UNIVERSITY OF EDUCATION, WINNEBA

NETWORK TRAFFIC ANALYSIS USING WIRESHARK IN SOLVING NETWORK PROBLEMS

SAMUEL BEKOE

A dissertation in the Department of Information Technology Education,
Faculty of Technical Education, submitted to the School of
Graduate Studies in partial fulfillment
of the requirements for the award of the degree of
Master of Science
(Information Technology Education)
in the University of Education, Winneba

MAY, 2020

DECLARATION

STUDENT'S DECLARATION

I, **SAMUEL BEKOE,** declare that this Dissertation with the exception of quotation and references contained in published works which have all been identified and duly acknowledged, is entirely my own original work, and it has not been submitted, either in part or whole, for another degree elsewhere.

SIGNATURE:…………………………………..

DATE:…………………………………………..

SUPERVISOR'S DECLARATION

I hereby declare that the preparation and presentation of this work was supervised in accordance with the guidelines for supervision of Dissertation as laid down by the University of Education, Winneba.

DR. FRANCIS OHENE BOATENG

SIGNATURE:…………………………………..

DATE:…………………………………………..

DEDICATION

I dedicate this work to my late mother Beatrice Nyamaa.

ACKNOWLEDGEMENT

Glory and honors to the Almighty God for how far He has brought me in my academic journey. I say Lord, I am most grateful. I wish to express my heartfelt appreciation to my supervisor Dr. Francis Ohene Boateng for taking time off his tight schedules to guide and shape the writing of this work. My family deserves special commendation particularly my late mother Beatrice Nyamaa. My warmest gratitude also goes to my friends James Tawiah Sackey, Prince Awuah Boadu and my boss Dr. Samuel Adu Gyamfi who has been my inspirer and motivator.

May the good Lord bless you all. Amen

TABLE OF CONTENTS

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATIONS

LIST OF TABLES

TABLE **PAGE**

LIST OF FIGURES

FIGURE **PAGE**

LIST OF ABBREVIATIONS

ARP:	Address Resolution Protocol
CAT:	Category
LAN:	Local Area Network
PC:	Personal Computer
SMTP:	Simple Mail Transfer Protocol
CD:	Compact Disc
DNS:	Domain Name Service
FTP:	File Transfer Protocol
GHz:	Giga Hertz
GPS:	Global Positioning System
TCP:	Transmission Control Protocol
UDP:	User Datagram Protocol
USB:	Universal Serial Bus
WAN:	Wide Area Network
HTTP:	Hypertext Transfer Protocol
ICMP:	Internet Control Message Protocol
IP:	Internet Protocol
LAN:	Local Area Network

ABSTRACT

Network traffic analysis is an arrangement of strategies that are progressively used to evaluate the transfer of packets across a network. This movement of information encompasses the period and length of the transferred packets and the state of the correspondence packets. The analysis of system activity gives data about the client as well as the destination of packets thereby enabling system administrators get information to help the network. Analysis of network activity is a significant undertaking resulting in valuable contributions both to the commercial sector and the research community. The study explores the use of networking tools as well as models to ensure network efficiency. To be able to capture data traffic, Wireshark application was installed and run to capture packets of data passing through the network interface card. Other tools like Microsoft Windows 7/Win Server 2008, Microsoft Office 2007, Mozilla Firefox and Wincap was used to support the study. The study revealed that when packets are captured using the Wireshark, the application indicates the date the activity was carried out on the subnet. Those packets traverses therefore if the source is known an inference can be made as whether at that time the use was supposed to work or not so that date and timing can be monitored. The study found that by virtue of knowing the source and destination of the IP address, administrators can easily identify the user of that particular personal computer. The study recommends that in order to maintain undisturbed links in the network, activities of users of the network must be monitored to find out what they do with their personal computers that is connected to the organization's network.

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

The usage of computer networks rises as more nodes are connected which force organizations to spend huge sums of money to ensure their network is efficient, speed and effective to be able to carry out their daily business activities (Mashitah, 2003). Every organization can now boast of many systems connected, nodes such as scanners, computers, servers, printers and many others in its premises for the smooth running of their businesses (Brownlee, 2002). Computer networks and Internet have become the very big thing that every company is trying to adapt to its business environment for the sake of transacting business right from their business premises to the outside world (Mashitah, 2003). However, due to the numerous activities of these organizations as well as intruders outside the organization, computer networks as well as internet connectivity of these organizations continue to run very slow (Mashitah, 2003) (Song, Beshley, Przystupa, & Beshley, 2020). Because of that, network administrators who have been hired by these organizations to manage the network systems have the arduous task of making sure that the network systems are up and running each and every passing day for smooth operation of the organizational business activities.

The increasingly slow paces of computer networks have given rise to the need for constant monitoring and analysis by administrators (Brownlee, 2002). Through innovation, systems have been changed from association between couple of PCs to interfacing number of PCs in numerous systems, depicting the term Internet. This network of systems would demand good administration (Morris, 1994). The Internet is a network of network where resources found on the network can be shared, like the trading of records on the World Wide Web (WWW) and the structure to bolster messages (Kapri, 2011). The idea to hypothesize network movement in an all-encompassing system is an aspect some system exploration techniques examine (Kapri, 2011). The Internet has undergone several studies and has been seen to handle a wide range of challenges, which includes attacks and security concerns (Kapri, 2011). Network analysis is a method of monitoring network traffic and examining it closely to regulate what is occurring in the network (Colasoft, 2018). It is also identified by other names: packet sniffing, protocol analysis, network analysis and packet analysis to name a few.

Network traffic analysis is an arrangement of strategies that are progressively used to evaluate the transfer of packets across a network (Sciancalepore et al., 2019). This movement of information encompasses the period and length of the transferred packets and the state of the correspondence packets. The examination of system activity gives data about the client as well as the destination of packets thereby enabling system administrators get information to help the network (Colasoft, 2018). Analysis of network activity is a significant undertaking resulting in valuable contributions both to the commercial sector and the research community (Susila, 2020). For example, researchers and Internet Service Providers (ISPs) are continuously looking for ways to enhance the network resources and undertake efficient traffic engineering (Colasoft, 2018). Traffic analysis not only delivers ways of recognizing a culpability in the network, but also aids in understanding the fundamental source of the fault and its influence on communication between the people who use the network (Ohlsson & Wernersson, 2014).

Most of the root cause of poor performance of network systems can be attributed to cable cuts, poor termination, hardware malfunction, software errors, unnecessary downloads, interferences from both local and remote links, nodes and human errors such as configuration mismatch etc (Wu, Leshem, Member, & Jensen, 2015). However, most of the causes of poor performance indicators are outside the control of network or Internet providers and therefore the need for non-router based monitoring technique (passive and active) to monitor the effects of transmission delay, bandwidth consumption, and congestion emanating from user's activities from both local and remote source (Sivanathan, 2019). This study attempts to analyze the reason for packet delays, bandwidth consumption, and congestion as applied to packets parameters and also source and destination of IP address as well as packet length. Sally and Vern (2001) depict the primary challenges individuals face when attempting to recreate the Internet and learn its properties. In any case, specialists share their insight into how to seek after these uncertainties (Paxson, 2004) (Wei, Member, Dragotti, & Member, 2016). In this study, the specialist seeks after the examination of moving packets, gathered each day, aimed at 15 minutes for a time of 2 months from UEW Local Area Network. With this dataset, there are a lot of alternatives to consider when it is examined both at TCP/IP layers (bundle and stream characteristics) and software utilizations (Muhtadi, Almaarif, & Info, 2020).

1.2 Statement of the Problem

One of the major problems in network research in terms of monitoring packets is non-router based approach and hence monitoring user's activities on the internet using network packet analysis and reporting individual user's technical problems (Sikos, 2020). One can only get access to the information on hosts through the node that manages it. It is done by specifying the desired attribute or names. For any system, the monitoring techniques depend on the needs of the application. Most of the time, administrators might not know the problems whenever they are called for technical assistance. The phone calls and the search for experts waste precious time. In addition, enterprise monitoring applications are normally deployed at strategic points across the networks with traffic in mind, and monitoring of the packets are also done. This may render certain hosts unseen to the monitoring software (Song et al., 2020). Thus, they might bring additional packet which is not needed at the time of use by their organization leading to the introduction of traffic on the network. In networks, traffic is the common problem that is mostly experienced by end users.

As the network expands, traffic becomes unbearable, thus possibilities of the network congestion are very high. The rapidly growing number of users and applications have led to the slow running of applications and complaints, and more importantly activities of both LAN/WAN users in organizations. The problems which arise are additional traffic during peak hours of work in the organization and how packet analysis in passive monitoring can be used to visually monitor activities of users with respect to delay, bandwidth consumption and congestion (Anil & Shina, 2020). We are looking at an application where clients could also report their own problems by just a click of a button in the network and its activities monitored (Muktar & Kyauta, 2017). Now, network monitoring problems in LAN can only be detected by professionals who sometimes cost organizations in terms of time to attend to LAN monitoring problems which are often trivial. This helps to reduce the time between failures and the time of restoration. For this reason, organizations are making plans to have strong network systems as a solution to their challenge. With systems designed for attaining this aim, administrators tasked with the networks of the organization need to perform the analysis of the network system from time to time.

Per the researchers' findings, University of Education Winneba Kumasi UEW-K uses a campus network. The network situation at UEW-K is quite disturbing. Although the network at UEW-K is used only by students, lecturers, administrators and other workers at UEW-K, the network is not strong enough. Students, lecturers and administrators of UEW-K sometimes complain about the strength of the campus network. Sometimes, in the cause of the day, when administrators of UEW-K want to use the network for administrative purposes it run slowly which cost them time. Lecturers sometimes face network problem when they want to use it for research. According to the researcher's observation, network problem at UEW-K does not favour students at all since they are assigned to a particular network in which all of them rely on the same network, it makes the network run slowly. This disturb students a lot in their academics. Thus, the study sought to analyze the network Traffic at UEW-K using Wireshark in solving network problems. Furthermore, the study investigated the use of Wireshark to monitor packets transfer between hosts and the effects of user activities characterized by bandwidth consumption, delay, and congestion on the packets parameters within the network/ internet.

1.3 Objectives of the Study

This study explores the use of networking tools as well as models to ensure network efficiency. The specific objectives are;

1. To use Wireshark to monitor packets transferred between hosts.

2. To verify network interface card and driver status without introducing additional traffic.
3. To effectively communicate monitoring results to clients and resolve their problems through the network.
4. Analyze the effects of user activities characterized by bandwidth consumption, delay, and congestion on the packets parameters within the network/internet.

1.4 Significance of the Study

The contribution is using IP address numbers to monitor the activities of users based on packet movement in the network. Network systems have a growth with businesses in the world today. Computer network engineers are responsible for the design, installation and management of Networks. For security and resource monitoring reasons, networks need to be monitored against threats and traffic which affect the business operations and others (Anil & Shina, 2020). The deployed work will make it possible to detect hosts communicating with each other via the Internet and in that way, administrators can easily track user's packets movement if they are not in line with the organizational objectives (Ohlsson & Wernersson, 2014). In addition, the effects of delay and congestion are analyzed using graphs. Another area in which this examination works is to obtain a system observing framework which will have the capacity to ensure the security of the whole system and know about any vulnerability. Other than the security, every activity which has been completed by the computer unit associated with the system, would be checked to give a fair idea about the usage of the network resource. It is recommended that the study would be good for network administrators and organizations that might want to know the network flow, in and out of the system and provide accountability for the network use and also provide optimization for better network performance.

1.5 Scope of the Study

The study looks at the following applications, which consist of the design of the various component involved. The application is expected to capture data for analysis. The main Stakeholders involved in the study are; IT administrators, IT technicians, Local Area Network and internet users as well as network engineers. The study will be conducted on the Local Area Network of the University of Education Winneba, Kumasi Campus.

1.6 Organization of the Study

This study is organized into five chapters. The preceding chapters provide an insight to the succeeding chapters. The First chapter presents the background philosophies of network observation using packet analysis and our research objectives as well as the significance of study. In Chapter two, the study reviewed literature and fundamental concepts related to the work and subjects relating it. The reasons why we selected the methodologies for the work are discussed, as well as related works. Chapter three covered the methodology of how the proposed work was planned from the requirements analysis. In addition, in-depth logical methods including UML diagram modeling was made. Chapter four dealt with packet analysis which comes from the data obtained from the captured packets through developed application. Finally, Chapter four has the implementation of the proposed solution. Conclusions were made grounded on the findings in the analysis and designed application. Chapter five served as the conclusion of the study and recommendation. The study revisited the research contributions and objectives with regards to the methods in Chapter three and the outcomes in Chapter four. Finally, a discussion and recommendations for future work with regards to the work is presented.

CHAPTER TWO

REVIEW OF RELATED LITERATURE

2.1 Introduction

This chapter reviews related literature of the framework that is being used as reference or guideline to the research. This chapter constitutes the introduction, Concept of TCP/IP Protocols, Data mining, Network Packet Capture, Capturing Packets, Network traffic data inspection techniques, and Remote Access.

It further reviews some information related to the study to get better understanding such as packet analysis, network traffic analysis and network monitoring as well as various network traffic analysis techniques (Muhtadi et al., 2020). Again, the researcher finds it proper to use Wireshark application to carry out effective network monitoring and analysis on the current Local Area Network of UEW-K. This has become necessary because of the increased number of users on the system, router and switch configuration, malicious attacks, and the overall architecture arrangement of the network. The research finds two key focuses needed for the execution of putty for monitoring and analysis; Network monitoring and Remote Access. Different ideas which have been useful in thoughtful innovations as a part of the usage of the product have been completely laid out in this section. Appropriate observation and examination of the system framework is constantly expected to discover shortcomings and disengagements which influence the execution in the system of any association (Slowman & Jonathan, 1994). Measures must be put in place as to how to monitor the whole system's execution and keep up any conceivable issues. In an establishment like a school, a critical look is noteworthy as in this case even a short postponement can drive the undertaking to disappointment in accomplishing its objectives. The emphasis of the network monitoring and analysis must be on the interior and exterior performances and events of the workstations linked to the network. Some exterior activities are linked with the events taking place in the network as well as connection of workstations to the network. Internal behaviours can be declared as the events of every workstation and the exchanges amongst them. In the internal behavior study, the administrator must monitor all the nodes that remain connected to the system. However, as the exterior performance of the network governs the behavior of the network, the emphasis for many set-up monitoring systems is on the exterior performances (Muktar & Kyauta, 2017).

2.2 Concept of TCP/IP Protocols

The Internet is controlled by two key protocols namely the Internet Protocol (IP) and the Transmission Control Protocol (TCP). The Internet convention which is mainly known to be IP collection not just includes layer conventions, (for instance, IP and TCP), yet it likewise determines the platforms used for normal applications to run, like, e-mail, document exchange and terminal imitating (Miller P. M., 2010). They can be used in many forms of interconnected systems and are similarly appropriate for LAN and WAN networks. In this manner, the work utilizes the TCP/IP as the basic infrastructure for communication. Internet Protocol (IP) is a system layer (layer 3) convention that is responsible for sending packets over the Internet. Alongside Transmission Control Protocol (TCP), IP speaks to the heart of the Internet convention (Fall & Stevens, 2011).

It gives connectionless mode of communication and conveys datagram between systems; and giving discontinuity and reassembly of datagram to bolster information joins with various extreme broadcast components (Hartpence, 2011). Each layer of the

TCP/IP protocol has its own function with regards to how data is transferred. The system interface layer, known as datalink layer, incorporates the administration of the gadget driver and the system interface card inside the host. This layer likewise gives an interface to the equipment that is physically connected to a host. The system layer oversees dealing with the vehicle of packets from a source host over the system to a goal host; using the address. The transport layer oversees the stream of information between two hosts in support of the application layer. The application layer provides an interface to share packets or data. The TCP/IP convention suite comprises of different protocols which performs different capacities. Figure 1 demonstrates the protocols utilized as a part of this review.

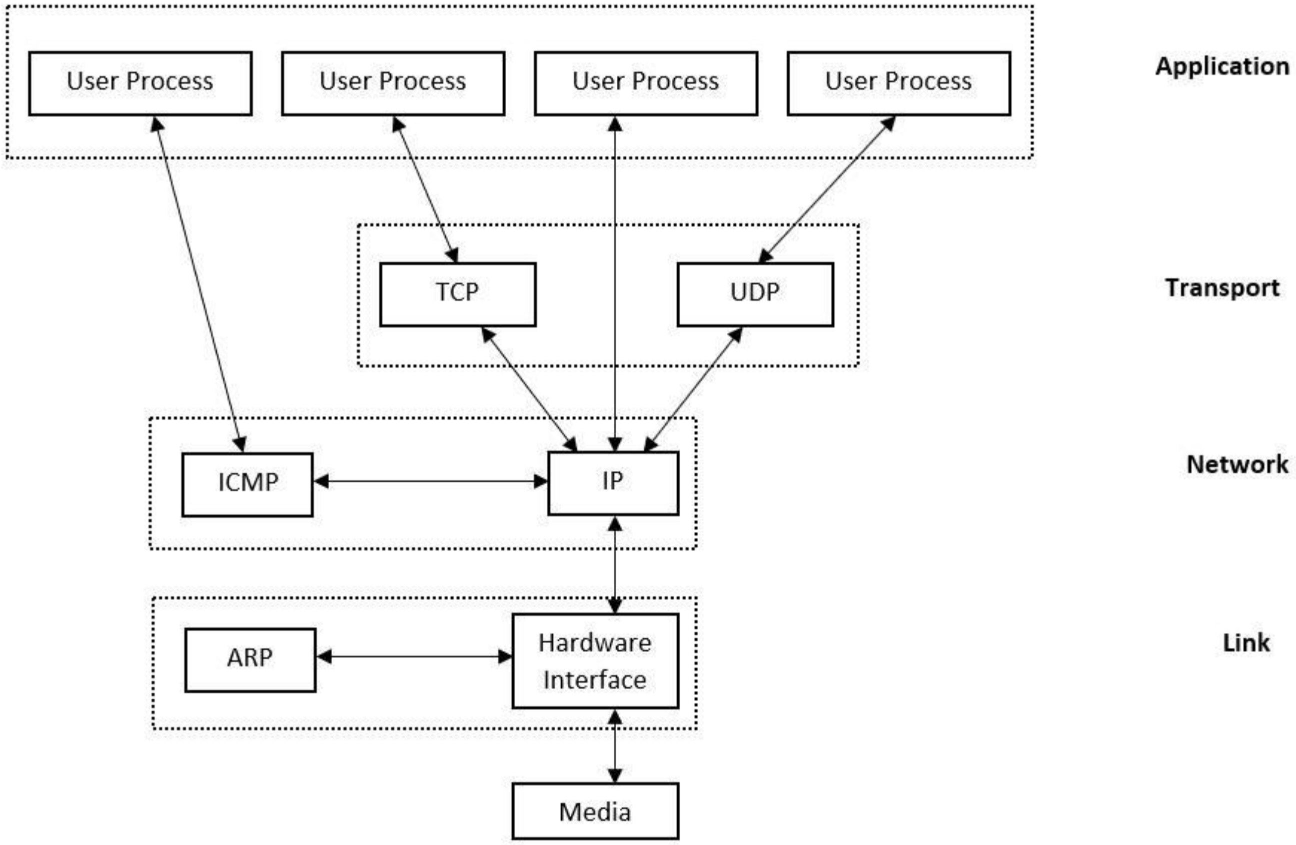


Figure 1: TCP/IP Protocol Suite protocols

2.2.1 UDP Protocol

UDP is standardized by IETFRFC 768, and its concept is dissimilar from TCP. The User Datagram Protocol (UDP) utilizes a straightforward connectionless transmission with a base prerequisite of the conventional system. It has no handshaking exchanges, and in this way uncovered trickiness of the fundamental system convention to the client's program. There is no assurance of transmission, requests, or copy insurance; however UDP provides checksum for data trustworthiness, and port numbers for tending to many volumes at the source and goal of the datagram (Postel J. B., 1981). By and large, UDP is a proficient convention for adequate data transmission. Notwithstanding, UDP can't transmit proficiently in some systems, for example, remote system. It is also a connectionless (Postel J. , 1980), unreliable protocol that delivers a mechanism for applications to forward encapsulated IP datagrams, without the need to create a link between the communicating nodes (Brownlee & Claffy, 2002). UDP does not have a flow control mechanism and lacks functions related to acknowledgement of received packets (Gopinath, Kumar, & Sharma, 2013). Error checking is not a prevailing feature: if it detects an error, the packet is dropped silently. UDP can be described as a no-frills, bare-bones transport protocol.

2.3 Network Monitoring System

Network monitoring system is the use of application systems, tools and techniques to constantly monitor the computer network system of an organization from slowing and failing components which notifies the system administrator in case of an outage (Easley & Kleinberg, 2010). Basically, a network monitoring system has three important responsibilities namely performing the smoothness and state of the network, providing reports on network position and providing report on event reports. A new model for network monitoring system was proposed by (Slowman

& Jonathan, 1994), which changes the event administration in formerly projected models. The key focus of this planned original model stayed on the actions and position of the linked arrangements of the network and the aim was creating monitoring report to be referred to the manager of the network. The Ping command was used to either spontaneously produce report on the administrator's demand (Slowman & Jonathan, 1994). As mentioned in first model, activity and status reports have been created for the administrator to analyze the network performance and also the connections with workstations. Figure 2 presents the design of the new system by Morris Sloman (1994).

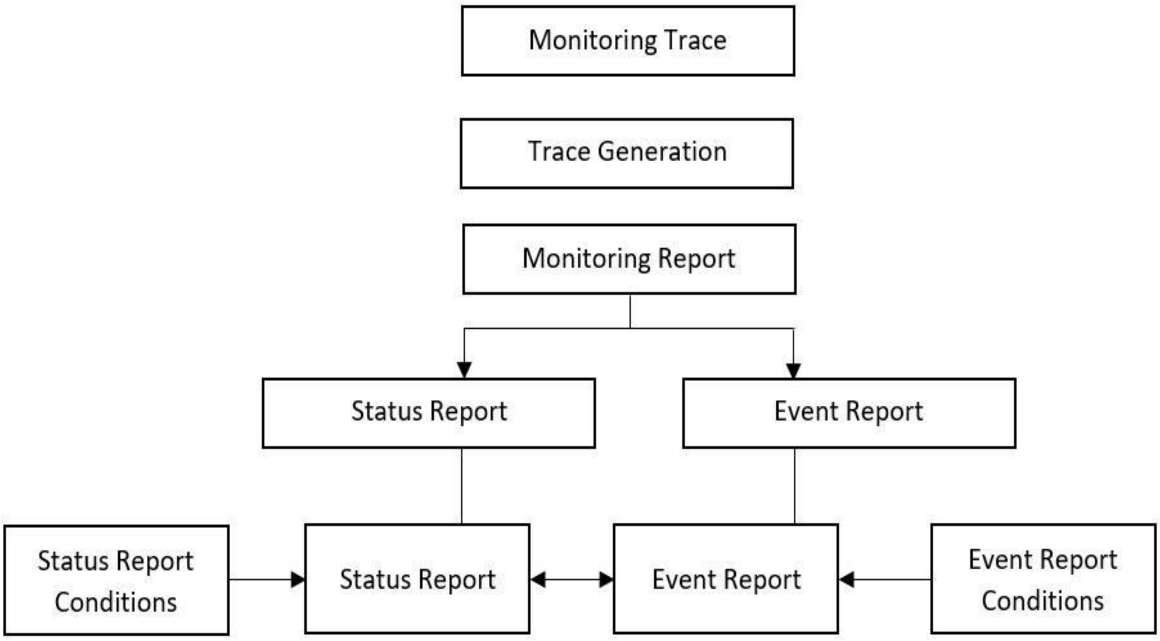


Figure 2: Network Management Model

2.4 Network Traffic Analysis

Network movement examination has become increasingly vital and important in present day networking for constant monitoring of the network traffic due to the ever-increasing number of packets that travel along the network. In previous years, administrators were monitoring only a few system equipment which is usually fewer than a thousand computers. The bandwidth of the network may be just less or about 100 Mbps (Megabits per second). Currently, managers need to contract advanced speed wired network (more than 1Gbps (Gigabits per second)) and various networks such as ATM (Asynchronous Transfer Mode) network and wireless networks (Ohlsson & Wernersson, 2014). They require additional modern network movement examination tools to manage network, resolve the network hitches swiftly to prevent network failure, and handle the network security.

2.5 Network Traffic Theory

Network traffic can be explained with reference to the interference or the flow of data from one end of the network to the other end of network whiles network traffic analysis is also explained as the observation of the data flow from a given source to a given destination (Muhtadi et al., 2020). In networking, data flow is always accompanied by time. So, the amount of data travelling must be able to reach its destination at a particular time. In effect, if the amount of data travelling is delayed, then it is proper that analysis of the network must be put in place to improve data flow in the network. In this study, the time taken as well as the purpose of the data flow is of greater interest (Marc, 2010).

2.6 Network Traffic Analysis Techniques

Network traffic analysis techniques involves obtaining information of the network data by capturing the header field of each packet, calculate them and generate the outcome thereof in order to increase its efficiency (Sciancalepore et al., 2019). One of the greenest web data analysis is packet decoding also called packets analysis where all headers fields are decoded and presented in a human readable format. Some network analyzers are tcp dump (Garcia L. M., 2008) and Wireshark (Chappell, 2012). The study discusses two notable techniques commonly used by Administrators. Network analysis is the procedure of capturing network traffic by monitoring carefully to regulate what is taking place on the network to take decisions based on performance (Tierney, 2004).

a. **Router based Monitoring techniques**: In this technique, they are hard-coded into the routers and therefore they do not suggest more flexibility.

b. Non-Router Based Techniques: Despite the fact that non-switch based strategies are as yet controlled in their capacities anyway they do provide more malleability than the switch created procedures. These procedures are delegated also dynamic or aloof.

i. Active Monitoring Techniques

Active monitoring technique conveys probes into the network to assemble measurements between a minimum of two endpoints in the network. Active measurement systems handle metrics such as: Network Availability, Routes, Packet loss and Packet delay regularly, utilities such as ping, which measures delay and loss of packets, and traceroute which aid in finding topology of the network, are instances of elementary active measurement tools. The above mentioned send Internet Control Message Protocol packets to a specified destination and wait for the host at the destination to reply to the sender at the source.

ii. Passive Monitoring

Unlike active monitoring, passive monitoring does not introduce traffic into the network or alter the traffic that is already on the network. Passive monitoring rather gathers data about only one point in the network that is being measured rather than between two endpoints as active monitoring measures. Passive measurements provides information such as: Protocol and traffic data, packet timing, inter-arrival timing, packet rates or precise bit, and accurate bit. Packet sniffing programs usually help in achieving passive monitoring. Passive monitoring has its own list of shortfalls. Measurements can only be examined off-line and not as they are collected with passive monitoring. This generates another challenge with processing the enormous data sets that are gathered.

2.7 Data Sets

Testing and evaluating is an important aspect of network traffic analysis (Sciancalepore et al., 2019). To evaluate the effectiveness of all research works using similar standard list, it is recommended to use standard data set as shown in figure 3 below. This study attempts to enlist a few important data sets that are being considered by researchers for network traffic analysis. These datasets include DARPA, NSS-KDD, CAIDA, Waikato, Berkeley, ACM SIG COM M '01. It was a first of network traffic analysis data set in relation to intrusion detection system (IDS) (Song et al., 2020). As per Stademeyer & C.W. Omlin (2009) DARPA incorporate two sorts of information caught from the system interface specifically TCP dump and framework review information.

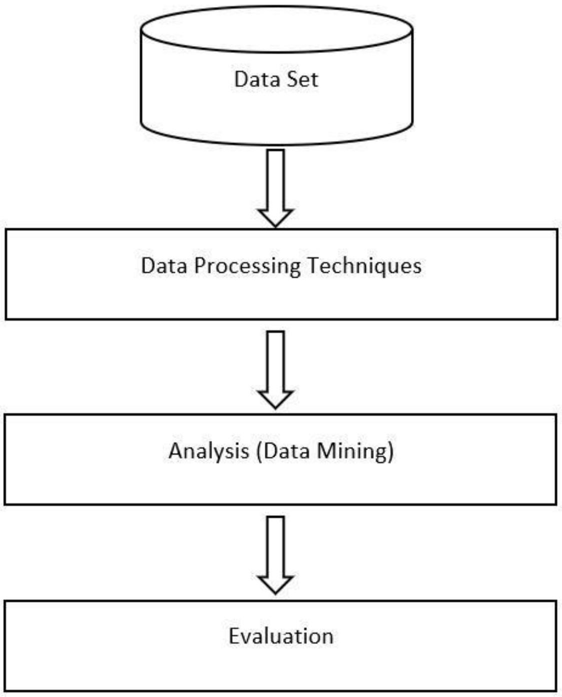


Figure 3: Generic structure of network traffic analysis

2.8 Feature Selection Method

Feature selection (FS) is a preprocessing method to be applied before applying data mining techniques. Feature selection used to improve the data mining techniques performance through the removal of redundant or irrelevant attributes. Feature selection methods generate a new set of attributes by selecting only a subset of the original attributes (Wu et al., 2015). Feature Selection is used mainly to reduce dimensionality of data set for improving network traffic analysis. We present various preprocessing techniques that are being used by researchers before actual analysis of network traffic. We have identified some techniques including principal component analysis, information entropy, rough set theory, feature selection is used frequently for preprocessing network traffic data.

2.9 Data Mining

Data mining (DM) is used for knowledge-discovery. Data mining plays an important role in analyzing network traffic. The intension is to present various data mining techniques that are used by researchers for analysis network traffics (Sciancalepore et al., 2019). Data mining techniques have been grouped under four broad categories namely; clustering, classification, hybrid and association rule techniques as shown in Figure 4. The detailed description of each technique and its usage is presented.

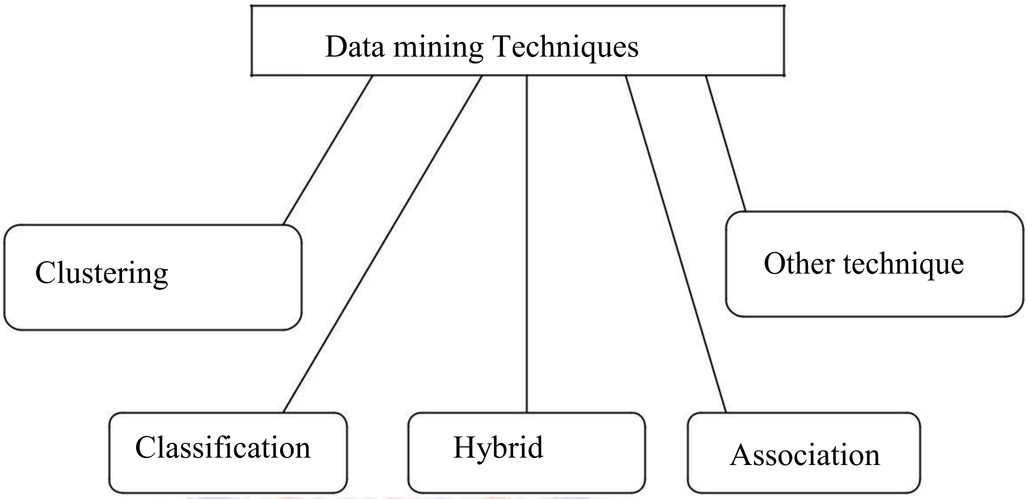


Figure 4: Data Mining Techniques in Network traffic

2.10 Packet

A packet is a piece of data which is routed transmitted from a source to a specified destination on the Internet or any specified network. The email, Web page retrievals, entire file downloads, all these Internet communications normally happen in the form of packets (Ohlsson & Wernersson, 2014). The Transmission Control Protocol (TCP) layer of TCP / IP divides the file into chunks of an efficient size when a given file is to be sent between parties for easy routing. The packets comprising the information are given separate numbers and has the IP address of the desired destination machine

(Sanders, 2007). Basically as series of digital numbers, it conveys the following: *the originating Internet Protocol address and port numbers*, *the destination Internet Protocol address* and *port numbers and hop count information*.

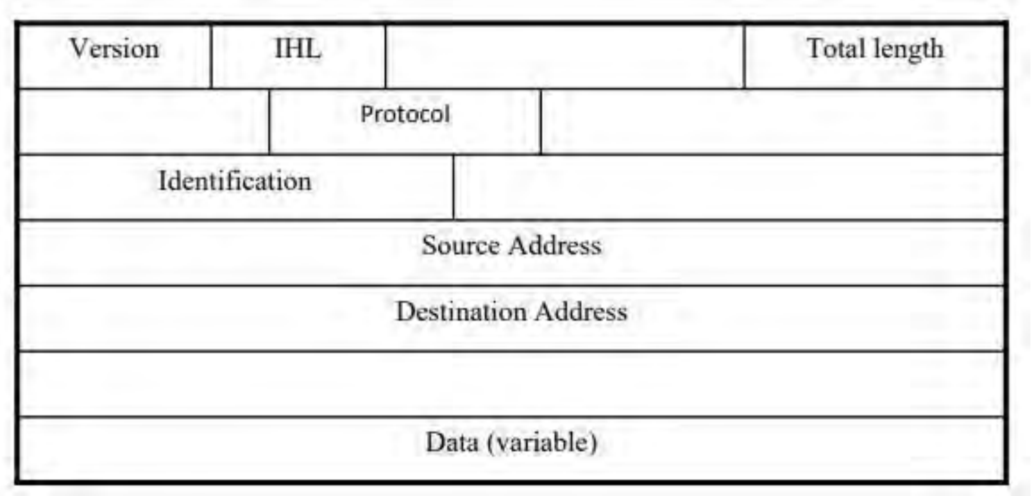
Liable to the protocol(s) that needs to support, the packets are transformed into a standard packet format. It formats include the body containing the message data (*payload*), a header, and *trailer* (Mingqiang, Z., Hui, H., & Qian, W, 2012). The packets convey the data in the protocols that is predominantly used by the Internet, TCP/IP. The body of the message can be found in each of the packets being transmitted. A normal packet has about 1,000 or 1,500 bytes (Lucas, 2010). Most packets are split into three parts:

Header – The header comprises instructions about the data carried by the packet. Some of these instructions may include:

- Protocol indicates the packet type being transmitted: Web page, streaming video, e-mail
- Destination address (where the packet is supposed to go)
- Originating address, where the packet emanated from (Dabir, A., & Matrawy, A, 2007). Each packet header has the right protocols, the originating address, packet number (1, 2, 3, or 4) and the destination. Routers in the network look at the destination address in the header and link it to their lookup table to locate where to transmit the packet. This nature of protocol was adopted for this work.

2.10.1. IP Packet Format

The information below defines the IP packet fields demonstrated in Figure 5 above. IP packet parameters that were used for the work are described as follows.



Version | IHL | | Total length

| Protocol |

Identification |

Source Address

Destination Address

Data (variable)

Figure 5: Fields of an IP packet (Harpence, 2011)

IP Header Length (IHL) — indicates the datagram header length in 32-bit words.

Add up to Length — determines the length, in bytes, of the whole IP bundle, including the information and header.

ID — comprises a whole number which recognizes the current datagram. This field is utilized to aid in assembling datagram parts (Lucas, 2010).

- Source Address — determines the sending hub.
- Destination Address — determines the accepting hub (Odom et al., 2006)

Despite the fact, the IP bundle organize has very number of parameters, the source and goal address, parcel length, distinguishing proof field, parcel sort, window size were utilized as a part of this work.

2.10.2 IP Addressing

As with any other network-layer protocol, an important procedure in routing IP datagram via a network is the IP addressing scheme (Susila, 2020). The given IP address has precise parts which normally has a fundamental structure. The given Internet Protocol addresses can be further divided and used to create addresses for sub-networks (Rooney, 2011). Based on this addressing scheme, users within organizations are monitored by the kind of domain or network they belong to, thereby tracking their source of generating traffic (Buck, 2001).

2.10.3 Internet Routing

The movement of packet from source to destination experiences some elements of setbacks called delay (Myipaddressinfo, 2006). For the purposes of this work, only transmission delay will be looked at: time it takes to transmit a packet. Furthermore, since packets crosses router, the number of router which is called hop count also affects the packet movement as a result of the delay (Packets, 2012).

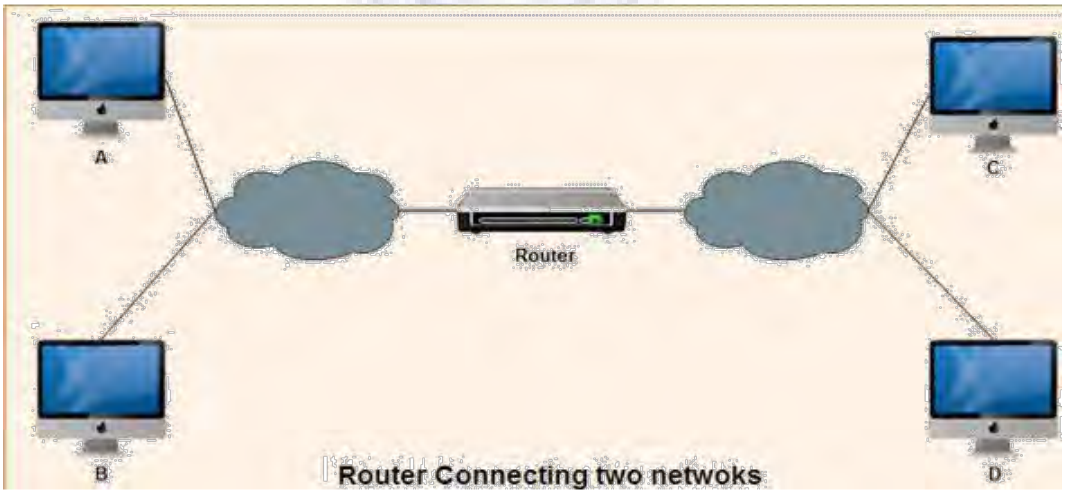


Figure 6: Movement of packets between two routers

Figure 6 has four work stations, workstation A, workstation B, workstation C and workstation D, and a router: router 1. Packets moves from workstations A or B, traversing the router to workstations C or D.

2.10.4 Port Numbers

Ports are transport layer (TCP and UDP) connection which are points numbered from 0 to 65,535. According to the Internet Assigned Number Authority (IANA) classification, the port space of 0 to 65,535 is broken down into three ranges in the process data travelling (Caliskan, 2011). These ports are used by the operating systems to make connections to the remote systems. Through ports, the system is able to identify which service is requesting from another system. In addition, the work can use the port number system to trace the source of traffic.

2.11 Network Packet Capture

Network packet capturing application, commonly are programs or libraries that get data packets flowing via a particular network segment through which the system is connected to by means of a network card (Clos, 2010). These captured packets from a network are processed, for example deciphering headers data and display it or obtaining data from headers for subsequent calculations. In this work, the major task will be obtaining packets from a network card or network interface, and analyze it (Satya & Srikanth, 2004). Network packet capture can capture packet data from the layer called data link of the ISO-OSI model. This includes payload and headers. Thus, Packet capture includes each packet that crosses a network segment, regardless of protocol, source (Casad, 2004). The captured packet serves as the base for the monitoring, and therefore this research work adopts this technique.

2.11.1 Packet Monitoring in an Ethernet Network

Packet monitoring can be attained in an Ethernet Network the reason being its broadcasting ability. There is a broadcast of packets to every machine in an Ethernet network. The component responsible for accepting or rejecting broadcast packets is the Ethernet card. The packets passing through an Ethernet Network can be captured. In addition, the Ethernet card has a many of modes which can be achieved with some drivers to capture packets (Wheaton, 2016).

2.11.2 Network Interface Card (NIC) and Driver

Drivers are created by computer programmers to achieve the needs of a specific computer application. The Operating System is able to communicate with input and output devices using drivers. Drivers act as translators which converts the basic requests obtained from the Operating System into instructions that precise peripheral controllers can comprehend. For network device communication, network interface card is needed (Calsoft, 2012). NIC drivers interface is directly fix to the hardware (NIC) at its lower edge and at their upper edge which provides an interface that helps the upper level drivers to send and obtain packets (Calsoft, 2012). Thus, any defect of the NIC may impair the capturing process, since device selections are based on it.

2.11.3 Modes in an Ethernet Interface Card

There are two main modes in an Ethernet card:

1. Directed

A frame that is destined to a specified machine has the end point machine's Ethernet address specified as its destination address. The machine with that physical address takes the frame while all the others reject it. The card can also be set to only receive fixed frames with an aid of a program (Degioanni, 2000).

2. Promiscuous

A card placed in this mode accepts all forms of packets running into it. Coupled with the broadcasting ability of the Ethernet, this mode is a key for the packet monitoring application to capture effectively. The PACKET.SYS driver aids to place the network card in all modes. The application with the aid of the PACKET.SYS places the card in the promiscuous mode to monitor the packets travelling or moving in the network.

2.11.4 The use of TCP/IP and UDP Traffic Monitoring Tool

The main objective was to analyze the TCP/IP and UDP traffic. The analysis was done by the use of Wireshark network traffic monitoring tool. Methodology used was an analysis, design, implementation and testing. The differences with this work are the development of this work with Wireshark program, adaptor capture and the main objective, which is users' activity with respect to delay, bandwidth, and congestion (Rafiq, 2005)

2.11.5 The use of Wireshark to Capture Packet

This work was used on network analyzing tool to capture packet from the network interface card, and analyze some protocol such as ICMP, TCP, and UDP. The packet-capturing program works on both windows and Linux Operating System platform. This study combines both passive and active monitoring, graphs and bandwidth analysis.

2.12 Capturing Packets

A packet sniffer is software that captures and analyses network traffic which transient through a computer's Network Interface Card (NIC). Figure 7 depicts an invader attacking data on a network.

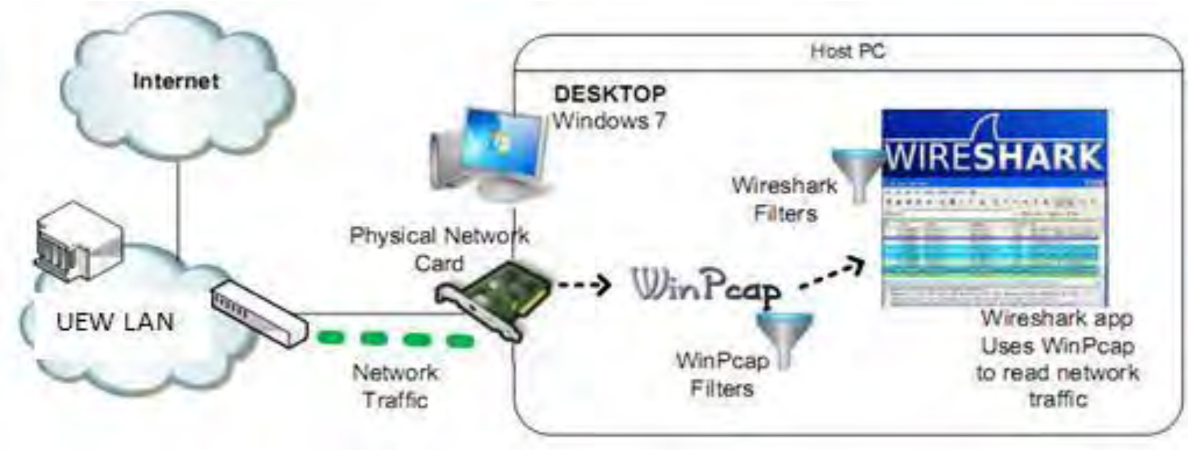


Figure 7: Capturing Data Traffic with Wireshark

Packets from the internet pass through the Local Area Network of UEWK. The Wireshark application provides an interface for detecting the Network Interface Card on which the data would be captured and analyzed by the use of another application called WinPcap. The Wireshark application specifically uses WinPcap to read the network traffic. Figure 8 depicts how packet is captured from the network through the physical layer using wincap and forwards it to wireshark which is the main application for the analysis of the data for further interpretation.

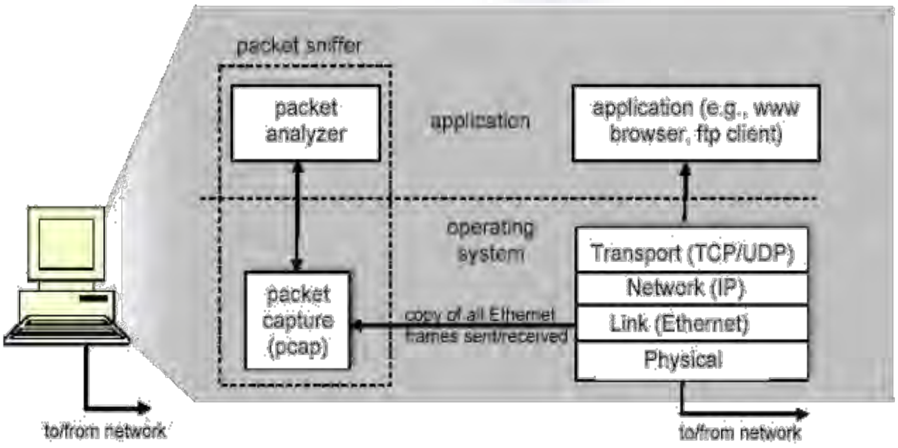


Figure 8: Packet Data Structure

Obtaining packets from a given network depends highly on the type of the network used and the type of configuration and topology of the network. LAN networks based on the IEEE 802.XX (physical and link layer protocols) protocol family, for example in the IEEE 802.3 protocol based networks, also referred to as *Ethernet networks*, and in the IEEE 802.11 (Odom & Thomas, 2006) based networks, called *Wifi or Wireless* networks demonstrate this. In IEEE 802.3 LAN network, a star topology is used, so all the nodes in the network are linked through their own cable to switch. Switches send packets to the port where the destination host is connected, by previously identifying all the hosts connected to each port (Ohlsson & Wernersson, 2014).

IEEE 802.11 based networks share access medium, so it may be faster than IEEE 802.3 switched networks to monitor packets, as having a network card being able to be set to promiscuous mode (actually monitor mode) is all the hardware required. However, some considerations have to be kept in mind. When placing a capturing system in wireless network, some packets or even all the packets sent by a certain host may be lost, due to environment conditions (shadowing) and the physical position of the capturing tool host and the other hosts in the network (Clos, 2010).

2.13 Network Traffic Data Inspection Techniques

Network data inspection techniques allows the administration of a system to get information of network data by examining network header fields of each packet, calculate them and yield outputs or results. The simplest network data inspection possible is packet decoding, also called packet analysis, in which all header‘s field are deciphered and presented in a human readable way. Network traffic analyzers like tcpdump (Garcia M, 2010), or Wireshark (Chappell, 2012), are some examples of packet decoding applications. This study captures the packet block decode it and extract the parameters, present it graphically and draw statistical information and pattern extraction. Graphical representation of packet data is of interest in this work, mainly in network metrics on user movement.

2.14 Remote Access

Remote access gives authorized access to the network administrator for performing, remote access gives approved access to network administrator for carrying out required actions on the systems linked to the network such as routers, workstations and switches. To provide remote access data of Transmission Control Protocol is interpreted by the proxy server; for the router's console port and obtains it back. With regards to this procedure, all linked devices to the stated proxy server would be prepared for remote access, performed by server's user. The administrator of the server decides the essential activity on each linked device. The network administrator creates a connection amongst two routers which are the connected one to network monitoring system's interface and the router which is connected to workstations. Every workstation must have the client version of the system. By creating the connection between workstations and server, all the required activities can be completed between them (Michalski, 2009). The Internet Protocol address of each connected device will be kept in the server for providing the ability of choosing specific client or workstation by server, to be able to achieve any essential actions on the desired device.

2.15 Some Remote Access Systems

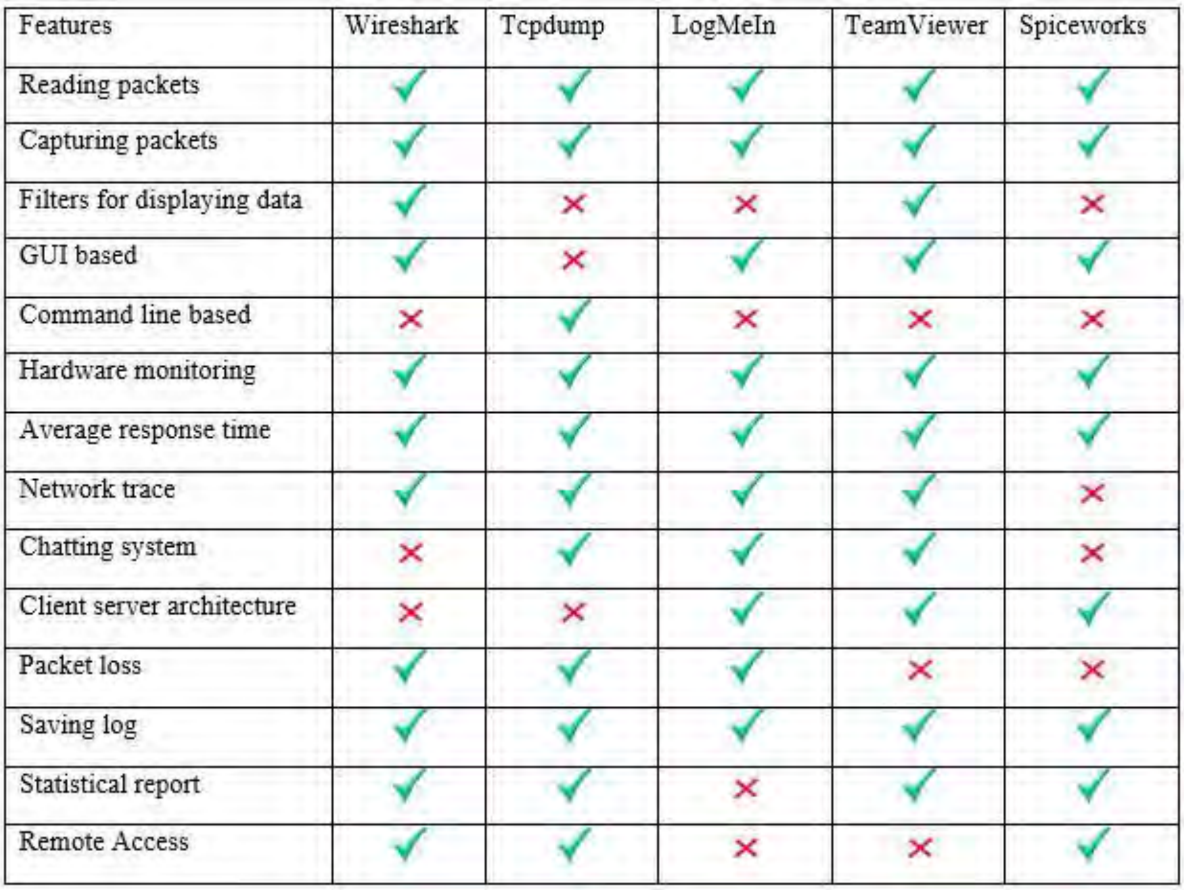
Remote access is the connection to a system from a secondary location other than that of the primary location of the system being accessed (Lahaie, 2013). Some existing remote access technologies include: Remote Frame Buffer (RFB) protocol, TCP/IP protocol, COMPROID, Ultra Virtual Network Computing, Remote Computer Access through Android Mobiles, TeamViewer, LogMeIn, Remote Control System, Windows Remote Assistance, SMS based Remote Control System, Wireshark, Spiceworks, OpManager, Tcpdump, Remote Desktop. The section of the study looks at each of these remote access systems, compares them with one another, indicates their strengths and weaknesses, and goes further to compare each with the system this research intends to implement.

2.15.1 Transmission Control Protocol/Internet Protocol

The Transmission Control Protocol/Internet Protocol (TCP/IP) is a suite used for communication and has become the standard in the industry for connecting hosts, networks and the Internet. The TCP/IP protocol is seen as the backbone behind the Internet and networks globally. TCP/IP protocol is simple and with a design goal built to interconnect networks can communicate services over heterogeneous physical networks. The TCP/IP protocol serves to enable communication among hosts on other networks separated by a vast geographical area. TCP/IP is based on three modules of security which are security control, security policy and data security layers. The application layer holds the security policy, with the transport layer holding the security control whereas the security layer is held between the transport and IP layers (Jeya., Ravichandran, & Ravichandran, 2012). System interacts with the security policy layer to determine the kind of security needed to be useful to data in communication; the security control layer provides the mechanism to render the security policy defined by the network administrator in the security policy module to ensure secure communications (Anil & Shina, 2020).

TCP/IP is also capable of creating a standardized concept of the methods for communication which every kind of network provides. TCP/IP ensures linked services that run amid the programming interface of a physical network and user applications (Wu et al., 2015). It also allows a shared interface for the applications, regardless of the underlying physical network. The user and application developer therefore is rendered oblivious of the architecture of the physical network. TCP provides a connection-oriented, reliable byte stream unlike UDP known for connectionless unreliable services (Fall & Stevens, 2011). When identifying hosts within the network, hosts are assigned IP addresses; hosts with several network adapters such as router are also assigned unique IP addresses. A TCP data header contains the source and destination port numbers and the source and destination IP addresses in the IP header which exclusively identifies every connection (Fall & Stevens, 2011). The TCP protocol allows for labour division, code testing, and alternative layer implementations development by separating communication software into layers (Song et al., 2020). The program that uses TCP/IP makes provision of the application layer for communication. An application refers to a user procedure cooperating with another procedure normally on a different host. Hypertext Transfer Protocol (HTTP), File Transfer Protocol (FTP) and Simple Mail Transfer Protocol (SMTP) are the common forms of applications. A virtual network image of an internet provided by the network layer buffers the higher levels from the physical network architecture below it. The IP could be seen as the integral in this layer but is connectionless and provides no support for reliability. Also part of the TCP/IP protocol is the Address Resolution Protocol which is used to locate the MAC address of hosts conforming to a particular IP within the same subnet (Nam et al., 2012). The ARP spoof poses major security problems to networks and can cause Man-In-The-Middle or Denial of Service (DoS) attacks (Dubey, Gupta, & Bhujade, 2011). As shown in the table 1.

Table 1: Comparison of provided Application with other Systems



Features	Wireshark	Tcpdump	LogMeIn	TeamViewer	Spiceworks
Reading packets	✓	✓	✓	✓	✓
Capturing packets	✓	✓	✓	✓	✓
Filters for displaying data	✓	✗	✗	✓	✗
GUI based	✓	✗	✓	✓	✓
Command line based	✗	✓	✗	✗	✗
Hardware monitoring	✓	✓	✓	✓	✓
Average response time	✓	✓	✓	✓	✓
Network trace	✓	✓	✓	✓	✗
Chatting system	✗	✓	✓	✓	✗
Client server architecture	✗	✗	✓	✓	✓
Packet loss	✓	✓	✓	✗	✗
Saving log	✓	✓	✓	✓	✓
Statistical report	✓	✓	✗	✓	✓
Remote Access	✓	✓	✗	✗	✓

2.16 Network Traffic Monitoring Analysis on Network Service Operation

The main objective this study was to make analysis of network traffic monitoring by using application like Wireshark or protocol: HTTP, FTP, Telnet, and SMTP, TCP. The findings in this work were the best way to improve the network performance by implementing the Solution design to improve organizational efficiency (Sciancalepore et al., 2019). This study highlighted how packet travels across a network and the use of applications alongside some of the common protocols but the emphasis was solely on the performance and accountability network (Mashitah, 2003). Network administrators should make it a point to draw up a program that would constantly analyze the amount of data travelling through the network in order to ease the traffic and improve performance (Muktar & Kyauta, 2017).

CHAPTER THREE

METHODOLOGY

3.1 Introduction

Chapter three covered the methodology of how the proposed work was planned from the requirements analysis. This chapter constitutes an introduction, research design, Technology used, Description of Modules used in the Study, Requirement Documentation, Conceptual design of the system, and Design.

The problem observed in this study was that when users of the network continue to multiply in numbers and also when some particular users are able to abuse time and resources allotted to them during peak hour while visiting other domain/internet when they are not supposed to do so. When this happens, it results in additional load thereby causing traffic in the network which then becomes difficult for the system administrator to be able to identify the source of problem at hand. The study intends to use Wireshark application tool as a traffic and monitoring analyzing tool. The study aimed at analyzing traffic metric like bandwidth consumed, bandwidth delays, packet captured and filtering of packets segment using packet analysis tool. Packet analysis tool like Wireshark was used to visually monitor the traffic congestion in the network which is additionally caused by the users of the network through the introduction of additional traffic with their activities on the network (Sikos, 2020).

The basic architecture of the analyzing and monitoring tool consist of client application that system administrators can use to analyze, monitor, and solve network problems (Baig, 2012). It is used to detect packet traveling through the network and filter the packets to avoid packet delays in the network. Considering the network environment, the Wireshark application is able to detect and show the basic operation system, network configuration information as well as driver status (Bhoria, & Garg, 2013). Additionally, the Wireshark application can detect and ensure basic security mechanisms to provide a degree of anonymity, including display of IP addresses through a mechanism called packet sniffing. The system monitoring server has many different logical instances which allow the monitor servers which is in charge of displaying the various events being untaken to implement and capture packets and to measure user connections for traffic analysis (Arya, & Mishra, 2011). Finally, the server has a database that stores all the captured packets and gathers them for data graph representation. The server will then verify the identity and the integrity of the captured packet, which is classified according to their subnet or domain of the communicating host thereby establishing if the packets obtained are valid and not affected by external or remote sources.

Consequently, a visual analysis of the length of the packet and subnet sends the analyzed information to the administrator through the application. Several procedures were put together to formulate the methods and also an adopted part of standard waterfall model was included to help in the design. Furthermore, secondary resources were derived from various publications such as books, journals and internet to support the design analysis and findings. In the quest to get the major data to be used as solution, another network monitoring tool was used to capture data traffic. It was then used for analyzing the objectives of the study and was used to capture data flowing through network. Generally, the best and the cheapest way to analyze network problems such as non-router base is passive and active monitoring as seen in the section. The use of this approach is far less expensive as open source package products are inexpensive as compared to router based (Dabir & Matrawy, 2007).

3.2 Research Design

To be able to acquire, capture, and analyze the network traffic generated in the application as a result of unwanted activities of users, series of methods were employed to help in solving the problem at hand. To be able to capture data traffic, Wireshark application was installed and run to capture packets of data passing through the network interface card. Again, as part of the design process, certain tools were introduced which have been briefly described in section 3.2 and tabulated in table 2. The requirement for the network analysis and design of the application were done in section 3.5 and 3.7. The design and use of the application was based on the waterfall model which divides the life cycle of the application process into phases. This is demonstrated clearly in figure 10. As the design of the packet capturing is completed, the next phase was to choose the source which would supply the data traffic. Hence, LAN/WAN/internet of UEW became the source to collect data from. Before analyzing the data traffic, the captured packets were stored in a database and each information was extracted from the packet blocked captured.

To be able to monitor the amount of data travelling through a test network as a result of the activities of users and the effect of the chosen metrics on the network parameters from the remote sources, an application is run as a test-bed as illustrated in figure 9 which was used in turns and also a series of packets were captured and analyzed for better results using the Wireshark. The capturing of the packets was run in two separate cases. It was first run in a homogeneous case where the entire hosts are in the same subnet or domain and also a heterogeneous case where the entire hosts are in different subnet or domain. This was done to make the analysis of the network meaningful by gathering data from the congested network traffic which also happens by taking packets running through the network and reading them (Sciancalepore et al., 2019). When capturing and analyzing the packets running through the network for better performance, the time taken by the packets, the destination of the packets and the source IP address of the packets, length of the packet, the hop count, the identification of field packets type and the size of the packets were noted for every captured session through the homogeneous and heterogonous cases one after the other.

The packets captured were run through the application to test the values for the delay, the bandwidth consumption, and inefficient windows size and analyzed based on the congested network (Eid, Darwish, Hassanien, & Kim, 2010). In the course of the network monitoring and analysis demonstration, the following assumptions were made. Firstly, all the bandwidth running through the network connections were assumed to be the same for all connection routes. Secondly, the delay of transmissions between the sources and the destination pair was measured by packet length (bits) per link bandwidth (bit/s). As a rule, a requirement analysis was performed in which an application had to record packets to a permanent storage so that the analysis could be performed on the data. Additionally, records of the various packets captured were maintained so that the analysis could be made out of the traffic patterns.

In determining the kind of traffic pattern that is generated by each of the nodes, there was the need to first and foremost gather traffic data for the network analysis. When the gathering of data is done, then the analysis of the data from the output was obtained with regards to the chosen metrics. With this data in hand, it was necessary to determine if the data gathered was enough to proceed into graphing.

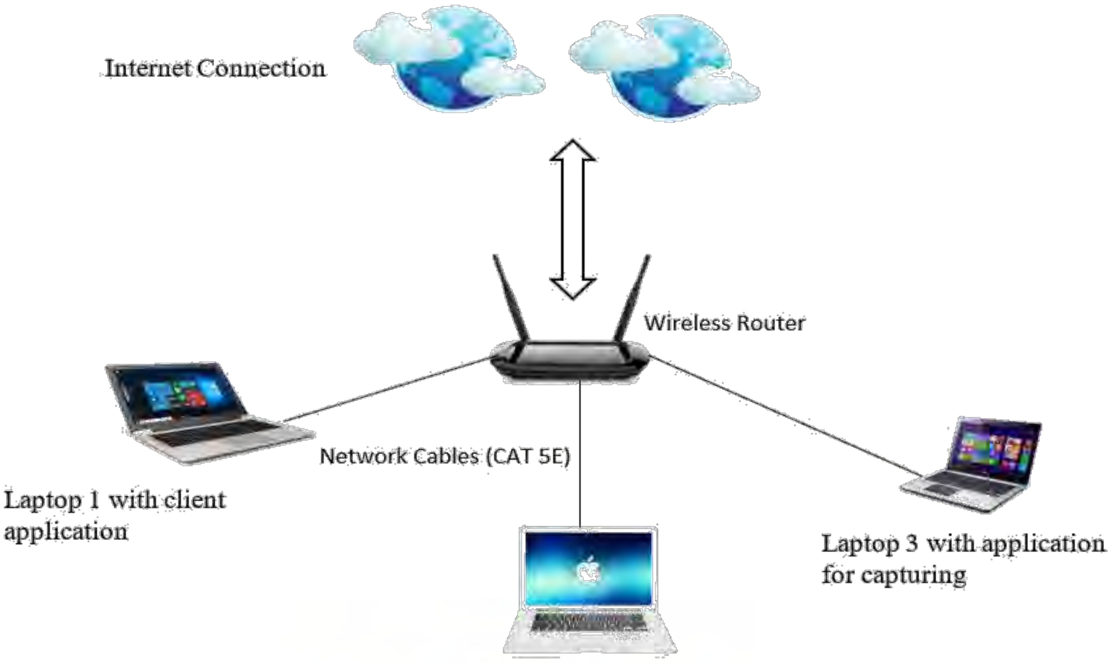

Internet Connection
Wireless Router
Network Cables (CAT 5E)
Laptop 1 with client application
Laptop 3 with application for capturing

Figure 9: Test and Demonstration Setup

3.3 Technologies Used

A commonest deficiency of an open source environment is the problem of repeating the duplication of system suites for the reason that the current suits do not contain the set of features required for a new technology (Erman, Arlitt, & Mahanti, 2006). In this study, several components from the scratch were not written rather minor modifications to present suits were done and implemented. The tools and technologies used are tabulated as shown below in Table 2.

Table 2: Tools and Technologies used for the study

Resource/Tool	Valid Percent
Microsoft Windows 7/Win Server 2008	Operating System
Microsoft Office 2007	Documentation
Mozilla Firefox	Default Browser
Wireshark Application	For Packet Capturing
Wincap	Windows Packet Capture

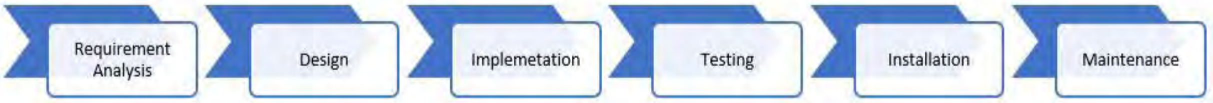


Figure 10: Classical Waterfall Model

3.4 Description of Modules used in the Study

i. Login/logout for Administrators

The System administrator is required to provide a login and logout credentials before accessing the application. The username and password is required in order to identify the authenticity of the user at a particular time for the purpose of security. A third party encryption was ignored for the fact it is only the administrator who is responsible for running the traffic analysis of the network system.

1. **Packet Capture:** When the application is ran, the inputs from the network is captured by the Wincap interface which is responsible for capturing the amount of packets flowing through the network interface card via a wire or wave. In addition to that, the Wireshark application has the following sub modules such as capture, packet capture analysis, load capture packets, view packets block, view graph and view packet details.
2. **Utility**: The network utility is a windows application that helps the Administrator to monitor the activities and the operations within the network system.
3. **Network Interfaces:** In order to be sure of the status of the interface**,** the module always gives information about all the hardware and software interfaces in the network whether they are running smoothly or not (Song et al., 2020).

3.5 Requirement Documentation

This is the requirements that are documented for the application. The system that would be developed would be used for capturing the packets flowing through the network, analyze them, and monitor status of the network interfaces as well as being able to visualize network configuration information with ease.

3.5.1 Scope

The study was taken at the University of Education Winneba – Kumasi. Kumasi grounds for University of Education Winneba have the College of Technology Education and is found 320 Kilometers from Winneba and 280 Kilometers north of Accra. The grounds has the accompanying resources; Faculty of Business Education, Faculty of Vocational, Faculty of Technical Education, and Faculty of Education and Communication Science (Kumasi, 2020). University of Education Winneba – Kumasi was chosen as a research site because the school has its' own network that are used by students, teaching and non-teaching staff. The research site was convenient for the researcher.

3.5.2 Functional Requirements

The functional specification is described briefly in tabular format in table 3. Its main components are shown which includes a brief description of the major functions of the system. Its main idea is to state the objectives and the functions which is related to the overview of the system. Also, state the main system block which is necessary for building, analyzing and monitoring the system. It is able to characterize packets travelling through a network. The system performs the function in the table through the server.

Table 3: System Functional Specifications

Component	Description	Related Component
Packet Capture (Server)	Captures packets from NIC	Analysis of the GUI User Component
View Packet block (Server)	View packet block before dividing into parameters	Analysis of the GUI User Component
View Packet Parameters (Server)	View detailed packet parameters required for the analysis	Analysis of the GUI User Component
Packet Analysis	Present analysis of the packet captured	Analysis of the GUI User Component
View graph (Server)	Allows you to view the graphs obtained from the metrics	Analysis of the GUI User Component
Network Configurations	Allows easy access to network configurations	Client Application
System specification (Client)	Allows easy access to basic operating system specification (OS type and version, CPU, DNS and memory)	Client Application

3.5.3 User Requirement Specification

These are the specified requirements that the user of the system expect from the system. These are follows:

User characteristics: The user of this system is only the network administrator who is responsible for creating user account, access right and privileges, access control as well, monitor and analyze the network traffic through the server and making sure the network is at its optimum level.

General Constraints and Assumptions: The main assumption is that the packets moving along the network are coming from only wired or wireless network.

3.5.4 Non-Functional Requirements

Nonfunctional requirements specify the criteria that can be used to judge the performance of the network system rather than the behavior of the system (Jeya, Ravichandran, & Ravichandran 2012). These are often called the qualities of the system which can be divided into two main categories.

i. Execution qualities, such as security and usability which are observable at run time

ii. Clarification of Portability and Accessibility Requirement: In today's network environment which is geographically distributed across a wide area, the ability of the administrator to gain access to monitor objects from a number of locations is becoming increasingly important.

Graphical User Interface

The user interface for this type of system should present the required information in a clear and concise format, giving accurate and timely information for the user when requested.

The user interface for this type of system should present the required information in a clear and concise format, giving accurate and timely information for the user when requested.

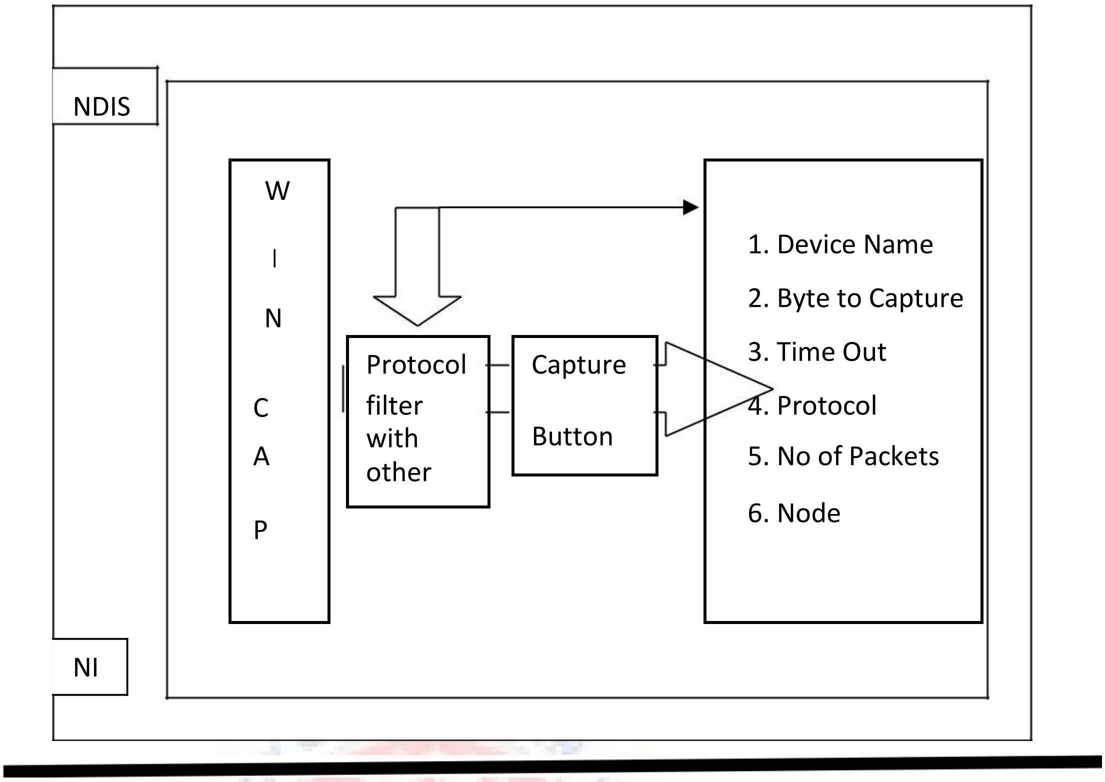


Figure 11: Block Diagram of the packet capture component interface

This diagram (Figure 11) represents the block diagram of a capturing component interface. An external site or packet travels across the network with a thick dark line.

Software Environment

This represents the unseen side of the system. This happens to be the one that supports the system.

Hardware Environment

This depicts the way and manner the system runs. It allows the users to interact with the system hardware which is also known as the physical component of the system. They are: Processor Speed: Intel Pentium P600 @ 2.13 GHz.

RAM	-	1.5 GB or Higher
HDD	-	30 GB or Higher

LAN - Enabled

The performance Constraints

The network speed should not be less than 10/100Mbps for LAN network.

3.6 Conceptual Design of the system

A UML modeling for the design which shows different system components and how data flow from one component to another in order to achieve the system objectives (Kerai, 2010).

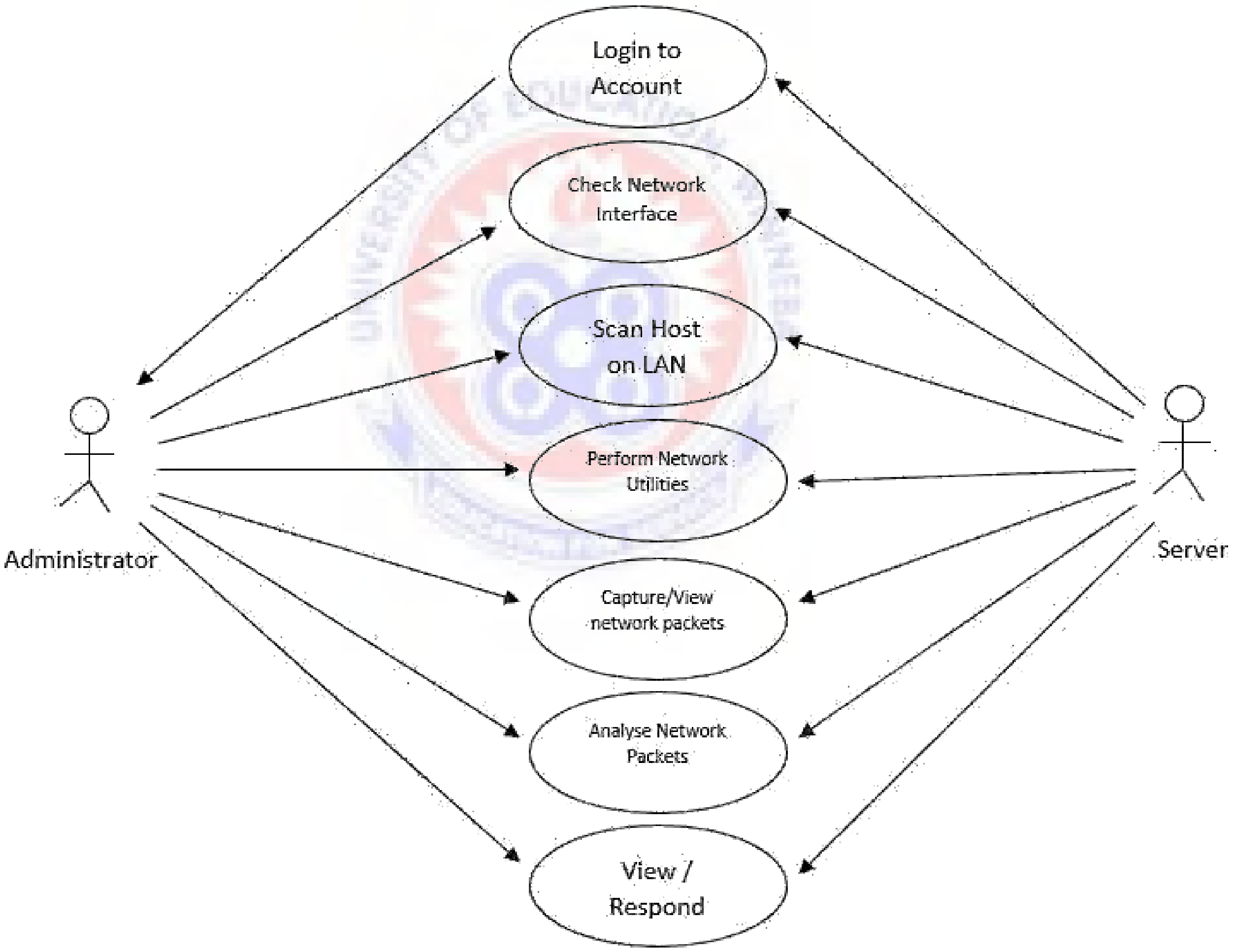


Figure 12: Use Case for system Design

Table 4: Use Case Description

Actor	Activity
Administrator	Manages the user accounts and logins. Views and checks network interfaces. Scan host on LAN Select Interface Card Filters Performs Network Utilities Capture Packets Setup System

3.7 Design

The researcher used UML Diagrams for the design.

3.7.1 UML Diagrams

a. Use Case

The diagram represents the Use Case of the system capturing tool and analysis tool. The main actor here is the administrator who is responsible for performing the packet feeder from the source of the packet by capturing any packet that is traveling through the network.

b. Data Flow Diagrams

A data flow diagram represents a graphical representation of symbols, numbers and system components which represent the flow of data from one end to another (lakhina, Joseph, & Verma, 2010). Usually, most data flow modeling methods use four kinds of symbols. The four symbols that represent the four kinds of system component are Processes, data stores, data flows, and external entities (Lakhina, Joseph, & Verma, 2010). The data flow diagrams for this particular work are shown in figure 13 and 14. The first diagram represents the entire packet process. It follows the step by step approach through packets flow s from one source to another. From the diagram, the inputs are the packets that are flowing from the network which are captured through the network interface card through to a direct mode.

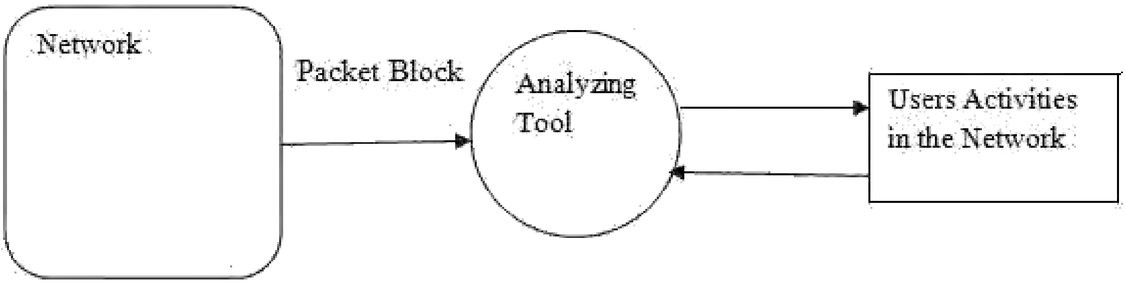


Figure 13: Principle behind network traffic analysis

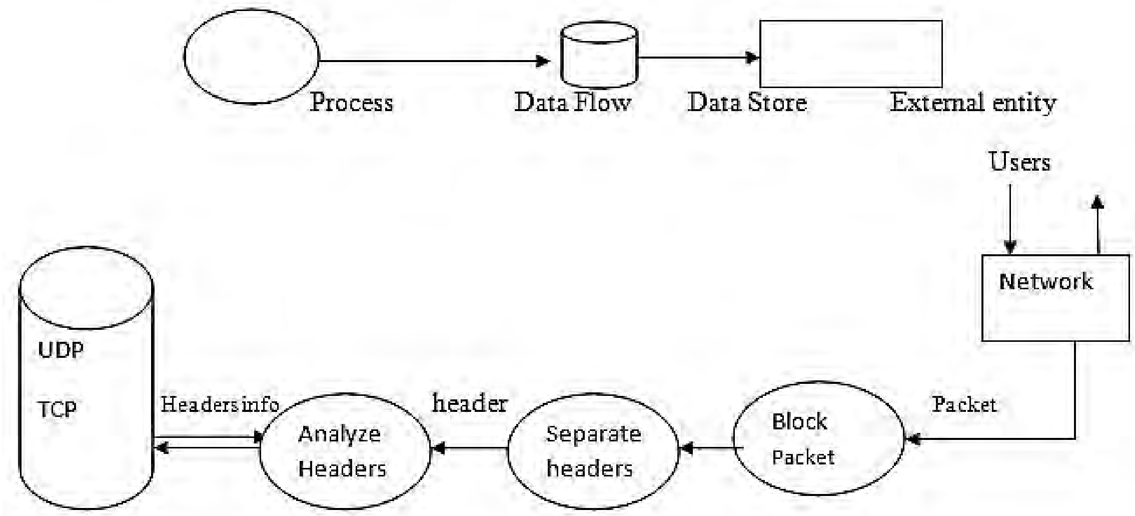


Figure 14: Data Flow Diagram of the Study

An input is obtained as packets pass through the network interface by the get packet process. The process defines lists of interfaces devices which obtains a raw packet from the network interfaces card and store them in a buffer. The buffer contains the packet which passes through the separate header process which strip off the header from the packet and passes them through analyze headers where they will be analyzed and the information sent to the database (wireshark.org).

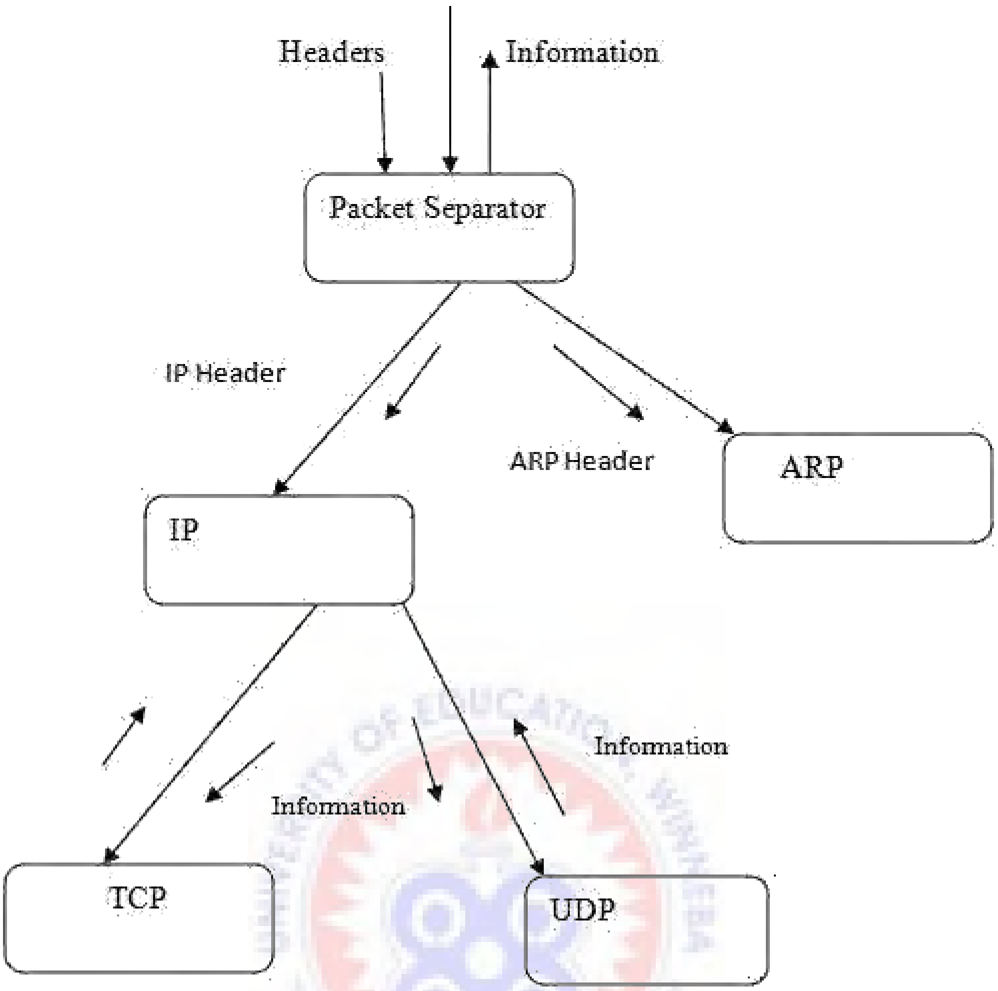


Figure 15: Packet Separator

The view packet block loads the packet through the packet graph and the packet graph analysis gets the information stored in headers of the packets as input from captured packet view which controls the packet view info, view packet block and view graph (Mingqiang, Hui, & Qian, 2012). The captured packet lets you load the packet from the database, view packet block and detail packet parameters and view the graph form (Miller, Deitrick, & Hu, 2011).

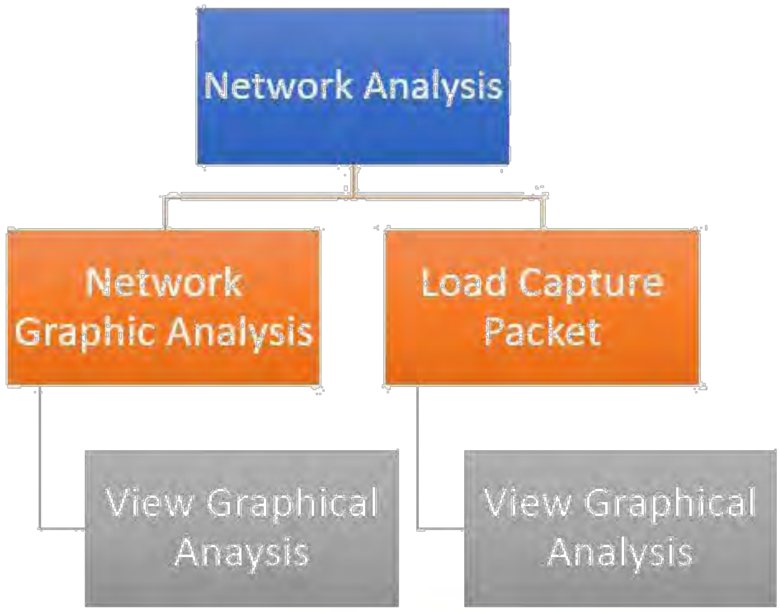


Figure 16: Network Traffic Analysis User Interface

The figure 16 above depicts the output that helps in the analysis of the detail packet parameters when the performance metrics are applied in the analysis. In this process, both source IP address and the destination, packet type, packet date, packet length, identification number, hop count, windows size and time displayed by the detail packet information as expected by the results of the analysis (Tian, Sun, & Hunga, 2008).

CHAPTER FOUR

IMPLEMENTATION, RESULTS AND DISCUSSION

4.1 Introduction

Chapter four constitute the introduction, Analysis of the System, Analysis of Packets Captured, and Statistic of Packets.

The Wireshark application in this study was used to track, monitor, analyze, and optimize congested network traffic which was generated because of various activities of users both internal and external. The research methodology that was applied in this study was descriptive in nature. The study used qualitative approach to design, develop and gather data for analyzing the packets passing through the network. After data has been gathered through the captured packets, the next step was by the use of demonstration method, the data for traffic analysis was obtained through graphical representation (Wu et al., 2015). The packets were captured using Wincap and Wireshark application which were presented in the form of tables and graphs to facilitate the analysis. The results obtained using graphs which were plotted to enable efficient and effective analysis of the system. The study use cause and effects of network traffic in analyzing packets passing through the network.

Furthermore, the information gathered from the captured packets which were generated by the applications was used to plot graphs for the analysis of the data. Finally, a test was designed to measure the accuracy of the data captured on packets which was run in chapter three of the study. Once the logical design is constructed which leads to the construction of the proposed design, there is the need to implement the features which is involved in design phase of the study. The applications that were used to capture data for the traffic analysis were implemented and the results communicated to the organization using plotted graphs captured from the server.

4.2 System Components

Here, the implementation of the traffic analysis tools is described. The study describes the main components in the Wireshark application interface. Before one can be able to do the analysis of the congested traffic very well, it is important for the System Administrators to be conversant with the Wireshark application interface taking into account the various menus on the Wireshark main window. The study will consider Wireshark application components such as login interface, captured component, graphical user interface and server dashboard. The study will concentrate more on the three major components of the Wireshark application.

4.2.1 Finding and Capturing Packets for the Analysis

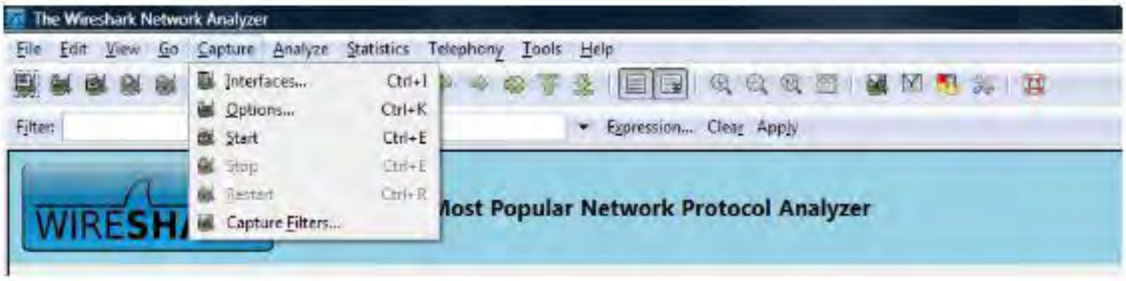
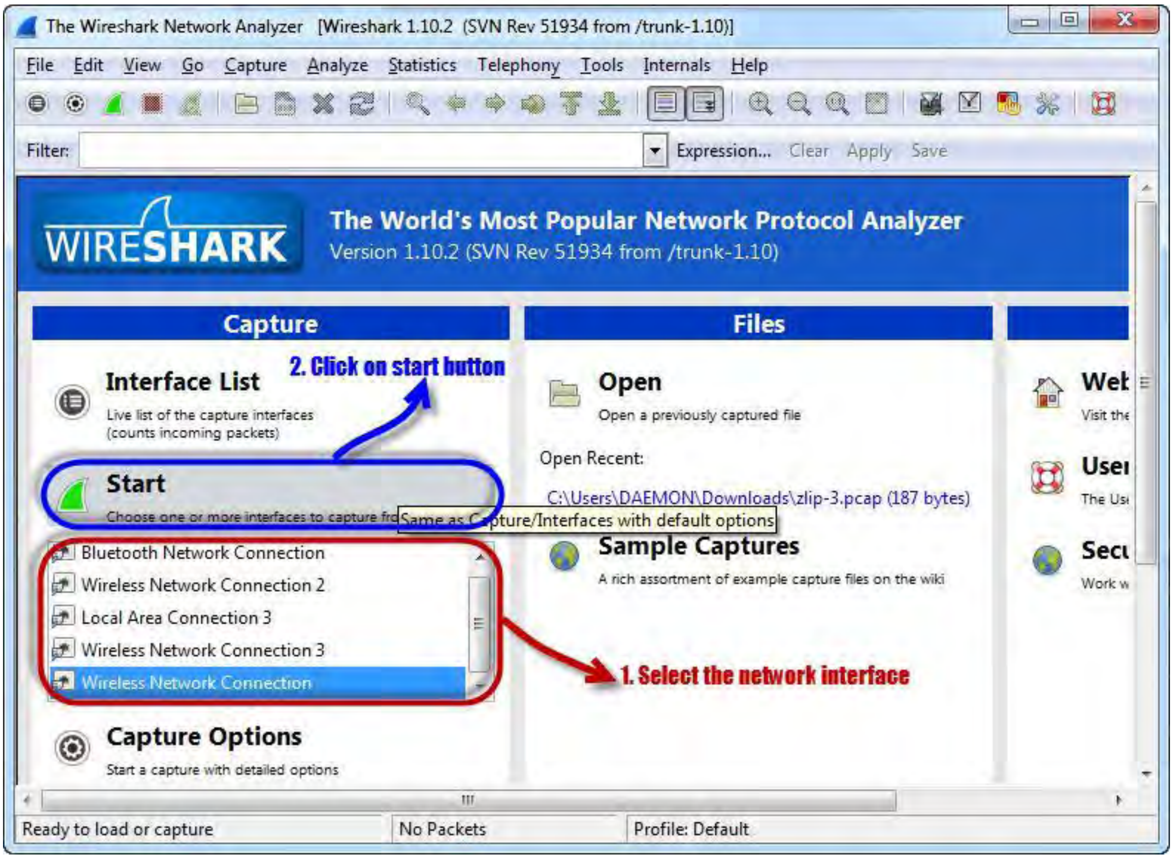

The Wireshark Network Analyzer
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Interfaces... Ctrl+I
Options... Ctrl+K
Start Ctrl+E
Stop Ctrl+E
Restart Ctrl+R
Capture Filters...
Filter:
Expression... Clear Apply
WIRESH
Most Popular Network Protocol Analyzer

Figure 17: Network Interface to Capture Packets

When the Wireshark application is first started, a default or blank window is shown in order to list the available interfaces on the network, the captured interface option menu must be first selected as shown in figure 17.

4.2.2 Network Hardware and Software Interfaces Verification

At any point in time the Wireshark application list all the hardware and software interfaces of the network configuration whether it is up and running or not (Zhou, Guangmin, & He, 2009). The hardware part takes part of the interface card whiles the software part takes care of the various drivers installed on the interface (Vijayakumar, & Parvathi 2010). Before any packet is captured or performed, the Wireshark component verifies both the software and hardware resources if they are working properly (Yu, & Fei, 2008).



The Wireshark Network Analyzer [Wireshark 1.10.2 (SVN Rev 51934 from /trunk-1.10)]

The World's Most Popular Network Protocol Analyzer
Version 1.10.2 (SVN Rev 51934 from /trunk-1.10)

2. Click on start button

1. Select the network interface

Interface List
Live list of the capture interfaces (counts incoming packets)

Start
Choose one or more interfaces to capture fro[m]
Same as Capture/Interfaces with default options

- Bluetooth Network Connection
- Wireless Network Connection 2
- Local Area Connection 3
- Wireless Network Connection 3
- Wireless Network Connection

Capture Options
Start a capture with detailed options

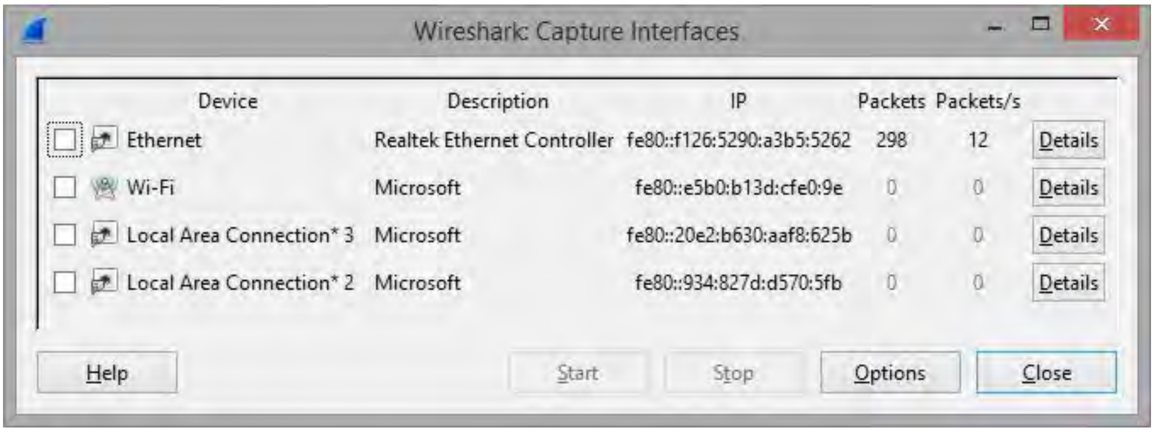
Open
Open a previously captured file

Open Recent:
C:\Users\DAEMON\Downloads\zlip-3.pcap (187 bytes)

Sample Captures
A rich assortment of example capture files on the wiki

Ready to load or capture | No Packets | Profile: Default

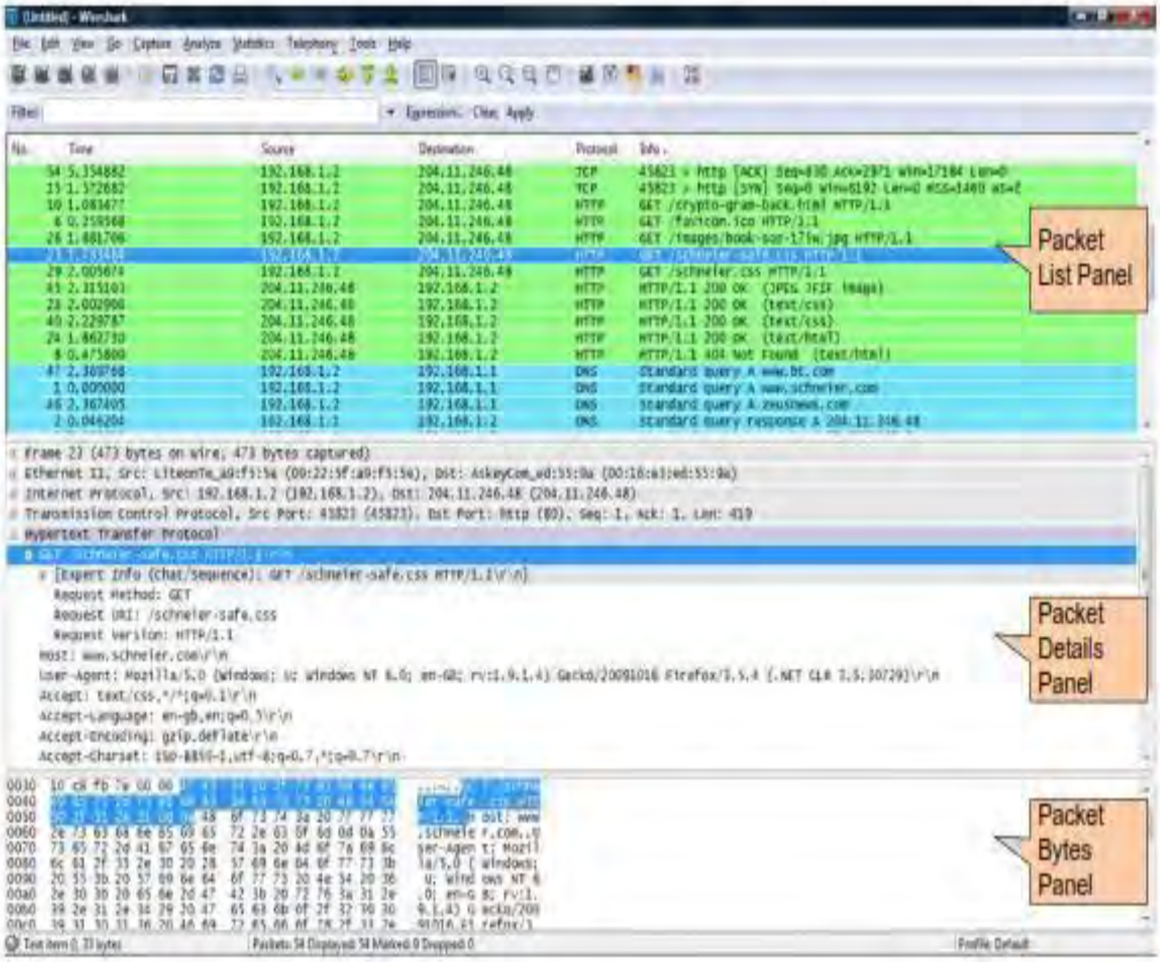
Figure 18: Wireshark displaying the hardware and software interfaces of the network



	Device	Description	IP	Packets	Packets/s	
□	Ethernet	Realtek Ethernet Controller	fe80::f126:5290:a3b5:5262	298	12	Details
□	Wi-Fi	Microsoft	fe80::e5b0:b13d:cfe0:9e	0	0	Details
□	Local Area Connection* 3	Microsoft	fe80::20e2:b630:aaf8:625b	0	0	Details
□	Local Area Connection* 2	Microsoft	fe80::934:827d:d570:5fb	0	0	Details

Figure 19: Wireshark Interfaces Windows

When the interface is selected, a Wireshark display window pops up as shown in figure 20 below. In order to capture the network traffic running through the network lines, the start button must be selected by clicking on the Start button for the interface which needed to be captured in the traffic. Windows consist of several simulated crossing point which comes before the Ethernet Network Interface Card (Chen, Cheng, & Hsieh, 2009). In the column to the left of the Start button, the entire inward packs for every crossing point are shown.


(Untitled) - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter:
Expression... Clear Apply
Packet List Panel
Packet Details Panel
Packet Bytes Panel

```
No.  Time        Source          Destination     Protocol  Info
 54  5.354882    192.168.1.2     204.11.246.48   TCP       45823 > http [ACK] Seq=830 Ack=2971 Win=17184 Len=0
 15  1.572682    192.168.1.2     204.11.246.48   TCP       45823 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
 10  1.083477    192.168.1.2     204.11.246.48   HTTP      GET /crypto-gram-back.html HTTP/1.1
  6  0.259568    192.168.1.2     204.11.246.48   HTTP      GET /favicon.ico HTTP/1.1
 26  1.881706    192.168.1.2     204.11.246.48   HTTP      GET /images/book-sor-17lw.jpg HTTP/1.1
 23  [illegible] 192.168.1.2     204.11.246.48   HTTP      GET /schneier-safe.css HTTP/1.1
 29  2.005674    192.168.1.2     204.11.246.48   HTTP      GET /schneier.css HTTP/1.1
 45  2.315103    204.11.246.48   192.168.1.2     HTTP      HTTP/1.1 200 OK  (JPEG JFIF image)
 23  2.002906    204.11.246.48   192.168.1.2     HTTP      HTTP/1.1 200 OK  (text/css)
 40  2.229787    204.11.246.48   192.168.1.2     HTTP      HTTP/1.1 200 OK  (text/css)
 24  1.862730    204.11.246.48   192.168.1.2     HTTP      HTTP/1.1 200 OK  (text/html)
  8  0.475800    204.11.246.48   192.168.1.2     HTTP      HTTP/1.1 404 Not Found  (text/html)
 47  2.369768    192.168.1.2     192.168.1.1     DNS       Standard query A www.bt.com
  1  0.000000    192.168.1.2     192.168.1.1     DNS       Standard query A www.schneier.com
 46  2.367405    192.168.1.2     192.168.1.1     DNS       Standard query A zeusnews.com
  2  0.044204    192.168.1.1     192.168.1.2     DNS       Standard query response A 204.11.246.48
```

```
⊞ Frame 23 (473 bytes on wire, 473 bytes captured)
⊞ Ethernet II, Src: LiteonTe_a9:f5:5a (00:22:5f:a9:f5:5a), Dst: AskeyCom_ed:55:9a (00:16:e3:ed:55:9a)
⊞ Internet Protocol, Src: 192.168.1.2 (192.168.1.2), Dst: 204.11.246.48 (204.11.246.48)
⊞ Transmission Control Protocol, Src Port: 45823 (45823), Dst Port: http (80), Seq: 1, Ack: 1, Len: 419
⊟ Hypertext Transfer Protocol
  ⊟ GET /schneier-safe.css HTTP/1.1\r\n
    ⊞ [Expert Info (Chat/Sequence): GET /schneier-safe.css HTTP/1.1\r\n]
      Request Method: GET
      Request URI: /schneier-safe.css
      Request Version: HTTP/1.1
    Host: www.schneier.com\r\n
    User-Agent: Mozilla/5.0 (Windows; U; Windows NT 6.0; en-GB; rv:1.9.1.4) Gecko/20091016 Firefox/3.5.4 (.NET CLR 3.5.30729)\r\n
    Accept: text/css,*/*;q=0.1\r\n
    Accept-Language: en-gb,en;q=0.5\r\n
    Accept-Encoding: gzip,deflate\r\n
    Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7\r\n
```

```
0030  10 c8 fb 7e 00 00 [illegible]                                ...... [illegible]
0040  [illegible]                                                 [illegible]
0050  [illegible] 48  6f 73 74 3a 20 77 77 77                     [illegible] Host: www
0060  2e 73 63 68 6e 65 69 65  72 2e 63 6f 6d 0d 0a 55            .schneie r.com..U
0070  73 65 72 2d 41 67 65 6e  74 3a 20 4d 6f 7a 69 6c            ser-Agen t: Mozil
0080  6c 61 2f 35 2e 30 20 28  57 69 6e 64 6f 77 73 3b            la/5.0 ( Windows;
0090  20 55 3b 20 57 69 6e 64  6f 77 73 20 4e 54 20 36             U; Wind ows NT 6
00a0  2e 30 3b 20 65 6e 2d 47  42 3b 20 72 76 3a 31 2e            .0; en-G B; rv:1.
00b0  39 2e 31 2e 34 29 20 47  65 63 6b 6f 2f 32 30 30            9.1.4) G ecko/200
00c0  39 31 30 31 36 20 46 69  72 65 66 6f 78 2f 33 2e            91016 Fi refox/3.
```

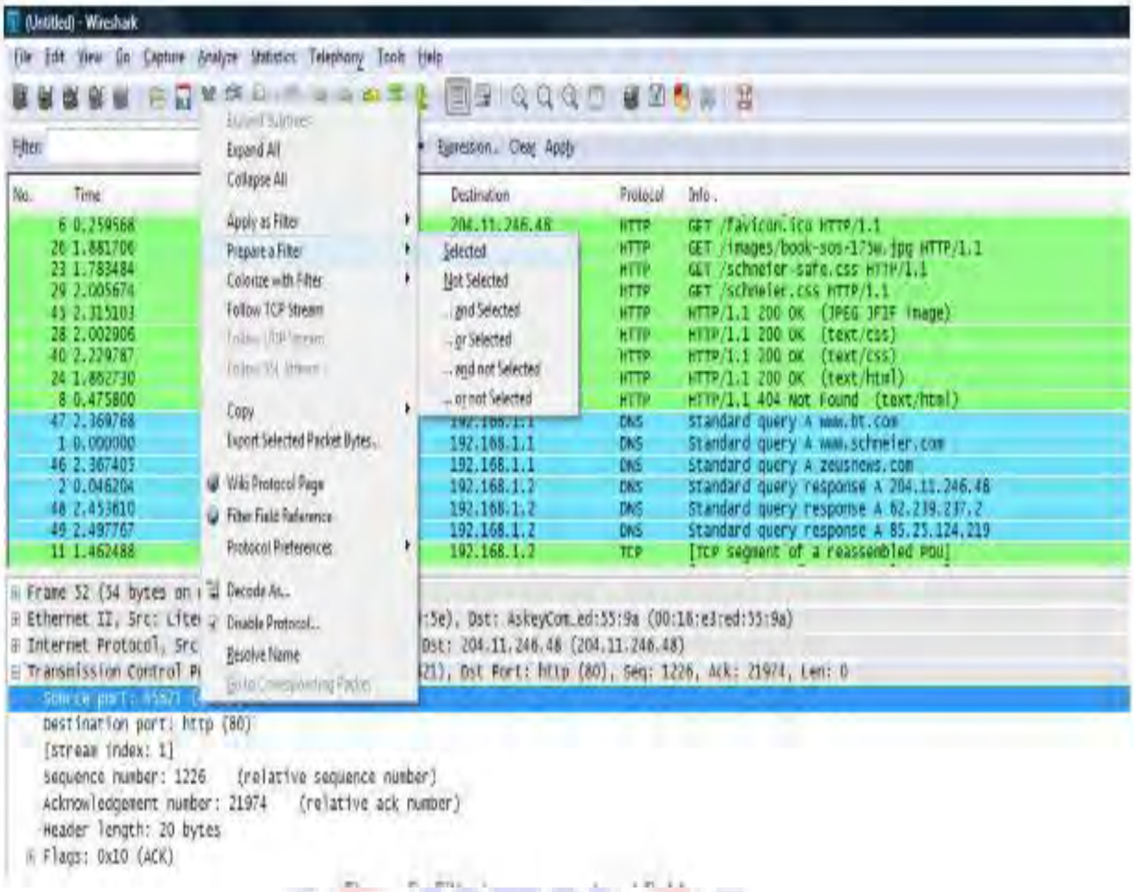
Text item (), 33 bytes | Packets: 54 Displayed: 54 Marked: 0 Dropped: 0 | Profile: Default

Figure 20: Wireshark Capturing Traffic

In figure 20, the next option to do is to produce certain system stream of traffic using a network browser like google chrome. After executing Wireshark, it captures the various packets that have created traffic in the network.

4.2.3 Displaying Filters

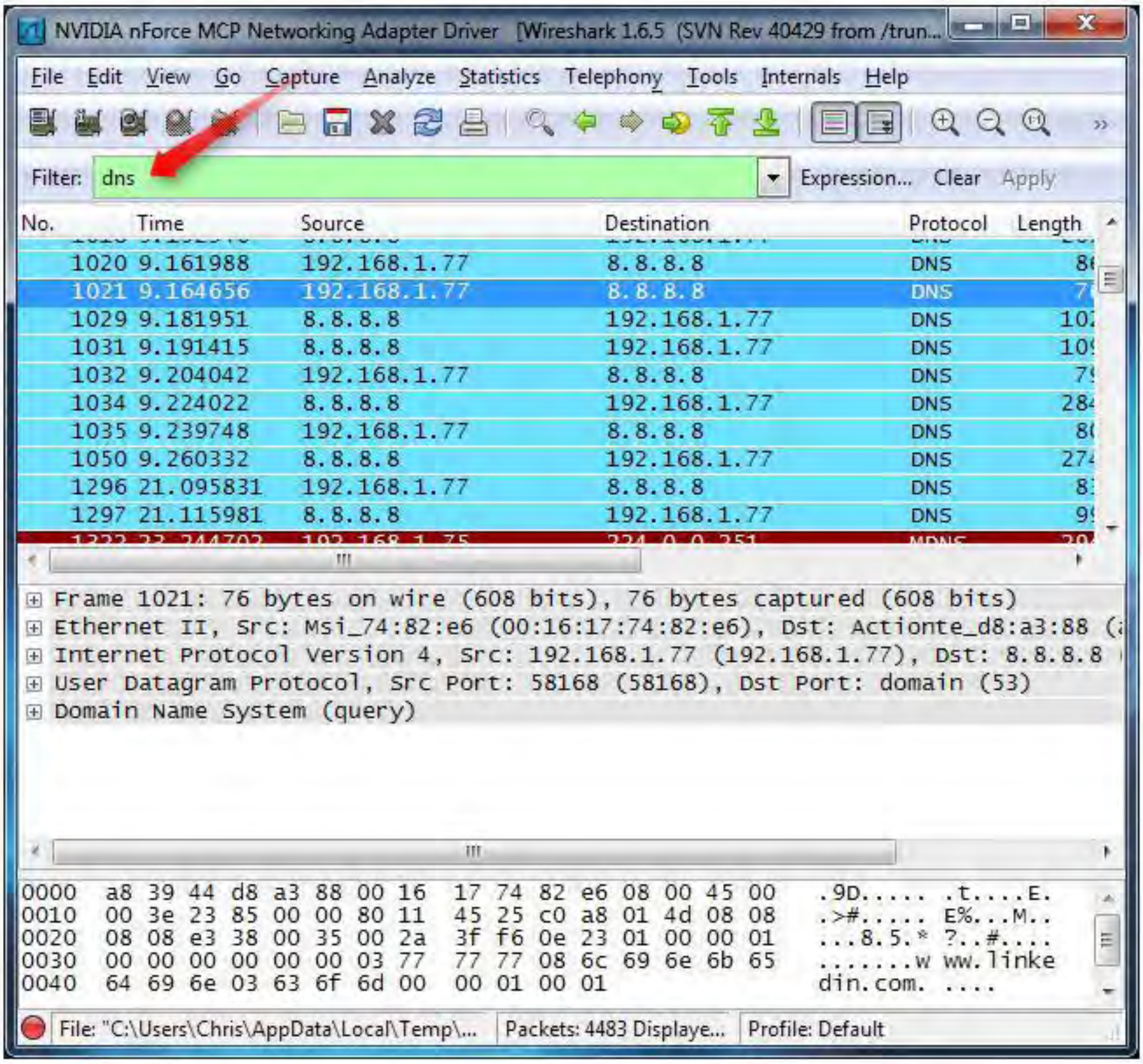
When the Wireshark application is run, the Source Port field should be select and then right click on it in the Packet Detail Panel. Prepare a Filter option should be selected then Wireshark automatically generates a display filter which is applied to the capture**.** The filter is then shown in the Filter Bar. Only the captured packets with a source port value should be selected and displayed.



```
No.   Time                 Destination      Protocol  Info.
   6 0.259568              204.11.246.48    HTTP      GET /favicon.ico HTTP/1.1
  20 1.881700                               HTTP      GET /images/book-sos-170w.jpg HTTP/1.1
  23 1.783484                               HTTP      GET /schneier-safe.css HTTP/1.1
  29 2.005674                               HTTP      GET /schneier.css HTTP/1.1
  43 2.315103                               HTTP      HTTP/1.1 200 OK  (JPEG JFIF image)
  28 2.002906                               HTTP      HTTP/1.1 200 OK  (text/css)
  40 2.229787                               HTTP      HTTP/1.1 200 OK  (text/css)
  24 1.862730                               HTTP      HTTP/1.1 200 OK  (text/html)
   8 0.475800                               HTTP      HTTP/1.1 404 Not Found  (text/html)
  47 2.369768              192.168.1.1      DNS       Standard query A www.bt.com
   1 0.000000              192.168.1.1      DNS       Standard query A www.schneier.com
  46 2.367403              192.168.1.1      DNS       Standard query A zeusnews.com
   2 0.046204              192.168.1.2      DNS       Standard query response A 204.11.246.48
  48 2.453610              192.168.1.2      DNS       Standard query response A 62.239.237.2
  49 2.497767              192.168.1.2      DNS       Standard query response A 85.25.124.219
  11 1.462488              192.168.1.2      TCP       [TCP segment of a reassembled PDU]

⊞ Frame 52 (54 bytes on
⊞ Ethernet II, Src: Lite          :5e), Dst: AskeyCom_ed:55:9a (00:18:e3:ed:55:9a)
⊞ Internet Protocol, Src          Dst: 204.11.246.48 (204.11.246.48)
⊟ Transmission Control Pr         21), Dst Port: http (80), Seq: 1226, Ack: 21974, Len: 0
    Source port: 45621 (
    Destination port: http (80)
    [Stream index: 1]
    Sequence number: 1226    (relative sequence number)
    Acknowledgement number: 21974    (relative ack number)
    Header length: 20 bytes
  ⊞ Flags: 0x10 (ACK)
```

Figure 21: Filtering on a protocol field

Another method of applying filter is shown in figure 22. It is actually the most basic way of applying filters. This is done by typing it into the filter box at the top of the current display window and hitting the Enter key of the keyboard. For example, type DNS you will only see DNS packets. As soon as you start typing Wireshark automatically complete your filter (Du, Yang, & Kang, 2008).


NVIDIA nForce MCP Networking Adapter Driver [Wireshark 1.6.5 (SVN Rev 40429 from /trun...
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Expression... Clear Apply

Filter: dns

```
No.   Time        Source         Destination    Protocol  Length
 1020 9.161988    192.168.1.77   8.8.8.8        DNS       8
 1021 9.164656    192.168.1.77   8.8.8.8        DNS       7
 1029 9.181951    8.8.8.8        192.168.1.77   DNS       10
 1031 9.191415    8.8.8.8        192.168.1.77   DNS       10
 1032 9.204042    192.168.1.77   8.8.8.8        DNS       7
 1034 9.224022    8.8.8.8        192.168.1.77   DNS       28
 1035 9.239748    192.168.1.77   8.8.8.8        DNS       8
 1050 9.260332    8.8.8.8        192.168.1.77   DNS       27
 1296 21.095831   192.168.1.77   8.8.8.8        DNS       8
 1297 21.115981   8.8.8.8        192.168.1.77   DNS       9
 1322 22.244702   192.168.1.75   224.0.0.251    MDNS      20
```

```
⊞ Frame 1021: 76 bytes on wire (608 bits), 76 bytes captured (608 bits)
⊞ Ethernet II, Src: Msi_74:82:e6 (00:16:17:74:82:e6), Dst: Actionte_d8:a3:88 (
⊞ Internet Protocol Version 4, Src: 192.168.1.77 (192.168.1.77), Dst: 8.8.8.8
⊞ User Datagram Protocol, Src Port: 58168 (58168), Dst Port: domain (53)
⊞ Domain Name System (query)
```

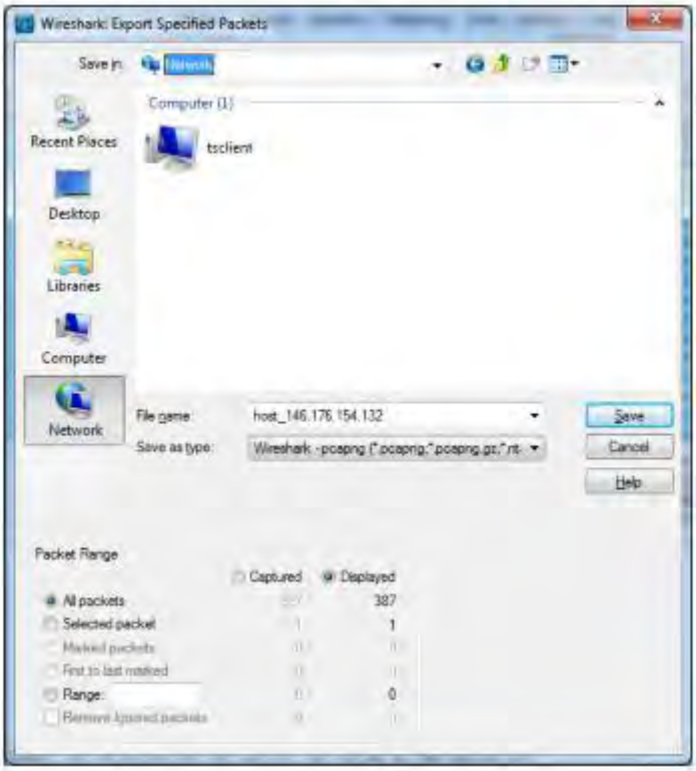
```
0000  a8 39 44 d8 a3 88 00 16  17 74 82 e6 08 00 45 00   .9D..... .t....E.
0010  00 3e 23 85 00 00 80 11  45 25 c0 a8 01 4d 08 08   .>#..... E%...M..
0020  08 08 e3 38 00 35 00 2a  3f f6 0e 23 01 00 00 01   ...8.5.* ?..#....
0030  00 00 00 00 00 00 03 77  77 77 08 6c 69 6e 6b 65   .......w ww.linke
0040  64 69 6e 03 63 6f 6d 00  00 01 00 01               din.com. ....
```

File: "C:\Users\Chris\AppData\Local\Temp\... Packets: 4483 Displaye... Profile: Default

Figure 22: Wireshark displaying basic filtering option

4.2.4 Saving Captured Packets

More often than not captures are supposed to be kept to CD. Saving a captured packet, the **File** menu must be selected followed by **SaveAs** and keep record of the trace. Automatically, the Wireshark create a **pcapng** file which can be read and write. For instance, a tcpdump output file can be read into Wireshark for analysis. This records all the captured packets into the file.



Wireshark: Export Specified Packets

Save in: Network

Computer (1)

tsclient

File name: host_146.176.154.132

Save as type: Wireshark - pcapng (*.pcapng;*.pcapng.gz;*.nt-

Packet Range

	○ Captured	● Displayed
● All packets	387	387
○ Selected packet	1	1
○ Marked packets	0	0
○ First to last marked	0	0
○ Range:	0	0
□ Remove Ignored packets	0	0

Figure 23: File save option

To save only the displayed packet, the File menu must be selected followed by export specified packets. The **radio** button must be selected instead of the **Captured** option button. This creates a **Pcap** file with only the filtered packets displaying the current filter.

4.3 Packets Analysis

Table 5: Packets traversing in same subnet within the same WAN

Source IP Address	Destination IP Address	Packet Length	Hop Count	Packet Type	Identity Number
192.168.1.69	192.168.1.77	108	78	TCP	1025
192.168.1.70	192.168.1.77	108	78	TCP	2452
192.168.1.71	192.168.1.77	108	78	TCP	2345
192.168.1.72	192.168.1.77	108	78	TCP	2673
192.168.1.73	192.168.1.77	108	78	TCP	2579
8.8.8.1	8.8.8.8	64	78	UDP	840
8.8.8.2	8.8.8.2	64	78	UDP	859

From table 5, each packet has an identified number which helps in fragmentation and defragmentation of the network system. The variation number means that they have a packet of their own. From the TCP/IP model there are different packet types that helps in data transmission during capture. The packets shown in table 6 were in transit.

Table 6: Packet type and number captured

Packet Type	Number Captured
UDP	35
TCP	89
ICMP	21
Total	**145**

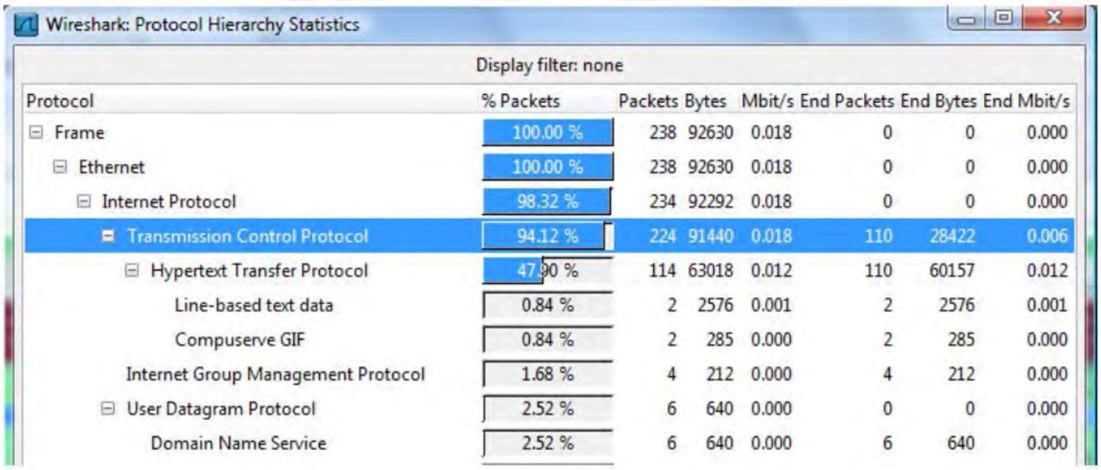
From table 6 a total of 145 packets were captured. Out of this 35 were UDP, 21 ICMP 21 and 89 were also TCP. The variation in numbers is as a result of what is being transmitted from the source. This means that, packet can only be captured when they are traversing in the network on top of an installed application running on the network system.

Table 7: Packet Size and time taken

Packet Size	Time
64	33
56	36
45	42
54	34
67	31
78	27

From table 7, there are variations in the arrival time of the packets which are based on the factors like speed of the link, TCP windows size, hop count and size of the packet itself.

4.4 Packets Statistics



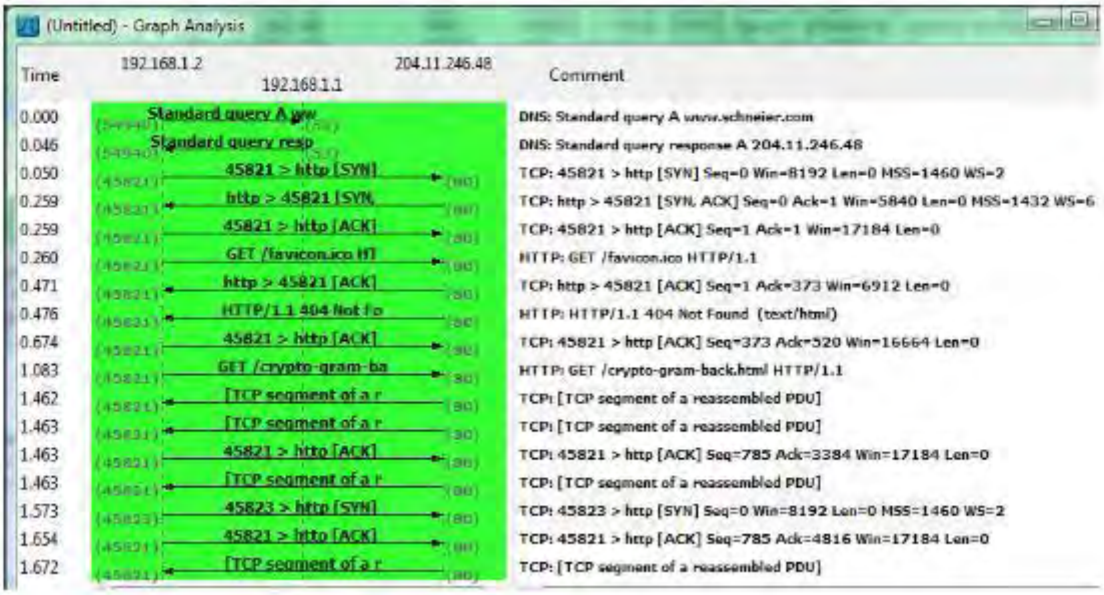
Wireshark: Protocol Hierarchy Statistics

Display filter: none

Protocol	% Packets	Packets	Bytes	Mbit/s	End Packets	End Bytes	End Mbit/s
⊟ Frame	100.00 %	238	92630	0.018	0	0	0.000
⊟ Ethernet	100.00 %	238	92630	0.018	0	0	0.000
⊟ Internet Protocol	98.32 %	234	92292	0.018	0	0	0.000
⊟ Transmission Control Protocol	94.12 %	224	91440	0.018	110	28422	0.006
⊟ Hypertext Transfer Protocol	47.90 %	114	63018	0.012	110	60157	0.012
Line-based text data	0.84 %	2	2576	0.001	2	2576	0.001
Compuserve GIF	0.84 %	2	285	0.000	2	285	0.000
Internet Group Management Protocol	1.68 %	4	212	0.000	4	212	0.000
⊟ User Datagram Protocol	2.52 %	6	640	0.000	0	0	0.000
Domain Name Service	2.52 %	6	640	0.000	6	640	0.000

Figure 24: Displaying Wireshark protocol hierarchy statistics

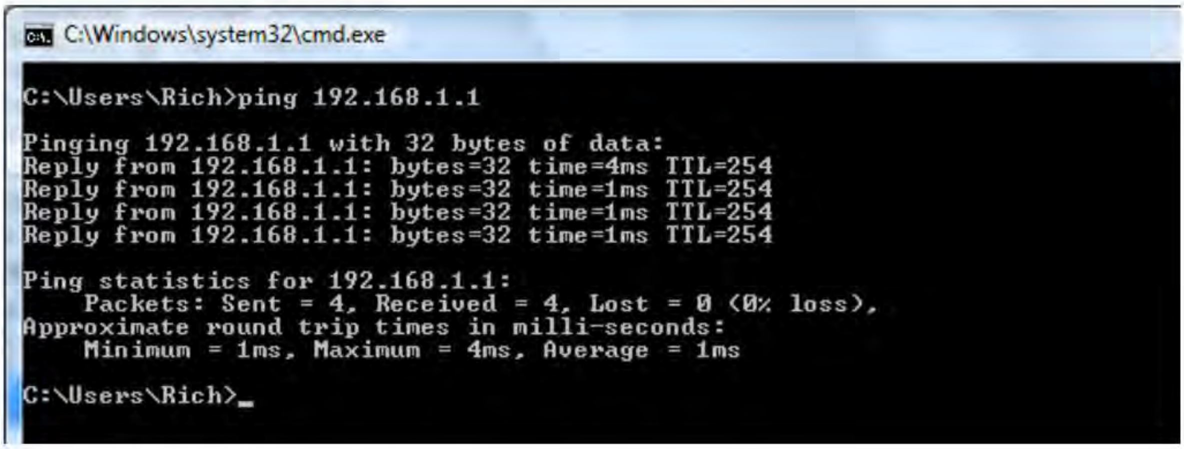
Figure 25 shows the statistics of number of packets, number of bytes running minute per second, the End packets, End Bytes and minute per second that is supposed to run on the statistics windows. It is activated by selecting the Statistics menu followed by the protocol hierarchy.



Time	Comment
0.000	DNS: Standard query A www.schneier.com
0.046	DNS: Standard query response A 204.11.246.48
0.050	TCP: 45821 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
0.259	TCP: http > 45821 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1432 WS=6
0.259	TCP: 45821 > http [ACK] Seq=1 Ack=1 Win=17184 Len=0
0.260	HTTP: GET /favicon.ico HTTP/1.1
0.471	TCP: http > 45821 [ACK] Seq=1 Ack=373 Win=6912 Len=0
0.476	HTTP: HTTP/1.1 404 Not Found (text/html)
0.674	TCP: 45821 > http [ACK] Seq=373 Ack=520 Win=16664 Len=0
1.083	HTTP: GET /crypto-gram-back.html HTTP/1.1
1.462	TCP: [TCP segment of a reassembled PDU]
1.463	TCP: [TCP segment of a reassembled PDU]
1.463	TCP: 45821 > http [ACK] Seq=785 Ack=3384 Win=17184 Len=0
1.463	TCP: [TCP segment of a reassembled PDU]
1.573	TCP: 45823 > http [SYN] Seq=0 Win=8192 Len=0 MSS=1460 WS=2
1.654	TCP: 45821 > http [ACK] Seq=785 Ack=4816 Win=17184 Len=0
1.672	TCP: [TCP segment of a reassembled PDU]

Figure 25: Capturing of the ARP and ICMP Protocol Traffic

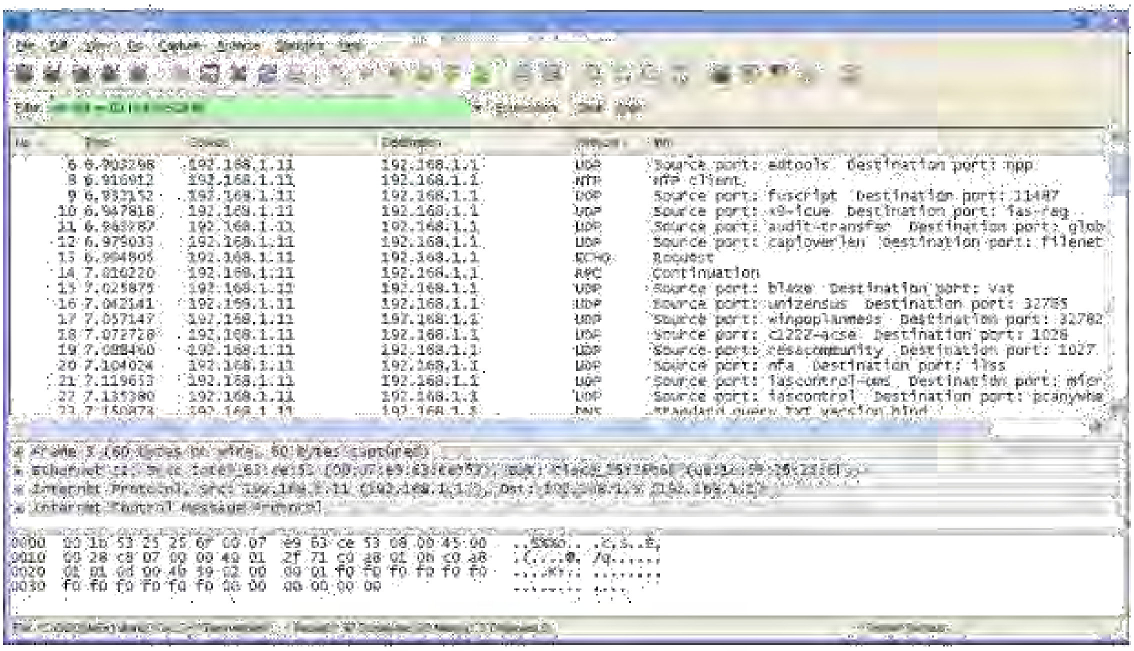
Figure 25 displays the graph flows of traffic flowing through the network. It is done by selecting the Statistics menu and then Flow Graph from the menu option. Select the General Flow and then Network Source options, and finally clicking on Ok button.



```
C:\Users\Rich>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time=4ms TTL=254
Reply from 192.168.1.1: bytes=32 time=1ms TTL=254
Reply from 192.168.1.1: bytes=32 time=1ms TTL=254
Reply from 192.168.1.1: bytes=32 time=1ms TTL=254

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 1ms, Maximum = 4ms, Average = 1ms

C:\Users\Rich>_
```

Figure 26: Windows Console of Wireshark

When a Wireshark capture is started, a window console should be opened by pressing windows key+R and typing cmd in the context menu. The ICMP traffic can be generated by using the ***ping*** command to check the connectivity of the neighboring machine.



```
No.  Time        Source         Destination   Protocol  Info
   6 6.903298    192.168.1.11   192.168.1.1   UDP       Source port: edtools  Destination port: npp
   8 6.916012    192.168.1.11   192.168.1.1   NTP       NTP client
   9 6.932152    192.168.1.11   192.168.1.1   UDP       Source port: fuscript  Destination port: 11487
  10 6.947818    192.168.1.11   192.168.1.1   UDP       Source port: x9-icue  Destination port: fas-reg
  11 6.963787    192.168.1.11   192.168.1.1   UDP       Source port: audit-transfer  Destination port: glob
  12 6.979033    192.168.1.11   192.168.1.1   UDP       Source port: capioverlan  Destination port: filenet
  13 6.994805    192.168.1.11   192.168.1.1   ECHO      Request
  14 7.010220    192.168.1.11   192.168.1.1   RPC       Continuation
  15 7.025875    192.168.1.11   192.168.1.1   UDP       Source port: blaze  Destination port: vat
  16 7.042141    192.168.1.11   192.168.1.1   UDP       Source port: unizensus  Destination port: 32785
  17 7.057147    192.168.1.11   192.168.1.1   UDP       Source port: winpoplanmess  Destination port: 32782
  18 7.072728    192.168.1.11   192.168.1.1   UDP       Source port: c1222-acse  Destination port: 1028
  19 7.088460    192.168.1.11   192.168.1.1   UDP       Source port: resacommunity  Destination port: 1027
  20 7.104024    192.168.1.11   192.168.1.1   UDP       Source port: nfa  Destination port: iiss
  21 7.119653    192.168.1.11   192.168.1.1   UDP       Source port: iascontrol-oms  Destination port: micr
  22 7.135380    192.168.1.11   192.168.1.1   UDP       Source port: iascontrol  Destination port: pcanywhe
  23 7.150873    192.168.1.11   192.168.1.1   DNS       Standard query TXT version.bind

⊞ Frame 3 (60 bytes on wire, 60 bytes captured)
⊞ Ethernet II, Src: Intel_63:ce:53 (00:07:e9:63:ce:53), Dst: Cisco_25:25:6f (00:1b:53:25:25:6f)
⊞ Internet Protocol, Src: 192.168.1.11 (192.168.1.11), Dst: 192.168.1.1 (192.168.1.1)
⊞ Internet Control Message Protocol

0000  00 1b 53 25 25 6f 00 07  e9 63 ce 53 08 00 45 00   ..S%%o.. .c.S..E.
0010  00 28 c8 07 00 00 40 01  2f 71 c0 a8 01 0b c0 a8   .(....@. /q......
0020  01 01 0d 00 40 59 02 00  00 01 f0 f0 f0 f0 f0 f0   ....@Y.. ........
0030  f0 f0 f0 f0 f0 f0 00 00  00 00 00 00               ........ ....
```

Figure 27: Displaying results after the traffic analysis

After the traffic analysis had been done the Wireshark shows results obtained from the scan done which shows that the data traffic has been decongested due to the analysis done on the network.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter constitutes an introduction, Monitoring Network host' or users' Movement on the Network, Discussion, Challenges of the System, Recommendation, and Suggestion for future research.

5.2 Summary

The study is design to use Wireshark application to monitor and analyze network traffic that has developed as a result of users' activities, malicious attack, and intruders' operations. The researcher made use of a router-based approach and packet analysis to solve the congested network problems generated in the network. This would help system administrators to identify or make meaning of the packet coming into the network at time and what the effect of the additional traffic being generated in the network. Secondly, the study would help the administrator to monitor activities of clients and communicate issues relating to the monitoring and analyzing packets in order to respond to client problems relating to network traffic causing slow data movement. However, for the purpose of network card and driver status verification, the network card, and the driver status verifier were included in the traffic analysis before beginning the packet capturing.

The research was also aimed at monitoring activities and movement of hosts or users on the network at a particular point in time which has an adverse effect on the performance of the network as a result of the additional traffic being generated in the network environment be it WAN, LAN or internet. Based on the result of the congested network traffic analyzed (Sciancalepore et al., 2019), users search for information on the network can affect the performance of the network. In order to maintain undisturbed links in the network, activities of users of the network must be monitored to find out what they do with their personal computers that is connected to the organization's network. The number of routers that users traverse or cross also affects the performance of the network as a result of transmission delay; in addition, packet size also affects the TCP throughput.

5.2.1 Monitoring Network host' or users' Movement on the Network

When packets are captured using the Wireshark, the application indicates the date the activity was carried out on the subnet those packets traverses therefore if the source is known an inference can be made as whether at that time the use was supposed to work or not so that date and timing can be monitored. From the captured results, the sources and IP addresses inform us where the packets are being generated from (sources) as well as where it is going. This indicates that by virtue of knowing the source and destination of the IP address, the administrator can easily identify the user of that particular personal computer.

5.2.2 Effects of User's Activities on the Bandwidth Consumption, Delay, and Congestion Affected by Packet Parameters

During the design stage bandwidth delay, TCP measures were included to facilitate the effect of users' movement. When users' activities on the network affect the network throughput, it causes delay in packets traveling through the network. The delay is based on the packet length. The packet length shows information regarding packet bytes which indicate the transmission and processing of information if the maximum transfer unit is less than the packet size. The bigger the size, the higher the transmission and queuing delay based on speed on link and buffer size of the routers and nodes. This delay is proportional to the packets length in bit. As more packet traverses to reach its destination, the greater the queuing, the higher the transmission delay is likely to occur. Queuing delay is a function of maximum segment size. This means that as users download bigger packets sizes there will be delays in the network which is caused by so many packet (congesting) tying to traverse at the same time.

5.2.3 How network administrators determine whether the network interfaces on the server is up and running

The monitoring of data flow on the network through the use of the Wireshark application in managing the network system is done through different means till date which means that checking the remote communication between network hosts should always be checked for economic purposes. The study used the Wireshark application as a network traffic analyzer and a network monitoring tool which helped in the network management using packet analysis to make the work of network administrators easy with the goal to increase speed and efficiency in the network system. The Wireshark application tool is easily understandable in their use and can act as a guide to network administrators. The main concept of this system is that it allows administrators to access information remotely in whatever environment they find themselves which also means that it will not be necessary for them to go from one environment to the other in order to access and collect information which is available on his (administrators') machine. This reduces the time taken by the administrator to go round, reduces stress, and increases flexibility thereby making information readily available as and when needed. More importantly when there are many users on the network with so many several activities going on the network system both inside (internet) or outside (subnet) which adds additional traffic. Both on the graph analysis and table presented, administrators should be able to monitor activities of network users and give account of the network used.

5.3 Conclusion

The researcher encountered some limitations in the study which was that; the packets captured and used for the study was limited to UEW-Kumasi network even though the study also made use of the internet environment as well. This means that it can be used in every network environment for managerial purposes once the test or the experiment was done in a live environment. The study faced some challenges in the course of the study which were getting access to the Application Packet Interface for the capturing of the packets, was a problem and wrong clashing of the application on the internet. However, in building the applications to capture packets on the network requires or involves downloading windows packet library (Wincap) whose process of downloading was always interrupted as a result of delays in the network. Since the UEW-Kumasi network system was very large, the analysis of the network system took several days before it was completed.

5.4 Recommendations

Since the Wireshark application tool can monitor and analyze congested traffic in the network due to the user's various activities of users on the Local Area Network (LAN) using hop count and IP address as well as the bandwidth consumption peculiar to a particular user it is recommended that this method should be adopted. The monitoring and analysis system adopted for this study has the capability of being used by students and novice users' network diagnosis and analysis.

5.5 Suggestion for Future Research

The packet analysis of the threat where the contents of the packet should be used in identifying and preventing hackers from penetrating into the network. However, enhancing the analysis details of the packets on the senders' side should be included in the study. Additionally, it is hope that real time analysis will be used as packets are being captured. Finally, as a result of the introduction of USB modems technologies into the system, it is hope that packets travelling along that communication medium would be securely monitored. Network cards and driver status verification should be automated.

REFERENCES

Anil, R. N. V, & Shina, K. (2020). Darknet Traffic Analysis and Classification Using Numerical AGM and Mean Shift Clustering Algorithm. *SN Computer Science*, *1*(1), 1–10. https://doi.org/10.1007/s42979-019-0016-x

Arya, I. B., & Mishra, R. (2011). Internet Traffic Classification: *An Enhancement in Performance using Classifiers, 2,* 663-667.

Baig, M. S., M., R., & P., B. (2012). Virtual Network Computing Based Remote Desktop Access. *International Journal of Computer Science and Telecommunications, 3*, 5.

Bhoria, P., & Garg, K. (2013). Determining feature set of DOS attacks. *International Journal of Advanced Research in Computer Science and Software Engineering*, *3*(5), ISSN: 2277 128X.

Brownlee, N., & Claffy, K. C. (2002). Understanding Internet traffic streams: dragonflies and tortoises. *IEEE Communications Magazine*.

Buck, G. (2001). *TCP/IP Addressing*: Designing and Optimizing Your IP Addressing.

Caliskan, E. (2011). *Campus network topology discovery and distributed firewall policy generation.*

Calsoft (2012). Retrieved from http://www.calsoftlabs.com/whitepapers/ethernet-network.html.

Casad, E. (2004). Sams Teach Yourself TCP/IP in 24 Hours.

Chappell, L. (2012). *Wireshark Network Analysis.* Nevada: Laura Chappel University.

Chen, R.C., Cheng, K.F., & Hsieh, C.-F. (2009). Using Rough Set and Support Vector. *International Journal of Network Security & Its Applications*.

Clos, M. (2010). Retrieved from a framework for network analysis using GPUs: http://upcommons.upc.edu/pfc/bitstream/2099.1/8800/1/Thesis.pdf

Colasoft (2018). Network Analysis. Retrieved from Colasoft: https://www.colasoft.com/resources/network_analysis.php

Dabir, A., & Matrawy, A. (2007). Bottleneck Analysis of Traffic Monitoring using Wireshark. *Innovations in Information Technology*.

Degioanni, L. (2000). *Development of Architecture for Packet Capture and Network Traffic Analysis.*

Du, X., Yang, Y., & Kang, X. (2008). Research of Applying Information Entropy and Clustering Technique on Network Traffic Analysis. *New Jersey: IEEE.*

Dubey, G. P., Gupta, P. N., & Bhujade, R. K. (2011). A Novel Approach to Intrusion Detection System using Rough Set Theory and Incremental SVM. *International Journal of Soft Computing and Engineering*, 15.

Easley, D., & Kleinberg, J. (2010). *Networks, Crowds, and Markets*: Reasoning about a Highly Connected World. Cambridge: Cambridge University Press.

Eid, H. F., Darwish, A., Hassanien, A. E., & Kim, T. H. (2010). Intelligent Hybrid Anomally Network Intrusion Detection System.

Erman, J., Arlitt, M., & Mahanti, A. (2006). Traffic Classification Using Clustering Algorithms. *SIGCOMMv'06 Workshops*, 1 - 4.

Fall, K. R., & Stevens, W. R. (2011). *TC/IP Illustrated, Volume 1 - The Protocols 2nd Edition.* New Jersey: Addison Wesley.

Garcia, L. M. (2008). Programming with Libpcap - Sniffing the Network from Our Own Application. Hacking.

Garcia, M. (2010). Retrieved from www.tcpdump.org

Gopinath, T., Kumar, A. S., & Sharma, R. (2013). Performance Evaluation of TCP and UDP over Wireless Ad-hoc Networks with Varying Traffic Loads. *Gwalior, India: IEEE.*

Hartpence, B. (2011). Packet Guide to Core Network Protocols. Rochester, New York: O'Reilly.

Jeya, P. G., Ravichandran, M., & Ravichandran, C. S. (2012). Efficient Classifier for R2L and U2R Attacks. *International Journal of Computer Applications, Volume 45 – No. 21*.

Kapri, H. (2011). Network Traffic Data Analysis. Louisa State University, USA. Louisa State.

Kerai, P. (2010). Tracing VNC and RDP Protocol Artefacts on Windows Mobile and Windows Smartphone for Forensic Purpose. *In Proceedings of International Cyber Resilience Conference* (p. 58). Australia: http://ro.ecu.edu.au/icr/7.

Lahaie C. (2013). TeamViewer Forensics. Retrieved from http://www.champlain.edu /Documents /LCDI/.../Team-Viewer-Forensics.pdf

Lakhina, S., Joseph, S., & Verma, B. (2010). Feature Reduction using Principal Component Analysis for Effective Anomaly–Based Intrusion Detection on NSL-KDD. *International Journal of Engineering Science and Technology*, 1790-1799.

Lakhina, S., Joseph, S., & Verma, B. (2010). Feature reduction using Principal Component Analysis for Effective Anomaly Based Intrusion Detection on NSL-KDD. *Int. J. of Engineering Science and Technology, Vol. 2(6)*, 1790-1799.

Lucas, M. (2010). *Network Flow Analysis.* No Starch publishers, ISBN 1593272030.

Marc, S. C. (2010). *A framework for network traffic analysis using GPUs.* Barcelona: UPC.

Mashitah, G. (2003). Network Traffic Monitoring Analysis on Quality of Service.

Michalski, M. (2009). A Software and Hardware System for a Fully Functional Remote Access to Laboratory Networks. *Fifth International Conference on Networking and Services, IEEE*, 561 - 565.

Miller, P. M. (2010). TCP/IP - The Ultimate Protocol Guide Volume 2. Florida: Brown Walker Press Boka Raton.

Miller, Z., Deitrick, W., & Hu, W. (2011). Anomalous Network Packet Detection Using Data Stream Mining. *Journal of Information Security*, 2, 158-168.

Mingqiang, Z., Hui, H., & Qian, W. (2012). A Graph-based Clustering Algorithm for Anomaly Intrusion Detection. *IEEE*, 978-1-4673-0241-8.

Muhtadi, A. F., Almaarif, A., & Info, A. (2020). Analysis of Malware Impact on Network Traffic using Behavior-based Detection Technique. *International Journal of Advances in Data and Information Systems*, *1*(1), 17–25. https://doi.org/10.25008/ijadis.v1i1.14.

Muktar, A., & Kyauta, A. (2017). The Role of Guidance and Counseling Service on Academic Performance among Students of Umar Suleiman College of Education, Gashua, Yobe State, Nigeria. *Kampala International University Journal of Social Humanities*, *2*(2015), 59–66.

Morris, S. (1994). Networks and Distributed Systems Management. Boston, MA, USA: Addison Wesley Logman Publishing Co., Inc.

Myipaddressinfo. (2006). Retrieved from myipaddressinfo.com: http://www.Myipaddressinfo.com/

Odom, W., & Thomas, A. (2006). Networking basics: CCNA 1 companion.

Ohlsson, L., & Wernersson, L. (2014). A 15-Gb / s Wireless ON-OFF Keying Link. *IEEE Access*, *2*, 1307–1313. https://doi.org/10.1109/ACCESS.2014.2364638

Packets (2012). Retrieved from http://computer.howstuffworks.com/question525.htm

Paxson, V. (2004). Strategies for Sound Internet Measurement.

Postel, J. (1980). User Datagram Protocol. RFC 768.

Postel, J. B. (1981). Internet Protocol. RFC 791.

Rafiq, C. (2005). Developing TCP/IP and UDP Traffic Monitoring Tool.

Rooney, T. (2011). IP Address Management Principles. *John Wiley & Sons*.

Sally, F., & Vern, P. (2001). Difficulties in Simulating the Internet. *IEEE/ACM Transactions on Networking*.

Sanders, C. (2007). Practical Packet Analysis. Publisher No Starch Press.

Satya, & Srikanth, P. (2004). Gigabit PickPacket: A network Monitoring Tool for Gigabit Networks.

Sciancalepore, S., Ibrahim, O. A., Oligeri, G., Pietro, R. Di, Sciancalepore, S., Ibrahim, O. A., & Oligeri, G. (2019). PiNcH : an Effective , Efficient , and Robust Solution to Drone Detection via Network Traffic Analysis. *Computer Networks*, (12), 1–50. https://doi.org/10.1016/j.comnet.2019.107044

Sikos, L. F. (2020). Forensic Science International : Digital Investigation Packet analysis for network forensics : A comprehensive survey. *Forensic Science International: Digital Investigation*, *32*, 1–12. https://doi.org/10.1016/j.fsidi.2019.200892

Sivanathan, A. (2019). IoT Behavioral Monitoring via Network Traffic Analysis Arunan Sivanathan. The University of New South Wales.

Slowman, M., & Jonathan (1994). *Networks and Distributed Systems Management.* Boston, MA, USA: Addison Wesley Longman Publishing Co. Inc.

Song, W., Beshley, M., Przystupa, K., & Beshley, H. (2020). A Software Deep Packet Inspection System for. *Sensors*, *20*(3), 1–41. Retrieved from www.mdpi.com/journal/sensors

Stademeyer, J., & C. W. Omlin. (2009). Feature set reduction for automatic network intrusion detection systems (maxborn institute for nonlinear optic and short pulse).

Susila, I. M. D. (2020). Port Session Communication Analysis Using Density-Based Clustering For Host Anomaly and Risk Activity Analysis. *2020 International Conference on Smart Technology and Applications*, 0–5. https://doi.org/10.1109/ICoSTA48221.2020.1570613749.

Tian, X., Sun, Q., Hunga, X., & A, Y. M. (2008). Dynamic Online Traffic Classification using Data Stream Mining. *IEEE.*

Tierney, B. L. (2004). Self-Configuring Network Monitor a High Performance Network Engineering Proposal: Network Measurement and Analysis.

Vijayakumar, M., & Parvathi, R. (2010). Concept mining of high volume data streams in network traffic using hierarchical clustering.

Wei, X., Member, S., Dragotti, P. L., & Member, S. (2016). FRESH — FRI-Based Single-Image Super-Resolution Algorithm. *IEEE Transactions on Image Processing*, *25*(8), 3723–3735. https://doi.org/10.1109/TIP.2016.2563178

Wheaton (2016). Retrieved from Wikipedia: http://wiki.wireshark.org/CaptureSetup /Ethernet. Date Accessed: 2019, December 21

Wu, Y., Leshem, A., Member, S., & Jensen, J. R. (2015). Joint Pitch and DOA Estimation Using the ESPRIT Method. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, *23*(1), 32–45. https://doi.org/10.1109/TASLP.2014.2367817

Yu, B., & Fei, H. (2008). Performance Impact of Wireless Mesh Networks with Mining Traffic Patterns. *New Jersey: IEEE.*

Zhou, Y., Guangmin, & He, H. W. (2009). Using Graph to Detect Anomaly. *IEEE.*