An Evaluation of Network Vulnerability Assessment Tools

By

Frank Maturu Namunaba

A DISSERTATION SUBMITED IN PARTIAL FULFILMENT FOR THE AWARD OF MASTER OF SCIENCE IN DATA COMMUNICATIONS IN THE FACULTY OF COMPUTING AND INFORMATION MANAGEMENT AT KCA UNIVERSITY

20 October 2014

DEED OF DECLARATION

I declare that this dissertation is my original work and has not been previously published or submitted elsewhere for award of a degree. I also declare that this contains no material written or published by other people except where due reference is made and author duly acknowledged.

Student Name: **Frank Maturu Namunaba** Registration No. **12/02690**

Sign: ______________________________ Date: _________________________

I do hereby confirm that I have examined the master's dissertation of

Frank Maturu Namunaba

And have certified that all revisions that the dissertation panel and examiners recommended have been adequately addressed

Sign: ______________________________ Date: _________________________

Dr. Alice Njuguna

Dissertation Supervisor

ACKNOWLEDGEMENT

I give all Glory and honour to the Almighty God for the grace upon me through my life and particularly during the research period. Without you Father, I am nothing.

I most sincerely thank my supervisor Dr. Alice Njuguna for the support, guidance and patience throughout the period of the research. I cannot find appropriate words to acknowledge your contribution to my study. All I can say is that may The Lord bless you abundantly as you continue empowering others who may come to you after me.

To the KCA University Faculty of Computing & Information Management, Graduate School and all the lecturers who contributed in one way or another, I thank you all.

I owe much gratitude to my fellow students for their encouragement and comments during the research. Samuel Masika, your encouragement can not go unappreciated. May God bless you all abundantly!

DEDICATION

I dedicated this work to my family because of their encouragement and support. To My Wife Mepo; Daughters Agnes and Abygael; Sons Thomas and Timothy; Your patience and tolerance played a big role in encouraging me to complete this work. Abby, at only age 11 you understood how important this work was and kept checking on progress and cheering me on. Your interest in the work kept me going.

To my parents Luka Namunaba and Agnes Nabwala, I wish you were here to share with me in this glory. May The Almighty Lord rest your souls in eternal peace!

ABSTRACT

The evolution of communication networks has resulted into complex interconnection of devices in wired/wireless networks. While these networks have many benefits, there are security concerns for private and public networks because of untrusted networks and malicious individuals.

The Network Security Administrator is concerned with the security and safety of networks in to prevent and mitigate malicious attacks and network security breaches. There are many commercial and free automated tools that can be used to ensure private/public networks are secure. There is however no framework that can be used to select appropriate tools that will ensure network security. Traditional Network Security use Firewall, Network Address Translation, Virtual Private Network, Network Router or a Proxy server to defense.

This Dissertation provides a framework that will guide the Networks Security Administrator in selecting appropriate vulnerability assessment tools for Intrusion Detection to ensure conformance to standards, laws and legislation. I conduct a detailed literature review and an in depth examination of automated tools and use the findings to develop the framework. An attack can be Host Based Attack or Network Based Attack; Attacks can also be classified as Inside Looking Around attack or Outside Looking in Attack. These attacks can be discovered by Network Misuse Detection or Network Anomaly Detection.

The framework comprises 3 phases: **Planning** (Network exploration, tool identification and classification), **Tool Analysis** (examine each tool on software metrics of reliability, portability, usability, maintainability, efficiency and functionality), **Evaluation** (assign each tool a weight for every software metrics, a ranking for the suitability of the tool to the particular network and match tools that can be used to ensure security and compliance).

The findings show there is no turnkey solution to network security and no tool can singly provide sufficient assurance. Proper selection of a set of tools can result into secure networks and low cost.

Keywords: **Vulnerability Assessment tools, Intrusion Detection, Inside Looking Around Attack, Outside Looking in Attack, Network Based Attack, Host Based Attack, Network Misuse Detection, Network Anomaly Detection, Firewall, Network Address Translation, Virtual Private Network, Network Router, Proxy Server**

Table of Contents

LIST OF FIGURES

ABBREVIATIONS

ASCII – American Standard Code for Information Interchange

BID – Business Intelligence Development

CIA **-** Confidentiality, Integrity and Availability

COBIT **-** Control Objectives for Information and related Technology

CSV – Comma Separated Values,

CVE – Common Vulnerabilities and Exposure

CVSS – Common Vulnerability Scoring System

DNS – Domain Name Service

DLL – Direct Library Linker

DMZ – Demilitarised Zone

FIM – File Integrity Monitoring

FTP – File Transfer Protocol

INI – INItialisation file format

HTTP – Hyper Text Transfer Protocol

HTTPs – Secure Hyper Text Transfer Protocol

IANA – Internet Assigned Names and Addresses

IDS – Intrusion Detection System

ABBREVIATIONS

IP – Internet Protocol

MSDE – Microsoft SQL Server Desktop Engine

NIDS – Network Intrusion Detection System

NIST – National Institute of Standards and Technology

NSA - National Security Agency

NSR - File type extension primarily associated with Nessus Projects

OSVDB – Open Source Vulnerability Database

OVAL – Open Vulnerability Assessment Language

RPC – Remote Procedure Call

SANS - A private US Institute that specializes in information and cyber security training

SIF - Setup Information File

SMB – Server Message Block

SMTP – Simple Mail Transfer Protocol

SQL – Standard Query Language

URI – Uniform Resource Identifier

URL - Uniform Resource Locator

VA – Vulnerability Assessment

XML - eXtensible Markup Language

DEFINITION OF TERMS

Vulnerability is a weakness that an attacker can exploit in order to compromise the security of communication infrastructure.

Cyber Security is the process of protecting data and communication infrastructure by preventing, detecting and responding to attacks.

Exploration informs the network security administrator the connected devices through live scanning.

Enumeration helps the administrator determine what processes are running on each identified device

Assessment helps the administrator determine the vulnerability potential posed by the running processes and system configuration.

Exploitation attempts to leverage on one or more of the vulnerabilities to gain privileged access to a host and utilise this access level to exploit the host or escalate exploit to another host or entire network.

1 Introduction

Institutions, business and individuals are continually embracing communication networks as a means of exchanging electronic information. The evolution of the communication devices person to person, within an organization and globally has resulted into a complex interconnection of devices which include Computers, Personal Digital Assistants, Mobile phones etc. in wired and wireless networks.

While these interconnections have many benefits there are security concerns for private and public networks as a result of the presence of untrusted networks and malicious individuals. This concern for network security forms an important task of the day to day responsibilities of the network security administrator in order to mitigate malicious attacks and network security breaches.

The network security administrator has the task of restoring a compromised network to normal operation and implementing new strategies to prevent similar future attacks. This reactive activities can be time consuming and expensive in terms of downtime. The desirable approach for the network security administrator is to undertake proactive effort to assess and detect network security vulnerabilities in order to protect the network from attacks.

1.1 Background of the problem

In the current business world, important information is stored, accessed and transported in electronic form. It is important that the systems storing and transporting the information are secured in order to preserve the reputation and ensure prosperity of the organization.

Firewalls and network configurations are implemented alongside defensive software as a means of protecting networks against malicious attacks. Irrespective of these efforts networks are compromised as a result of increasing complexity and new strategies by the attackers.

Network security administration takes place in a dynamic landscape of more complex systems resulting into new vulnerabilities and more sophisticated tools. According to (PAULSON, 2004) tools and practices for security administration evolve in tandem with threats.

Network security management is an important task for the network administrator. Lack of efficient security measures can result into vulnerability exploitation of the network leading to denial of service, confidentiality and integrity loss. When this occurs, the effort required to restore normal operations can be colossal and lead to unwarranted destruction of organisational assets and unnecessary employment of human resource in the restoration.

Network administrators can use automated tools to proactively detect and prevent attack before it occurs (ESER, Kandogan, EBEN, & Haber, 2005). When vulnerability specifications and effects are detected preventative measures can be employed before an attack occurs. The automated tools can also be used to conduct system audits in order to ensure compliance to existing security standards. There are many ways performing vulnerability assessment with different relative costs and results.

This research aims to propose an efficient, convenient and cost effective framework for conducting network vulnerability assessment.

1.2 The Problem Statement

Cyber security threats exploit the complexity and connectivity of communication infrastructure systems to expose individuals, organisations, businesses and government to security, safety, economic and health risks. Proactive network security vulnerability assessment is an important task for the networks security administrator in order to mitigate malicious attack that may result to loss of service and/or theft/alteration of information. It is important that the right tools are used in the correct environment.

There exist many tools both commercial and freeware for the network security assessor to use in conducting the network security assessment. There are applications and strategies which network security administrators can use to detect, evade and mitigate cyber attacks (PAULSON, 2004).

The challenge is how much effort should go towards vulnerability assessment and which set of tools is suitable for the job and whether it is good enough to be trusted. Claims of certain network security techniques, assessment and monitoring tools being the best are simply based on user experience and not formal or expert evaluation of these tools and techniques. There is limited knowledge and research on methods for evaluating network security assessment tools.

1.3 Aims and objectives of the project

The purpose of this research is to evaluate network vulnerability assessment tools and propose a framework for economically implementing the correct, efficient and effective tools and techniques.

1.4 Specific Objectives

a) Identify available network security vulnerability assessment tools

b) Analyze network security vulnerability assessment technique for each identified tool

c) Propose a framework for implementing a convenient, reliable, cost effective and efficient method of conducting network security vulnerability assessment

d) Validate the proposed network vulnerability assessment framework

1.5 Significance of the study

It is the desire of every individual, business or government to build communication infrastructure with enhanced security and resilience in order to achieve efficiency while promoting security, safety, integrity, privacy and confidentiality. The ability to detect and prevent an attack before it occurs cannot be over emphasised.

Every Network Security administrator wants to achieve the assurance of safety using affordable and efficient automated tools. A lot of work has been done on developing these network assessment tools with every developer claiming that what they developed is the best. Every user claims that the tools they use are the best based on experience. This can be confusing to novice network administrators.

A framework for implementing network security vulnerability assessment tools will assist inexperienced network administrators to choose the right network security assessment tool. The framework will also help the security administrator understand the network vulnerabilities and therefore take corrective measures before an attack occurs. This can save the organisation a lot of resources with regard to downtime, loss/destruction of data and reputation that result from a compromised system. The purpose of this study is to propose a framework that will assist the network administrator select the right tools that will help to identify and detect vulnerabilities, protect network resources and quickly recover from an attack.

1.6 Scope and Limitations of the Study

It is not the aim of this study to develop network vulnerability assessment tools. The proposed framework is to be used as a tool for evaluating and selecting from the many network security assessment tools with respect to their capabilities, efficiency and weaknesses.

Due to the high cost of the commercial network security assessment tools this study is limited to evaluating freeware. It is not the intention of this study to promote or market any of the tools.

Much of the security concerns for a network can be addressed through organisational policy. This research does not at all discuss in details ICT policies except to assume that every organisation has one in place which will guide the implementation of the penetration testing and evaluation of tools for the organisation.

2 Literature Review

Computer networks are more than a group of interconnected devices. While the network is required to provide continuous connectivity for services such as email and internet access to many users it contains volumes of highly valuable data. The network system administrator therefore has to balance between Confidentiality, Integrity and Availability (CIA).

Network vulnerability assessment or simply network intrusion detection tools are an important topic of interest in the recent times due to its importance as a countermeasure to the activities of rogue users or connections on the public network (internet). Most operating systems include tools for detecting and mitigating network vulnerability. The network attacker is well aware of these tools and will endeavour to circumvent their abilities so as to launch an attack unnoticed. In addition, operating system and application software are released with bugs that the attacker identifies and exploits.

2.1 State of the art in network vulnerability assessment

Because of the complexity of networks and the CIA challenge current networks are generally vulnerable and are prone to attacks which cannot be easily anticipated due to variations in network designs and configurations (Gupta, K. K. et al, 2006).

2.1.1 Firewalls

Firewalls are network security components placed between autonomous networks by means of rules and policies. According to (Oppliger, 1997) firewalls examine incoming and outgoing traffic in order to prevent infiltration and attacks from untrusted networks. Firewalls are capable of detecting impending attacks and prevent invalid traffic in accordance to the defined rules and policies so that private networks can connect to untrusted networks without causing harm.

Firewalls are implemented in the form of hardware, software or a combination of both. Most Operating Systems include software based firewalls to protect private networks from public networks while routers that transmit data between networks have firewall components. Firewalls can also perform basic routing functions.

The main purpose of installing firewalls is to prevent private networks (intranets) which are also connected to the internet from attack. All messages into and out of the private network pass through the firewall which examines the traffic blocking those that contravene the specified rules and policies.

2.1.1.1 Application Based Firewall

While a conventional firewall merely blocks or permits traffic by examining the source and destination based on the user configuration, an application firewall is enhanced to limit access to an operating system. The application firewall prevents execution of suspect programs or DLLs (malicious code) to prevent damage from an intruder who gets past the conventional firewall into a computer, server or network.

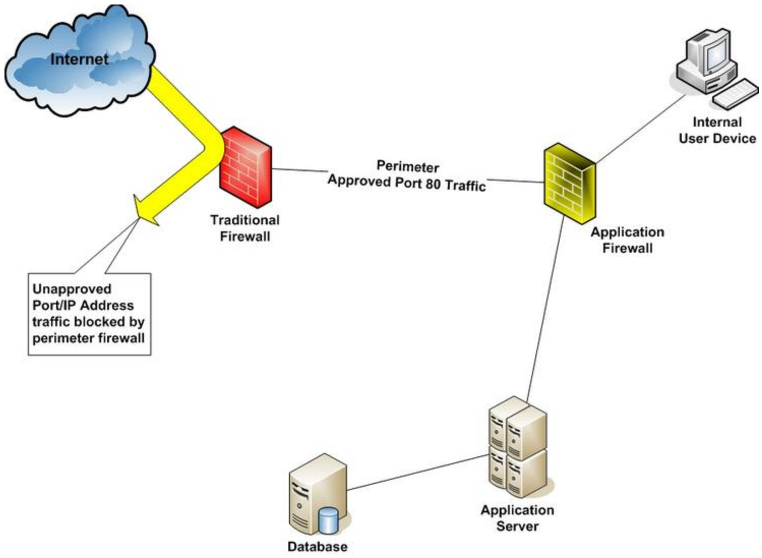


Figure 2-1: Application Based Firewall

A network based application layer firewall operates at the application layer of a protocol stack (Medina, 2003) and may be implemented by software running on a host or independent hardware node on the network. A host based application firewall monitors any input/output or service call to/from an application by examining traffic trough system calls in addition to the network stack to protect applications running only on the same host.

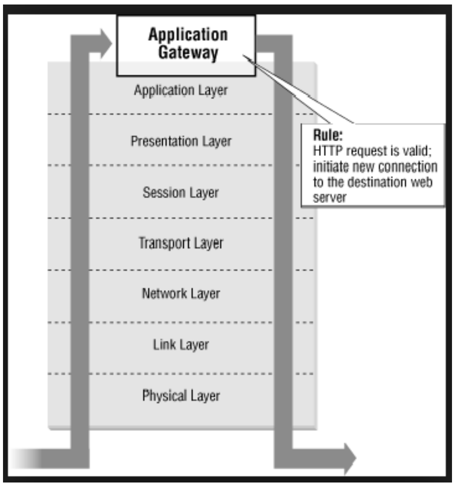


Figure 2.2: Application Level Firewall

An application level firewall has a higher level of security because it uses set rules from high level protocols, maintaining state information and examining traffic using multiple connections. However, it has a large processing overhead as a result of complex filtering that demand access control resolution and specific submission for every network application (Brian, 2010).

2.1.1.2 Stateless Packet Filter

Data traverses the network in the form of packets with each packet having a header which contains the source and destination. A packet filtering firewall only allows those network packets that are permitted in the firewall policy. It inspects the header and drops or permits traffic at the IP Layer based on the IP address and port number in the UDP or TCP header.

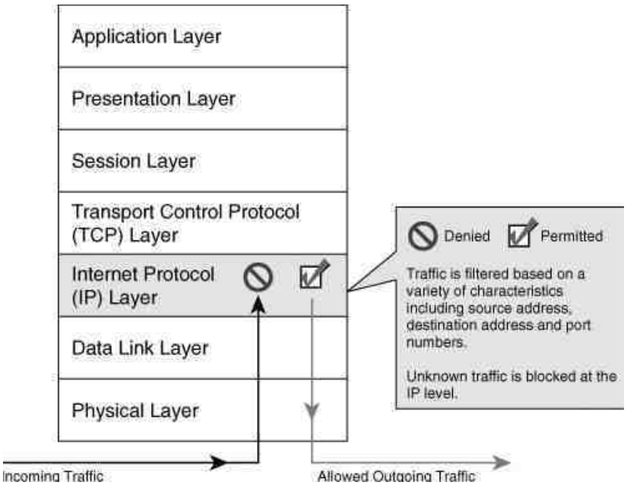

Application Layer
Presentation Layer
Session Layer
Transport Control Protocol (TCP) Layer
Internet Protocol (IP) Layer
Data Link Layer
Physical Layer
Denied
Permitted
Traffic is filtered based on a variety of characteristics including source address, destination address and port numbers.
Unknown traffic is blocked at the IP level.
Incoming Traffic
Allowed Outgoing Traffic

Figure 2.3: Stateless Packet Filter

Stateless packet filters inspect every incoming packet without attempting to predict the state based on previously received packets and accepts or denies packets depending on the set rules using the default rule which can be ‘accept all’ or ‘deny all’ if a corresponding rule is not found. Stateless packet filters can easily be compromised by hackers. However, Stateless packet filters prevent IP spoofing (Brian, 2010) and (Wes Noonan, 2010).

2.1.1.3 Stateful Packet Filter

Stateful packet filters also known as dynamic packet filters gather state information from the transport layer inspecting the header for new or existing connections (Gouda, 2008). It keeps a current connection table and permits traffic from existing connections without further inspection. According to (NIST, 2010), stateful packet filters are fast, flexible and secure because they operate at Layer 3 of the OSI Model. However stateful packet filters are susceptible to attack because they do not examine information in the upper layers.

Packet filters must be able to maintain a good log, have a good GUI, or command line language for rules and exceptions and must be carefully evaluated for proper use. Generally packet filters cannot be relied upon in isolation for network security.

2.1.1.4 Proxy Server

A proxy server is a hardware or software that acts as an intermediary between a client seeking services from a server. Today, most servers are web proxies providing anonymous connection to the www (World Wide Web). A proxy server may also be configured to be used as a firewall.

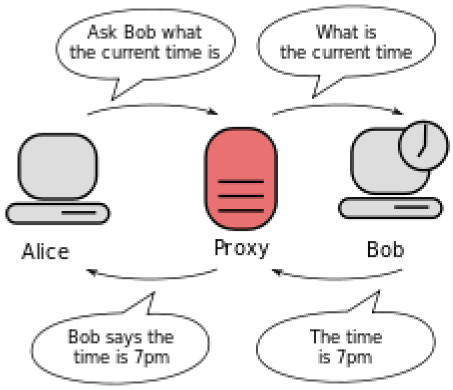


Figure 2.4: Proxy Server

A proxy server represents the entire private network as a single IP address to the internet rather than exposing the true identities of the internal users to the world. The purpose of the proxy server is to prevent unauthorised external entities from accessing internal resources and preventing internal users from accessing unauthorized external resources. A NAT (Network Address Translator) is used to facilitate communication. Proxy servers provide secure and private browsing, caching for fast access and administration optimisation of resources (Thomas, 2006).

2.1.1.5 Network Address Translation (NAT)

NAT is a method of mapping network address in the Internet Protocol Datagram packet headers while in transit on an IP network so that internal IP addresses are shielded from the public network. NAT enables a network to use non-internet routable address on the private network either through one to one address mapping or one to many address mapping.

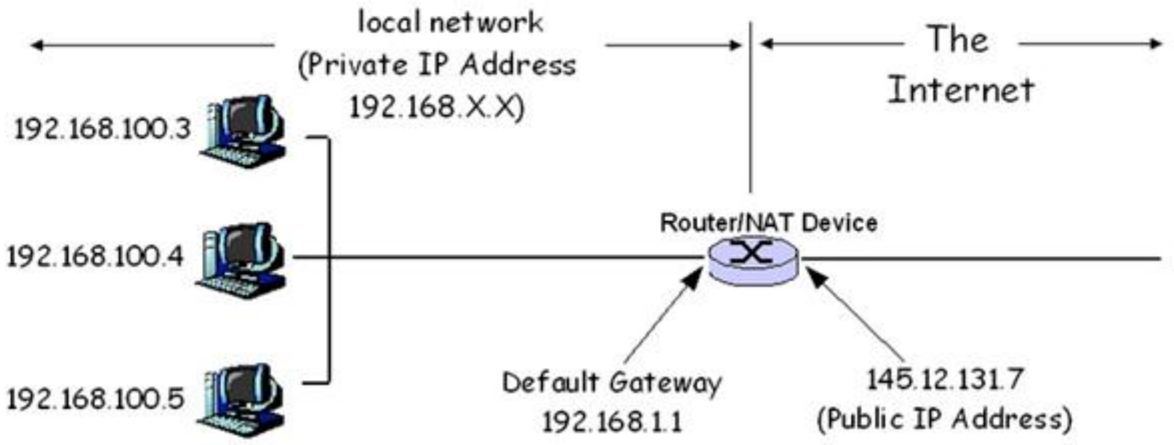


Figure 2.5: Network Address Translation

NAT empowers the firewall to control external connections by limiting inbound traffic to make it difficult for an attacker to reach the internal network. However, in a dynamic address allocation, responses may be lost or directed to wrong hosts. In embedded IP addresses, NAT may have a problem to understand the protocols in order to preserve the packet legitimacy.

When data in transit is encrypted, the integrity check to ensure that data is not tampered with may fail.

2.1.1.6 Firewalls: Benefits

According to (Gouda, 2008) firewalls protect private networks from attackers or intruders from public networks. Firewalls use an ordered set of rules to predict and decide the action to take applying boolean operations on the fields of the data packet like the source/destination IP address, Port number and protocol type.

Firewalls examine inbound and outbound network traffic against port states to prevent IP spoofing via protocol and application scouting, application level attack and identity spoofing by inspecting packet headers at application layer. Firewalls perform stateful inspection between hosts to confirm and approve connection (Brian, 2010).

According to (Sheth, 2011) firewalls operate on the 3 layers of the OSI model (Network Layer, Transport Layer and Application Layer). Firewalls inspect traffic at the data link layer through transparent firewalling (bump in the wire/stealth firewall) which connects the same network both on the inside and outside interfaces) to provide Virtual Private Network (VPN), Port Address Translation (PAT) and Network Address Translation (NAT).

Firewalls prevent universal locator and content filtering of network traffic using packet filtering servers (Brian, 2010). They enhance network performance by deleting unutilized connections, caching www requests and responses.

They provide enhanced security by support for IPv4 and Ipv6 including unicast and multicast. Unicast routing is a one to one connection between a client and a server using TCP/IP and UDP protocols while Multicast routing is a broadcast routing to all listening clients on the server. While IPv4 addresses uses a 4 byte address, the IPv6 address type uses a 16 byte address also called ‘anycast’ address (Microsoft, 2014). In addition, firewalls also provide context based security via proxy and DHCP servers (Brian, 2010).

2.1.1.7 Firewalls: Shortfalls

According to (Kobayashi, 2003) firewalls cannot provide complete protection against attacks and intrusion. Firewalls are unable to prevent interior attacks (Katkar, 2010). This is because new threats are emerging on daily basis.

Once a static rule or policy is set, the firewall cannot react to an unpredicted attack or initiate a corrective measure. Firewalls are usually configured manually. More often than not they are not properly configured due to the complex nature of networks and lack of knowledge of emerging threats. According to (Gouda, 2008) a firewall can only be as good as the expertise of the network administrator and the network environment.

According to (Wes Noonan, 2010) most firewalls do not examine contents of data packets that comprise the network traffic. Firewalls usually do not scan incoming files, emails and messages for malicious code as this can cause a performance bottleneck on the network.

According to (Sheth, 2011) internal configurations e.g. dialup for mobile users may bypass the firewall and a firewall does not prevent attack from users on the intranet or attackers who employ social engineering techniques. In addition, firewalls themselves can be compromised to achieve denial of service.

These shortcomings of the firewall make it necessary for new research, ideas, techniques and strategies for the vulnerability detection and mitigation. Intrusion detection techniques have been used in the past for perimeter security to prevent denial of service. In the recent times, there has been increasing need for techniques that will safeguard data stored in a network against malicious modification or disclosure to unauthorised persons (Dan Pei, Lixia Zhang, Dan Massey, April 2004).

In addition to the firewall, organisations use antivirus software both at the server and client end to protect networks from malicious attacks. It is an agreed fact in the computer industry today that the speed with which worms and viruses are transmitted across networks needs technologies which can in advance predict virus or worm outbreaks. Technologies that can predict worm or virus outbreaks require computing power that is beyond that of the conventional personal computer.

2.1.2 Routers

A router is a layer 3 gateway device that connects 2 or more networks. Routers operate at the network layer in the OSI model to forward data packet between networks to secure gateways to the internet.

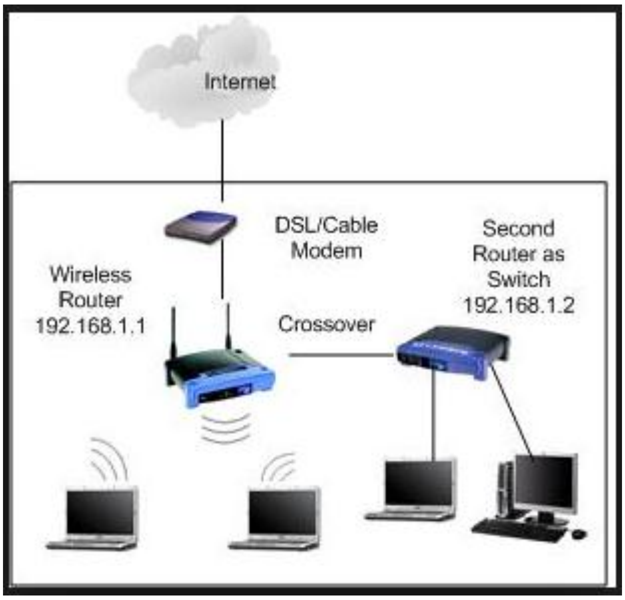

Internet
DSL/Cable Modem
Second Router as Switch 192.168.1.2
Wireless Router 192.168.1.1
Crossover

Figure 2.6: Networker Router

By maintaining a routing table, wired or wireless routers are able to filter inbound or outbound traffic based on the source or destination IP address. Routers dynamically update routing tables but some can allow administrators to manually update the routing table. Broadband routers combine the functions of a router, switch and firewall into a single device (Kozierok, 2005).

2.1.2.2 Routers: Benefits

Access to the internet requires a public IP. In the absence of a router, every computer that connects to the net will require a public IP address. A router with NAT enables clients on a network to use a single public IP address with multiple UDP ports to connect to the www. This reduces the cost of connecting to the internet and enhances security for the clients on the networks as their true identity will be concealed.

Routers maintain a static or dynamic routing table for directing traffic to destinations. In dynamic routing, the router identifies the most efficient and cost effective route between IP hosts on separate networks. This enhances the network performance and user experience.

2.1.2.3 Routers: Shortfalls

By maintaining a routing table and calculating the most economical path per traffic instance, a router adds a processing overhead to the network making it slower.

A router is also a single point of failure on the network. It is therefore necessary to maintain redundant routers to ensure a failsafe networking operation. Setting up multiple routers can be complicated and may result in a routing loop that may cause the network to work inefficiently or fail all together.

2.1.3 Virtual Private Networks (VPN)

A VPN is an extension of a private network to enable users in different geographical locations securely share resources and data as if they are directly connected to the private network (Mason, 2002). By using VPN regionally dispersed offices and roaming users can securely connect to head office and share data and resources. Users can also connect to proxy servers in order to hide their identity and geographical location.

VPNs use tunnels to encrypt and secure the data in transit over the network using datagram as the transport layer over the Internet Protocol (Technet, 2001). A packet from a client is patched with an Authentication Header (AH) for routing and authentication as it passes through the VPN router or gateway. The data is then encrypted and enclosed with an Encapsulating Security Payload (ESP) which constitutes the decryption and handling instructions for the receiving end.

The target router/gateway isolates the header information using the decryption and handling instructions and then forwards to the destination node. This makes it difficult for an attacker by Man in the Middle because the attacker must not only intercept the data but also decrypt it. A VPN provides an effective means of connecting remote nodes securely to the internet by employing multiple layers of authentication and encryption.

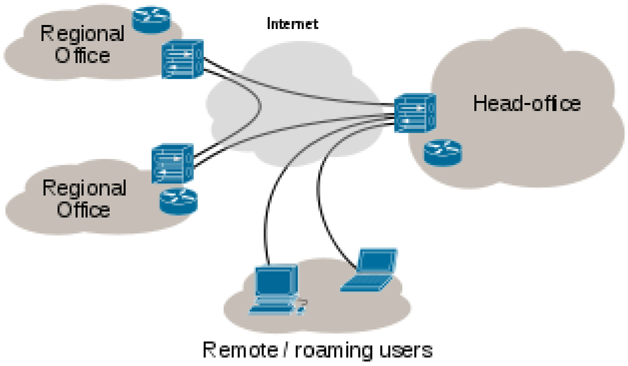


Figure 2.7: Virtual Private Network

VPNs ensure confidentiality and data integrity by only allowing authenticated users connect and exchange encrypted information using secure protocols like IPSec (internet Protocol

Security), tunnelling protocol, Secure Socket Layer (SSL), Transport Layer Security(TLS), PPP(Point to Point Protocol), SSH(Secure Shell), etc.

2.1.3.1 VPNs: Benefits

Organisations with many branches can link to each other using the public telecommunications network to link the branches. This saves money as compared to the linking of each branch using a dedicated line.

Data in transit over VPNs is encrypted and source nodes and protocols used are concealed. This enables VPNs to implement remote protocols that are complex to firewalls. Data in transit is isolated from the internet access.

2.1.3.2 VPNs: Shortfalls

VPN deployment is in public and requires high level of network security knowledge for installation and configuration to ensure security. In addition, performance and reliability of the VPN is based on the quality of service of the ISP.

VPN connections are slower than conventional connections due to data encryption and password requirement.

2.2 State of Practice in Network Vulnerability Assessment

While a lot of work has gone into designing and building secure networks using configuration and firewalls, there is little evidence of effort towards continuously monitoring networks for assurance of protection against attacks from emerging vulnerabilities. Operating system designers have many inbuilt resources that can be used to conduct vulnerability assessment. Users are however unaware of these resources and usually the resources are difficult to use and no single resource can be said to achieve conclusive results.

There are many commercial and free suites of applications that utilize the individual operating system utilities to conduct network vulnerability assessment. The problem is that the cost of the commercial applications may be prohibitive for small and medium enterprises. No single network can be said to be foolproof against attack from new vulnerabilities.

It is the responsibility of the Network Security Administrator to continuously monitor the evolving network security landscape in order to defend their systems by detecting and preventing attacks. According to (ESER, Kandogan, EBEN, & Haber, 2005) existing tools depend on the cognitive ability of the Network Security Administrator. In small and medium enterprises, usually the Network Security Administrator is also in charge of system management, end users and procurement. (Raja, F. et al, 2008) conducted interviews with 10 IT Security practitioners from small and medium organisations and established that a Networks Security Administrator requires knowledge and skills to perform inferential analysis, pattern recognition in order to predict and prevent unanticipated attacks on the complex and ever changing network configurations.

Network monitoring is an intrusion detection technique equivalent to spying on a network for a good cause (Stiawan, Shakhatreh, Idris, Bakar, & Abdullah, 2012). The aim of intrusion detection systems (IDS) is to attempt to compromise network security pillars namely: Confidentiality, Integrity and Availability. An Intrusion Prevention System (IPS) is an improvement of IDS which is capable of detecting and preventing intrusion. According to (Ashoor & Gore, July-2011) IDS and IPS are developed to fill in the gap for the shortfalls or inadequacies of Firewalls, routers and VPN as techniques for detecting and preventing intruders. There are two kinds of Intrusion Detection Systems (IDS); Host Based IDS and Network based IDS. In addition, there are commercial and freeware tools.

2.2.1 Host Based Vulnerability Assessment (HBVA)

HBVA utilise host scanners than can identify system level vulnerability that can include incorrect file or registry permissions and errors in software configuration to make sure that the target host complies to the organisation set out security policies and rules.

Network security administrators use host-based Vulnerability Assessment tools to manage system securities and standardize security policies. This includes enforcing policies, file access permissions, user rights and registry settings throughout the corporate network. Host-based tools such as Pedestal Software Inc.'s SecurityExpressions, System Scanner, ISS (Internet Security Systems) and Symantec's Enterprise Security Manager ™ can be used to execute this process. Although the three tools can perform the same functions; they differ in the deployment method within the network.

An agent must be installed on all the target systems for both Symantec's Enterprise Security Manager and ISS System Scanner to function. They are therefore time consuming to deploy for the required vulnerability assessment SecurityExpressions is however an agent-less deployment and is therefore easier to deploy to audit and enforce corporate security policy and system security. SecurityExpressions software package is featured below because it takes a few minutes to install and deploy. SecurityExpressions is utilized by over 1,700 organizations worldwide in all major industries.

2.2.2 Network Based Vulnerability Assessment (NBVA)

NBVA is carried out using network scanners with the ability to identify running services, detect open ports and simulate attacks to expose possible vulnerabilities. There are many commercial and open source network scanners available.

The choice of IDS depends on costs, effectiveness and impact on the organisational security. (Mathew, 2002) asserts that implementations of IDS coupled with strong organisations policies and procedures is an integral part of securing a network system but also cautions that no particular system can be said to be fully secured. IDS must be implemented alongside properly configured routers, firewalls and VPNs.

According to (Shirbhate & Patil, 2012), network process monitoring is becoming an important aspect for understanding and improving cyber security by implementing IDS using a combination of static/stateful pattern matching to identify and prevent the activities of hackers. Solutions for IDS and IPS must be economical, practical, cost-efficient and commercially viable.

Vulnerability Assessment scanners available in the market today include SAINT from SAINT Corporation, ISS (Internet Security System), Internet Scanner® and Nessus open source security scanner. These three Vulnerability Assessment scanners are well known and recognized by network security professionals for being robust, configurable and fast in scanning.

Both Internet Scanner® and SAINT™ are expensive. Nessus is however an open source VA scanning tool which is free of charge. Although many IT professionals do not trust open source and free software because they think that they do not provide same capabilities as commercial tools, Nessus is an exception because it is even more powerful than some of the commercial VA tools available in the market. In fact in 2002, Nessus was declared winner by the Information Security Magazine Excellence award for the 7th annual well connected tool in the Vulnerability Assessment Tool category award and was declared by NMAP users to be among the "Top 50 Security Tools". In addition, top products like SecureScan from Vigilante trust the capability and quality of Nessus. Vigilante incorporates Nessus in their inhouse and other commercial products to provide network vulnerability services to businesses. It is due to this high rating that Nessus is extensively covered in this study.

2.3 Technological Development in Intrusion Detection

Network intrusion detection has recently become a subject of interest due to the wide use of data communication systems and the frequent attacks where organisation networks have been paralysed for hours causing huge financial and reputation loses. Intrusion detection techniques can be classified as anomaly or misuse detection.

2.3.1 Network Misuse Detection

The aim of misuse detection is to detect known attacks and their variations thereof using a well defined set of rules. Misuse detection has a low rate of false alarms and cannot easily detect new attacks.

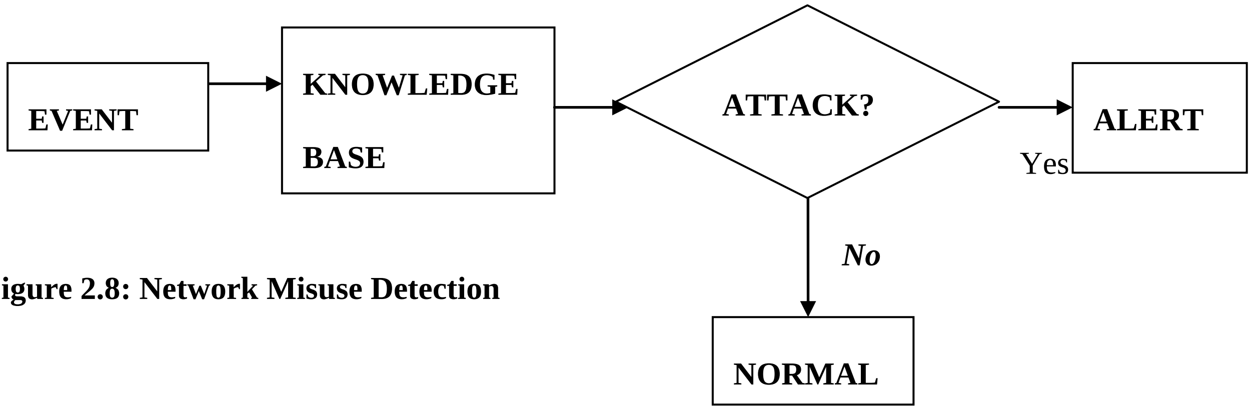


Figure 2.8: Network Misuse Detection

2.3.2 Network Anomaly Detection

Network anomaly detection aims to capture deviations from user profile activities and normal system behaviour pattern. Anomaly detection is difficult and has a tendency of generating many false alarms (Danie lBarbar ́a, Ningning Wu and SushilJajodia, (Undated))

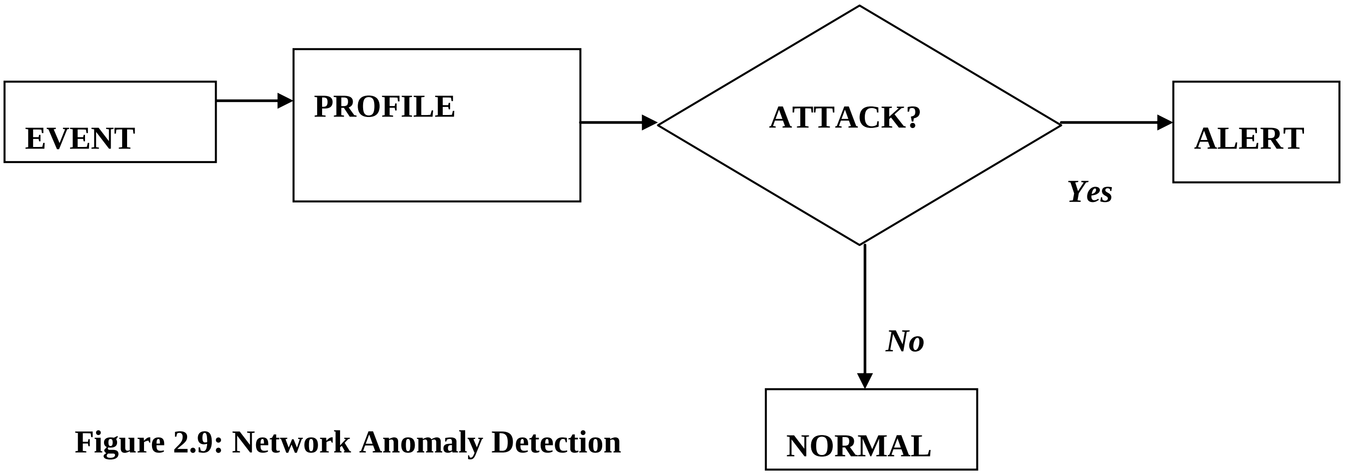


Figure 2.9: Network Anomaly Detection

2.3.3 Network System Testing

Specialised audit trails for intrusion detection have been recommended by experts to detect and analyse abnormal patterns. Intrusion Detection Expert Systems (IDES) record various intrusion detection measures of an aspect for each user in the form of connection, files read, CPU usage and system calls (Daniel Barbará, et al).

Types of system and network security testing are a factor of time and cost as shown in the following figure.

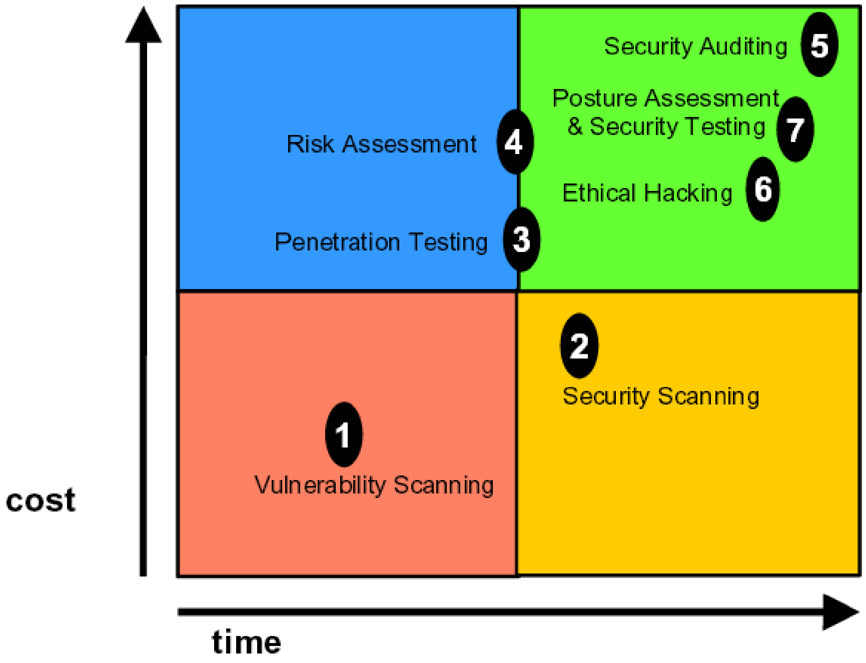


Figure 2.10: Network System Testing (Source: Institute for Security and Open Methodologies)

1. **Vulnerability Scanning:** automated checks for known vulnerabilities in a network system
2. **Security Scanning:** vulnerability scans which include false positives verification, custom professional analysis and weakness identification
3. **Penetration Testing:** goal oriented project aimed at gaining privileged access using pre-conditional means
4. **Risk Assessment:** security analysis by interviews and middle level research for business, legal and industry justifications
5. **Security Auditing: h**ands-on privileged inspection of the operating system and applications of a network
6. **Ethical Hacking:** penetration tests to discover valuable information in a network within a set time limit
7. **Posture Assessment & Security Testing:** risk assessment of the network using professional analysis on a security scan applying penetration tests to confirm false positives and negatives within a reasonable time

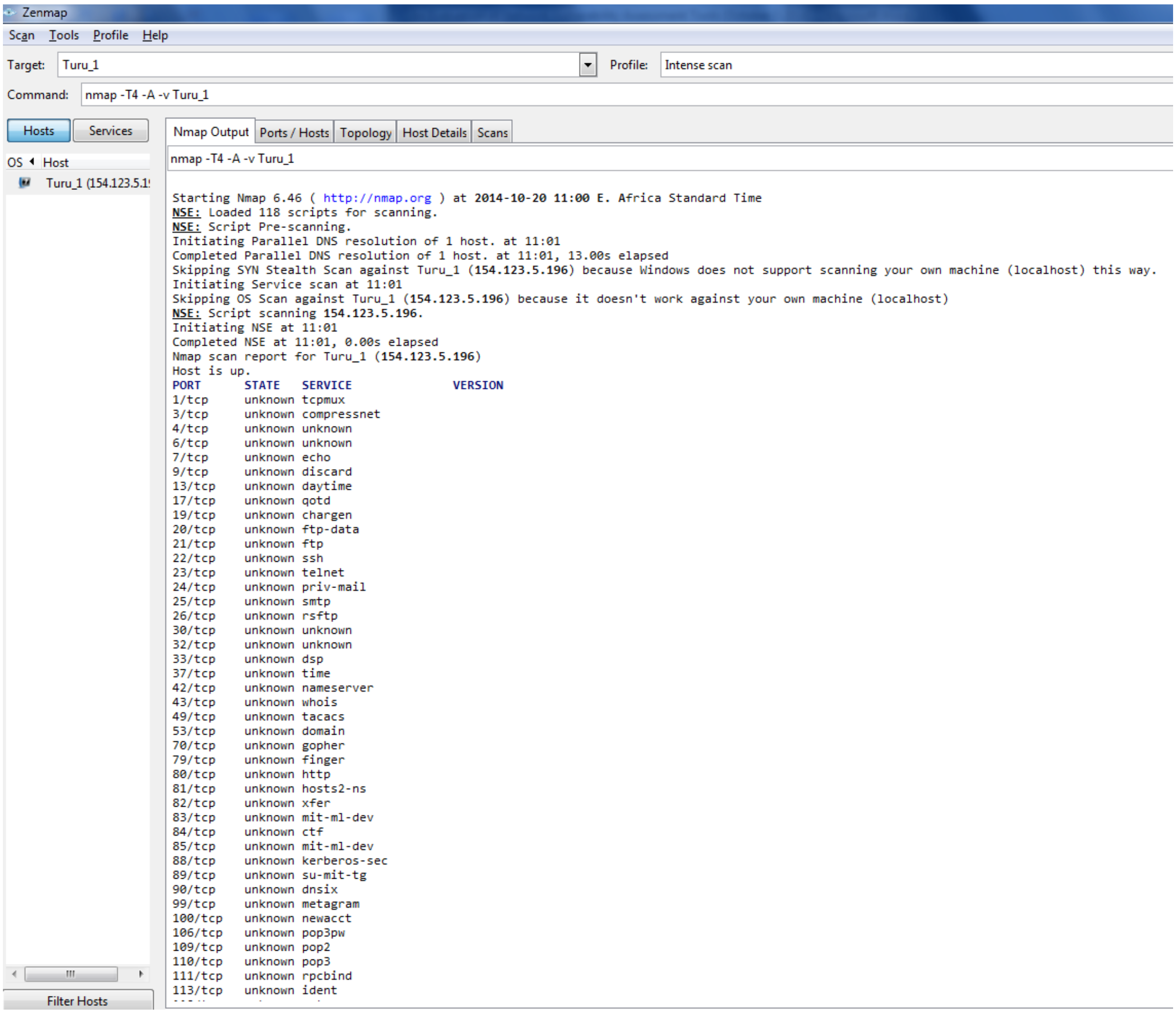
Today, network technologies include wired networks, wireless networks, mobile networks; next-generation converged networks; and social networks. (ESER, Kandogan, EBEN, & Haber, 2005) in a field study identify tools for vulnerability assessment to include Scanning tools, Global intrusion detection tools, Host/File integrity tools, Communications tools, AntiVirus software and Honeypots .

There are many commercial and free tools for network vulnerability assessment. The choice of the tool to use depends on the network architecture, types of threats to be detected and economic justification of the choice. Below follows an analysis of tools among those recognised by IT professional.

2.3.4 Scanning tools-NMAP (Network Mapper)

NMAP is a free network security tool for discovering services and hosts on a computer network and creating a network map for windows and Linux networks. NMAP is command based and is available for Linux as well Windows (NMAP). NMAP is a utility that can be used for host discovery (identify hosts on a network), port scanning (List ports that are open on a target host), version detection (determine names and version of services).

NMAP by itself is a command line system and can be difficult for novice network administrators. There is however an option for installing the ZENMAP frontend with a GUI.



Target: Turu_1

Profile: Intense scan

Command: nmap -T4 -A -v Turu_1

```
nmap -T4 -A -v Turu_1

Starting Nmap 6.46 ( http://nmap.org ) at 2014-10-20 11:00 E. Africa Standard Time
NSE: Loaded 118 scripts for scanning.
NSE: Script Pre-scanning.
Initiating Parallel DNS resolution of 1 host. at 11:01
Completed Parallel DNS resolution of 1 host. at 11:01, 13.00s elapsed
Skipping SYN Stealth Scan against Turu_1 (154.123.5.196) because Windows does not support scanning your own machine (localhost) this way.
Initiating Service scan at 11:01
Skipping OS Scan against Turu_1 (154.123.5.196) because it doesn't work against your own machine (localhost)
NSE: Script scanning 154.123.5.196.
Initiating NSE at 11:01
Completed NSE at 11:01, 0.00s elapsed
Nmap scan report for Turu_1 (154.123.5.196)
Host is up.
PORT      STATE   SERVICE            VERSION
1/tcp     unknown tcpmux
3/tcp     unknown compressnet
4/tcp     unknown unknown
6/tcp     unknown unknown
7/tcp     unknown echo
9/tcp     unknown discard
13/tcp    unknown daytime
17/tcp    unknown qotd
19/tcp    unknown chargen
20/tcp    unknown ftp-data
21/tcp    unknown ftp
22/tcp    unknown ssh
23/tcp    unknown telnet
24/tcp    unknown priv-mail
25/tcp    unknown smtp
26/tcp    unknown rsftp
30/tcp    unknown unknown
32/tcp    unknown unknown
33/tcp    unknown dsp
37/tcp    unknown time
42/tcp    unknown nameserver
43/tcp    unknown whois
49/tcp    unknown tacacs
53/tcp    unknown domain
70/tcp    unknown gopher
79/tcp    unknown finger
80/tcp    unknown http
81/tcp    unknown hosts2-ns
82/tcp    unknown xfer
83/tcp    unknown mit-ml-dev
84/tcp    unknown ctf
85/tcp    unknown mit-ml-dev
88/tcp    unknown kerberos-sec
89/tcp    unknown su-mit-tg
90/tcp    unknown dnsix
99/tcp    unknown metagram
100/tcp   unknown newacct
106/tcp   unknown pop3pw
109/tcp   unknown pop2
110/tcp   unknown pop3
111/tcp   unknown rpcbind
113/tcp   unknown ident
```

Figure 2.11: ZenMAP – Frontend for NMAP

2.3.5 Global Intrusion detection Tool - SNORT

Snort is a free and open source Network Intrusion Prevention System (NIPS) and Network Intrusion Detection System (NIDS) (Carr, Jeffrey, (2007-06-05). In 2009, Snort entered the infoworld's open source hall of fame as "one of the greatest pieces of open source software of all time" (The GreatestOpen Source Software of All Time, 2009).

Snort runs over IP networks in real time to analyse traffic and packet logging for intrusion detection. Snort is able to carry out protocol analysis, content matching and probing to identify attacks such as buffer overflows, SMB probes, operating system finger printing, stealth port scans and CGI attacks (Vyatta, 2011). Snort uses adjustable rules language together with a detection engine which utilises a modular plug-in architecture to examine network traffic and can be installed beside a firewall. Snort comprises 4 major components:

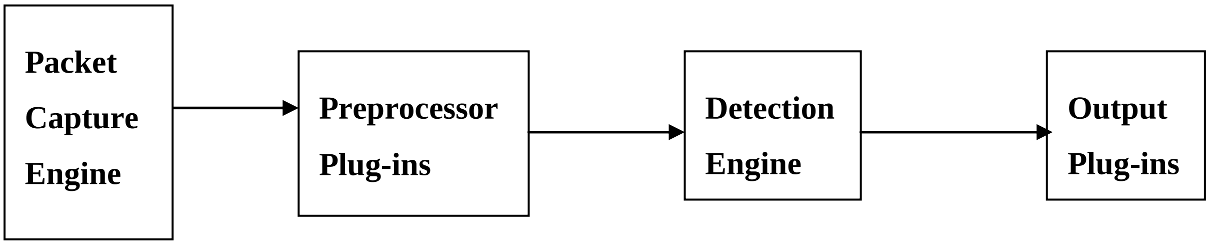


Figure2:12: Components of SNORT

2.3.5.1 Packet Capture Engine

The Snort packet capture engine uses WinPcap or libPcap (Pcap) to pick network traffic. Pcap are libraries that applications use to receive datagrams which are parcels in which data link level (2nd layer of the OSI Model) data is transported. The Network interfaces card captures physical data from the network media and hands it over to the NIC drivers that interface with the kernel of the operating system. Pcap picks the data from the OS kernel and hands it over to snort drivers which interface with the pre-processor component. Pcap libraries eliminate the need to pick data from the NIC in promiscuous mode.

2.3.5.2 Preporocessor Plugins

The Snort preprocessor component examines data received from Pcap and decides on whether to analyse, change, reject it or generate an alert. Snort preprocessors change URIs and URLs to a standard packet format, detect port scans, analyse stateful TCP/IP packets, Decode RPC and telnet packets. Preprocessors reject undesirable, potentially malicious packets that could curtail the functioning of Snort or degrade system performance. Packets that are not rejected are passed to the Detection engine.

2.3.5.3 Detection Engine

The detection engine decodes packets in accordance to the structure of the level of layer protocols starting with the lowest so as to systematically compare each packet to the rules. The engine then tests part(s) of the packets against strings or values related to a rule iteratively until all the rules known to snort are exhausted before it moves onto the next packet, identifying each match as a hit. Engine detection plugins can be used to enhance the capability of identifying attacks.

2.3.5.4 Output Plugins

The function of output plugins is to produce results that are displayed to the intrusion detection analysts. Snort uses rules in the pre-processor, decode and detection engines to create alerts.

2.3.6 Host/File integrity tool – OSSEC

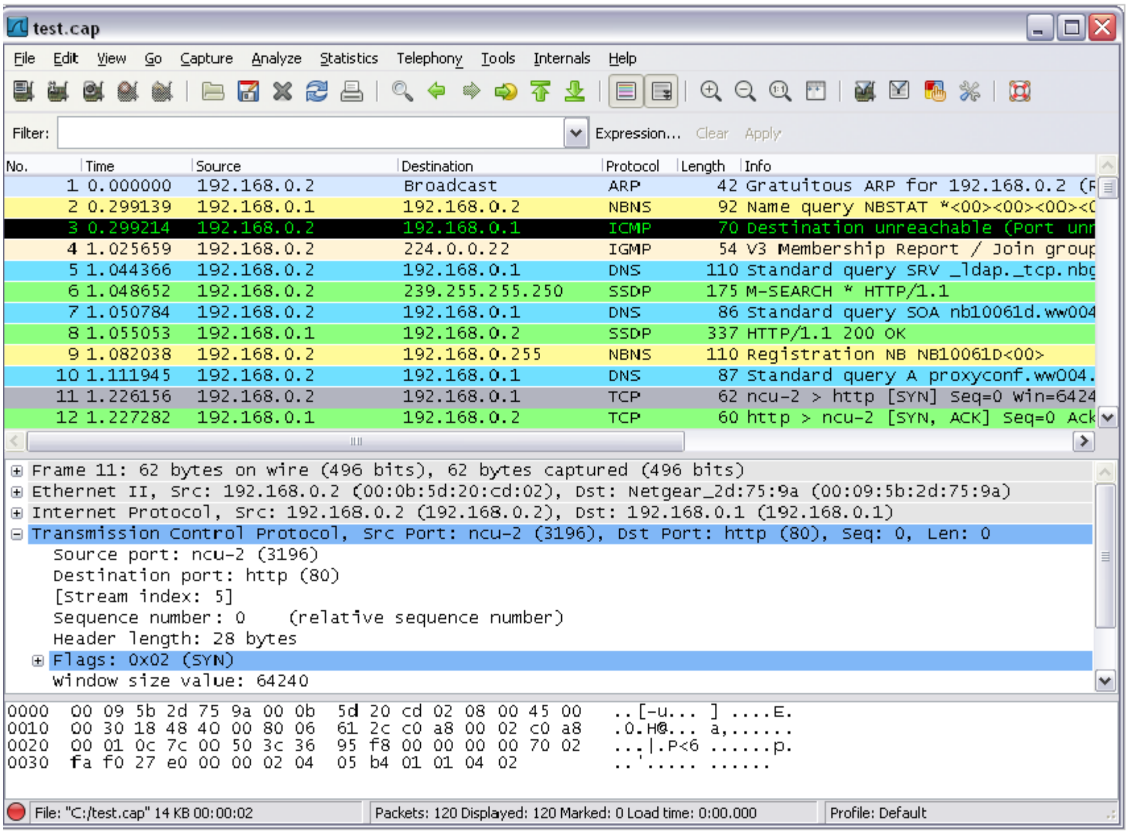
OSSEC is a host based intrusion detection system that is open source with support for many operating systems that include Linux, Solaris, Windows and MAC OS X. OSSEC can perform log analysis, rootkit detection, check registry/file integrity, monitor policies and generate real time alerts and active responses.

OSSEC has a log analysis engine which can analyse and correlate logs from many devices and formats that include: FTP servers, Mail Servers, SQL Databases, Web Servers, Web applications, Firewalls and many antivirus programs.

2.3.7 Communications tools - WireShark

Wireshark is an open source packet analyser applied in network troubleshooting, analysis, education and communication protocol development (InfoWorld, 2007). Wireshark is used for network troubleshooting, examining network security problems, debugging protocol implementations and learning network protocols.

Amongst the major features of Wireshark are support for Windows and Unix, Live packet capture from networks, support for packet data captured by TCPdump, Windump and many others, import for packet text files with packet data hexdumps, detailed protocol information display for packet data, export/import of packet data to other packet analysers, and search/filter/statistics/coloured display on many criteria


test.cap
File Edit View Go Capture Analyze Statistics Telephony Tools Internals Help
Filter:
Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.2	Broadcast	ARP	42	Gratuitous ARP for 192.168.0.2 (R
2	0.299139	192.168.0.1	192.168.0.2	NBNS	92	Name query NBSTAT *<00><00><00><0
3	0.299214	192.168.0.2	192.168.0.1	ICMP	70	Destination unreachable (Port unr
4	1.025659	192.168.0.2	224.0.0.22	IGMP	54	V3 Membership Report / Join group
5	1.044366	192.168.0.2	192.168.0.1	DNS	110	Standard query SRV _ldap._tcp.nbg
6	1.048652	192.168.0.2	239.255.255.250	SSDP	175	M-SEARCH * HTTP/1.1
7	1.050784	192.168.0.2	192.168.0.1	DNS	86	Standard query SOA nb10061d.ww004
8	1.055053	192.168.0.1	192.168.0.2	SSDP	337	HTTP/1.1 200 OK
9	1.082038	192.168.0.2	192.168.0.255	NBNS	110	Registration NB NB10061D<00>
10	1.111945	192.168.0.2	192.168.0.1	DNS	87	Standard query A proxyconf.ww004.
11	1.226156	192.168.0.2	192.168.0.1	TCP	62	ncu-2 > http [SYN] Seq=0 Win=6424
12	1.227282	192.168.0.1	192.168.0.2	TCP	60	http > ncu-2 [SYN, ACK] Seq=0 Ack

```
⊞ Frame 11: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)
⊞ Ethernet II, Src: 192.168.0.2 (00:0b:5d:20:cd:02), Dst: Netgear_2d:75:9a (00:09:5b:2d:75:9a)
⊞ Internet Protocol, Src: 192.168.0.2 (192.168.0.2), Dst: 192.168.0.1 (192.168.0.1)
⊟ Transmission Control Protocol, Src Port: ncu-2 (3196), Dst Port: http (80), Seq: 0, Len: 0
    Source port: ncu-2 (3196)
    Destination port: http (80)
    [Stream index: 5]
    Sequence number: 0    (relative sequence number)
    Header length: 28 bytes
  ⊞ Flags: 0x02 (SYN)
    Window size value: 64240
```

```
0000   00 09 5b 2d 75 9a 00 0b   5d 20 cd 02 08 00 45 00    ..[-u... ] ....E.
0010   00 30 18 48 40 00 80 06   61 2c c0 a8 00 02 c0 a8    .0.H@... a,......
0020   00 01 0c 7c 00 50 3c 36   95 f8 00 00 00 00 70 02    ...|.P<6 ......p.
0030   fa f0 27 e0 00 00 02 04   05 b4 01 01 04 02          ..'..... ......
```

File: "C:/test.cap" 14 KB 00:00:02 | Packets: 120 Displayed: 120 Marked: 0 Load time: 0:00.000 | Profile: Default

Figure 2.13: WireShark– Network Packet Analyser

2.3.8 Honeypots

A honeypot is a decoy system either inside or outside the network DMZ to gather information regarding the network intruder. Honeypots are forms of IDS and are additional levels of security besides firewalls designed specifically to deceive potential intruders for the purpose of gathering information regarding the attacks, their intentions and strategies (Schwartau, 1999). Such information is then used to protect the real production environment. Information gathered by a honeypot may be used as evidence in a court of law to prosecute. Available commercial honeypots include Cybercorp Sting by Network associates, Tripwire by Tripwire, Deception toolkit by Fred Cohen and Associates and Man Trap by Resource Technologies.

2.3.9 Anti-Virus Software

Antivirus software is a suite of programs designed to detect, search, prevent and eliminate malicious software like viruses, Trojan horses, Marlware, Adware amongst others. There are many brands of antivirus programs in the market with varying capabilities and costs. Examples are Symantec's Norton, Kaspersky, Eset Nod32, Avira, etc.

2.3.10 Simulators - Nessus

Nessus uses IP address range, subnet and host name or IP address to scan a target for vulnerabilities in HTTP, FTP, SMB protocols (Van Den Berg,et al., 2002). For any vulnerability found, Nessus attempts to exploit it but does not assume that the service is using IANA assigned ports. If ports 21, 80, and 8080 are open for HTTP services, Nessus will also perform appropriate security checks for FTP on all these ports. Each security check shares findings with subsequent checks to optimise checks. If an FTP server does not provide anonymous logins, the subsequent checks on the same target will skip this test (Deraison, 2003).

2.3.10.1 Nessus - Scanning Features

Nessus uses IP address range, subnet and host IP address/name to scan target systems. It connects to the target system and simulates various application protocols to initially investigate the system. To check for web server vulnerabilities, Nessus pretends to be a web browser and sends HTTP messages. On the other hand to check a windows file server, it pretends to be a windows client by sending SMB messages.

Nessus never assumes that target systems are always using designated IANA ports. Even when Nessus detects web services are running on the targer system on ports 8080, 80, and 21, it will attempt to detect whether the same ports are running any other services and if the services are running on any other ports.

Nessus scans log and share results of scans with subsequent security checks in order to be efficient. If Nessus detects that anonymous logins are not permitted on the target FTP server, then it will not perform similar checks in the subsequent checks on the FTP server. In this way Nessus completes Vulnerability Assessment scans faster, allowing the network security administrator to simultaneously scan an unlimited number of hosts. The power of the processor of the system hosting Nessus is therefore the factor that limits the number of hosts that can be scanned at a time. Nessus includes a ‘failsafe’ option that carries out vulnerability assessment based on service banners instead of actual vulnerability exploitation. This ensures that production servers do not go offline while undergoing vulnerability scanning.

2.3.10.2 Nessus Architecture

The features of Nessus include detection of the Operating System, scanning ports for vulnerabilities, information gathering and simulation of attacks. Nessus employs other open source security tools to accomplish some of these features as opposed to reinventing them. It uses NMAP for advanced port scanning and operating system identification., Hydra for brute force attacks for services like telnet, www and POP, Whisker and NIKTO for particular CGI and web server attacks and tests. Nessus must be launched from the ‘root’ directory in order to enable it launch these supporting third party programs. Nessus implements the client-server architecture. This enables it to allow a central server to carry out all the attacks on the target system and the client to provide a GUI (graphical User interface) which connects to the server to present options for scanning, view and save results. The server is based on POSIX (FreeBSD, Solaris, NetBSD, GNU/Linux) while the client can run on either UNIX (X Windows) or MS Windows platform. In addition, Nessus is so flexible that it uses command line to communicate with the scan engine to carry out VA scans. If the system hosting the Nessus server component has OpenSSL, then the communication between the server and the target is encrypted to test for SSL services on the target system.

2.3.10.3 Nessus Vulnerability Assessment Reporting

Nessus network vulnerability assessment reports provide a complete overview of the vulnerabilities of the target systems with a list of detected open ports and running services, vulnerabilities associated with these ports and services together with recommended patches and fixes complete with BID identification and CVE information for the identified vulnerabilities.

Every Nessus identified vulnerability is classified in as High/Low/Medium or Information. Nessus refers to High severity vulnerabilities as 'Holes', Medium/Low vulnerability as warnings and informational vulnerabilities as open ports.

Outcomes of the vulnerability assessment can be exported into different formats: Standard Query Language (SQL), command line file, NSR, Extended NSR, CSV, XML, ASCII text, HTML, Adobe PDF or centrally stored MySQL database. Any Nessus client can import NSR files. The rest of the file formats cannot be freely imported. HTML file formats have 2 reporting options only in POSIX; straightforward Vulnerability Assessment report and the same report alongside graphs and pie charts showing the type and number of vulnerabilities identified on the scanned system. Graphs and charts are excellent visual aids for emphasizing the findings, impact and extent of the vulnerability on the ICT infrastructure.

2.3.11 Scanning Features of SecurityExpressions

SecurityExpressions vulnerability assessment scans support MS Windows, SUN Solaris, LINUX, IBM-AIX and HP-UX platforms and key Microsoft applications like SQL Server, Internet Explorer, Outlook, and MicroSoft Office. It applies the latest patches and hotfixes to keep Microsft applications and Solaris platforms up to date

SecurityExpressions uses target systems' host name or IP address to implements quick and apparently trouble free scanning methods. It is capable of simultaneously scanning and fixing a maximum of 200 target systems running separate tasks and subtasks while waiting for the network to be available. It simplifies the process of scanning by offering the network security administrator three different approaches to enumerate available target systems on the network: "ping discovery" by sending echo Internet Control Message Protocol (ICMP) request traffic to systems in a given IP range and logging all those that respond; Microsoft uses network neighborhood utility to discover members of a domain; LDAP extracts lists of nodes form LDAP compliant and active directories. The network security administrator is able to manually add/import lists to create a custom target system list for batch or scheduled scans.

2.3.11.1 SecurityExpressions - Architecture

SecurityExpressions is unique as compared to other Hostbased scanners in that it uses an architecture without an agent (agent-less architecture). It uses MS Windows networking (NETBIOS) on ports 135 to 139, 445 and RPC on port 593 to scan and correct MS Windows target. SecurityExpressions uses MS Windows networking to confirm if the logged in user has the prerequisite security privileges to for example change registry entries and file permissions on the target system. If the currently logged on user lacks proper access privileges to perform these functions, then it is required to provide a user ID and password for privileged access and rights on the target system. The agent-less architecture scanning of target Unix systems is done through Secure Shell (SSH - port 22) using an administrator user ID and password on the target system. SecurityExpressions supports agents (port 9002) installed on the target Unix system to provide the necessary access if SSH service is not available on the target system. With the agent-less architecture, network security administrator saves valuable time because it is not necessary to install agents on the target system.

2.3.11.2 SecurityExpressions - Security Checks

SecurityExpressions uses a set of security rules set out in predefined policies to carry out security checks. The predefined policies are based on the type of operating system and the services (server, workstation, etc.) offered to the business environment by the target system. SIF files contain all policies defining all rules according to standard INI file format used by Microsoft Windows Operating systems and other programs to initialize and/or set parameters. SecurityExpressions by using this common format provides a lot of flexibility to network security administrators to easily modify/customize policies in line with company policies. A Graphical User Interface wizard is provided to simplify this process. The SIF files can also be further extended and customized using Perl, VBScript code or even Javascript.

For companies that do not have predefined security policies, SecurityExpressions allows them to use industries' best security practice guidelines to test their target systems Best industry practices and guidelines are included in a couple of SIF files like NIST Windows Security Guidelines; Internet Explorer Compliance Checks, SANS Securing Windows NT Security Step-by-Step Guide; NSA Windows and XP Security Guidelines; Microsoft Word and Excel macros security settings, lockdown for Linux/Solaris, sample rules for use with Unix systems and recommended security patches for Sun Solaris. SecurityExpressions also includes SIF files that scan for missing Microsoft patches and hotfixes, reports weak Windows systems passwords that can be easily guessed and audits applications installed on Windows systems in order to flag those ones that do not meet company security policies and standards. An online SIF library is also available from Pedestal Software Inc. providing up-to-date policy files for selected Microsoft applications and all platforms.

2.3.11.3 Vulnerability Repairing

SecurityExpressions unlike network-based vulnerability scanners repairs many of the problems found on the target system by executing scripts, modifying registry settings or installing hotfixes and patches. By so doing the network security administrator is empowered to quickly lock down company systems from a central location and ensure that the organizational network is consistent with uniform security settings. Installation of new patches and hotfixes to the target systems is made easy with a click of a button. Batch processing can be implemented to automatically correct all deviations or where necessary corrections can be done one item at a time in order to monitor impact on the system performance. SecurityExpressions maintains a complete log of all changes made on every target system. If the target system fails as a result of the changes, there is a provision for rolling back to the initial settings. A history of repair activities is also logged for future reference incase the network security administrator needs to undo their previous changes.

2.4 Critique of the literature

A network/host vulnerability scanner is an application that tests a network/host for vulnerabilities and reports the findings. A network Security Administrator may employ the same tools as the attacker. The intention of the administrator is to identify security holes so as to fortify the system while the intention of the attacker is to exploit the weakness. It is therefore desirable that the network security administrator works ahead of the attacker. In real life it is the other way round. It is the attacker who is ahead and the network security administrator is chasing.

Operating systems include very powerful programs for vulnerability scanning. Users however are not aware of these tools. There is a lot of free software that can be downloaded and many commercial applications for vulnerability scanning but the later may be cost prohibiting for small and medium enterprises.

There is little research to inform Network Security Administrators on the available tools and metrics for selecting an appropriate set of tools that can assure the organisation of safety of the network, applications and data. Network security administrators therefore depend on their own level of expertise, network configurations, firewalls and antivirus programs to thwart attacks. All these measures are based on known attacks. The network security administrators only come to learn of attacks after they are hit. A framework that will assist Network Security Administrators increase their level of confidence in the security of their networks will be useful.

Conventional intrusion detection systems do not provide a complete solution. According to (Youssef & Emam, 2011) a combination of Data Mining (DM) and Network Behaviour Analysis (NBA) techniques are more effective intrusion detection systems for misuse and anomaly detection. Most commercial IDS utilise misuse approach by storing known intrusion patterns in the system as signatures. By searching network traffic for known patterns, the IDS is able to send an alert only if a known intrusion pattern is detected. Anomaly detection systems make profiles based on usual network behaviour and can detected deviations from the normal behaviour which may be an intrusion pattern or a new behaviour that must be profiled.

2.5 Conclusions

There are too many variations in network configuration, connectivity, threat models and security policies. As a result there is no single complete turnkey security solution. Network security vulnerability assessment is a dynamic process because as the network administrator attempts to build secure networks, new systems and applications are being released and the attacker is continually punching holes into these systems. Vulnerabilities exist in systems because of:

i. Vendor errors (web applications, software bugs, missing patches or insecure default configurations)
ii. Administrator errors (incorrect configurations or lack of appropriate policy)
iii. User errors (malice, backdoors, improper sharing of resources)

Factors to consider when selecting a network vulnerability scanner are functionality, reliability, maintainability, efficiency, ease of use and portability. Although a well configured network, with a perfect implementation of IDS and IPS does not guarantee 100% security, a good mix of tools can give a higher level of assurance to the system administrator and the organisation at large. According to (Gomez, Gil, Padilla, Banos, & Jimenez, 2009) a good implementation of IDS and IPS should not require human intervention, be fault tolerant and survivable, have minimum processing overhead, able to accurately detected deviations from normal network behaviour and difficult to fool.

3 Research Methodology

This chapter describes the research method used in carrying out the study. It examines current research methods in evaluating NIDS and NIPS, describes the tool to be used in the research detailing the data collection, processing techniques and tools used for the research.

3.1 Current Methods in Evaluation of Network Security VA tools

Securing data systems and technologies is complex because they are dynamic. Given time, resources and motivation any attacker can break into almost any system. Security procedures and technologies currently in use cannot guarantee the safety of the network resources. There are many procedures and technologies in use to ensure that network systems are safe from intruders. Equally, there are many tools that can be used for vulnerability assessment. According to Dr. Paul Dorey, The Director of Digital Business Security, BP Plc., UK:

"*Information security provides the management processes, technology and assurance to allow businesses' management to ensure business transactions can be trusted; ensure IT services are usable and can appropriately resist and recover from failures due to error, deliberate attacks or disaster; and ensure critical confidential information is withheld from those who should not have access to it"*

There are many commercial and free tools for vulnerability assessment that one can choose from to conduct a vulnerability assessment. The COBIT (Control Objectives for Information and related Technology) framework is perhaps the most comprehensive resource that IT practitioners can use to adopt IT governance and control.

The COBIT framework identifies a baseline in form of survival kits for Home users, Professional users, Managers, Executives, Senior Executives and the organisational board members. The COBIT process framework for IT security comprises 34 generic IT processes grouped in four domains; Plan and Organise, Acquire and Implement (AI), Deliver and Support (DS), Monitor and Evaluate (ME). These processes endeavour to provide information that is effective, Confidential, Efficient, available with integrity, compliant and reliable using IT resources that comprise people, infrastructure, applications and information.

The framework details what is at risk, the possible consequences of exploitation and what could be done to mitigate the risk. There is no evidence of a framework for evaluating commercial or freeware vulnerability assessment tools that can be used in network monitoring and evaluation. Network Security Administrators depend on own experience, advertisements and recommendations from peers on what tools to use.

3.2 Research Design

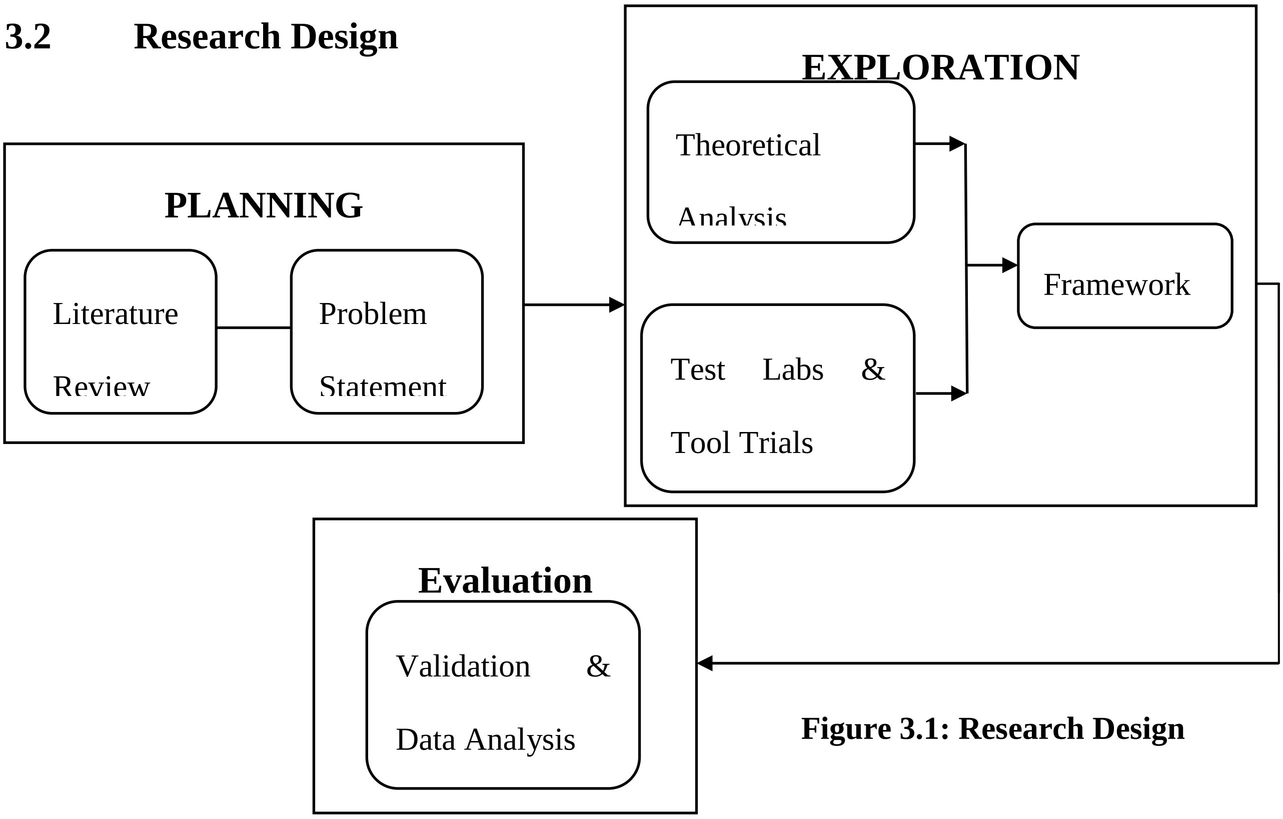


Figure 3.1: Research Design

The development and design of the framework for network vulnerability assessment will consisted of three phases namely Planning, Exploration and Evaluation.

3.2.1 Planning Phase

In this phase a literature review was carried out in order to guide the study, formulate the main research problem and identify the presence and/or limitations of existing frameworks for evaluating network vulnerability assessment tools.

3.2.2 Exploration

In the exploration phase, an analysis of networks was carried out to identify the network architecture and what is at risk. Based on this available tools for conducting vulnerability assessment were identified. Each of the identified tools was examined so as to classify it into one of the categories as scanning tool, global intrusion detection tool, host/File integrity tool, communications tools, anti-Virus Software or honeypot.

3.2.3 Evaluation

An analysis was carried out to establish the effectiveness of tools comparing results with the aim of establishing number of hits, false negatives and false positives. Effort was made to assign rankings for the performance indicators listed in 3.4.2 above.

It was necessary to do this so as to evaluate tools in each class. Tests and trials were carried out in a lab environment to establish the efficiency and capability of the tools. An intrusion simulator was selected and each tool subjected to the attacks by the simulation tool. Attacks were simulated from remote physical locations and within the LAN while all firewalls and user account controls were disabled so as to effectively test the intrusion detection tool. Each tool was examined for the following performance indicators Platform portability (Portability), Operational functionality (Functionality), Performance reliability (Reliability), Usability, Efficiency and Maintainability.

3.3 Conclusion

Use of firewalls and network intrusion detection systems are the main technologies used to monitor and guarantee modern day network systems (Alfaro, J. G., et al., 2007). To properly configure these systems, it is necessary that multiple sets of rules are used. Existence of anomalies especially in distributed multi-component systems can negatively impact the network security policy. Discovery and removal of the anomalies can be a serious and complex task for the network security administrator.

While there are many network vulnerability assessment tools, the task of finding the suitable tools can be daunting and heavily depends on the expertise of the network security administrator. No standard framework exists for identifying suitable tools for a given network configuration. The purpose of this research is to propose a framework that will assist the novice network security administrator ensure a secure networking environment.

4 Design of the VA Framework

4.1 Introduction

This chapter proposes a framework for network vulnerability assessment. The network vulnerability assessment tools are broadly categorized into Network-based Vulnerability Assessment Tools and Host-based Vulnerability Assessment Tools. It is therefore necessary for the assessor to clearly understand the technology under scrutiny in order to select the appropriate tools to subject to the framework.

When carrying out tool assessment for host vulnerabilities, the set of tools to be compared must be geared towards this technology and not to compare a tool meant for network vulnerabilities with that meant for host based vulnerabilities. This does not mean that there is no convergence in the tools but each tool will have strength for which it was created.

4.2 Conceptual Design

To effectively evaluate a network vulnerability tool, it is necessary to carry out a comparison of the effectiveness, efficiency and capability of tools in a managed environment. The aim is to determine which tool yields accurate results (Hits) and which one gives false alarms (False hits) and the number of misses.

The examination is done iteratively for each set of tools comparing at least results from 3 tools in order to determine the convergence of HITS so as to detect MISSES and FALSE HITS. The important factor is to hold all conditions constant for every tool. In case there is a dispute in the results for example where tool A detects a HIT, tool B misses, a third tool C is employed to verify HITS from MISSES and FALSE HITS.

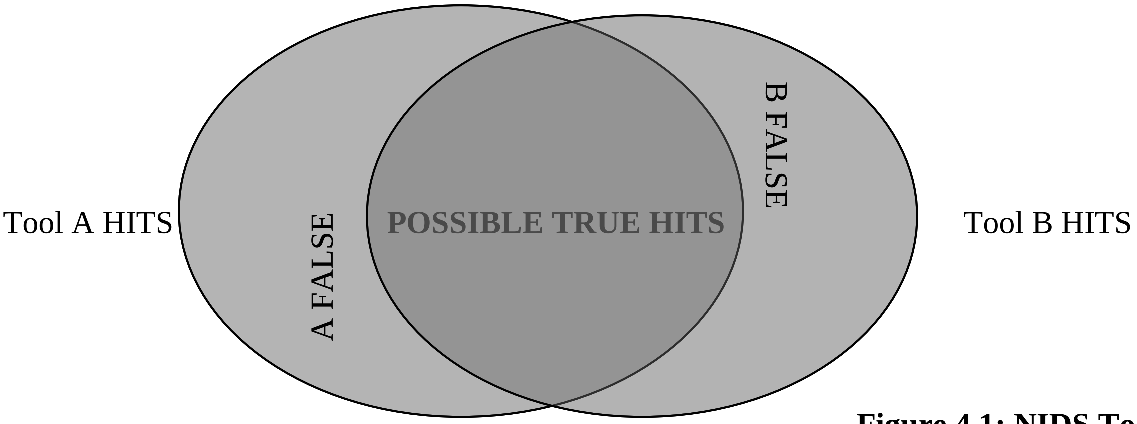


Figure 4.1: NIDS Tool Comparison

Tool A and Tool B capability comparison

Each circle represents the range of vulnerabilities each tool claims or is said to cover. The intersection represents vulnerabilities which both tools identify. The area in tool A outside the intersection represents vulnerabilities identified by tool A but not tool B (Potential Tool A FALSE HITS) and the area in B outside the intersection of tool A and B represents vulnerabilities identified by tool B alone (Potential B FALSE HITS).

Armed with the set of results for TRUE HITS, Potential A and B FALSE HITS, the network conditions are held constant and the network subjected to vulnerability assessment using a third vulnerability assessment tool C. The intersection of all three tools is considered as the true vulnerability range covered by the tools. Tools A, B and C are then ranked according to the size of the area of the circle in the convergence zone. The area of each tool in the intersection represents the reliability of the tool for the vulnerability assessment.

The ideal security presence in a network would be that all tools circles intersect forming one circle, meaning that each tool has a reliability ranking of 1. However due to software bugs and delayed upgrades to cover unknown vulnerabilities, there is never such an ideal situation and no single tool seldom attains the ranking of 1.

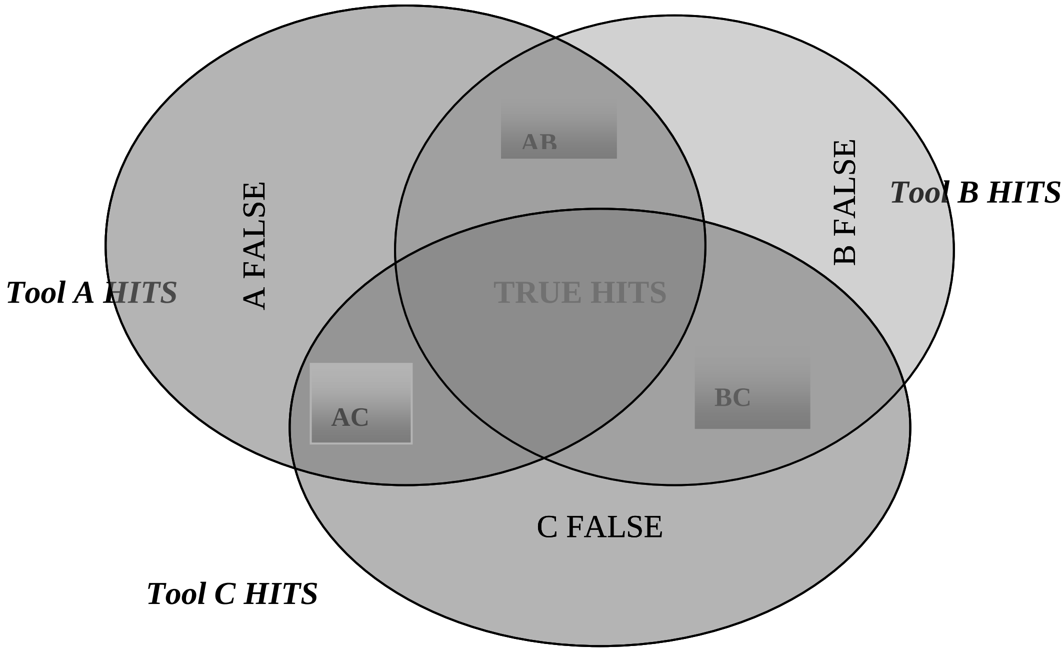


Figure 4:2: NIDS Tool Verification ***Tool A, B, C capability comparison***

The intersection AB can represent either false hits by both tool A and B or MISSES by Tool C. The intersection BC can represent either false hits by both tool B and C or MISSES by Tool A. The intersection AC can represent either false hits by both tool A and C or MISSES by Tool B. In case of doubt the network can be subjected to a 4th VA tool.

4.3 The Proposed Framework

This framework is developed according to critical review of existing literature, findings from network simulations in line with the research objective of developing a framework for the evaluation of network vulnerability assessment tools.

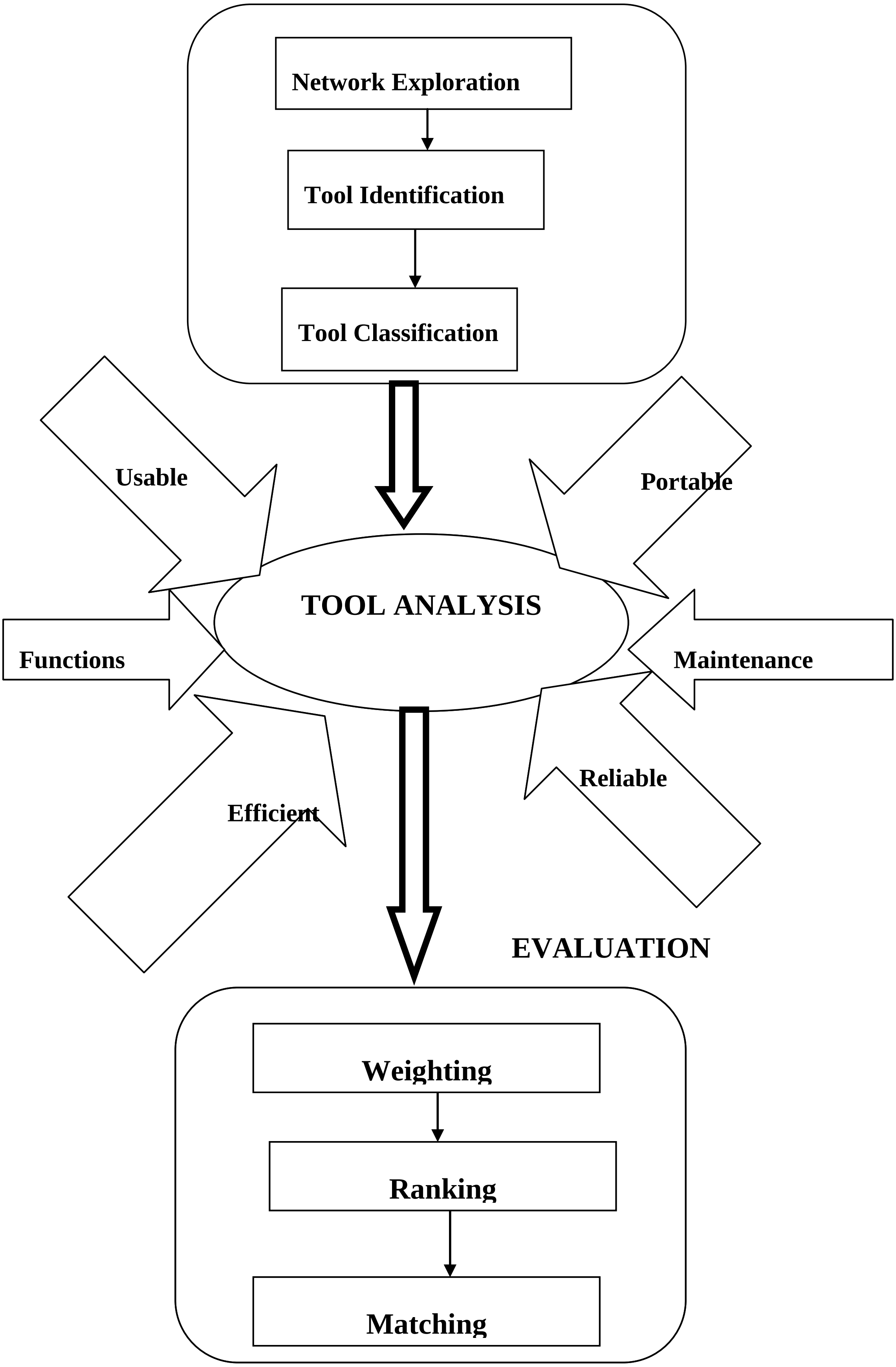


Figure 4.3: Framework Implementation

The major components of the framework are planning, tool analysis and evaluation.

4.3.1 Planning

To deploy an efficient and successful Vulnerability Assessment system in a business, the ICT security department must have robust and well documented policy and procedures. The documentation must clearly state the principles that outline the action to be taken in planning and carrying out all the activities of network and host vulnerability assessment. There must exist a change management system developed for follow-up of issues arising from the vulnerability assessment and a means of ensuring that the issues are resolved in the recommended manner. In the absence of documentation and change management system, vulnerability assessment cannot be carried out consistently and regularly. Quality of results from such scans cannot be guaranteed.

The planning component of the framework comprises network exploration, Tool identification and tool classification. Network exploration helps the assessor to identify the various architectural components of the network in order to identify what is at risk. Not all tools are efficient in every element of the network.

Based on the information regarding the network architecture and what is to be protected, tools are identified from journals and magazines or even vendor advertisements. Tools are then classified in accordance to the scope of vulnerability that they claim to cover. The main purpose of carrying out a VA is to ensure the presence of security.

4.3.1.1 VA for Security Presence

.According to the Open Source Security Testing Methodology, security presence in a network is an environment of security tests and comprises six overlapping basic elements. These basic elements are Physical, Communication, Internet Technology, Process, Information and

Wireless security. A strategy for carrying out a network vulnerability test must cover all the six aspects of security.

All the security elements must be evaluated so as to attain perfect security. The vulnerability assessor must observe a holistic view of the organization while conducting micro tests to identify vulnerabilities. Perfect security is subjective, depends on every organization and is achieved by observing best practices, industry regulations, business justification, security policy and legal issues for the region

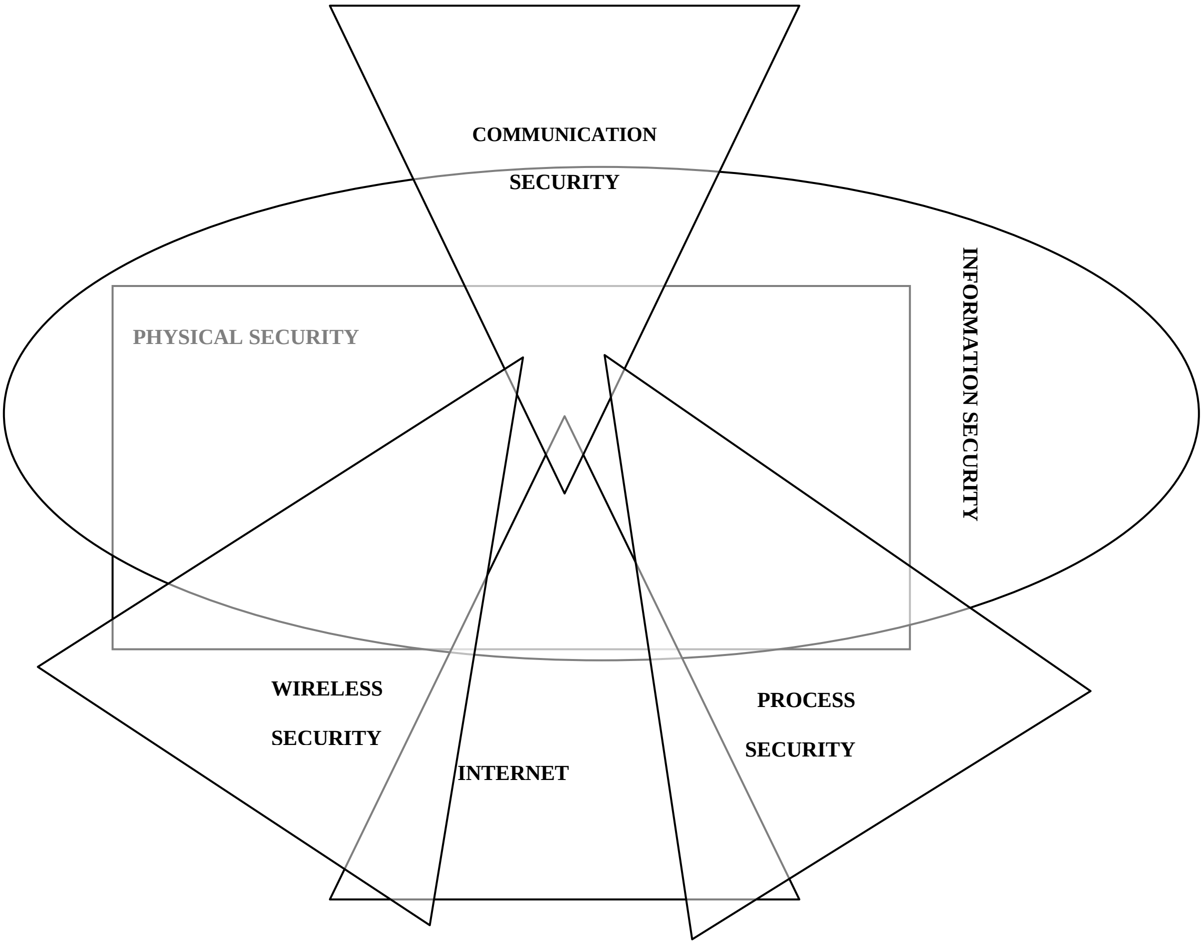


Figure 4:4 VA for Security Presence Assurance

4.3.1.2 Physical Security

This is the protection of the network, users, programs, data and hardware from physical circumstances or events that can result on lose/damage to persons, enterprise, institutions or agency as a result of fire, burglary, vandalism, natural occurrences, theft or even terrorism.

Vulnerability assessment for physical security is usually non-technical meaning that it cannot be automated but is a very important element of security. Vulnerability assessment for physical security can be done by examining fences, walls, smoke detectors, safe, water drainage, etc. In addition, surveillance and alert systems like sensors, smoke detectors, intrusion detectors and cameras must be verified to be in working condition. Last but not least are measures that can apprehend perpetrators before they cause harm and institutionalizing ways and means of recovering from the unfortunate events of attack, fire or accidents.

4.3.1.3 Wireless Security

Vulnerability assessment of wireless security involves the examination of the security/ condition of wireless networks and devices. These include Electromagnetic radiation devices, 802.11 networks, Bluetooth, wireless/handheld devices like PDAs, cameras, RFID and infrared devices.

4.3.1.4 Communications Security

Communication security involves the examination of the security of remote access control, PABX/Voice/Fax/VIOP communication, modem survey and X.25 packet switched Networks.

4.3.1.5 Internet Technology

Internet technology is perhaps the most vulnerable element of network security. It is necessary to examine all logistics and controls after enumerating all the network devices. Intrusion detection and password cracking tests must be examined alongside the ability of the network to survive an attack. Tests on routing, access control, alert and alarm control systems must be done.

4.3.1.6 Process Security

It is necessary to identify all running processes, whether they are necessary and how they respond to forward and reverse requests in addition to the access permissions.

4.4 Tool Analysis

The purpose of this module in the framework is to ensure that an efficient cost effective set of tools is used for the vulnerability assessment. Automated network vulnerability tools are basically software and their quality can be measured using software quality metrics which are Functionality, Portability, Reliability, Maintainability, Efficiency and Usability.

As mentioned before the quality and level of network security presence is dependent on the organizational policy, regulations and most importantly the capability and level of expertise of the network security administrator. The software quality metrics listed earlier will therefore have varying importance based on organization policy, existing regulations and expertise of the security administrator.

4.4.1 Functionality

Refers to the ability of a software to take input and yield consistent output according to design as per the systems requirements specification (Sommerville, 2006).

4.4.2 Portability

Software portability refers to the ability of software to run on different versions of software and hardware with little or no modifications (Garen, 2007).

4.4.3 Reliability

Software reliability is ability of a computer program to perform its intended functions and operations under system environment without crashing (Pressman, 2005).

4.4.4 Maintainability

It is the ability of a computer program to be retained in its original form, the ease with which it can be restored to this form in case of failure or be adapted to changing system functions (Martin, 2001).

4.4.5 Efficiency

Efficient software performs the intended function using system resources sparingly and without interfering with the overall system performance (Graham Bolton, 2008).

4.4.6 Usability

Software usability is a result of a well designed program that can be operated intuitively with ease and speed to solve problems for which it is designed (Graham Bolton, 2008).

4.5 Evaluation

The evaluation module of the framework comprises performance weighting, tools ranking and matching.

4.5.1 Weighting

In accordance to the organizational policy, rules, regulations and ability of the network security administrator, each of the software performance metrics discussed above is assigned a weight. It is important that this is done objectively and without prejudice.

4.5.2 Tool Ranking

Having assigned each of the metrics a weight, a test is then simulated for each tool in order to assign a corresponding value in accordance to the performance of the tool. Again this must be done as objectively as possible in order to achieve an efficient vulnerability assessment solution. The sum of the assigned values will yield the ranking of the tool in its class i.e communications, internet technology, wireless, process or information security.

4.5.3 Tool Matching

As mentioned earlier, several tools may cover various elements of security with varying capabilities. According to the ranking obtained above, for each of the security elements a set of two or more tools will be selected to be used for vulnerability assessment. It is important to note that no single tool can give total security assurance and that there is never a situation where total security is assured because of the dynamic nature of networks and threats.

5 Implementation and Testing

5.1 Introduction

This chapter highlights the Vulnerability Assessment proposal and presents a diagrammatic representation of the implementation framework

Poor network design (ie. Firewalls implementations), low staffing and low departmental budgets are factors that hinder efficiency by IT security professionals. The effect of these obstacles can be more pronounced in conducting vulnerability assessment on regular basis because the assurance that a network is secure is not tangible. It is upon this background that I have designed a framework that is low cost, efficient and easy to follow in order to overcome the said obstacles in conducting host and network based vulnerability assessment so as to ensure that the organizational ICT infrastructure is free from vulnerabilities, known and unknown in accordance with the business policy and prevailing best practices and legislation.

For best results, it is recommended that both host and network based vulnerability assessments are carried out at the same time. This is because host based and network based vulnerability assessment do not execute the same tests on target systems and therefore do not yield same results but are complimenting to each other. To gain a comprehensive view of the network security risk, host and network vulnerability assessment must be carried out at the same time and the results used together.

I recommend using Nessus and SecurityExpressions software on a laptop. The use of a laptop makes the scanner mobile to overcome limitations of network design for example internal firewalls. This solution allows the ICT security manager to conduct regular, efficient and cost effective vulnerability assessment on all business systems using in house tools.

This solution overcomes the obstacles created by an internal firewall in a normal network setup where the vulnerability assessment scanner would be prevented from accessing the target system. The vulnerability scanner will be scanning against a firewall and will produce inaccurate results. The firewall will not allow the vulnerability scanner to directly access all open ports on the target system. If for instance the target system has ports 21, 22, 80 and 1433 open and the firewall only permits access to port 22 and 80, then the vulnerability scanner will only have access to ports 22 and 80. Access to ports 21 and 1433 will be denied because they are blocked by the firewall. The vulnerability scanner will miss out critical vulnerabilities on services running on these ports. In addition, most vulnerability scanners use ports 22, 137 – 9, 445, 593, 5600 and 9002 to establish communication with target systems.

Such communication links will be barred by the firewall indicating that there are no vulnerabilities when in the real sense, they could exist. My strategy provides for both network and host based vulnerability assessment which can be launched from a mobile device and be directly connected to the target system. This bypasses internal firewalls to perform accurate vulnerability assessment on the target.

One may argue for the permanent location of the server component of the vulnerability assessment scanner on the same side of the firewall and the target system considering that the flow of communication between the vulnerability assessment scanner client and server components is permitted through the firewall. In most network implementations, this would not work due to the presence of internal firewall implementations that would hinder the VA scanner server component from accessing all the internal target systems. A network design obstacle would therefore still be experienced.

Furthermore installing several VA scanner server components on demilitarized zones throughout the corporate network is logically difficult considering upgrades and centralized management. An insecure host for the VA scanner server component would cause vulnerability from internal attackers who can hijack the VA scanner server host to identify vulnerable systems and exploit them.

My implementation of launching the host and network based VA scanner from a laptop overcomes these hindrances. The implementation not only includes the client and server component of the VA scanner but also reduces cost of hosting the server side of the VA scanner. The VA scanner launch pad is the mobile laptop which is not always connected to the network and cannot be hijacked. It is however important to be aware that the implementation of this flexible VA scanning system demands careful planning and employment of appropriate set of hardware and software.

5.2 Hardware Requirements

The key hardware required to launch this strategy is a laptop of minimum Pentium DualCore Processor operating at least 1.8GHZ with minimum memory 1000 MegaBytes. It is important to note that this is only a minimum requirement and the higher the laptop configurations, the better the scanning performance giving the VA scanning administrator ample time to audit the target system. A laptop with Intel Core i3 Processor operating at 2.5GHZ with 4GB RAM, 500GB Hard disk is recommended and would cost about $US 500. A network switch will also be required with an appropriate Cat 5e or 6e patch cord (crossover/straight) depending on the network design.

5.3 Software Requirements

MS Windows and Linux must be installed on the laptop. It is recommended to install Linux and Windows in the same partition with Linux as the host OS and Windows the guest. Microsoft Windows 7 Professional with service pack 3 works well as the guest OS and Linux

Red Hat 8.0 as the host OS on using VMware Workstation 4. VMware workstation 4 is said to be the only virtual machine software for the desktop that runs both windows and Linux to provide exceptional performance, functionality and stability. One needs to harden both operating systems before installing any other applications. On operating systems like Windows 7 it is a prerequisite that SecurityExpressions is installed with Nessus Windows's client (Nessus WX) and Nessus installed on the host operating system (e.g. – Linux Red Hat).

It is necessary to purchase only Microsoft Windows 7 Professional operating system, Vmware™ Workstation and SecurityExpressions. This makes this strategy low cost because Linux Red Hat 8.0, Nessus and NessusWX are free open source codes.

5.4 Target System Requirements

Nessus and SecurityExpressions require target systems to meet some basic requirements in order to achieve accurate vulnerability assessment results; Target systems must be switched on, connected to a common network with ping and basic services enabled and no firewall protection to be scanned by Nessus. On the other hand SecurityExpressions also requires that the target systems be on power, connected to a common network with no firewall protection and that the logged on user must have appropriate user privileges such as a system administrator. To use SecurityExpressions NetBIOS must be enabled on Microsoft windows operating system and on Linux systems, OpenSSH must be installed. Only when all these basic conditions are met, then Nessus or SecurityExpressions network vulnerability assessment can be carried easily.

5.5 Implementation

As mentioned earlier, there are host based scanners also known as inside looking around vulnerability scanners and network based vulnerability scanners referred to as outside looking in scanning. For each of the scanning approaches, there are various tools that are available for carrying out the scanning.

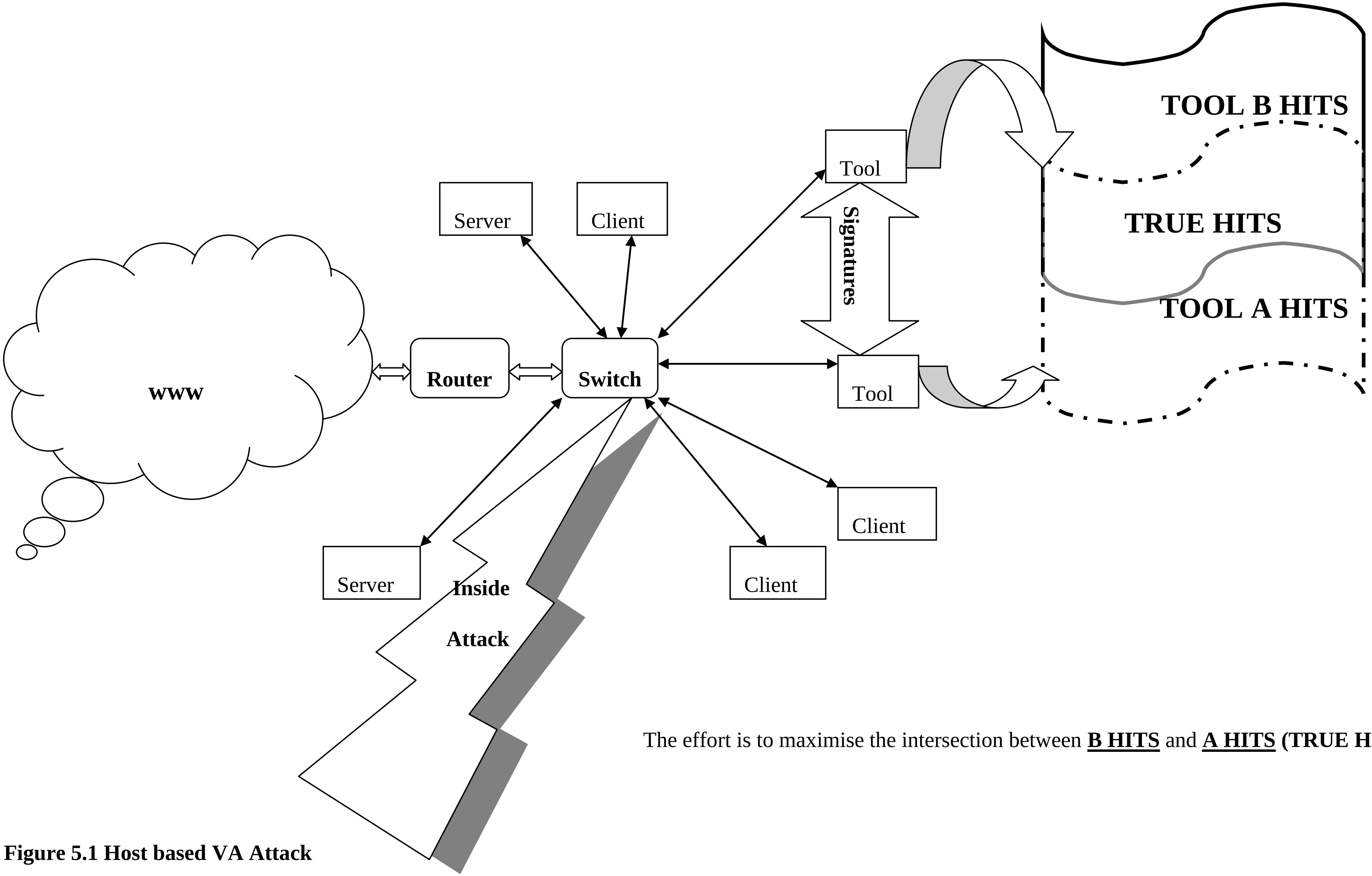
Nessus network simulator is used to simulate network attacks and Security Expressions is used to simulate host based vulnerability assessment.

5.5.1 Inside Looking Around Approach

From this viewpoint, the assessor assumes the role of a trusted network user with elevated permissions. The assessor can see internal resources such as file and print servers, databases and many more shared resources.

In many network installations, a lot of effort goes into keeping the intruders out forgetting to secure internal resources like authentication procedures for local resources, department firewalls and user control lists. From this position the assessor is capable of conducting penetration tests using the information gleaned from the vulnerability assessment.

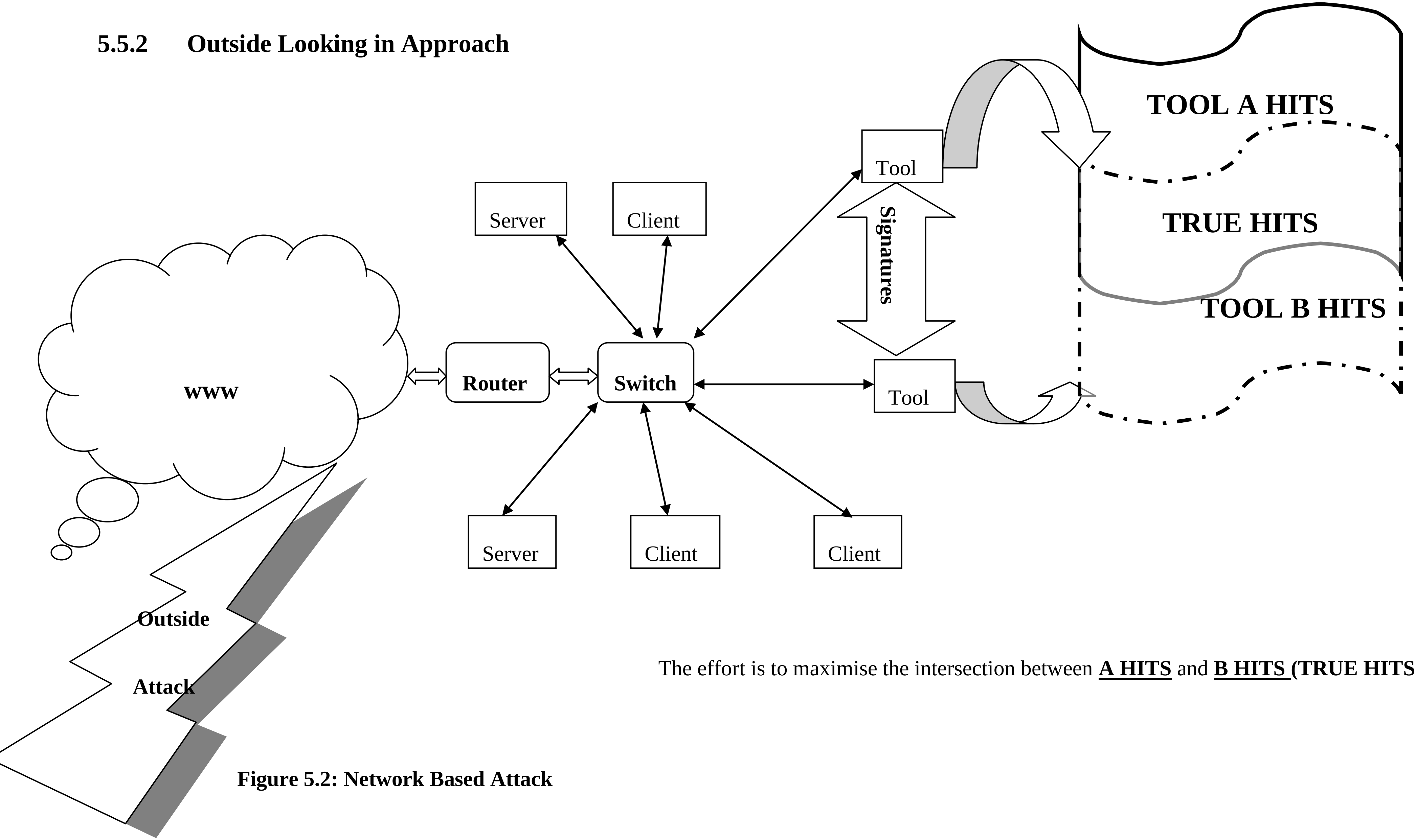
According to (Holland, 2004), **SECURITY = VISIBILITY + CONTROL.** IDS provide live visibility of what is going on in the network and stores the results for analysis or reporting at a later time to aid in decision making and network security policy formulation based on real world data that is quantifiable.



The effort is to maximise the intersection between **B HITS** and **A HITS** **(TRUE HITS)**

Figure 5.1 Host based VA Attack

5.5.2 Outside Looking in Approach



The effort is to maximise the intersection between **A HITS** and **B HITS (TRUE HITS)**

Figure 5.2: Network Based Attack

In this approach, the vulnerability assessor assumes the hackers viewpoint and attempts to compromise the system from outside. The assessor is able to see the public resources like the publicly routable IP addresses, external interfaces of the firewall and resources in the network 'demilitarized zone' (DMZ) which constitute nodes accessible to the internet such us the HTTP (Web), FTP, DNS and SMTP servers (Mail).

The DMZ simply constitutes a subnet or computer that sits in between a trusted network (Private LAN) and a public network (internet). From this viewpoint, the assessor has an untrusted status and has limited access to systems and resources. The motivation is to try and find security holes and vulnerabilities which can be used to penetrate the network. This stage is therefore referred to us vulnerability assessment.

5.6 Data analysis methods

Every vulnerability assessment tool will have its own reporting method, standard and terminology. This may make it difficult for a novice network administrator to compare results from the various tools. One tool may refer to an attack by a different name. The attack type may be a variant of the original attack.

The analytical skills of the vulnerability assessor are called upon to standardize and classify attacks for effective comparison. The objective is to find a common intersection between what tool A and B refer to as vulnerability. The result is a table with 6 columns.

Tool A Vulnerabilities	**Tool B**	**Common name**	**A Value**	**B Value**	**Total**
name by tool A	name by tool B	Standard name	1 if Hit by A, otherwise 0	1 if Hit by B, otherwise 0	A + B Value
			Sum (A)	**Sum (B)**	**Grand Sum**

Tool A Ranking= $\frac{Sum\ (A)}{\text{Grand Sum}}$ x 100

Tool B Ranking= $\frac{Sum\ (B)}{\text{Grand Sum}}$ x 100

If both tools ranking tend towards 100, then the 2 are reliable and can be depended upon for vulnerability assessment. On the other hand if there is a big difference between tool A and tool B, then further comparison is necessary to approve or reject one of the tools.

Tool C Ranking= $\frac{Sum\ (C)}{\text{Grand Sum}}$ x 100

If there is a large discrepancy between ranking for tool A and Tools B and C and Tool A has a high value, then tool A is eliminated because of too many **False Positives**.

If there is a large discrepancy between ranking for tool A and Tools B and C and Tool A has a low value, then tool A is eliminated because of too many **False Negatives**.

6 Conclusions and Future work

6.1 Conclusions

For a company to achieve sustained development and growth it is imperative that the communications systems are secure and protected against vulnerabilities. The network security administrator requires tools and knowledge in order to identify system vulnerabilities and restore compromised systems quickly. The network security administrator must be vigilant because of the dynamic nature of IT processes in an organization. Frequent Network and host vulnerability assessment must be done to assure the organizations that the IT infrastructure is secure and in line with company security policies as well as the general legal requirements and standards. Frequent vulnerability assessments will enable businesses to secure and maintain networks efficiently with low budgets.

In order to avoid legal complications, all planned assessment activities including IP address ranges; types of attacks to be conducted etc. must be documented in advance and signed by management and the assessment parties. The value of a security assessment to a company can be difficult to measure in that there may be no tangible outcome besides feedback on whether the network is considered “secure”. Whereas being “secure” means a system is safe from threats, security compliance means the system conforms to a given set of security requirements. Compliance provides assurance that a certain level of security has been achieved. This compliance can be required by government legislation, industry standards organisations or an organisation’s own policies.

6.2 Suggestions for Further Study

This framework only forms a foundation for evaluating network vulnerability assessment tools. It provides useful experience in evaluation of VA scanners for risk analysis, tool integration and correlation technologies. Feedback from network security experts who will use this framework will go a long way in shaping future research in the evaluation of network evaluation vulnerability assessment tools. This can help develop a comparative analysis of security policy developed with and without vulnerability assessments. This will inform network security managers the robustness of each policy type and the long term cost of maintenance answering the following questions for each: How robust? What is the cost of maintenance in the long term?

Further study and research is required in order to develop a robust, secure and centralized solution for implementing enterprise network infrastructure.

References

Daniel Barbará, et al. (n.d.). *George Mason University*. Retrieved 7 19, 2014, from Audit Data Analysis and Mining: http://cs.gmu.edu/~dbarbara/adam.html

Alfaro, J. G., et al. (2007). Complete Analysis of Configuration Rules to Guarantee Reliable network Policies. *International Journal for Information Security* .

Ashoor, A. S., & Gore, S. (July-2011). Intrusion Detection System (IDS) & Intrusion Prevention System (IPS): Case Study. *International Journal of Scientific & Engineering Research, Volume 2, Issue 7* , ISSN 2229-5518.

Brian, K. (2010). *Firewall for Dummies* (ISBN: 0-7645-4048-3, 2nd ed.).

Carr, Jeffrey. ((2007-06-05, June 5). Retrieved July 19, 2014, from "Snort: Open Source Network Intrusion Prevention": http://www.esecurityplanet.com/prevention/article.php/3681296/Snort-Open-Source-Network-Intrusion-Prevention.htm

Dan Pei, Lixia Zhang, Dan Massey. (April 2004). A Framework for Resilient Internet Routing Protocols. *IEEE Network special issue on Protection, Restoration, and Disaster Recovery* .

Danie lBarbar ́a, Ningning Wu and SushilJajodia. ((Undated)). *Society of Industrial and Applied Matehmatics*. Retrieved July 19, 2014, from Detecting Novel Network Intrusions Using Bayes Estimators: http://epubs.siam.org/doi/pdf/10.1137/1.9781611972719.28

Deraison, R. (2003, February 26). Retrieved July 20, 2014, from Nessus Data Sheet: http://www.nwfusion.com/reviews/2002/0204bgrev.html

ESER, Kandogan, EBEN, & Haber. (2005). *Security Administration Tools and Practices. In: Security and Usability.* O'Reilly, p. Chapter 18.

Garen. (2007). Software Portability: Weighing Options, Making Choices. *The CPA Journal 77 (11)* , 3.

Gomez, J., Gil, C., Padilla, N., Banos, R., & Jimenez, C. (2009). Design of a Snort-Based Hybrid Intrusion Detection System. Omatuet al. (Eds.): IWANN. *The International Work Conference on Artificial Neural Networks, Part II,* (pp. 515-522). LNCS 5518.

Gouda, M. (2008). A model of stateful Firewalls and it's Properties. *IEEE International Conference on Dependeble Systems and Networks,* (pp. 1-15).

Graham Bolton, S. J. (2008). *IfSQ Level-2 A Foundation-Level Standard for Computer Program Source Code.* IfSQ, Institute for Software Quality.

Gupta, K. K. et al. (2006). Attacking Confidentiality: An agent based approach. *Proceedings of IEEE International conference on Intelligence and Security Informatics, Lecture Notes in Computer Science* (pp. 285–296). Springer Verlag, vol. (3975).

Holland, T. (2004, February 23). Understanding IPS and IDS: Using IPS and IDS together for Defense in Depth. *GSEC Practical v1.4b, Option1, SANS Institute* .

InfoWorld. (2007, 9 10). Retrieved July 19, 2014, from Network protocol analyzer: Wireshark: http://www.infoworld.com/%5Bprimary-term-alias-prefix%5D/%5Bprimary-term%5D/best-open-source-in-networking-832&current=5&last=1#slideshowTop

Institute for security and Open Methodologies. (2003). *The Open Source Security Testing Methodoly Manual.* OSSTMM.

John The Ripper. (n.d.). Retrieved July 20, 2014, from http://www.openwall.com/john

Katkar, D. (2010). Tolerant Distributed Intrusion Detection System Using Packet Filter and State Transition Tables. *International Journal of Computer Applications , Vol. 8 No.11 (Novel Archtechture for Intrusion)*, 29-32.

KENNETH I, STEPHANIE F. ((Undated)). *A History and Survey of Network Firewalls*. Retrieved July 22, 2014, from University of New Mexico: http://www.cs.unm.edu/~treport/tr/02-12/firewall.pdf

Kobayashi, Y. B. (2003). *Intrusion Detection Systems: Technology and Development.* Nihon and Beihang University: IEEE Computer Society Press.

Kozierok, C. M. (2005, September 20). *Overview Of Key Routing Protocol Concepts: Architectures, Protocol Types, Algorithms and Metrics*. Retrieved July 22, 2014, from The TCPIPGuide.com: http://www.tcpipguide.com/free/t_OverviewOfKeyRoutingProtocolConceptsArchitecturesP.htm

Martin, R. (2001). Managing vulnerabilities in networked systems.

Mason, A. G. (2002). Cisco Secure Virtual Private Network. Cisco Press p.7.

Mathew, D. (2002). Choosing an Intrusion Detection System that Best Suits your Organization. *GSEC Practical v1.4b Option A* .

Medina, L. F. (2003). In *The Weakest Security Link Series* (p. 54). IUniverse.

Microsoft Corporation. (n.d.). Retrieved July 16, 2014, from Technet : http://technet.microsoft.com/en-us/library/cc959354.aspx

Microsoft. (2014). *What Is Unicast IPv4 Routing?* Retrieved August 21, 2014, from Microsoft Technet: http://technet.microsoft.com/de-de/library/cc736574%28WS.10%29.aspx

NMAP. (n.d.). Retrieved July 19, 2014, from Nmap Network Scanning: http://nmap.org/book/nmap-overview-and-demos.html

Oppliger, R. (1997, May). Internet Security: FIREWALLS and BEYOND. *Communications of the ACM 40* , p. 94.

PAULSON, L. D. (2004). *Researchers Develop Network-Security Visualization Tools.* Computer.37(4), pp. 17-18.

Pressman, S. (2005). Software Engineering: A Practitioner's Approach (Sixth, International ed.). McGraw-Hill Education.

PStools. (n.d.). Retrieved July 20, 2014, from Sysytem Internals: http://www.sysinternals.com/utilities.html

Raja, F. et al. (2008). *Effectiveness of IT Tools in Practice.* Vancouver, Canada.

SC Magagazine. (2010, February 1). Retrieved July 19, 2014, from SAINT Integrated Vulnerability Scanner: http://www.scmagazine.com/saint-integrated-vulnerability-scanner/review/3087/

Schwartau, W. (1999, July 7). *Lying to hackers is okay by me: Part 9 of 9" 7 July 1999*. Retrieved august 26, 2014, from http://www.nwfusion.com/newsletters/sec/0705sec2

Sheth, C. (2011). Performance Evaluation and Comparative Analysis of NetwrokFirewalls. *IEEEInternational Conference on Devices and Communications (ICDeComm)., III(9),* (pp. 1-15).

Shirbhate, R. S., & Patil, P. A. (2012, January). Network Traffic Monitoring Using Intrusion Detection System. *International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 1, ISSN: 2277 128X* .

Sommerville, I. (2006). *Software Engineering.* ISBN 978-0-321-31379-9.

Stiawan, D., Shakhatreh, A. Y., Idris, ,. M., Bakar, K. A., & Abdullah, A. H. (2012, June). Intrusion Prevention System: A Survey. *Journal of Theoretical and Applied Information Technology, Vol. 40 No. 1* , pp. 44-54.

Technet. (2001, September 4). *Virtual Private Networking: An Overview*. Retrieved July 22, 2014, from Microsoft Technet: http://technet.microsoft.com/en-us/library/bb742566.aspx

The GreatestOpen Source Software of All Time. (2009, August 17). Retrieved July 19, 2014, from InfoWorld: http://www.infoworld.com/d/open-source/greatest-open-source-software-all-time-776?source=fssr

Thomas, K. (2006). *Beginning Ubuntu Linux: From Novice to Professional.* Apress.

Van Den Berg,et al. (2002, January 11). Retrieved July 20, 2014, from Nessus F.A.Q.: http://www.nessus.org/doc/faq.html

Vyatta. (2011). *Intrusion Prevention Systems Web Filtering.* Canada: Vyatta inc.

Wes Noonan, I. D. (2010). *Firewall Fundamentals.* ISBN 1-58705-221-0.

Youssef, A., & Emam, A. (2011, December). Network Intrusion Detection Using Data Mining And Network Behaviour Analysis. *InternationalJournal of Computer Science & Information Technology (IJCSIT) Vol. 3, No 6* , pp. 87-98.