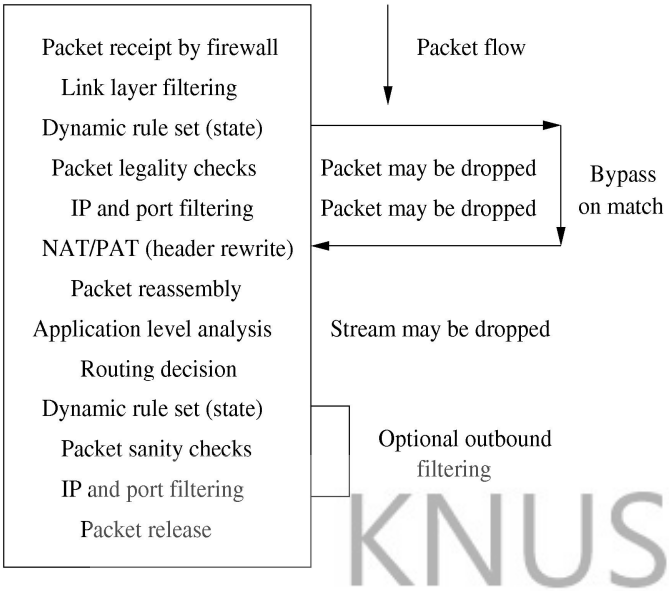

Fig.1.1: Firewall Operation and Data Flow

1.2.1 Packet filter

The most basic feature of a firewall is the packet filter. Older firewalls that were only packet filters were essentially routing devices that provided access control functionality for host addresses and communication sessions. These devices, also known as stateless inspection firewalls, do not keep track of the state of each flow of traffic that passes though the firewall; this means, for example, that they cannot associate multiple requests within a single session to each other. Packet filtering is at the core of most modern firewalls, but there are few firewalls sold today that only do stateless packet filtering. Unlike more advanced filters, packet filters are not concerned about the content of packets. Their access control functionality is governed by a set of directives referred to as a ruleset. Packet filtering capabilities are built into most operating systems and devices capable of routing; the most common example of a pure packet filtering device is a network router that employs access control lists.

In their most basic form, firewalls with packet filters operate at the network layer. This provides network access control based on several pieces of information contained in a packet, including:

i. The packet's source IP address - the address of the host from which the packet originated (such as 192.168.1.1)

ii. The packet's destination address - the address of the host the packet is trying to reach (e.g., 192.168.2.1)

iii. The network or transport protocol being used to communicate between source and destination hosts, such as TCP, UDP, or ICMP

iv. Possibly some characteristics of the transport layer communications sessions, such as session source and destination ports (e.g., TCP 80 for the destination port belonging to a web server, TCP 1320 for the source port belonging to a personal computer accessing the server)

v. The interface being traversed by the packet, and its direction (inbound or outbound).

Filtering inbound traffic is known as ingress filtering. Outgoing traffic can also be filtered, a process referred to as egress filtering. Here, organizations can implement restrictions on their internal traffic, such as blocking the use of external file transfer protocol (FTP) servers or preventing denial of service (DoS) attacks from being launched from within the organization against outside entities. Organizations should only permit outbound traffic that uses the source IP addresses in use by the organization - a process that helps block traffic with spoofed addresses from leaking onto other networks. Spoofed addresses can be caused by malicious events such as malware infections or compromised hosts being used to launch attacks, or by inadvertent misconfigurations.

1.2.2 Stateful Inspection

Stateful inspection improves on the functions of packet filters by tracking the state of connections and blocking packets that deviate from the expected state. This is accomplished by incorporating greater awareness of the transport layer. As with packet filtering, stateful inspection intercepts packets at the network layer and inspects them to see if they are permitted by an existing firewall rule, but unlike packet filtering, stateful inspection keeps track of each connection in a state table. While the details of state table entries vary by firewall product, they typically include source IP address, destination IP address, port numbers, and connection state information. Three major states exist for TCP traffic-connection establishment, usage, and termination (which refer to both an endpoint requesting that a connection be closed and a connection with a long period of inactivity.) Stateful inspection in a firewall examines certain values in the TCP headers to monitor the state of each connection. Each new packet is compared by the firewall to the firewall's state table to determine if the packet's state contradicts its expected state. For example, an attacker could generate a packet with a header indicating it is part of an established connection, in hopes it will pass through a firewall. If the firewall uses stateful inspection, it will first verify that the packet is part of an established connection listed in the state table.

In the simplest case, a firewall will allow through any packet that seems to be part of an open connection (or even a connection that is not yet fully established). However, many firewalls are more cognizant of the state machines for protocols such as TCP and UDP, and they will block packets that do not adhere strictly to the appropriate state machine. For example, it is common for firewalls to check attributes such as TCP sequence numbers and reject packets that are out of sequence. When a firewall provides NAT services, it often includes NAT information in its state table.

Table 1.1 provides an example of a state table. If a device on the internal network (shown here as 192.168.1.100) attempts to connect to a device outside the firewall (192.0.2.71), the connection attempt is first checked to see if it is permitted by the firewall ruleset. If it is permitted, an entry is added to the state table that indicates a new session is being initiated, as shown in the first entry under "Connection State" in Table 1.1. If 192.0.2.71 and 192.168.1.100 complete the three-way TCP handshake, the connection state will change to "established" and all subsequent traffic matching the entry will be allowed to pass through the firewall.

Table 1.1: State Table

Source Address	Source Port	Destination Address	Destination Port	Connection State
192.168.1.100	1030	192.0.2.71	80	Initiated
192.168.1.102	1031	10.12.18.74	80	Established
192.168.1.101	1033	10.66.32.122	25	Established
192.168.1.106	1035	10.231.32.12	79	Established

Because some protocols, most notably UDP, are connectionless and do not have a formal process for initializing, establishing, and terminating a connection, their state cannot be established at the transport layer as it is for TCP. For these protocols, most firewalls with stateful inspection are only able to track the source and destination IP addresses and ports. UDP packets must still match an entry in the state table based on source and destination IP address and port information to be permitted to pass - a DNS response from an external source would be permitted to pass only if the firewall had previously seen a corresponding DNS query from an internal source. Since the firewall is unable to determine when a session has ended, the entry is removed from the state table after a preconfigured timeout value is reached. Application-level firewalls that are able to recognize DNS over UDP will terminate a session after a DNS response is received, and may act similarly with the Network Time Protocol (NTP).

1.2.3 Application Firewall

A newer trend in stateful inspection is the addition of a stateful protocol analysis capability, referred to by some vendors as deep packet inspection. Stateful protocol analysis improves upon standard stateful inspection by adding basic intrusion detection technology - an inspection engine that analyzes protocols at the application layer to compare vendor-developed profiles of benign protocol activity against observed events to identify deviations. This allows a firewall to allow or deny access based on how an application is running over the network. For instance, an application firewall can determine if an email message contains a type of attachment that the organization does not permit (such as an executable file), or if instant messaging (IM) is being used over port 80 (typically used for HTTP). Another feature is that it can block connections over which specific actions are being performed (e.g., users could be prevented from using the FTP "put" command, which allows users to write files to the FTP server). This feature can also be used to allow or deny web pages that contain particular types of active content, such as Java or ActiveX, or that have SSL certificates signed by a particular certificate authority (CA), such as a compromised or revoked CA. Application firewalls can enable the identification of unexpected sequences of commands, such as issuing the same command repeatedly or issuing a command that was not preceded by another command on which it is dependent. These suspicious commands often originate from buffer overflow attacks, DoS attacks, malware, and other forms of attack carried out within application protocols such as HTTP. Another common feature is input validation for individual commands, such as minimum and maximum lengths for arguments. For example, a username argument with a length of 1000 characters is suspicious - even more so if it contains binary data. Application firewalls are available for many common protocols including HTTP, database (such as SQL), email (SMTP, Post Office Protocol [POP], and

Internet Message Access Protocol [IMAP]), voice over IP (VoIP), and Extensible Markup Language (XML).

Firewalls with both stateful inspection and stateful protocol analysis capabilities are not full-fledged intrusion detection and prevention systems (IDPS), which usually offer much more extensive attack detection and prevention capabilities. For example, IDPSs also use signature-based and/or anomaly-based analysis to detect additional problems within network traffic

1.2.4 Application - Proxy Gateways

An application - proxy gateway is a feature of advanced firewalls that combines lower-layer access control with upper-layer functionality. These firewalls contain a proxy agent that acts as an intermediary between two hosts that wish to communicate with each other, and never allows a direct connection between them. Each successful connection attempt actually results in the creation of two separate connections - one between the client and the proxy server, and another between the proxy server and the true destination. The proxy is meant to be transparent to the two hosts - from their perspectives there is a direct connection. Because external hosts only communicate with the proxy agent, internal IP addresses are not visible to the outside world. The proxy agent interfaces directly with the firewall ruleset to determine whether a given instance of network traffic should be allowed to transit the firewall.

Like application firewalls, the proxy gateway operates at the application layer and can inspect the actual content of the traffic. These gateways also perform the TCP handshake with the source system and are able to protect against exploitations at each step of a communication. In addition, gateways can make decisions to permit or deny traffic based on information in the application protocol headers or payloads. Once the gateway determines that data should be permitted, it is forwarded to the destination host.

Application-proxy gateways are quite different than application firewalls. First, an application-proxy gateway can offer a higher level of security for some applications because it prevents direct connections between two hosts and it inspects traffic content to identify policy violations. Another potential advantage is that some application-proxy gateways have the ability to decrypt packets (e.g., SSL-protected payloads), examine them, and re-encrypt them before sending them on to the destination host. Data that the gateway cannot decrypt is passed directly through to the application. When choosing the type of firewall to deploy, it is important to decide whether the firewall actually needs to act as an application proxy so that it can match the specific policies needed by the organization.

Firewalls with application-proxy gateways can also have several disadvantages when compared to packet filtering and stateful inspection. First, because of the "full packet awareness" of application-proxy gateways, the firewall spends much more time reading and interpreting each packet. Because of this, some of these gateways are poorly suited to high-bandwidth or real-time applications - but application proxy gateways rated for high bandwidth are available. To reduce the load on the firewall, a dedicated proxy server can be used to secure less time-sensitive services such as email and most web traffic. Another disadvantage is that application-proxy gateways tend to be limited in terms of support for new network applications and protocols - an individual, application-specific proxy agent is required for each type of network traffic that needs to transit a firewall. Many application-proxy gateway firewall vendors provide generic proxy agents to support undefined network protocols or applications. Those generic agents tend to negate many of the strengths of the application-proxy gateway architecture because they simply allow traffic to "tunnel" through the firewall.

1.2.5 Dedicated Proxy Servers

Dedicated proxy servers differ from application-proxy gateways in that while dedicated proxy servers retain proxy control of traffic, they usually have much more limited firewalling capabilities. Many dedicated proxy servers are application-specific, and some actually perform analysis and validation of common application protocols such as HTTP. Because these servers have limited firewalling capabilities, such as simply blocking traffic based on its source or destination, they are typically deployed behind traditional firewall platforms. Typically, a main firewall could accept inbound traffic, determine which application is being targeted, and hand off traffic to the appropriate proxy server (e.g., email proxy). This server would perform filtering or logging operations on the traffic, and then forward it to internal systems. A proxy server could also accept outbound traffic directly from internal systems, filter or log the traffic, and pass it to the firewall for outbound delivery. An example of this is an HTTP proxy deployed behind the firewall - users would need to connect to this proxy en route to connecting to external web servers. Dedicated proxy servers are generally used to decrease firewall workload and conduct specialized filtering and logging that might be difficult to perform on the firewall itself.

1.2.6 Virtual Private Network

Firewall devices at the edge of a network are sometimes required to do more than block unwanted traffic. A common requirement for these firewalls is to encrypt and decrypt specific network traffic flows between the protected network and external networks. This nearly always involves virtual private networks (VPN), which use additional protocols to encrypt traffic and provide user authentication and integrity checking. VPNs are most often used to provide secure network communications across untrusted networks. For example, VPN technology is widely used to extend the protected network of a multi-site organization across the Internet, and sometimes to provide secure remote user access to internal organizational

networks via the Internet. Two common choices for secure VPNs are IPsec6 and Secure Sockets Layer (SSL)/Transport Layer Security (TLS).

The two most common VPN architectures are gateway-to-gateway and host-to gateway. Gateway-to-gateway architectures connect multiple fixed sites over public lines through the use of VPN gateways - for example, to connect branch offices to an organization's headquarters. A VPN gateway is usually part of another network device such as a firewall or router. When a VPN connection is established between the two gateways, users at branch locations are unaware of the connection and do not require any special settings on their computers. The second type of architecture, host-to-gateway, provides a secure connection to the network for individual users, usually called remote users, who are located outside of the organization (at home, in a hotel, etc.) Here, a client on the user machine negotiates the secure connection with the organization's VPN gateway. For gateway-to-gateway and host-to-gateway VPNs, the VPN functionality is often part of the firewall itself. Placing it behind the firewall would require VPN traffic to be passed through the firewall while encrypted, preventing the firewall from inspecting the traffic.

All remote access (host-to-gateway) VPNs allow the firewall administrator to decide which users have access to which network resources. This access control is normally available on a per-user and per-group basis; that is, the VPN policy can specify which users and groups are authorized to access which resources, should an organization need that level of granularity. VPNs generally rely on authentication protocols such as Remote Authentication Dial in User Service (RADIUS). RADIUS uses several different types of authentication credentials, with the most common examples being username and password, digital signatures, and hardware tokens. Another authentication protocol often used by VPNs is the Lightweight Directory Access Protocol (LDAP); it is particularly useful for making access decisions for individual users and groups.

To run VPN functionality on a firewall requires additional resources that depend on the amount of traffic flowing across the VPN and the type of encryption being used. For some environments, the added traffic associated with VPNs might require additional capacity planning and resources. Planning is also needed to determine the type of VPN (gateway-to-gateway and/or host-to-gateway) that should be included in the firewall. Many firewalls include hardware acceleration for encryption to minimize the impact of VPN services.

1.2.7 Network Access Control

Another common requirement for firewalls at the edge of a network is to perform client checks for incoming connections from remote users and allow or disallow access based on those checks. This checking, commonly called network access control (NAC) or network access protection (NAP), allows access based on the user's credentials and the results of performing "health checks" on the user's computer. Health checks typically consist of verifying that one or more of the following comply with organizational policy:

i. Latest updates to antimalware and personal firewall software
ii. Configuration settings for antimalware and personal firewall software
iii. Elapsed time since the previous malware scan
iv. Patch level of the operating system and selected applications
v. Security configuration of the operating system and selected applications

These health checks require software on the user's system that is controlled by the firewall. If the user has acceptable credentials but the device does not pass the health check, the user and device may get only limited access to the internal network for remediation purposes.

1.2.8 Web Application Firewalls

The HTTP protocol used in web servers has been exploited by attackers in many ways, such as to place malicious software on the computer of someone browsing the web, or to fool a person into revealing private information that they might not have otherwise. Many of these exploits can be detected by specialized application firewalls called web application firewalls that reside in front of the web server. Web application firewalls are a relatively new technology, as compared to other firewall technologies, and the type of threats that they mitigate are still changing frequently. Because they are put in front of web servers to prevent attacks on the server, they are often considered to be very different than traditional firewalls.

1.2.9 Firewall for Virtual Infrastructures

Many virtualization solutions allow more than one operating system to run on a single computer simultaneously, each appearing as if it were a real computer. This has become popular recently because it allows organizations to make more efficient use of computer hardware. Most of these types of virtualization systems include virtualized networking, which allows the multiple operating systems to communicate as if they were on a standard Ethernet, even though there is no actual networking hardware.

Network activity that passes directly between virtualized operating systems within a host cannot be monitored by an external firewall. However, some virtualization systems offer built-in firewalls or allow third-party software firewalls to be added as plug-ins. using firewalls to monitor virtualized networking is a relatively new area of firewall technology, and it is likely to change significantly as virtualization usage continues to increase.

1.3 Firewall and Network Architecture

Firewalls are used to separate networks with differing security requirements, such as the Internet and an internal network that houses servers with sensitive data. Organizations use

firewalls whenever their internal networks and systems interface with external networks and systems, and where security requirements vary among their internal networks. This sections explains where firewall within organization should be placed and where other networks and systems should be located in relation to the firewalls. Since one of the primary functions of a firewall is to prevent unwanted traffic from entering a network (and, in some cases, from exiting it) firewalls should be placed at the edge of logical network boundaries. This normally means that firewalls are positioned either as a node where the network splits into multiple paths, or inline along a single path. In routed networks, the firewall usually resides just on the network at the location immediately before traffic enters the router (the ingress point), and is sometimes co-resident with the router. It is rare to place the firewall for a multi-path node after the router because the firewall device would need to watch each of the multiple exit paths that typically exist in such situations. The vast majority of hardware firewall devices contain router capabilities, and in switched networks, a firewall is often part of the switch itself to enable it to protect as many of the switched segments as possible.

A firewall takes traffic that has not been checked, checks it against the firewall's policy, and then acts accordingly (e.g., passes the traffic, blocks it, and passes it with some modification). Because all traffic on a network has a direction, policies are based on the direction that the traffic is moving. Traffic that has not yet been checked is coming from the “unprotected side” of the firewall and is moving towards the “protected side.” Some firewalls check traffic in both directions - for example, if they are set up to prevent specific traffic from an organization's local area network (LAN) from escaping to the Internet. In these cases, the protected side of the firewall is the one facing the outside network.

1.3.1 Network Layouts with Firewall

Figure 1.2 shows a typical network layout with a hardware firewall device acting as a router. The unprotected side of the firewall connects to the single path labeled “WAN,” and the protected side connects to three paths labeled “LAN1,” “LAN2,” and “LAN3.” The firewall acts as a router for traffic between the wide area network (WAN) path and the LAN paths. In the figure, one of the LAN paths also has a router; some organizations prefer to use multiple layers of routers due to legacy routing policies within the network.

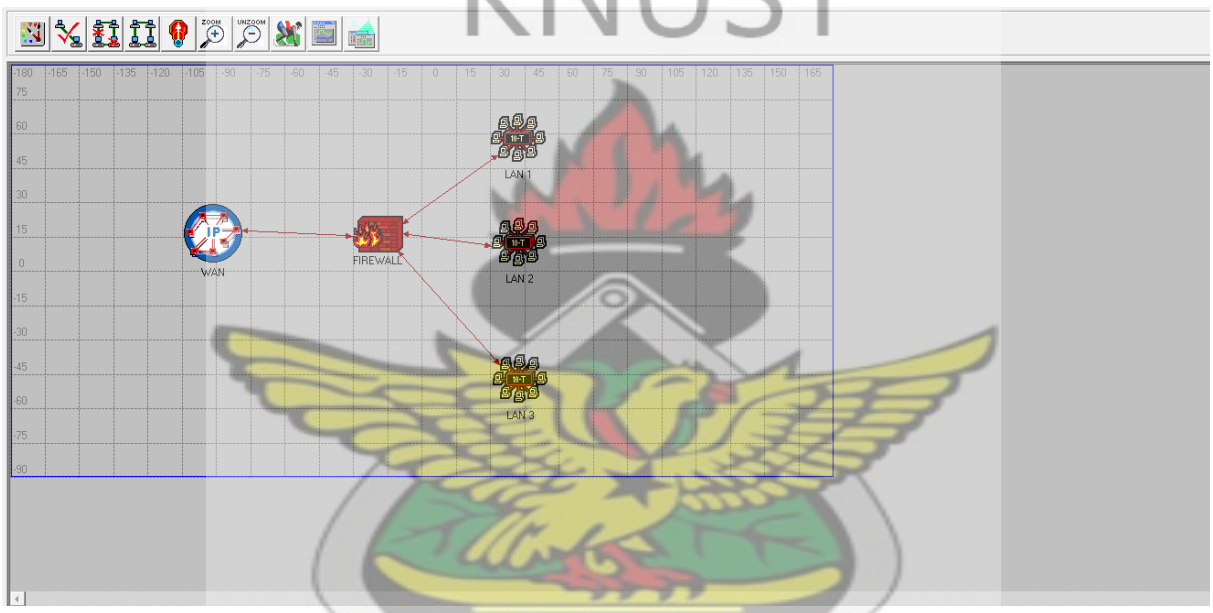

Fig.1.2: Simple Routed Network with Firewall Device

Many hardware firewall devices have a feature called DMZ (demilitarized zones); they are usually interfaces on a routing firewall that are similar to the interfaces found on the firewall’s protected side. The major difference is that traffic moving between the DMZ and other interfaces on the protected side of the firewall still goes through the firewall and can have firewall protection policies applied. DMZs are sometimes useful for organizations that have hosts that need to have all traffic destined for the host bypass some of the firewall’s policies (for example, because the DMZ hosts are sufficiently hardened), but traffic coming from the hosts to other systems on the organization’s network need to go through the firewall.

It is common to put public-facing servers, such as web and email servers, on the DMZ. Traffic from the Internet goes into the firewall and is routed to systems on the firewall's protected side or to systems on the DMZ. Traffic between systems on the DMZ and systems on the protected network goes through the firewall, and can have firewall policies applied.

Most network architectures are hierarchical, meaning that a single path from an outside network splits into multiple paths on the inside network - and it is generally most efficient to place a firewall at the node where the paths split. This has the advantage of positioning the firewall where there is no question as to what is "outside" and what is "inside." However, there may be reasons to have additional firewalls on the inside of the network, such as to protect one set of computers from another. If a network's architecture is not hierarchical, the same firewall policies should be used on all ingresses to the network. In many organizations, there is supposed to be one-ingress to the network, but other ingresses are set up on an ad-hoc basis, often in ways that are not allowed by overall policy. In these situations, if a properly configured firewall is not placed at each entry point, malicious traffic that would normally be blocked by the main ingress can enter the network by other means.

1.3.2 Firewall Acting as Network Address Translator

Most firewalls can perform NAT, which is sometimes called port address translation (PAT) or network address and port translation (NAPT). Despite the popular misconception, NAT is not part of the security functionality of a firewall. The security benefit of NAT-preventing a host outside the firewall from initiating contact with a host behind NAT-can just as easily be achieved by a stateful firewall with less disruption to protocols that do not work as well behind NAT. However, turning on a firewall's NAT feature is usually easier than properly configuring the firewall policy to have the same protections, so many people think of NATs as primarily a security feature.

Typically, a NAT acts as a router that has a network with private addresses on the inside and a single public address on the outside. The way a NAT performs this many-to-one mapping varies between implementations, but almost always involves the following:

i. Hosts on the inside network initiating connections to the outside network causes the NAT to map the source port of the connection to a different source port that is controlled by the NAT. The NAT uses this source port number to map connections from the outside back to the host on the inside.

ii. Hosts on the outside of the network cannot initiate contact with hosts on the inside network. In some firewalls, the NAT can be configured to map a particular destination port on the NAT to a particular host on the inside of the NAT; for example, all HTTP requests that go to the NAT could be directed to a single host on the protected side of the firewall. This feature is sometimes called pinholing.

Although NATs are not in and of themselves security features of a firewall, they interact with the firewall's security policy. For example, any policy that requires that all HTTP servers accessible to the outside be on the DMZ must prevent the NAT from pinholing TCP port 80. Another example of where NATs interact with security policy is the ability to identify the source of traffic in a firewall's logs. If a NAT is used, it must report the private address in the logs instead of the translated public address; otherwise the logs will incorrectly identify many hosts by the single public address.

1.3.3 Architecture with Multiple Layers of Firewalls

There is no limitation on where a firewall can be placed in a network. While firewalls should be at the edge of a logical network boundary, creating an "inside" and "outside" on either side of the firewall, a network administrator may wish to have additional boundaries within

the network and deploy additional firewalls to establish such boundaries. The use of multiple layers of firewalls is quite common to provide defense-in-depth.

A typical situation that requires multiple layers of network firewalls is the presence of internal users with varying levels of trust. For example, an organization might want to protect its accounting databases from being accessed by users who are not part of the accounting department. This could be accomplished by placing one firewall at the edge of the network (to prevent general access to the network from the Internet) and another at the edge of the internal network that defines the boundary of the accounting department. The inner firewall would block access to the database server by anyone outside the accounting network while allowing limited access to other resources on the accounting network. Another typical use for firewalls inside a network with a firewall at its edge involves visitors who need access to the Internet. Many organizations deploy specific wireless access points within their networks for visitor use. A firewall between the access points and the rest of the internal network can prevent visitors from accessing the local network with the same privileges as an employee.

One common problem with using multiple layers of network firewalls is the increased difficulty it presents in tracing firewall problems. If one firewall stands between a user and a server, and the user cannot connect to the server, it is easy to check that firewall's logs to see if the connection is being permitted. But if multiple firewalls are involved, the problem becomes more difficult because an administrator must locate all firewalls in the chain and check their logs to find where the problem originates. The presence of multiple layers of application-proxy gateways is particularly daunting, because each gateway can change a message, which makes debugging even more difficult.

1.4 Firewall Policy

A firewall policy dictates how firewalls should handle network traffic for specific IP addresses and address ranges, protocols, applications, and content types (e.g., active content) based on the organization's information security policies. Before a firewall policy is created, some form of risk analysis should be performed to develop a list of the types of traffic needed by the organization and categorize how they must be secured-including which types of traffic can traverse a firewall under what circumstances. This risk analysis should be based on an evaluation of threats; vulnerabilities; countermeasures in place to mitigate vulnerabilities; and the impact if systems or data are compromised. Firewall policy should be documented in the system security plan and maintained and updated frequently as classes of new attacks or vulnerabilities arise, or as the organization's needs regarding network applications change. The policy should also include specific guidance on how to address changes to the ruleset.

Generally, firewalls should block all inbound and outbound traffic that has not been expressly permitted by the firewall policy-traffic that is not needed by the organization. This practice known as deny by default, decreases the risk of attack and can also reduce the volume of traffic carried on the organization's networks. Because of the dynamic nature of hosts, networks, protocols, and applications, deny by default is a more secure approach than permitting all traffic that is not explicitly forbidden. This section provides details on what types of traffic should be blocked.

1.4.1 Policies Based on IP Addresses and Protocols

Firewall policies should only allow necessary IP protocols through. Examples of commonly used IP protocols, with their IP protocol numbers, are ICMP (1), TCP (6), and UDP (17). Other IP protocols, such as IPsec components Encapsulating Security Payload (ESP) (50) and Authentication Header (AH) (51) and routing protocols may also need to pass through

firewalls. These necessary protocols should be restricted whenever possible to the specific hosts and networks within the organization with a need to use them. By permitting only necessary protocols, all unnecessary IP protocols are denied by default.

Some IP protocols are rarely passed between an outside network and an organization's LAN, and therefore can simply be blocked in both directions at the firewall. For example, IGMP is a protocol used to control multicast networks, but multicast is rarely used, and when it is, it is often not used across the Internet. Therefore, blocking all IGMP traffic in both directions is feasible if multicast is not used.

1.4.2 IP Addresses and other Characteristics

Firewall policies should only permit appropriate source and destination IP addresses to be used. Specific recommendations for IP addresses include:

i. Traffic with invalid source or destination addresses should always be blocked, regardless of the firewall location. Examples of relatively common invalid IPv4 addresses are 127.0.0.0 to 127.255.255.255 (also known as the local host addresses) and 0.0.0.0 (interpreted by some operating systems as a local host or a broadcast address). These have no legitimate use on a network. Also, traffic using link-local addresses (169.254.0.0 to 169.254.255.255) should be blocked.

ii. Traffic with an invalid source address for incoming traffic or destination address for outgoing traffic (an invalid "external" address) should be blocked at the network perimeter. This traffic is often caused by malware, spoofing, denial of service attacks, or misconfigured equipment. The most common type of invalid external addresses is an IPv4 address within the ranges in RFC 1918, Address Allocation for Private Internets, which are reserved for private networks. These ranges are 10.0.0.0 to 10.255.255.255 (10.0.0.0/8 in Classless Inter-Domain

Routing [CIDR] notation), 172.16.0.0 to 172.31.255.255 (172.16.0.0/12), and 192.168.0.0 to 192.168.255.255 (192.168.0.0/16).

iii. Traffic with a private destination address for incoming traffic or source address for outgoing traffic (an “internal” address) should be blocked at the network perimeter. Perimeter devices can perform address translation services to permit internal hosts with private addresses to communicate through the perimeter, but private addresses should not be passed through the network perimeter.

Firewalls at the network perimeter should block all incoming traffic to networks and hosts that should not be accessible from external networks. These firewalls should also block all outgoing traffic from the organization’s networks and hosts that should not be permitted to access external networks. Deciding which addresses should be blocked is often one of the most time-consuming aspects of developing firewall IP policies. It is also one of the most error-prone, because the IP address associated with an undesired entity often changes over time.

1.4.3 TCP AND UDP

Application protocols can use TCP, UDP, or both, depending on the design of the protocol. An application server typically listens on one or more fixed TCP or UDP ports. Some applications use a single port, but many applications use multiple ports. For example, although SMTP uses TCP port 25 for sending mail, it uses TCP port 587 for mail submission. Similarly, FTP uses at least two ports, one of which can be unpredictable, and while most web servers use only TCP port 80, it is common to have web sites that also use additional ports such as TCP port 8080. Some applications use both TCP and UDP; for example, DNS lookups can occur on UDP port 53 or TCP port 53. Application clients typically use any of a wide range of ports.

As with other aspects of firewall rulesets, deny by default policies should be used for incoming TCP and UDP traffic. Less stringent policies are generally used for outgoing TCP and UDP traffic because most organizations permit their users to access a wide range of external applications located on millions of external hosts.

In addition to allowing and blocking UDP and TCP traffic, many firewalls are also able to report or block malformed UDP and TCP traffic directed towards the firewall or to hosts protected by the firewall. This traffic is frequently used to scan for hosts, and may also be used in certain types of attacks. The firewall can help block such activity - or at least report when such activity is happening.

1.4.4 ICMP

Attackers can use various ICMP types and codes to perform reconnaissance or manipulate the flow of network traffic. However, ICMP is needed for many useful things, such as getting reasonable performance across the Internet. Some firewall policies block all ICMP traffic, but this often leads to problems with diagnostics and performance. Other common policies allow all outgoing ICMP traffic, but limit incoming ICMP to those types and codes needed for Path Maximum Transmission Unit (PMTU) discovery (ICMP code 3) and destination reachability.

To prevent malicious activity, firewalls at the network perimeter should deny all incoming and outgoing ICMP traffic except for those types and codes specifically permitted by the organization. For ICMP in IPv4, ICMP type 3 messages should not be filtered because they are used for important network diagnostics. The ping command (ICMP code 8) is an important network diagnostic, but incoming pings are often blocked by firewall policies to prevent attackers from learning more about the internal topology of the organization's network. For ICMP in IPv6, many types of messages must be allowed in specific circumstances to enable various IPv6 features.

ICMP is often used by low-level networking protocols to increase the speed and reliability of networking. Therefore, ICMP within an organization's network generally should not be blocked by firewalls that are not at the perimeter of the network, unless security needs outweigh network operational needs. Similarly, if an organization has more than one network, ICMP that comes from or goes to other networks within the organization should not be blocked.

1.4.5 Policy Based on Network Activity

Many firewalls allow the administrator to block established connections after a certain period of inactivity. For example, if a user on the outside of a firewall has logged into a file server but has not made any requests during the past 15 minutes, the policy might be to block any further traffic on that connection. Time-based policies are useful in thwarting attacks caused by a logged-in user walking away from a computer and someone else sitting down and using the established connections (and therefore the logged-in user's credentials). However, these policies can also be bothersome for users who make connections but do not use them frequently. For instance, a user might connect to a file server to read a file and then spend a long time editing the file. If the user does not save the file back to the file server before the firewall-mandated timeout, the timeout could cause the changes to the file to be lost.

A different type of firewall policy based on network activity is one that throttles or redirects traffic if the rate of traffic matching the policy rule is too high. For example, a firewall might redirect the connections made to a particular inside address to a slower route if the rate of connections is above a certain threshold. Another policy might be to drop incoming ICMP packets if the rate is too high. Crafting such policies is quite difficult because throttling and redirecting can cause desired traffic to be lost or have difficult-to diagnose transient failures.

1.5 Problem Statement

Internet connection sharpens the competitive edge of most business to day since it gives them and their customer's timely access to information. In the cyber age, threats to computer networks are beyond any dispute. Any corporation would prioritize safety of their networked resources, availability of IT infrastructure, confidentiality, integrity and availability of information they store and or transmit (H. Garantla et al., 2002). Threats come in various forms; malicious attacks, viruses, Trojan horses, spam, malware, masquerading, eavesdropping, theft, deletion, corruption, etc. Researchers and developers are working round the clock to combat security risks. Firewalls are essential components in improving network security. Anti-virus developers always recommend the use of a separate firewall. Setting up a firewall for private network sites in organizations and at home is no longer a too fancy thing. On the other aspect, performance impact may cause major concerns. Commercially deployed firewalls often carry tens of thousands of rules, creating performance bottlenecks in the network. Is there a significant performance loss while incorporating a secure environment using a firewall for the Internet connection? To what level of security should we expect without sacrificing the network performance? These are the mind boggling questions that the research work intends to find an answer to.

1.6 Objectives

This research work intends to emphasize on the impact of different security controls on network performance. The goal of the research is to evaluate the relationship between network security and performance; the effects of firewalls on network performance.

The specific objectives are

i. Examine whether there is a significant performance loss while incorporating a secure environment using a firewall for the Internet connection?

ii. To determine what level of security should we expect without sacrificing the network performance

iii. To examine the effects of firewall in the link utilization

1.7 Research Questions

Based on the statement of problem, the objectives of the study, the researcher pose the following questions:

i. Is there a significant performance loss while incorporating a secure environment using a firewall for the Internet connection?

ii. To what level of security should we expect without sacrificing the network performance?

iii. What will be the effect of firewall in network link utilization?

1.8 Research Hypothesis

On the basis of the statement of problem, objectives of the study and research question, the following hypothesis have been formulated.

i. H1: Is there a significant performance loss while incorporating a secure environment using a firewall for the Internet connection

ii. H0: There is no significant performance loss while incorporating a secure environment using a firewall for the Internet connection

iii. H1: There is a level of security that affect a network performance

iv. H0: There is no level of security that affect the performance of a network

v. H0: Firewall has an effect on a network link utilization

vi. H1: Firewall has no effect on network link utilization.

1.9 Significance of the study

The growing demand for using firewall by the internet users and companies to provide more protection for them and the influence of apply firewall policy in the network performance gave the impetus for this study. The study is to help non-technical people understand why security is necessary and also to realize the impact of firewall security on network performance.

1.10 Scope of the study

The study will be limited to computer simulations. Because of the security sensitivity of the research it will not be demonstrated on a production environment.

1.11 Research Methodology

The main aim of this research is to evaluate the performance of a distributed system against firewall security policy. We shall evaluate the relationship between performance and security under three (3) different scenarios: Since the interest here is the performance of a network incorporating firewalls, we modeled networks with and without firewalls and different firewall functionality and simulated such networks with an eye on their performances.

The simulation will be done for 300 work stations and will be simulated in a way that all the 300 work stations access a database, ftp, email and web application under three different scenarios. The following performance metrics will be used for performance evaluation under the three scenarios.

i. HTTP page response time is estimated for the web application

ii. Data Base (DB) query time and response time for the database application

iii. Email download response and upload response time

iv. Ftp download response and upload response time

v. Node level statistics like server DB query response time and load are also estimated for the database application

vi. Link level and utilization statistics are also estimated across the simulation process

OPNET IT guru will be used as the simulation tool. The three scenarios are created using this simulation tool and simulation is run for two hours and the results evaluated.

1.12 Thesis Organization

Overview of firewall technologies, firewall policy, firewall and network architecture, problem statement, objectives, scope and limitation of the study were presented in chapter 1. In chapter 2, we provide a literature review in the areas of network security, security policies, policy enforcement, and firewall policy management systems. Chapter 3 provides an in-depth description of the research methodology. Chapter 4 gives the result of the simulation of the three scenarios of the methodology. Finding, conclusion and recommendations are presented in chapter 5

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Security is a major issue for organizations, both the legal issues associated with computer and network security as well as its implementation. Threats such as malware and DoS constantly test organizations security. Thus, it is necessary to take the concept of security as an ideal, rather than an implementation applied to various components of a system. It is necessary to take the entire system as an entity and the application of security to the entire network. This chapter explored literature in the areas of network security, security policies, policy enforcement, and firewall policy management systems.

2.2 Security Policy

According to (Aronson et al. 1997) a security policy is a formal statement of the rules by which people who are given access to an organization's technology and information assets must abide. According to the authors this is an ongoing process, with regular reviews and auditing of security policies and mechanisms, providing feedback to improve the security policy.

(Danchev, 2003) postulate that it is extremely important to involve the users in the implementation of a security policy. Understanding of security problems by users, and giving them clear and easy to follow rules, can be a key factor in the successful implementation of the policy. (Danchev, 2003) calls this the "Security Awareness program" and emphasizes that the latest technical security measures, such as firewalls and Intrusion Detection and Prevention System (IDPS), can be rendered useless by careless, or badly informed, end-users.

2.2.1 Enforcing Security Policies

Security policies protect the confidentiality, integrity, and availability of the assets of an organization. To enforce this, security services should be deployed, such as authentication, encryption, antivirus software and firewalls. To do this the security policy documents are used to create technical security procedures, and guidelines, which can then be implemented in the network (Macfarlane, 2009). Security Policies can be split into several components, which combine to achieve the security goals of the organization. These components are enforced by various security mechanisms or procedures. The authentication policy could be enforced using usernames and passwords, software tokens, and VPNs. The accountability policy may use IDPS and firewalls to enforce auditing and incident response capabilities (Aronson et al. 1997).

The access control part of the security policy deals with making sure that authorized individuals can perform the tasks they are authorized to and those others cannot. It is typically referred to as the 'access control policy' (Samarati et al. 2000). Access control makes sure that requests to access a specific resource are only granted if the request agrees with the security policy definition. In terms of networks, the most commonly used access control mechanisms are firewalls and filtering routers (Corbitt, 2002). Firewalls control access to resources by filtering network traffic, only allowing access that is specified by the security policy.

The network access control policies, defining which traffic can cross network boundaries, are implemented as policies on network devices which have access control functionality, such as traffic filtering capabilities. The system administrator is typically tasked with manually creating these low level policies, or configurations. In order to determine whether to grant an access request, access control mechanisms uses a number of criteria. The primary criterion

being the network address of the machine from which the traffic originates. Other criteria, which can decide whether access is granted or not, would include network service and destination of the traffic (Corbitt, 2002). The most common technique used to by firewalls to filter traffic is known as Packet Filtering

2.2.2 Policy Enforcement Problems

According to (Mayer et al. 2006) "the protection that these firewalls provide is only as good as the policy they are configured to implement". The policy should be clear, concise, and easy for the administrator to follow. If a policy is not well designed, then it will not be enforced properly and the security goals will not be met (Madigan et al. 2004). Conversely, polices are only as good as the configurations which enforce them (Schneider, 2004, Cheswick et al. 1994). The enforcement of policies is not always an easy task. Policy management can be difficult as policies grow and become increasingly complex (Blakley, 1996, Wood, 2004). According to (Blakley, 1996) "Policies do not scale well and their complexity quickly increases as systems grow and diverge, which makes them unmanageable". (Madigan et al. 2004) categorizes violations of security policies, and shows that violations from network issues were by far the largest type reported. The author also states that the network violations were among the most time consuming to correct.

The configuration of a firewall is probably the most important factor in terms of the security a firewall provides (Rubin et al. 1997), but are often configured incorrectly (Wood, 2004). Firewall policies are made up of rule sets, and these rule sets are ever expanding due to new rules continually being added and very few removed, so device access policies tend to be large and always increasing in size (Cardwell et al. 2003, Wood, 2004). It follows that the management of these policies at the network device level can be extremely complex, error-prone and expensive as the policies expand (Wong, 2008). The configurations are typically

hand crafted and bespoke for each individual system by network administrators, which can be error prone work (Cuppens et al 2004). This is a serious problem as errors in the firewall policies mean that the intended security policy will not be enforced.

Administrators are typically tasked with the creation of the low level device policies, which implement the security policy of the organization. In terms of firewalls, these are the firewall rule sets. The administrators will add, delete, and change the rules to match changes to the high level security requirements. For example, when new web servers are added to the organization's network, new rules would be added to the perimeter firewalls to allow appropriate access to them from outside and inside the organization's network. The complexity of the rule sets increase as they increase in size, but also the complexity can change depending on the rules used within them(Macfarlane, 2009).

(Wool, 2004) created a classification system for complexity of rule sets, based on the number of rules, the objects (traffic filtering parameters) and the interfaces rules could be applied to. According to the author, the following rule set complexity measure is defined:

> Rule Complexity = Number of Rules + Network Objects + Number of Interfaces (Number of Interfaces -1) /2

For most firewalls the ordering of the rules in a rule set are important, as in the common 'first match' filtering mechanism, the position of the rules in the rule set dictate if they are matched against traffic or not. The earlier in the rule set the higher the priority the rule has when matching against traffic (Wool, 2006). Thus filtering rule sets, that use first match semantics, are complicated to create and amend due to this on the rule ordering. As the size and complexity of the rule set increases it becomes more difficult for the administrator to predict the impact of a rule on the overall rule set. This makes rule sets extremely difficult to manage (Wool, 2004).

Effectiveness of security policies can be compromised due to poor policy management, especially when enforcing a security policy across a range of devices around a network (Ioannidis et al. 2000). According to the authors, security policies can be spread over a range of different security devices. Packets can take multiple paths through a network, with multiple filtering devices on each different path. An administrator needs to understand the interaction of combinations of these devices for each traffic path (Hamed et al. 2004). The low level configurations, which implement the security policy, can span heterogeneous networks, and may be spread over many network devices.

2.3 Network Security Threat

Network security consists of the provisions and policies adopted by a network administrator to prevent and monitor unauthorized access, misuse, modification, or denial of a computer network and network-accessible resources. Network security involves the authorization of access to data in a network, which is controlled by the network administrator. Users choose or are assigned an ID and password or other authenticating information that allows them access to information and programs within their authority (Simmonds et al. 2004). The NIST defined security threat as any circumstance or event with the potential to adversely impact organizational operations (including mission, functions, image, or reputation), organizational assets, or individuals through an information system via unauthorized access, destruction, disclosure, modification of information, and/or denial of service. In computer security a threat is a possible danger that might exploit a vulnerability to breach security and thus cause possible harm (Wikipedia). Since firewalls, which are supposed to filter and block attacks, some network threats will be looked at in this section. As there are so many kinds of threats, only the most common ones will be presented.

2.3.1 Worms, Viruses and Trojans

Viruses, worms and trojans are all considered malicious software or malicious code. The NIST (1995, p. 27) describes these three types of malicious software as the following:

i. A virus is a segment of code that get attached itself to an existing executable (an application) and is executed when the user runs the contaminated application. Usually, viruses carry out malicious actions such as overwriting or deleting data.

ii. A worm is a self-replicating program that does not need to be attached to a host program. No user intervention is required for the worm to run and replicate itself before it tries to propagate to other host systems throughout a network.

iii. A trojan horse is disguised malicious software that looks like a legitimate one. Once executed, it will carry out malicious actions such as opening a backdoor (opening a port) on the host allowing outside access to that machine.

2.3.2 Network Scanning

The purpose of network scanning is to gather information about a network (Wagh, 2009) such as network addresses and open network ports. This threat is usually the first step an intruder takes before actively attacking a network (Buchanan, 2011).

The most common scanning techniques are ping sweep, port scan and port sweep. Ping sweep refers to the scanning of a wide range of network addresses on a given subnet work and aims at knowing which machines are responsive, and thus running, on a network. Port scan technique refers to the scanning of TCP ports for a range of hosts on a given subnet work in order to find out what ports are open on the host machines. Port sweep is very similar to port

scan, but focuses instead on specific ports rather than scanning every one of them. One of the most famous programs that permits network scanning is Nmap (Lyon, 2011)

2.3.3 User Privilege Gain

This type of threat consists of gaining a foothold on a network. Usually it involves obtaining access to a low-level user account by guessing of login and passwords using techniques such as brute force dictionary attacks. Once on the system, an intruder will try to move up through the privilege hierarchy (Buchanan, 2011) using for example software flaws to exploit weaknesses, where in some cases a program would not properly identify the requester. The ultimate goal of a user privilege gain attack is to obtain the highest level of access such as an administrator or root account. If an intruder manages to gain access to such an account, he becomes in control of the entire system. One of the most famous dictionary attack program is Hydra (THC, 2011). It supports many different services such as SSH, FTP or HTTP. In this type of attack, the program attempts to gain access to a target using user name and password lists trying all possible combinations (Shirey, 2007).

2.3.4 Denial of Service Attack

A denial of service (DoS) attack consists of flooding a target with continual requests for services, which eventually reduces its performance (Buchanan, 2011). It consists of various types of carrying out DoS attack, but the three main ones that remains a serious threat nowadays are smurf attack, flood attack and SYN attack (Georgieva, 2009). With a smurf attack, also called ICMP flood, the attacker sends an ICMP echo request to a broadcast address with the source address of the echo request being the IP address of the victim. The attacker target a server within the victim's network that will broadcast the ICMP echo request throughout the entire network resulting in all the network's machines returning a response.

SYN attacks exploit a weakness in the TCP/IP handshaking mechanism, which is the state retention TCP performs for some time after receiving a SYN request on an open port. If an attacker floods the target with SYN messages, the SYN buffer will quickly get full not allowing new legitimate connections to be established (Eddy, 2007).

Flood attack is one of the first forms of DoS attack and its mechanism is simple, it consists of sending more traffic to a server than it can handle. This type of attack can only succeed if it uses Distributed Denial of Service (DDoS) attack. A DDoS attack uses multiple computers to send out requests to the target. The attack is generally controlled by a single computer (Master) which communicates with a network of infected machines which have a bot installed on them to initiate the attack. These networks of infected machines are known as botnets (Nazario, 2008).

2.4 Summary

In this chapter we reviewed literature on network security, security policies, policy enforcement, firewall policy management systems and network security threat. The methodology is presented in the next chapter.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter the three scenarios are modeled using the OPNET IT Guru Academic Edition 9.1 as a simulation tool. The National Health Insurance Authority's (NHIA) headquarters network connects to the Internet through a CISCO PIX Firewall. The Various regional and district mutual scheme users use various online applications including e-mail, web browsing, and membership card authorization. In addition, it is assumed that users are doing illegal file transfers for pirated music and videos. NHIA's most critical application is membership card authorization. It is required to have a response time of less than 3 seconds. First we'll evaluate the application performance with no firewall policies. Thus, no illicit traffic is blocked. A detail explanation of the three scenarios are discussed in the following sections

3.2 OPNET IT Guru as a Simulation Tool

OPNET IT Guru Academic Edition 9.1 provides a virtual environment for modeling, analyzing, and predicting the performance of IT infrastructures, including applications, servers, and networking technologies. This virtual approach saves the time and expense of building the real network in order to plan and test network changes and additions. It can also be used to diagnose problems, such as traffic growth and network failures.

The following elements are used to create the network models; Project, Scenario, Objects, Applications Definition Config Node, Profile Definition Config Node

3.2.1 Project

A project in IT Guru is a network simulation. This is where you specify the objects that will form the simulation, the applications that will run on the network, and which objects will run these applications

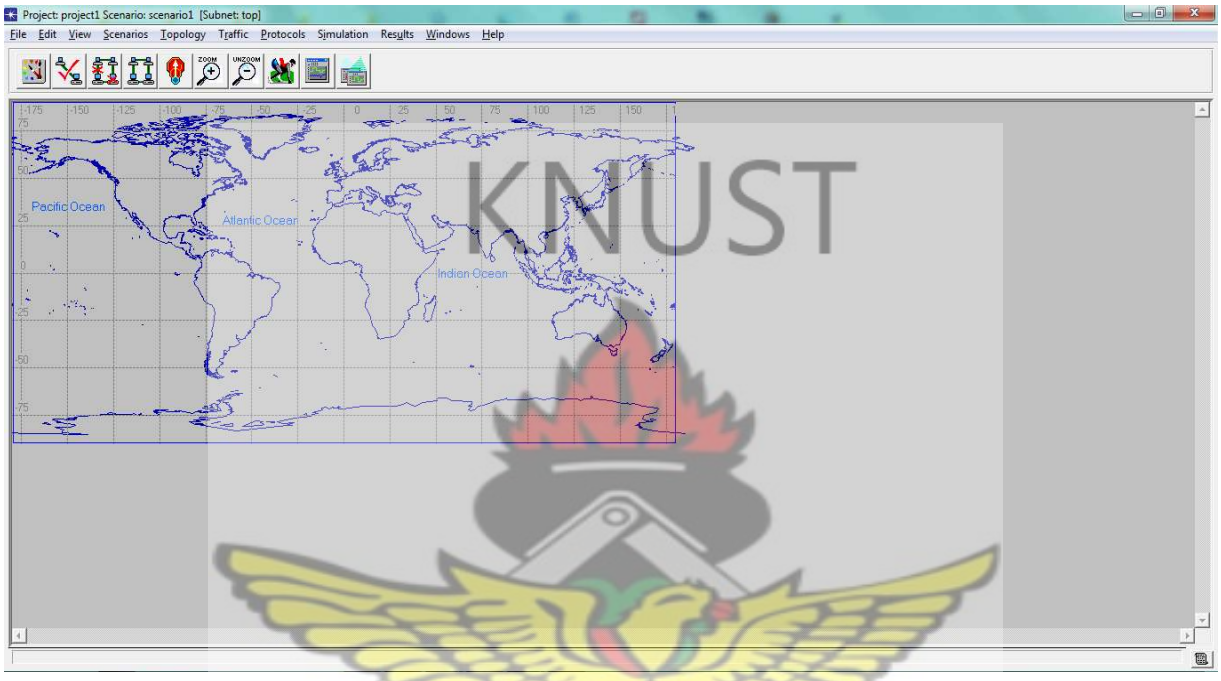

Fig.3.1: Project Editor Window

With the palette showing, the devices needed for the network of interest can be picked or drag and drop. These can be of two types; node and link. Nodes are devices that can send and receive information, such as switch, workstation, printer, and server. Links are a communication medium that connects nodes to one another. Links can represent electrical or fiber optic cables.

3.2.2 Scenario

A scenario is used to alter a project so that what if analysis can be done. For example, characteristics of objects or applications can be altered to see how this changes network

performance. This includes things such as changing equipment or altering loads on the network

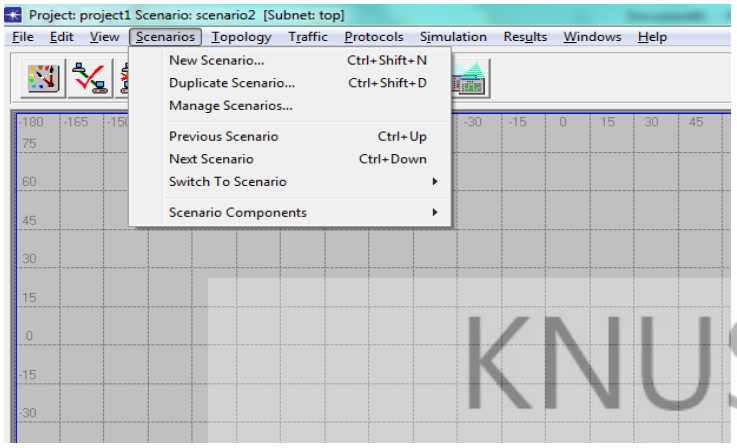

Fig. 3.2: Scenario

When creating a new network model, you must first create a new project and scenario. A project is a group of related scenarios that each explores a different aspect of the network. Projects can contain multiple scenarios.

3.2.3 Object

An object in IT Guru is anything that might appear in a real network. This is anything that you can drag and drop into a project. For example, workstations, servers, switches, routers, a T1 line are all objects. An object has attributes that define how it operates in the simulation.

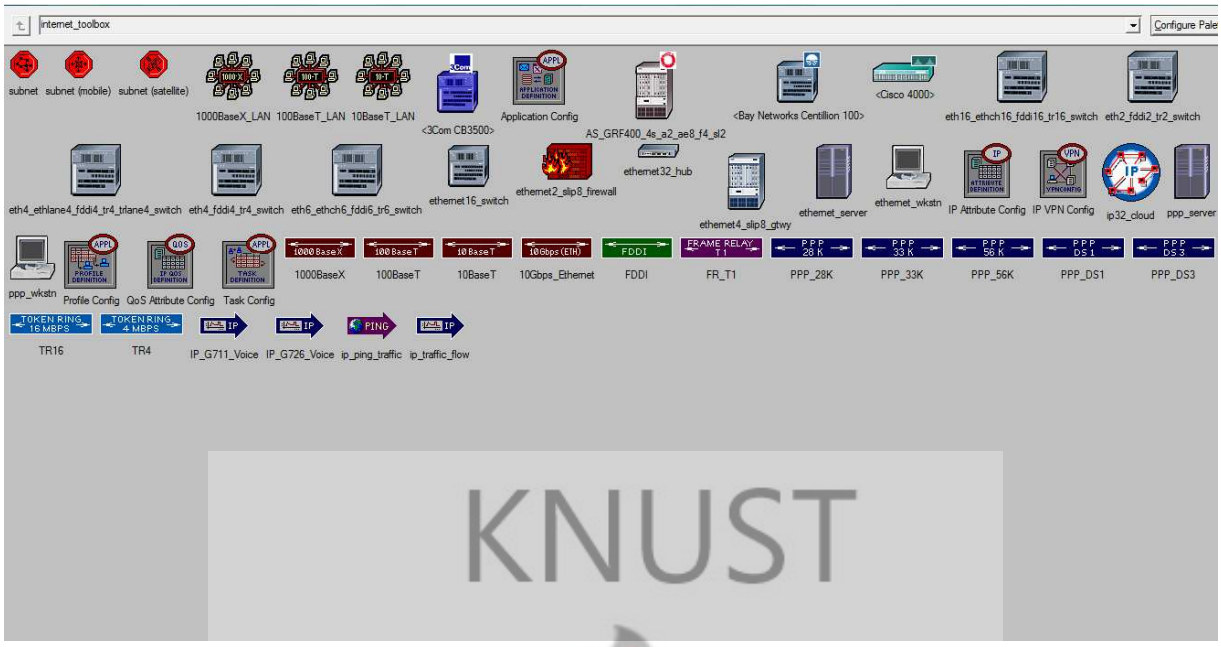

Fig.3.3: Object Palette

3.2.4 Application Definition Config

A special element needs to be dragged into any project to define what applications will generate traffic on the network. This special element is the Applications node. It contains the attributes for the applications used in the network, such as Web browsing, Emails, Ftp and database

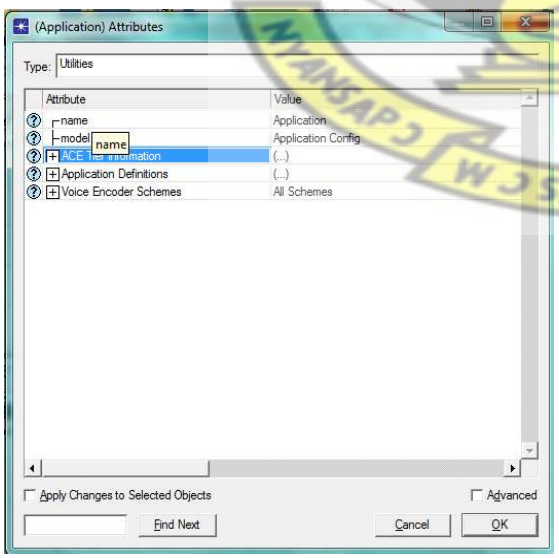

Fig.3.4: Application Attribute

3.2.5 Profile Definition Config

The last element commonly used is the Profiles node. This element is also dragged onto the project network. The Profiles node is used to associate the applications with the objects that will use them

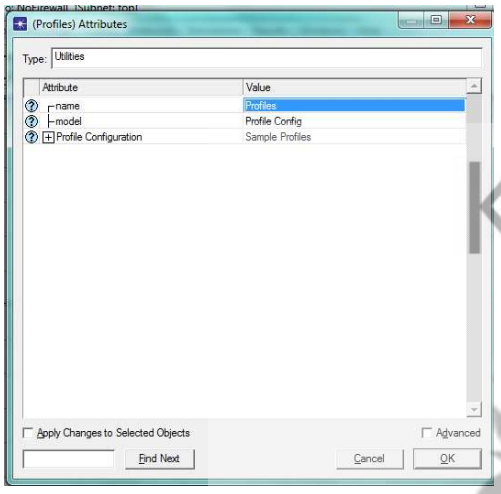

Fig.3.5: Profile Attribute

3.3 No Firewall Scenario

In this scenario no firewall is imposed on the entire network. An IP based cloud is used as the required cloud and this cloud acts as the internet cloud and connects two or more subnets, which represents the regional and district mutual schemes. Two routers are used across the simulation process among those where one of the router act as the firewall routers. Four different applications are created across this scenario; these are the database application, the web application, email and ftp applications. The required application traffic is created by configuring the applications at the application configuration and profile configuration level. A heavy database access application is used in this simulation such that imposes more database queries over the database server. Required configurations are done at the application and the profile config level and the performance of the cloud in terms of database

applications, web application, email and ftp applications are evaluated. Figure 3.6 shows the basic workspace of OPNET IT Guru Academic Edition 9.1

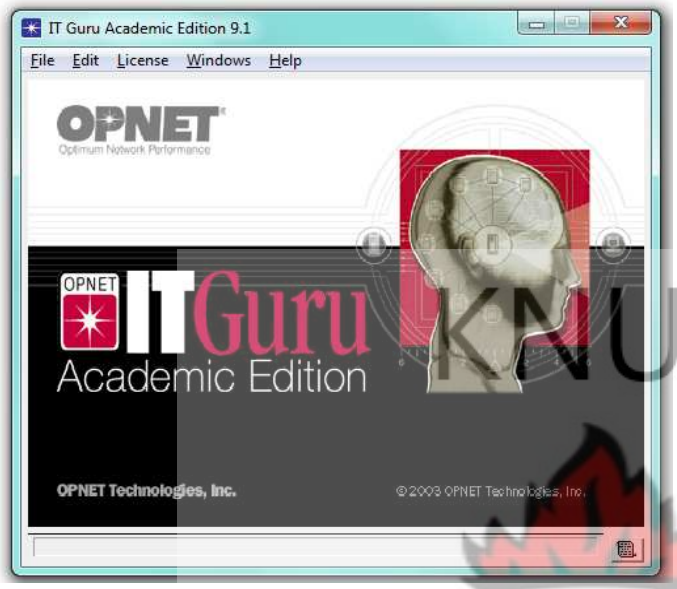

Fig. 3.6: OPNET Startup Screen

A new project is created by clicking on the file menu and the required scenarios are also created at this level as shown in Figure 3.7

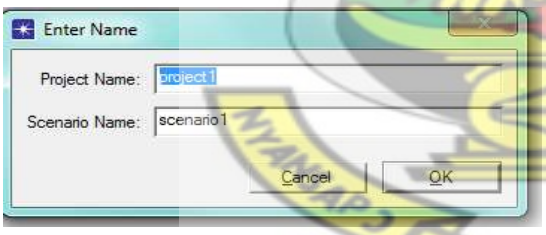

Fig. 3.7: New Project

The internet used across this simulation is done for 300 workstations and it is simulated in a way that all the 150 workstations access the database application and 200 workstations access the web application and 100 workstations use emails and ftp to download and upload file onto the file server. Following are the performance metrics used for the performance evaluation of internet when there is no security across the internet.

i. HTTP page response time is estimated for the web application

ii. DB query time and response time for the database application are estimated

iii. Email download response time and upload response time

iv. Ftp download response time and upload response time

v. Node level statistics like server DB query response time and load are also estimated for the database application

vi. Link level and utilization statistics are also estimated across the simulation process

vii. Data throughput which is the amount of data transferred in the network per time unit is evaluated through statistics like Traffic Received and Traffic Sent (bits/sec) which indicates the value of throughput. A more efficient network should allow more traffic to pass that leads to larger throughput.

viii. Packet Delay

ix. Traffic drop

x. Task processing time of the server is also evaluated

xi. Jitter: Packets arrive at destination with variable delay. Jitter depends on the congestion of the network. In computer networks, the term jitter means variations in delay of packets received. Jitter is an essential quality of service (QoS) factor in evaluation of network performance

The same performance metrics is used for the two scenarios. A packet size of 32MB (low), 100MB(medium) and 200MB(high) are imposed across the network and a link speed of 10Mbps, 1Gbps and 10Gbps are set between the router and the cloud and the above performance metrics is evaluated in each packet sizes and data rate to investigate applications performance.

3.4 Firewall scenarios

The first scenario is duplicated and the required firewall scenario is created. In this particular scenario a firewall router is created and a constant packet latency of 0.05 seconds is imposed for packet filtering. Similar performance metrics are used as in the first scenario

3.5 Firewall: With Packet Filtering Capabilities scenarios

This scenario is created by duplicating the second scenario and the main aim of this scenario is to block the unauthorized web, e-mail and ftp access. After the three scenarios are created the simulation is run for two hours and the corresponding performance of the network is evaluated.

3.6 Simulation Procedure

Since the goal of this research is to find the impact of different security controls on network performance and also to evaluate the relationship between network security and performance and the effect of firewall for three different scenarios like no firewall, firewall and firewall with packet filtering, OPNET IT Guru Academic Edition 9.1 is used as the simulation tool. The various scenarios are detailed in the following sections.

3.6.1 Simulation of No Firewalls Scenario

In this section, the procedure to simulate the no firewall case is presented. Firewall is a router that can impose some security policies over the network. Firewalls can monitor and regulate the traffic that passes across the network and internet and such a firewall is used in this application. In this simulation a home office LAN network is used as the destination and all the communication is done through the cloud and few routers. The following steps are used to create the simulation

A new project is created and project name and scenario are given as thesis and No_Firewall as shown in the figure 3.8

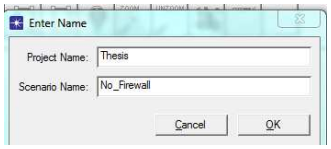

Fig .3.8: No Firewall Scenario

Once the required project name and scenario name are set, the following steps are used to create the basic network

i. Create Empty scenario is chosen as the option for Initial topology option and click on next
ii. The world is chosen as the required network scale and next
iii. In the maps US is selected and click next. We are constrained by the software so we picked US to define the geographical size of the network
iv. Click next for two times and the basic workspace with the required object palette are displayed as shown below

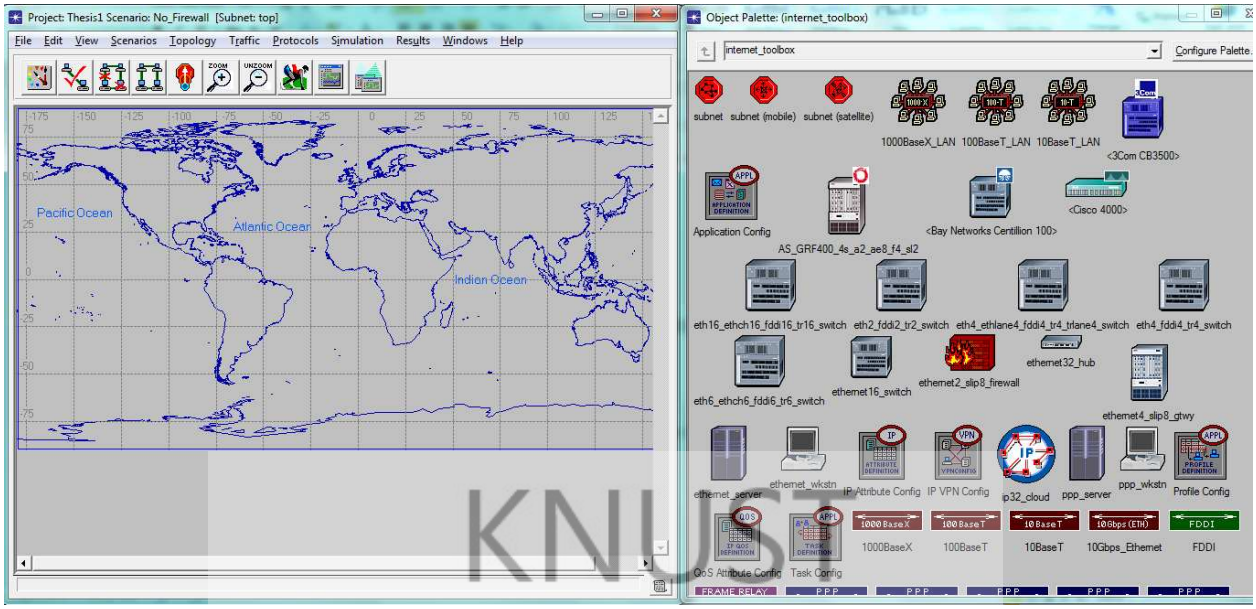

Fig.3.9: Workspace for Network Definition

The following objects from the object palette are drag onto the work space

i. The application configuration object is used to define the applications. In this simulation database, emails, ftp and Http applications are used

ii. The profile configuration object is used to define the application profiles

iii. Ip32_cloud object is used to act as the internet cloud

iv. Two ethernet4_slip8_gtwy's are used to act as router A and router B

v. 10BaseT_LAN object is used to act as the home office which supports 300 workstations

vi. Three ppp_server objects are used to act as the file server, database server and web server

The screenshot below shows the network layout after the objects are placed on the workspace.

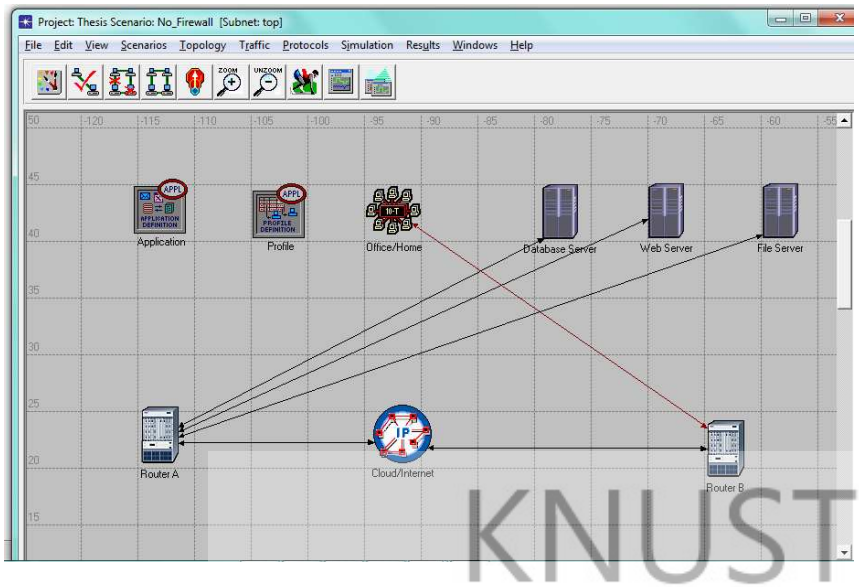

Fig.3.10: Network Layout

3.6.2 Application Configuration

Four applications are created in this scenario to generate the required traffic over the cloud. OPNET IT guru provides a separate object known as application config and the required applications can be created at this level. The following procedure illustrate how to configure the application

i. Right click on the Application config object and choose the edit attributes
ii. Add four rows to the applications definitions tab, such that four applications are created
iii. Rename a row as Database and choose the heavy load database against the Database application
iv. Rename another row as web and choose heavy browsing against HTTP application and the corresponding screenshot is as shown below
v. The rest of the rows are rename as email and ftp and choose high load against both FTP and Email application
vi. Click on ok and save the project

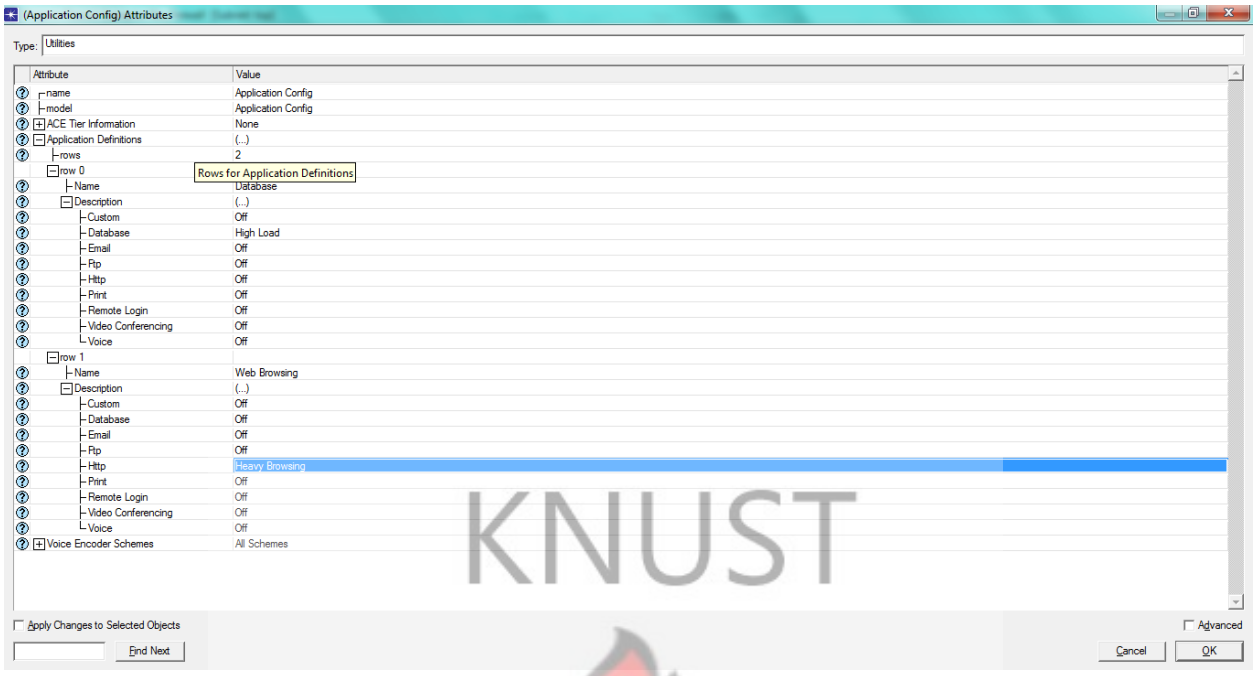

Fig.3.11: Application Configuration

3.6.3 Profile Configuration

An application needs to generate traffic over the internet. OPNET IT Guru provides a profile configuration object which can be used to generate the necessary traffic. The steps below details how to set the profile definition

i. Right click on the profile configuration object and edit the attributes

ii. Add four rows for profile configuration

iii. Name a row as Database_User and choose database as the desired application

iv. Name a row as Email_User and choose email as the desired application

v. Name another row as Ftp_User and choose ftp as the desired application

vi. Name another row as Web_User and choose web as the desired application as shown in the screenshots below

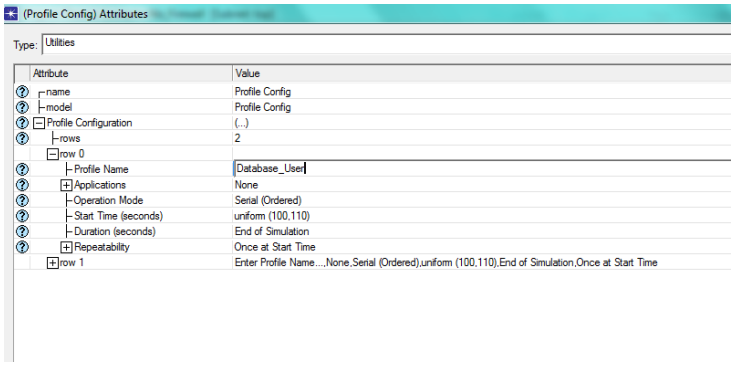

Fig.3.12: Database Profile Config

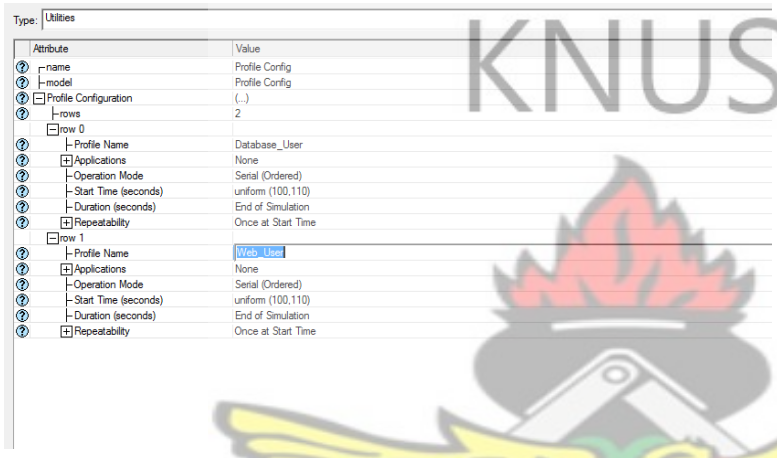

Fig.3.13: Web Profile Config

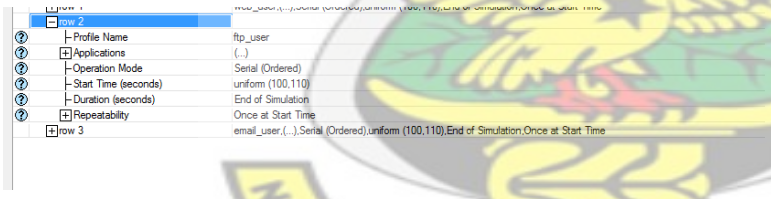

Fig.3.14 Ftp Profile Config

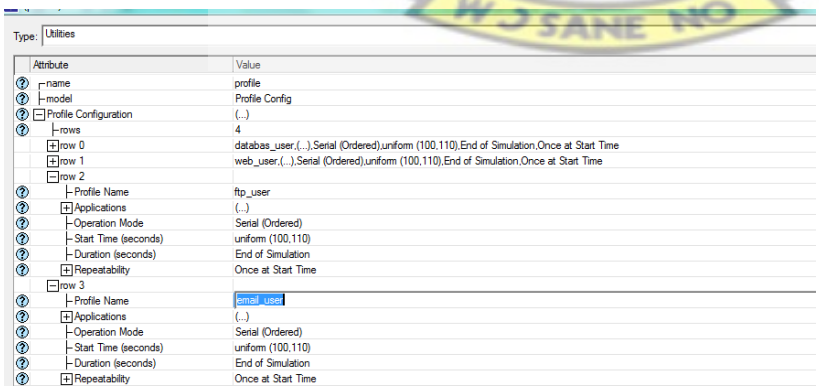

Fig.3.15 Email Profile Config

With both profile configured, the applications are ready to generate the necessary traffic.

3.6.4 Cloud/Internet Configuration

OPNET IT guru provides an IP32 cloud which acts a simple public internet based cloud. In this project this cloud is used to support the database, ftp, email and web applications. The steps show how to configure the cloud/internet.

i. Right click on the cloud and choose edit attributes

ii. Edit the packet latency attribute and set the value as constant 0.05 seconds as shown

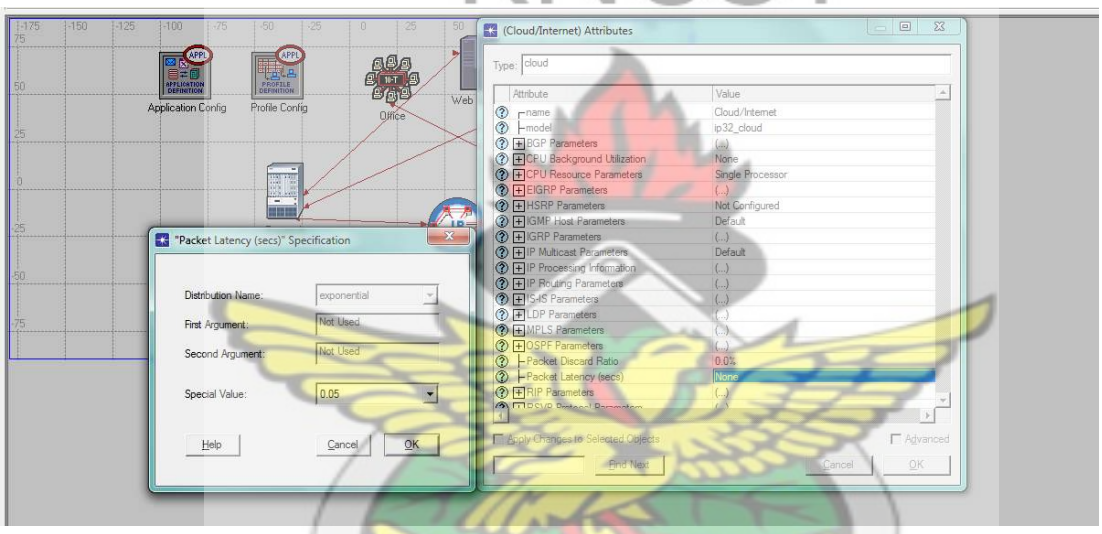

Fig.3.16: Cloud/Internet Configuration

When the packet latency is set to 0.05 seconds it indicates that, the maximum packet delay across the cloud due to the web, ftp, email and database applications is 50ms. Each and every packet is processed across the cloud with this limited delay.

3.6.5 Office Configuration

10BaseT_Switch_LAN is used to construct the office network and the following steps shows how to configure the office LAN

i. Right click on the office object and edit the attributes

ii. Number of workstations are set to 300

iii. An application supported profiles section is expanded and four rows are added

iv. Database profile is added the number of users are set to 150

v. Another profile is set to web profile and the number of users are set to 200 as depicted by the figure below

vi. The email and ftp profile are also added and the number of users are each set to 100

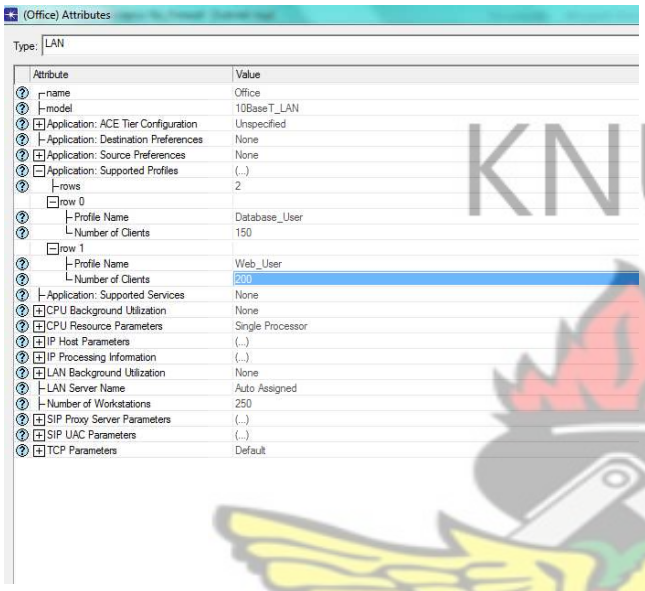

Fig.3.17: Office Configuration

The office network is connected to the Router B using 10BaseT links

3.6.6 Server Configuration

The three servers' web, file and database are configured to support web, file and database applications respectively in the following steps.

i. Right click on the database server and choose edit attributes

ii. Edit the application supported profiles and set Database application as supported

iii. A similar procedure is followed for web server, where the web application is supported at this level as shown below

iv. A similar steps are followed for the file server , where the email heavy and file transfer heavy application are supported

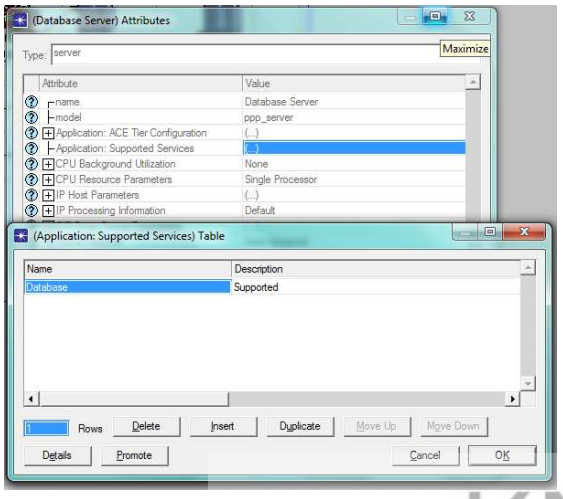

Fig.3.18: Database Server Configuration

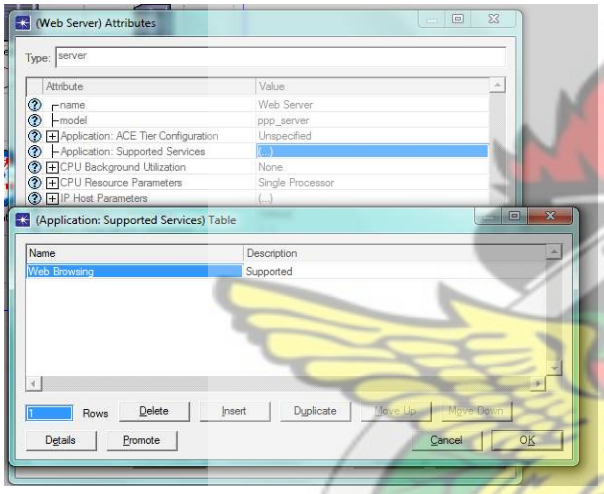

Fig.3.19: Web Server Configuration

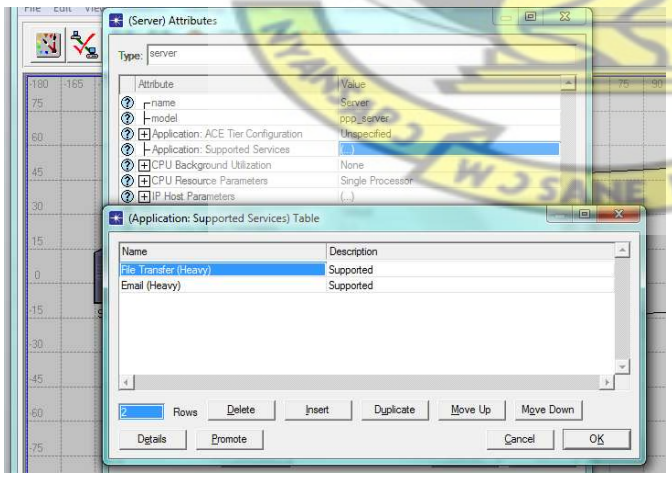

Fig.3.20 File Server Configuration

The three servers are then connected to the Router A using a PPP_DS1 links.

3.6.7 Performance Metrics

To evaluate the performance of cloud against the database and web applications few parameters are required. OPNET IT Guru provides three levels of performance evaluation like at the global level, node level and link level all of them are used in this simulation. The following steps configure the performance metrics

i. Right click on the workspace and choose the option Choose Individual statistics
ii. Now a separate window is opened where the option to choose the global, node and link level metrics is as given below

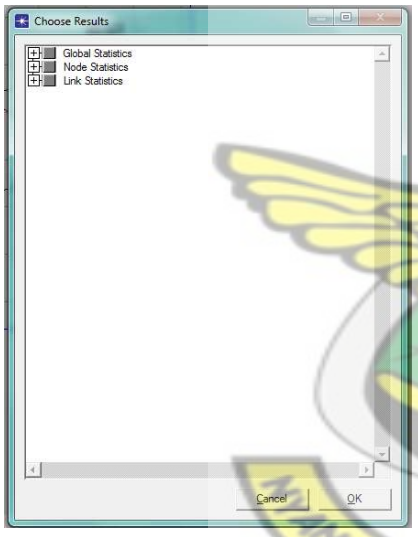

Fig.3.21: Three Level Performance Metrics

Following metrics are chosen for performance evaluation

i. From the Global level statistics, expand the DB query option and choose response time, traffic sent and received
ii. From the global level statistics, expand the Http option and choose the page response time, traffic sent and received and the corresponding Figure is as given below

iii. Also the download, upload response time, traffic sent and received options are checked for the email and ftp options.

iv. Expand the voice options and check the packet delay variation

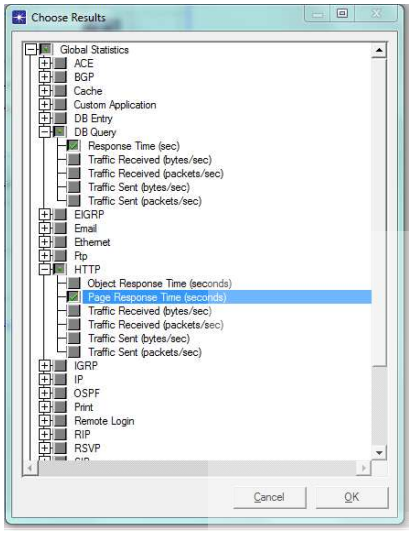

Fig.3.22: Global Statistics Performance Metrics

From the node level statistics the following metrics are chosen

i. Expand the server DB query and choose load, task processing time(sec)

ii. Expand server HTTP and choose load(request/sec) and task processing time(sec) as shown below

iii. Similarly expand the server FTP and server Email and choose load(request/sec) and task processing time(sec)

iv. Expand the IP and choice traffic drop and packet delay

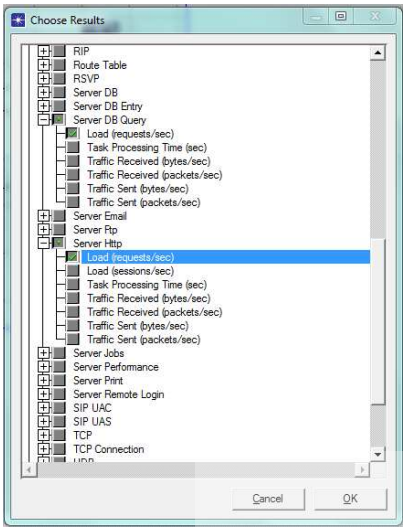

Fig.3.23: Node Level Performance Statistics

The following are chosen from the link level statistics

i. Expand point to point option and check both the utilization metrics

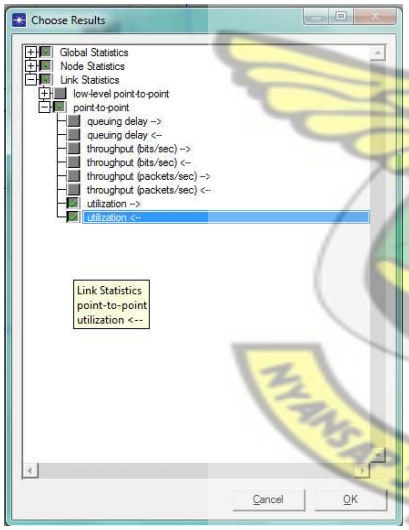

Fig.3.24: Link Level performance Statistics

The performance metrics for the first scenarios is done. The next section explains the next scenario- Firewall.

3.6.8 Firewall Scenario

We duplicate the scenario in the last section to create the scenario for this section. In this scenario we impose the firewall policies over the cloud/internet. The firewall will allow the required traffic across the network and a packet filtering is done. The duplicate procedure is shown below.

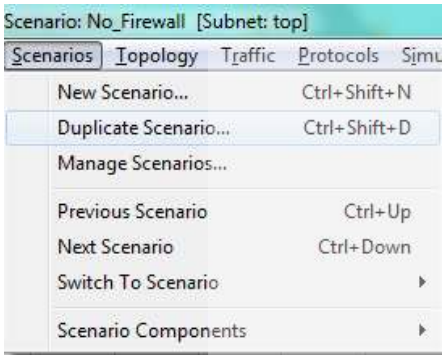

Fig.3.25: Duplicate Scenario

The steps below is used to configure the firewall

i. Right click on the Router B and edit the attributes
ii. From the option model choose, ethernet2_slip8_firewall such that the router now acts as a firewall
iii. Proxy server information option is expanded and the row 1 option is edited and the latency is set to a constant value of 0.05
iv. Expand the row 4 and set the latency to a constant value of 0.05

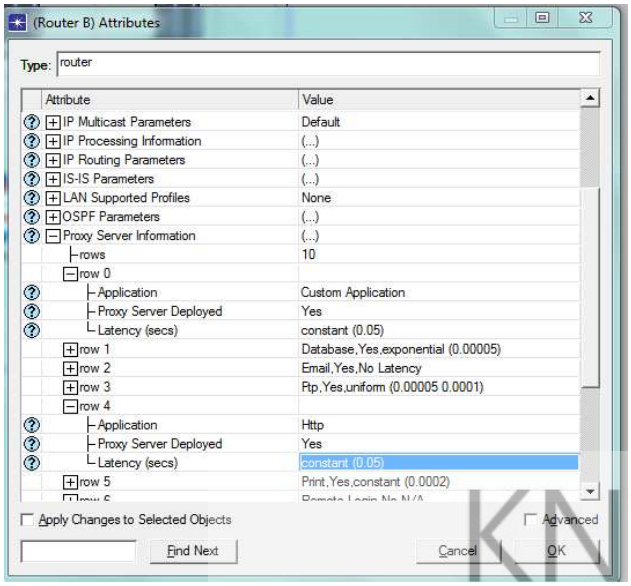

Fig.3.26: Firewall Configuration

The latency for database, file and web application is set to a constant value of 0.05 and this indicates that, the packet filtering is done at the firewall and thus a delay of 50ms is incurred over the router. The network is thus shown below

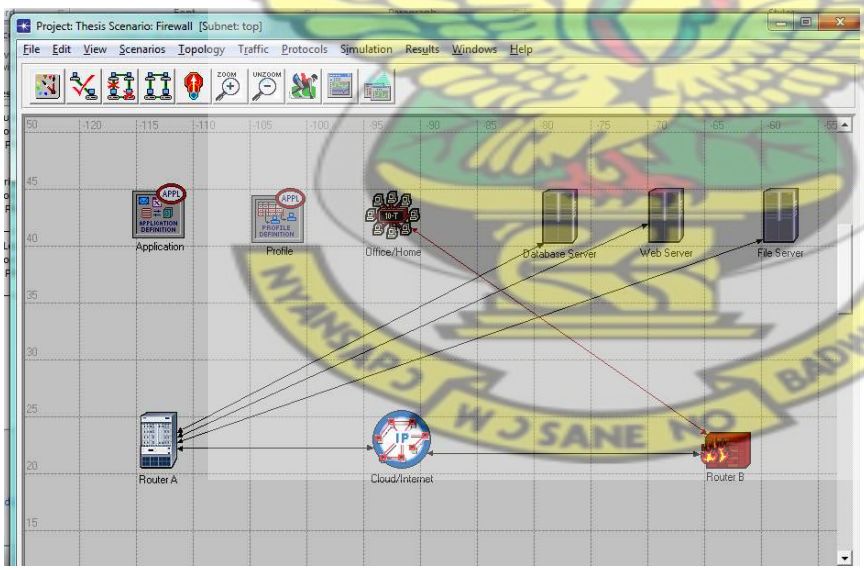

Fig.3.27: Firewall Scenario Setup

With this the simulation of firewall scenario is done and the same performance metrics are used across this scenario as in the first scenario.

3.6.9 Firewall Blocking Scenario

In this scenario we block the web traffic over the network and this scenario is created by duplicating the second scenario. Some changes are made to this network in this scenario in the following steps

i. Right click on the Router B and edit the attributes

ii. Expand the Proxy server information and choose the row 4 i.e. HTTP

iii. Set the proxy server deployed option to no as shown in the figure below

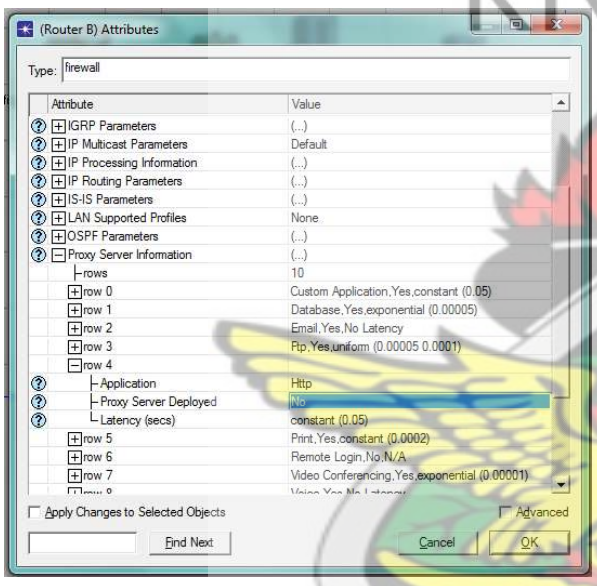

Fig.3.28: Web Traffic Block

With this all the web traffic across the cloud is blocked and thus the simulation of firewall blocking scenario is done.

3.7 Running the Simulation

Once all the three scenarios are done the simulation is run for two hours. It can be done from the scenarios menu by choosing the manage scenarios option as shown below.

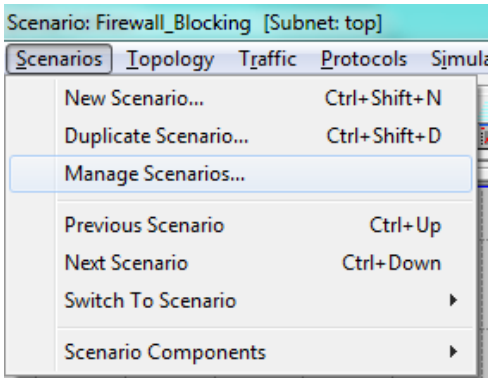

Fig.3.29: Manage Scenarios

With this option selected, a new window is opened and the simulation is run for two hours and the corresponding Figure is as shown below

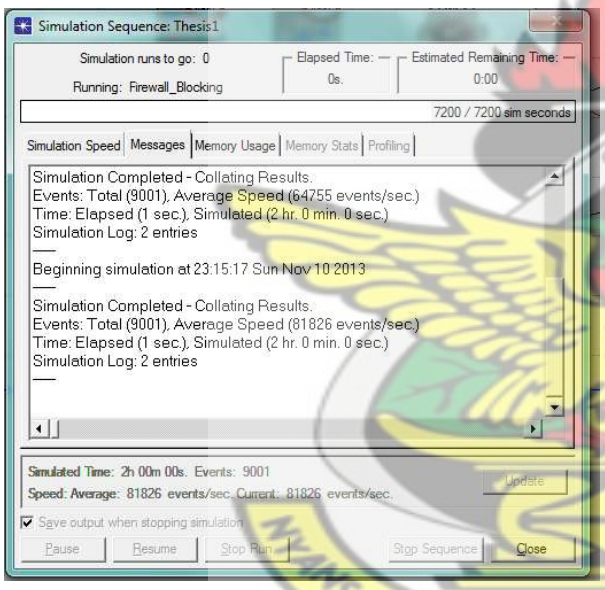

Fig.3.30: Simulation Runs for 2 hours

Once the simulation is done the next step is to evaluate the results and a detailed evaluation of results is given in the next chapter. All the three scenarios are compared against the performance metrics chosen.

3.8 Result of the Simulation Experiment

In this section, the result of the simulation of the three scenarios are presented

3.8.1 Result for Database Application

This section illustrates the result for the database application.

3.8.2 Database Query Response Time

The table below shows the database query response time, taking after the first 40 minutes of the simulation time.

Table 3.1: database query response time with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.21	0.11	0.18	0.18	0.12	0.06
Firewall	0.57	0.41	0.55	0.43	0.70	0.52
Firewall Blocking	0.22	0.22	0.25	0.18	0.21	0.21

Table 3.2: database query response time with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.03	0.02	0.03	0.01	0.02	1.15
Firewall	8.42	6.23	91.62	157.02	29.01	16.21
Firewall Blocking	1.30	1.31	1.18	1.23	1.12	1.15

Table 3.3: database query response time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.19	0.19	0.18	0.18	0.19	0.19
Firewall	132.32	197.60	171.09	245.21	100.28	139.64
Firewall Blocking	29.67	64.41	30.35	65.14	37.78	73.33

3.8.3 Server Database Query Load

The values in the table below shows the server loads across the network when no security policy is enforced.

Table 3.4: server database load with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	2.54	2.62	2.10	2.14	2.39	2.41
Firewall	0.002	0.001	0.002	0.001	0.001	0.001
Firewall Blocking	2.35	2.36	2.35	2.37	2.29	2.36

Table 3.5: server database load with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	6.14	6.14	5.72	5.73	5.40	5.39
Firewall	0.003	0.001	0.002	0.001	0.001	0.001
Firewall Blocking	5.78	6.06	5.63	4.33	6.18	6.32

Table 3.6: server database load with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	6.50	6.54	6.97	7.40	6.10	6.15
Firewall	0.001	0.001	0.001	0.001	0.001	0.001
Firewall Blocking	5.02	5.11	5.13	5.27	4.63	4.76

3.8.4 Result for E-mail Application

This sections list the result for the email application after running the simulation.

3.8.5 E-mail Download Response Time

The table below shows the e-mail download response time when no security is imposed on the network.

Table 3.7: E-mail downloads response time with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.52	0.54	0.62	0.60	0.61	0.63
Firewall	12.13	6.77	9.01	4.91	9.66	5.05
Firewall Blocking	0.003	0.003	0.003	0.003	0.003	0.003

Table 3.8: E-mail downloads response time with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.11	0.11	0.12	0.12	0.11	0.11
Firewall	3.76	3.84	3.85	3.87	3.68	3.74
Firewall Blocking	0.11	0.11	0.11	0.11	0.23	0.17

Table 3.9: E-mail downloads response time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr20mins
No Firewall	0.12	0.12	0.12	0.12	0.12	0.12
Firewall	10.29	10.16	11.29	11.29	9.35	9.49
Firewall Blocking	0.12	0.12	0.12	0.12	0.12	0.12

3.8.6 E-mail Upload Response Time

The table shows the upload response time when no firewall is imposed on the network.

Table 3.10: E-mail Upload Response Time with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.78	0.93	0.54	0.70	0.97	0.97
Firewall	0.97	1.15	0.89	1.09	2.17	1.63
Firewall Blocking	0.003	0.003	0.003	0.003	0.003	0.003

Table 3.11: E-mail Upload Response Time with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.11	0.11	0.11	0.11	0.11	0.11
Firewall	5.84	5.92	5.22	5.27	5.14	5.15
Firewall Blocking	0.11	0.11	0.11	0.11	0.11	0.11

Table 3.12: E-mail Upload Response Time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr20mins
No Firewall	0.12	0.12	0.12	0.12	0.13	0.12
Firewall	10.35	10.40	10.91	10.87	9.36	9.26
Firewall Blocking	0.12	0.12	0.12	0.12	0.11	0.13

3.8.7 Server E-mail Load

The table below shows the load on the e-mail server when no security is imposed on the network.

Table 3.13: E-mail server load with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.06	0.04	0.06	0.05	0.06	0.04
Firewall	0.02	0.03	0.02	0.02	0.02	0.02
Firewall Blocking	0	0	0	0	0	0

Table 3.14: E-mail server load with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.18	0.19	0.17	0.17	0.18	0.18
Firewall	0.003	0.001	0.004	0.002	0.003	0.002
Firewall Blocking	0	0	0	0	0	0

Table 3.15: E-mail server load with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.32	0.32	0.29	0.28	0.29	0.27
Firewall	0.005	0.003	0.005	0.003	0.005	0.002
Firewall Blocking	0	0	0	0	0	0

3.8.8 Result Web Application

The results for the web application are listed in this section.

3.8.9 Http Page Response Time

The table below shows the page response time.

Table 3.16: page response time with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.81	0.82	0.80	0.82	0.84	0.83
Firewall	4.91	3.11	4.75	3.04	6.36	4.50
Firewall Blocking	0.005	0.005	0.005	0.005	0.005	0.005

Table 3.17: page response time with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.08	0.08	0.08	0.08	0.08	0.08
Firewall	4.25	4.32	4.20	4.24	4.24	4.36
Firewall Blocking	0.08	0.08	0.08	0.08	0.08	0.08

Table 3.18: page response time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.05	0.03	0.05	0.03	0.05	0.04
Firewall	9.24	9.28	9.40	9.70	8.33	8.34
Firewall Blocking	0.02	0.02	0.02	0.02	0.02	0.02

3.8.10 Http Server Load

The table below shows the server http load in request per second when no firewall is imposed on the network

Table 3.19: server http load with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	1.06	1.01	1.03	0.92	1.06	0.91
Firewall	0.62	0.71	0.61	0.75	0.56	0.72
Firewall Blocking	0	0	0	0	0	0

Table 3.20: server http load with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	59.01	59.82	58.35	59.04	58.84	58.87
Firewall	0.02	0.009	0.02	0.007	0.009	0.006
Firewall Blocking	0	0	0	0	0	0

Table 3.21: server http load time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	9.27	9.24	9.03	9.33	9.20	9.22
Firewall	0.02	0.008	0.02	0.01	0.01	0.007
Firewall Blocking	0	0	0	0	0	0

3.8.11 Result for Ftp Application

This section list the result for the ftp application

3.8.12 Ftp Download Response Time

The results in the table below shows the ftp download response time when no firewall is imposed on the network

Table 3.22: Ftp downloads Response Time with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.52	0.61	0.60	0.60	0.77	0.82
Firewall	0.61	0.81	3.51	1.85	19.93	11.74
Firewall Blocking	0.004	0.004	0.004	0.004	0.004	0.004

Table 3.23: Ftp downloads Response Time with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.11	0.12	0.12	0.12	0.12	0.12
Firewall	3.38	3.47	3.33	3.38	3.69	3.77
Firewall Blocking	0.12	0.13	0.12	0.12	0.12	0.12

Table 3.24: Ftp downloads Response Time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.24	0.24	0.26	0.26	0.25	0.25
Firewall	26.72	28.08	28.71	28.23	23.50	23.46
Firewall Blocking	0.27	0.25	0.26	0.25	0.26	0.26

3.8.13 Ftp Upload Response Time

The following table results shows the ftp upload response time when no firewall is imposed on the network.

Table 3.25: Ftp uploads Response Time with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.03	0.03	0.03	0.02	0.03	0.02
Firewall	23.44	11.47	11.97	13.06	1.74	1.26
Firewall Blocking	0.004	0.004	0.004	0.004	0.004	0.004

Table 3.26: Ftp uploads Response Time with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.12	0.12	0.12	0.12	0.11	0.12
Firewall	5.46	5.45	6.22	6.34	7.17	7.26
Firewall Blocking	0.12	0.12	0.12	0.12	0.13	0.13

Table 3.27: Ftp uploads Response Time with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.22	0.22	0.23	0.25	0.22	0.22
Firewall	23.77	24.10	27.41	27.25	27.15	27.56
Firewall Blocking	0.23	0.24	0.23	0.23	0.23	0.23

3.8.14 Server Ftp Load

The figures in the table below show the load on the ftp server in all the three scenarios.

Table 3.28: Server Ftp load with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.03	0.03	0.03	0.02	0.03	0.02
Firewall	0.009	0.01	0.01	0.01	0.01	0.01
Firewall Blocking	0	0	0	0	0	0

Table 3.29: Server Ftp load with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.08	0.08	0.08	0.09	0.09	0.09
Firewall	0.001	0.001	0.002	0.001	0.001	0.001
Firewall Blocking	0	0	0	0	0	0

Table 3.30: Server Ftp load packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	0.14	0.14	0.14	0.14	0.16	0.17
Firewall	0.001	0.001	0.001	0.001	0.005	0.002
Firewall Blocking	0	0	0	0	0	0

3.8.15 Cloud Performance

The table below shows the link utilization across the three scenarios

Table 3.31: cloud utilization with packet size of 32MB (Low)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	1.34	0.27	0.28	0.25	0.28	0.26
Firewall	0.18	0.16	0.18	0.17	0.18	0.17
Firewall Blocking	0.07	0.07	0.07	0.07	0.07	0.07

Table 3.32: cloud utilization with packet size of 100MB (Medium)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	94.68	96.22	95.24	96.12	95.85	95.41
Firewall	0.86	0.89	0.86	0.89	0.87	0.89
Firewall Blocking	1.78	1.89	2.18	2.05	1.92	1.91

Table 3.33: cloud utilization with packet size of 200MB (High)

Scenarios	10Gbps		1Gbps		32Mbps	
	40mins	1hr 20mins	40mins	1hr 20mins	40mins	1hr 20mins
No Firewall	95.57	97.96	96.08	97.75	95.39	97.42
Firewall	0.37	0.33	0.38	0.33	0.38	0.37
Firewall Blocking	4.21	4.35	4.10	4.17	4.56	4.55

3.9 Conclusion

In this chapter we presented the methodology for evaluating the network performance against firewall security policy. Three different scenarios are simulated using OPNET IT Guru Academic Edition 9.1 as a simulation tool and the network is simulated for two hours. The results of the simulation were also presented. In the next chapter, we shall discuss the result of the experiment.

CHAPTER 4

RESULTS AND EVALUATION

4.1 Introduction

In this chapter, the result of the simulation of the three scenarios are presented and analyzed. The result discussed the evaluation result after running it for two (2) hours. The three scenarios simulated in this research work are

i. No Firewall scenario where there is no firewall security imposed on the network, so all the four applications that generated the required traffic across the distributed system are allowed to pass through the router.
ii. Firewall scenario where a firewall is imposed to filter some packet of the four applications.
iii. The third scenario like the firewall with blocking capability where the three applications are blocked and only allowed the database application to pass through.

The performance of the database, email, ftp and web application are evaluated in this chapter based on the performance metrics chosen at all the three levels like global level, node level and link level. All the obtained graphs are compared against the performance metrics and a detailed analysis is given.

4.2 Result for Database Application

The database application is one of the applications that generated traffic used in this simulation and the performance of the database application is estimated against the database query response time. A packet size of 32MB (low), 100MB(medium) and 200MB(high) are imposed across the network and a link speed of 10Mbps, 1Gbps and 10Gbps are set between the router and the cloud and the database query response time is evaluated in each packet

sizes and data rate to investigate applications performance. This section discusses the performance evaluation of the database application under the three scenarios. Database Query Response Time is the elapsed time between the end of an inquiry, query or demand on a computer system (e.g. Database server) and the beginning of a response; for example, the length of the time between an indication of the end of an inquiry and the display of the first character (result) of the response at a user terminal. The lower the query response time of a database operation, the higher the performance of the database application

4.2.1 Database Query Response Time - No Firewall Scenario

This scenario allows all the applications to pass through the router without any filtering or restriction to the flow of traffic. The tables 3.1-3.3 show the database query response time.

It can be seen from the tables that, the query response time has a low value of 0.01 seconds when no restrictions is imposed on the firewall when the packet size is 100MB.

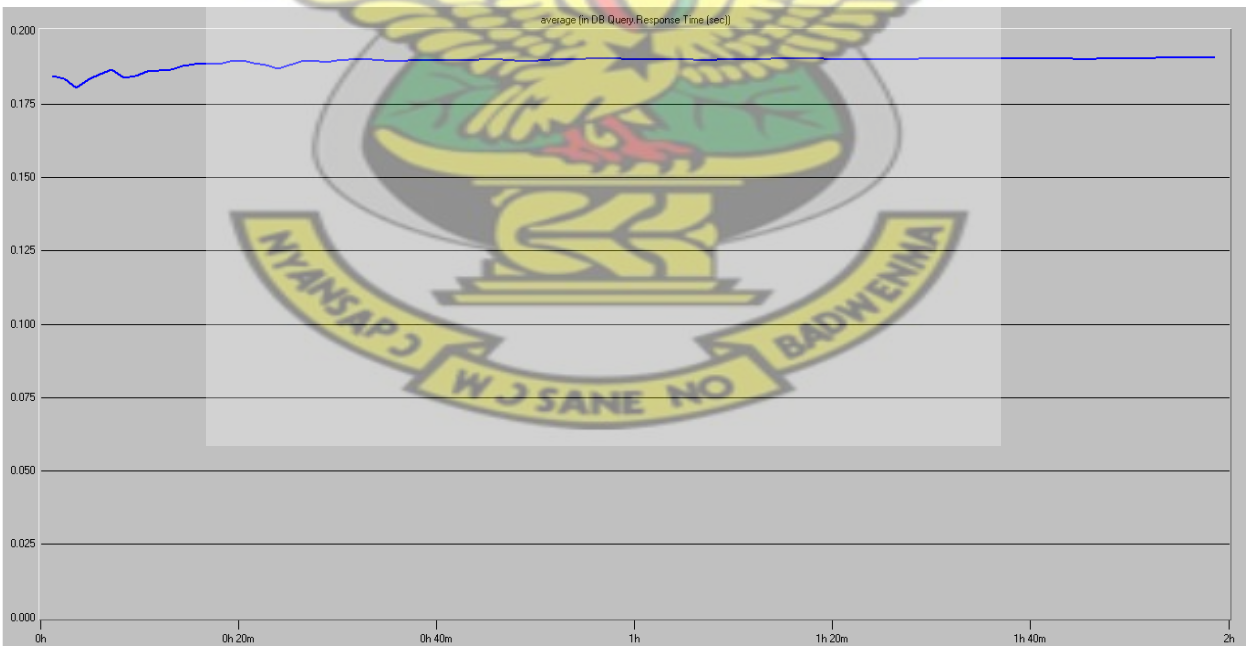

Fig.4.1: Database Query Response Time - No Firewall Scenario

From the graph above, the response time is constant throughout the simulation time with a value between 0.180 seconds and 0.191 seconds. As expected, since there is no restriction to the flow of traffic across the network, it will have a smaller response time.

4.1.1. Database Query Response Time - Firewall Scenario

In this scenario, a packet latency of 0.05 is imposed on the network to filter the packets. Latency is the time required by a system to complete a single transaction from start to finish. In this scenario, a latency of 0.05 introduces a delay of 50ms into the network. The tables 3.1-3.3 show the database query response time.

From the tables it can be analyzed that, the database query response time increases with each increase in data packet. When the simulation was simulated for the first 40 minutes with a data rate of 10Gbps, the response time was 0.57 seconds, 8.42 seconds, and 132.32 seconds respectively for data packet sizes of 32, 100 and 200MB. This implies that increasing the packet size increases the database query response time. However, the response time change just a little with different data rates. The figure below shows the database response time when firewall is imposed on the network.

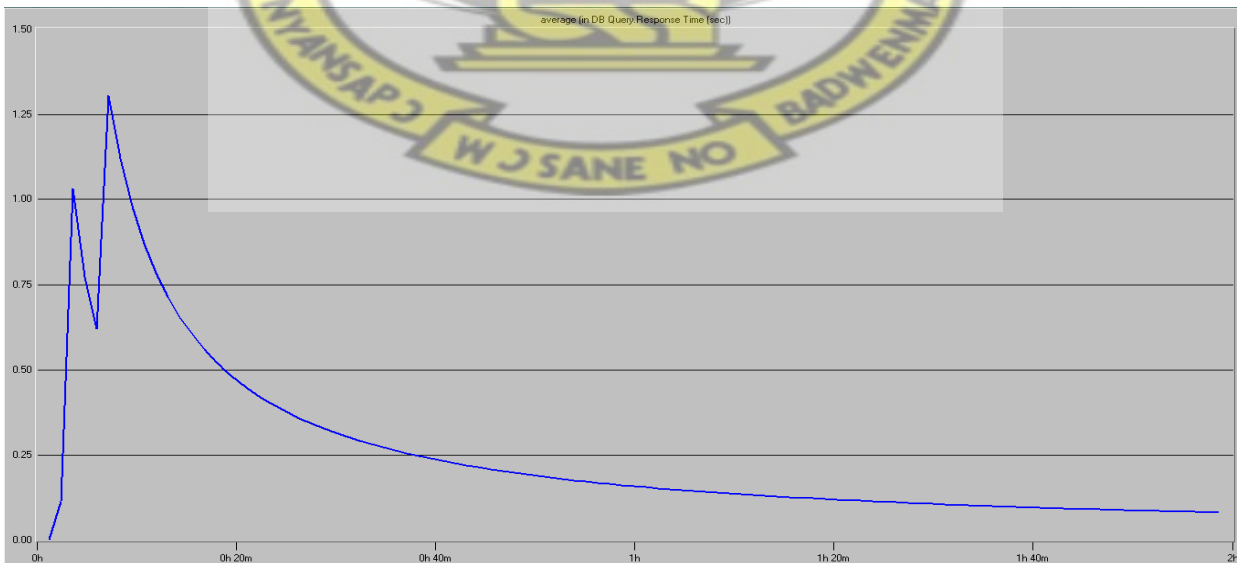

Fig.4.2: Database Query Response Time - Firewall Scenario

From the figure above, the database query response time is as high as 1.306 seconds. This is due to the packet filtering imposed on the distributed system. When the distributed system has firewall protection, everything that goes in and out of it is monitored. The firewall monitors all this information traffic to allow 'good data' in, but block 'bad data' from entering computer network. Firewalls use packet filtering methods to control traffic flowing in and out of the network. Firewall software uses pre-determined security rules to create filters – if an incoming packet of information (small chunk of data) is flagged by the filters, it is not allowed through. Packets that make it through the filters are sent to the requesting system and all others are discarded. All these activities delay the response of the systems hence a high value in the database query response time.

4.2.2 Database Query Response Time - Firewall Blocking Scenario

In the third scenario, the functionality of the firewall is further increased incorporating filtering http, email and ftp traffics entering the system. The tables 3.1-3.3 show the response time when other applications are filtered by the firewall.

The graph below shows the database query response time when other applications are blocked. It has a response time of 5.196 seconds. This is due to the fact that all other applications are blocked and only the database application goes through. As the packets reaches the firewall interface, the firewall looks at its filter table before making the decision to allow only the database packet to pass through hence the higher value since some packet filtering also occurs before decision are taken.

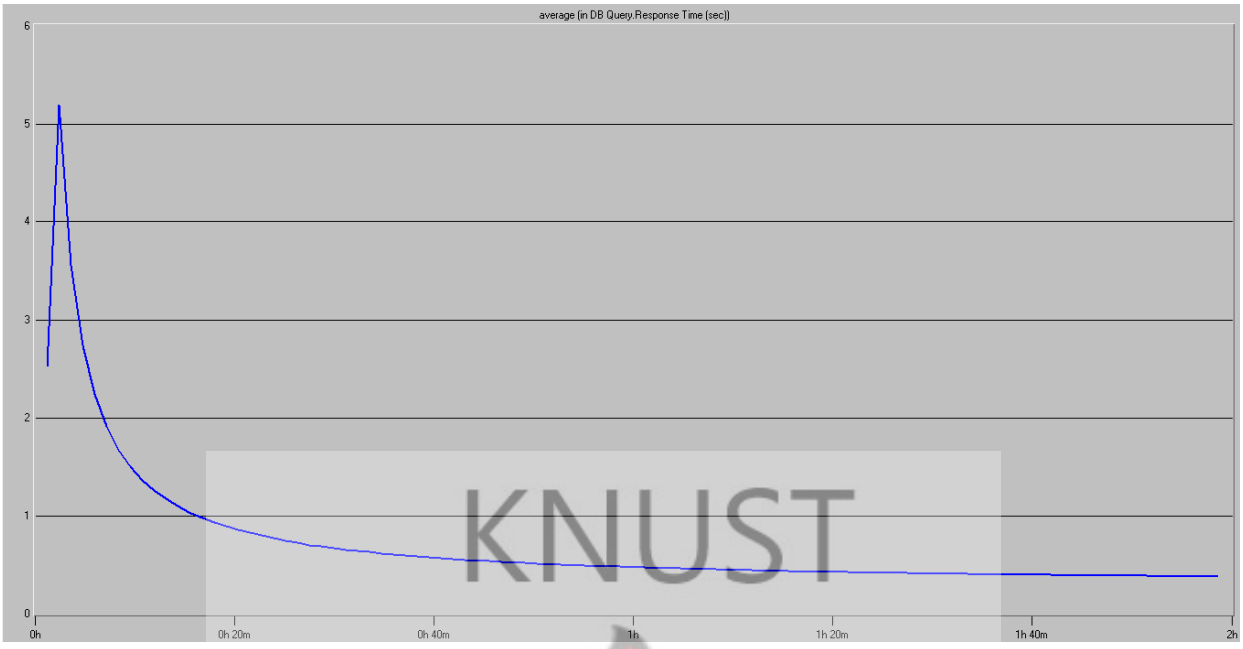

Fig.4.3: Database Query Response Time - Firewall Blocking Scenario

4.2.3 Server Database Query Load

The load on the database server is evaluated and analyzed in this section. The server database query load is in request per seconds. The higher the value of the server database query load, the longer user request from the database server waits and this degrades the performance of the server.

4.2.4 Server Database Query Load - No Firewall Scenario

The values in the tables 3.4-3.6 show the server loads across the network when no security policy is enforced.

From the tables, it can be seen that the load on the server is more when the data packet size is high. A packet of 200MB that traverse on the network has a load of 6.50 seconds and 6.54 seconds respectively when simulation runs for 40 minutes and 1hour 20 minutes. The figure below shows the load on the server.

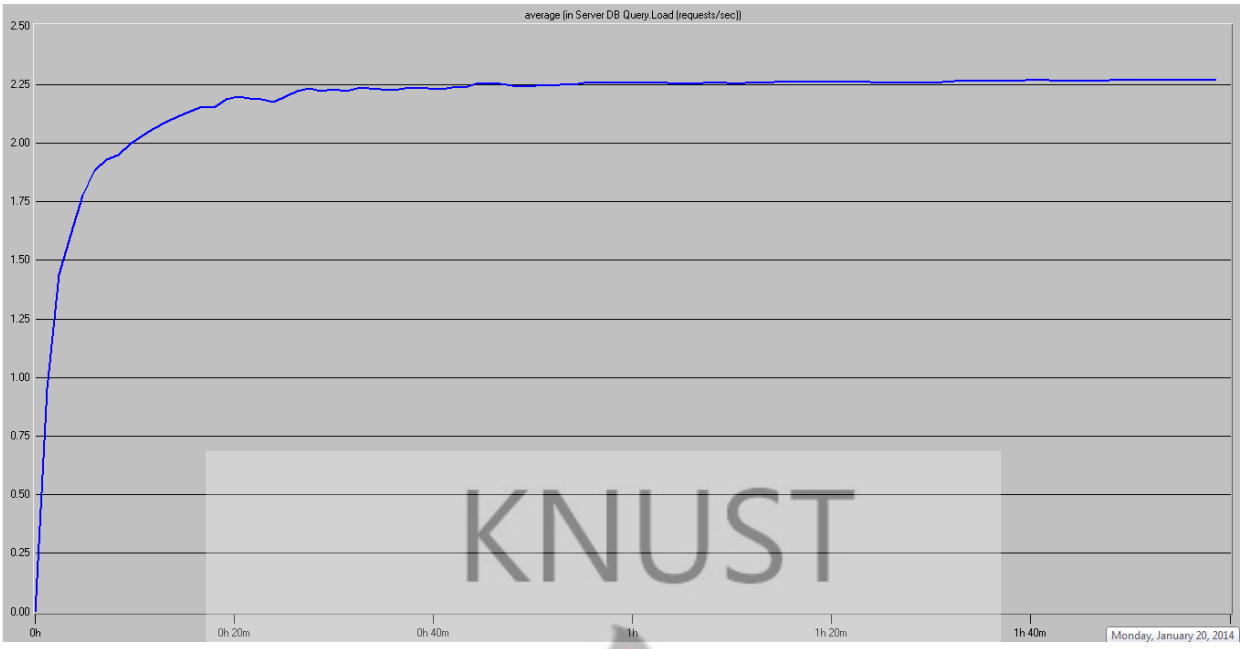

Fig.4.4: Server Database Query Load - No Firewall Scenario

The higher load on the database server is as a result of a lot of request that goes through the router into the server. A database server (program) has defined load limits, it can handle only a limited number of concurrent client connections per IP address (and TCP port) and it can serve only a certain maximum number of requests per second. When a web server is near to or over its limit, it becomes unresponsive and this leads to a delay in user request.

When no security is applied, almost all the clients are connecting to the database server in a short interval, and this increases the load on the server since the server tries to process each request. The network slowdowns, so that client requests are served more slowly and the number of connections increases so much that server limits are reached.

4.2.5 Server Database Query Load - Firewall Scenario

The tables 3.4-3.6 show the load on the server when firewall is imposed on the network.

As expected, the load on the server is very low when security is enforced across the network. Since most of the unwanted traffics have been blocked, only the legitimate packets goes into the server, so it quickly response to the user request hence the low value.

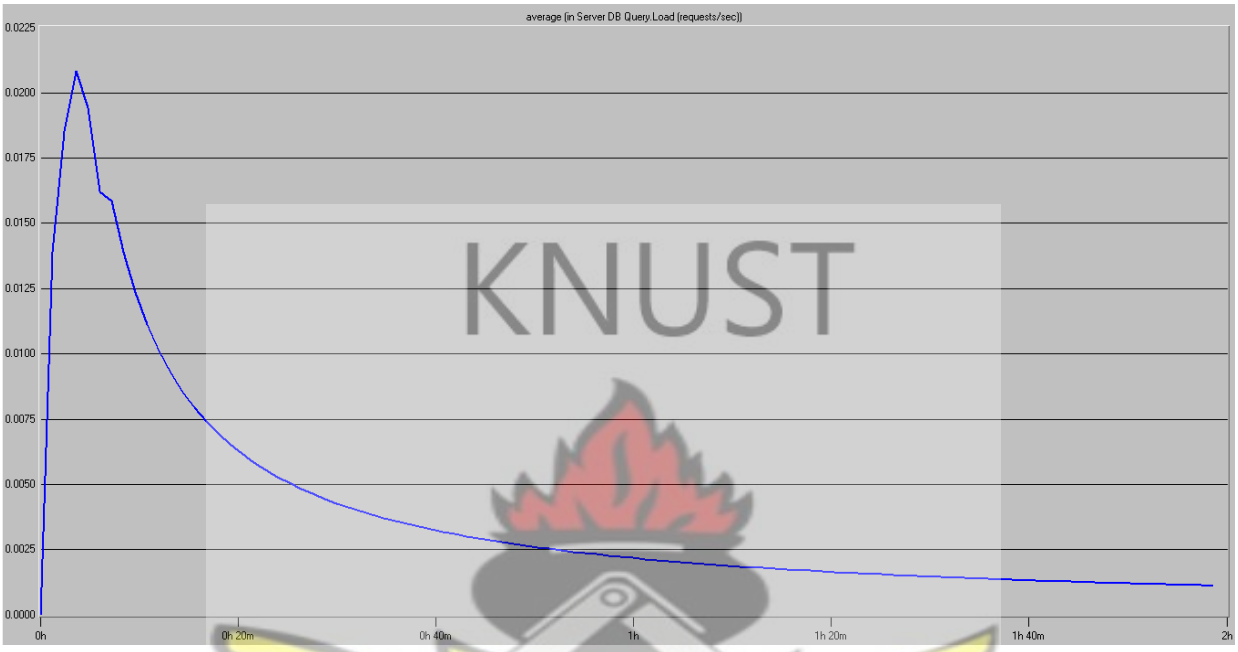

Fig.4.5: Server Database Query Load - Firewall Scenario

The figure above shows the load when firewall is imposed. From the graph, it has a higher value of 0.0208 seconds during the iniatial stage and drops to 0.0032 seconds and finally with 0.0013 seconds. This shows that when firewall is imposed, because of the filtering of the unwamted packets, only some small legitimate packets goes to the server, hence the low load. When firewall is imposed, it limits the number of request that goes to the server for processing hence the lesser load on the server.

4.2.6 Server Database Query Load - Firewall Blocking

In the third scenario, all the other applications are blocked and only the database application is allowed access through the router. The tables 3.4-3.6 show the load when others applications are blocked.

The load on the server is high and almost equals the value when no firewall is imposed. In this scenario, all the applications are blocked and only the database application gets to the server, hence the higher load.

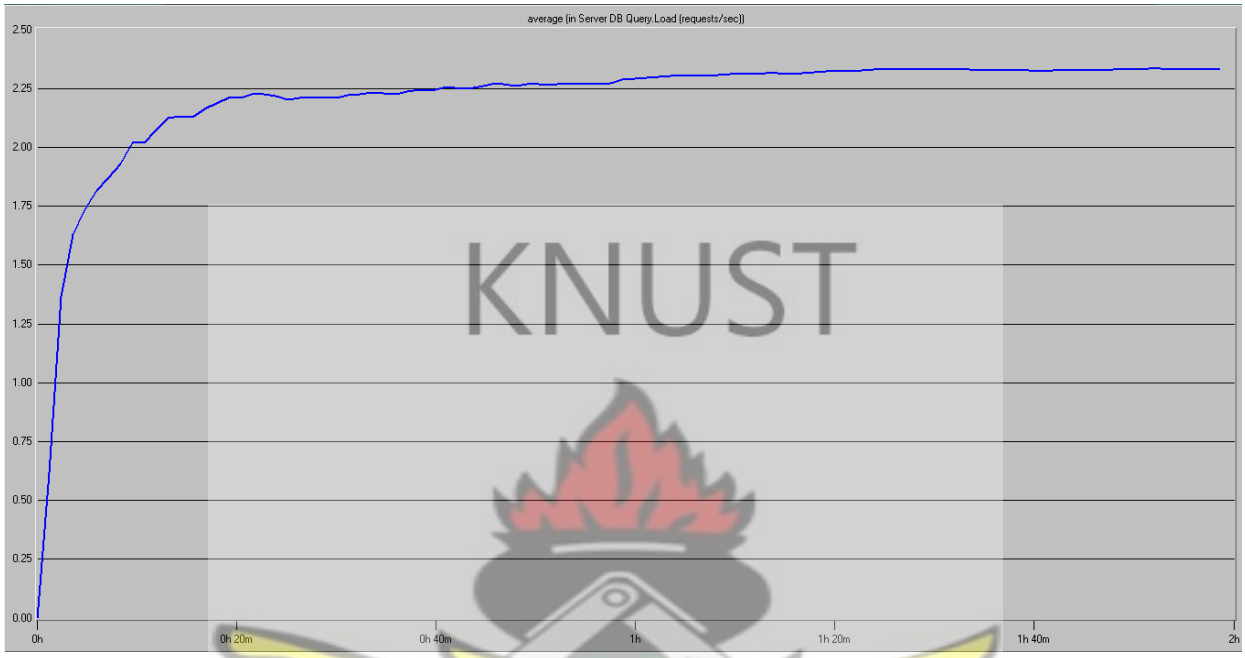

Fig.4.6: Server Database Query Load - Firewall Blocking Scenario

From the figure above, it has an initial load of 2.0249 seconds and later increase to 2.2975 seconds and later has a value of 2.3332 seconds. As expected, the load on the server in this case is high because the database application goes into the server without any restrictions hence the higher value. Combining the graphs of fig.4.4, fig.4.5, and fig.4.6, the resultant graph is shown below

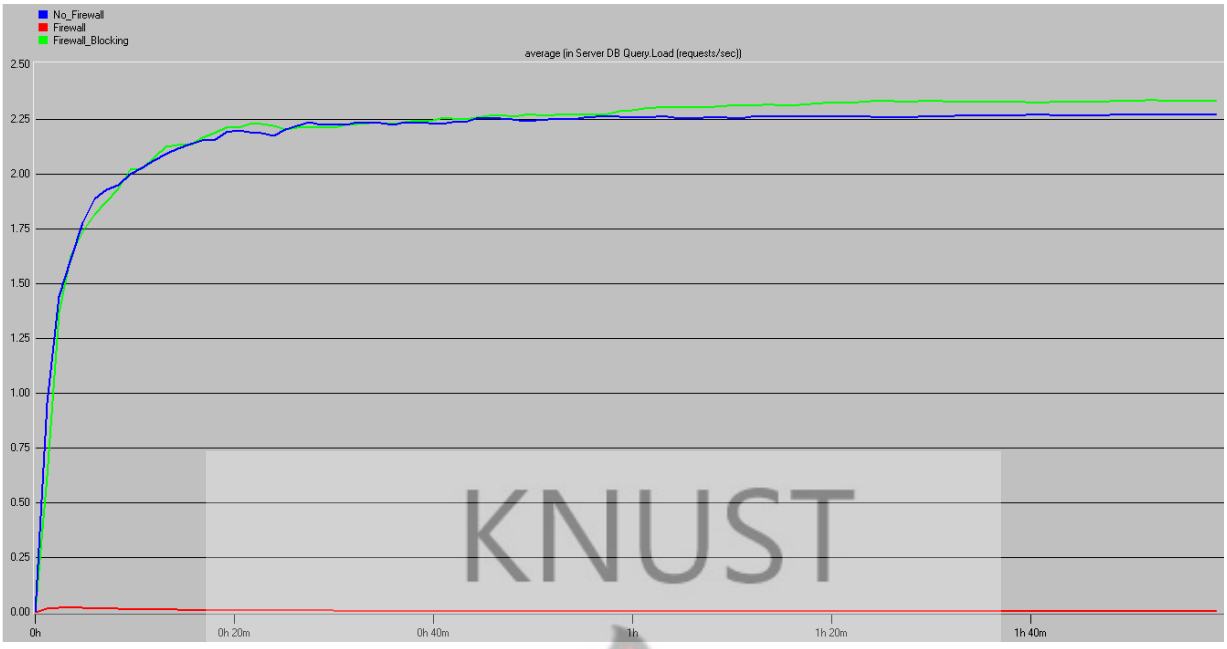

Fig.4.7: Server Database Query Load

From the graph, it can be seen that the load on server is almost equal in the case of firewall blocking and no firewall scenario. It has a low constant value when firewall is imposed on the networked. The overall analysis is that, imposing firewall on the network degrades system performance since users experince a delay in the system due to packet filtering on the router.

4.3 Result for E-mail Application

E-mail application is evaluated in this section against the email downloads and uploads response time when the three scenarios are considered. E-mail with packet size of 32MB, 100MB and 200MB are used and link speed of 10Gbps, 1Gbps and 32Mbps are used and evaluated against the performance metrics.

4.3.1 E-mail Download Response Time – No Firewall Scenario

The tables 3.7-3.9 show the e-mail download response time when no security is imposed on the network.

From the tables, it can be observed that the download response time has a low value of 0.12 seconds and 0.11 seconds when a packet of 100MB and 200MB are considered. Since there is now limitation to the flow of packets through the router, the download time is very low.

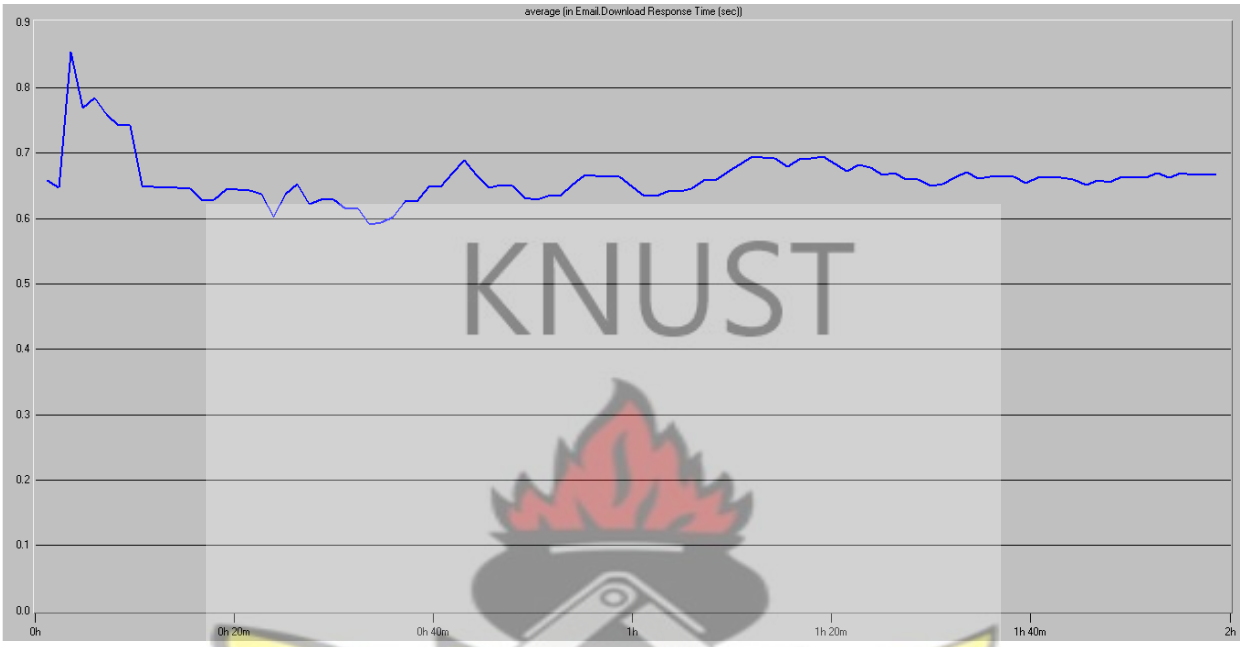

Fig.4.8: E-mail download response time - No Firewall Scenario

The above figure shows the email download response time. Since there is no blockage to the packets it leads to a faster download time.

4.3.2 E-mail Download Response Time – Firewall Scenario

The result of the download response time when security is imposed on the network is shown in tables 3.7-3.9. The result shows an increase in downloads response time when data packet sizes increases. It can be seen that, the download response time is 5.84 seconds for 100MB and 10.35 seconds for 200MB. The more the packet size the longer it takes to download from the email server. Also from the tables, it can be seen that the download response time is almost equal across the different data rates imposed. The figure below shows the download response time

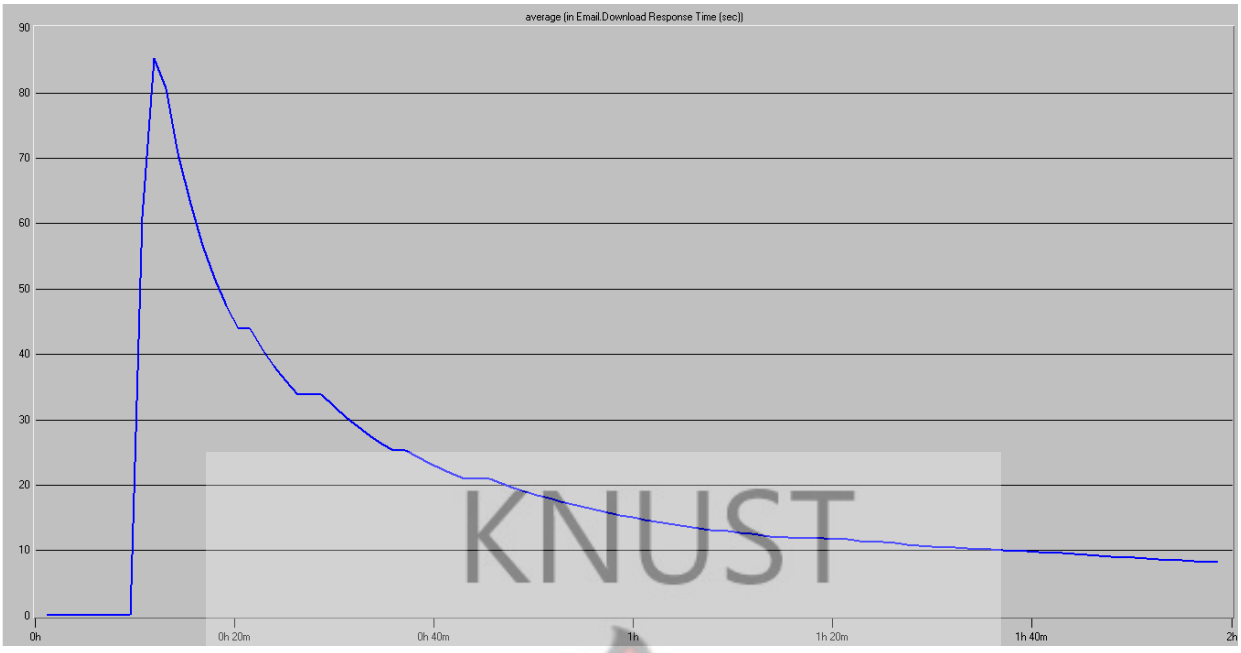

Fig.4.9: E-mail download response time - Firewall Scenario

The graph above shows a high value in the download resposne time. In order to filter traffic, firewalls use predefined rule sets. These are ordered lists that contain packet qualities matched to an action. The actions a firewall can take are to block a packet from crossing or to allow it to do so. Rules separate packets based on several qualities within the header. Many rules distinguish packets based on their origin and destination. Rule lists are written sequentially. The firewall iterates through rules and stops at the first rule that matches the packet being held. Many rules may apply to a packet. If a packet does not match any rule, it is filtered based on the default rule. Most firewalls allow outbound packets by default and block inbound packets by default. When security is imposed on the router, the router takes an extra time processing the incoming request against its policy and deciding wheather to allow a packet through or drop it. This therefore increase the time it takes to download hence users experience some delay in downloading their files.

4.3.3 E-mail Download Response Time – Firewall Blocking Scenario

In the third scenario, the email application is blocked at the router and only the database application is allowed access through the router. Tables 3.7-3.9 show the result when email is blocked. When the email is blocked, it has a value of 0.12 seconds for 200MB data size. When the packet reaches the router interface, it checks it against its policy rule before detecting that the email application is not allowed to pass through hence the value of 0.12 seconds.

4.1.2. E-mail Upload Response Time – No Firewall Scenario

The upload response time is evaluated in this section. The tables 3.10-3.12 show the upload response time when no firewall is imposed on the network.

As expected, since there is no blockage to the flow of traffic, the user experience low response time when uploading files to the email server.

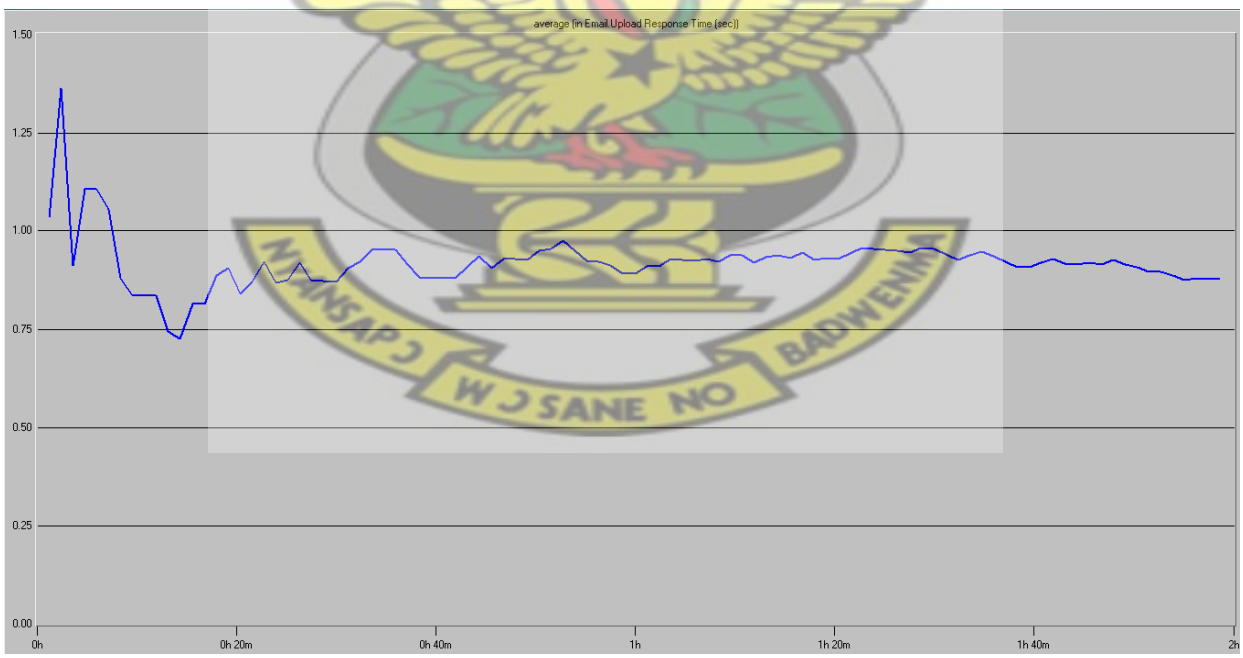

Fig.4.10: Email Upload Response Time – No Firewall Scenario

The figure above shows the upload response time. It has a peak value of 1.3621 seconds and minimum value of 0.7313 seconds when the simulation was run for 2 hours. The low value is due to no restriction to the flow of traffic.

4.3.4 E-mail Upload Response Time – Firewall Scenario

The tables 3.10-3.12 show the values when firewall is implemented on the network. A packet latency of 0.05 is imposed to induce delay into the system.

A firewall is a piece of hardware or software that is capable of filtering network traffic. This is generally performed strictly based upon the origin and/or destination of the data packets. A packet is a container used to break up large messages into smaller more manageable segments. Each packet contains a header and data. The header contains its origin address, destination address and other information about the packet itself. Firewalls go through a simple three step process to determine whether a packet should be accepted or rejected. The firewall first analyzes the packet header. It then uses this information to determine if the packet matches any open connections within the state filter. Finally, if it does not match any state, a predetermined rule set is used to determine the action that should be taken. Due to the filtering of packet at the router, users experience a delay when uploading file into the server, hence the higher value in the upload response time.

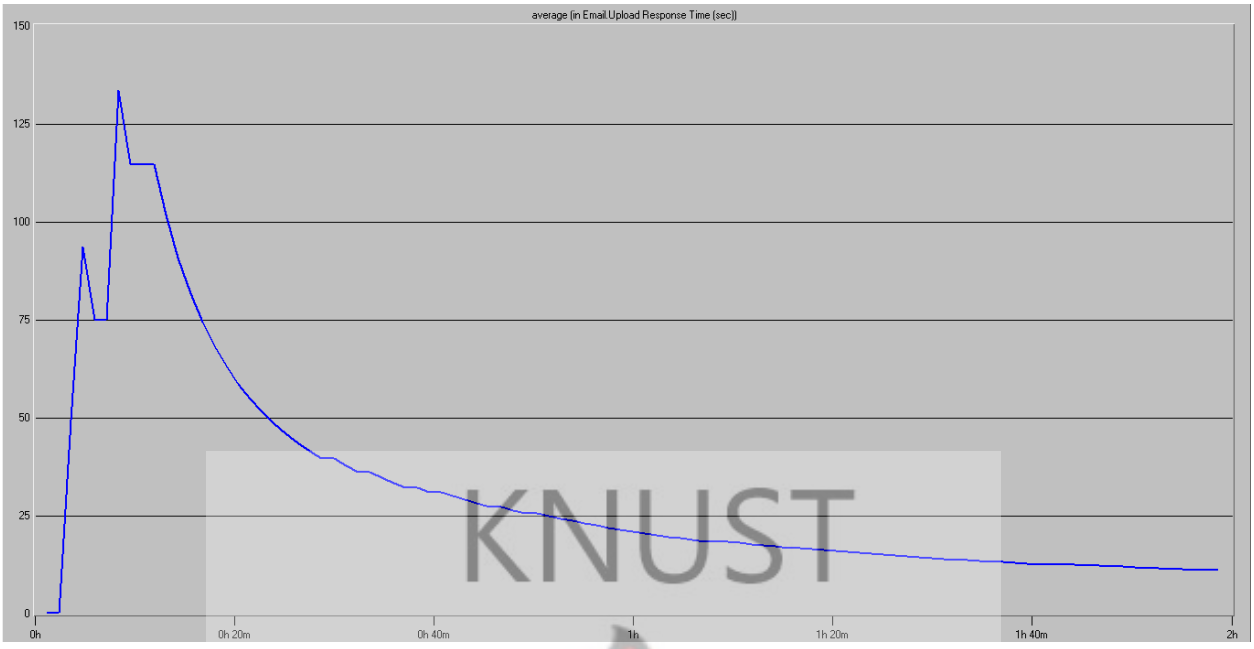

Fig.4.11: E-mail upload response time - Firewall Scenario

Due to the extra processing incured on the router, as it takes time to process each packet that tries to access the email server, the higher value in the response time.

4.3.5 E-mail Upload Response Time – Firewall Blocking Scenario

In the third scenario, the email application is blocked by the router. The results in tables 3.10-3.12 show the response time when email is blocked.

When a packet of size 100MB tries to be uploaded into the server, the user experiences a delay of 0.11 seconds. In this scenario, the email application is blocked and the router takes 0.11 seconds before returning a denial request to the user since it must take some time to check its security policy before deciding which packet to pass through.

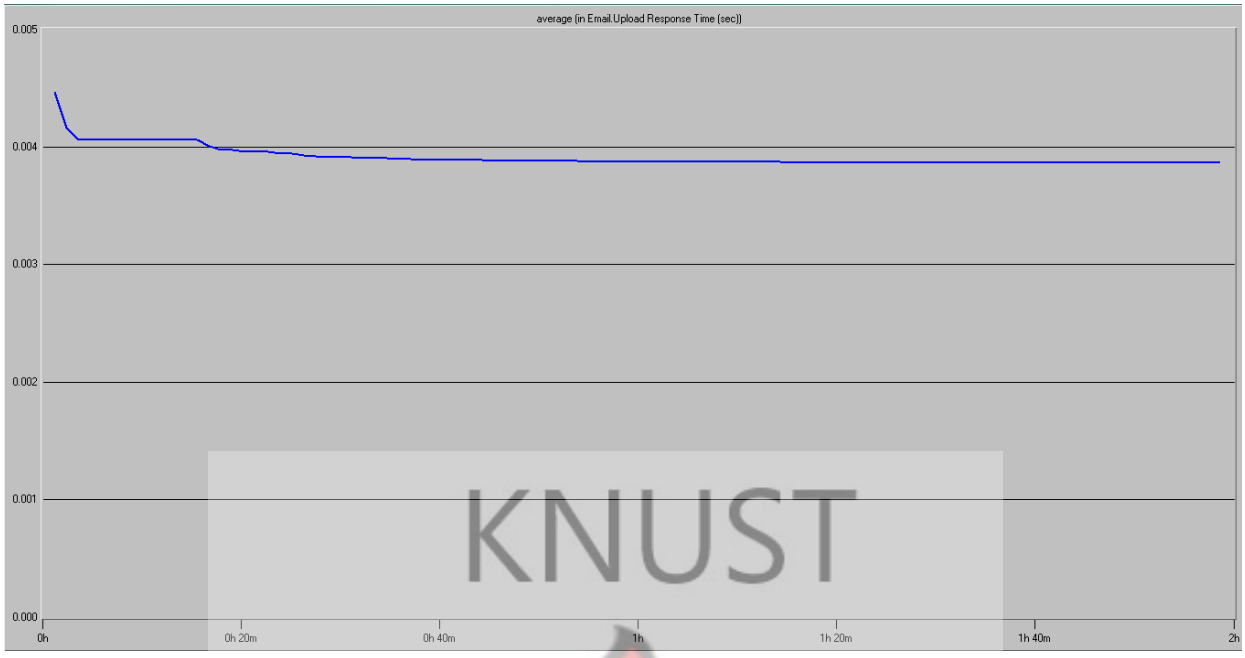

Fig.4.12: E-mail upload response time - Firewall Blocking Scenario

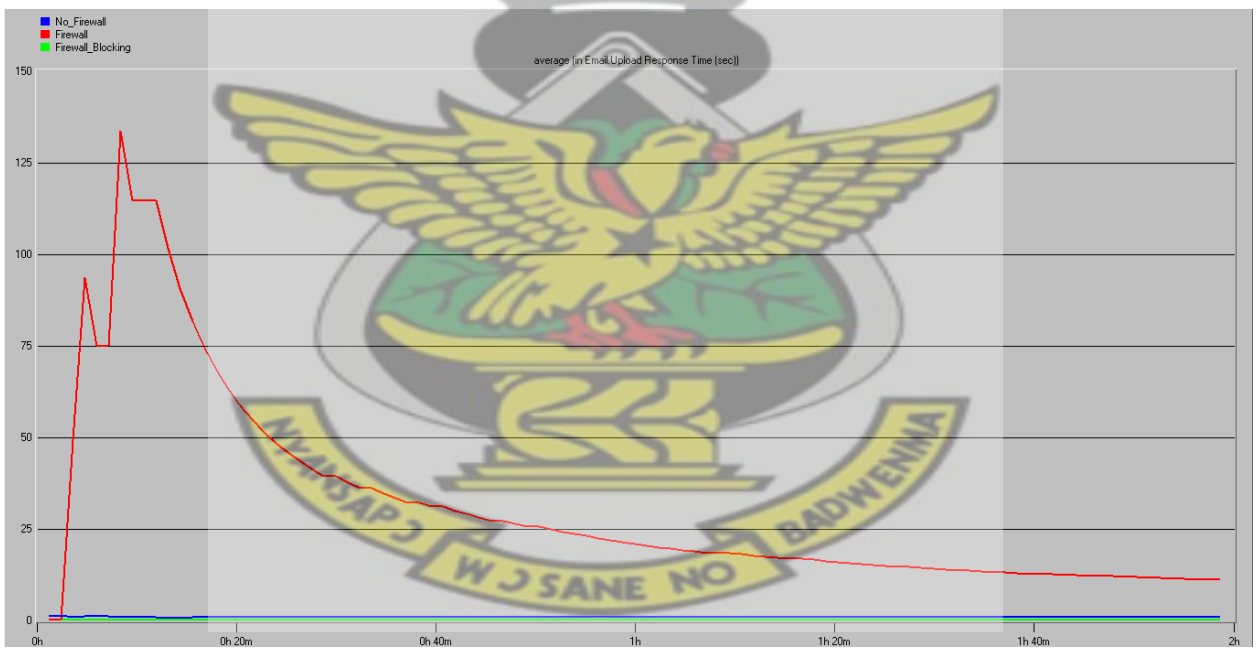

Fig.4.13: E-mail upload response time

The figure above shows the results when all the three scenarios are considered. It can be seen that the upload response time is high when firewall is imposed on the network. Due to the extra processing time taken by the router to process all packets that wants access, the router takes time to examine its policy table before allowing access hence the high response time. In

the no firewall scenario, the response time is low since there is no processing done to the packet at the router. With no firewall, the system performance is enhanced. But the system degrades as more security is imposed on the network.

4.3.6 Server E-mail Load

The load on the email server is evaluated in this section under the three scenarios

4.3.6.1 Server E- mail load – No firewall Scenario

The tables 3.37-3.39 show the load on the e-mail server when no security is imposed on the network.

When the data packet is 200MB, the server takes 0.32 seconds to process user request. This result is high as compare to other scenarios. When no security is imposed on the network, a lot of traffics get to the server and that put too much load on the server since the server must spend some time to process each request.

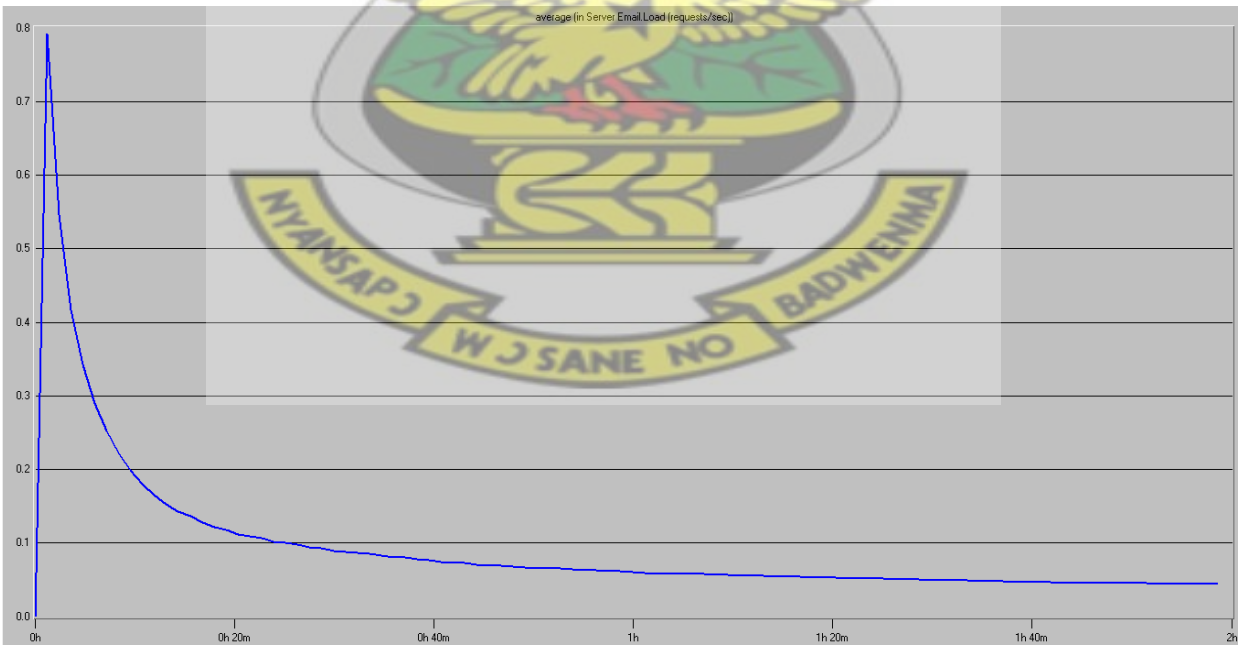

Fig.4.14: Server E-mail Load – No Firewall Scenario

In this scenario, since there is no limit to the flow of traffic, a lot of requests get to the server and these accounts for high load when compared with the other scenario.

4.3.7 Server E-mail load – Firewall Scenario

When firewall is implemented to filter some of the unwanted traffic across the network, the load on the server is shown in tables 3.13-3.15.

It can be observed that the load is very low when firewall is implemented. As the firewall blocks some packet from passing through to the e-mail server, the server has little processing to be done since only legitimate packet gets to the server for processing hence the low response time of 0.005 seconds when 200MB of data is being downloaded.

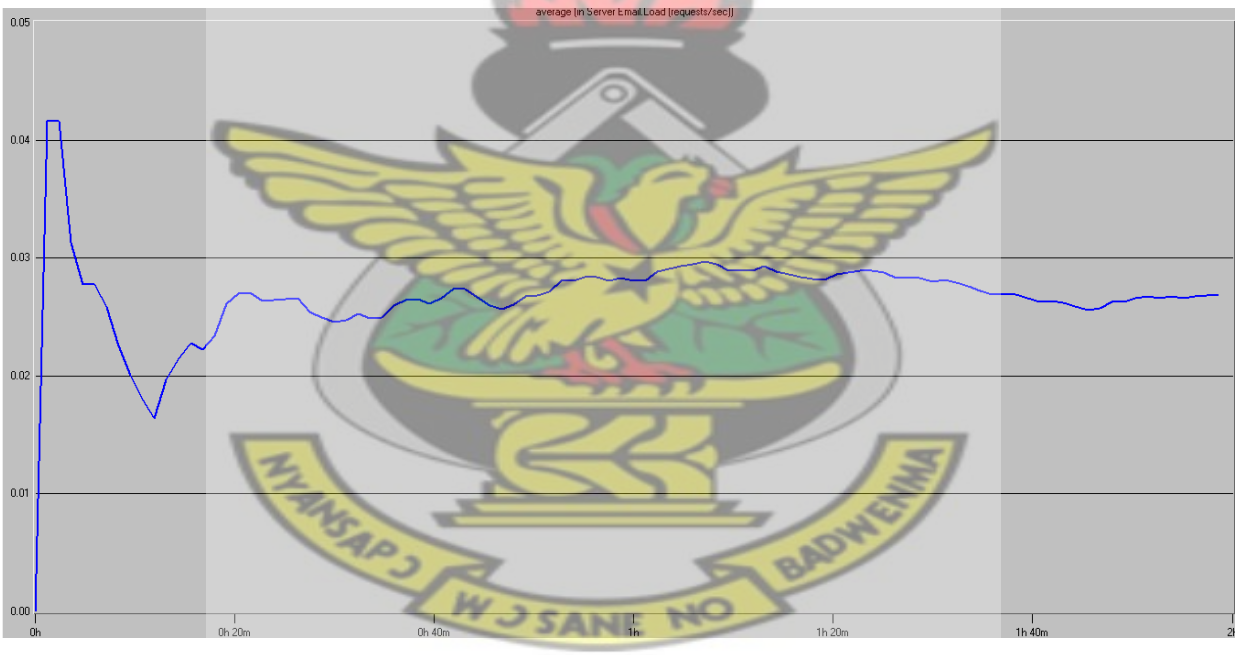

Fig.4.15: Server E-mail Load – Firewall Scenario

Since the packets are deny access or block by the router, only legitimate packets goes to the server for processing. The server has an ample time to process each request quickly hence the low load on the server.

4.3.8 Server E-mail load – Firewall Blocking Scenario

In the third scenario, the email application is blocked by the router and only the database application is allowed access through the router. Table 3.13-3.15 show the load when the email application is blocked

From the tables, it is evident that the load is 0 seconds since the application is prevented by the router.

The figure below shows the load on the server when the three scenarios are considered.

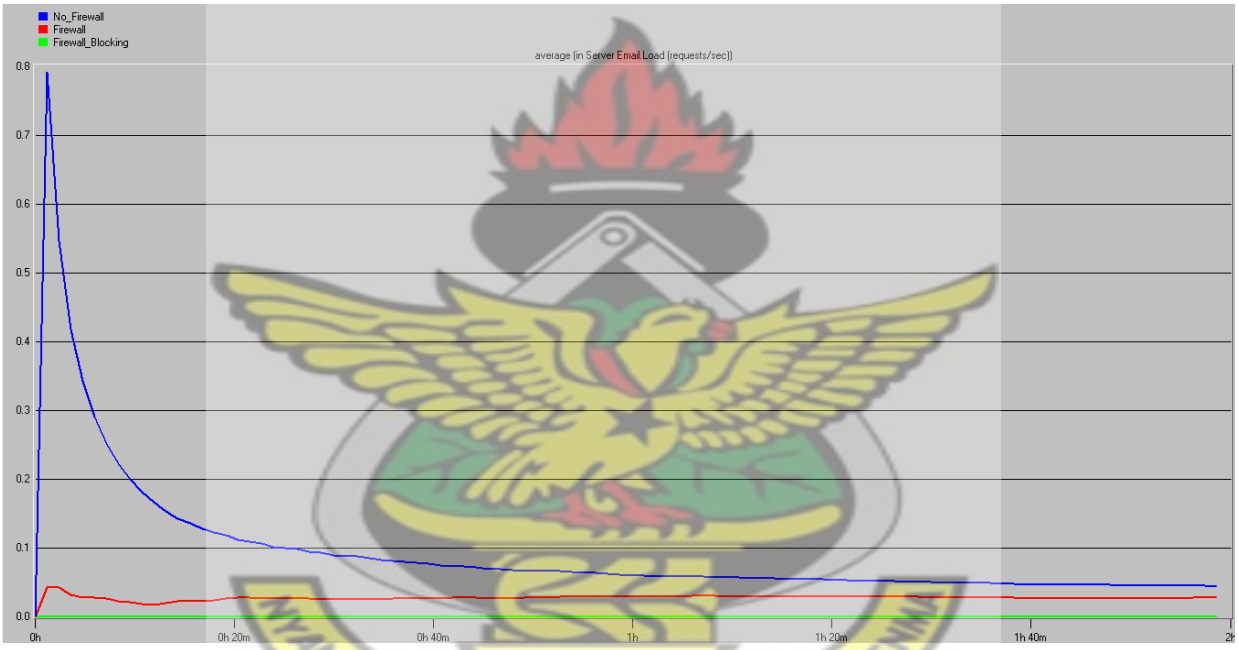

Fig.4.16: Server E-mail Load

It can be observed that the load on the server is high when no firewall is implemented. A lot of user request gets to the server, so the server takes time to grant each request hence the high load on the server. When firewall is imposed, the load on the server is reduced drastically as can be seen from the figure. Due to the overhead of the router processing packets and deciding on which one to allow access or deny, only small legitimate packets gets to the

server for processing hence the low load. In the third scenario, the email application is blocked so there is no processing on the server.

4.4 Result for Web Application

The web application is one of the applications that generated traffic used in this simulation and the performance of the web application is estimated against the page response time. A packet size of 32MB (low), 100MB(medium) and 200MB(high) are imposed across the network and a link speed of 10Mbps, 1Gbps and 10Gbps are set between the router and the cloud and the page response time is evaluated in each packet sizes and data rate to investigate applications performance.

4.4.1 Http Page Response Time - No Firewall Scenario

Tables 3.16-3.18 show the page response time when no security has been imposed on the network.

From the tables, the page response time has a low value of 0.08 seconds and 0.03 seconds when packet size of 100MB and 200 MB are considered. The lower response time shows that there is no restriction to the flow of traffic across the network.

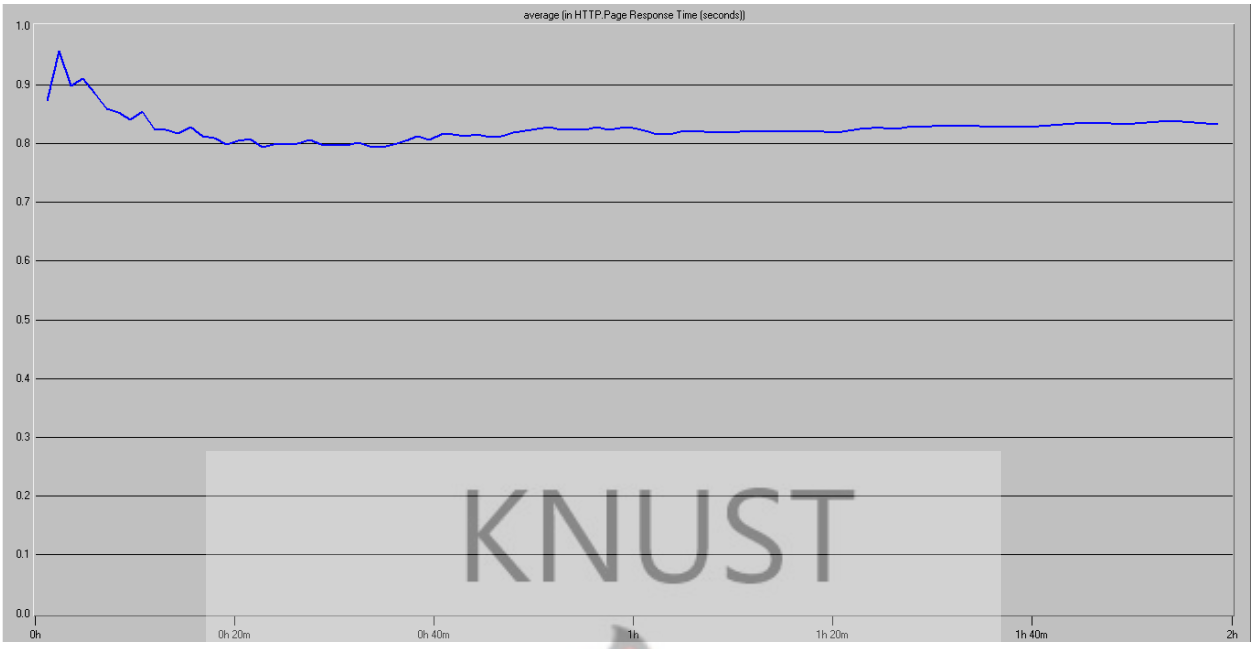

Fig.4.17 Http Page Response Time – No firewall Scenario

The figure above shows the page response time when no firewall is imposed across the network.

4.4.2 Http Page Response Time - Firewall Scenario

The tables 3.16-3.18 show the page response time when firewall is implemented in the system. The firewall introduces some delay into the network.

From the tables, the page response time has a higher value when the data packet size is 200MB. Page response time is very high when there is a firewall implementation over the cloud and when packet size increases. Due to the security policies and the packet latency time imposed over the firewall, the overall page response time is increased.

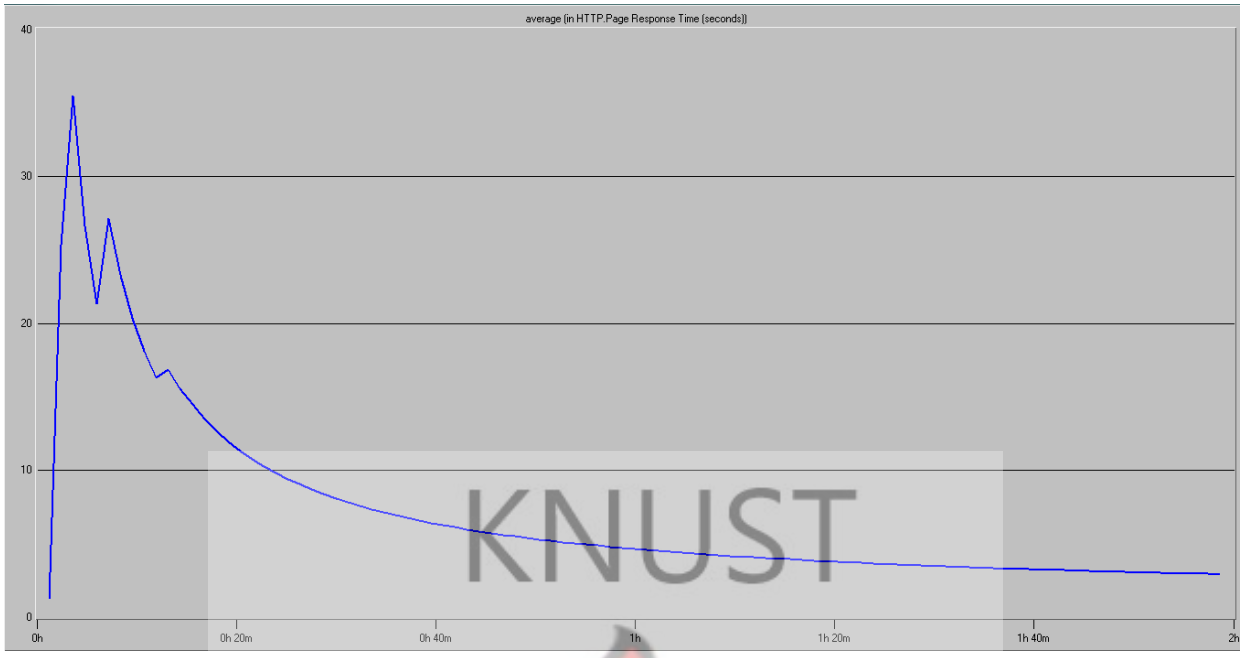

Fig.4.18: Http Page Response Time – Firewall Scenario

Firewalls sorting traffic is a relatively simple process. First, they examine the packet headers. Then, they check the active state table for matches. Finally, they search through the predefined rule set until a match is found. Every packet will either match a state or rule and therefore be blocked or admitted. If a packet is to be blocked, it is simply not forwarded. The next packet to be examined will overwrite it and it will disappear. If a packet is allowed to pass, it is pushed through the firewall towards its destination before the packet behind it in line can overwrite it. From the above graph it can be understood that the average page response time is more when there is a firewall. This is due to the packet filtering, and the packet latency of 0.05 set across the firewall router and thus the delay is incurred in the system.

4.4.3 Http Page Response Time - Blocking Firewall Scenario

The page response time when the other applications are blocked is shown in tables 3.16-3.18. As expected, the page response time is very low when the other applications are blocked. It has a low value of 0.02 seconds.

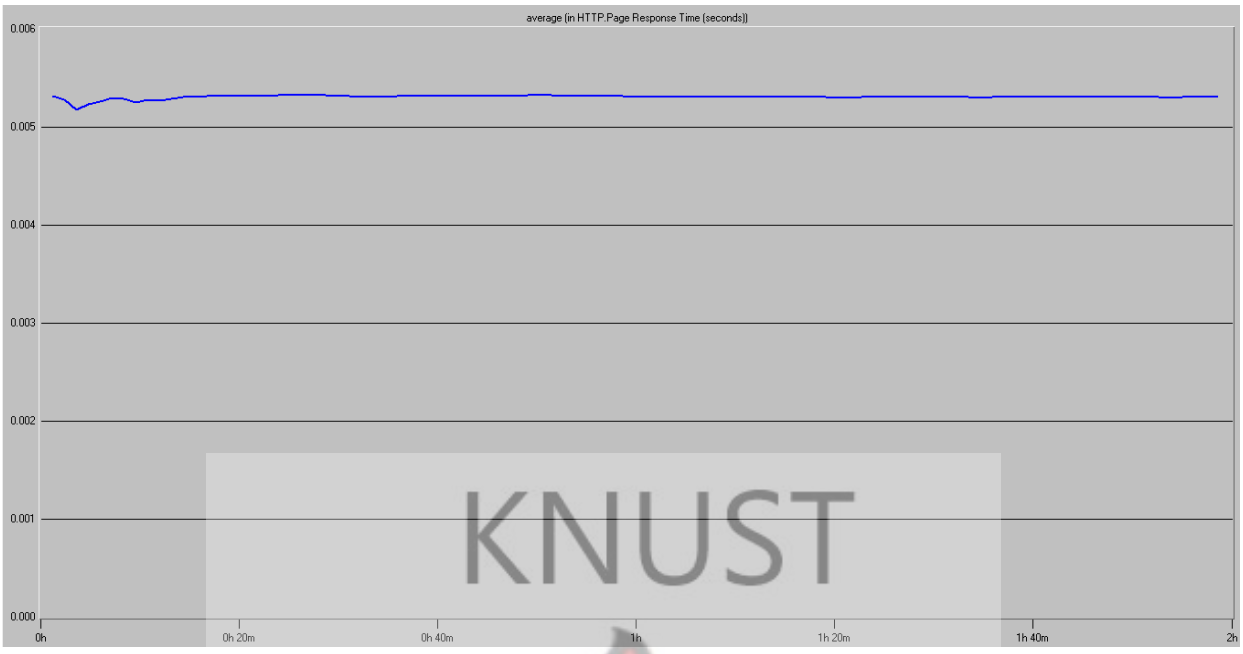

Fig.4.19: Http Page Response Time - Firewall Blocking

The figure above shows the page response time when other applications are blocked.

Combining the three scenarios produce a graph shown below.

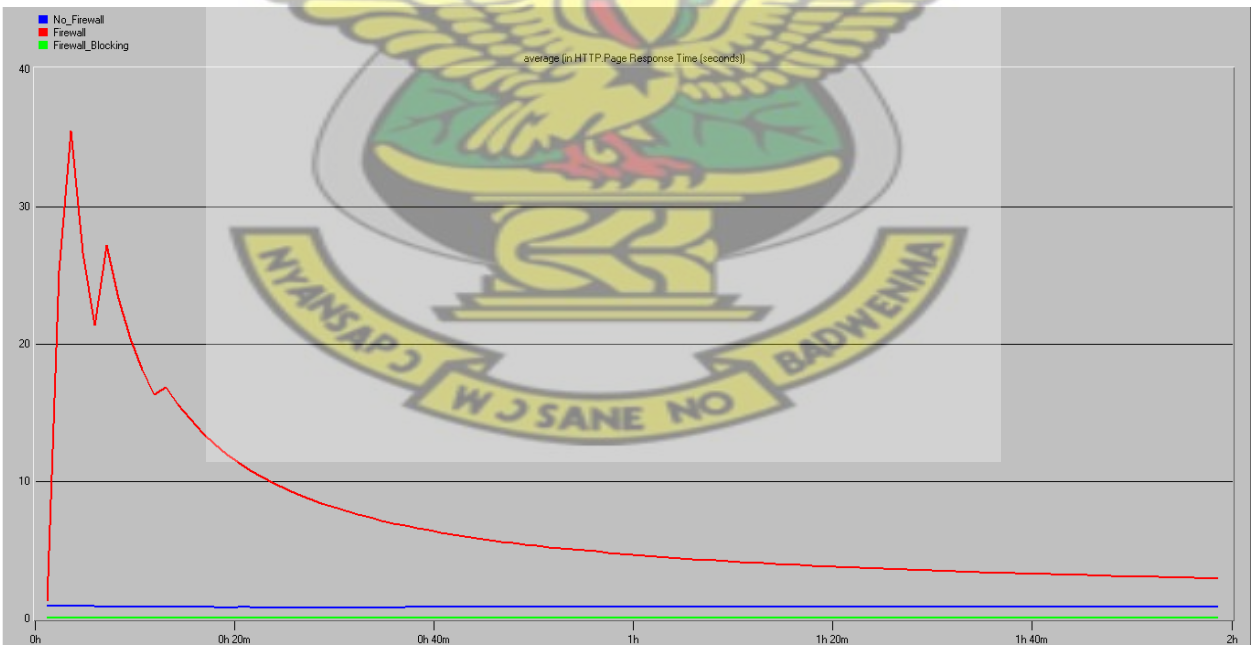

Fig.4.20: Http Page Response Time

From the above figure, it is observed that the page response time is more when firewall is imposed on the network. Due to the overhead of firewall filtering on the router, it introduces

extra processing which leads to an increase in page response time, i.e. degrade system performance. In the case where there was no firewall, the page response time is reduced. The third scenario blocks the web application from the network hence the low response time.

4.4.4 Server Http Load

The load on the web server is evaluated in this section under the three scenarios considered in the simulation exercise.

4.4.5 Server Http Load - No Firewall Scenario

The tables 3.19-3.21 show the server http load in request per second when no firewall is imposed on the network.

From the result, the load on the server is high. When no firewall is implemented on the network, there is no restriction to the flow of traffic across the router. As a result, a lot of request gets to the server which corresponds to the high load since the server needs to process a lot of the applications.

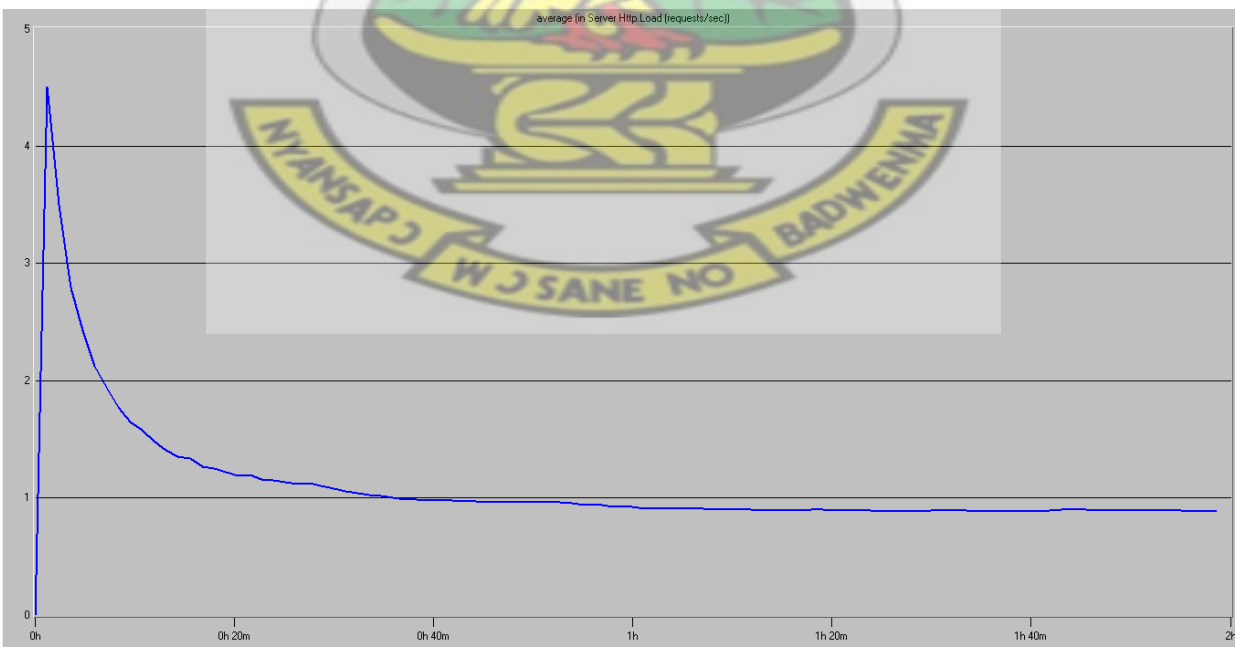

Fig.4.21: Http Server Load - No Firewall Scenario

The figure above shows the load when no security is imposes. As a result of the free flow of traffic, the server takes a lot time or busy processing user request.

4.4.6 Server Http Load - Firewall Scenario

The results when firewall is imposed are shown in the tables 3.19-3.21.

As a result of filtering some packet from getting to the web server by the firewall, it has a low value when compared to the no firewall scenario. The firewall blocks some unwanted traffic from traversing the network, so only legitimate packets get to the server, hence the load on the server decreases.

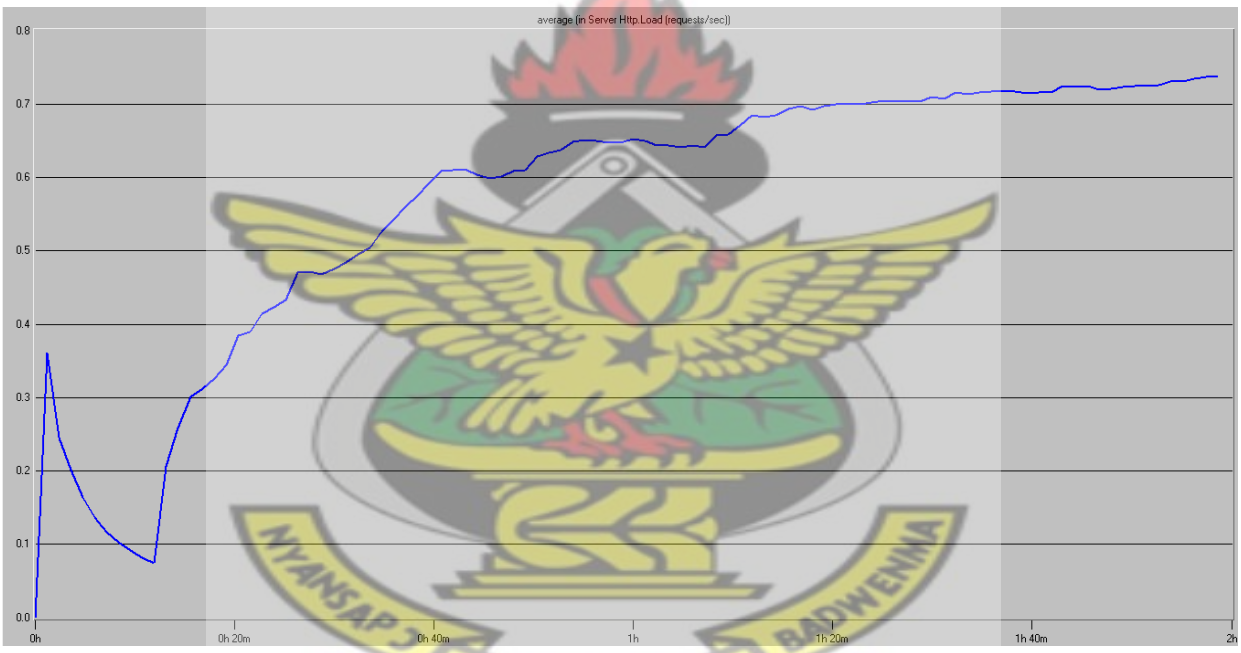

Fig.4.22 :Http Server load - Firewall Scenario

From the graph above, the load on the server is fluating. It has a value of 0.3613 seconds at the start of the simulation. It then drops to 0.0760 seconds for the first 20 minutes of the simulation time and again rise to 0.7389 seconds when the simulation runs for 2 hours. This is as a result of the filtering process that is taking place on the router. As the packets reach the

server, some are deny access and others are allowed access as they reach the server hence the fluating server load.

4.4.7 Server Http Load - Firewall Blocking Scenario

In the third scenario, the web application is blocked from accessing the server hence it has no value. Meaning there is no load on the server when the web is blocked. The result of the three scenarios are shown in the figure below

Fig.4.23: Http Server Load

From the figure, it is evident that the load on the web server is more when no firewall is imposed on the networked as compared to when a firewall is implemented. In the third scenario the web application is blocked hence the values 0 on the server load. In no firewall scenario, all packets that traverse the network are allowed through hence the server spends a lot of time processing the user request. This accounted for the high value in the web server load. In the case where the firewall is imposed, the firewall filters all the packets and only legitimate packets that conforms to the security policy of the organization is allowed through. It can be concluded that when firewall is imposed on the network it degrades network

performance since there is packet filtering taking place on the router. This overhead of the firewall impedes on the system performance.

4.5 Result for Ftp Application

This section discusses about the ftp application, which is one of the applications that generated traffic used in the simulation experiment. Ftp application is evaluated against download response time and upload response time. The load on the ftp server is also evaluated to investigate the performance of the network under the three different scenarios.

4.5.1 Ftp Download Response Time – No firewall Scenario

The results in the tables 3.22-3.24 show the ftp download response time when no firewall is imposed on the network

The result shows a low download response time when no security is imposed on the network. Since there are no restrictions to the flow of traffic across the network, it leads to a faster response time.

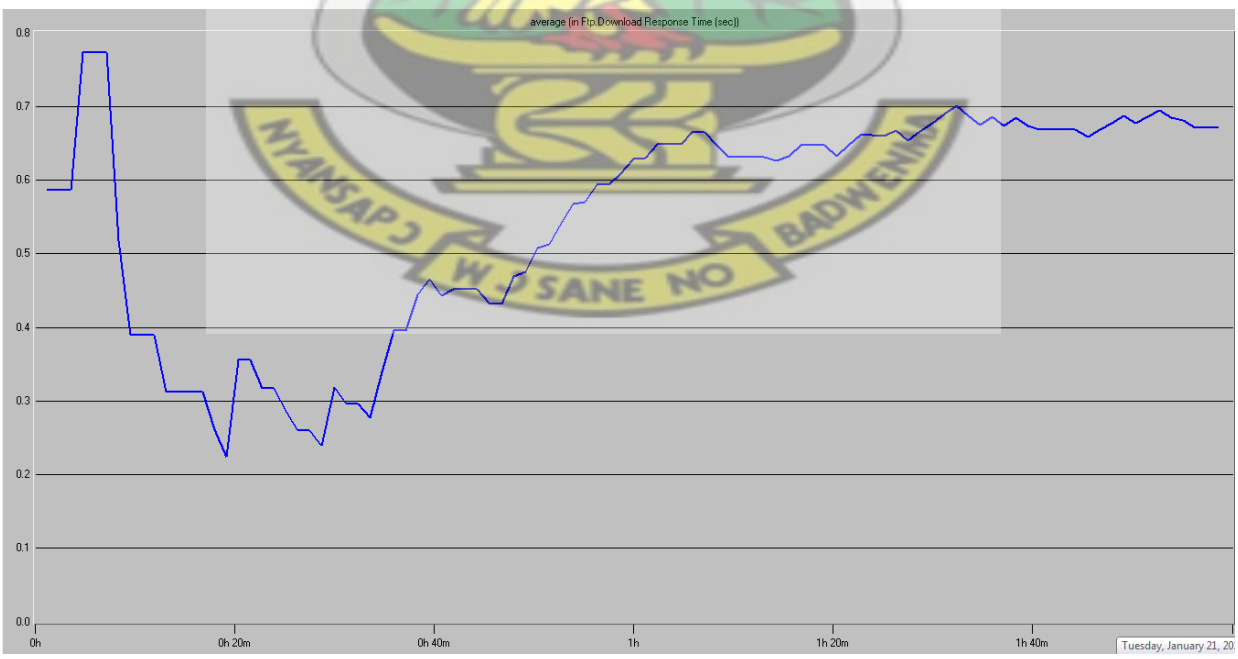

Fig.4.24: Ftp Download Response Time – No Firewall Scenario

From the figure above, it can be observed that the ftp download response time is not constant but varies along the simulations. It has a initial value of 0.5869 seconds and the drops to 0.2422 seconds and up to 0.6953 seconds.

4.5.2 Ftp Download Response Time - Firewall Scenario

The result of the download response time when firewall is implemented is shown in the tables 3.22-3.24.

When firewall is implemented, the download response time has a high value of 28.23 seconds when a packet size of 200MB is considered. The higher value is as result of the overhead encounter by the firewall when processing the request and also the packet latency of 0.05 imposed to induce some delay into the system.

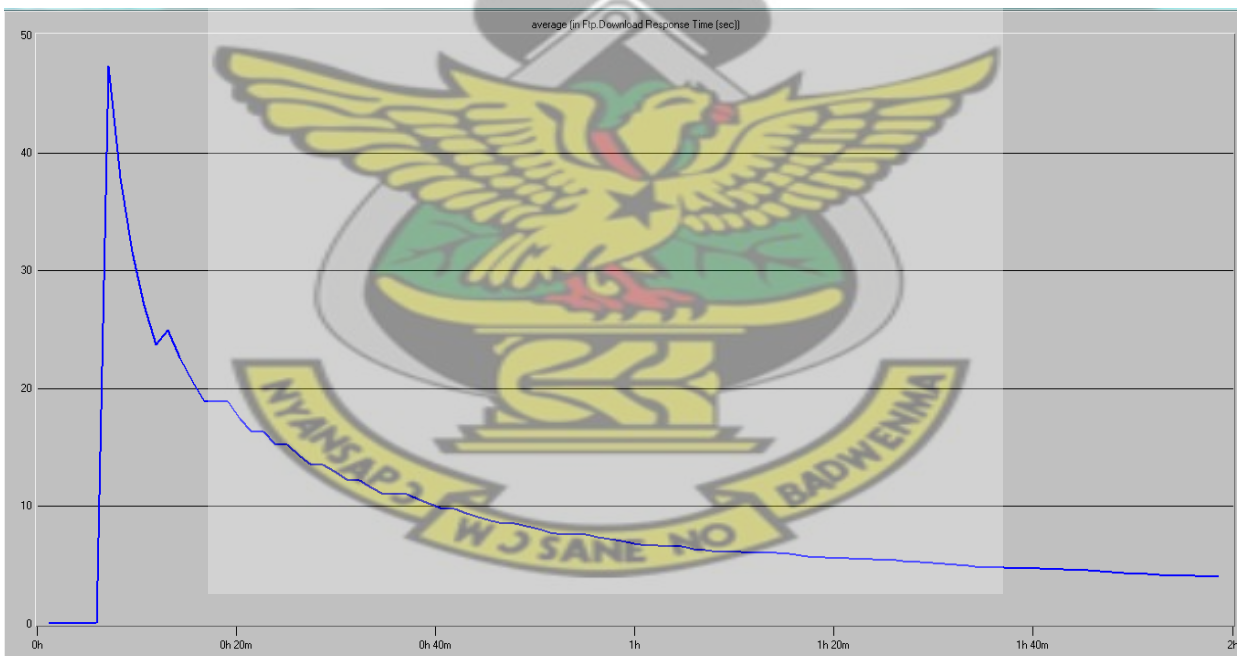

Fig.4.25: Ftp Download Response Time – Firewall Scenario

A firewall is a piece of hardware or software that is capable of filtering network traffic. This is generally performed strictly based upon the origin and/or destination of the data packets. A packet is a container used to break up large messages into smaller more manageable

segments. Each packet contains a header and data. The header contains its origin address, destination address and other information about the packet itself. Firewalls go through a simple three step process to determine whether a packet should be accepted or rejected. The firewall first analyzes the packet header. It then uses this information to determine if the packet matches any open connections within the state filter. Finally, if it does not match any state, a predetermined rule set is used to determine the action that should be taken. All these decisions by the router implementing the firewall take a considerable time on the processors hence the higher value.

4.5.3 Ftp Download Response Time – Firewall Blocking Scenario

In the third scenario, the ftp application is blocked from passing through the router. The results are shown in tables 3.22-3.24.

From the tables, it can be observed that the ftp download response time increases with an increase in the packet size. But the values are almost the same across the different data rates.

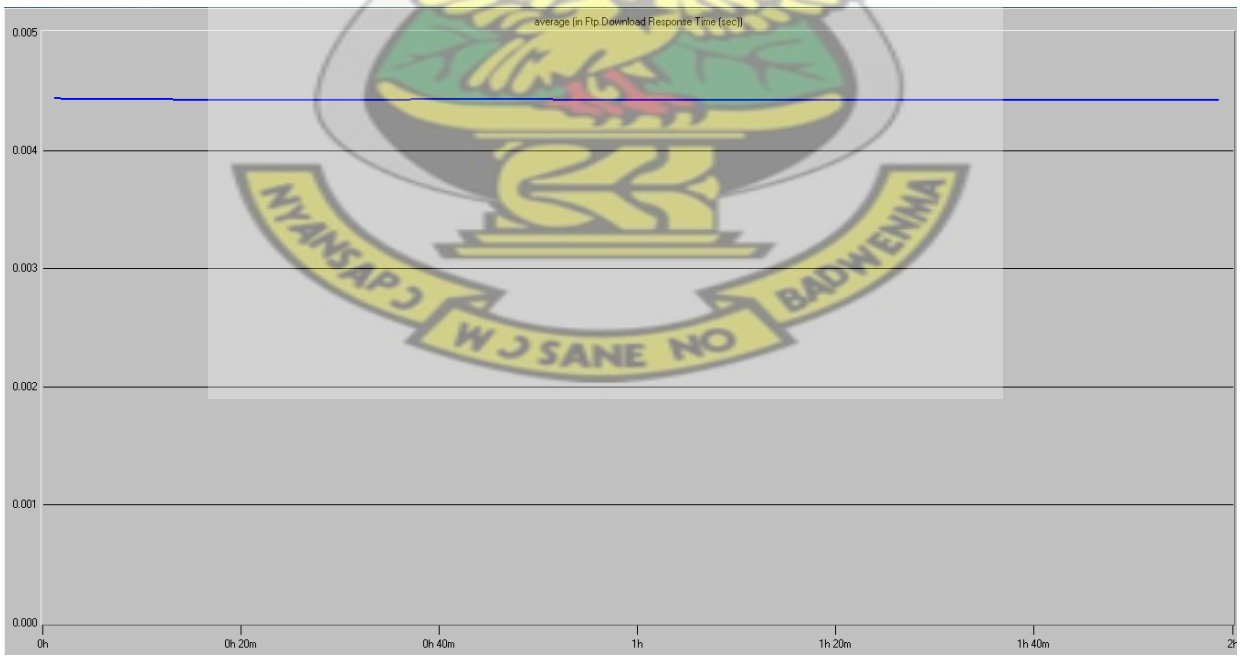

Fig.4.26: Ftp Download Response Time – Firewall Blocking Scenario

From the figure above, it can be observed that the response time is constant through out the simulation exercise. It has a value of 0.004 seconds across since the ftp application has been blocked access. When all the three scenarios are considered, the resultant graph is shown below,

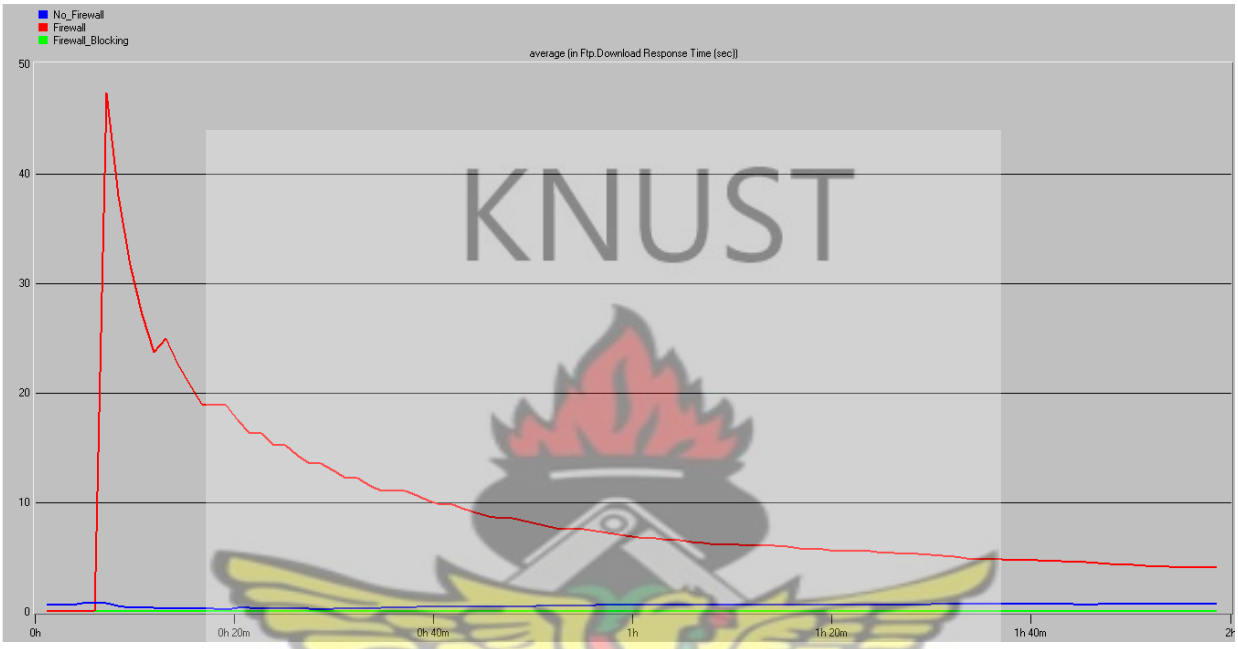

Fig.4.27: Ftp Download Response Time

From the figure above, it can be seen that, the download response time is high when security is imposed on the network as in the second scenario. It can also be observed that the ftp download response time increases with an increase in the data packet. In the case where no security is implemented on the network, and some applications are blocked, the download response time sees reduction. The reduction in the download response time is as a result of no restriction across the network.

4.5.4 Ftp Upload Response Time

The upload response time is discussed in this section

4.5.5 Ftp Upload Response Time – No Firewall Scenario

In the first scenario, no firewall is imposed on the network. Tables 3.25-3.27 show the ftp upload response time.

The results from the tables 3.25-3.27, show a low value of 0.02 seconds when 32MB of data was uploaded into the ftp server. From the tables, it can be observed that the upload response time increases with each increase in data packets but the values are almost constant through the simulation when different data rates are involved.

The figure below shows the graphical display when no firewall is implemented.

Fig.4.28: ftp upload response time – No firewall scenario

The low value as seen from the graph is as a result of the no restriction to the flow of traffic across the router.

4.5.6 Ftp Upload Response Time – Firewall Scenario

In the second case, a firewall is imposed on the network and a packet latency of 0.05 set so as to experience some delay in the distributed system. Tables 3.25-3.27 show the result when security has been imposed on the network.

The figure above shows a high value when security is imposed on the network. The upload response time is high meaning users experience some delay when uploading their files onto the ftp server. This high value degrades the system performance since users have to wait a considerable amount of time before their request been granted by the server.

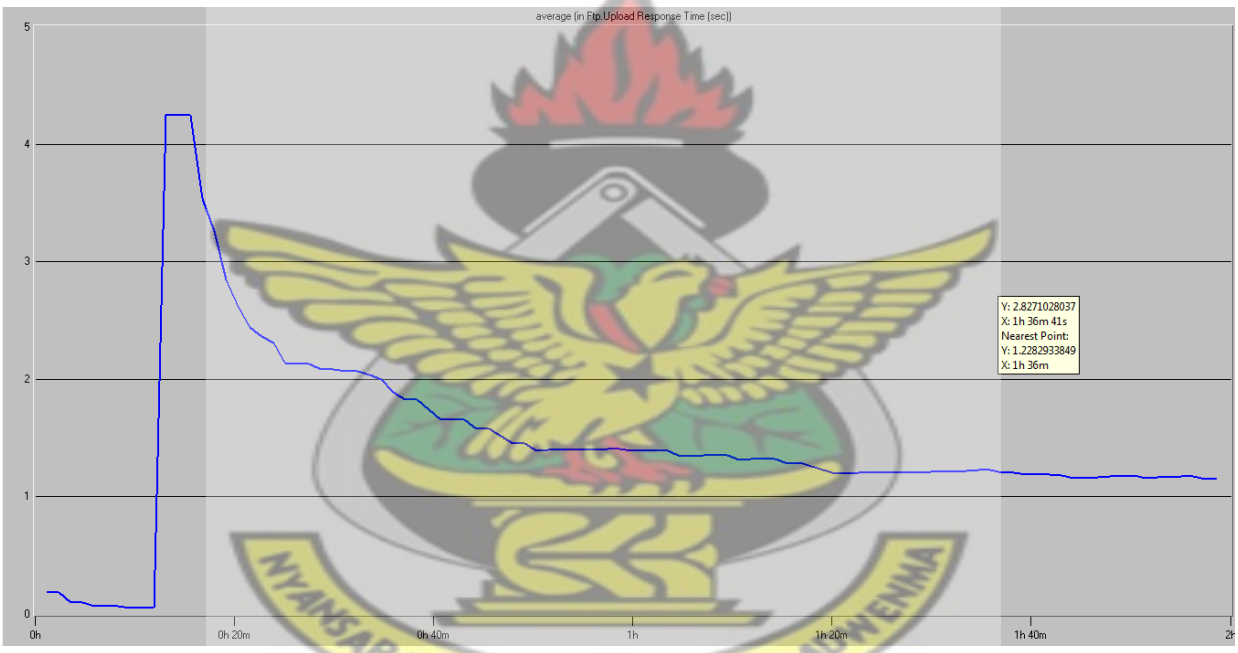

Fig.4.29: ftp upload response time – Firewall scenario

The high value in the upload response time when firewall is imposed is as a result of the extra processing being done by the router to filter out any illegitimate packets from accessing the router. Due to this overhead of the router, users get some delay in the system.

4.5.7 Ftp Upload Response Time – Firewall Blocking Scenario

In the third scenario, the ftp application is blocked at the router. Tables 3.25-3.27 show the results when other applications are blocked.

The upload response time is 0.23 seconds when trying to upload 200MB of data onto the server.

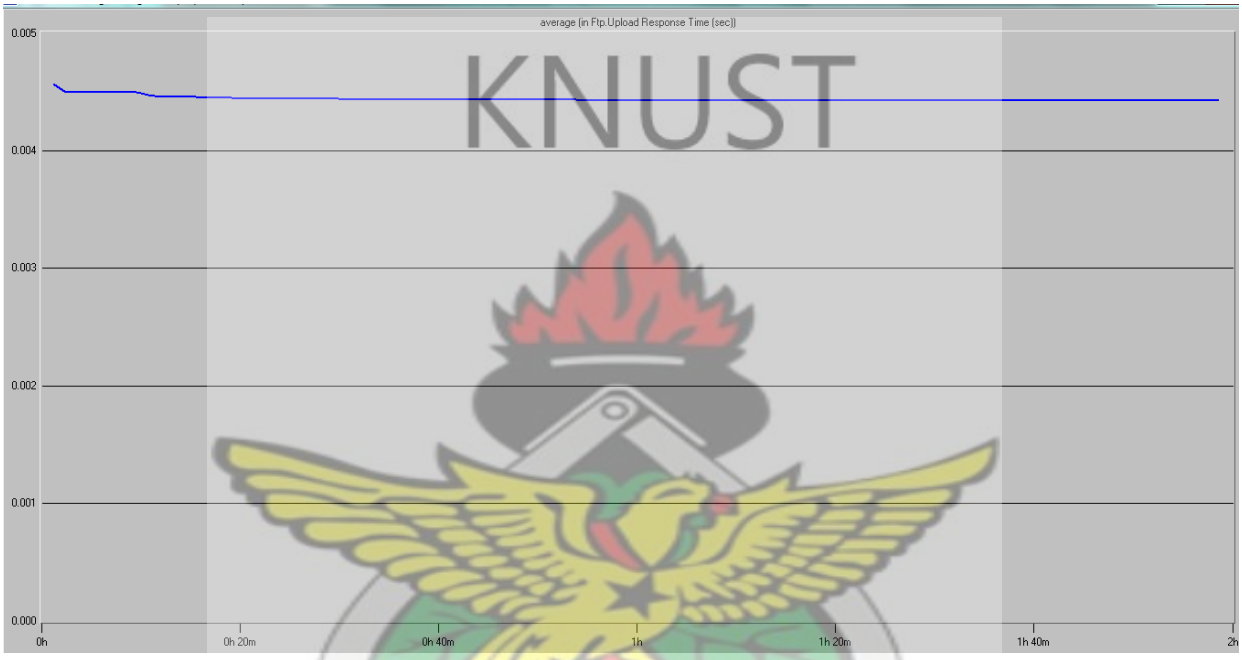

Fig.4.30: ftp upload response time – Firewall Blocking scenario

The figure above shows the graphical displace of the upload response time when the ftp application is blocked. Since the router takes some time to check its policy (filter) table before taking action on the packet, hence the value even though ftp is blocked. Taking the three scenarios into consideration produces the figure below

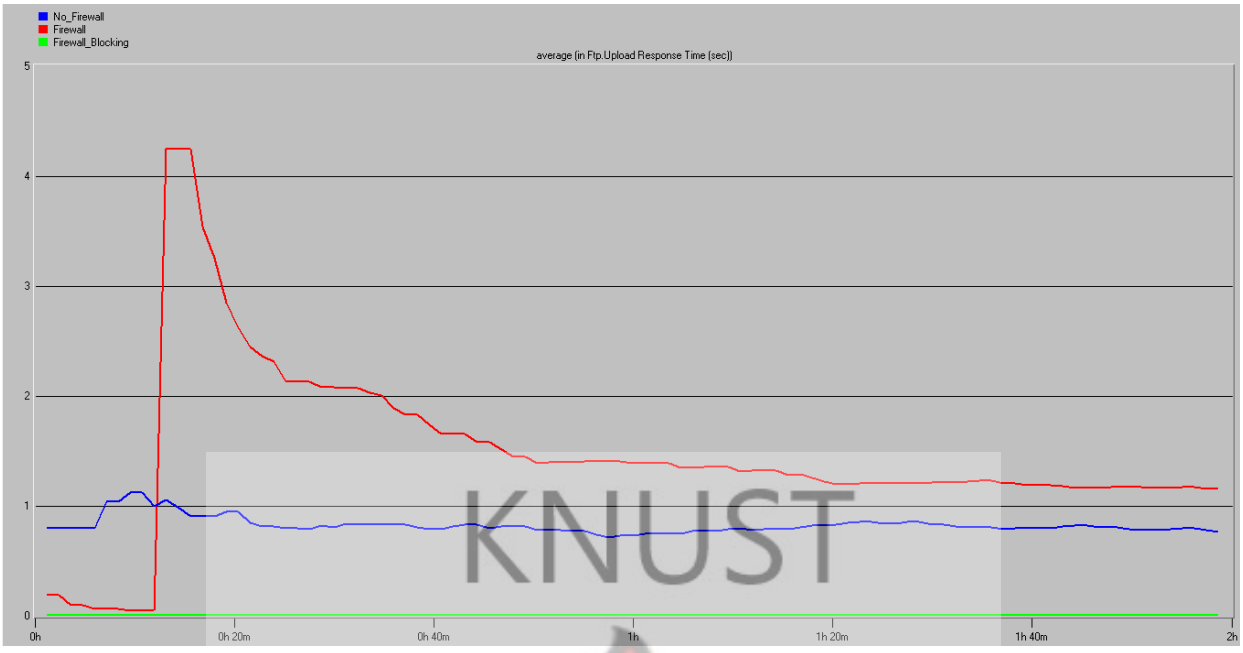

Fig.4.31: ftp upload response time

It is evident from the above figure that, when some applications are blocked from accessing the ftp server, it leads to a low response time. When no firewall is imposed sees a decrease in the upload response time. But the upload response time is very high in the case of firewall as a result of packet filtering occuring on the router.

4.5.8 Server Ftp Load

Tables 3.28-3.30 show the load on the ftp server in all the three scenarios.

From the tables, it is evident that the load is more when no firewall is imposed on the network. It has a high value of 0.14 seconds. In the second scenario where there is security implemented on the system, the load reduces to 0.001 seconds. But in the third case, since the ftp application is blocked, it has a value of 0. In the second scenario, a value of 0.001 seconds is the load on the server, meaning only small amount of packets gets to server for processing hence the server spend only 0.001seconds processing user request. The analysis is that imposing firewall on the network increases response time of user request. This unresponsive

nature of the system degrades the system performance since users request are not granted quickly.

4.6 Cloud Performance

This section discusses the cloud utilization. It is evaluated against the point to point utilization. Network utilization is the ratio of current network traffic to the maximum traffic that the port can handle. It indicates the bandwidth use in the network. While high network utilization indicates the network is busy, low network utilization indicates the network is idle. When network utilization exceeds the threshold under normal condition, it will cause low transmission speed, intermittence, and request delay

The tables 3.31-3.33 show the link utilization and graph below shows the link utilization across the three scenarios

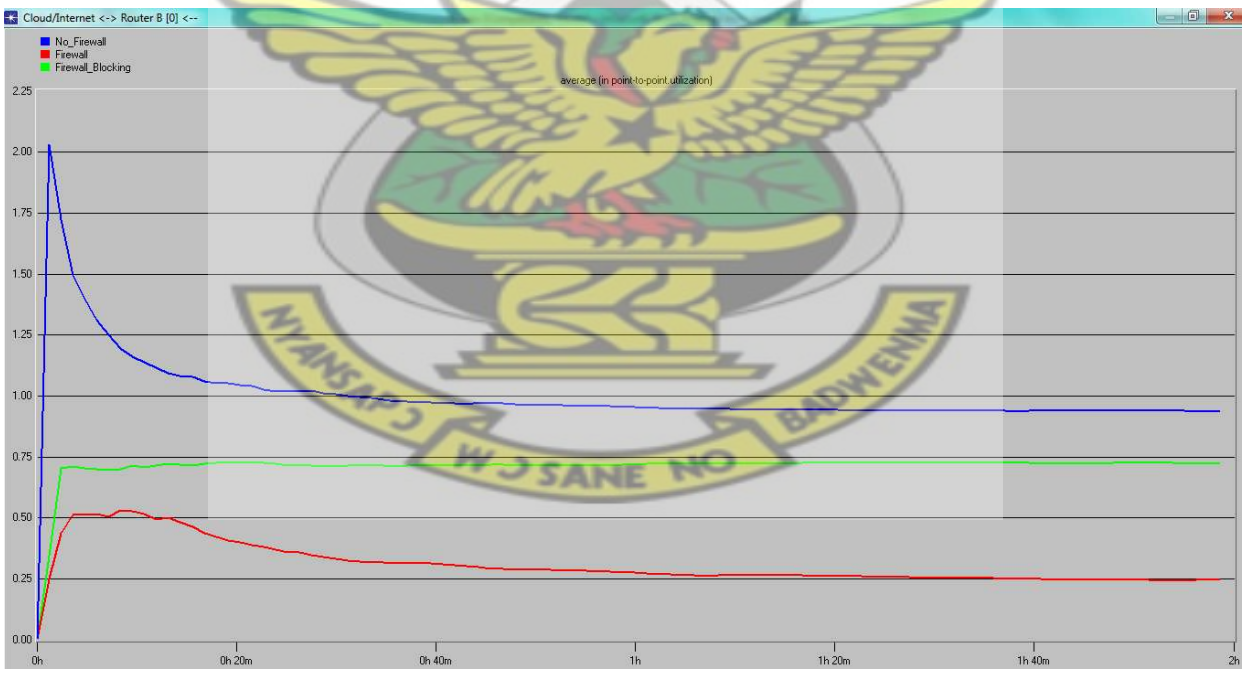

Fig.4.32: Cloud Point to Point utilization

It can be observed from the above graph and tables that the overall point to point utilization of cloud is more when there is no firewall across the network as the cloud needs to process

the database, email, ftp, and Http packets continuously. As the firewalls imposes some security policies and also delays the packets due to packet filtering, the cloud utilization is decreased. When the third scenario where the web, ftp and email traffics are blocked the overall utilization of the cloud is further reduced as shown in the above graph. As the traffics are blocked, the cloud has ample space to process the database packets and the overall utilization is reduced. Thus from the overall analysis it can be estimated that the overall utilization of the cloud can be optimized when firewall is imposed on the network.

CHAPTER 5

FINDING, CONCLUSION, AND RECOMMENDATION

5.1 Findings

The simulation experiment was used to measure the following:

i. database query response time

ii. http page response time

iii. email/ftp upload and download response time

iv. Point-to-point link utilizations.

Simulation results given in fig.4.1, fig.4.2 and fig.4.3, shows the database response to user requests under the three different scenarios. Response time is low in the first scenario and the third scenario. Introduction of a firewall increases response times, however, when other applications traffics are filtered; the database response time improves over no firewall scenario. The low response time corresponds to a higher network performance.

Fig.4.20 shows the http response time with, without a firewall and firewall blocking for the network. The performance of the two scenarios i.e. no firewall and firewall blocking are very close and have a low response time. There is high page response time showing some performance degradation when a firewall is in use.

Similarly, fig.4.13 and fig.4.27 shows the result for the ftp and e-mail applications. Again the download/uploads response time is very close and low for the ftp and e-mail applications when no firewall is imposed and some applications blocked. Also it was evident from the results that the chosen performance metrics increases with an increase in data size but almost the same with different data rates. This increases the network performance since users see a quick response to their request. The chosen performance metrics have a higher value when

firewall is imposed on the network. This means that when security is imposed on the network, the network performance degrades.

The general conclusion is that network security and network performance are inversely related, which implies that imposing more security on the network, correlates to decrease in the network performance.

5.2 Conclusion

The need for firewalls has led to their ubiquity. Nearly every organization connected to the Internet has installed some sort of firewall. The result of this is that most organizations have some level of protection against threats from the outside. This study has found out that network security and network performance are inversely related. As seen from the result of the simulation, network performance is adversely affected when firewall is implemented. There is performance degradation when security policies of the organization are implemented. Nevertheless firewalls do not only secure a network but also contribute to network performance by stopping attacks, improving network availability, and reducing unnecessary processing of illegitimate requests.

5.3 Recommendation

Based on the result of the study we recommend that organizations turning to implement security on their network should be prepared to experience a little decrease in network performance.

REFERENCES

Al-Shaer E. and Hamed, H. (2004) “Discovery of policy anomalies in distributed firewalls”, in Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol.4

Al-Shaer, E.S., and Hamed, H.H. (2004) “Modeling and Management of Firewall Policies”, [online]:Available:www.mnlab.cs.depaul.edu/projects/FPA/files/tnsm04.pdf [September 05]

Bellovin, S. and Cheswick, W. (1994) “Network firewalls”, IEEE Communications Magazine, vol.32, no.9, pp.50–57

Blakley, B., (1996) “The emperor’s old armor”, in Proceedings of the 1996 workshop on New security paradigms, ACM Press New York, USA, pp.2–16

Buchanan, W. (2011) Advanced Security and Network Forensics, Napier University.

Caldwell, D., Gilbert, A., Gottlieb, J., Greenberg, A., Hjalmtysson, G., and Rexford, J. (2003) “The cutting edge of ip router configuration”, in Proceedings of 2nd ACM Workshop on Hot Topics in Networks Hotnets-II.

Corbitt, T. (2002) “Protect your computer system with a security policy,” Management Services, vol. 46(5), pp.20–21,[Online].Available:http://findarticles.Com/p/articles/mi_qa5428 /is_200205/ ai_n21313131/ pg_2? Tag =artBody;col1 [2002.]

Cuppens, F., Cuppens-Boulahia, N., Sans , T., and Miege, A. (2004) “A formal approach to specify and deploy a network security policy”, in Formal Aspects in Security and Trust, ser. IFIP International Federation for Information Processing, vol. 173/2005, Springer Boston, p.203.

Danchev, D., (2003) “Building and implementing a successful information security policy,” [online] Available: http:// www.windowsecurity.com [2003]

Deal, R. A. (2004) Cisco Router Firewall Security, Cisco Press,

Eddy, W. (2007) TCP SYN Flooding Attacks and Common Mitigations, IETF RFC 4987

Ehlert, S., Zhang, G., and Magedanz, T. (2008) "increasing sip firewall performance by ruleset size limitation", in PIMRC, pp 1-6

Fraser, B., Aronson, J. P., Brownlee, N., and Byrum, F. (1997) "Site security handbook (rfc2196)," [Online] Available: http://www.ietf.org/rfc/rfc2196.txt? Number =2196 [Sep 1997]

FreeBSD, F FreeBSD handbook 2010 FreeBSD Document Project 2010

Georgieva, T. (2009) "Types of Denial of Service (DoS) Attacks, Suite 101 Webmaster Resources", [Online] Available:http://www.suite101.com/content/types-of-denial-of-service-dos-attacks- a14301 [11, July 2011]

Hamed, H., and Al-Shaer, E. (2006) "Dynamic rule-ordering optimization for high speed firewall filtering" in the proceedings of the 2006 ACM symposium on information, computer and communication security, ACM press pp.332342

Hunt, R., and Verwoerd, T.(2003) "reactive firewalls - a new technique" Computer communications, Elsevier, UK. Vol. 26, No 12

Hwang, J. H., Tao X., Chen, F., and Liu, A. X. (2011) "Systematic Structural Testing of Firewall Policies"

Ioannidis, S., Keromytis, A., Bellovin, S., and J. Smith. (2000) "Implementing a distributed firewall", in Proceedings of the 7th ACM conference on Computer and communications security, New York, USA, pp.190–199 Journal of Information Security, vol.5, no.3, pp.125–144

Lodin, S. W. and Schuba, C. L. (1998) "Firewalls fend off invasions from the net", IEEE Spectrum, vol. 35, no. 2, pp. 26–34.

Lyon,G. (2011) Nmap Reference Guide. [Online] Available: http://nmap.org/book/man.html [11, July 2011]

Lyu, M. R., and Lau, L. K. Y. (2003) Firewall Security: Policies, Testing and Performance Evaluation, [online] Available: http://www.cse.cuhk.edu.hk/~Lyu/Paper _Pdf/kylau .pdf

Madigan, E. M., Petrulich, C., and Motuk, K. (2004) "The cost of non-compliance: when policies fail'' in SIGUCCS 04 in proceeding of the 32nd annual ACM SIGUCCS

Marmorstein, R. (2005) "a tool for automated iptable firewall analysis", in freenix Track, USENIX Annual Technical Conference, pp. 71-82.

Mayer, A., Wool, A., and Ziskind, E. (2006) "Offline firewall analysis", International Journal of Information Security vol.5 no.3 pp. 125-144

Nazario, J. (2008) Bot and Botnet Taxonomy, CSIS Security Exchange.

NIST, (1995) "An Introduction to Computer Security", the NIST Handbook National Institute of Standards and Technology, U.S. Department of Commerce

NIST, (2002) "Guidelines on Firewalls and Firewall Policy", [Online] Available: http://csrc.nist.gov/publications/nistpubs/800-41/sp800-41.pdf SP800-41 [October 5, 2006]

Ogletree, T. W. (2000) Practical Firewalls – Question; 1st edition, policies fail", in SIGUCCS '04: Proceedings of the 32nd annual ACM SIGUCCS conference on Network Firewalls: India

Richard. J. Macfarlane (2009) "An Integrated Firewall Policy Validation Tool"

Reddy A.V. (2012) Usage of Opnet IT tool to simulate and test the security of cloud under varying firewall conditions, Texas USA

Rubin, A. D., Geer, D., and Ranum, M. J. (1997) Web Security Sourcebook. Wiley

Samarati, P. and de Vimercati, S. C. (2000) "Access control: Policies, models, and mechanisms", Lecture Notes in Computer Science, vol.2171, pp.137–196.

Scarfone, K., and Hofman, P. (2009) “Guidelines on Firewalls and Firewall Policy”, Recommendations of the National Institute of Standards and Technology

Schneider, F. B. (2000) “Enforceable security policies”, ACM Transactions on Information and System Security (TISSEC), vol.3, no.1, pp.30–50

Sheth, C., and Thakker, R. (2011) Performance Evaluation and Comparative Analysis of Network Firewalls

Shimonski R.J, Shinder D.L, D.S.T. (2003) best dawn firewall book period Syngress

Shirey, R. (2007) Internet Security Glossary Version 2, IETF RFC 4949

Simmonds, A., Sandilands, P; Van Ekert, L. (2004) “An Ontology for Network Security Attacks”, Lecture Notes in Computer Science user services New York, NY, USA: ACM, pp.47–51.vol.37, no.6, pp.62–67

Wagh, A. (2009) “Purpose of Scanning the Network: Stealth Attacks. Hackers Enigma, April 2009”, [Online] Available:http://www.hackersenigma.com/network-security/purpose-of-scanning-the- network-stealth-attacks-2/ [11, July 2011]

Whitman, M., and Mattord, H. (2005) Principles of Information Security: Thomson Boston, Massachusetts

Wong, T. (2008) “On the usability of firewall configuration”, in Symposium on Usable Privacy and Security

Wool, A. (2004) “A quantitative study of firewall configuration errors”, Computer society,

Wool, A. (2006) Packet Filtering and Stateful Firewalls Wiley, Firewall Architectures,p.526