NETWORK FORENSIC FRAMEWORK FOR MANAGING SECURITY INCIDENTS

PETER KIPRONO KEMEI

A Thesis Submitted to the Institute of Postgraduate Studies of Kabarak University in Partial Fulfilment for the Requirements for the Award Doctor of Philosophy in Information Technology Security and Audit

KABARAK UNIVERSITY

NOVEMBER, 2024

DECLARATION

1. I do here by declare that:
    i. This thesis is my own work and to the best of my knowledge, it has not been presented for the award of a degree in any university or college.
    ii. The work has not incorporated any material from other works or a paraphrase of such material without due and appropriate acknowledgement.
    iii. The work has been subjected to processes of anti-plagiarism and has met Kabarak University 15% similarity index threshold.
2. I do understand that issues of academic integrity are paramount and therefore I may be suspended or expelled from the University or my degree may be recalled for academic dishonesty or any other related academic malpractices.

Signed: _____ Date:_____

Peter Kiprono Kemei

GDI/M/1199/09/13

RECOMMENDATION

To the Institute of Postgraduate Studies:

The thesis entitled "**Network Forensic Framework for Managing Security Incidents"** written by **Peter Kiprono Kemei** is presented to the Institute of Postgraduate Studies of Kabarak University. We have reviewed the thesis and recommend it for acceptance in partial fulfilment of the requirement for award of Doctor of Philosophy in Information Technology Security and Audit.

Signed: _____ Date:_____

Dr. Joel Cherus

Lecturer, School of Science, Engineering and Technology

Kabarak University

Signed: _____ Date:_____

Dr. Moses Thiga

Senior Lecturer, School of Science, Engineering and Technology

Kabarak University

COPYRIGHT

ACKNOWLEDGMENTS

The journey of my research studies have been long, tough and lonely but on the way there are persons who deserve my heartfelt gratitude. I would start my gratitude by forever being grateful to the Almighty God for the peace, presence and providence throughout my studies. God present and grace kept me moving to the end. God provide all what I need for the study. I gave all glory and honour to God.

I would also give special gratitude to my supervisors, Dr Joel Cherus and Dr. Moses Thiga from the school of Science Engineering and Technology of Kabarak University, for the guidance and advice they has provided throughout my time as PhD student. I have been extremely delighted to have supervisors mentor and supported me so much about my work. I would also like to thank all members of staff of school of science engineering and Technology of Kabarak University who helped me during my studies. I would give my gratitude to members of staff of institute of postgraduate studies of Kabarak University for the support they give me over long period of studies.

I express my gratitude to my family members starting from my wife and children for the continued support and encouragements all through during the entire period of study work. I am indebted to them for their support. Thank you all and God bless.

DEDICATION

I dedicate this work to my late parents, Daniel Kirui and Hellen Kirui, my dear wife Ednah Kemei and my children; Ian Kiplangat Rono, Brian Kipchirchir Rono and Amos Kipchumba Rono for the great support and encouragements

ABSTRACT

Network forensics is a science of determining and retrieving evidential information in a computer networked environment about a criminality in such a way as to make it admissible. The established computer networks forensic field lays strong foundation for network forensics as standard security frameworks, tools and techniques are in place for detection, collection, preservation and presentation of evidence phases. However, little has been done to address challenges in examination, analysis and investigation phases. The challenges identified on these respective phases were identification and correlation, multidata fusion, trace back and attribution to source of incident. The study objectives were to investigate, develop and evaluate a network forensic framework which addresses the challenges in examination, analysis and investigation phases. The research methodologies were interrogative literature review, quantitative approach and evaluation based on datasets prototype implementation which addresses the challenges in examination, analysis and investigation phases. The proposed technique in examination phase was identification and correlation. The identification provided attempts made in compromising a system and assist during reconstruction of intruded information. The correlation validated the particular intrusion and guide in decision to proceed with investigation. The techniques resulted in confirmation of DDoS, Portscan and XSS attacks dataset. The proposed techniques in analysis phase were combination of multidata fusion security sensors and integration algorithm. Sensors relies alerts attacked network events evidence which was subjected to confusion matrix and FAR metrics to validate the evidence accuracy. The Algorithm resulted in minimizing evidence file size from 100% to 92.96% saving the system storage capacity by 7.04%. The proposed techniques in investigation phase were trace back and attribution techniques based on ASDPM, DIRM and marking algorithm. The techniques resulted in marking and logging of attacked packets or hybrid both towards particular source of attack and recorded accurate attached evidence based on evaluation metrics set by ISP.

Keywords: *Network, Forensic, Framework, Examination, Analysis, Investigation*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS AND ACRONYMS

AAPM	Advanced Authenticated Packet Marking
AAST	Authenticated Autonomous System Traceback
ACID	Analysis Console for Intrusion Databases
AD	Attack Diagnosis
AIDF	Analytical Intrusion Detection Framework
AI	Attacker Identification
ANN-PCA	Artificial Neural Network and Principal Components Analysis
AR	Attack Reconstruction
AS	Autonomous System
ASBR	Autonomous System Border Routers
ASBDPM	Autonomous System based Deterministic Packet Marking
ASBEM	Autonomous System Based Edge Marking
ASN	Autonomous System Number
BASE	Basic Analysis Security Engine
BS	British Standards
CFRaaS	Cloud Forensic Readiness as-a-Service
CGI	Common Gateway Interface
CSS	Cross Site Scripting
DCI	Directorate of Criminal Investigation
DDOS	Distributed Denial of Service
DEC	Digital Evidence Custodian
DERM	Deterministic Edge Router Marking
DFIF	Digital Forensic Investigation Framework
DFRWS	Digital Forensic Research Workshop
DGA	Data Generation Agents
DIDSDFM	Distributed Intrusion Detection System based on Data Fusion Method
DigForNet	Digital Forensic in Networking
DNF	Dynamical Network Forensics
DNS	Domain Name System
DPM	Deterministic Packet Marking
DPM-RD	Deterministic Packet Marking Based Redundant Decomposition
DRDC	Defence Research and Development Canada

DRIM	Deterministic Router and Interface Marking
ERA	Eliminate Redundancy Algorithm
FTP	File Transfer Protocol
GA	Genetic Algorithm
GBF	Generalised Bloom Filter
GNF	General Network Forensics
HTTP	Hyperactive Text Transfer Protocol
ICMP	Internet Control Message Protocol
IDENT	Identification protocol
IDS	Intrusion Detection System
IEAAS	Intrusion Evidence Automated Analysis System
IEFAF	Integrated E-mail Forensic Analysis Framework
IFERD	Information Fusion Engine for Real-time Decision-making
IP ID	Internet Protocol Identification
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IPS	Intrusion Prevention Systems
IRPCM	Incident Response Probabilistic Cognitive Maps
ISO	International Organization for Standardization
ISP	Internet Service Providers
I-TLA	Investigation-based Temporal Logic of Actions
I-TLMC	Investigation based Temporal Logic Model Checker
LAN	Local Area Network
LCA	Log Collection Agent
LDPM	Logging and Deterministic Packet Marking
MA	Marking Agent
MAC	Media Access Control
M3L	Multi - Metric - Multi – Link
MANETs	Mobile Ad-hoc Networks
NFATs	Network Forensic Analysis Tools
NFS-FLES	Network Forensic System based Fuzzy Logic and Expert System
NIDS	Network Intrusion Detection System
Ns-4	Network Simulator version 4

NSSA	Network Security Situation Awareness
NTE	Network Traffic Exploration
PBN	Probabilistic Boolean Networks
PCA	Principal Component Analysis
PNFEC	Portable Network Forensic Evidence Collector
PNM	Path Number Marking
PRJVDAM	Privacy Violation Detection and Monitoring.
PPM	Probabilistic Packet Marking
RIM	Router Interface Marker
SBL	Session Based Packet Logging
SCAR	SPIE Collection and Reduction
SBPM	SYN Based Packet Marking
SNF	Strict Network Forensics
SNTCH	Simple Noval IP Traceback using Compressed Headers.
STM	SPIE Traceback Manager
SPIE	Source Path Isolation Engine
SRDM	Smart Recording and Data Mining
STOP	Session Token Protocol
TANDI	Threat Assessment of Network Data and Information
TCP/IP	Transport Control Protocol/Internet Protocol
TOE	Target of Evaluations
UDP	User Datagram Protocol
VAST	Visibility Across Space and Time

OPERATIONAL DEFINITION OF TERMS

Attacker Identification is the ability to accurately pinpoint the source(s) of the attack or infection.

Attack Reconstruction is the process of inferring which communications carry the attack forward.

Autonomous System is collection of connected Internet Protocols (IP) routing prefixes controlled by network operator using a single domain that present a common well-defied rule or policies to the global network or internet

Confusion Matrix is a mathematical metric technique based on belief function and has the capability to combine diverse variety of evidence through its rule of combination. The rule of combination is used to fuse information from multiple sensors and obtain high level of evidence.

ForNet is a distributed logging mechanism used in network forensic over global networks.

Framework: A layered prototype implementation indicating what kind of processes, steps or phases should be built and how they are interrelated in accomplishing common objective.

Fuzzy logic is a set of rules and functions that can operate on imprecise data set, but the algorithms still need to be coded

Incidents id define as unexpected network events that affect normal network operations and threat information security.

MATLAB: Software that provides an environment where statistical analysis, session analysis and protocol analysis can exchange data.

Redundant Data/Information: Multiple copies of the same information which are stored in captured or stored in more than one place at a time.

OpenBSD is UNIX operating systems default functionality used to secure accessibility, packet capturing, file encryption, preservation of evidence, disk formatting and wiping.

Palantir is a software environment that supports the collaborative response and investigation process.

OSCAR is an optimization methodology exploiting spatial correlation in multicore design spaces

TOPO is a topology path used to identify attacker enabling router to records the packet signature and predecessor information.

Venn diagram is a graph organiser that visually represents the similarities and difference between sets objects or numbers and cab have overlapping circles which compare and contrast two sets.

CHAPTER ONE

INTRODUCTION

1.1 Introduction

This chapter gives a briefly introduction and background information that informed the study of network forensic and frameworks that manages security incidents. It outlined statement of the problem, purpose of the study, the objectives of the study, research questions, and research hypothesis. It further proceeds to significance of the study, scope of the study, limitations of the study and assumptions of the study. Network forensics deals with determination and retrieval of evidential information in a networked environment about a criminality in such a way as to make it admissible (Alharbi, 2019). It attempts to capture traffic data, evaluate, analyse and provides evidence that characterises attacked or mischievousness attributes (Atonu,2020).

According to (Sikos, 2020) network forensics analysis tools are used in monitoring network traffic and determining if there is an anomaly in the traffic. It also ascertain whether there is indication ofany form of attack. If an attack is detected, then the nature of the attack is also determined. Haddadi (2022) and Tinubu (2022) argues there need for a novel approach enables investigators to traceback the origin of the DDoS service attack and events that took place. A single packet based on deterministic and probabilistic router packets marking logs specified by the packet marking algorithms in designated perimeter points of interests.

The power of numerous available open-source network forensic analysis tools available can be incorporated so that investigator can have an advantage over the attacker. The storage to handle bulky volumes of network traffic data and computing power to analyse the same are available at cheaper rate. In the argument of (Nickolaos,2022)intelligent

satellite deep learning network forensic framework for smart satellite networks system proliferates of the network criminalities and decrease network attack rates. The ultimate goal was to provide sufficient evidence to allow the perpetrator to be prosecuted based on smart and clear admissible evidence.

Network administrators are permitted to screen networks traffic, collect entire information about abnormal traffic, network intrusion security incidents investigation and generating an appropriate incident reaction. NFATs can paint a general picture of all the events happening on the network. They can decrease the time sent on evidence gathering and data analysis. Ghabban(2021) established comparative analysis of network forensic tools and network forensics processes. However, as we advance network in home and business, there is need to advance network forensic view from local host to the network and web application level. It is necessary into consideration these transitions into concepts, designs, models, frameworks, prototypes capabilities and implementation.

1.2 Background to the Study

Network forensics is a scientific determination and recovery of evidential information in a networked setup about an attack using an approach that makes it permissible. It established investigation process in reaction to the intruder involvement of security incidents communal. It involves capturing, recording and analysing of network actions in order to determine the source of attacks using comprehensive tools and techniques according to (Sirajuddin, 2021). The network data traffic was identified using intrusion detection systems collected from existing network security sensors tools. This data was examined for attack classification and scrutinised to traceback the intruders. The procedure can convey out insufficiencies in security products, which can be applied to guide placement and enhancement of these tools. The methodology gathers the required evidence for incident response and investigation of the crime. Network forensics

expedites recording evidence for examination and assist in understanding the intruder's techniques and tricks of attacking the networking (Mei, 2024). It provides to understand the tools used by the intruder and new techniques in which edge defences were evaded. According to (Dalal, 2021), network forensic information convey inadequacies in the current sensors network security tools and application. These tools can be hardened to become robust enough to stand the onslaught of many hybrid and zero day attacks.

There are many frameworks proposed for investigation based on network traffic attacks scenarios in network forensics. According to DFRW enhanced by (Anita, 2021) network forensics framework comprises of the seven phases as follows: "identification, preservation, collection, examination, analysis, presentation, and investigation. Identification phase identifies an incident from indicators and defines the type of intrusion. Preservation phase separate, protect and preserve the formal of physical and intrusion evidence. Collection phase record the evidence from physical scene and duplicate evidence using acceptable standards and procedures. Examination phase deals with comprehensive organised examination of evidence linking to the alleged attack. Analysis phase determine significance, reconstruct fragments of data and draw conclusions based on evidence found. Presentation phase summarise and provide explanation of conclusions while investigation phase handles attribution of attack or crime to a particular host or network with valid proofs. Network forensics was usually useful to wired environs, was concentrated on IPv4, and associated protocols with network layer of the TCP/IP protocol suite.

In steganography forensic, numerous intrusions use "light" procedures of cryptography to extract the recognition of intrusion patterns to supplement challenges in examination phase, analysis and investigation phases which then would have been simply captured by network forensic security sensors and investigators according to (Dalal, 2021). In

honeypot forensics, honeypots are positioned to be compromised and deliver information on the black hat's procedures and tools, before and after the attack on the honey pot. New systems of root kits, Trojans, and possible zero-day abuses can be revealed. An enhanced understanding of the areas of importance and concealed links between black hat's sides can be examined (Koroniotis, 2020).

IPv6 forensics Internet provides mischievous manipulators a short-term safe refuge, as actions are poorly logged and examined. Numerous permitted tunnel brokers offer humble, moderately unidentified connectivity (Ordabayeva,2020). The systematic review transformation from IPv4 to IPv6 takes time when matching the two protocols in order to inter-operation contrivance. This twin stack arrangement will convey novel security weaknesses and challenges, which will require network forensic analysis (Abdullah, 2021).Wireless network forensics corporations are implementing wireless technology at a swift bound and the incidence of data leakage and attacks was persistently increasing. There is abundant essential for profiling operators' actions highlighting the necessity in detecting network cyber-attacks using an integrated statistical approach (Bouyeddou, 2020). There is a strong deficiency of tools and measures for forensic computing investigations to efficiently handle wireless devices. Hence, there are numerous forms of exploitation that overlook examination and analysis phases (Aymen, 2020).

Ghabban (2021),demonstrated that forensic and processes predicted an impending predicament in network forensics as numerous viewers have recognised the persistence of recent tendencies. There necessity to make network forensics research more effective completes the design of new concepts for data demonstration network forensic handling and investigation. In the application layer network forensics, intrusions have moved from network layer and transport layer to application layer of the OSI and TCP/IP protocol suite respectively. Intrusion on web security comprise of buffer overflows, SQL injection

CSS, IP spoofing etc. Reliable network incidents evidence are not identified and correlated by most of the current tracing forensic frameworks from packet structure being transmitted over the network according to (Athanasios, 2019). This impact negatively when decision making to proceed in forensic investigations and quality to admissible evidence.

Cloud forensics requires a transformation to organisation and security procedures regarding network access, remote wireless access, the usage of the data over a web browsers, privacy and examination mechanisms. Recording systems, managing and integrating in what way data is protected on a hired computer system that can be anywhere in the world is challenging task to implement. The difficulty sequences of inter-linkages among the internet service providers and the consumers' offers an abundant ground for intruders. Network forensics DDoS attacking techniques and defence mechanisms in the networknecessitates a current analytical approach posits (Vishwakarma, 2020). Intelligent network forensics structure reconstructs attacks set-ups and makes intrusion incidents attributions necessitate knowledge about attacks signatures, data intrusion evidences, effects and goals. Delinquent resolving knowledge that defines in what way the system can use sphere knowledge to evaluate mischievous actions is critical and necessitates current strategy method of ontology-based smart sound digital forensics analysis for web services forensics analysis according to (Aymen,2020).

1.2.1 Network Forensic Frameworks Shortcomings

The shortcoming of network forensic framework according to (Sirajuddin, 2021),(Tiwari, 2021) and (Ghabban, 2021) are categorised based on phases according to DFRWS model 2001 and reviewed by (Anita, 2021).

Identification: Intrusion must be recognised instantly and initiate network forensics process. The network events, which are malicious, must be identified. Future and zero-day attacks must be predicted based on the common attack features. The hacker groups have common types of attacking tools and the frequently utilised techniques.

Preservation: Network traffic evidence is very volatile and must be identified and preserved immediately, otherwise it is lost forever. Most network security tools do not produce hash values for captured data or utilise the same hash algorithms resulting in inconsistencies. Integrity of collected data has to be preserved so that the captured data will pass stringent legal procedures and qualify as evidence in a court of law.

Collection: Identification of real time network traffic transmitted by high-speed networks, without network traffic packets being dropped or lost, is an important challenge. Full packet captures will result in a very large amount of data. The process can be made efficient by collecting useful data only. Data collected may be reduced by filtering the data according to rules customised for a specific purpose. Network security devices must be able to handle unique input formats and produce different output formats. They must also facilitate universal time synchronisation and display time in different formats and time zones between the devices.

Examination: Packet identified are examined to capture protocol details, which are manipulated by intruders. This information is correlated with attack events and the compromise is validated. Validation of attack takes the process to the investigation phase. Packets are reorganised into individual transport-layer connections between machines and the attack behaviour are analysed by replaying the attack.

Analysis: End-to-end and connection encryption technology checks captured network traffic from being analysed. Logging data from different locations can give

reconnaissance of the attacking behaviour. The analysis of the aggregation of the data sets, which are from multiple sources, such as firewalls, IDSes and sniffers, can build the chain of the clues and display the full scene of the crime.

Presentation: Numerous network security devices do not have the capability for visually scrutinising the network and captured traffic. Documentation necessities every phase in directive to guarantee that all securities have been considered and that no confidentiality intrusion have taken place.

Investigation: IP traceback approaches can track a stable stream of unspecified Internet packets back towards the original source of attack. These techniques do not depend on knowledge or collaboration from prevailing ISPs along the route. The intruders may promotion the intruders' shortest time and use merely insufficient packets making the traceback process challenges.

The preservation, collection and presentation phases have been extensively studied and researched (Johnson,2021) and (Sikos,2020) proposed standard procedures and tools are in place for identifying, preserving and collecting intrusion evidence The presentation and decision phases work more closely with the legal system and follow admissibility procedures. Less studies and research has been done on examination, analysis and investigation phases to strengthen the network forensic frameworks.

1.3 Statement of the Problem

Computer forensic lays a strong foundation for network forensics. There are standard procedures and tools already implemented to detect, collect, preserve and present network security incidents evidence. Additionally, there no clear procedures on how to examine, analyse and investigate network security incidents as examination, analysis and investigation phases still experiencing forensic challenges. DCI (2020) computer

forensics lab overall function is to examine, analyse and investigation all electronic devices related cyber enabled offences reported so as to investigate evidence for presentation in court of law for prosecution purposes. According to (Muniu, 2020) the PwC survey cybercrime was implicated in 34% of all fraud occurrences in Kenya due to unclear clear procedures on how to examine, analyse and investigate network security incidents evidence.

The examination phase lacks effective mechanism to identify, correlate and validate the features of packets that have been manipulated by attackers. The useful network events are not identified for detecting the attacks. The various protocols features being manipulated by attacks are not listed. Correlation of the attacks features with possible attacks scenarios are not performed. The attack evidence are not identified and there no validation done before making decision to proceed with investigation analysis. Effective examination phase framework is not in place to identify attacks features and evidence incidents packet captured.

The analysis phase presents attacked packets and alerts, which are not admissible since there lacks of reconnaissance performed from various security sensors. The attack information and alerts are not taken from various security sensors as no single security tool gives comprehensive alert information. Attacked information are not considered from tools comprised network for reconnaissance. Data fusion of these attacked information and statistics are not validate to ascertain accuracy and admissible attacked form of evidence. There no data redundancy performed to eliminate duplicate evidence and data integration performance on capture evidence by existing analysis phase framework. These challenges makes existing analysis phase network forensic framework not able to present and provide admissible evidence.

The investigation phase has a challenge in determining the source traceback and hence unable to attribute attacker to a particular host within a network. Analysis of alerts, logs and network traffic does not lend to particular the source of attacks. Suspicious source address can be determined in the analysis phase but IP spoofing will hide the true about attacker. Traceback to the source of the attack using IP address is major challenge. These challenges deter analysis phase network forensic framework not able to present the source of security incidents to particular source making forensic network evidence inadmissible.

These challenges deter the principles of evidence, which states that evidence should be admissible, authentic, complete, reliable and believable, and the criteria for admissibility of scientific evidence. This study therefore aims to develop a framework that would address the challenges inherent in examination, analysis and investigation phases. This would enhance general forensic principles of evidences and the criteria for admissibility of scientific evidences.

1.4 Purpose of the Study

To develop a network forensic framework for managing security incidents to enhance the challenges that are inherent in examination, analysis and investigation phases in existing frameworks.

1.5 The Objectives of the Study

i. To investigate the challenges of the examination, analysis and investigation phases of network forensic frameworks in managing security incidents.

ii. To analyse the network forensic security techniques used to address the challenges of examination, analysis and investigation phases of network forensic frameworks.

iii. To develop a network forensic framework that incorporates network forensic security techniques that addresses the challenges of examination, analysis and investigation phases.

iv. To evaluate the effectiveness of the developed network forensic framework in addressing the challenges of examination, analysis and investigation phases based on derived metrics and computer simulations.

1.6 Research Questions

i. What are the challenges of the examination, analysis and investigation phases of the current network forensic frameworks in managing security incidents?

ii. Which network forensic security techniques are used to address the challenges of examination, analysis and investigation phases of network forensic frameworks?

iii. How can a network forensic framework that addresses the challenges of examination, analysis and investigation phases be developed?

iv. What is the effectiveness of the developed network forensic framework in addressing the challenges of examination, analysis and investigation phases challenges based on derived metrics and computer simulations?

1.7 Research Hypothesis

i. There are challenges in the examination, analysis and investigation phases of network forensic frameworks in managing security incidents.

ii. There are .network forensic security techniques used to address the challenges of examination, analysis and investigation phases of network forensic frameworks

iii. Network forensic framework that incorporates network forensic security techniques in addressing the challenges of examination, analysis and investigation phases can be developed.

iv. Developed network forensic framework that is effective in addressing the challenges of examination, analysis and investigation phases based on derived metrics and computer simulations.

1.8 Significance of the Study

The existing network forensic frameworks lack consistence due to increasing volume of network intrusions, diverse and inaccurate nature of captured evidence because of challenges experience in examination, analysis and investigation phases. The knowledge gained from this strengthen the investigation in network forensics by networks forensic practitioners, lawyers, computer forensics investigators, computer ethnical hackers and security agencies.

The enhanced network forensic framework for managing security based on network security technologies aims to improve the quality of examination, analysis and investigative phases to reveal the creditable attacked evidence that are reliable by network forensic security investigator by aggregating similar security incidents as well as filtering the low quality security incidents. Network technologies security sensors implemented to reduce the security incidents dimensionality and optimize the performance. This technique is able to facilitate improved network security incidents classification admissible results.

The proposed network forensic framework for managing security will help to examine, analyse and investigate the identity of specific network incidents of a network attack and improve the performance of anomaly detection system identification and correlations,

analyse of network security incidents based on defined false positive rates and tracking of source of attack based on IP attribution technique. The proposed network forensic framework for managing security integrates the security selection techniques (chi square, confusion matrix metric or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed framework when analysing, tracing and attributing source of attacked vectors of evidence need for investigation and validating the attack. To optimize the performance of the overall network incidents. Unlike other forensic frameworks which requires updating rules frequently to discover attack strategy which are less practical and required high costs due to large database and labour intensive the proposed framework has capability of facilitating a deeper faster examination, analysis and investigation of admissible attacked networked evidence which are vital to invigilating network forensic security personal and other concern parties.

1.9 The Scope of the Study

The scope of this study is restricted following to the following limitations:

Performing real attacks in real networks generated by network security sensors, monitoring techniques and alerts as datasets over a specified period is not realistic. This work examine protocols attributes features used in examination phase for identification and correlation for attacked network traffic, analyses of the KDD datasets and USNW-NB15datasets as well as analyses metrics used in analysis phase and IP attribution tracking technique in investigation phase widely used benchmarks. These network security techniques identified therefore shares the limitations with other published research works which use publicly available standards data permitting to draw direct comparisons.

This research focused on examining, analysing and investigating the network security incidents detected and captured by network security techniques to predict intrusion methodologies as a guide in development of Network Forensic Management Framework (NFMF). NFMF is responsible to plan and develop effective response framework mechanisms. Hence, development of the responsive system is excluded in this work.

The improvement on quality of network security techniques evidence based on exclusion of false positive, invalid and data redundancy alerts. Numerous limitations of current network security forensic techniques that lead to the research of identification and correlation such as protocol attributes and real time response are not persistent addressed mainly in examination phase. The identification and correlation of network traffic such as normal or abnormal and classify the attack type such as probe, user to Root attack, port scan, DDoS, cross site scripting, remote network attacks are not consistent addressed. The KDD cap dataset is evaluated metric in analysis phase is based on five measures, detection accuracy (or) True Positive Rate(TTR), False Positive Rate (FPR), Precision, Total Accuracy (TA), and False Alarm Rate (FAR) which some time has limitation depend on key factors.The diversity of attacked networked evidence has great negative impact in determination and consolidation cohesive timelines through numerous sources resulting to inaccurate, unpredictable and unclear outcomes by the IP traceback and attribution techniques used in investigation phase.

1.10 Limitations of the Study

i. Network security techniques generate huge volume of low-quality dataset evidence and in different format produced by distributed network security sensor systems which may result to inaccurate output. To overcome this limitation the recommendation is implement used of diversity multiple network sensors tools due to varying rate detection of similar security incidents.

ii. The popular dataset KDDs Cup 99 analysis technique is more absolute and may not be reliable to analysis some attacks. To mitigate this limitation the recommendation is implement other dataset analysis techniques like USNW-NB15 and WEKA which are more precise and current.

iii. The diversity of attacked networked evidence has great negative impact in determination and consolidation cohesive timelines through numerous sources resulting to inaccurate, unpredictable and unclear outcomes. To over this limitation to examine, analysis and investigate network traffic in real time basis.

1.11 Assumptions of the Study

The new frameworks incorporate ISO/IEC two main assumptions according to the current ISO/IEC 27001:2018 procedure.

i. The network forensic practitioner is authorized, trained and qualified with specialized knowledge, skills and abilities for performing attacked evidence acquisition, handling and collection tasks.

ii. The network forensic practitioner observes the requirements that their actions should be auditable (through maintenance of appropriate documentation), repeatable where possible (in that using the same tools on the same item under the same conditions would produce the same results), reproducible where possible (in that using different tools on the same item would produce substantially similar results) and justified.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter is organised as follows, it start with overview of network forensic, investigate of network forensics, challenges of examination, analysis, investigation phase and their limitations. It also span gaps of knowledge based on existing challenges and limitations of network frameworks, framework network security techniques and conceptual framework for managing forensic network security incidents.

Network forensics is a scientific determination and recovery of evidential information in a networked setup about an attack using an approach that makes it permissible. It expedites recording evidence for examination and assist in understanding the intruder's methodology there are many frameworks proposed for investigation based on network traffic attacks scenarios in network forensics based on seven phases that includes "identification, preservation, collection, examination, analysis, presentation, and investigation according to DFRW (Anita, 2021) forensic network roadmap review.

Computer forensic field lays a strong foundation for network forensics as standard procedures and tools are in place for detecting, collecting, preserving and presenting attacked evidence. Little has been done to address the challenges found in examination, analysis and investigation phases. These challenges deter the principles of evidence, which states that evidence should be admissible, authentic, complete, reliable and believable, and the criteria for admissibility of scientific evidence. There was need for availability of a network forensic framework that addresses challenges for the examination, analysis and investigation phases strengthen the forensic mechanism.

2.2 Investigation of Network Forensics

The concept of network forensics caters for data found across a network connection mostly ingress and egress traffic from one host to another. Network forensics tries to analyse traffic data logged through firewalls or IDS or at network devices like routers and switches. Rajput (2020) attributed integrated cybercrime and cyber security model with aim of discovering the origin of network security incidents intrusion or additional problematic network incidents. Network forensics comprises of screening of network traffic and determination of an irregularity in the network traffic and determining whether there are indicators of network intrusion. If intrusion was identified, then the actual type of the intrusion was also identified. Network forensic methodologies and framework permit investigators to trace source of attacks. The critical objective provides adequate evidence that permit the intruder to be prosecuted in court or interest parties by the forensic investigator (Amal, 2022).

2.2.1 Standards and Guidelines in Network Data Acquisition

Ali(2020) suggest a novel three-tier intrusion detection computer forensic evidence framework that implements short-term changes within the environment in which digital forensic practitioners work and consistent long-term standardised approach forensic investigation activities. In order to achieve this, the three stages of the (Al-Dhaqm, 2020)model addressing the challenges of before and after incident identification problem of overarching principles which are then reflected in policies which leads to specific procedures and techniques. This hierarchy moves from industry best practice principles through to organisational polices for ensuring quality and efficiency and culminates in those activities that are likely to be introduced or modified to cater for changes in technology such as the introduction of a new software tool or the release of a different type of storage device. Common areas for identifying the details of best practice are

published standards and guidelines. There are two main standards references and guidelines described in subsequent section that are associated on how digital data and evidence processes are managed.

The main international body that is relevant to standards within the field of digital forensics is the ISO according to ISO/IEC (2018) which is an international standard setting body that is composed of representatives from 164 countries.ISO publishes technical reports, guides and other technical literature, normally based on the output from special committees that are established for a particular purpose.

The British Standards Institute has produced a standard, BS 10008, whose title, 'Evidential weight and legal admissibility of electronic information specification, suggests that it may be related to the acquisition of digital evidence. However, the standard relates to the production of electronic documents that may be required as evidence of business transactions and provides advice for practices and procedures involving information management systems**.** In contrast to the limited number of standards associated with the field of digital forensics. There are several guidelines on particular areas such as law enforcement, electronic discovery and commercial forensic and incident response on digital forensics.

2.2.2 Network Forensics Systems Classification

Classification of network forensic systems is based on the following characteristics:

Purpose: GNF emphases on improving security. The network data flow is examined and intrusion patterns are exposed. SNF comprises stiff legal necessities as the output attained will be used as an indication for network crimes prosecution (Abdullah, 2021).

Packet Capture: These are network devices implemented to network packets transiting through particular point and afterwards each packet is examined this led requiring large

storage capacity. These are also network devices implemented for purposes of analysing each packet in memory and only certain traffic data is stored for future analysis, necessitating a faster processor according to (Abdullah, 2021).

Platform: There is network forensic hardware installed with NFATs. They are used for capturing network traffic, analyse and display the outcomes on a graphics used interface for interpretation. NFATs are capable of examining captured packets stored or net-flow logs records.

Time of Analysis: Moneymaking network forensic analysis applications software comprise of real time network reconnaissance, signature-based abnormality detection, information analysis and for investigating network forensic incidents. Numerous open source software tools are developed for post-mortem examining packet captures. Full traffic packet captured by networks sensor sniffer tools, stored in a local host and analysed off line in future date or time.

Data Source: Stream based systems gather statistical data established on some principles within the network packets traffic as it transit through the network. The network devices gather this information, pass it to a stream accumulator to store, and scrutinize the data. Packet established systems comprise of full packet captures at several network entry points. The packets are gathered and stored for profound packet examination.

2.2.3 Network Forensic Analysis Tools

NFATs assistance in examining the network intruder and exploitation of network resources as well as host, predict intrusion targets in future time, implement risk valuation, assess network performance, regulate hardware and network protocols utilisation, collects data from multiple security tools as well as helps in protection of

intellectual property. The requirements of multi-source multi-domain data fusion for cyber-attack detection in power systems as illustrated by (Sahu,2021). NFATs can paint a general picture of all the events happening on the network. They can decrease the time sent on evidence gathering and data analysis.

NFATs gather the complete network traffic, permitting users to examine network traffic depending to their requirements and determine substantial features concerning details of traffic. NFATs together with IDS and firewalls makes extended span of network traffic preservation accounts for speedy examination, analysis and investigation. The intruded traffic can be rerun and intruders' traffics can be examined for mischievous actions (Sirajuddin,2021). NFATs enable party capture network traffic to be observed as specific transport layer association among machines, which allow the operator to examine protocol layers, packet contents, and transmitted data and excerpt traffic forms among numerous machines. There are particular NFATs existing that deliver consistent data achievement and powerful examination abilities. A partial list of (Sirajuddin,2021)with a brief description is given in appendix I.

2.2.4 Network Forensic Process Models

Established analytical procedures and approaches are used for the customary computer forensic chastisement. Nevertheless, as we advance computer networked applications, wireless networks in home and business, this necessitated intensification of forensic examination from local host to the computer networks as well as web applications. It was important to consider this shift into conceptions, strategies, models and frameworks. Numerous network forensic models has been proposed to cater for the computer networked environs as from the year 2001 where some may have been implemented and

some may not have been implemented to date. The word ‘model’ was use to entail a theoretical exemplification steps involved in network forensics investigation.

The initial proposal in network forensic science was seconded as the leading aims in the first DFRWS 2001 and a framework enhanced roadmap reviewed by (Frank, 2021). The network forensic framework phases comprised of identification, preservation, collection, examination, analysis, presentation, and investigation. In the position of (Anita 2021) framework reviewed (Reith, 2013) model lead to an abstract digital forensic model, which is not reliant on a specific technology or criminality. Preparation and methodology approach phases have been included and evidence was taken back to investigation phase.

An event response methodology network process model, which was accurate and simple, was proposed by (Prabhjot,2019), an original response phase determines the event and design of a response approach was included as additional in capturing forensic evidence. Analysis phase and collection phase were incorporated in investigation phase as in the previous proposed models. Resolution phase and reporting phase were termed as presentation phase. It recommended enhancements, modifications and extensive span resolutions.

An investigative network forensic model process inspires a comprehensive laborious examination, guarantee appropriate evidence management and decreases unintended errors improvised by (Hamza, 2023).Separately from the conjoint phases, valuation phase validated the event and assessment was reserved whether to withstand with the analysis. Gathering, lessening, association and examination phases organised the records with the least set with greater possible evidence. Influence and evidence phases obtains simplified understanding or in simplified form.

Al-Dhaqm (2020) recommended integrated incident response model for database forensic investigation that recognizes the procedures using physical stepwise process examinations. Readiness phase guarantees processes and arrangement was geared up. Review, examination and collection phases collect and manage the data. Analysis phase was comparable to reconstruction phase in other models. All the captured evidence is recorded in documentation phase in this process model.

Victor (2020) proposed holistic digital forensic readiness framework for IoT-enabled organizations. The framework process model examinations cybercrime, which signifies the information streams of traffic and seizures complete analysis. Responsiveness was the leading phase, which proclaims analysis, and approval was reserved from interior as well as exterior units. Preparation comprises approaches, procedures and distribution is done for managing forthcoming examinations and processes.

Victor (2020) reviewed a comparative analysis of digital forensic readiness model using CFRaaS as a baseline.The readiness model was established on the corporeal criminality examination procedure. Traceback and dynamite innovative phases were included in this new model. The two phases have other sub-phases comprising of authorization, investigation, and communication and reconstruction open-handed simplicity and holistic to the main phases.

In addition, (Nejma, 2020) recommended multi-tier process hierarchical model, objectives based network framework for analytical process in compared to the single tier advanced imperative network process models. The network process model comprises of the joint phases in first tier, if streamlined assessment and a theoretical understanding. These mutual phases comprise of sub-phases located in lowest tiers that offered simple, granularity process model guided by guidelines and goals. The sub-phase assembly for the Data Analysis Phase was accessible and examinable.

Abdullah (2019) reviewed a behavior analysis of distinct operating systems that detect and identify the host in IPv6 networkand integrates legitimate network intrusions concerns. Mohammad classified eight main roles that accomplish the essential principles for forensic analysis specifically, investigation, consistency and relevancy. Mohammad listed six queries for every role that attempts to answer the following questions what, why, how, who, where and when. These roles and queries are integrated into the Zachman's framework for initiative approach and FORZA composed. The process model automated by intensifying a data attainment scripts creator which determinate gathering of significant information from the network traffic records and systems logs.

Ilyas & Alharbi(2022) reviewed the mapping process among the procedures or events and outcomes for each phase in DFIF. A research of the present network forensic process models designed and then the mapping built based on the outcome. The authors group and merge the same activities or processes that offer the similar outcome into a suitable phase. The mapping process designed in order to balance the process on achieving the overriding goal that can produce concrete evidence for presentation in a court of law. The phases are preparation, collection and preservation, examination and analysis, presentation and reporting, and disseminating the case.

An unboxing network digital forensic investigation process which consists of the following phases: preparation, examination, categorizing, cybercrime, determining investigation precedence, examining damaged crime targets, illegitimate, outlining consultant and investigation, tracing suspects, examining injurer crime targets, summoning intruders, additional analysis, writing criminal description and presentation of report documentation was proposed by (Sunde, 2022).

Subsequently, (Richard,2019) proposed an advanced Investigative process model (the SDFIPM) for conducting digital forensic investigations, encompassing the 'middle part'

of the digital investigative process, which is formal in that it synthesizes, harmonises and extends the existing models. Among the first network forensic investigators to discuss network forensics taxonomy, conceptual model, legal principles, key techniques, canonical processes and its accessory facilities, system architecture and deployment as per (Ghabban, 2021). Moreover, (Ghabban, 2021) suggested that all process models discussed above are appropriate to computer forensic investigation as well as network forensics in a generalised process model form. The generalised model had the following phases: capture, copy, transfer, analysis, investigation and presentation. The model lack examination phase and complex procedures involved during copying and transfer of admissible evidence.

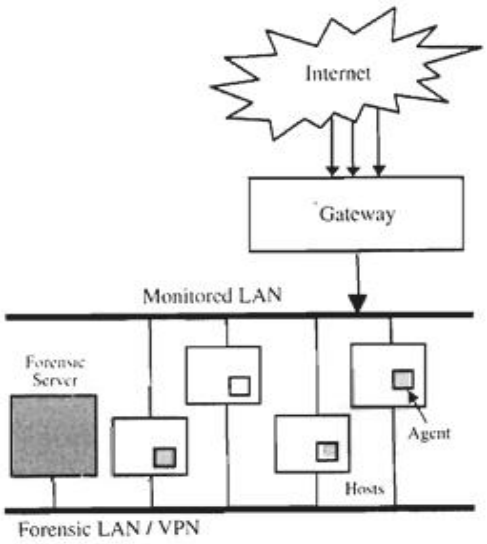
2.2.5 Network Forensic Frameworks

Many variant network forensic process models that have been proposed in previous section consist of diverse phases. Investigators have technologically advanced several network process frameworks, which some have been implemented partially and some implemented fully. These network forensic frameworks have been grouped based on software computing, distributed system, honeypots, heterogeneous systems and aggregation systems.

Local area networks, web applications systems and internets are scattered all over globally and networks intrusion actions are recorded at numerous localities by the peers systems. This prompts necessity of centralised server collections of records, examine and scrutinize them for admissible evidence. The figure1 below showed a universal arrangement for the distributed systems based frameworks. The forensic server mainly stores forensic evidence captured within the networked setup through monitored LAN restricted forensic LAN/VPN consisting of hosts and agent.

Figure 1

A Universal Arrangement for Distributed Systems Based Frameworks



Accordly to (Ghaleb,2019) proposed an agentless endpoint security monitoring process DNF network framework system based on the multi-agent and artificial immune theory. The framework provides a generic agentless endpoint framework for security monitoring of computing systems. The hosts are accessed by the monitoring framework running on a central server. The monitoring framework is separate from the hosts for which the monitoring is being performed and the various security models of the framework performs data retrieval and analysis without utilizing agents executing within the computing hosts. The monitoring framework retrieves transparently raw data from the monitored computing hosts that are then fed to the security modules integrated with the framework. These modules analyse the received data to perform security monitoring of the target computing hosts. As a use case, a real-time intrusion detection model implements and detects abnormal behaviours on computing hosts based on the data collected using the introduced framework. The framework utilises host, agent-proxies designs to gather the intrusion leads and examine the evidence. The benefit of the framework is it represent the dispersed configuration of network and Internet. The

framework lacks inadequate in detecting the network features appropriately, specific modules are not functioning and complex to implement.

Toraskar (2019)proposed a theoretical framework efficient computer forensic analysis using machine learning approaches digital forensics that ensures the forensic readiness of the evidence available for the investigation process. The framework helps organisations reduce the cost of the investigation process because it provides manageable components and live analysis. The components proposed in the high-level view make the implementation and automation of the framework more difficult to create automated tools.

Alharbi (2019) proposed a functional process model to map the digital investigation process. The model has two components. The first one is the proactive digital forensic component, which includes five phases: proactive collection, event triggering function, proactive preservation, proactive analysis, and preliminary report. The second component is a reactive digital forensic component that also has five phases: identification, preservation, collection, analysis, and final report. The proposed proactive component is similar to the active component of the multi-component process such that they share the same reactive component process. The components proposed in the high-level view make the implementation and automation of the framework more difficult to create automated tools. The components proposed in the high-level view make the implementation and automation of the framework more difficult to create automated tools. The limitation of framework fails to identify the implantation requirements and has limited capabilities because it does not include all the anti-forensic techniques, which could affect the ability of the components to resolve the cybercrime in an efficient manner.

Koroniotis (2020) recommendednetwork forensic framework based on deep learning for Internet of Things networks. It was established on dispersed methods as long as an incorporated boards for spontaneous network intrusion traffic evidence gathering and capable data storing, informal incorporation of recognised ascription approaches and an intrusion designation diagram generation devices. The process framework was established designed agents and network proxies at different locations. Designed agents gathers, minimises, processes, analyses and storage of data. Network proxies at different locations create the intrusion designation diagram and implement marching storage examination. Framework objective provides a technique to gather, examine and storage of forensic information. It was capable of identifying network security incidents evidence automatically and swift responses to network intrusion. The framework requires and demand large storage systems of both real time and post attack evidence storage implementation.

Ghabban (2021) recommended a distributed supportive network forensic framework system based on client server design. Server identifies network events, forms planning topology records, filters, tips, alters the network events torrent into record values, data mining forensic database, and repeats network actions. In additional it does survey of network events, intrusion network statistical examination and network visualization. The disseminated manager peers incorporate data from network firewall, intrusion detection systems, honey pots and isolated traffic. The objective of this network framework is removal of mischievous packets established based on configured setup filter procedures, examining the complete compliant record to determine the possible mischievous and repeating mischievous for network forensic investigation. It can determine the outline of the invader and attain evidences intended for advance analysis. The frameworks incur lack of conclusive evidence interpretation due to unified tools uniqueness.

Ghabban (2021) advanced a previous framework as dispersed manager established on actual network invasion forensics system. The objectives of the framework include record system information collecting, adaptive internment of network traffic, and dynamic reaction for tentative forensics, incorporate forensics data and storage of past network mismanaged of network traffic pattern. The four features in the framework-based system include forensics network server, network forensics agents, network observer and network forensic investigator. Forensic network forensic agents' machines are used for data collecting, data abstraction and data protection transference. Network observer was a packet internment machine which adaptively internments the network traffic. Network forensic investigator was the network examination machine. Network forensics server incorporates the forensics data, examines the traffic evidences and presents an analysis platform on the network forensic investigator. The framework advance the analysis of a network intrusion event and increase facilitate occurrence intrusion response but the framework became cumbersome and complexity in the analysis phase implementation.

Abdullah (2021) proposed generic proactive IoT Cybercrime evidence analysis model for digital forensics. The model focused on the classification of evidences in advance based on its significance and relation to past crimes, as well as the severity of the evidence in terms of the probability occurrence of a cybercrime. The model save time and effort during the automated forensic investigation process. The model creates implementation complexity during examination, analysis and investigation phase due to varying forensic tools sensing tools on real time traffic identification process.

Samir (2022) recommend computer and network security intrusion detection system using mobile agent. Based on JADE agents. An attached device acting as a network server presents network forensics-agent, reports portable agents to examined diverse

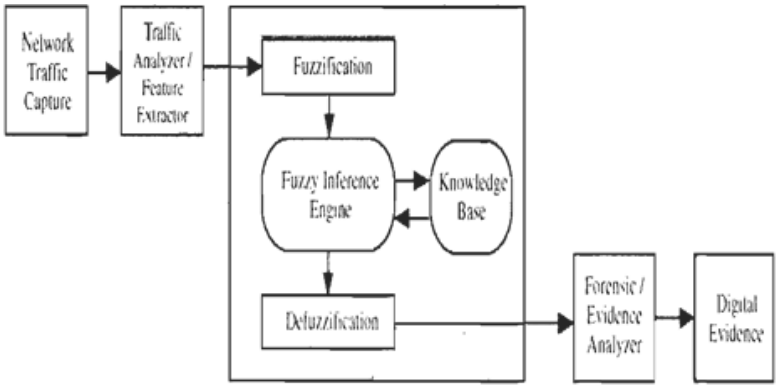
locations. They collect network traffic records, inspect them and outcome results presented on a graphic user interface systems. The interface permits the forensic investigator to identify traffic stream to be captured and examine the subsequent network actions revealed. The results presents gathering of network traffic from distributed diverse systems by means of mobile agents. The application was accessible, decreases network data traffic, reports particular socket of failure and delivers proactive monitoring of network traffic. The framework requires regular updates batches for mobile agents to analysis net mischievous attacks.

Thomas (2023) reviewed DigForNet where the functional component Syn-App, developed to review and recall networks actions specific time and a centralised forensic domain server accomplishes a set of SynApps in that domain. A centralised forensic server collects requests from external domains, processes the same together with SynApps and passes inquiry outcomes back to the investigator after verification and authorization. The general design comprises a network filter, outline engine, outline controller, conformation manager, safety manager, database manager and inquiry processor. Substantiation leads of criminalities can be establish in packet structure main headers that contains source and destination IP address or application reliant packet data units. DigForNet recognises network actions corresponding to TCP links creation, connection logs facts, scanned ports and procedures trace filters to give other leads actions. The framework is only suitable for examination, analysis and investigation connection oriented networks implementation.

The soft computing based framework applications analysis identifies network traffic and categorise comprised traffic. Neural network and Fuzzy tools are utilised for authentication of intrusion occurrence. A universal fuzzy logics network based frameworks system is shown in Figure 2.

Figure 2

Universal Fuzzy Network Based Frameworks System



Paulo (2020) developed a universal fuzzy hybrid model system for the construction of expert systems in malware detection.It delivers examined network traffic for a forensic professional reducing the time spend and lows charges of forensic investigation. The network forensic framework comprises of six modules namely traffic analyser, defuzzification, fuzzy inference engine, knowledge base, defuzzification defuzzifies and forensic analyser. Traffic analyser module internments network traffic streams and evaluates it by means of assembling related evidence in nature. Defuzzification module defines the membership functions based on rules for each fuzzy set and a crisp value of degree of membership. Fuzzy inference engine module derives per output linguistic values using aggregation and composition. Knowledge base module mainly composed of storage unit mainly used to store rules written for various attacks using linguistic variables and terms by the fuzzy inference engine. Defuzzification defuzzifies module output values into crisp values and the forensic analyser module validates implementation of the occurrence of an attack.

Sahu(2021) recommended multi-source multi-domain data fusion for cyberattack detection in network computing systems based on ANN-PCA. The main shortcoming

encountered in network investigation multi domain data fusion forensic framework is enormous data traffic stream for storage that requires examination. Abstraction of significant evidence decreases storage by associating the evidence with intrusion. ANN-PCA procedures are utilised towards detecting potential abuses, abstract structures and construct signatures for fresh intrusion. Grouping is complete by means of algorithm using FAAR to determine relationship guidelines and compute the PCA principles. Grouping precision proliferations and evidence to be store mass declines subsequently evidence abstraction is achieved using ANN- PCA.

Sikos (2020) and Sirajuddin (2021) reviewed packet analysis for network forensics comprehensive surveythat encounters the shortcoming of significant evidence to be recorded and examined meant for computer network forensic. The Neuro fuzzy resolution is established on fuzzy logic and ANNs utilised for evidence diversity interested in usual and irregular traffic streams. ANNs achieve information processing by learning from data and specifying solutions that can be taken. Fuzzy logic is implementing to produce a rate of association to diverse activities so that intrusion evidence is identified.

Ghabban(2021) proposed the comparative analysis of network forensic tools and network forensics processes. The process relies on actual time, phases and five specific phases which are tailored based on (Sirajuddin,2021)framework. The five phases include preparation, detection, incident response, collection and preservation. The four phases in the second group act as post-investigation phases, which include the examination, analysis, investigation, and presentation phase. The first five phases work proactively because they work during the occurrence of the cybercrime saving time and cost during the investigation process. All the activities of network forensics are included in this

framework; the present research adopts the phases of this model as a baseline to show how the analysis phase integrates with the other phases.

In Ghabban (2021) all phases requires a certain amount of time to accomplish its processes with proposed forensic tools. Each phase works in real time; thus, the phases require the same amount of time and processing cost to accomplish their forensic processes. Given that the other four phases work reactively, it is assume that they require more time and processing cost compared with the first five phases. The reason for this assumption is that reactive phases work after the cybercrime happens; therefore, the required amount of time and cost increases during the investigation process.

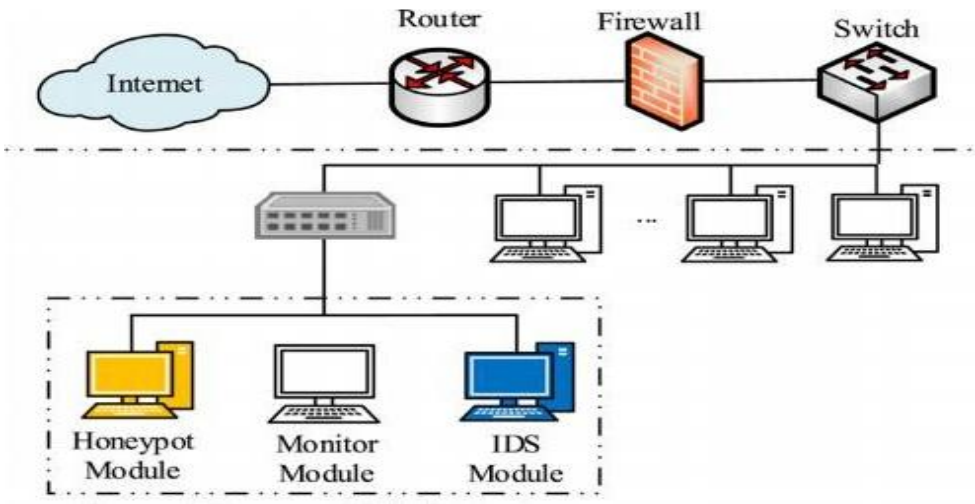
Abirami(2023) proposed artificial intelligence-based proactive network forensic framework, that combines a reactive and proactive framework and assure collected from various network security sensors are reliable and consistent where the proactive part use machine learning based classification algorithms to forecast the attack. It also emphasizes integrity and confidentiality by proposing an encryption method that encrypts the proactive module's report before decrypting it in the reactive module.

Honey pot based system frameworks a security resource whose value is determined by its ability to be explored, exploited, or hacked. The main Intention forth system is to be examined, attacked, and potentially exploited by attackers. It is primarily goal is detection and reaction tool, with minimal utility in preventive. Honey pots do not block specific intrusions or the transmission Instead, they gather data and detect attack trends. Defenders can respond to this evidence by constructing stronger defences and countermeasures against future security threats. The framework are used to gather evidence or information and to learn as much as possible about attack patterns, attacker purposes ,motives and widely utilized programs launched by them.

Honeypot module are built on remote virtual machines networks. Honeypot based system frameworks are used to appeal the intruders to enhance their procedural approach to facilitate observation and examination in order to progress protection mechanisms as shown in Figure 3 below.

Figure 3

Honeypot Based System Framework



Amal(2022) recommend as a fraud implement to gather traffic stream about black hat actions and acquire their procedures so that defence and protection mechanisms can be conveyed. Honey net based systems framework appeals intruders to move in a host by matching recognised system defensive weakness. As soon as an intruder infiltrates a honey trap, traffic are apprehended to identify and logged in main activities. This information can be used to outline the strategies and techniques used by the intruders placing the detectives in an invasive approach.

Two structural design, parallel and serial, simplify the forensic examination. The serial design implements honey trap in between hosts system and internet system. Acknowledged users are clarified to the source systems and black hats are confined within the honey trap. The parallel design permits honey trap to be autonomous of the

source system. When the system identifies the incidence of black hat, the forensic vigilant system triggered. If the intruder is identified, forensic progressions are triggered on the honey trap and host systems. Once the intrusion is restricted, the examination procedure begins to establish the uniqueness of the intruder on the host system.

A framework for intrusion patterns' detection in enhanced honey netattack pattern discovery information recommended by (Koroniotis, 2020). Their objectives was based on pattern discovery of groups of network leading to sharing of different varieties of extremely related attack patterns contained by an intrusion records established. They strategy a malleable gathering tool and examine one precise feature of the honey net facts and time sequence of intrusions. Mischievous network traffic is achieved from the disseminated established responders of Honeypots. Time signature is implemented as a principal gathering artefact and intrusion patterns are revealed using intrusion leads resemblance.

Intrusion is identified as sequences of associates. Polymorphic and zero-day intrusions are identified or established on resemblance to other types of intrusions and information from the Honey net is used in invasion discovery determinations. The gathering process does piece of collection, abstraction, outlines a pattern closeness quantity and collections related patterns. The outcome of gathering function to time sequences examination permits discovery of botnets and worms in the traffic flow gathered by honeypots.

Furthermore,(Wang, 2020) examined computational Intelligence for Information security network forensic framework that established evidence in reaction to internet and cybercrime intrusions. The two main odds of the framework are complication when examining new traffic and numerous amount of information to be examined. The framework incorporates automatic network forensic tool to output information recorded by numerous tools into a sole system that abuses computational ability to decrease

human interference. Open source networks forensic tools are implemented for gathering information. The information collected by numerous tools in single stage is considered and altered for use for other supplementary tools in the subsequent stages.

Time wastage and error susceptible to procedures are automated and determined. The data sets are segregated; system is proficient and then verified. Honey nets encounter the potentials of network forensic analysers but they cannot be used for analytical determinations, as evidence created is not effective by investigating agencies and parties.

Himanshu(2022)review network forensic framework and trends that identify common phases in previous frameworks and match to produce a mapping of digital investigation framework that comprises of three phases. The phases consitsts of acquisition, analysis and presentation. The framework misses the pre-incident preparation phase which is very important requirment and also the details previded is unclear in relation to each phase. The framework provides a only summary of the results and conclusion without subjecting it to Carrier & Spafford framework or daubbert criteria that is based on physical crime scene investigations.

Al-Dhaqm(2020) built upon the examination mobile forensic investigation process model framework based on nine phases (Identification, Preparation, and Approach strategy, Preservation, Collection, Examination, Analysis, Presentation and Returning) in relation to digital evidence acquisition. These components are seen as being a complete representation of the process undertaken by a digital forensic investigators and practitioner. Al-Dhaqm model is a good representation of the forensic process since the model includes all forms of digital storage which is important for presentation purposes. This concept is supported by (Rachana, 2022) who point out that upon receiving notification of an event the techniques to be used in the investigation will be part of the response. The main shortcomings in this model that have been identified includes: The

model approach is too general to be applied practically in digital evidence acquisition. It very difficult to determine a way of testing the model and it too cumbersome. The model assumed the chain of custody to be automatically incorporated without reference to the model explicitly. The model has significant shortcomings in practicality and ability to accommodate changes without becoming excessively complex.

Humaira(2020)proposed formal knowledge model for online social network forensics framework based on the physical crime scene. The framework consists of five phases that include the sub-phases, i.e., readiness (operation and infrastructure readiness), development (detection and notification and confirmation and authorization), physical crime scene investigation (search and reconstruction), presentation, and digital crime scene investigation phase. Each phase in this framework has a clear goal and requirements to achieve the expected results but concentrate on digital crime scene in the same way physical crime scene and incorporate as both as of digital forensic evidence which leads to conflicting processes in terms analysing evidence.

Abdulalem (2021) reported that the integrated phases, when combined, are insufficient to investigate real cybercrime cases because these phases have not mention the completeness of each phases. Computer Forensic Field Triage Process framework introduced by (Ricci,2019) has six phases which includes planning, triage, usage or user profiles, chronology or timeline, Internet activity, and case-specific evidence phases. The framework provides the identification, analysis, and interpretation of cybercrime evidence within a short time frame without the need to generate a complete forensic image of the lab. The main limitation experienced by the model is suitability for investigating all types of cybercrimes because evidence is very difficult to distinguish and collect.

Cyber Forensics and Comparative Analysis of Digital Forensic investigation was presented by (Singh, 2019). The frameworks maps process and output produced by different phase in the network forensic frameworks that have been examined from previously proposed frameworks and represented a comparative mapping of all frameworks. This model is more comprehensive than the other IDIP framework because it encompasses almost all the investigation activities but the model needs more evaluation in terms of scalability to ensure that it analyses evidence efficiently. The model also is based on single-tier processes, focuses on the abstract layer in each phase. The advantage of single-tier processes is that they produce unambiguous outputs. The main limitation of single-tier processes, as reported by (Koroniotis, 2020) reduces the pattern discovery attack path and hasten investigation when more details are required from the user.

Sunde (2022) addressed this limitation by proposing a multi-tier, hierarchical framework to guide digital investigations. The framework has six phases, namely, preparation, incident response, data collection, data analysis, presentation, and incident closure. The framework introduces objective-based phases and sub-phases to each layer in the first tier with the ability to add more details in advance to guide digital investigations, especially in data analysis. The main limitation of this framework is that it is incomplete and requires a more methodical approach to identify the objectives of each layer. The proposed general network model that performs standardization processes to cover the fundamentals of network forensics. The model includes six steps namely capture, copy, transfer, analysis, investigation and presentation. These steps are divided into three process stages. The first stage identifies the basic techniques to preserve the security process. The second stage describes the status of the transformation process. The final stage provides the architecture of the proposed network forensic system to indicate the

integrity of the system components. The analysis step is the most comprehensive and sophisticated step. The model does not include the analysis of network traffic, which remains an open issue.

Al-Dhaqm (2020)proposed integrated incident response model that clarify the definition of network forensics based on database management evidence of interrelated evidence , which groups all the existing processes into three stages, namely, preparation, investigation, and presentation, which are implemented as guidelines in network forensics. The proponents of the framework claim that the framework offers scalability to allow the addition of more required stages in the future. However, understanding how the framework addresses all phases of network forensics in the main stages is very difficult in clarification.

Ricci (2019) proposed block chain distributed digital forensics investigation framework that focuses on the legal rules and participants in the organization rather than the technical procedures. The framework solves complex problems by integrating the answers with the questions what (the data attributes), why (the motivation), how (the procedures), who (the people involved), where (the location), and when (the time) questions. The framework includes eight rules: case leader, system or business owner, legal advisor, security or system architect or auditor, digital forensic specialist, digital forensic investigator or system administrator or operator, digital forensic analyst, and legal prosecutor. The main drawback of this framework is that it is human dependent. It requires more tools to conduct a network forensic analysis and to provide accurate results in the investigation phase.

Nickolaos (2020) proposed network forensic framework based on deep learning for Internet of Things networks based on optimization and deep learning. The framework consists of multiple phases of a digital investigation process, focusing on smart home

deployments. It includes a method, based on Particle Swarm Optimization (PSO) for selecting the hyper parameter values for the deep neural network (DNN) and shows through acquired metrics, that the produced DNN achieves very high accuracy while minimizing false alarm rates. The framework increases the complexity of multiple phases and investigation stages.

According to (Richard, 2019) a similarity exists which is generic in that it can be applied in the three fields of law enforcement, commerce and incident response. The two present a common process model for both incident response and computer forensics to improve the investigation phase. The model includes a set of steps grouped into three main phases, consisting of pre- analysis (detection of incidents, initial response, and formulation of response strategy), analysis (live response, forensic duplication, data recovery, harvesting, reduction, and organization), and post-analysis (report and resolution). Incident response is conducted in the model during the actual analysis. The procedures and methods of incident response are unclear in terms of the type of evidence that is utilized to analyse the incident. No standard method of detecting and collecting evidence exists, which produces insignificant evidence and affects the accuracy of the incident response.

Multi-source multi-domain data fusion for cyberattack detection (Sahu,2021) consists of ten phases: investigation preparation, classifying cybercrime and deciding investigation priority, investigating damaged (victim) digital crime scene, criminal profiling consultant and analysis, tracking suspects, investigating injurer digital crime scene, summoning suspect, additional investigation, writing criminal profiling, and writing report. The model presented the block diagram without any technical details or methods to manipulate with these phases. This indicates that the main focus was on the number and

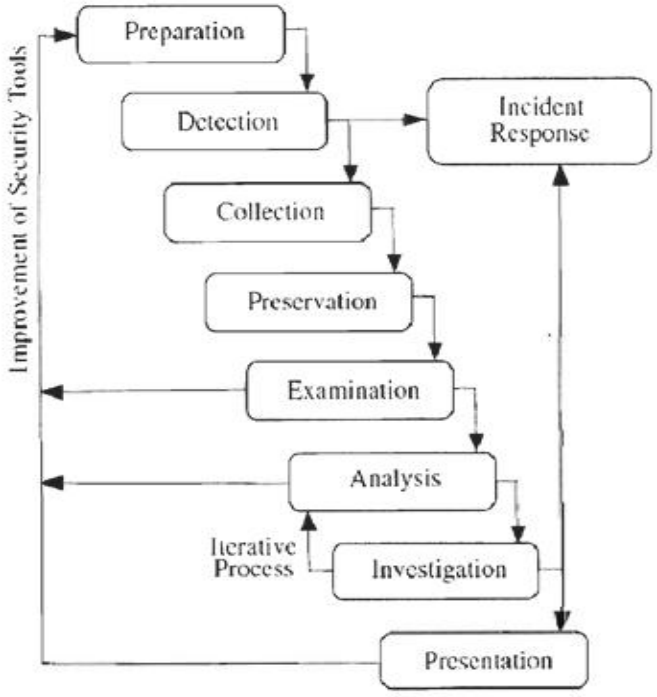
the type of the network forensics phases rather than how it works and how they conduct the outcomes.

Rachana(2022) simplified the digital forensics framework as the collection and preservation phase, examination and analysis phase, and presentation and reporting phase, clarifying the output for each phase. The main challenge is the output of the examination and analysis phase which doesn't mention the methods and techniques which could be used to conduct the output from this phase. Despite the framework being integration of previous models, it omitted details such as the proactive preparation requirements from the multi-source multi-domain data fusion for cyberattack detection systems framework (Sahu, 2021).

Network forensic analysis comprises many phases and various network forensic security tools are used for precise phases (Thomas, 2023) as shown in process framework for network forensics in Figure 4.

Figure 4

Existing Process Framework for Network Forensics



The review of network forensics process framework and a proposed systematic model recommended by (Thomas,2023) based generic network framework proposed by (Sirajuddin, 2021) on as shown in figure.4 divides the phases into two groups. The first group relies on actual time and includes five phases: preparation, detection, incident response, collection, and preservation. The four phases in the second group act as post-investigation phases, which include the examination, analysis, investigation, and presentation phase. The first five phases work proactively because they work during the occurrence of the cybercrime saving time and cost during the investigation process. The first phase prepares the network forensic software and legal environments, such as the IDS firewalls, packet analyser, and authorisation privilege. The second phase detects the nature of the attack by generating a set of alerts through the security tools. The third phase extends from the detection phase; it initialises the incident response based on the type of the attack and organisational policy. The fourth phase, which also extends from the detection phase, collects network traffic through suitable hardware and software programs to guarantee the maximum collection of useful evidence. The fifth phase backs up the original data, preserves the hash of all trace data, and prepares a copy of the data for utilisation in the analysis phase and other phases.

The other four phases of this model work after the investigation phase and act as a reactive process begin with the examination phase to integrate the trace data and identify the attack indicators; the indicators are then prepared for the analysis phase. The seventh phase is the analysis phase, which reconstructs the attack indicators by soft computing or through statistical or data mining techniques to classify and correlate the attack patterns. The phase aims to clarify the attack intentions and methodology through the attack patterns and provides feedback on how to improve the security tools. The eighth phase is the investigation phase, which aims to identify the path of the attack and the suitable

incident response based on the results of the analysis phase. The final phase presents and documents the results, conclusions, and observations about the cybercrime. All the activities of network forensics are included in this model; the present research adopts the phases of this model as a baseline to show how the analysis phase integrates with the other phases.

In generic framework, each phase in the first five phases requires a certain amount of time to accomplish its processes. Each phase works in real time; thus, the phases require the same amount of time and processing cost to accomplish their processes. Given that the other four phases work reactively, it is assume that they require more time and processing cost compared with the first five phases. The reason for this assumption is that reactive phases work after the cybercrime happens; therefore, the required amount of time and cost increases during the investigation process.

i. **Preparation.** Clarification of moderate amount of forensic preparation in an organisation can mitigate the impact of a major incident and can enable the organisation to obtain restitution. Forensic readiness reduces the per incident costs and improves an organisation's overall security posture when used in conjunction with an information security program. It integrates proper evidence handling mechanisms into an organisation's incident handling capabilities.

ii. **Detection:** Detecting the occurrence of an attack is the goal of the established field of network security. Countermeasures exist for many of the attacks, which take place and regularly keep re-occurring. Many security tools like IDS, intrusion prevention systems, firewalls, network monitoring systems (IPS), network statistic tools exist in commercial and open source varieties. These tools can detect most of the current attacks.

iii. **Collection:** Capturing and storing data packets from networks consume many CPU power and storage capacity resources. Mario (2020) emphasised the development of a network forensic control mechanism which can dynamically adjust the amount of data to be collected on an evidence flow according to the storage capacity level on the storage subsystem. Their solution is able to select an appropriate full collection (FC) and selective collection (SC) margins to minimise data loss associated with storage subsystem saturation while preserving reasonable acceptance ratio of new forensic collection requests.

iv. **Preservation**: Kulandaivel (2022) describes network support for IP Traceback Model in Wireless Sensor Networks Using Quantum Annealing Method. It minimize the quantity of malicious packets entering into the network we put forth a quantum annealing technique to identify and alleviate the DDoS attack. The attack messages are minimized by utilizing client puzzle as a part of the ingress router; the path fingerprint is used at the egress side. Simulation studies prove that the proposed mechanism is optimally successful in recognizing and mitigating the DDoS attacks.

v. **Presentation:** Network investigators are responsible for ensuring that evidence is reliable enough to be admissible in court or to be useful in corporate disciplinary or termination proceedings. Ferguson (2020) gave some basic guidelines to make sure the evidence is protected and that notes made at the time are professional. The evidence must also be presented in a comprehensible fashion to be useful.

The aggregation frameworks strengthen network forensic tools to expedite forensic examination moderately than designing a new tool. Rusydi (2021) proposed a network forensics based on analysis of conti ransomware attack on computer network with live forensic method. The arrangement comprises of three core modules namely capture,

logging and marking modules respectively. The marking module chooses whether a transiting traffic has some criminality evidence due to suspicious IP addresses reports from alerts information by network sensors. The capturing modules time for the marked packets by marking module. They are organised in sequence for consistent transfer to the logging module for storage purposes. The logging module is a system data mining where the intruded packets are stored. It implements three forms of information loggers namely: - sensor loggers utilised for storing alarms, host logger stores information transmitted by capture module and raw logger is used as a backup logger when other loggers fail it implements function of other two loggers. Sebek network forensic sensors implements capture module, according to (Gupta, 2020) snort IDSs implements marking module of packets and Snort barnyard tool, ACID Lab, TCPDump and Sebek implements logging module functionality.

In addition to that, Rusydi (2021), designed PNFEC constructed by means of low-priced open source software and embedded hardware. It functions in three modes –user, server and investigator modes respectively. The portable, convenient and compact device was intended for traffic gathering between a single node and network taking particular approaches of action, prompt distribution and furtive inline action. The Ethernet bridge traffic is wantonly apprehended using numerous caps established by capturing tools and stored in local system.

The system software, setup files, extra software and detective actions records are stored on compacted storage systems. Access controls list, numerous features of the device like start up, programming, configuration filters meant for capturing intrusion data, forensic tools and utilities such as maintaining and conveying the evidence. The PNFEC is simple to setup and control. The network traffic gathered can be stored in encoded method to prevent the integrity of evidence. PNFEC also manages classification of identified

information using TCP Dump to ensure there are no confidentiality exploitations. A script is utilised to generate encryption cryptographic hash of the traffic internment files and conserved using OpenBSD.

A visualisation NTE system developed using DRDC for security incident and Diane (2014) recommends traffic examination. This system associates six main well-designed applications into a single set. They are invasion discovery (anomaly and signature based), packet examination, scripting package, traffic replay, imagining composition and effect valuation. NTE has three main parts with MATLAB as advance setting, small level traffic examination reference library and combined presentation interface. It offers a background where numerical examination, session investigation and protocol investigation can transfer information.

Arulanand (2021)IP spoofing is simply referred as generating forged (fake) IP address by an intruder with intension of hiding identity of sender. This communication enables the attack to progress and fortunately, this communication is visible to the network. The authors outline a high-level vision of an investigative capability for the Internet that permits identification and fine-grained analysis of the communication patterns leading to an attack.

They build the framework on two fundamental components AI and AR (Tiffanie, 2021) recommended a novel framework based on graph representation toward network forensics examination. A classified intellectual framework comprising of two parts - confined local cognitive targets to deduce the practical conditions of network units from localized interpretations and universal cognitive targets to classify essential units from the graph configuration and excerpt collections of compact interrelated members in the intrusion situation. The basic architecture has Evidence collection module, Evidence

reprocessing module, Attack knowledge base, Assets knowledge base, Evidence graph manipulation module, and Attack reasoning module.

In the same field, (Bijalwan,2021) pointed out DigForNet, which is useful to analyse security incidents and explain the steps taken by the attackers. DigForNet uses intrusion response team knowledge and formal tools to reconstruct potential attack scenarios. They integrate the analysis performed by the IRT on a compromised system using the IRPCMs. They also provide a formal approach to identify potential attack scenarios using I-TLA. They cause executable possible intrusion situations using a model checker tool called I-TLC.

In addition, (Sikos,2020) proposed packet analysis network forensics framework for online network forensics, a concept that helps in analysis packets traffic entering and exit the network topology. They designed and implemented a model, Bloodhound, which tags and tracks information between the kernel and the application and correlates symptoms of exploits with high-level data. The tasks include preferentially recording network traffic, searching through these flows after the application has been healed, and replaying the relevant flows to test this repair. The work can be improved by dealing with taint tracking through the application itself.

Moreover, (Jan, 2020) recommended netfox detective novel open-source network forensics analysis tool. That provides a heuristically based engine for traffic processing that can be easily extended using robust parsers. It integrates the following aspects into a single package. It can efficiently develop enormous internment files, remove extraordinary traffic and is capable to authenticate every interpretation. The design architecture includes input/output source, virtual file system, and scanners. It also has the following network forensics modules like the stream re-assembler, packet handlers, and stream dissectors. The model needs to be improved to accommodate large number of

protocols present on the Internet and there is a dire need to flexibly and quickly adapt to these new protocols.

Furthermore, (Qadir, 2021) proposed Applications of machine learning in digital network forensic based on advanced IEFAF. The main components functionality of IEFAF includes the ability to investigate network logs archives and compute the required statistical distribution results. Different visualisation techniques, capability of keyword searching, development of data mining models help classify e-mails in different categories or cluster them according to some undiscovered relationships. Detection of anomalous behaviours by matching the observed e-mail communication with the pre-recorded normal communication model of users, performance of e-mail authorship analysis based on stylometric features, and capability to map selected IP addresses by applying geographical localization technique to determine the physical location of that IP.

2.2.6 Challenges of Examination phase, Analysis phase and Investigation phase

Saeed (2023) emphasis the network technologies emerging such as block chain, big data, artificial intelligence, data analytics, the industrial Internet of Things and cloud computing are critical enablers for digital transformation. Due to extensive benefits businesses accelerating the digital transformation drive. A network security incident has grown into a significant challenge for business and to gain business continuity, organizations need to secure their digital transformation tools, methodologies and artefacts.

A method for the description of the threats for developing protection profiles or countermeasures by introducing the concept of the assets protected by TOE by was proposed also proposed. The security environments consist of assumptions, threats, and

organisational security policies and an editor of the PPM describes the threats. Rachana (2022) proposed a method for applying security engineering to build security countermeasures. It identifies the threats, undesirable event characterized in terms of a threat agent, a presumed attack method, a motivation of attack, and an identification of the information or systems under attack. DDoS threats deplete the network resources rapidly particularly link parameters. Modelling these attacks provides a strong base for analysing the attack characteristics.

Gupta(2021) recommended grid-aware distributed model predictive control of heterogeneous resources in a distribution network: The model is based on real-time systems, stochastic processes, processor scheduling, computational modelling, predictive control, voltage control, reactive power ADMM dispatch, distributed control, model predictive control, and power distribution network. Law enforcement agencies need uses in monitoring, detecting and analysing the network traffic to when investigating network security incidents on real time bases. This may be against the goal of maintaining privacy of individuals whose network communications are being monitored.

Furthermore, (Haider,2022) proposed an enhanced interface selectivity technique that improve the quality of service for the multi-homed node.Capability based alert correlation uses notion of capability to correlate IDS alerts where capability is the abstract view of attack extracted from IDS alerts/alert. To make correlation process semantically correct and systematic, there is a strong need to identify the algebraic and set properties of capability. Fornasini (2019) proposed framework bade on observability and reconstructibility properties of PBN on a finite time interval addressed state. The state update follows a probabilistic rule, while the output is a deterministic function of the state that is investigated under what conditions the knowledge of the output measurements allows the exact identification either of the initial state or of the final state

of the PBN. Time-based notion was added which avoiding temporal ambiguity between capability instances.

In addition,(Javed, 2022) presented a comprehensive survey on computer forensics: state-of-the-art, tools, techniques, challenges, and future directions. The time interval between events promises to reveal many key associations across events, especially on multiple sources. The time interval is then used as a parameter to a correlation function that determines quantitatively the extent of correlation between events. Network security tools can be applied to network traffic and data fusion performed on various output values generated. Many models have been proposed for data fusion of intrusion detection data. They are mostly based on confusion matrix theory of evidence. The models are surveyed to obtain the direction for data fusion in the context of network forensic analysis.

In the same area, (Tiffanie,2021) Machine learning or artificial intelligence for sensor data fusion model. Pieces of evidence from heterogeneous defence systems are fused or combined to detect the attacks for anomaly detection and localisation Yuan (2021). Data fusion techniques effectively combine evidence from multiple sensors or a single sensor used in multiple places. A multi-source fusion system that uses the DS technique to combine beliefs from multiple security tools is investigated.

In addition, (Steffen, 2019) proposed Efficient Attack Correlation and Identification of Attack Scenarios based on Network-model that different types of NIDS. The set of alerts can be partitioned into different alert tracks. IDSDFM consists of alert correlation module, security estimation module, and management & control module. Two types of alert aggregation is done - alerts that make up an attack and alerts that represent the behaviour of a single attacker. According to (Steffen, 2019) DIDSDFM consists of two

layers: lower layer and upper layer. The lower one consists of host and network based sensors, which collect local features, and differentiates easy-to-detect attacks. The upper layer is a fusion control centre, which makes global decision on these events by adopting confusion matrix combination rule.

Humayun (2020) reviewied cyber security threats and vulnerabilities systematic mapping study that highlighted a notion where situation security analysis is made to understand threats and vulnerabilities from intrusion alerts and take appropriate actions. The network security situation elements are analysed, data is fused, and correlation identified using collared petri-net and the confusion matrix theory of evidence. NSSA fuses data from tools of IDS, virus detection system, firewall, net flow records to monitor the network for intrusions and predict course of action. The security events are pre-processed and situation assessment is done through correlation to gather information about the attacks.

Furthermore, Hussein (2021) proposed a pattern recognition approach is applied to network intrusion detection based on the fusion of multiple classifiers. Each member of the classifier ensemble is trained on a distinct feature representation of patterns, then the individual results are combined using a number of fusion rules. Expert knowledge about the characteristics that distinguish attacks from normal traffic can be used to extract features based on content (payload), intrinsic (network connection information) and traffic features (statistics). This evidence is combined to produce a final decision.

On the other hand, Bouyeddou (2020) recommended an advanced IEASS framework that collects the evidences from integrated statistical approach based on cyber-attacks. Alerts of NIDS are treated as primary evidences and logs from vulnerability scanner, network monitors, firewalls and others can be used as secondary evidences. The LCA collects intrusion logs from various sensors, pre-aggregates and adds a signature. LCA has

security event parsers that use regular expressions to automate log aggregation. An ERA is proposed for retaining effective information after removing redundant information.

AIDF proposed for information integration and realisation of a distributive IDS environment with multiple sensors and a mechanism for selecting and integrating the probabilistic inference results to aid most probable forensic explanation.

The probabilistic approach is also used for integrating information from different sensor sources in a distributive NIDS environment. A novel cyber data fusion system (Sahu, 2021) proposed to specifically address the tracking and projection of multi-domain data fusion detection evidence attacks. It uses information fusion to provide situation awareness and threat prediction from massive volumes of sensed data. The system is based on INFERD and TANDI. INFERD efficiently correlates IDS alerts, identifies individual multistage attacks, and provides situational measures of the identified attacks. TANDI fuses information extracted from each attack track estimates, to determine threatened entities and differentiates them by threat scores.

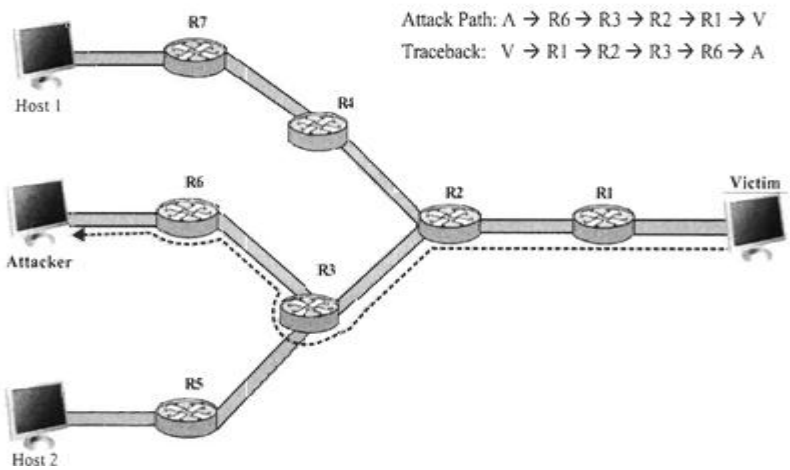
Sheikhalishahi (2022) proposed model based model that provides a technique of Privacy preserving data sharing and analysis for edge-based architectures. The proposed framework computes the best trade-off among privacy and result accuracy, based on the privacy requirements of data providers and the specific requested analysis algorithm. Internet protocol traceback difficult refers to detecting the genuine origin of every packet sent through the Internet. It is an essential approach to comprehend the current intrusion or to examine and features the intrusion in the post investigation stage (Athanasios, 2019) and (Abdullahi, 2020). IP traceback mechanism has no ability to avert and alleviate the intrusion. They can merely recognise the origin of intruded packets. According to (Abdullahi, 2020) the evidence are used to examine post mortem analysis

of the intrusion as shown in figure 5.The path of attack is traced from the victim to source as per participating routers traceback and attribution attack path.

The traceback techniques are categorized as proactive or reactive. A traceback technique is referred as reactive when the procedure is instigated in reaction to an intrusion and investigation start after the incidents has taken place. System connection testing is a reactive procedure, aimed at input restoring and precise overflowing (Rusydi, 2021). These approaches create use of the enormous quantity of information in a DDoS intrusion and create intrusion discovery assessments whereas the intrusion is in advancement.

Figure 5

IP Traceback Mechanism



The procedures flop when the intrusion packets decreases and henceforth they are not appropriate for post-mortem investigation. A technique is proactive while the traceback packets are simultaneously flow or stored as the traffics are transmitted passing the network. Proactive processes comprise of packet logging, packet marking approach (probabilistic and deterministic), combination of both logging and marking approach (hybrid) and Autonomous System level traceback approach.

Packet logging approach at main routers enables proof of identity of the correct source of intruded traffic over the Internet. The main challenge is the handling and storing properties essential at the routers. Kalangi (2021) recommended a hybrid IP traceback mechanism to pinpoint the attacker. The mechanism trackback a sole IP traffic stream via packet logging and has an integrated manager traceback application referred as pinpoint the attacker DGA, collects and reduce agents SCAR. A hash of various IP packet header fields is calculated and recorded in the IP space tables using competent tint filters. When a traceback appeal is made, STM reports the evidence to proper SCARs, which request the pinpoint evidence active router and STM rebuilds the intrusion path using the outcomes.

Arulanand(2021) recommended that every IP packet had path fingerprint that indicated the path through which the packet has traversed in the net- work in the IP header. The server keeps up a mapping table which contained the IP address and its corresponding path fingerprint. Sensors identify the intrusion and direct the tracing appeals. Tracers instigate forwarding nodes; preserve record evidence concerning inbound packets and OSI layer two (data link) identifiers. The tracer matches the record information with data about the packet tracer and discovers a tracing pathway.

In addition, (Sirajuddin,2021) recommended a trace filter established topology that alert particular packet IP traceback scheme using improved OSCAR methodology. TOPO develops router's local system topology evidence for traceback. When a packet passes through TOPO active router, it logs the packet traces and prototype evidence. If the casualty, the casualty's address, captures an intruded packet mark and arrival time are conveyed to TOPO as per a traceback appeal. Entirely reactions from inquired TOPO enabled routers are gathered by TOPO to generate the attacked graph.

The intrusion graph is implemented for additional examination and traceback. Packet marking comprises assigning the routers portion or comprehensive address into the IP packet alongside the attacked pathway. Packets are marked both deterministically or probabilistically. Packets are marked by choosing them arbitrarily with a stable probability PPM or packet might be marked merely as soon as by the inbound edge router DPM. PPM approaches necessitate numerous packets for group of attacked information.

Kulandaivel(2022) recommended the PPM, IP traceback approaches enable the victim to traceback attacks and they will not be able to minimize the attack when the attack is in progress. The casualty can create the attack path containing of entire PPM active routers afterwards it has conventional sufficient packets. The IP ID inside the IP header field is utilised for storing the traceback data. Many variants of PPM have been proposed.

Mauro(2020) recommended packet marking and Markov modeling for IP Traceback. The Traceback method consists in deploying IP tracing technology: the source address in the packet header can be forged, and it needs a security mechanism to determine the attack sources and attack paths when a DDoS attack occurs. The algorithm can reduce the number of packets collected to reconstruct the attack path, especially in situations when an enormous number of counterfeit attack packets exist. It can also identify the correct attack path, and the tracing scheme uses a probability labelling method. The proposed model is a modelling based on classic probabilistic PM algorithms as Markov chains, allowing a probabilistic closed form to be obtained for evaluating of the correct number of received marked packets in order to build a meaningful attack graph. The model creates complexity based on probabilistic similar traced intruder traffic.

On the other hand, (Kulandaivel, 2022) recommended algebraic packet marking that facilities algebraic approaches from the field of coding theory to compute the ideals of

marks as facts on polynomials. Numerous patterns similar to complete path encoding, arbitrary path encoding and edge encoding are implemented. Several attack path-rebuilding approaches are presented and encoded path evidence and stored within IP fragment ID field of the IP format header and decoding is using matrix of Vandermonde. This approach of IP traceback mark mechanism is complex and requires large amount of storage capacity especially during encoding and decoding implementation. Kulandaivel (2022) proposed a switch that is established on PPM. This scheme engages header firmness to intensify the quantity of bits obtainable for addition of traceback evidence. If an early frame is directed with a complete header, consequent frames can be directed without the fixed details being incorporated in the packet header.

In addition, (Shui,2021) recommended speedily internet traceback that takes a packet marking mobile internet installed at routers and path rebuilding algorithms implemented by local peers. Speedily internet traceback packet markings comprise of three features: a section of the hash of the marking IP address of router's, the amount of the hash section marked in the packet and the length field. Casualty implements the hash section and length calculations between the markings in aggregation and its mapping router.

DPM schemes are appropriate for network forensics as traffic of insufficient packets can adequately regulate the origin of the intrusion. They are deliberated in features in the subsequent fragment. Hybrid schemes association records and mark the packets as per (Mingxing,2021) recommended distributed link list traceback established on "mark, forward and store approach". A sole marking packet field is assigned in every single packet. Some router that resolves to mark the packet, stores the present IP address establish in the marking packet field alongside with the packet ID in a superior data structure termed as marking table preserved at the routing table. That point marks the packet by renewing the marking field by its particular IP address and then passes the

packet as normal. The marking packet field aids as an indicator to the preceding router that ensured the marking for the particular packet and router-marking table that comprises an indicator of the preceding marking router.

Fadel (2021) recommended hybrid distributed single-packet Low-storage IP traceback frameworkmechanisms for marking, evidence gathering and traceback handling. The framework permits router to takes a MA for recording the marking evidence into its reserved record database. TSP accomplishes the MAs and accumulates records from them into a centralised log database. ECA is in charge for gathering marking traffic as evidence for intrusion. The13 bit off field and 16 bit ID field are implemented to encrypt the marking evidence, which comprises of 8 bit Old time to live quantity and 21 bit hash quantity of the IP MAC address.

A hybrid sole packet IP traceback based system affixing router ID marking field scheme and calculate log packet details was established by (Abdullah, 2021).Traceback permitted routers inspection packet and a server traceback devising the network topology evidence creates intrusion graph by individual inquiring routers, which comprises of 15 bits ID field. The mark is time stamped congestin the ID packet field. The leftward bit is utilised as recording bit, enabled to 1 if router binds recording. Furthermore, (Fadel, 2021) recommended framework that enables a dispersed single IP traceback based scheme. AS is an autonomous entity of the Internet. The objective of low storage IP track the superior edge linking inbound and out border routers. The 64-bit ID field is accounted for storage of the AS ID, 13-bit field fragment is accounted for storage of router ID respectively, and they have proactive messaging procedures.

In addition, (Dalal, 2021) recommended that every router probabilistically chooses information and creates an iTrace that is flow to the similar endpoint as the traffic for steganography and stage analysis network forensic. Single iTrace reports, created for

each 20000 packets, comprises of router ID, timestamp, preceding and subsequent IP addresses respectively, MAC addresses and particular HMAC verification records. A hybrid Traceback based network forensic technique to identifying origin of cybercrime evidence framework can be as recommend by (Rachana, 2022). This appeal is established by the edge routers, which establish an intent bit in the upstream forwarding packet table.

Suresh (2020) proposed packet marking feasible attack source traceback scheme by deterministic multiple packet marking mechanism. Every edge router marks each packet before the packet enters the network with its uniqueness. DPM implement the 16 bits ID field and the 1-bit ID field is reserved for purpose of marking the packets. The edge routers IP address is fragmented into two fragments with 16 bits respectively. Casualty can restore the address as soon as it obtains both the fragments from the identical router. Single bit is implemented as a flag to specify which part of the IP address is conceded intrusion.

Moreover, (Mei,2024) recommended DERM forensic framework for advanced persistent threat attack attribution through deep learning. Edge router with the 16-bit hash of the 32-bit IP address marks 16-bit ID field packet based on deep learning attack attribution. The attacked networked traffic are stored in database comprising of hashed mark of internal IP address list matched with unique listed.

In addition, (Abdullah,2021) recommended a strong and accessible forensic framework system based on operating system detect and identify the host in IP hashed functions utilised to moderate the possibility of address abstract collisions. Three bits are implemented to differentiate the eight unrelated types of marks and the outstanding fourteen bits transmit incomplete address evidence containing the marks. The system direct each bit of the IP address at least two times and permits compromise among

incorrect positive rate and incorrect negative rate while allowing for lost packet due to congested traffic.

A DPM-RD for IP traceback based multi domain data fusion detection that comprises of information and index sections marking fields was recommended by (Sahu, 2021). Each inbound edge router dispatches its conforming IP address into numerous fragments with adjacent fragments taking particular redundant bits with each system. The IP ID field is marked with unique fragments and redundant dispatches create the address disintegration more scalable while reducing incorrect positives rate packets. Furthermore, (Wang, 2020) recommended a scalable DPM (FDPM) that discovers the actual source of intruding packets. It implements a scalable mark length approach for compatibility to dissimilar network locations and it variations the marking rate allowing to the load of the contributing router by a scalable flow-system based marking pattern.

Autonomous system based traceback is a linked set of single or additional IP prefixes control by single or extra network operators, which has a particular and evidently well-defined routing strategy and hybrid planning using learning and model checking for autonomous systems (Ashutosh, 2020). Every AS is recognised with a universally exceptional ASN is 16 bit integer, assigned and managed as recommended by (Abdullah, 2021) used in the transferring external routing data using Hashed IP address detected and identity by means of operating systems.

Athanasios (2019) proposed AAST AS numbers system that marks packets probabilistically using authenticated and AS marking schemes. The scheme necessities 25 marking bits and the 8 bits for TOS, 16 bits for ID field and 1 unused fragment flag bit. Marking is implemented at ASBR, when a packet is passed to a router in another AS domain where 16 bits are used for ASN and 3 bits for AS distance field. Authenticated

marking accepts a symmetric key structure in every AS with 25 bits AS marking field dispersed a cipher text generated.

Moreover, (Athanasios,2019) reviewed and investigated network frameworks where a single inbound edge routers of every AS mark packets with AS number permitting definite possibility and packets are not remarked by entire extra routers. The 32-bit marking data comprises of four fragments, 16-bit AS_PATH for storage of the altered ASPATH evidence, 1-bit FLAG representing whether the packet remained marked, 3 bits for footage of length of ASPATH and 12-bit hash function of the IP address. Casualty necessitates reception of single few packets to recreate the intruded path route. ASPATH characteristics offer a well-ordered list of ASes to be navigated and verify the path.

On the other hand, (Vishwakarma, 2020)recommended DDoS attacking techniques and defence mechanisms in the networkbased system that compact with the TCP SYN and replication DDoS intrusions. The core objective of the pattern is to avert the intrusions traffic at the inbound edge router that is bordering to the origin of intrusion. The outbound edge router that is linked to the casualty network informs the casualty's particulars comprising of 16-bit hash value of the 32-bit IP address to all additional inbound routers inside the ISP/AS domain. The outbound router authenticates the traffic that is intended to the casualty's network and marks the packet with the exceptional ID of the inbound router. The inbound edge router relates inbound filtering on' the stream intended to the casualty. Inhibition of the intrusion traffic will be complete until the inbound edge routers acquire a reset indication from the outbound edge router.

An AS level distribution of record based 32-bit number based IP traceback as well as AS-level particular packet traceback (AS-SPT) scheme was proposed by (Mario,2020). It records packet précises at the egress routers of contributing ASes and tracks a

particular intrusion packet near its source at the AS-level. Every AS-SPT permitted AS retains an AS traceback (AST) server that examines the process of edge nodes recording the packets. When the traceback inquiry attains from targets, AST inquiries the edge routers and directs the reply back with gathered evidence or forwards the inquiry replication to its previous ASes in case the edge routers have not recorded the packets.

In addition, (Mario, 2020) recommend an AS-level edge network based system that functions on the boundary routers of an AS and constructs an edge network after transmitting evidence with BGP. The marking system supplements information into the GBF of an IP traffic to the routers. The public features in the inform communications of BGP is used to collection end that segment the identical public features. Marking is implemented by an exclusive OR operation logic gate of 8MSBs of 0's, 8 bits TTL and 16 bit AS number.

Router Interface marking techniques are recommended as an atomic entity for traceback as an alternative as compared to router marking itself. Abdullah(2021) recommended interface router marking for IP traceback where a RIM permitted router probabilistically marks every packet with the identifier of one of the hardware interfaces that managed the packet. RIM expenditures a sequence collection of internal router input unique IDs as a universal exclusive track identification. 5 bits are reserved for distance in RIM, XOR functional used 6 bits and 6 bits are reserved for unique ID. A router probabilistic marks traffic by reorganising the reserve field to 0 and replicate the ID of the traffic inbounds interface fields of XOR and ID respectively.

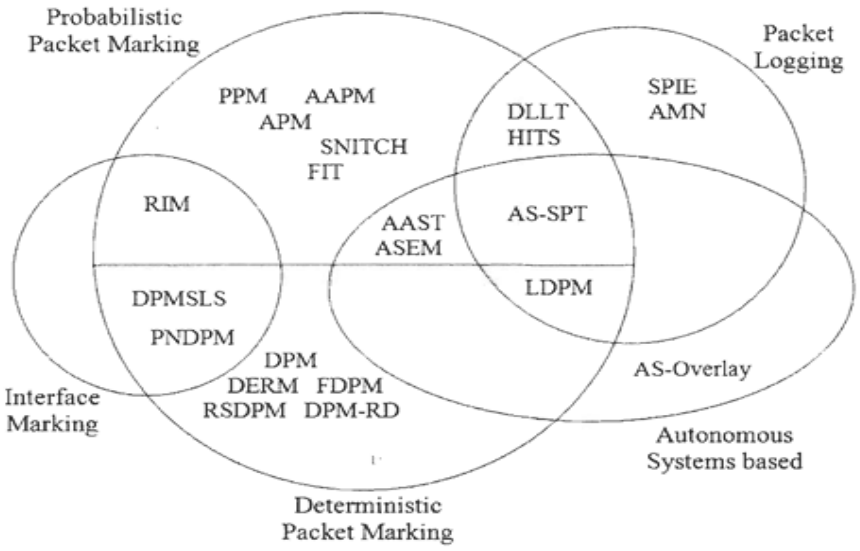
Suresh (2020) recommend an enhancement of the above methods for AD which is an innovative intrusion alleviation pattern, implementing approach of divide-and-conquer, stimulated by the casualty subsequent detection prior detection of an intrusion. The casualty initiates the inbound routers to mark deterministically packets and traceback

source of main intrusion. AD associates the conceptions of packet marking and pushback. Intrusion discovery is accomplished closer to the casualty client and traffic clarifying is performed nearby to the sources of intrusion. Through initiating its inbound routers to deterministically mark packets, the casualty accomplishes to traceback single source of intrusion and facility an AD permitted router adjacent to the origin of intrusion to drop the packets.

They are various types of traceback mechanisms that uses packet marking techniques to traceback and attribution source of attack namely, interface marking, deterministic packet marking, autonomous systems based and packet logging. Each type of traceback mechanism share some similarities and difference across other types of traceback mechanisms. The figure 6 below illustrates traceback mechanism in form of Venn diagram. The big cycles represented in Venn diagram represent traceback mechanism techniques which share major similarities features. The small cycles represents minor features which are shared by various traceback mechanisms. The Venn gives illustrates the comparison and contrast between different techniques features of traceback mechanism. The of traceback mechanism features are determined based on evaluation metric in terms of type of mechanism, packet for traceback, marking field length and processing overhead discussed in Table 18.

Figure 6

Relation between Various Traceback Mechanisms



Kalangi(2021) recommended a hybrid IP traceback mechanism to pinpoint the attacker where routers participate in marking deterministically and updates are based on the specific marking. The entire path information is available in each packet and single packet IP traceback is possible from marked packet path.

In addition, Shui(2021) recommended an improved and legitimate DPM where route numbering is utilised for traceback. There are two types of routers: PNM enabled and DPM enabled. PNM enabled routers bordering the origin of the packet mark every single traffic with the route credentials signifying the route associating them to the enabled DPM routers. DPM enabled installed at the egress of a network subnet to mark every single traffic navigating by the inbound router interface. The casualty cannot only trace and drop the intrusion, however it also acquire precise evidence by the authentic packet marks. The relative amongst numerous mechanisms of traceback was shown in figure5.

Moreover, (Chourasiya, 2024) reviewed network forensic tracebacktracing techniques for IP network forensic traceback and categorized them so that the command of forensics

might be enhanced and the drawbacks of typical instance behaviour and reaction abilities could be counteracted. The emphasis is on their client centred, network centred or, active or responsive, centralised or distributed internal network or global network and numerous reengineering utilities to be implemented.

Furthermore, (Chourasiya,2024) reviewed the STOP contribution in the forensic examination and traceback of a mischievous client. The protocol is centred on the IDENT and is designed to spontaneously trace invaders recording through a sequence of moving gravels. STOP protects the client and application level data related with a specific TCP association and proceeds arbitrary marks. It also permits clients that do not exist in the association sequence to make requirements on behalf of alternative client. STOP transforms the entreaty message to afford additional possibilities and the reaction message to defend confidentiality. The request types permit marks to be spawned alongside the complete route of clients. ID request type stores the username and returns an arbitrary mark. SV type saves the user name and the data associated with the progression. SVREC and ID_REC are the repetitive daemons, which necessitate an arbitrary identifier gathering.

Thomas (2022) recommended the systematic network forensic Investigation model that aim to establish appropriate policies and procedures for practitioners and organizations. The model does classifications based on infiltration detection systems, traceback, distribution models, and attack maps. The model integrates passive methods that do not transform packets, but they collection interpretations for future examination. The model has network screens that transfer with examination platform from side to side a recording component. The recording unit delivers the surveillance evidence and a mechanism unit deduces directions from the investigation program. The examination

program can enquiry the screens for clarifications and associate clarifications to regulate the source of network data features.

In addition, (Thomas, 2022) recommend double trivial original methods, SBL and SYNPM, for traceback by provided that humble and operational recording. These approaches store record information for extensive periods and respect the confidentiality of communications as well. SBL uses the SYN and FIN packets to examine simply the vital information like IP network addresses and the period of message above the classification period. The header packet information and the main four bytes of details payload of every single recorded packet examined. SYNPM permits the router to enclosure unique identifiers in the leading SYN packet each time it paths it. The identifiers are superior signatures of routers and they are attached to the packets to examine the router alongside the specific route.

Jan(2020)argued that the difficulty of identifying the real origin behind schedule of the network address interpretation access. The model presents disentangling detected packets into distinct origin and trusts on association of a quantity of discrete which permit the credentials of sources of intrusion. Author established the attribution framework on packets described as a set of packets by means of the identifying source addresses, destination addresses and sources, which are set of packets attributed to the identical client. Consignment of packets to a specific source is controlled in an enhanced approach using vitality function. The vitality function diminishes when an accurate consignment is prepared and proliferations otherwise. Vitality function can be assembled centred on credited details cookies, HTTPs referrers, IP IDs, etc.

PAS is unique essential modules in a network forensics schemes permitting examination on the internet cybercrimes. Abdullah (2022) recommended various novel approaches for

payload attribution that apply round robin fingerprinting, shingling and examining. The correctness of attribution proliferations with the span of the extract and the specificity of the probe. Abdullah (2022) expounded digital forensics as the scientific investigation of network illegal activities, illegal attempts and cyber-attacks through computer systems. It is becoming a crucial aspect to identify, preserve, examine, and analyse network intrusion evidence using proof approve and efficient techniques for eventual demonstration of evidence that help to take further actions.

2.2.7 Network Forensics Domain

The general purpose frameworks have been proposed and recommend as approaches which aim for directly supporting the inherent workflows of the network forensics domain. Most approaches operate on raw network traffic and offer search capabilities at varying granularity. Bro (2022) network security monitor offers a high-level policy for network analysis along with a rich type, event monitoring based scripting language. Bro reconstructs raw packets into transport-layer byte streams, which protocol analysers then dissect into fine-grained streams of application-specific events. User can write handlers for these events to perform arbitrary computation. Bro also ships with a library of scripts which record the protocol activity in detailed log files. In previous work, we developed the Bro cluster Matthias (2020) to scale the analysis to multi gigabytes links.

This forensic framework mode, a load-balancer dispersed the network traffic of packets over the entry and exit of network nodes devices, such that packets from the similar connection arrive at the same terminal node. Since Bro generates only network log files and does not come with a perseverance constituent, physical search rapidly runs into large scale issues. According to VAST(2021)supplements Bro by allowing a scalable output for perseverance that can natively capturing and storing specific logs files.

According to (Gregor, 2020)the timestamp network protocols records raw network traffic and builds tree indexes for a specific packet headers fields. In order to manage stream of network traffic capacities, the network security systems and tools tracks connections allowing the packets belonging to the similar flow after the connection byte stream has attain a specific threshold. The outcome of tree-based indexes avertsenough conformation of hits and evidence. For instance the querying of source and destination IP address necessitates tree index which spans both packets fields. The design does not provide the higher dimensions of scalability. Likewise according to Francesco (2021), pcapIndex indexes networks packet fields headers, but relies on bitmap indexes for enhanced composability. VAST act as superset of both time learning machine and pcapIndex providing supports of cutoff support functionality that which is vital in network forensic. VAST, provides distinctive type of input inform of identification and capturing intruded packets.

According to (Jihyung,2021)proposed FloSIS network forensic framework which delivers in-depth network traffic storage monitoring using the interface of the network edged devices at the granularity of stream network traffic instead of packets. The FloSIS framework capture and examine the flow stream of network traffic efficiently by implementing two stage indexing methodology. The first stage level uses two timestamp and four bloom filters to connect four tuple to connect the beginning and end to check if the flow of data contain the queried intruded information. The second stage level uses lookup logarithmic methodology to sort array of Meta base data flow. The framework exclusively exhibits form of network forensic architecture on network traffic similar to pcapIndex and time machine learning system.

According to (Memon,2019) proposed NetStore network forensic framework that main task implements network traffic flow archiving, stores data in flow of column format and

accelerate search of specific stored data using two forms of indexing.One form of indexing implements the functionality of temporal constraints which select suitable based column using interval tree concepts. The other form of indexing implements its function using five tuple connections inverted indexes. The main challenge of NetStore framework has been identified as slow data rate of insertion of 10K record per second and revelation of query latencies of 62M records within rate of 10 seconds. This leads to low performances for figurative interactive data query at a moderate capacity data leading to non-scalable beyond the single machine system architecture deployment.

According to (Francesco, 2020) proposed NET-FLi network forensic framework which utilizes single indexer network traffic NETFlow to accelerate searching of bitmap based on leverages. The NETFLi framework gives optimistic result based encoding scheme which based on bit victor machine learning. The framework gives slow traffic indexing based on specific machine specification and architecture.

Numerous network forensic models and frameworks have been developed to address the challenges according to Warusia (2020) and Hemdan (2021).The initiate recording of network traffic by logging information into database for inspection purposes of specific intruded attack evidence of packets. Diverse attacked features are stored which includes flow source and destination identifiers IP addresses, ports, some statistical evidence data about packets that includes packet size and packet length intervals. Altered algorithms that include network protocol analysis, apriori testing hypothesis and invulnerable that are implemented to trail network attack activities from the network logged files. The second fact relates to where the packing marking technique applied for marking the network packets among the receiving routers back to the sending routers Yonghui (2020). These heuristic methodologies and machine learning have been adapted for modelling, examining, analysing and investigating throughout network security incidents

events (Kotha, 2021)and(Qadir,2021)). These methodologies assisting in examination phase, analysis phase and investigate phase in identification and validation correct recognition between normal and attacked packets network events (Moustafa,2019).

Warusia (2020), Aladaileh (2020), Wang(2021), Yang (2020)&Soliman (2023) proposed network forensic frameworks based on packet IP traceback path.Aladaileh (2020) proposed network forensic framework that trace the attack paths of DDoS as well as IP spoofing based on traceback information which identify the main source of packet. Wang (2021) proposed network forensic framework with multiple traceback methodologies that incorporate and defend against attacks evidence especiallly fron large scale networks. The framework assist in determinition and tracking of DoS and DDoS based on IP deterministic packet marking traceback technique.

According to (Sahu, 2021) proposed multi-domain data fusion network traceback to the real sources of attack are identified by multilayer where the cloud with centralized storage and analysis systems, provide various monitoring and management services to render users more comfortable and convenient network traceback to the real sources of attack devices.

Warusia (2020) &Soliman (2023) proposed converged network forensic framework that defines evidence based on digital VoIP communication . The attacked VoIP depends on variation of normal voice packets from malicious voice packets. Soliman (2023) developed a framework for digital forensics of encrypted real-time network traffic, instant messaging, and VoIP application. Ashwini (2024) proposed collecting and analyzing network model based on network evidence and network vulnerability. The network vulnerability were reconstructed from the reasoning techniques based on the backtracing and malicious activities packets from the orginal evidence.Yusof (2021)

identify the all possible attacked paths from bayesian network graph based on analysis of discovers, visualization from network devices and computer systems.

According to (Hemdan, 2021) recommended an efficient digital forensic model for cybercrimes investigation in cloud computing probablistic forensic based graph path inference evidence. The model address both false postives evidence and inspection of evidence depeding on the computation of subsequent probabilites. According to Aladaileh (2020) proposed a detection techniques of distributed denial of service attacks on software-defined networking controller framework implemented on forensic server.Sadegh (2020) proposed scalable platform network forensic framework that enable the forensic investigation of exploited IoT devices and their generated unsolicited activities. Also proposed frameworkexamine, analysis , investigate and monitors suspicious patterns packets within scalable network events. Carvalho (2021) proposed network forensic based on intrusion detection systems with dynamic, static, execution and inspection of malicious packets. According to Wang(2021) proposed hybrid network forensic framework for detecting any form of network attacks mainly for machine to machine networks. The framework was design and developed for purposes of examining DDoS attacks within anti-honeypot distributed forensic system and architecture.

Even though the current forensic frameworks have capacity to examine, analysis and investigate security incidents to some level, the large scale of the currents distributed systems, networks and the high speed operations has leads to great challenges when extracting essential information from these architecture. The information comprises from suspicious network events, privacy issues that discloses to what levels the risks of those security incidents. Furthermore, current network forensics frameworks takes great computational system and human resources when examining, analysing and investigating

distributed networks in large scale architecture without aggregating significant network traffic streams exclusive of suspicious network security incidents.

Challenges summary associated with examination, analysis and investigation phases

Challenges association with Examination phase includes the follows:-

i. The useful network events are not identified for detecting the attacks
ii. The various protocols features being manipulated by attacks need are not listed
iii. Correlation of the attacks features with possible attacks scenarios are not performed
iv. The attack must be identified and no validation done before decision making decision to proceed with investigation analysis
v. Effective mechanism is not in place to identify attacks features form packets capture.

Challenges association with Analysis Phase includes the follows:-

i. Attack information and alerts are not taken from various security sensors as no single security tool can gives comprehensive alert information.
ii. Information are not considered from tools from comprised network for reconnaissance
iii. Data fusion of these alert and statistics are not being performed to validate attack.

Challenges association with Investigation phase includes the follows:-

i. Analysis of alerts, logs and network traffic does not lend to particular the source of attacks.
ii. Suspicious source address can be determined in the analysis phase but IP spoofing will hide the true about attacker.
iii. Traceback to the source of the attack using IP address is major challenge.

iv. The investigation does not enable the attribution of the attack to a particular host of network.

These challenges motivated the study of network forensic framework for managing network security incident especially that addresses the challenges associated with examination, analysis and investigation phases by identifying key tools and components that collect only essential information for network forensic purposes.

2.2.8 Proposed Network Framework Security Techniques

The conceptual framework addresses the challenges in examination phase was identification and correlation of network events using open source applications to open the file, read the contents of the file, encode and extract various protocol features indicating each field with a self-explanatory attribute name. Identifying important sessions of suspicious activity will reduce the data to be analysed. The correlation of events will validate the occurrence of the malicious incident and guide the decision to proceed with the investigation.

In analysis phase the no data fusion performed on the alert and attack information generated by network security sensors so that the attack evidence is more accurate to ascertain the validity of the attack occurrence. In investigation phase the existing framework lack traceback and attribution techniques to specific source of attack. Traceback and attribution techniques based on packet marking, packet logging or hybrid approaches were used to identify both internal and autonomous interface systems.

The attack attribution were done by analysing the data packets transmitted, applications being run, traffic patterns observed and protocols violated.

2.2.9 Discussion and Analysis of Current Forensic Framework Security Techniques

Table 1

Analysis of Existing Network Frameworks, Key findings and Gap/ linking to Research Study

Existing Framework Approach	Key References	Process/Activity/ Phases	Key Finding(S) And Discussion	Gaps/Links With Study
Functional Process framework for Proactive and Reactive Digital Forensics Investigation System,	Alharbi (2019)	Proactive Digital Forensics Component (Proactive Collection, Event Triggering Function, Proactive Preservation, Proactive Analysis, and Preliminary Report) and Reactive Digital Forensics Component	Hybrid (Reactive /Proactive) and the model fails to identify the implantation requirements	Lacks to identify the implantation requirements. The framework has limited. Capabilities because it does not include all the anti-forensic techniques.
Block Chain Distributed Digital Forensics Investigation Framework	Ricci (2019)	Case Leader, System/Business Owner, Legal Advisor, Security/System Architect/Auditor, Digital Forensics Specialist, Digital Forensics Investigator/System Administrator/Operator , Digital Forensics Analyst, and Legal Prosecutor	Reactive and the model is based on an internal incident response scenario rather than existing theory for a physical crime investigation	Lack examination, analysis and investigation phase making forensic framework unclear during investigation process.
Formal knowledge model for online social network forensics	Humaira (2020)	Readiness , Deployment, Traceback, Dynamite and Review	Reactive and the model is not practical	Use of terms 'multiple analysis' and 'synchronized' that are not explained and that do not have an obvious meaning in the context of analysis. Lack essential elements such as the 'chain of custody'. The model indicates a heavy focus on

				incident response thereby reducing its practicality in other areas.
Integrated digital forensic process model	Al-Dhaqm, (2020)	Preparation, Investigation, and Presentation	Reactive and overall model less generic since it is tailored for each environment	Lack examination and analysis phases respectively. No clarity for other phases
Generic Proactive IoT Cybercrime Evidence Analysis Model for Digital Forensics	Rasmi (2021)	Identification, Preservation, Collection, Examination, Analysis, Presentation and Decision.	Reactive and the model lack clarity of procedures	Examination, Analysis phases lacks chain of evidence, incomplete evidence, questionable policies, procedures and resources used gathering the evidence.
An Efficient Proactive Approach For Network Forensic	Rashmi, (2021)	Collecting Evidence, Analysis of Individual Events, Preliminary Correlation, Event Normalizing, Event De-confliction , Second Level Correlation, Timeline Analysis, Chain of Evidence Construction, and Corroboration	Reactive and not suitable for simple digital forensic investigations as well as no definition of complex investigations provided	No examination, analysis and investigation lack key components to link with proactive and reactive components. Not practical and too complex to implement
Architecture of Digital Twin for Network Forensic Analysis Using Nmap and Wireshark	Kumar (2021)	Pre-Incident Preparation, Detection of Incidents, Initial Response, Formulate Response Strategy, Investigate the Incident, Reporting, and Resolution	Reactive and No clarity of phases which lead to complexity to implement	Lack investigation and analysis phase components and tools which take the framework incomplete for forensic investigation. Too complex to implement
New Network Forensic Investigation	Rachana (2021)	Awareness, Authorization, Planning, Notification,	Reactive and the models is too narrow	Lack analysis and investigation phases.

Process Model		Search and Identify Evidence, Collection, Transport, Storage, Examination, Hypotheses, Presentation, Proof/Defence, and Dissemination		No clear components and tools for analysis and investigation phases
Artificial Intelligence (AI) Model Forensics.	Tiffanie (2021)	Collection, Examination, Analysis, and Reporting	Reactive, model's focus on incident response and lack of a separate planning stage prior to the physical collection of data which is a serious shortcoming	The framework is brief since it lack other phases which are essential for examination, analysis and investigation phases.
Data-Driven Decision Support for Optimizing Cyber Forensic	Antonia (2021)	Initialization, Evidence Collection, Evidence Examination and Analysis, Presentation and Case Termination	Reactive and the models lack flexibility	No preparation and preservation phases.
Bringing Forensic Readiness to Modern Computer Firmware framework	Tobias (2021)	Pre-Analysis (Detection Of Incidents, Initial Response, and Formulation Of Response Strategy), Analysis (Live Response, Forensic Duplication, Data Recovery, Harvesting, Reduction and Organisation), and Post-Analysis (Report and Resolution)	Hybrid (Reactive /Proactive) And the model misses details such as the Pre-incident Preparation requirements from the Common Process Model	Lack examination and investigation phases make forensic framework incomplete.
Digital Forensic Frameworks BasProcesses Models,	Talib (2021)	Investigation Preparation, Classifying Cyber Crime and Deciding Investigation Priority, Investigating Damaged	Reactive and the model is too complex to implement and deals with priority crimes	Framework is based on an internal incident response scenario rather than existing theory for a physical crime

		(Victim) Digital Crime Scene, Criminal Profiling Consultant and Analysis, Tracking Suspects, Investigating Injurer Digital Crime Scene, Summoning Suspect, Additional Investigation, Writing Criminal Profiling, and Writing Report	which is narrow based	investigation. Information supplied it is also not clear that the framework is practical and follows the steps of an actual investigation as the worked example is too theoretical to confirm practical applicability
Validating Mobile Forensic Metamodel Using Tracing Method	Abdulalem (2021)	Preparation, Collection and Preservation, Examination and Analysis, Presentation and Reporting, and Disseminating the Case	Reactive and the model is are too complex	Each phase works in real time (thus, the phases require the same amount of time.
Roadmap of Digital Forensics Investigation Process framework	Anita (2021)	Preparation, Detection, Incident Response, Collection, Preservation, Examination, Analysis, Investigation, and Presentation	Hybrid (Reactive /proactive) and the model is generic	Each phase works in real time (thus, the phases require the same amount of time and processing cost to accomplish their processes. No link with proactive/ reactive components and tools between examination, analysis and investigation phases
Network Forensics Investigation: Behaviour Analysis of Distinct Operating Systems to Detect and Identify the Host in IPV6 Network	Abdullah (2021)	Generic Proactive (classification of evidences, based on its significance and relation to past crimes, severity of the evidence and probability occurrence of a cybercrime)	Generic Proactive model is difficult to automate is there no clear link in all components and no clarity mapping processes	Implementation and automation of the framework is more difficult to create automated tools since there no clear link in all components. No clear mapping of processes for examination, analysis and investigation phases since no clear

				components, tools and traceback techniques
Digital Forensics Investigation Procedures of Smart Grid Environment	Abdullah, (2022)	Readiness (Operations, and Infrastructure), Deployment (Detection and Notification , and Confirmation and Authorization), Physical Crime Scene Investigation (Preservation, Survey, Documentation, Search and Collection, Reconstruction, and Presentation), Digital Crime Scene Investigation (Preservation, Survey, Documentation, Search and Collection, Reconstruction, and Presentation) and Review	Reactive and the model is not practical to implement	The examination and Investigation phases are too general to be applied in practice There does not seem to be a way of testing the framework. As the model is developed to increase its detail it becomes more complex and more cumbersome to use. No examination, analysis and investigation phases mapping proactive and reactive components.
Comprehensive Framework	Samuel (2022	Planning, Triage, Usage/User Profiles, Chronology/Timeline, Internet Activity, and Case Specific Evidence	Reactive and the models are too broad	Lack examination, analysis and investigation phases. Too general for forensic investigation
Hierarchical Framework for Digital Investigations	Sunde (2022)	Preparation, Incident Response, Data Collection, Data Analysis, Presentation, and Incident Closure	Reactive and lack completeness	Lack examination and investigation phase. Lack completeness and phases clarity
Systematic Model for Investigation	Thomas (2023)	Based on infiltration detection systems, traceback, distribution models, and attack maps	Reactive model .Integrates passive methods that do not transform packets and collection interpretations	Examination and Analysis phases lack collection interpretations for future examination.
Artificial intelligence-based proactive	Abirami (2023)	Uses machine learning based classification algorithms to forecast	Combines a reactive and proactive	Encryption method that encrypts the proactive module's

network forensic framework		the attack	framework	report before decrypting it in the reactive module. Complexity procedures and algorithms
A framework for digital forensics of encrypted real-time network traffic, instant messaging, and VoIP application.	Soliman (2023)	encrypted real-time network traffic, instant messaging and VoIP application	real-time model depends on variation of normal voice packets from malicious voice packets	Complex analysis processes and tools aim to collect VoIP application resulting to cumbersome system.
Collecting and analyzing model network-based on evidence	Ashwini (2024)	Collection and analyze phases based on network vulnerability	Proactive model. Reasoning techniques based on the backtracing and malicious activities packets	Reasoning techniques relies on network vulnerability
A Novel Network Forensic Framework for Advanced Persistent Threat Attack Attribution Through Deep Learning	Mei (2024)	Collection, examination, analysis and investigation phases depend on deep learning	Reactive and proactive model. Attack attribution through deep learning. Deep learning attribution trace hashed marked comprising of unique list	Complexity algorithms to implement deep learning to list unique hashed mark. Requires large database storage facilities

Based on the study from literature the following research gaps were identified that links research study;

i. There is general lack of cohesiveness in examination, analysis and investigation phases in existing frameworks in the face of the heterogeneous nature and the growing volumes of digital evidence.

ii. The results from examination, analysis and investigation phases of one or more framework and tools do not yet lend to integrated analysis requiring significant manual effort to establish corroboration. It requires the identification of associations in digital evidence and grouping the associated elements in a manner that is forensically productive.

iii. The heterogeneity of attacked evidence has significantly challenged the ability to generate unified timelines across multiple sources, often leading to ambiguous or inconsistent timelines.

2.3 Conceptual Framework for Managing Forensic Network Security Incidents

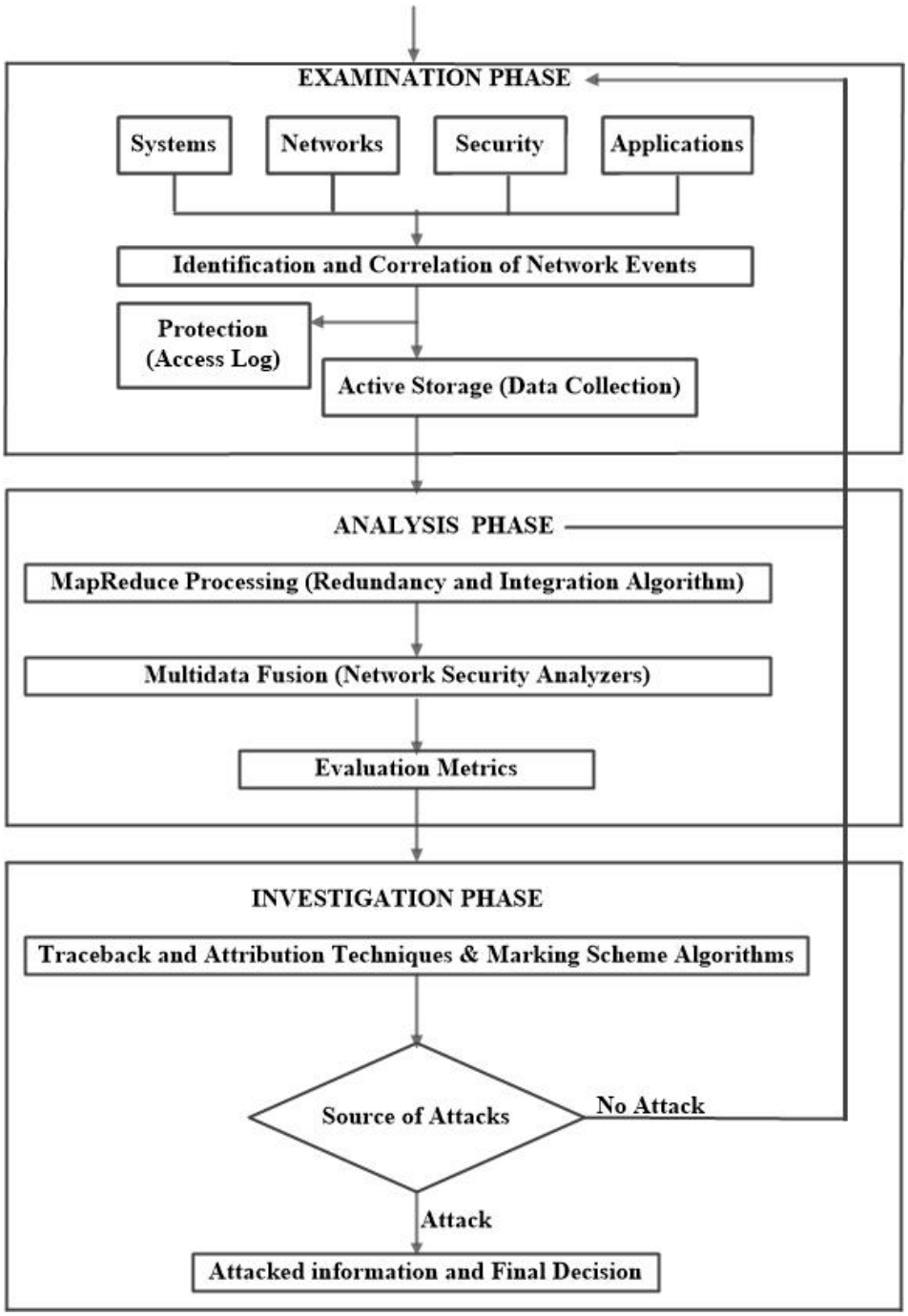
Conceptual framework for managing forensic network security incidents addresses the shortcomings and research gaps identified in examination, analysis and investigation phases associated with different phases in other frameworks. The framework addresses research objectives associated with examination, analysis and investigation phases. The main challenges of examination phase in other framework are identification and correlation of network events with attack features. The main challenge in analysis phase is data fusion. The main challenge in investigation phase is source traceback and data attack attribution as shown in the figure 7 below.

To handle and manage fake evidence identification examination phase implement correlation technique which checks evidence to ascertain the fake evidence. In the analysis phase redundancy algorithm check for fake evidence and also the evidence is subjected to multi data fusion sensors to check for fake evidence In the investigation phase the fake identification is identified through packet marking scheme algorithm in both traceback and attribution techniques as indicate conceptual framework.

2.3.1 Conceptual Framework for Managing Forensic Network Security Incidents

Figure 7

Conceptual Framework for Managing Forensic Network Security Incidents



The Figure 7 and figure 8 illustrates the conceptual frameworks and flow diagrams respectively for managing network security incidents. The examination phase framework implements identification and correlation of network events captures network packets

from suspicious attacked network hosts and transfer the attached packets to analysis forensic active storage (database server) and preservation access log for future references. All networks protocols attributes are captured and correlated, then each specific protocols attributes are transfer to unique created database table. Each packet is recorded with unique time stamp associated with frame number generated automatically.

The analysis phase implements multi-sensor data fusion on forensic evidence implements integration of identification and correlation of data set captured from examination phase alerts attack information. Packets captured are integrated from multiple sources across the network and redundant packets are dropped using proposed algorithm. The analysis framework is built by aggregating the strengths of open source tools in accomplishing the task of collection and analysis. The network security forensic sensors implement the functions of complementing and contradicting evidence, which may arise due with weakness of other tools. Security tools with similarity build the redundancy, reliability and diversity among the tools to ensure versatility of attack information.

Data fusion is performed on the alert and attack information generated by these sensors so that the decision is more accurate. These network sensors analyses the packet attacks in order to increase the confidence level of evidence. The data integration and multi data fusion model validates sample data set with attack packets generated. The accuracy of these tools is validated using evaluation matrix criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance. The proposed scheme for analysing and tracing the attacked vectors of evidence for fusion to detect and validated the crucial decision to proceed with investigation phase.

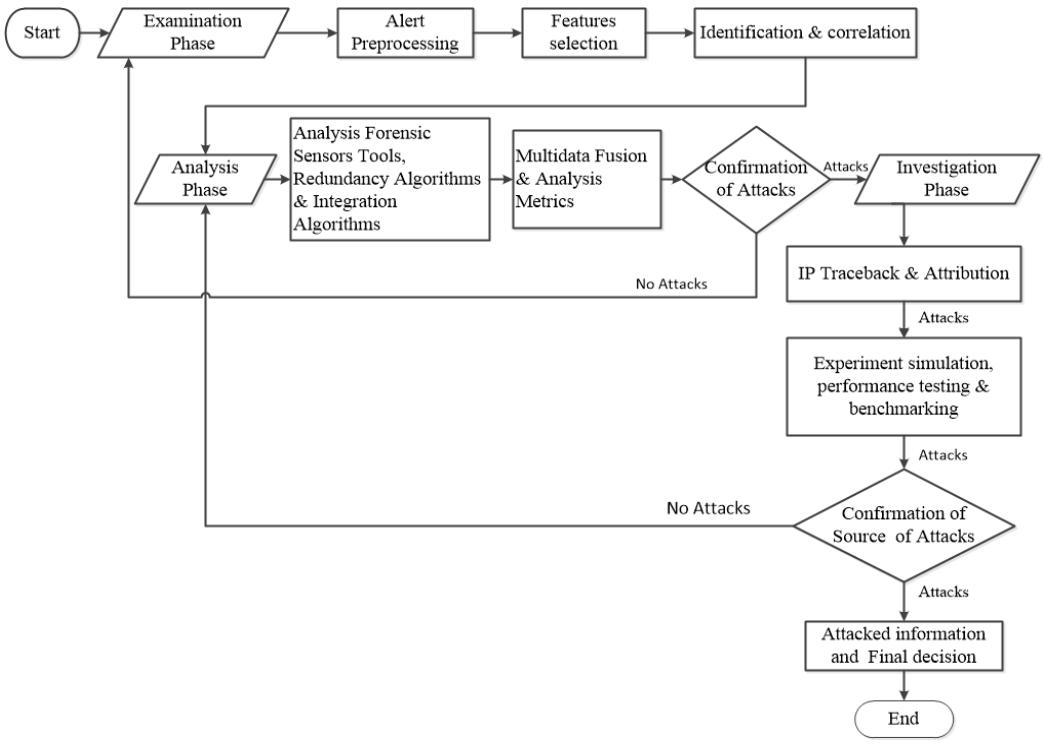
The investigation phase implements the role of source traceback and attribution using AS based Deterministic Packet Marking (ASDPM) and Deterministic Router Interface

Marking (DRIM) approaches. The ASDPM technique uses a two level traceback mechanism. The first level mechanism marks every individual packet deterministically by hashing first internal IP address of the router within the AS. The DRIM technique implements marking and hashing deterministically every packet that passes the first inbound edge router IP address as well as the number of the interface that the packet uses to connect the same router. The markings is done by the inbound router using define marking algorithm which manage to traceback and attribution of source of attack from particular system. This conceptual framework for managing forensic network security incidents improves broad forensic perspectives, standards and admissible forensic scientific evidences.

2.3.2 Conceptual Framework Flow Diagram for Managing Forensic Network Security Incidents

Figure 8

Conceptual Framework Flow Diagram for Managing Forensic Network Security Incidents



CHAPTER THREE

RESEARCH METHODOLOGY

3.1 Introduction

This chapter presents the methodology on designing and evaluation of the architecture of the proposed network forensic framework for managing network security incidents. It starts with the addressed challenges of examination, analysis and investigation phase with applied solution concepts. Based on this, the research design and research framework is provided. The rest of the chapter provides comprehensive description on how to achieve the specific objectives, performance measurements, the dataset used in this research as well as forensic technologies used for simulation experimentation and methodological assumptions that are adopted in the development of forensic framework.

3.2 Investigative Research Design

The overall approach for research study is interrogative literature review, quantitative approach and evaluation based on Design Science Research Process (DSRP) (Marlien, 2020). DSRP approach is established on a ‘problem centred’. Problem centred implies DSRP methodology approach enable in achieving the research objectives. It incorporates principles, practices and procedures required to carry out research and attain the four research objectives, provides interrogative and consistent literature review that enable to identify existing network challenges and gaps linking the research study. It also provided normal process framework development and provides critical results analysis framework for presenting and evaluation design science research in network forensic in achieving desired research objective and results. This research addresses the issues of improving the quality of network forensic security incidents that are identified and captured by multiple security sensors and recognizing the attack strategy from the network security sensors alerts information. It is examined, analysis and investigated through a series of

evaluation performance of experiments and testing to achieve the goal of objectives of the research. This approach preferred as the main method due to certain characteristics, such as identification and correlation, dataset evaluations and the usability of the results. The method evaluation of framework performance depends on periodical experiments and testing as illustrated in figure 9 that map challenges of examination, analysis, investigation phases and solutions. The solutions for each challenges is designed on examination, analysis and investigation phases respectively based on the following three main security techniques concepts as follows:-

i. Identification and Correlation: The research study addresses the main objective of developing a network forensic framework for managing network security incident. Saeed (2023) recommended the need to implement different methodologies and frameworks related to digital forensics investigation and incident response and elaborated the impact of forgery and tampering in the evidence chain-of-custody. The network traffic events comprises of header information of each packet in the capture file are read and various fields that are manipulated for attacks are examined leading to specific identification and correlation of attacked information relied from network security incidents pre-processing information. These information present more details based on representation attributes and attacked features, scaling all attributes into selected range identification of optimum features. The solution is to standardize and improving the quality of network forensic information into a unified form and normalize it into an acceptable input before presenting to subsequent examination phase for further analysis.

ii. Multi data fusion and Integration algorithm: The attacks identified in the examination phase analysed by performing data fusion of information from multiple security sensors. Security sensors with complementary and contradictory functions are used. Security tools with similarity build the redundancy and reliability of attack information. Open source tools, commonly available on the network to gather attack information and generate alerts, attack statistics and information based on specific attacked information evidence. They are designed for active and post-mortem investigation of packet captures. Full packet data is captured by this tools, stored in a host and analysed off line at a later time. They also collect statistical information based on some criteria within the network traffic as it passes through the network. The network equipment collects this data and sends it to a flow collector which stores and analyses the data. These tools are able to trace *p*acket based systems involving full packet captures at various points in the network. The packets are collected and stored for deep packet inspection by forensic investigators. The information produced by these tools in one stage are characterized and transformed for use by other tools in the succeeding stages. The data sets are partitioned, system are trained and then tested and integrated so that the investigator can have an edge over the attacker. These tools identify and tracks files across local area networks and across the World Wide Web, for the purpose of gathering intelligence and forensic evidence and standard means of extending the functionality of other tool architecture. Time consuming and error prone processes are identified and automated. These tools are used to converted attack packets back into the same format harnessing their strengths which enable analysis. Diversity among the tools ensures versatility description of network forensic analysis tools and description of Network

Security and Monitoring (NSM) tools as illustrated in appendix I. The main role of examination phase implement the diversity of network sensors techniques enhanced analysis of feature selection method where network traffic has many features to measure. The problem is that with the huge amount of network traffic we can measure many irrelevant features. These irrelevant features usually affect the performance of detection rate and consume the network security techniques resources, clustering and classification of network security incidents and data integration. These steps leads grouping of selected attacks to differentiate false positives, redundancies, true alerts and reducing the size of information to manageable level. The is achieved through multi-data fusion through implementation of combination of security sensors which enhance attacks evidence accuracy and apply proposed integration algorithm to reduce the size of identified attacks to manageable level for better unitization of network security resources.

iii. IP traceback and attribution Techniques: Source traceback and attribution methodology for reliable determination of the origin of a packet on the network techniques based on packet marking, packet logging or hybrid approaches. The traceback attack attribution is done by analysing the data packets transmitted, applications being run, traffic patterns observed and protocols violated using Information theory metrics. Deterministic packet marking is more suitable for network forensics as many attack packet streams may consist of only few packets and investigation needs to be performed using the limited evidence. For well-known attacks evidences, the causes usually given in the traceback back and attribution signature files or/and locally or remote labelled by the AS based on their professional technical knowledge and experiences. But for attack security

incidents alerts, the traceback and attribution techniques approach must have capability to determine the path and source of attacks. Traceback and attribution techniques is based suitable in determination in discover of source of attack stages membership can be performed by deterministic packet marking techniques implemented within forensic framework investigation phase.

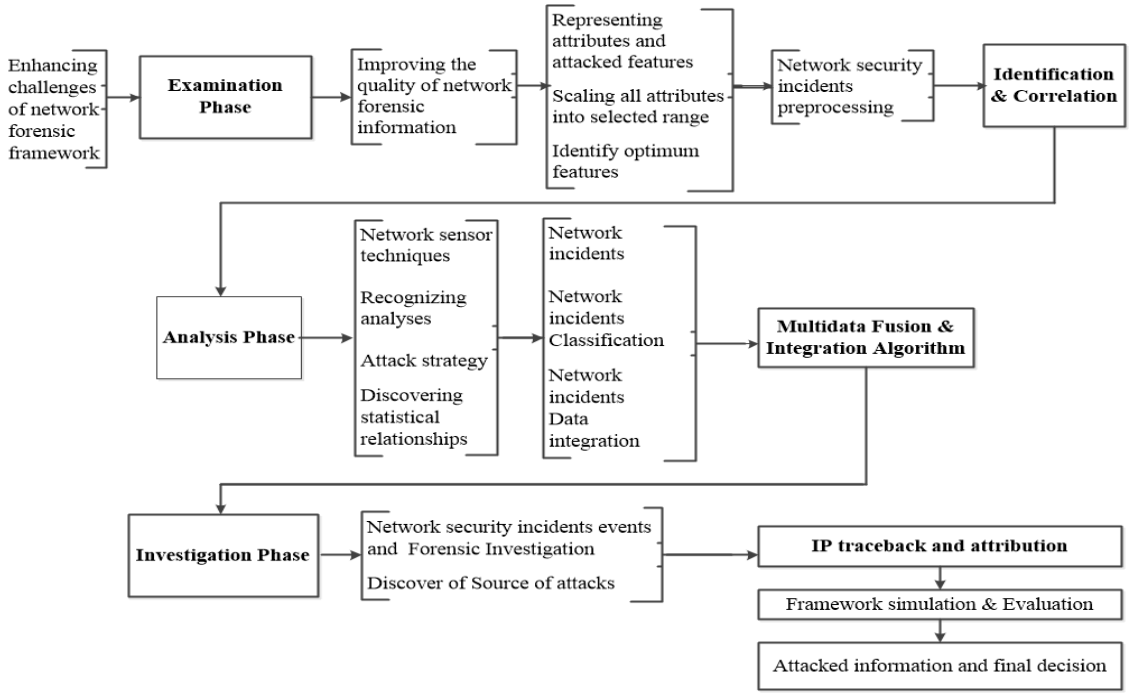
iv. Performance evaluation metric: The network security incidents to be identified, correlated, analysed, traceback and attributed to specific source of attack not only effectively, but also completely and optimally forensic investigation. Therefore, the correlation performance needs to be improved. To offer optimality, identification, correlation, analyses, traceback and attribution among selected attacked evidence features and attributes has to be considered as well because some selected features and attributes may have dependency and redundancy associations within each other through performance and evaluation metric. For instance, a specific targeted protocol port number makes certain host IP address to be spoofed or becomes a source in an attack graph path. Such case shows that the protocol port number features and attribute is dependent to the IP address traceback and attribution. Thus, the protocol features and attributes dependency strength has to be measured.

The Figure 9 shown below illustrates mapping of challenges of examination, analysis, investigation phases and security techniques solutions for each specific phase.

3.3 Flow Diagram Mapping Challenges of Examination, Analysis, Investigation Phases and Solutions

Figure 9

Flow Diagram Mapping Challenges of Examination, Analysis, Investigation Phases and Solutions



The mapping on network forensic framework challenges in examination, analysis and investigation phases with the respected solution concepts used as guidance during the development of the network forensic framework for managing security incidents discussed in the next section.

3.2.1 Investigative Research Framework

The research framework for developing proposed network forensic framework for managing network security incidents is illustrated in figure 10 shown below. The examination phase input from system, networks, security system and application captures data network traffic for that has been identified and correlated by specific protocol through selection of attacked attributes and key features selection. The data traffic examined are preserved through Map Reduce processing algorithm access through

active storage collection systems. The analysis phase input consists of network sensor techniques, recognition analysers and attack strategy identifiers. The outputs incidents are clustered, classified and integrated through multidata fusion networks sensors, data redundancy and integration algorithms. The investigation phase input consists of forensic investigation deterministic packet marking techniques that manage to discover source of attacks through traceback and attribution.

The output of the proposed examination phase framework manages the attributes and attacked features, scaling all attributes into selected range, identify optimum features and incidents pre-processing that are examined resulting to identification and correlation of attacked information. The output of the proposed analysis phase framework manages data integration through the proposed integration algorithm and multidata fusion to ascertain the attack information accuracy. The output of the proposed investigation phase framework manages IP traceback and attribution in identification of source of attack to particular system. The proposed network forensic framework for managing network security incidents consists of three main tasks:

i. The purpose of examination phase to examine data captured and identify by multiple network protocols. In order to perform and support automated identification and correlation on the captured attacked information. In this research attributes and selection of attacked information are normalization and merging actions are included.. The groupings of attacked information are based on the similarities of several attributes and features values using selection algorithm. Series of attack steps are revealed by identifying the amount of clusters produced. Common attack steps can be recognized by looking at the large clusters. A collective for feature evaluation, features selection algorithms

were invoked to select the set of relevant features and data integration algorithm were invoked for data size reduction.

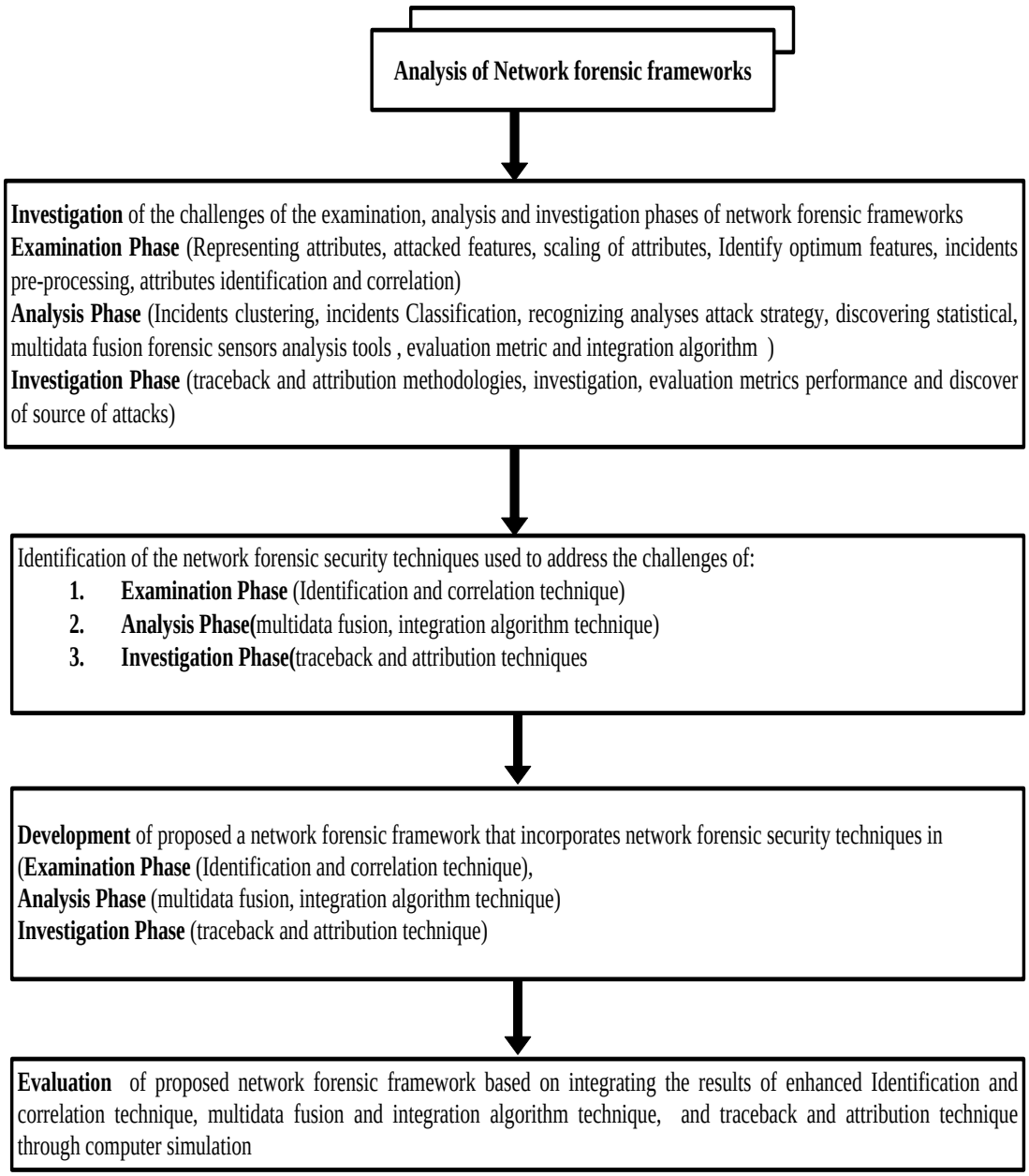
ii. The purpose of analysis phase to analysis examined attacked information using diversity network security sensors to ascertain the attacked information accuracy .It can produce good performances due to the elimination of irrelevant and less important features are removed. Post clustering deals with discarding redundant alerts and false positives for improving the alerts quality. It is based multidata fusion, verification and Alert Filtration to remove the identified redundant with high detection accuracy and reduce false positive of attacked information based on data evaluation metric. The aim analysis phase to produce better classification accuracy and to offer more complete correlation compared to existing works.

iii. The investigation phase framework manages IP traceback and attribution techniques using proposed deterministic router interface marking for internal inspection and autonomous system based deterministic packet marking for external inspection. This is conducted sequentially in order to recognize source of attack and attack strategy and improve the investigation forensic performance in determination of admissible forensic evidence.

3.4 Investigative Research Framework Flow Diagram

Figure 10

Investigative Research Framework



The steps used in investigation of the challenges of the examination, analysis and investigation phases of network forensic frameworks are discussed in the subsequent section below.

3.4.1 Investigation of Examination Phase

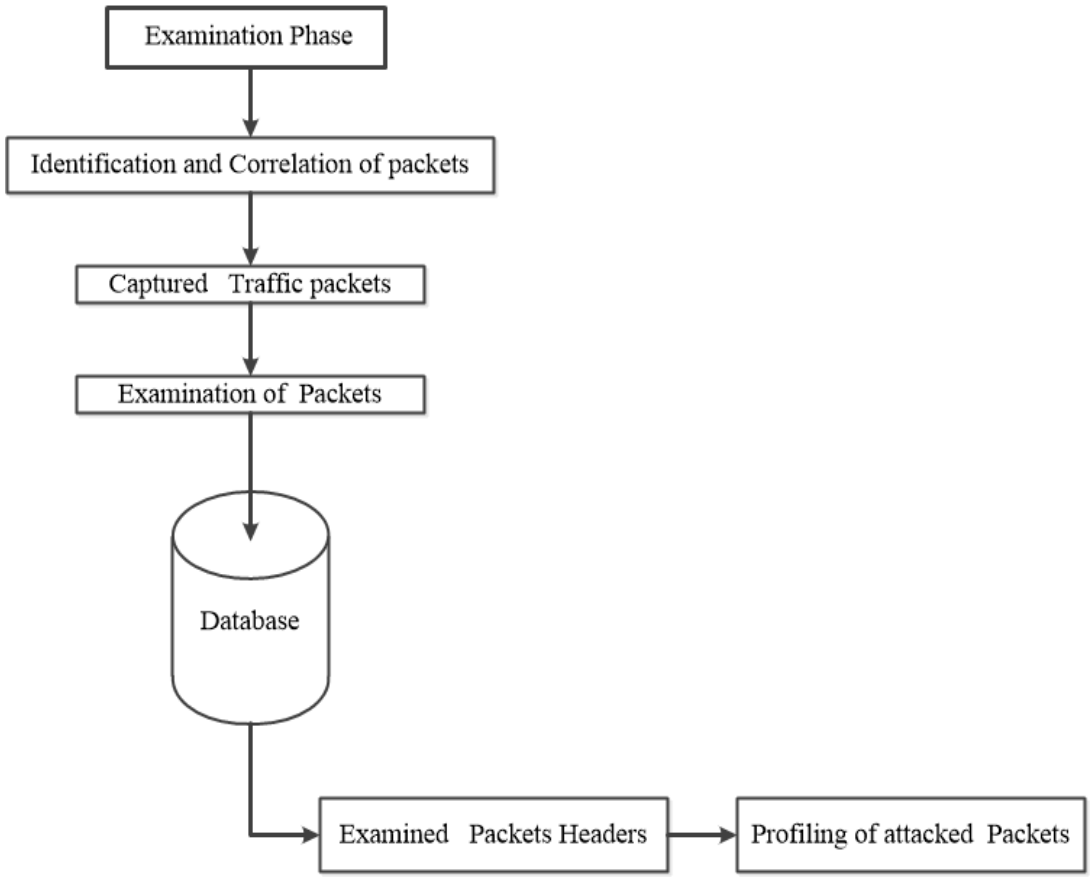
To make efficient use of the available network traffic dataset for examination phase the data pre-processing is required to provide solutions to clean the data to remove duplicate information, redundant and noise information then deal with any incomplete or missing data an efficient algorithm based on examination identification and correlation techniques. The different features are representing attributes, attacked features, scaling of attributes, identify optimum features, incidents pre-processing. This is what is done to features like protocol type, source IP address, destination IP address, protocol version, service, and offset flags and so on. One of those is security incidents attack and the other is usual perfect connection. For example, an intrusive connection is a hacking form of attack that uses a fake source IP address to send a flooding quantity of (ping) ICMP echo traffic to several clients.

The destination clients respond to the echo requests with an echo replies and flood the target client whose IP address is faked by the intruder. The observation were some protocol features and attributes (e.g., src bytes, dest bytes and count) in the attacked record indicate that the features and attributes values of the smurf attack are unrelated from those of the usual connection. For example, the source byte features and attributes shows that the intrusion connection spoofing the IP of some host sends out considerably more data than a usual connection. An identification and correlation technique was figured out from profile to designate the features and attributes of usual or unusual connections and use the profile to detect intrusions following the flow chart of examination phase as shown figure 11 below.

3.5 The Flowchart of Examination Phase

Figure 11

The Flowchart of Examination Phase



3.5.1 Investigation of Analysis Phase

Many of the existing open source network security sensors tools are install and implemented for monitoring and detecting security incidents for specific tasks in network forensics. Nevertheless, proprietary network sensors tools use the strength and functionality that are explicitly constructed and implemented to captured only specific type of network traffic with different level of evidence. The most used open source network sensors are discussed in subsequent session alongside alerts strength and indicators. Various security sensory techniques are run on this pcap file and information fusion to ascertain the validity of the attack occurrence. The suspicious attack information are used to collect suspicious packet records in the integrated pcap file.

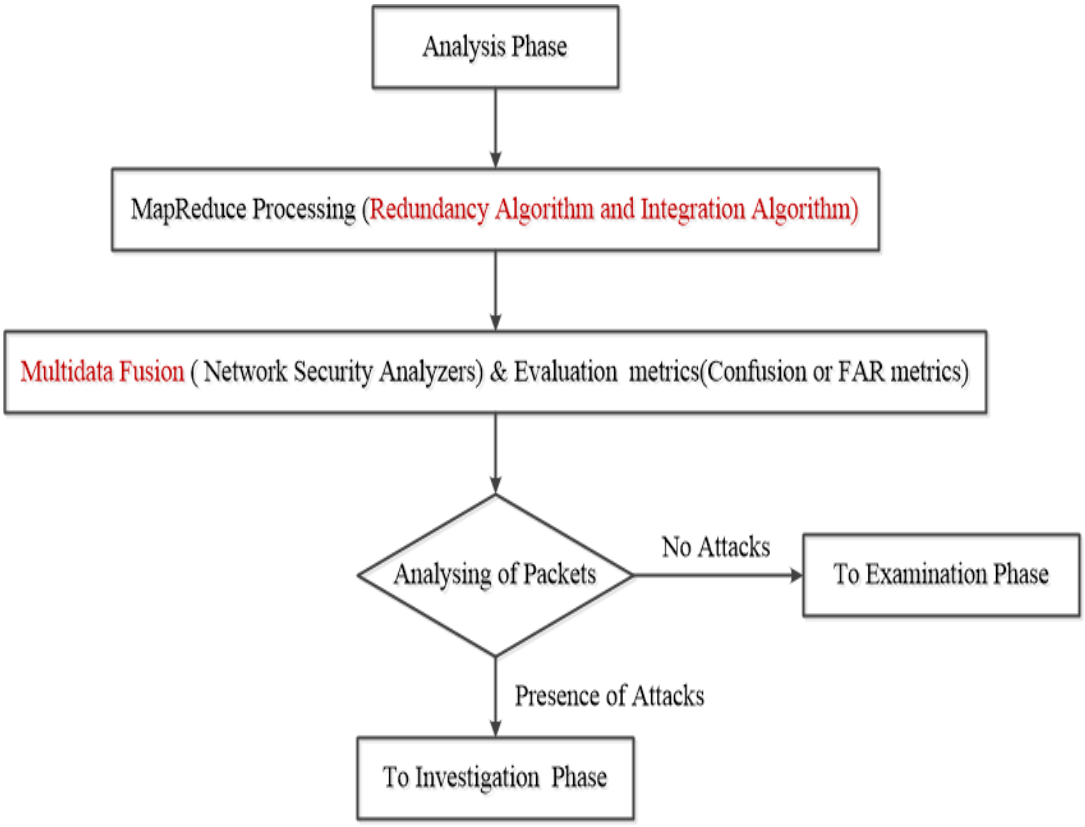
3.5.2 Feature Selection Method

Data mining on huge amounts of data is time-consuming operation, making such analysis impractical or infeasible. Data reduction technique have been used to analyse reduced representation of the dataset without compromising the integrity of the original data and yet producing the quality knowledge.

3.6 Analysis Phase Framework Architecture for Multi Data Fusion

Figure 12

Analysis Phase Framework Architecture for Multi Data Fusion



The analysis framework is built by aggregating the strengths of open source tools in accomplishing the task of collection and analysis. Network security sensors with self-contradictory and corresponding utilities are used. Security tools with similarity build the redundancy and reliability of attack information. Diversity among the tools will ensure versatility. Data fusion is performed on the alert and attack information generated by

these sensors so that the decision is more accurate. In case suspicious packets are detected an NIDS givens notifications inform of alerts. Packet capture and analysis tools or sniffers identify sessions or connections with anomalies in network traffic. Traffic statistics were read from packet captures or from Netflow records taken from the network connection through a monitored device. The following network sensors tools were used in analysis phase framework architecture for multi data as shown in Figure 12.

Fusion Snort: It is NIDS open source software with ability of examining, detecting, analysing and logging network packet traffic during transit with suspicious features and contrary to define rule set configured by the user. Snort has also ability of decoding, printing logs, alerting and capturing full packets headers evidence messages by default. One of the features of snort is fast alert mode, which has ability to write, read and analysis packet file format detailing the alert massage, timestamp, source IP, destination IP and port numbers.

Wireshark: Wireshark is open source network software with ability of capturing and analysing real time network traffic during transmission. It has a module used to output packet information longest side protocol evidence. The sensors ability is to captures the packet where the forensic investigator can be a position to read, import, and export and saved the contents. It can also filter, search contents based on specified criteria and create numerous statistics specified by the investigator. Wireshark can be integrated with other network sensors in decoding protocols contents in order to dissector other high-level protocols as well.

Tcpstat is open source software able to monitor and report by reading certain network statistics interfaces. It has ability to read previous tcpdump stored file as well as calculate the packet traffic such as bandwidth, speed, packet average size, load of particular interface, standard deviation of packet size etc. It calculates statistics like bandwidth,

number of packets, packets per second, average packet size, standard deviation of packet size, load of particular interface and ability to manage various transiting packets per second.

Bro is an open source UNIX based NIDS software capable of detecting and monitoring network traffic that has been manipulated and attacked by intruders based on contents and characteristic nature. It collects, filters, and analyses traffic that passes through a specific network location. Bro occur with fixed procedure code calculated for detection of most common intrusion of Internet application. These policies incorporate a signature matching facility that looks for specific traffic content. It can also analyse network protocols, connections, transactions, data amounts and many other network characteristics.

Basic Security Analysis Engine (BASE): It is originates from ACID code and has ability of offering front-end web interface for analysing and querying incoming alerts from other network security sensors such as snort. It has high sensing ability to detect attacks that snort cannot detect in case of intrusion. It is used to supplement other security sensors in a network forensic investigation through web interface module. It allows security investigator flexibility to make decisions based on what and how much each user can access information through authentication mode. We convert the contents of the pcap file into a database and reconvert the attack packet records in the database to a new pcap file using the Net::Pcap module of the Perl language. The file contents of lib-pcap captures packet information containing the protocol types used such as TCP, IP, UDP, ICMP and Ethernet alongside encapsulation details. These protocol features are extracted recursively from Lib-pcap file and inserted in to database table.

Another lib extension file is Net: Pcap that can used to encode and extract protocol features. The main protocol attributes captured by Net. Cap file extension and extract

include protocol ver, tos, hlen, id, offset, ttl, cksum, src_ip, dest_ip, tos, proto_type and options. These attributes are encapsulated, copied and stored into specific table in forensic database. These packets attribute creating header evidence of network protocols intruded by an intruder in compromising the user network end systems. A new packet capture file is created from the attack packets which is minimum in size and with maximum possible information as evidence. The pcap file with only attack packets is very much reduced in size when compared to the integrated file.

In investigation phase, we selected the best traceback and attribution approaches based on deterministic relevant attacked dataset features. The features selected in the previous analysis and examination phases were ranked based on their relevance value to each attack class. The two approach selected was based on information theory metric deterministic packet marking as listed in appendix III figure 1.1.One approach selected was based on ASDPM traceback and attribute records of internal router within the source of AS determines relevant attacked packets features more closer to the attacker. The other selected approach was based on DRIM that record information from the source traceback and attribution as it move one step by identifying the interface on which the packet reaches the first AS router and determines the source of the attacker much closer as shown appendix III figure 1.2. The proposed algorithm used in two approaches is shown in Figure 37 and 38 respectively.

3.6.1 Investigation of the Network Forensic Security Techniques

As network security incidents increases in posing security threats in most organisation and businesses in drastic and advance ways, depending on appropriate measures used in network forensic framework security techniques implemented by organizations for identifying, examining and investigating network security incidents activities. The current network forensic frameworks security techniques requires a massive amount of

information during examination, analysing and investigating the attacked information which overwhelmed network forensic security specialists as they require high levels of human involvement during the identification of relevant information of admissible attacked evidence . Thus, identification of the network forensic security techniques **is** essential to disclose their identification and correlation by grouping alerts with common attributes. Identification of forensic multidata fusion sensors and integration algorithm techniques that can be used during analysis phase of attacked information. Identification of forensic traceback and attribution technique that guide to trace the attacker more closely to source of attack.

The main goal in this phase is to improve the network forensic framework for security incidents based on essential based identification of examination, analysis and investigation phase's techniques to improve the quality of attacked information evidence by grouping evidence based on common attributes based on identification of best network forensic techniques. Our focus is to improve the quality of network forensic framework based on attacked evidence as much as possible, but not to substitute them. Therefore, network forensic framework for managing security incidents is proposed to improve forensic examination, analysis and investigation phases to enhance quality of attacked information and discover the procedures attackers used in launching network security incidents.

The main goal of the examination phase framework was identification and correlation of network events for capturing network packets from suspicious attacked network hosts and transfers the attached packets to analysis forensic database server. The details evidence captured from libcap.cap libraries contains packet header, captured number of packets, and the length of captured packet. The main libcap.cap library file used to extract packet header attributes snaplen details is perl language module Net::Pcap, which

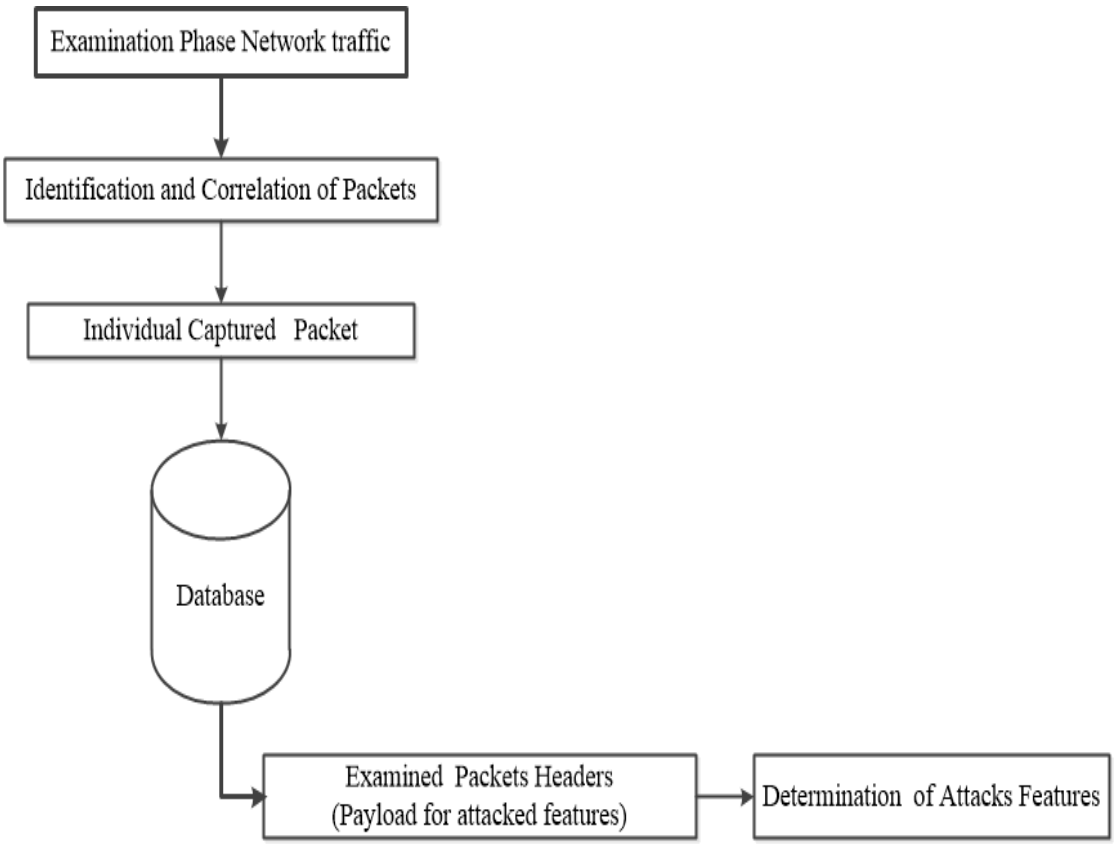
enables to capture and correlated all packet length size stored. The details information contained in the libcap.cap file for numerous packet headers such as TCP, IP, UDP and ICMP protocols were extracted from the file and stored in the forensic network database Net. Pcap server respectively. The libcap.cap enables encoding of captured packets and extracts numerous network protocols attributes that specify each packet field name. The packets encoded and extract corresponding to ver, TTL, flags, ID, offset bit, cksum, proto, srcip, destip, and options attributes.

The specific associated with TCP protocol packet header attributes captured, and correlated by Net. Pcap file includes src_port, dest_port, seq_num, , ack_num, flags, hlen, flags reserved, urg, winsize and options. The specific associated with protocol packet header attached attributes captured and correlated by Net.Pcap file includes src _mac, dets_mac, ID, tos, flags, srcip, destip, TTL and options. For tracking the non-stationary properties of flow identifiers, we apply the ‘count’ functions to determine all possible combinations of these flows as shown in Figure 13.

3.6.2 The Flow Diagram for Examination Phase Framework for the Identification and Correlation of Network Events

Figure 13

The Flow Diagram for Examination Phase Framework for the Identification and Correlation of Network Events



Steps:

Step (1) Read the packet captured file as inputs. The individual packet captured file have been processed by specific forensic network protocols Method are read from the database as inputs examination phase.

Step (2) Save the examined packets and experimental results. The examined and experimental results are recorded and saved in the database. It includes the details on all of the identified attacked packets and steps as well as the statistical analysis.

Step (3) The ‘count’ functions was applied to determine all possible combinations of these flows, as follows to examine headers (Payload for attacks features) as follows

- *Select COUNT(*) as flows, srcip, dstip from network_data group by srcip, dstip;*
- *Select COUNT(*) as flows, srcip, srcport from network_data group by srcip, srcport;*
- *Select COUNT(*) as flows, dstip, dsport from network_data group by dstip, dsport, srcport;*

Step (4) Save the examination and experimental phase results. The examination and experimental phase results are recorded and saved in the database. It includes the details on all of the identified and correlates of network events examined in network and transport layer as well as in the application layer of TCP/IP model attack packets and steps used to launch network security incidents.

The main objective of integrating all files captured from numerous clients into one single file so that entire attacked information are available at one place as shown in figure 14. It is also easy to analyse a single packet capture file against a series of security tools. The integration resulted into data reduction, as some of the packets collected by the multiple hosts will be similar. There will be broadcast and multicast packets, which toggle by all hosts. The major issue to integrate the files will be to handle the timestamps of redundant packets with same information. The other issue is to identify which files to be integrated directly and which files need a redundancy check before integration. This decision depends on the location of the compromised hosts from which the files are collected,

whether the system is within a particular subnet based on developed integration algorithm.

The packet captures (.pcap files) are collected from compromised systems, which are identified in a network. These files are integrated by converting them to a database, identifying unique packets and recreating a single file. This is achieved by developing an algorithm that identify files for integration and another algorithm that handle integration after checking redundancy.

The steps include.

i. The algorithm decides which files are to be integrated directly and which files are to be checked for redundancy issues before integration.
ii. Identified related packets field of all transmitted network traffic examined after being integrated from the timestamp selected within the dataset range.
iii. The algorithm enables and identifies the specific dataset similarity by selecting the packet header and payload information.
iv. Algorithm implements the rules for forming groups of hosts from which the files must be integrated, only after removing redundant packets.
v. Packet logs existing in a particular group are only integrated after removing the redundancy.

Many of the existing open source network security tools are used for specific tasks in network forensics. We used the open source network security and monitoring tools as shown flowchart in figure 14 to read the packet capture file post attack, and give various alerts and indicators. The attacked information generated by ‘m’ number of security tools on ‘n’ number of packet captures are same as the alerts generated by ‘m’ number of security tools on an integrated file.

There is no single security sensor, currently available, which can accurately detect, locate and identify all the attacks and give a complete picture of the attacker strategy. We use various security sensors to gather the alerts, indicators and statistics from the integrated file. These fragments of captured evidence are fused together from number of selected five security network sensors tools and individual based captured evidence outputs.The information on the type and nature of attacks is also stored as a report, which can be used to collect all the suspicious packets in the ingress and egress traffic from the packet capture file.

The combination rule of fusing captured evidence from multiple security sensors achieving a greater level of admissible and acceptable trust worth evidence. The evidence was evaluated and analysis based on statistical methods using the confusion matrix theory metric or criteria of accuracy and False Alarm Rate (FAR) metric for determination of correct form of attacks. The steps involved are as follows:

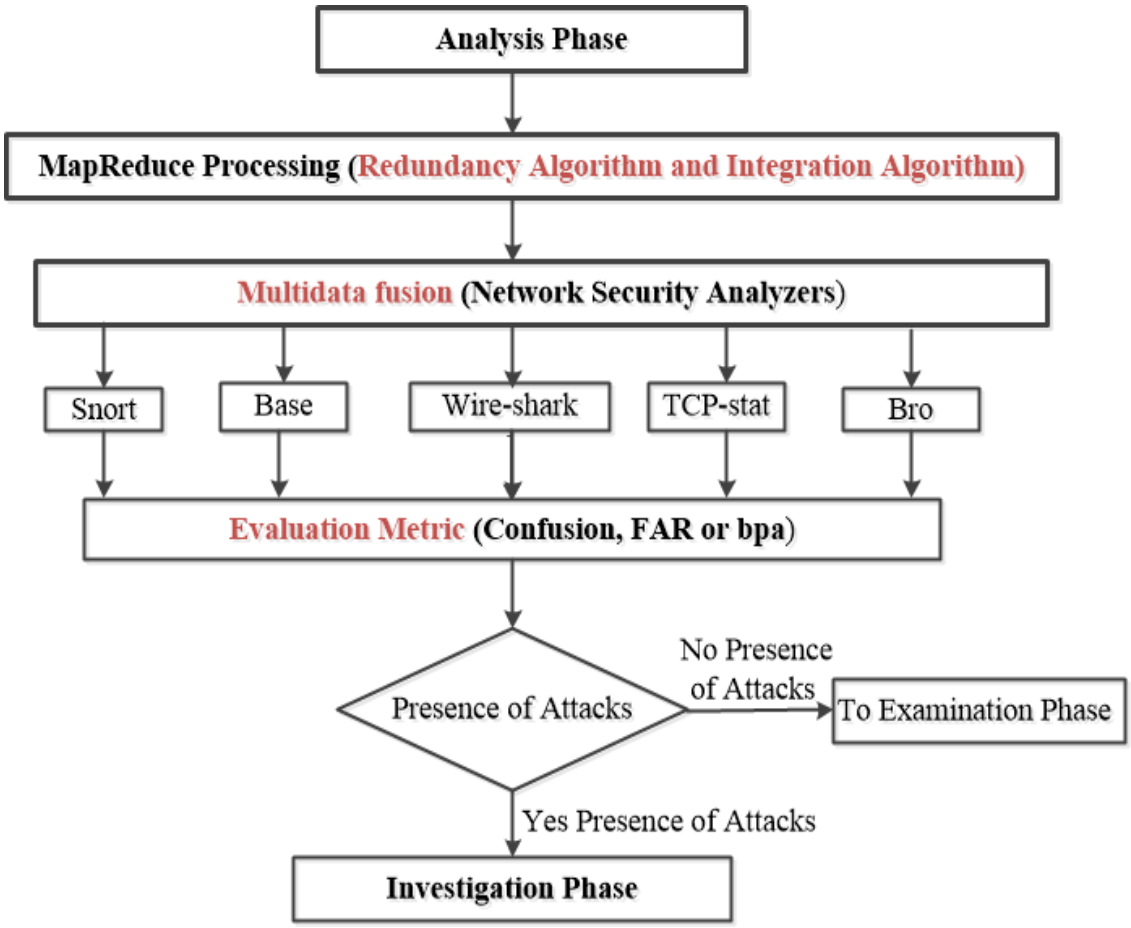
i. Various security tools are run on this file and information fusion is done using confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric to achieve comprehensive picture on the attacks.

ii. The multiple fused captured evidence are useful for identification of specific packets attacked from the unified packet format file and suspicious packets are marked as attacked. A new packet capture file was created from the attack packets, which is minimum in size and with maximum possible information as evidence.

3.6.3 Analysis Phase Framework Architecture Flow diagram for Multi Data Fusion

Figure 14

Analysis Phase Framework Flowchart Diagram for Multi Data Fusion

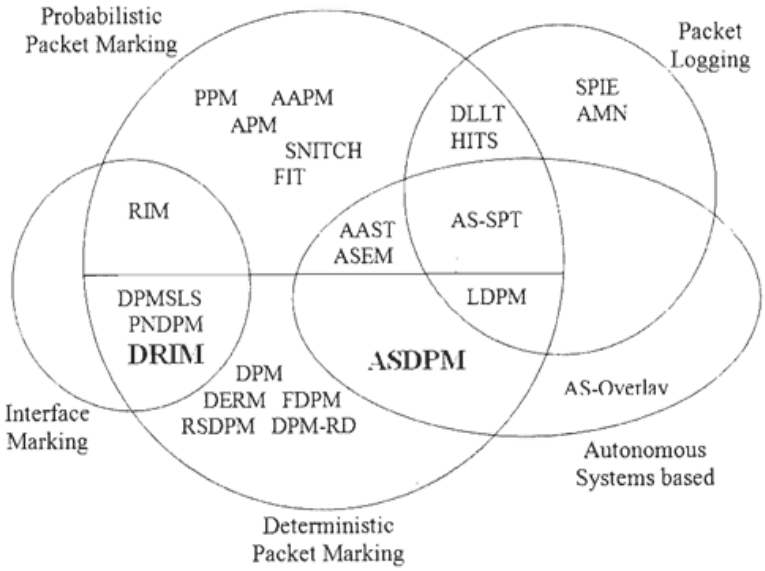


The proposed source track back and attribution approaches are Deterministic Router Interface marking (DRIM) and Autonomous System based Deterministic Packet Marking (ASDPM) as shown in figure 15 below, since they cover both Interface marking as well as autonomous system based. DRIM is able to mark interface packet from the first interfaces of the local router based on developed algorithm and ASDPM is able to mark packet from first interface of the autonomous system based router based on developed algorithm.

The basic idea in both of approaches was to record the access point as close as possible to the intruder or attacker. The information were obtained from the source where the intruder cannot manipulate. The marking of information ware done for each packet deterministically by the first ingress edge router. No other router modifies this marked information. To traceback an attack a single attacked packet was enough to detect the source of attack as it carries the mark attribution manipulated by an intruder traced by network security sensor.

Figure 15

Proposed Traceback and Attribution Techniques in Relation to Other Existing Techniques



The following three main values steps was used for the two proposed approaches

Step (1). The source autonomous system associated with a specific number

Step (2). The first ingress edge router address number

Step (3). A packet is associated with specific router interface

3.7 The Proposed Architecture Network Forensic Framework For Managing Security Incidents

The challenges with existing network frameworks for managing security network incidents associated with examination phase involves how to identify and correlate network attacks, the challenges associated with analysis phase involves on how to multi data fusion of various network traffic and the challenges associated with investigation phase involves how to traceback and attribute attacked network packets to specific source of intrusion. The major disadvantages with these frameworks are they deter the principles of evidence, which states that evidence should be admissible, authentic, complete, reliable and believable, and the criteria for admissibility of scientific evidence.

The proposed framework solves these challenges by integrating several classification forensic techniques incorporated in examination phase in order to identify, correlate and integrated algorithm to capture network attacked features based on network protocols. In analysis phase the forensic technique incorporated consist of multidata fusion based on open network forensic sensors. In the investigation phase the network forensic technique incorporated comprises of traceback and attribution based on proposed deterministic packet marking. The main idea is to manage network security incidents and to achieve attacked evidence that are admissible, authentic, complete, reliable and believable. The proposed framework employs a multi-level phase architecture, which takes into account the categories of network forensic techniques and algorithms used for identification, correlation, multidata fusion, traceback and attribution network attacks to specific source. The following are framework steps processing levels.

i. Identification and correlation of feature selection: Extracts attacked packets features from synthetic network events packets dataset based on ensemble feature selection examination phase forensic techniques.

ii. Examination phase algorithms for handling attacked packets network events redundancy and timestamp issues. One algorithm enables and identify specific dataset similarity by selecting the packet header and payload information. The another algorithm identifies related packets field of all transmitted network traffic examined and integrate from the timestamp selected within the dataset range to manageable reduced size.

iii. Analysis phase forensic techniques implements multidata fusion on attack network events generated by combination of network security sensors and output accurate evidence that are valid to ascertain attack occurrence.

iv. Investigation phase forensic techniques implements traceback and attribution technique based on packet marking, packet logging or hybrid approaches.

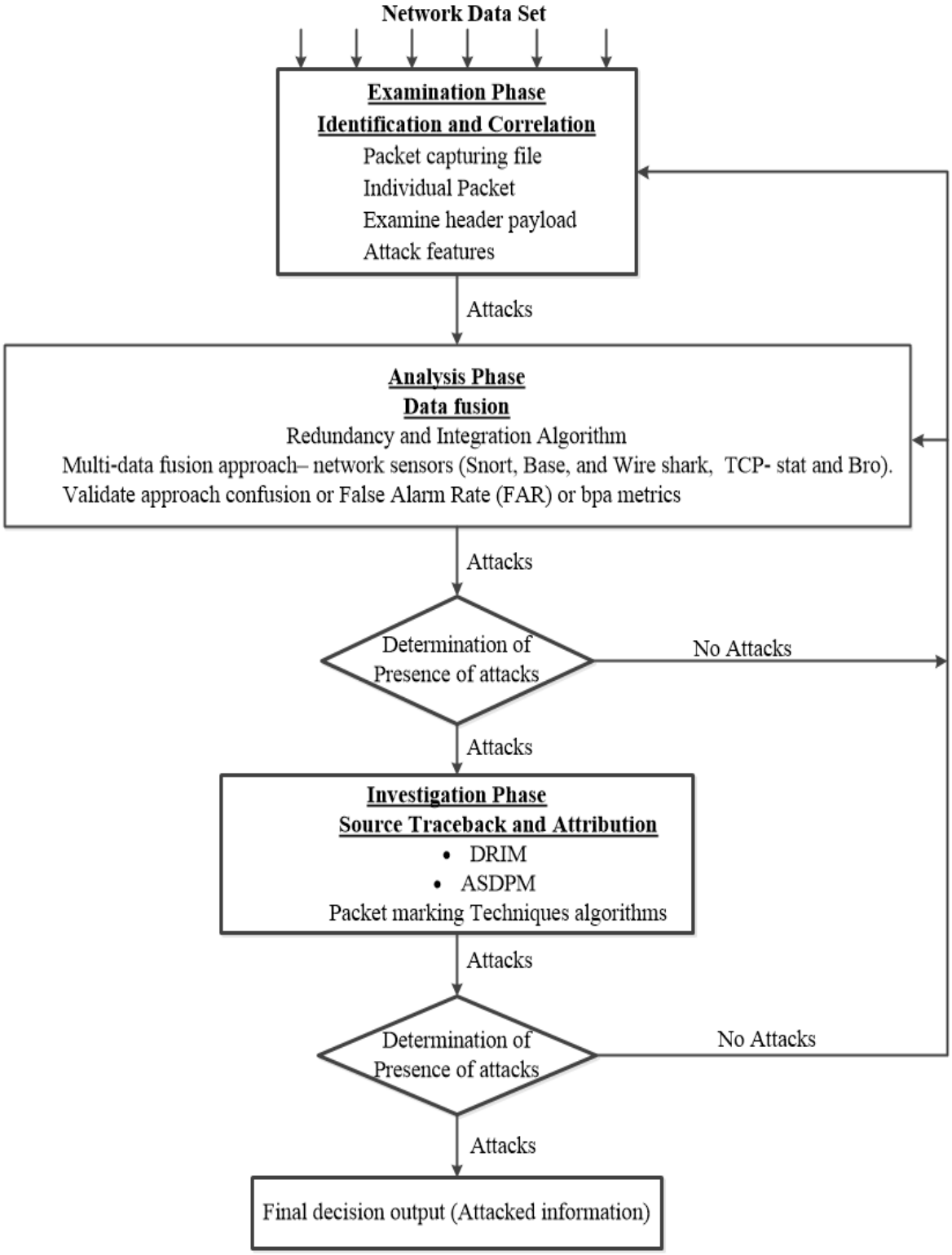
v. Statistical Correlation Tests calculate the strength of dependencies among the examine, analysed and investigated attacked packets attributes to discover and increased the evidence prove and confidence level

vi. Benchmark evaluates and compares with current works

3.7.1 The Proposed Network Forensic Framework for Managing Security Incidents

Figure 16

The Proposed Architecture Network Forensic Framework for Managing Security Incidents



3.7.2 Description of the Network Traffic Data Sets

In this study, NSL-KDD dataset was used derived from original KDD-99. The NSL-KDD dataset has the following characteristics according to (Moustafa, 2019).

i. The existing KDD99 dataset contain huge numbers of redundant records which causes the learning algorithms to be biased towards frequent records like port scan attacks, DDOS and CSS attacks thus preventing them from detecting unknown records that fall under less frequent attack categories like network spoofing attack that an attacker may take advantage of to penetrate a computer networks. Moreover, the evaluation of the network forensic sensors tends to be biased because of the existence of these redundant records in the test dataset (Godwin, 2020) due to technique used for evaluating the dataset based on network sensors implemented

ii. It has been observed from the results obtained by many researchers that the number of selected records from each difficulty level group is inversely proportional to the percentage of records in the original KDD data set. As a result, the classification rates of distinct machine learning methods vary in a wider range, which makes it more efficient to have an accurate evaluation of different and efficient techniques.

iii. The numbers of records in the train and test sets are reasonable, which makes it affordable to run the experiments on the complete set without the need to randomly select a small portion. Consequently, evaluation results of different research works will be consistent and comparable.

The simulated attacks in the NSL-KDD dataset fall in the following according to Moustafa, 2019):

i. DDoS or DoS, where attempts are to shut down, suspend services of a network resource remotely making it unavailable to its intended users by overloading the server with too many requests to be handlede.g. syn flooding. Relevant features include source bytes and percentage of packets with errors. Examples of attacks includes back, land, Neptune, pod, Smurf, teardrop

ii. Probe attacks, where the hacker scans the network of computers or DNS server to find valid IP, active ports, host operating system and known vulnerabilities with the aim discover useful information. Relevant features include duration of connection and source bytes. Examples includes IP sweep, n map, port sweep, Satan

iii. Remote-to-Local (R2L) attacks, where an attacker who does not have an account with the machine tries to gain local access to unauthorized information through sending packets to the victim machine in filtrates files from the machine or modifies in transit to the machine. Relevant features include number of file creations and number of shell prompts invoked. Attacks in this category includes ftp_ write, guess_ passwd, I map, multi hop, phf, spy, warezclient, warezmaster

iv. User-to-Root (U2R) attacks, where an attacker gains root access to the system using his normal user account to exploit vulnerabilities. Relevant features include Network level features – duration of connection and service requested and host level features - number of failed login attempts. Attacks includes buffer overflow, load module, Perl, rootkit

In the NSL-KDD data, each connection has 42 features (including its class label) that contain information about the session. The features can be divided into four categories: basic, content, traffic, and class.

i. Basic attributes that represent an alert and they are in IDMEF format. Examples of these attributes include timestamp, signature identifier, messages associated with alerts, protocol, IP source and IP destination addresses, source port and destination address, Time to live and identification field.

ii. Content features: The features of suspicious behaviour in the data portion should be captured in order to detect attacks. E.g. number of failed login attempts. Those features are called content features. The R2L and U2R attacks normally don't appear in intrusion frequent sequential patterns, as they have been embedded in the data portions of packets and only request a single connection. While the DoS and Probing attacks involve many connections to hosts and show the attribute of intrusion frequent sequential patterns.

iii. Time-based traffic features: Only the connections in the past two seconds are examined, which have the same destination host/service as the current connection, and of which the statistics related to protocol behaviour, service, etc. are calculated.

iv. Connection-based traffic features: Some slow probing attacks scan the hosts/service at an internal much longer than two seconds, e.g. once in every minute, which cannot be detected by the time-based traffic features, as it only examines the connections in the past two seconds. In such case, the features of same destination host/service connections can be re-calculated at an interval of every hundred connections rather than a time window.

A description of each of the 42 features is listed in the appendices 1 NSL-KDD dataset consists of KDD Test for data testing and KDD Train for data training. Each dataset has 41 features to recognize four types of attacks (DoS, U2R, R2L, Probe) and 1 normal state that occurs on a computer network. The features of the dataset include basic features of

TCP connection, content features from domain knowledge and traffic features. The set consists of a number of records called connections. A connection is a sequence of packets in a time frame when data flows to and from a source IP address to a target IP address under some well defined protocol. In the TCP protocol, a connection has multiple packets. For the UDP protocol, each connectionless packet is treated as a connection. Each connection is labelled as either normal or a specific attack type. There are 4,898,431 connections in the training set and 311,029 in the testing set.

To make the data more realistic, the data in the testing set is not taken from the same probability distribution as the training data. More importantly, the testing data contains 19 attack types not found in the training data. In our experiments, the training data is divided into two parts: 70% is used for building the profile; the rest is validation data and used to tune the parameters. This work will apply 5% KDD Train+ to get the proper cluster (6300 data), then 100% KDD Train+ (22544 data) for training data and 100% KDD Test+ (125973 data) for testing data. Dataset composed of normal data and 4 categories of attacks, namely DoS, U2R, R2L and Probe.

Table 2

The Composition of the Training and Testing Datasets

Data	Number of each category				
	Normal	DDoS	User to Root (U2R)	Remote To Local (R2L)	Probe
NSL KDD Train	67343	45927	52	995	11656
NSL KDD Test	9711	7458	200	2754	2421

The proposed model hybridizes data pre-processing technique, feature selection technique, structural based technique, and causal based and statistical correlation tests to

boost the overall correlation performance and measure the dependency strength among alert attributes**.**

3.7.3 The UNSW-NB15 Dataset

The UNSW-NB15 dataset (Moustafa,2019) for research purposes in network intrusion detection systems. It is a hybrid of attack activities include real traffic and synthesized activities in a computer network traffics and comprises of nine different moderns attack types as compared to fourteen (14) attack types in NSDL-KDD datasets activities of normal traffic that were captured with the change over time (Jihyung, 2021). The UNSW-NB15 dataset has forty-nine (49) features that comprised the flow based between hosts (like, client-to-server or server-to-client) and the packet header which covers in-depth characteristics of the network traffic. This data set contains 2,540,044 observations.

In UNSW-NB15 data set, there are nine categories of attacks:

i. Fuzzers: In this attack, randomly generated data is feed into a suspend program or network.
ii. Reconnaissance: Attacker gathers information from the system and stimulates the attacks.
iii. Shellcode: It is code used as the payload of a network packet to exploit network attacks.
iv. Analysis: This attack includes port scan, spam and HTML files penetrations.
v. Backdoors: Access of a system is gained by silently bypassing the security mechanism.

vi. Denial of Service where attempts are to shut down, suspend services of a network resource remotely making it unavailable to its intended users by overloading the server with too many requests to be handled.

vii. Exploits: The attacker exploits the vulnerabilities of the system through the known loopholes of the system.

viii. Generic: The attack is implemented without knowing how the cryptographic primitive is implemented and works for all block ciphers.

ix. Worms: The attack replicates itself to spread through the network.

Table 3

Attack Distribution in UNSW-NB15 Data Set

Category	Training	Set Testing set
Normal	56,000	37,000
Analysis	2,000	677
Backdoor	1,746	583
DDoS	12,264	4089
Exploits	33,393	11,132
fuzzers	18,184	6062,
Generic	40,000	18,871
Reconnaissance	10,491	3,496
Shellcode	1,133	378
Worms	130	44
Total Records	175,341	82,332

The UNSW-NB15 dataset has been divided into two Training datasets (82, 332 records) and a Testing dataset (175, 341 records) including all attack types and normal traffic records. Both the Training and Testing datasets have 45 features. The features scrip, sport, dstip, stime and ltime are missing in the Training and Testing dataset.

The UNSW-NB15 data set has several advantages when compared to the NSLKDD data set according to (Jihyung,2021).. First, it contains real modern normal behaviours and contemporary synthesized attack activities. Second, the probability distribution of the training and testing sets are similar. Third, it involves a set of features from the payload and header of packets to reflect the network packets efficiently. Finally, the complexity of evaluating the UNSWNB15 on existing classification systems showed that this data set has complex patterns. This means that the data set can be used to evaluate the existing and novel classification methods in an effective and reliable manner.

Several data mining techniques which includes data cleaning and pre-processing, clustering, classification, regression, visualization and feature selection have been implemented in WEKA (Waikato Environment for Knowledge Analysis) (Jihyung, 2021).WEKA also offers some functionality that other tools do not, such as the ability to run up to six classifiers on all datasets, handling multi-class datasets which other tools continue to struggle with tools. WEKA has tools for various data mining tasks. WEKA is considered as a landmark of data mining and machine learning as compared to other data mining and knowledge discovery tools and software like Tanagra, the Konstanz Information Miner (KNIME), and Orange Canvas (Kumar, 2021). Due to its Graphical User Interface (GUI) and easy access it has achieved a wide acceptance in every field. WEKA contains classes which can be accessed by other classes of WEKA.

The relevant classes in WEKA are attribute and instance. An attribute is represented by an object of class attributes which contains attribute types, name, type, nominal values of attributes. It is user friendly with a graphical interface that allows for quick set up and operation. WEKA operates on the predication that the user data is available as a flat file or relation, this means that each data object is described by a fixed number of attributes that usually are of a specific type, normal alpha-numeric or numeric values.

Table 4

Description of Explorer User Interface in WEKA

Data Mining Task	Description	Examples
Data Pre-Processing	Preparing a dataset for analysis	Discretizing, Nominal to Binary
Classification	Given a labeled set of observations, learn to predict labels for new observations	Bayes Net, KNN, Decision Tree, Neural Networks, Perceptron, SVM
Regression	Learn to predict numeric values for observations	Linear Regression, Isotonic Regression
Clustering	Identify groups (like., clusters) of similar observations	K-Means, EM,
Association rule mining	Discovering relationships between variables	Apriori Algorithm, Predictive Accuracy
Feature Selection	Find attributes of observations important for prediction	Cfs Subset Evaluation, Info Gain
Visualization	Visually represent data mining results	Cluster assignments, ROC curves

WEKA consists of several user interfaces which suited examination, analysis and investigation phase's security techniques during setup of network attacked features. But the functionality can be performed by any one of them as they give the same result. In WEKA user interface is classified into four categories;

i. **Explorer** – GUI, very popular interface for batch data processing; tab based interface to algorithms. Each of the packages includes Filters, Classifiers, Clusters, Associations, and Attribute Selection is represented in the Explorer

along with a Visualization tool which allows datasets and the predictions of Classifiers and Clusters to be visualized in two dimensions.

ii. **Knowledge Flow** – GUI where users lay out and connect widgets representing WEKA components. Allows incremental processing of data. WEKA components are selected from a tool bar, positioned a layout canvas, and connected into a directed graph to model a complete system that processes and analyzes data. Components available in the Knowledge Flow: data source, filters, Clusters, classifiers, Evaluation and Visualization.

iii. **Experimenter** – GUI allowing large scale comparison of predictive performances of learning algorithms. The experimenter, which can be run from both the command line and a GUI, is a tool that allows you to perform more than one experiment at a time, maybe applying different techniques to a dataset, or the same technique repeatedly with different parameters. For example, the user can create an experiment that runs several schemes against a series of datasets and then analyse the results to determine if one of the schemes is (statistically) better than the other schemes.

iv. **Command Line Interface (CLI)** – Provides a simple command line interface that allows direct execution of WEKA commands for operating systems that do not provide their own command line interface.

3.8 Performance Evaluation Metrics

The evaluation of the developed network forensic for managing security incident was based on quantitative approach as preferred method because of evaluation metric adapted such as confusion matrix and correntropy-variation technique among others depend on specific phases evaluated.

3.8.1 Evaluation of Examination Phase

The events specific to DDoS attacks, port scan attacks and CSS attacks were used as a case study to evaluate identification and correlation phase in examination phase. The attacks cannot be classified based on a single packet information and can only be decided upon observing a sufficient number of packets. The tables are created to perform statistical analysis and calculate the thresholds for various attacks. These derived attribute values are calculated from basic attributes of a packet and help in attack detection queries (Iftikhar, 2022).

Apart from selecting only substantial network traffic flow stream, the substantial features in the network traffic flow was incorporated during the network forensic framework for managing security incidents during implementation stage.We applied statistical the chi-square feature selection method x^2according to (Saputra, 2022) owing to its simplicity application at real-time.X^2 was used to calculate and measure the security intrusion attacks incidents of two independent variables in relation to their class label and then the top most variables ranked were picked as significant features using equation (1) according to (Saputra, 2022).

$$x^2 = \sum_{i-1}^{y} \sum_{j-1}^{c} \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}} \quad (1)$$

Where x^2 refers to the chi-square of independence, $O_{i,}$ represents the observed value of two variables and $E_{i,j}$ represents the mean of two variables according to (Saputra,2022).

The statistical thresholds were calculated from these features for numerous intrusions using statistical correntropy-variation technique equations 2, 3 and 4 which is a combination of correntropy according to (Yunfei, 2021) for measuring estimates similarities intrusion security attack instances, normal traffic and a variation threshold

when identifying attacks discovers. The nonlinear correntropy is a comparison function that exposes the associations among abnormal and normal observations, whereas the variation estimating how far the abnormal instances from the normal ones. Resulting features ideals were calculated from elementary features of a packet. This evidence were used to excerpt suspicious traffic from the incarceration records. The attacked traffic was transformed back into the origin traffic internment format permitting examination via existing current open source components and tools. Identification and Correlation of protocol features, which remain influenced by the intruders, is an endless practice. Identifying the intruders from traffic internment records requirements capability in checking packet data for over a period. A datasheet of entire potential traffic and at entire layers was organised over a period. The public methods and interactive forms of the intrusions also scrutinised and correlated. This assisted during determination and examining original network intrusions security incidents.

According to (Yunfei, 2021) a correntropy of two random variables (f1 and f2) is estimated using equation (2) as follows:

$$V_{\sigma}(f_1, f_2) = E[K(f_1, f_2)] \quad (2)$$

Where E[ ] represents the mean of the features $\boldsymbol{K_{\sigma}}$ represents the Gaussian function $\boldsymbol{\sigma}$ represents the size of the kernel computed through the following equation (3)

$$K_{\sigma}(.) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(.)^2}{2\sigma^2}\right) \quad (3)$$

The combined probability density function $(P_{FI,F2}(f1, f2))$ is anonymous, although a predetermined number of observations $(\{f_1, f_2\}^M \; i, j = 1)$was identified. The correntropy estimate is computed using equation (3) as follows:

$$V_{M\,\sigma}(A, B) = \frac{1}{M} \sum_{i\,j-i}^{m} K(f_1 - f_j) \quad (4)$$

To carry out the experiments, we chosen random samples based on proposed redundancy algorithm from the examined dataset with some captured and saved based on timestamp integration algorithm sample based on sizes between define range limit from selected the essential features using the chi-square method and investigating attack activities using equation 1,2,3 and 4 correntropy-variation technique.

3.8.2 Evaluation of Analysis

In cross evaluate of the framework effectiveness, the available dataset is randomly from each framework phase subsets and one of them is used as the test set and the remaining sets are used for building the classifier (Moustafa,2019). In this process, the test subset is used to calculate the output accuracy while the N_1 subset is used as a test subset and to find the accuracy for each subset. The process is repeated until each subset is used as test set once and to compute the output accuracy of each subset from each individual framework phase. The final accuracy of the system is computed based on the accuracy of the entire disjoint subsets. The Confusion matrix according to (Heydarian2020) are to evaluate the performance of the analysis phase implemented based on imperative selection features.

According to (Heydarian2020) a multi-label confusion matrix is a specific table layout that allows visualization of the performance of each forensic networked sensors. Each column of the matrix represents the instances in a predicted class, while each row represents the instances in an actual class. The name stems from the fact that it makes it easy to see if the system is confusing two classes (like. commonly mislabelling one as another). In the binary class networks security sensors, the intrusion detection system is mainly discriminate between to classes, "Attack" class (malicious threats or abnormal data) and "Normal" class (normal data).

Table 5

Confusion Matrix

		Predicted
	Normal	Attack
Actual normal	TP	FP
Actual attack	FN	TN

In Table 5,the elements of the classification metrics are CM= {TP, TN, FP, FN}, where TP (True positive) is the number of the correctly classified attacks, TN (True Negative) denotes the number of the correctly classified normal rows, FP (False Positive) is the number of the misclassified attacks, and FN (False Negative) refers to the number of the misclassified normal records. The selected measurements used for performance validation and benchmark in this research for the enhanced SAC, enhanced CAC and proposed HAC are justified and described in the following:

a. Structural-based Alert Correlation (SAC). The applied measurements for validating the enhanced SAC include:
    i. Clustering Error (CE) is the number of alerts that are wrongly clustered.
    ii. Error Rate (ER) is the percentage of wrongly clustered alerts, ER = (CE ÷ Total number of alerts observed) x 100,
    iii. Accuracy Rate (AR) is the percentage of alerts that are accurately clustered as they should be, AR = 100 – ER, and
    iv. Time is the algorithm processing time in seconds.
b. Causal-based Alert Correlation (CAC). CAC model is concerned about how good it can classify the known and new alerts. Therefore, this research implemented the standard classification measurements in validating and evaluating the enhanced CAC model. They are:

i. TPR: TP/ (TP+FN), also known as detection rate (DR) or sensitivity or recall.

ii. The False Alarm Rate (FAR) is the rate of the misclassified to classified records, as denoted in Equation (5). Equations (5) and (6) allow calculation of the False Positive Rate (FPR) and the False Negative Rate (FNR), respectively. FP/ (TN+FP) also known as the false alarm rate.

a) FPR = FP/ (FP +TN) (5)

b) FNR = FN/ (FN +TP) (6)

iii. Precision (P): TP/ (TP+FP) is defined as the proportion of the true positives against all the positive results.

iv. Total Accuracy (TA): (TP+TN)/ (TP+TN+FP+FN) is the proportion of true results (both true positives and true negatives) in the population.

v. F-measure: 2PR/ (P+R) is the harmonic mean of precision and recall.

For the proposed correlation models, the optimal setting of the parameters is done based on repeated trials over a period of specified time limit. But, for retesting the current works, the parameters are set based on their information given in their published research resources or papers. In the case that information is not given, the default parameters are adopted. Almost all types of computers can be used to code and run the proposed correlation models because there is no specific special hardware is needed. Furthermore, all the software and tools needed are either freely available or easily purchasable. Nevertheless, the minimum computer hardware requirements are Core-i5 processor, 2.5GHz speed, and 8GB RAM. But a higher specification is better for maximum installation and smooth experiments.

3.8.3 Evaluation of Investigation Phase

A detailed study of evaluation against various performance metrics, in comparison with other related techniques, validation of the traceback and attribution packet marking approaches. Evaluation performance and observations between ASDPM and DRIM after conducting simulation between the two proposed techniques approaches. The comparison between ASDPM and DRIM with other existing techniques in terms of specific evaluation metrics with various traceback approaches. The evaluation metrics were set on ISP in terms of packets required for traceback after an attack, deployment effect, overhead processing, overhead bandwidth, required storage memory, prevention of attack evasion, safeguard, scale of attackers accommodation, number of implementable network devices, capability to handle major DDoS attackers and capability to traceback altered packets. The two approaches ASDPM and DRIM when compared to other existing approaches in order to identify specific advantages based on determination metrics used during evaluation process as summarize in table 18.

CHAPTER FOUR

DATA ANALYSIS, PRESENTATION AND DISCUSSIONS

4.1 Introduction

This chapter presents the findings, interpretations and discussion of the research objectives as stated in chapter One. The proof of frameworks, outcomes and examination obtain are found on the recommended forensic network framework for managing security incidents. The major objective of the research is to develop a network forensic framework that addresses the challenges of examination, analysis and investigation phases. To attain this, the research study is accompanied on four main parts. They include investigation of the current network forensic frameworks challenges in examination, analysis and investigation phases, to analyse the main components and tools that address the challenges of examination phase, analysis phase and investigation phase then develop a network forensic framework for managing security incidents based on individual main phase challenges.

4.2 Investigated Challenges Associated with Examination, Analysis and Investigation Phases

The existing network forensic framework as reviewed in chapter two highlighted major challenges in existing with examination, analysis and investigation phases. The key challenges for specific phase are discussed below.

4.2.1 Challenges Associated with Examination Phase

Examination phase in the present frameworks are not capable to identify useful network events and detect attacks. The various protocols features being used by attackers are not listed to capture attack evidence and correlation of the attacks features with possible attacks scenarios is not performed. The various forms of network attack are difficult to

identified and no validation are done before decision making decision to proceed with investigation analysis. Effective frameworks are not in place to identify attacks features of packets capture format. Identification of protocols attributes which are being altered by attackers at the network, transport layer and application layer in TCP/IP protocol suite is major challenge for forensic investigators. These key attributes of the protocol are very challenging and give forensic investigators difficulty in extracting and determining the specific form of intrusion. The main shortcoming of examination phase in the current forensics networks frameworks are they not in position to examine valuable network actions and detect at least illustrative form of attack that would possibly be essential as set of admissible evidence that can be presented by forensic investigators when addressing network security incidents.The existing examination phase lack effective framework for identification and correlation of network dataset with attack features of packet captured format and related protocols used by attackers in TCP/IP protocol stack to launch security as discussed in following section.

Network packets security sensors monitoring tools are designed, configured and used in by forensic investigators in examining any existing form of network security incidents. In order to achieve this, the main requirement is to capture complete data packets file and examine each packet captured format in detail as they pass the network. Packets can be captured in libpcap (.pcap) as shown in figure 11 files by running packet sensor security sniffers like tcpdump tools. These packet sensors detect collects and assist in determining essential information pertaining the flow of network traffic. The existing examination framework lack basic requirement of packet capturing threshold for management of network security incidents.

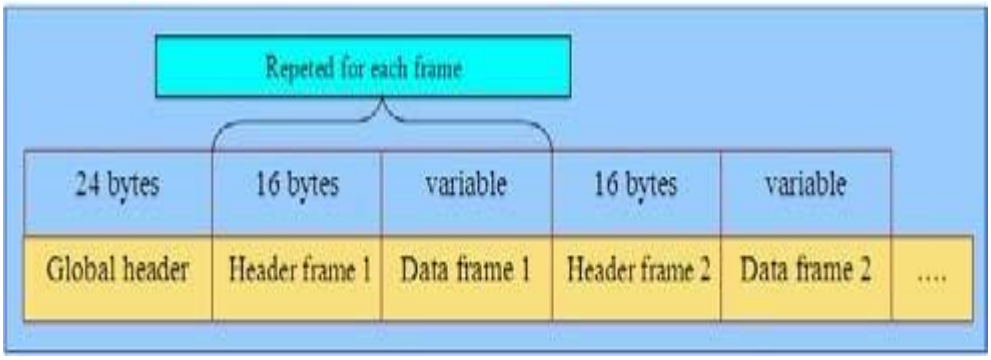
Libpcap.cap is network sniffer software that has a library extension file format that implements detected and captured all essential information of attacked network traffic

data format. The file contains all essential packet header global evidence recorded for each examined and captured attacked packet. The current network forensic frameworks does not implement libcap.cap file when capturing network traffic packetas they appeared on the network leading less admission evidence is relied upon by forensic investigators. The captured file comprises of the first N bytes of each full examined and captured network traffic packet. The snapshot length contains the rate of N packets bytes with details captured form of attacked packets. The rate N be larger than the established possible packet to accept there is no network packet in that was captured and examined distributed with a distinguishing value of 65535.

The global packet header is positioned as the first one in the libcab.cab file with packet fields specifying the file type format, order of the bytes and specify number version. It also identifies the accurate time to live (TTL) in seconds during traffic network packet transit, the time local zone and packet capture format precise time stamps. The following figure 17illustrate Libpcap file format packet examined and captured with N length specified by the each specific field snaplen details.

Figure 17

Libpcap File Format



The first field segment detail global header containing 24 bytes followed by packet header frame containing 16 bytes and variable data frame header alternating successive. The packet header detailing evidence fields information providing specific date and time

in (*ts_sec)*when the packet was examined and captured. It also examines and capture the evidence in microseconds (*ts_usec)* offset. The quantity of data packet truly examined, captured and saved in the libcap.cap file is illustrated in form of inclusive length (*incl_len).* The packets field provides entire packet in its original length form captured and presented on the networking form of (*orig_len).* The real instantaneous evidence of a packet is tracks from the packet header as records dribble of bytes detriment particular byte assignment in present in form of incl_*len* same as its packet original length examined and captured.

The TCP/IP protocol stack provides diffident, operative, open network inter-communication system arrangement in an academic and common environment. The five layers of TCP/IP protocol with associated network protocols as

shown in figure18. Intruders use the vulnerabilities and weakness in the protocol stack to exploit and promote forms of attacks in case of network security incidents. Essential protocols in each partition layer and the attacks corresponding to the protocols are examined and discussed in subsequently sections.

Figure 18

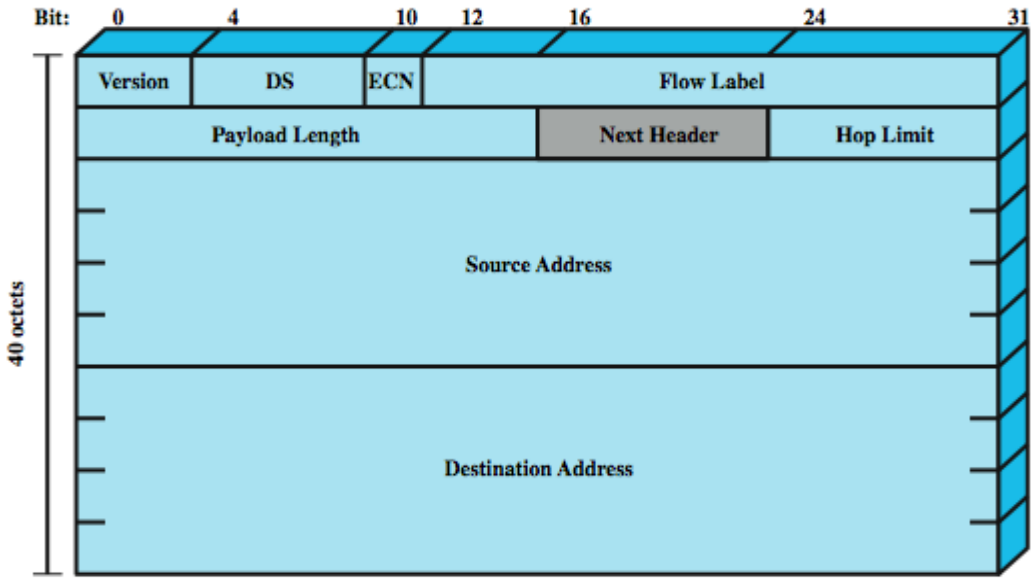
TCP/IP Protocol Stack and Corresponding Protocols

<table>
<tr><th>TCP/IP MODEL</th><th>PROTOCOLS</th></tr>
<tr><td>Application Layer</td><td>SMTP, FTP, DNS, DHCP, IPSec, SSL, HTTP, NFS, ICMP</td></tr>
<tr><td>Transport or
Host to host Layer</td><td>TCP, UDP, SCTP</td></tr>
<tr><td>Internet Layer</td><td>IP, ARP, RARP, ICMP, IGMP.</td></tr>
<tr><td>Network Access Layer</td><td rowspan="2">NO UNDERLINE PROTOCOLS</td></tr>
<tr><td>Physical Layer</td></tr>
</table>

The core purpose of IP protocol at the TCP/IP stack internet layer is to make sure that network traffic packet is routed using defined path to designated location from the host source. The network traffic encapsulated packets pass from end to end a numbers of routers where individual router establishes the subsequent hop and step for the network packet toward its destination. There is probability where by two packets originating from same source network devices transit towards the same destination but may transit using dissimilar route; this could lead to inconsistent examination and capturing of network packet file libcap.cab format. The main file format structure architecture for IP is as shown in Figure 19.

Figure 19

Internet Protocol Packet Structure

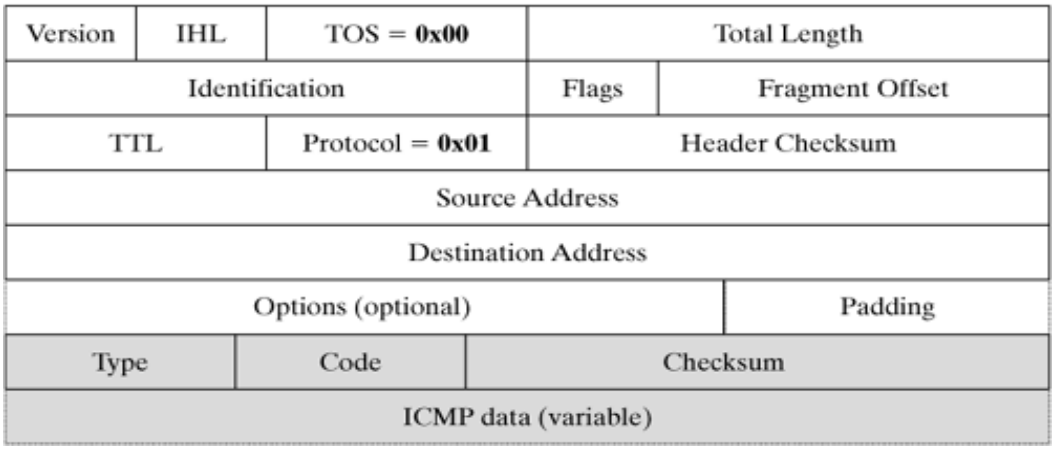


ICMP enables one way transmission of network traffic from the source network device to specific destination host within the network. It has several data structures which specify its own packet format and it's transported within payload of IP header. Normally destination network devices or routers utilise ICMP to notify source network devices in case there are errors during transit of packet or during processing of datagram. It also

permits routers engaged in control of messages or send errors to other network hosts or other routers involve in transmission of packet. It also facilities network layer for transmission of packets between communicating network devices. These protocols implement two types of processes; “probe the network with request and reply operation” and “report non-transient error conditions". The protocol messages are classified into two categories: ICMP Error Messages and ICMP Query Messages. Each message of ICMP is allocated message unique identity number that is used in identifying each individual message send. There also additional assigned code number that is used to specific type of each ICMP message. The protocol header is shown in Figure 20 below.

Figure 20

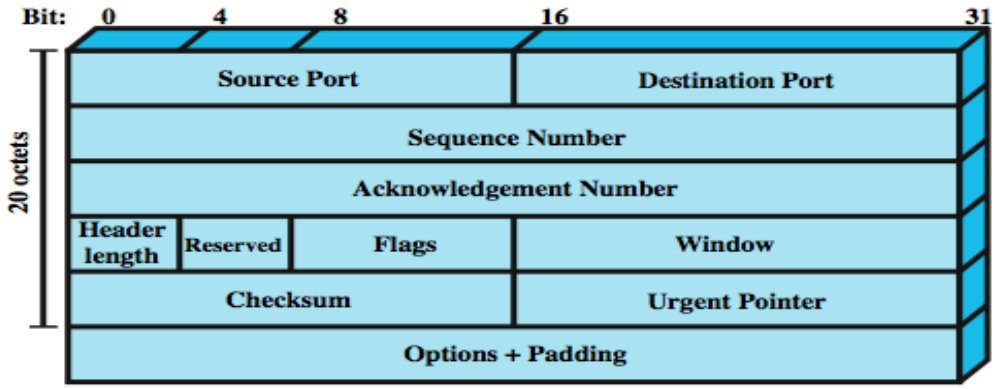
Internet Control Message Protocol Packet Structure



TCP facilitates connectivity between source and destination of network devices in form of “connection oriented services”. It delivers assurance of packets and guarantees delivery of packets in orderly format. It implements three-way handshaking, timers, packet sequence number and acknowledgment mechanisms where facilitating the connectivity between communicating devices and messages transmission. The TCP structure is shown in Figure 21.

Figure 21

Transmission Control Protocol Packet Structure



UDP is basically an application interface to IP. It is connectionless oriented protocol that offers and implements processes of making sure that datagram application is send to another application within network communication devices. The UDP format thin layer requires minimum overheads and necessitates by making sure that it take charge in its obligation for recovering application errors. The UDP structure is shown in Figure 22.

Figure 22

The UDP Packet Structure

HTTP protocol is mainly used for accessing information on World Wide Web (WWW). Information is mainly transmitted between servers and clients via HTTP messages. These messages are interpreted through clients' browsers and interpreted by the servers. Messages request and response format are similar for both clients and servers. Messages

request consist of request header, request line and request body. Message response consists of status line and other specific actions methods. It is also used to offers other functions such as multimedia information systems, collaborative and distributive application-level protocol. It is stateless generic protocol use for hyper texting, distributed object system management, naming servers, request methods through extensions such as headers and error codes. A partial list of methods packet structure is given in Table 6.

Table 6

HTTP Methods Packet Structure

Attack	Protocol Fields Examined
GET	Requests some massages from the server
POST	Sends some clients massages to the server
HEAD	Requests massages about a document
PUT	Sends server messages to the client
TRACE	Echoes the inbounding request

Examination phase lack effective framework for identification and correlation of network dataset with attack features of packet captured format and related protocols used by attackers in TCP/IP protocol stack to launch network security incidents.

4.2.2 Challenges Associated with Analysis Phase

The challenges of network forensic are complexity of data captured with attacked information which are in raw format and make them difficult to formulate critical decision by forensic investigators. It is not impossible to investigate but it requires a lot of time and skills since the quantity of data to analysis are massive. It is very challenging, expensive and difficult in analysing each individual network packet. It requires a lot of storage to store large volume of network packets from high speed

network which complicate analysis phase to capture attack packets. Storage resources cannot accommodate large and high speed exhausted by enormous data. The archival time of network traffics logs are limited and old data has to be overwritten before analysing.

These challenges are multiplied when packets are collected from many clients within the network forensic investigation. The existing network frameworks analysis phase does not have mechanism to integration all these individual packets from multiple clients to a single independent file before it pass to investigation phase for evaluation process. Independent file may also have redundant information when the packets are captured from examination phase. Dataset in packets capture format are not available for validating frameworks proposed for post–mortem forensic investigation**.** The popular dataset KDDs Cup 99 metrics is more than ten years old and has attacks which are costly and obsolete.

Real world dataset do not exist for validating developed network forensic framework. They are many network incidents which occur in various business and organisation. These datasets in packets captured in examination phase contain payload with confidential encrypted information specified as per packet captured and type of protocol used to launch network security incidents. Validation of attack generates important information about sources of the attacks and facilitates data reduction. The generated information facilities origin of attack source but there no technique to fusion all attack information after identification from examination phase to increase level of confidence and accuracy for critical decision before proceeding to investigation phase.

The main challenge association with analysis phase is how data fusion network dataset are with attack features is performed after identifying and correlating in examination

phase. Attack information and alerts are not taken from various security sensors as no single security tool can gives comprehensive alert information. Information is not considered from tools in comprised network for reconnaissance and data fusion of these alert and statistics are not be performed to validate attack. They lack effective framework to identified attacked packets captured from examination phase to eliminate redundancy, reduce level of confidence, accuracy as well as they use old data dataset used and evaluation metrics to validate the captured attacked packets.

4.2.3 Challenges Associated with Investigation Phase

The main function investigation phase in network forensic framework is identification of main origin of network intrusion incidents. The weakness of the existing framework analysis phase focus on DDOS attacks and they proposed a techniques aimed at prevention, detection and mitigation of attacks. This technique has not been implemented and deployed because it is very complex. IP traceback technique is an approach use to identify the actual source of an attacked packet in network setup but does not attribute to particular host. IP traceback is an important approach that gives assurance of the source of an attacker in post- mortem investigation procedures.

The main drawback of IP traceback when using TCP/IP approach is enabling IP address header for identification of packet source and packet destination in case of IP spoofing which lead to difficult in IP traceback when determining actual source of attack.NAT approach allows many clients after a gateway and the entire packet sent by these clients have gateway public IP address as their source address. The attacker may use indirect attack using comprised host and zombies before reaching the victim. The main challenge non-trial task is how to reconstruct the footprint path of an intruder and traceback the main source of intrusion. The existing framework approaches used does

not record the access point as close as possible to the attacker which allow narrowing the attacker near to the source of attack.

The methodology used doesn't identify attack packet mark that can assist to trace the original source of intrusion. The methodology implemented does not factor in forensic traceback and attribution in cases where by intruders are intelligent create and send broadcast packets, several intruders may performance attacks using methodology which is well-coordinated taking into consideration in mind awareness of traceback measures and techniques in place. Routers lack powerful processors leading to less packets been processed and storage capabilities to store enormous packets after it has been processed. There are number of routers which are not part of participation in pass attacked packets and other may facilitates pass of attached packet which make traceback and attribution more challenging task as well. Routers between the hosts are usually stable but packets can be recorded or lost. Some of the attack stream may consist of only few packets and investigation needs to be performed by means of partial evidence, which give means of investigating the original source of intrusion, by traceback and attributing to specific network device as well as intruder.

The main challenge of current forensic framework associated with investigation phase is traceback and attribution of source attack. Analysis of alarms, records and network packets traffic doesn't provide particular origin of network intrusion. Suspicious source address can be determined in the analysis phase but IP spoofing will hide the true about attacker. Used of IP address as an approach used in traceback and attribution in investigating source of attack is a main challenge. This approach doesn't guarantee traceback and attribution of intrusion to particular host within the network that with admissible evidence.

4.2.4 Network Forensic Security Techniques Addressing Challenges of Examination, Analysis and Investigation Phases

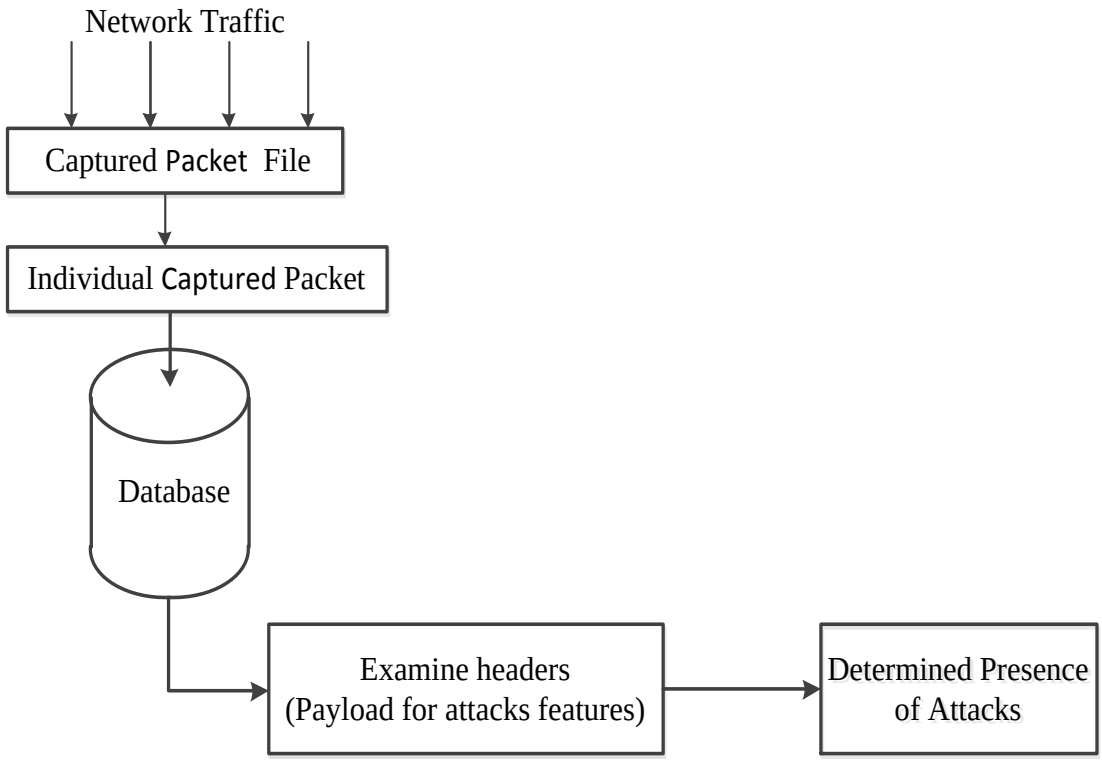
The main essential component and tool that addresses challenges of examination phase is identification and correlation of network events. The challenge of the current forensics networks frameworks for managing security incidents particularly examination phase are identification of and correlation of network events and select the least characteristic set that would possibly present network security incidents of attacked packets as admissible evidence. The network packets protocol key attributes manipulated and deployed by attackers should be identified and correlated with network events to particular network security incidents. These protocol key attributes is retrievable using proposed identification and correlation of network events model in figure 23 and transfer them to database for analysis purposes. The statistical attributes identified and correlated onsets are calculated for numerous specific form of intrusion.

These attributes identified and correlated serve as information containing suspicious feasible attacked admission. This information are collected, examine, identified and correlated from log files and captured trace system as admissible evidence and suspicious attacked packets. The identified and correlated attack evidence packets reconstructed to its original format using existing network open source security sensors analysis using existing open source tools that harnesses admissible confidence. The objective of the proposed identification and correlation of network events tool within the examination phase is to address the challenges of minimum the amount of packet captures files for efficiency and shorten time taken during analysis phase. The flow diagram model for examination phase framework for the identification and correlation of network events with attack features associated with examination phase is shown in flow diagram infigure23 below.

4.2.5 Examination Phase Framework for the Identification and Correlation of Network Events

Figure 23

The flow diagram for examination phase framework for the Identification and Correlation Of Network Events



The examination phase framework for the identification and correlation of network events captures network packets from suspicious attacked network hosts and transfer the attached packets to analysis forensic database server. The details evidence captured from libcap.cap libraries contains packet header, captured number of packets, and the length of captured packet. The main libcap.cap library file used to extract packet header attributes snaplen details is perl language module Net::Pcap, which enables to capture and correlated all packet length size stored.

The details information contained in the libcap.cap file for numerous packet headers such as TCP, IP, UDP and ICMP protocols were extracted from the file and stored in the

forensic network database Net.Pcap server respectively. The libcap. capenables encoding of captured packets and extracts numerous network protocols attributes that specify each packet fieldname. The packets encoded and extract corresponding to ver, TTL, flags, ID, offset bit, cksum, proto,srcip, destip, and options attributes.

The specific associated with TCP protocol packet header attributes captured, and correlated by Net.Pcap file includes src_port, dest_port, seq_num, , ack_num, flags, hlen, flags reserved, urg, winsize and options. The specific associated with UDP protocol packet header attached attributes captured and correlated by Net.Pcap file includes src _mac, dets_mac, ID, tos, flags, srcip, destip, TTL and options. For tracking the non-stationary properties of flow identifiers, we apply the 'count' functions to determine all possible combinations of these flows, as follows.

- *Select COUNT(*) as flows, srcip, dstip from network_data group by srcip, dstip;*
- *Select COUNT(*) as flows, srcip, srcport from network_data group by srcip, srcport;*
- *Select COUNT(*) as flows, dstip, dsport from network_data group by dstip, dsport, srcport;*

In the above queries, flows denote the number of flows, which occurred between any two attributes, srcip refers to the source IP address, dstip refers to the destination IP address, srcport refers to the source port, dsport is the destination port, and proto refers to the protocols. Every query retrieves the number of flows which takes place amongst the features.

All networks protocols attributes are captured and correlated, then each specific protocols attributes are transferred to unique created database table. Each packet is recorded with unique time stamp associated with frame number generated automatically.

The database tables are useful during performance of statistical packet analysis and calculation of numerous attacks thresholds. These statistical analysis and calculation subjected to theory metric to test the validity of confidence admissible evidence. In this research study, the metric used to test the validity of confidence of open source security sensors was confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metrics applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors. A single attack captured and correlated packet cannot be used for determine the form of attack but sufficient number of packets based on other security sensor is sufficient to conclude that the formalized the type of attack based on confidence of admissible evidence.

The networks open sources security sensors creates a record of suspicious protocol attributes addresses and Net.Pcap module in Libcap.cap file transfer the specific attributes to forensic database server for analysis purposes. The identified and correlated suspicious protocols attributes are transferred to specific unique database table using the alert protocol packet attributes. Suspicious packets may be identified and correlated within flowing legitimate network traffic by filtering the evidence statistical thresholds stored from unique database tables. The network events protocol attributes identified and correlated were examine specifically from TCP/IP protocol stack at network, transport and application layers.

TCP/IP protocol stack architecture model does not conceptualised and offer security consideration mechanisms in terms of authentication, privacy and integrity during design stages. An intruder exploits these vulnerabilities and weakness and of TCP/IP stack to initialise all form of network security intrusion and breach. There are many classifications of attacks which are categorised according to individual protocols in network and transport layers respectively. The two common form of intrusion considered

and examine in these two layers are port scan and distributed denial of service intrusion examined. The security forensic experts consider the greatest and least challenges form of attacks is web based attacks, since it is difficult to understand all the risks in relation to information availability, confidentially and integrity. The web based intrusions are significantly and persistent altered than other types of intrusion. The types of intrusion occur occasionally at the application layer of TCP/IP suite.

The beginning of initial types of web applications was inadequate in their capability to offer some extra evidence than a catalogue you might obtain in the e-mail. Static HTML was provided as a tool to display pictures and inert information. Consequently, as the internet and web access became more and more ubiquitous so too did the needs of those users who were accessing web applications. The web has enable user flexibility and convenience when using the application such as browsing, posting, uploading, downloading and searching.

CGI protocol offers a form for web based users and systems to communicate with web pages and sites by linking meta-data and information into forms. Upon submission, back end CGI scripts would process this data presented and represent HTML back to the end user. CGI relates with end systems and users efficiently turn out to be one of most the foremost web applications that attracts most intrusion more frequently to launch and exploit networks. There are current frameworks web applications has incorporated new security attributes allowing users more interactive features, power and flexibility. This current framework web application includes ASP.NET, ASP, J2EE, Ruby, Rails, AJAX and PHP.

Securing web applications has become incredibly important, as the information processed by web applications has become critical to corporations, customers,

organizations and countries. Web applications manage a wide array of information including financial data, medical records, social security numbers, intellectual property and national security data. There is even a serious need to record digital evidence to be used for investigating the attack or at least understand the attacker's methodology.

We develop a vulnerable website and illustrate the most commonly occurring attack in the cyberspace, Cross-Site Scripting (CSS) attack. We log the data on the web server, identify the attributes, and correlate the network events with the attack. The web application attacks involve payload and are more complex than attacks in the lower layers. Privacy protection of legitimate users is to be ensured while analysing the payload.

CCS is a type of computerized intrusion launch over the internet that aimed on web application servers, i.e. a website with forms to be filled in. This intrusion is dissimilar to other most forms of intrusion because it uses a website vulnerabilities application to permit a mischievous personally to intrude other systems and comprises users' services. Generally, CSS is mainly categorize into three main parts namely;

i. Intruder or attacker
ii. Comprise website application
iii. The victim or user of the website

CSS attacks are those attacks against web applications in which an attacker gets control of a user's browser in order to execute a malicious script (usually an HTML or JavaScript code) within the context of a trusted web application's site. As a result, and if the embedded code is successfully executed, the attacker might then be able to access, passively or actively, to any sensitive browser resource associated to the web application (e.g., session IDs, cookies, etc.).

CSS attacks are of two types: HTML Persistent attacks/Injection attacks and non-persistent attacks or reflected CSS attacks/stored attacks. HTML Injection attacks the malicious JavaScript code injected by the attacker into the web application is persistently stored into the application's data repository. In turn, when an application's user loads the malicious code into its browser and since the code is sent out from the trusted web site's application. This permit the client browser cookies code to open its repository. Thus, the script is allowed to steal victim's sensitive information and send the same to the website of the attacker. The code evades the simple normal step procedures of security of web application kernel implementation source code engine. Any source code engine controls the entry of information to restrict those codes that belong to the similar source where the evidence was implemented. Persistent CSS attacks are traditionally associated to message boards web applications with weak input validation mechanisms.

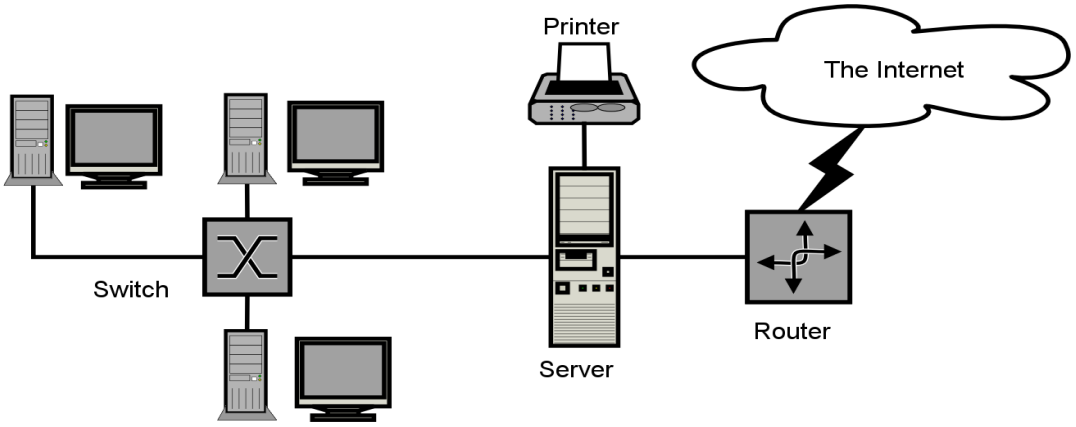
Reflected CSS exploits the vulnerability that appears in a web application when it utilises information provided by the user in order to generate an outgoing page for that user. The malicious code itself is directly reflected back to the user by means of a third party mechanism, an alternative means of keeping the mischievous script inserted into a message by the intruder. By using a spoofed email, for instance, the attacker can trick the victim to click a link, which contains the malicious code. If so, that code is finally sent back to the user but from the trusted context of the application's web site. The client browser implements the script within the expected domain application's and may permit it to direct associated information (e.g., session IDs and cookies) vast storage application without compromising the similar source procedures of the interpreter client browser's. Non-persistent CSS attacks are the most common type of CSS attacks against current web applications, and are commonly combined together with other techniques, such as phishing and social engineering.

We integrate packet capture files from various clients to effect data reduction and solve the quantity problem. The networks sniffers tools (tcpdump or Wireshark) and configured in the victim network as network forensics was possible only when the clients were prepared to record forensic evidence and particular attack was analysed depending on the captured packet collected on compromised machines.

Libcap. cap executes network packet traffic at a low-level by reading, writing and analysing all files captured. The libcap.cap library file provides the capturing and filtering of packet through execution engines of many other open source and proprietary network security sensors tools such as IDS, protocols analysers' and packet sniffers. The extension file of Libcap, cap implements identification and recording of simple file format used by intruders to launch attacks from captured stored network information. The packet sniffer tools log the network traffic information on each client in a packet capture file. The packet capture files collected from many hosts are integrated into a single file in the forensic server as shown in figure24 below in the proposed packet integration capturing network framework.

Figure 24

The Framework for Packet Capturing Integration



The main objective of integrating all files captured from numerous clients into one single file so that entire attacked information are available at one place as shown in figure 24. It is also easy to analyse a single packet capture file against a series of security tools. The integration will also result in data reduction, as some of the packets collected by the multiple hosts will be similar. There will be broadcast and multicast packets, which toggle by all hosts. The major issue to integrate the files will be to handle the timestamps of redundant packets with same information. The other issue is to identify which files to be integrated directly and which files need a redundancy check before integration. This decision depends on the location of the compromised hosts from which the files are collected, whether the system is within a particular subnet.

The packet captures (.pcap files) are collected from compromised systems, which are identified in a network. These files are integrated by converting them to a database, identifying unique packets and recreating a single file. This is achieved by developing an algorithm that identify files for integration and another algorithm that handle integration after checking redundancy.

The Perl language module Net:pcap file library extension enables conversion of packets captured files contents and stores them automatically to unique database table. The packet captured contents file, which contains packet header and payload, are also part of what is stored into forensic database table. If ‘n’ packet captures are to be integrated, there will be ‘n’ database tables. The ‘UNION’ operation is performed on these ‘n’ tables resulting in a single table. When payload is same for a packet in two tables, only one copy is placed with a header from any file. This single table created is reconverted back into a packet capture file resulting in a single integrated file. Once the integrated packet capture file is ready for analysis, it is passed on to the fusion process.

Many of the existing open source network security tools can be used for specific tasks in network forensics. Nevertheless, proprietary network sensors tools lack of strength and functionality that are explicitly constructed and implemented to captured only specific type of network traffic with different level of evidence. We used the open source network security and monitoring tools as shown in figure 25 to read the packet capture file post attack, and give various alerts and indicators. The attacked information generated by ‘m’ number of security tools on ‘n’ number of packet captures will be same as the alerts generated by ‘m’ number of security tools on an integrated file.

There is no single security sensor, currently available, which can accurately detect, locate and identify all the attacks and give a complete picture of the attacker strategy. We use various security sensors to gather the alerts, indicators and statistics from the integrated file. These fragments of captured evidence are fused together from number of selected security network sensors tools and evidence. The evidence was evaluated and analysis using the confusion matrix theory metric or criteria of accuracy and False Alarm Rate (FAR) metric for determination of correct form of attacks.

Confusion matrix theory metric or criteria of accuracy and False Alarm Rate (FAR) according to (Heydarian, 2019) based on the belief function and capability to combine diverse variety of evidence through its rule of combination. The combination rule of fusing captured evidence from multiple security sensors is to achieve a greater level of admissible and acceptable trust worth evidence. The information on the type and nature of attacks is also stored as a report, which can be used to collect all the suspicious packets in the ingress and egress traffic from the packet capture file. The various steps involved are as follows:

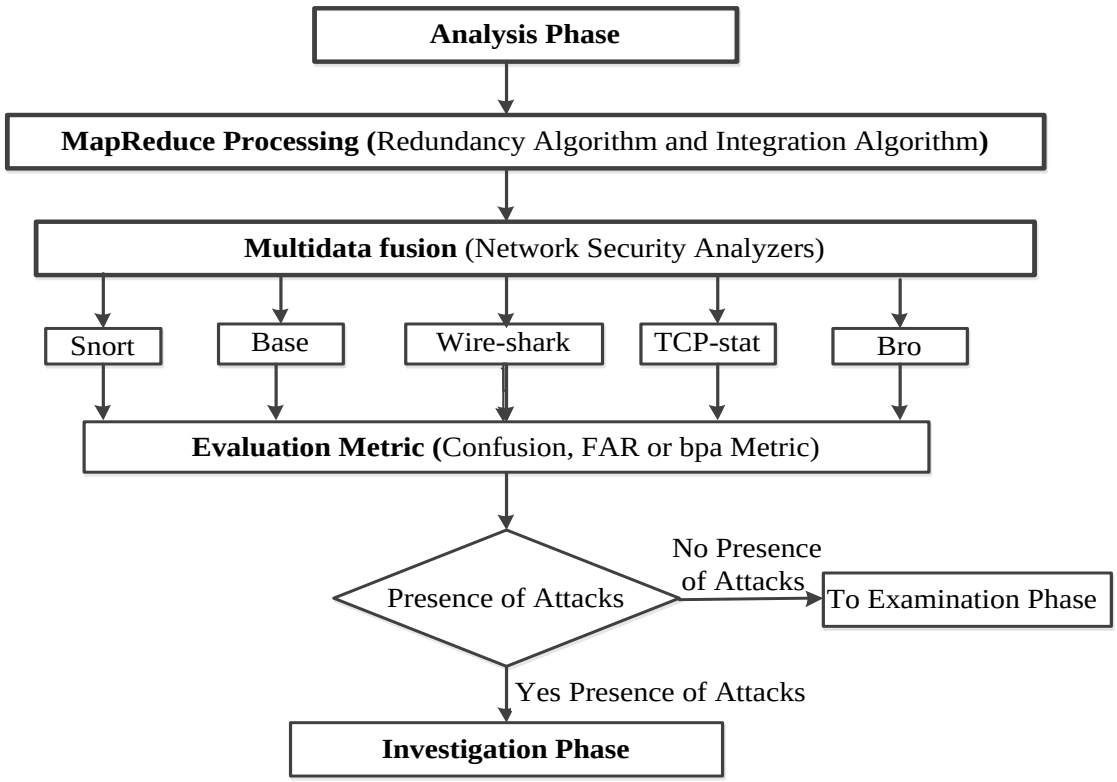
i. Various security tools are run on this file and information fusion is done using confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric to achieve comprehensive picture on the attacks.
ii. The multiple fused captured evidence are useful for identification of specific packets attacked from the unified packet format file and suspicious packets are marked as attacked.
iii. A new packet capture file is created from the attack packets, which is minimum in size and with maximum possible information as evidence.

Various security tools are run on this file and information fusion is done using confusion matrix theory metric or criteria of accuracy and False Alarm Rate (FAR) metric of evidence to ascertain the validity of the attack occurrence. The IP addresses of suspicious attackers in the fused attack information are used to collect suspicious packet records in the integrated pcap file.

4.2.6 Analysis Phase Framework Architecture for Multidata Fusion

Figure 25

Analysis Phase Framework Architecture for Multi Data Fusion



The analysis framework is built by aggregating the strengths of open source tools in accomplishing the task of collection and analysis. Network security sensors with self-contradictory and corresponding utilities are used. Security tools with similarity build the redundancy and reliability of attack information. Diversity among the tools will ensure versatility. Data fusion is performed on the alert and attack information generated by these sensors so that the decision is more accurate. In case suspicious packets are detected an IDS givens notifications inform of alerts. Packet capture and analysis tools or sniffers identify sessions or connections with anomalies in network traffic. Traffic statistics can be read from packet captures or from Netflow records taken from the

network connection through a monitored device. The following network sensors tools were used in analysis phase framework architecture for multi data as shown in Figure 25.

Fusion Snort: It is NIDS open source software with ability of examining, detecting, analysing and logging network packet traffic during transit with suspicious features and contrary to define rule set configured by the user. Snort has also ability of decoding, printing logs, alerting and capturing full packets headers evidence messages by default. One of the features of snort is fast alert mode, which has ability to write, read and analysis packet file format detailing the alert massage, timestamp, source IP, destination IP and port numbers.

Wireshark: Wireshark is open source network software with ability of capturing and analysing real time network traffic during transmission. It has a module used to output packet information longest side protocol evidence. The sensors ability is to captures the packet where the forensic investigator can be a position to read, import, and export and saved the contents. It can also filter, search contents based on specified criteria and create numerous statistics specified by the investigator. Wireshark can be integrated with other network sensors in decoding protocols contents in order to dissector other high-level protocols as well.

Tcpstat is open source software able to monitor and report by reading certain network statistics interfaces. It has ability to read previous tcpdump stored fileas well as calculate the packet traffic such as bandwidth, speed, packet average size, load of particular interface, standard deviation of packet sizeetc. It calculates statistics like bandwidth, number of packets, packets per second, average packet size, standard deviation of packet size, load of particular interface and ability to manage various transiting packets per second.

Bro is an open source UNIX based NIDS software capable of detecting and monitoring network traffic that has been manipulated and attacked by intruders based on contents and characteristic nature. It collects, filters, and analyses traffic that passes through a specific network location. Bro occur with fixed procedure code calculated for detection of most common intrusion of Internet application. These policies incorporate a signature matching facility that looks for specific traffic content. It can also analyse network protocols, connections, transactions, data amounts, and many other network characteristics.

Basic Security Analysis Engine (BASE): It is originates from ACID code and has ability of offering front-end web interface for analysing and querying incoming alerts from other network security sensors such as snort. It has high sensing ability to detect attacks that snort cannot detect in case of intrusion. It is used to supplement other security sensors in a network forensic investigation through web interface module. It allows security investigator flexibility to make decisions based on what and how much each user can access information through authentication mode. The convert the contents of the pcap file into a database and reconverted the attack packet records in the database to a new pcap file using the Net::Pcap module of the Perl language. The file contents of lib-pcap captures packet information containing the protocol types used such as TCP, IP, UDP, ICMP and Ethernet alongside encapsulation details. These protocol features are extracted recursively from Lib-pcap file and inserted in to database table.

Another lib extension file is Net: Pcap that can used to encode and extract protocol features. The main protocol attributes captured by Net. Cap file extension and extract include protocol ver, tos, hlen, id, offset, ttl, cksum, src_ip, dest_ip, tos, proto_type and options. These attributes are encapsulated, copied and stored into specific table in forensic database.

These packets attribute creating header evidence of network protocols intruded by an intruder in compromising the user network end systems. A new packet capture file is created from the attack packets minimum in size and with maximum possible information as evidence. The pcap file with only attack packets is very much reduced in size when compared to the integrated file.

Datasets in packet capture format contain the payload with confidential information. Each anonymous datasets are secured for the resolution of integrity, privacy and confidential purposes. We created an attack dataset using 3 hosts corei7 on a LAN in our research lab. Two systems had Ubuntu v26.2 and one system had Windows 10 operating systems integrated with services pack version 4. The three systems captured the packets by recording the packets using tcpdump network security sensors. Normal internet browsing and downloading of various files was carried on these three systems.

The network topology initializes DDoS and portscan attacks respectively. To identify these two types of attacks network sensors were setup to capture there form of attacks. Nmap 5.00-2 and hping 3.a2. DS2-4 were used for port scanning and flooding and for distributed denial of service numerous C program executable launched. Brief description of screenshot of the tools and sample code for some of attacks is shown in Figure 26 below:-

Figure 26

Specific Sensor Tool alongside Portscan Sample Code Attack

- SYN Scan:

```
nmap -sS -sU -sV -T4 -O -A -v -PE -PP -PS21, 22, 23, 25,
80, 113, 31339 -PA80, 113, 443, 10042 -PO --script all
192.168.100.111
```

- ACK Scan:

```
nmap -sA -sS -sU -sV -T4 -O -A -v -PE -PP -PS 21, 22, 23,
25, 80, 113, 31339 -PA80, 113, 443, 10042 -PO --script
all  192.168.100.111
```

- FIN Scan:

```
nmap -sF -sS -sU -sV -T4 -O -A -v -PE -PP -PS21, 22, 23,
25, 80, 113, 31339 -PA80, 113, 443, 10042 -PO --script
all  192.168.100.111
```

- TCP Connect Scan:

```
nmap -sP -sT -PE -PA21, 23, 80, 3389 192.168.100.111
```

- Xmas Scan:

```
nmap -sS -sU -sV -sX -T4 -O -A -v -PE -PP -PS21, 22, 23,
25, 80, 113, 31339 -PA80, 113, 443, 10042 -PO --script
all 192.168.100.111
```

- Null Scan:

```
nmap -sN -sS -sU -sV -T4 -O -A -v -PE -PP -PS21, 22, 23,
25, 80, 113, 31339 -PA80, 113, 443, 10042 -PO --script
all 192.168.100.111
```

Hping 3 (Network Ping Tool): Hping3 is capable of directing modify TCP, ICMP or UDP network packets and output the echo just in similar methodping tool uses to replies ICMP replies. It takes care of arbitrary and fragmentations packet actual size body that files transfer uses by any specific protocols.Hping3is capable of identifying trace route, port scanning, auditing TCP/IP under different protocols and fingerprinting system operating systems remotely. Intruders also used this tool to promote and initial network attack as well as Xmas Scam.

Attack Codes: Various executable programs were collected from the Internet to launch attacks on various protocols in the TCP/IP suite. The Distributed Denial of Service attacks are launched with executable files of attacks like beer, jolt, bonk, boink, newtear,

nestea, teardrop, syndrop, fraggle, smurf2. The executables attack codes in C program of a usable example files for DDoS are shown in Table 7.

Table 7

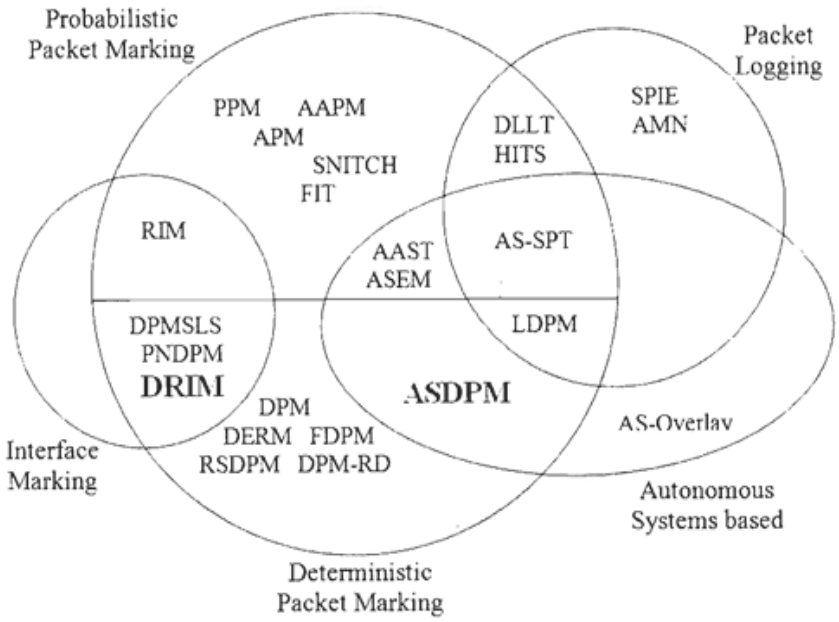
Usage of the Executable Files of Attacks

Command	Parameters
./beer	< dstaddr > <number>
./jolt	<dstaddr> <srcaddr> [number]
./bonk	<dstaddr> <srcaddr> [number]
./boink	< srcaddr > < dstaddr > <startport> <stopport> [number]
./newtear	<srcaddr> <dstaddr> [-s srcport] [-t dstport] [-n number]
./nestea	<srcaddr> <dstaddr> [-s srcport] [-t dstport] [-n number]
./teardrop	<srcaddr> <dstaddr> [-s srcport] [-t dst port] [-n number]
./syndrop	<srcaddr> <dstaddr> [-s srcport] [-t dstport] [-n number]
./fraggle	< dstaddr > <spoofaddr file> <num> <packet delay> [dstport] [srcport]
./smurf2	< dstaddr > <spoofaddr file> <num> <packet delay> <packet size>

The recommend source track back and attribution approaches are deterministic router interface marking (DRIM) and autonomous system based Deterministic Packet Marking (ASDPM) as shown in figure 27below, since they cover both Interface marking as well as autonomous system based. These two are proposed since DRIM is able to mark interface packet from the first interfaces of the local router, and ASDPM is able to mark packet from first interface of the autonomous system based router.

Figure 27

Proposed Traceback and Attribution Techniques in Relation to other Existing Techniques



They use the following three main values:

i. The source autonomous system associated with a specific number

ii. The first ingress edge router address number

iii. A packet is associated with specific router interface

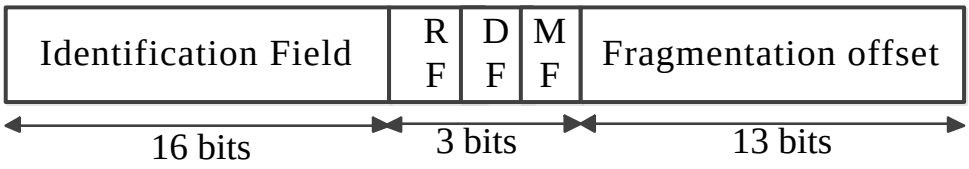
The basic idea in both of our approaches is to record the access point as close as possible to the intruder or attacker. The information is obtained from the source where the intruder cannot manipulate. The marking of information was done for each packet deterministically by the first ingress edge router. No other router modifies this marked information. To traceback an attack a single attacked packet is enough to detect the source of attack as it carries the mark attribution manipulated by an intruder traced by network security sensor.

IP header marking encoding mechanism stores information about the packet. In total there are 16 bits for ID field, where fragment offset use 13 bits and fragment flag filed

use 3 bits. The figure in 28 below illustrates the mapping between IP header fields and marking fields. The length of IPv4 protocol is 32 bits long where it holds fragmentation information. Identification field can be used for fragmentation together with flags field as well as offset field for fragments resembling as longest as the ID number is identical in both all fields in order to eliminate conflict. In the internet application fragmentation is very intermittent traffic where only 0.25% is used, so all 32 bits are used for marking purposes. The least 13 bits of offset are for storing IP address that has been hashed purposely the first internal router traversed the packet encounters as AAST. The flag is the most significant bit used for marking purposes that has taken place and also identification after marking has taken place.

Figure 28

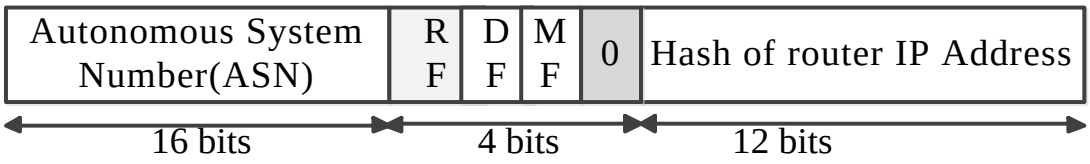
Marking Encoding fields in the IP Header



The 16 hashed bits are stored as DERM in router which is essential to reduce has collisions due to minimum false positives bits in router interface due to DERM. The reversed flag is set to 1 and 1D field stores both 16 bits together with AS number. In order to reduce the false positives, we use the 16-bits ASN for marking rather than the 32-bit IP address of the AS boundary router. The 16- bits ID field can be represent uniquely as ASN field as shown in Figure 29 below.

Figure 29

Marking Encoding Fields Overloaded for Marking



The ASDPM approach uses a two level traceback mechanism. The first level mechanism marks every individual packet deterministically by hashing first internal IP address of the router within the AS. The second level traceback mechanism marks every individual packet using AS Number (ASN) of the AS boundary router (ASBR) when it is leaving the source AS. The internet hierarchy is made up autonomous components regulated by single or numerous network operators. These network operator(s) implements a consistent, clearly and restricted stated set external routing policy.

Core routers are connected to network directly if they belong to the same domain or subnet. AS edge routers interchange routing table information with routers of other Autonomous Systems networks. ASBRs advertise AS external routes throughout the AS and every router in a given AS knows the path to ASBR. The first entry router marks each packet individual deterministically hashed value with 12 bits from 32 bits IP address. The AS boundary router marks the packet with its 16-bit AS number when it is leaving for the next AS. The specified mark on the packet cannot be overwritten once marked by any of the two levels marking mechanism.

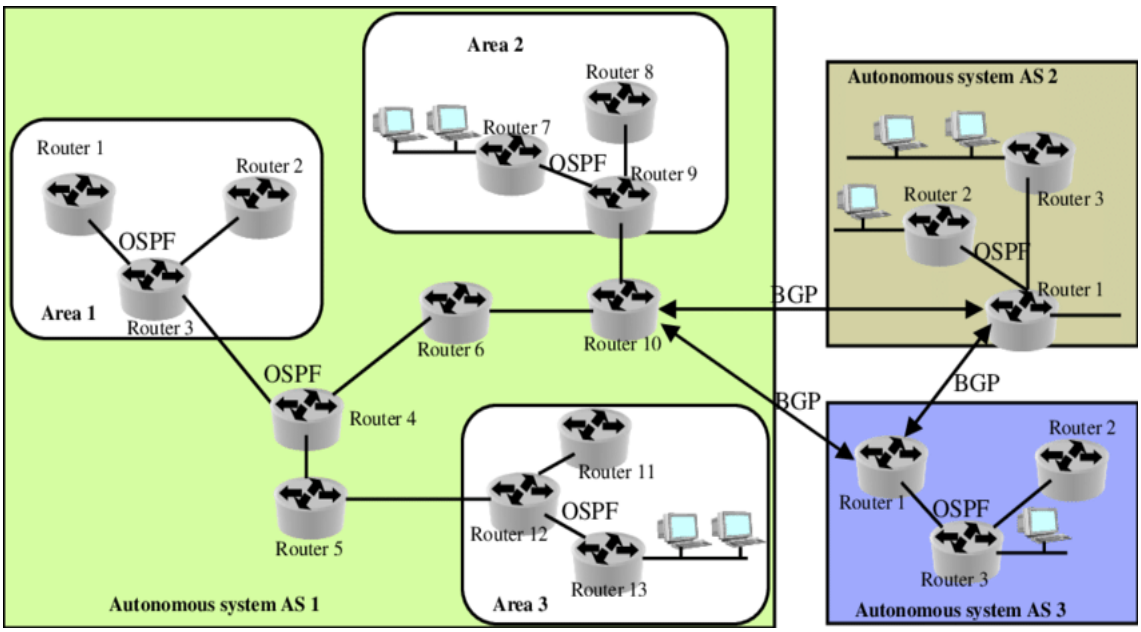
Incoming packets are not usually marked, and any outgoing packets are usually marked as per the policy in autonomous systems. Every packet information contains about a router and autonomous system which is essential enough to identify source in case there is intrusion that has taken place, auditing, traceback and attribution.

The figure 30shows the network topology consisting of numbers of internally connected routers, AS boundary routers and several ASes. An intruder launch an attack using a host in AS1 zone connected through router R1 for internet access. The packets manipulated by an intruder to launch attack are deterministically marked by R1 and no other router does the packet marking to avoid conflict of overwriting the marking.

The packets pass within the AS and reach the AS boundary router ASBR1. Whenever a packet transit one of the AS, may be AS2, for the second time the packet is marked by first router within AS, for this case router R1. If it is travelling to another area within the AS, it is not marked. Thus, any packet is marked twice, once by the first internal router and once by the AS boundary router (ASBR1) when it is leaving the AS. The network topology shows AS based deterministic packet marking approach technique in figure 30 below and traceback and attribution path in subsequent figure 1.1 in appendix III.

Figure 30

AS based Deterministic Packet Marking Approach Technique



First level in the proposed two-level marking involves marking at the first internal router, which the packet traverses. The internal routers examine the most significant bit of the offset field to check if previous routers have marked the packet and then forward it. Whenever the packet is not marked, hash value of 12 bits of the 32-bits IP address are copied into the 12 least significant bits of the offset field before forwarding. The following algorithm in figure 31below shows how it handles the marking mechanism at the first internal router (R1).

Figure 31

The First Internal Router R1Marking Scheme Algorithm

```
for each outbound packet P do
        if P.offset[0] ='0' then
          P.offset[0]? '1';
          write HashIP12(Ri) into P.offset[1.....12];
    end
    forward (P);
end
```

The second level marking is at the AS boundary router (ASBR) from which the packet is leaving to another AS. The AS boundary routers examine the reserved flag field to check if the packet has been marked by previous ASBRs and then forward it. If the packet has not been marked and if the packet is traversing into another AS, then the 16-bit AS number of the ASBR is copied into the 16-bit identification field before forwarding. The algorithm below handles the marking mechanism at the AS boundary router as shown in Figure 32 shown below.

Figure 32

Marking algorithm at the AS Boundary Router

```
for each outbound packet P do
if P.flag [0] = '0' and P has the destination address in another AS, ASBRj then
      P.flag[0] ←'1';
      write ASN (ASBRi) into P.idenfication;
   end
   forward (P);
end
```

Every individual packet has some form unique information that can be used in identification of the connected intruder to AS zone and the first internal router making traceability scheme very simple methodology. IP header contains ASN of 16 bits identification field with clear source of the packet and original AS zone encountered by the intruder. The other part of IP header contains hash function value of 12 bits in the offset filed with clear IP address of first internal router encountered by the intruder. Algorithm below explains the traceback operation extracting the ASN and IP address of internal routers at the victim side as shown in Figure 33.

Figure 33

ASN and Internal Router IP address Traceback Marking Algorithm

```
foreach attack packet P reaching victim V do
        read HashIP12 (Ri) from P.offset[1 ……12];
        extract IP from HashIP12 (Ri);
        read ASN(ASBRi ) from P.identification;
        return (IP, ASN(ASBRi ));
end
```

The DRIM methodology works by marking and hashing deterministically every packet that passes the first inbound edge router IP address as well as the number of the interface that the packet uses to connect the same router. The markings is done by the inbound router using define marking algorithm and the marking is not overwritten by any other

router within same topology at any given time. Outbound packets are the only ones, which are marked as per the policy implemented in the router to avoid marking of inbound packets. The architecture of the model is shown in figure 34 as well figure 1.2 in appendix III.

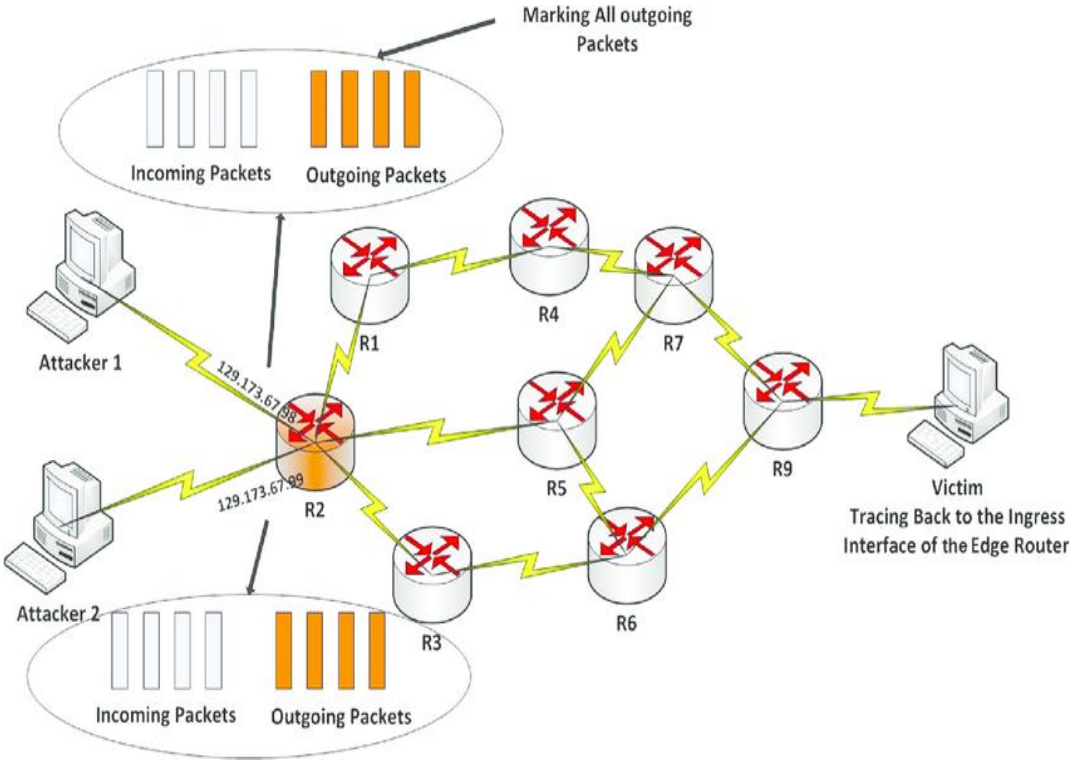
The IP subnets router connects more than two interfaces logically that represent number of other IP subnets to form global network. Thus, it has at least one physical interface for packets to pass and forward to other subnets. Before the router forward, a packet it has to identify using the routing table algorithm the shortest path among many possible paths. It is upon the router to make decisions on which IP address using unique interface to forward the required packet to required final destination. The packet is delivered locally and not considered for forwarding if the destination is an IP multicast address since it belongs to associated group of membership unique interface. This membership interface is identified by the router since the packet is pass to targeted destination network multicast system. This identity is associated with a number referred as Interface Number (IN). The router through routing table policies has ability to differentiate between the external and internal interface connected to it. The various interfaces may connect a router to a host, a LAN switch or another router. The marking algorithm ensures that only packets reaching the router from the internal set of local interfaces local will be marked.

The figure 34illustrates the network topology architecture consisting clients, VLANs switches, router and configured respective enabled interfaces. R2 in the topology is the main edge router interface that the intruder target to launch the port scan or DDOS form of attack. The main interface 12 allows the attacked packet to pass through within the topology and reach the victim host. Switch S1 connects client, R2, R3 and additional interfaces 1, 3, 4 and5. R1 through enabled interface marks every packet passing

through it's deterministically by hashing since it configured to implement the policy of marking. The other routers are not enabled and configured to mark the packet in order to avoid conflicting and overwriting the marking done by R1. This assist in traceback and attributing the attacked packet path. The packets passing the other router interfaces are already marked by R1 as it is the main edge router within the topology. R6, R7, and R9 receives the packets already marked as attacked and no further marking is allowed to taken place in these routers even if R9 connects victim host. The packet that passing through R2 is marked once using two router IP address hash value and router interface number. A marked attacked packet is sufficient to traceback and attribute the source of attack through information contain in the router R1 interface as it act as the entry point to the network topology.

Figure 34

Deterministic Router and Interface Marking (DRIM)



The marking technique used in IP header uses ID field of 16 bits used for storing information for marking packet information. The 3 bits stores information containing flag fragment field and 13 bits are used for storage of fragment offset. The 32 bits IP header figure 35 shows the specific fields representation of ID field, flag field and fragmentation offset field.

Figure 35

Marking Encoding Fields in the IP Header

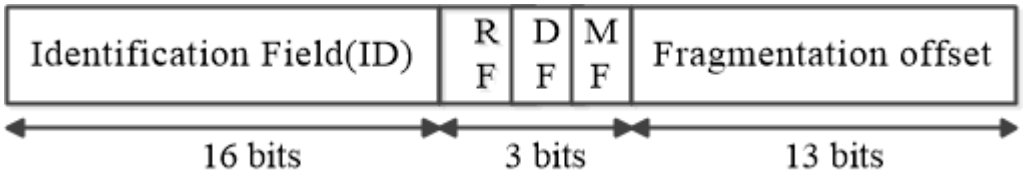
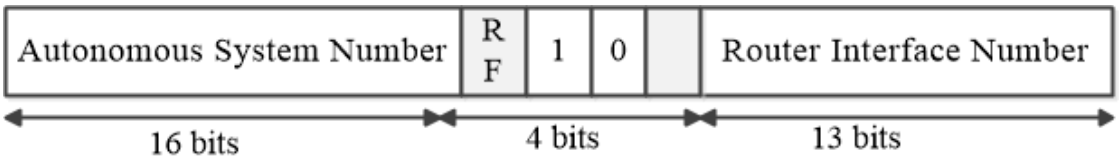


Figure 36

Marking Encoding Fields Overloaded for Marking



The main aim for overloading these 32 bits in figure36 illustrates marking information of IP header using Autonomous System based Deterministic Packet Marking (ASDPM) as discussed in subsequent section. The first field consists of 16 hash bits of 32 IP addresses that stores information containing identification field of inbound edge router. DRIM used the same marking scheme but included only the hash of router's address. The hash function used for converting the 32-bit IP address into a 16-bit value may result in some collisions and hence yield some false positives. In this approach, it stores the interface number in the least significant 12 bits of the offset field. The maximum number of interface that a router can connect at any given instants are 4096.Hence, any router can

remember each interface with a unique number using a maximum of 12 bits. The DF bit is set to I and MF bit to 0 to indicate packet marking and disable fragmentation.

The proposed marking mechanism enables the first ingress edge router to place two marks in each of the packet traversing it. The 16-bit hash of its 32-bit IP address and the number ID, associated with the interface through which the packet has reached are marked. The router is able to identifies all the information about the interfaces connected to its via routing table and policies configured on its consisting of packets passing through these interfaces, local, connected to local components like hosts and switches which are internal to the network. All packets passing through the router are copies to the routing table before they are forward to respectively destination without any alteration. The algorithm below illustrates the marking mechanism as shown in Figure 37.

Figure 37

Marking Algorithm at the First Ingress Edge Router

foreach *outbound packet P reaching router* R_i *through interface* $I \varepsilon I_{local}$ ***do***
    write Hash $IP_{16}(R_I)$ *into P. Identification;*
    write I_j *into P. offset (1.....12);*
    P.DF? 1 ; P.MF ? 0;
end

The traceback and attribution technique proposed relies on the first inbound router and particular interface information contained in every individual packet so long as the packet is allow to pass and also the packet the router drop depend on the policy setup. The IP address of the router is specified within the identification field of IP header with 16 bits hash value and the IP address of the router interface is specified within the offset field with 12 bits hash value. The interface IP address that the intruder uses to enter the network with information contained within the attack packet is very vital as it is one-step that can be used to move closer in identification of the attacker from the edge router and

stops the attack packet. The marking traceback and attribution algorithm shown below in Figure 38.

Figure 38

Traceback and Attribution Marking Algorithm at the first Inbound Router

```
foreach attack packet P reaching victim V do
        read Hash IP16(Ri) from P. identification;
        extract IP from Hash16 (Ri);
        read IN from P. offset (1…….12);
        return (IP, IN)
end
```

Combination of both ASDPM and DRIM enables move closer source of attack. These two techniques assist in identifying the interface of internal edge router from the source AS attacker as well as information contain in the packet respectively. These techniques assist in traceback and attributing source of network attack which in essential in network forensic when managing security incidents.

4.3 Proposed Network Forensic Framework for Managing Security Incidents

The proposed network forensic framework for managing security incidents as shown in figure 39 below, addresses the challenges inherent in examination, analysis and investigation phases. The proposed examination phase implements identification and correlation of network events.

The examination phase framework for the identification and correlation of network events captures network packets from suspicious attacked network hosts and transfer the attached packets to analysis forensic database server. The details evidence captured from libcap.cap libraries contains packet header, captured number of packets, and the length of captured packet. The main libcap.cap library file used to extract packet header attributes snaplen details is perl language module Net::Pcap, which enables to capture and correlated all packet length size stored.

The details information contained in the libcap.cap file for numerous packet header such as TCP, IP, UDP and ICMP protocols were extracted from the file and stored in the forensic network database Net.Pcap server respectively. The libcap.cap enables encoding of captured packets and extracts numerous network protocols attributes that specify each packet field name. The packets encoded and extract corresponding to ver, TTL, flags, ID, offset bit, cksum, proto, srcip, destip, and options attributes. The specific associated with TCP protocol packet header attributes captured and correlated by Net.Pcap file includes src_port, dest_port, seq_num, , ack_num, flags, hlen, flags reserved, urg, winsize and options. The specific associated with UDP protocol packet header attached attributes captured and correlated by Net.Pcap file includes src _mac, dets_mac, ID, tos, flags, srcip, destip, TTL and options.

All networks protocols attributes are captured and correlated, then each specific protocols attributes are transfer to unique created database table. Each packet is recorded with unique time stamp associated with frame number generated automatically. The database tables are useful during performance of statistical packet analysis and calculation of numerous attacks thresholds. These statistical analysis and calculation subjected to theory metric to test the validity of confidence admissible evidence. In this study, the metric used to test the validity of confidence of open source security sensors was confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metrics applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors. A single attack captured and correlated packet cannot be used for determine the form of attack, but sufficient number of packets based on other security sensor is sufficient to conclude the formalise type of attack based on confidence of admissible evidence. The networks open sources security sensors creates a record of

suspicious protocol attributes addresses and Net.Pcap module in Libcap.cap file transfer the specific attributes to forensic database server for analysis purposes.

The analysis phase implements multi-sensor data fusion on forensic evidence implements integration of identification and correlation of data set captured from examination phase alerts attack information. Packets captured are integrated from multiple sources across the network and redundant packets are dropped using proposed algorithm in figure 41. The analysis framework is built by aggregating the strengths of open source tools in accomplishing the task of collection and analysis. The network security forensic sensors implement the functions of complementing and contradicting evidence, which may arise due with weakness of other tools. Security tools with similarity build the redundancy, reliability and diversity among the tools to ensure versatility of attack information. Data fusion is performed on the alert and attack information generated by these sensors so that the decision is more accurate. The suspicious packets are monitored and examined by advanced persistent intrusion a system that gives alerts whenever these packets are encountered. Packet capture and analysis tools or sniffers identify sessions or connections with anomalies in network traffic. Traffic statistics can be read from packet captures or from Netflow records taken from the network connection through a monitored device.

The identified and correlated attack packet is subjected through various multi sensor data fusion tools listed in appendix I. These sensor analyses the packet attacks in order to increase the confidence level of evidence. The data integration and multi data fusion model validates sample data set with attack packets generated in desk check test. The accuracy of these tools is validated using confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed

scheme for analysing and tracing the attacked vectors of evidence for fusion to detect and validated the crucial decision to proceed with investigation phase.

The investigation phase implements the role of source traceback and attribution using AS based Deterministic Packet Marking (ASDPM) and Deterministic Router Interface Marking (DRIM) approaches. The ASDPM technique uses a two level traceback mechanism. The first level mechanism marks every individual packet deterministically by hashing first internal IP address of the router within the AS. The second level traceback mechanism marks every individual packet using AS Number (ASN) of the AS boundary router (ASBR) when it is leaving the source AS.

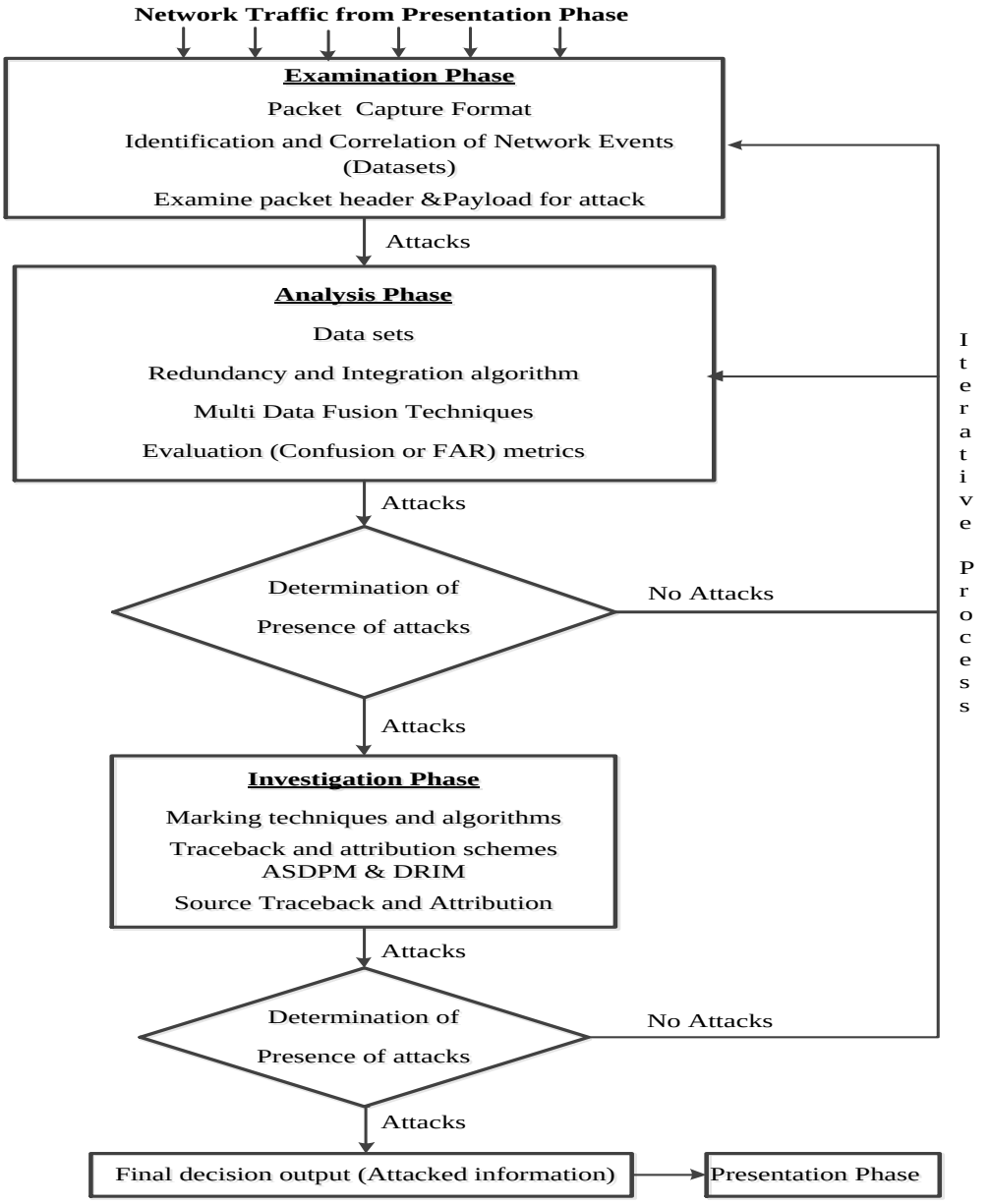
The DRIM technique implements marking and hashing deterministically every packet that passes the first inbound edge router IP address as well as the number of the interface that the packet uses to connect the same router. The markings is done by the inbound router using define marking algorithm and the marking is not overwritten by any other router within same topology at any given time. Outbound packets are the only ones, which are marked as per the policy implemented in the router to avoid marking of inbound packets.

When these two techniques are combined they permit the requirement of network forensics, where the investigation of attacks may involve only few packets. This improves broad forensic perspectives, standards and admissible forensic scientific evidences.

4.3.1 Proposed Network Forensic Framework for Managing Security Incidents

Figure 39

Proposed Networks Forensic Framework for Managing Security Incident



4.3.2 Proposed Framework in comparison with Other Existing Network Frameworks

Table 8

Proposed Framework in comparisons with other Existing Network Frameworks

Proposed Framework (2024)	Mei (2024)	Abirami (2023)	Abdullah (2022)	Anita (2021)	Al-Dhaqm (2020)	Alharbi (2019)
Identification & Correlation Examination	Examination	Examination	Crime Scene Investigation	Examination	Examination	Examination
Multi-sensor Data Fusion Analysis	Analysis	Hypothesis	-	Hypothesis	Analysis	Analysis
Traceback & Attribution, Investigation	Investigation	-	-	-	-	-

The proposed network forensic framework (2023) enhanced security techniques for identification and correlation examination in examination phase, multi sensor data fusion analysis in analysis phase and traceback and attribution investigation. These security techniques are lacking in the other existing network frameworks which make forensic framework for managing security incidents acquire inadmissible evidence. The proposed security techniques in the examination, analysis and investigation phase enhance the forensic investigation evidence admissibility.

4.4 Performance Evaluation and Discussion of Results

4.4.1 Evaluation of Examination Phase based on derived metrics and computer simulation

The intruder objective is to attack specific victim in a network setup using particular type of protocol to manipulate packet field and gain access to victim system and information compromising the confidently, privacy and integrity without the knowledge of the owner.The analysis attacked information evidence of various forms of intrusion which were identified, examined and correlate which specific packet parameters. The two types of intrusions that were analysed at the network and transport layers are port scan and Distributed Denial of Services (DDoS) respectively. The identified intrusion and correlated protocol attributes fields examined are as shown in table 9 and table 10 in subsequent sections.

Table 9

Intrusion and Protocol Attributes Correlation of DDoS Attacks

Intrusion	Protocol attributes fields examined
Teardrop	Overlapping Fragment Offsets
Jolt	Protocol = ICMP, Large Fragment Offsets
Ping of Death	Length of all fragments of a packet > 65535
SYN Flood	SYN Flag in source and ACK Flag in destination addresses
Fraggle	UDP echo on 7, 13,17,19
Smurf	Type = 0 without sending type = 8
Boink	Manipulated Fragment offset Field in IP Packets
NewTear	Manipulated Fragment offset field in IP Packets

Table 10

Attack and Protocol Feature Correlation of Port Scan Attacks

Attack	Protocol fields to be examined
Connect Scan	Enormous amount of unsuccessful connects and successive port number requests
SYN Scan	SYN Flag and no corresponding ACK
FIN Scan	FIN Flag and Sequence number
ACK Scan	ACK Flag and RST as replies
Null Scan	No Flags set
XMas Scan	URG, PUSH and FIN Flags set
UDP Scan	UDP requests and ICMP port unreachable messages
Ping Sweep	Type = 8 and code = 0

Full packet capture is performed on the web server using Wireshark in application layer to evaluate cross-site scripting (CSS) Attacks details. The identification HTTP attributes and correlation of the payload were performed with the attack vectors. The packet capture file is opened in Wireshark and all the packets who comprise of the server's IP address as source or destination were selected. Attacker login and cookie information can be read by examining the payload of TCP ACK segments being sent from the server to the client. Attacker placing the malicious script for stealing victim's cookies was seen in the payload.

We applied the correntropy recommend by (Yunfei, 2021) for multivariate forensic network data, as provided in equation (4), we calculate it for both normal and suspicious attacked network vectors observations as

$$I_{1N} = \begin{bmatrix} f_{11} f_{12} \\ f_{21} \quad f_{22} f_{ij} \end{bmatrix},$$

$$Y_{1N} = \begin{bmatrix} C_1 \\ C_i \end{bmatrix} \qquad (5)$$

Such that I was the observations of network data, Y was the interested class label (c) of each observation, N is the number of observations and f was the number of features. The mean of correntropy values of normal vectors (normalcorpy) was computed using equation (6) in the examination phase. In the observation phase, the correntropy value (corpytest) was estimated for each record based on equations 4 and 5.

We design a baseline between the μ (normalcorpy) and each testcorpy using the standard deviation measure(σ), which estimates the amount of variation between the mean of normal correntropy values and each correntropy of testing records. If the variation between the two values is greater than or equal (2σ), the testing vector was considered as an attack, as given in equation 7. This was because such a vector was so far from the dispersion of normal correntropy values and was difficult to fit it within the same distribution of normal data. We called this threshold a Risk Level(RL) according to (Yunfei, 2021) identifies all attack observations with low false alarm rates. The RL is scaled in a range of [0, 1] in order to exactly specify to what extent the abnormal activates deviate from normal ones.

$$\mu(corpy^{normal}) \quad = \frac{1}{N}(corpy^{normal}) \tag{6}$$

$$RL = \begin{Bmatrix} \mu(corpy^{normal}) - (corpy^{test}) \geq 2\sigma \ attack \\ else \qquad\qquad\qquad\quad normal \end{Bmatrix} \tag{7}$$

For example, Table 11 lists some flow identifiers from the UNSW-NB15 as shown in appendix IV dataset with estimated RL values.

Table 11

UNSW-NB15 Features of the Proposed Examination Framework

Category	Feature Numbers
Normal	11,34,19,20,21,37,6,10,11,36,47
Dos	6,11,15 16,36,37,39,40,42,44,45
Fuzzers	6,11,14,15,16,36,37,39,40,41,42
Backdoors	6,10,11,14,15,16,37,41,42,44,45
Exploits	10,41,42,6,37,46,11,19,36,5,45
Analysis	6,10,11,12,13,14,15,16,34,35,37
Generic	6,9,10,11,12,13,15,16,17,18,20
Reconnaissance	10,14,37,41,42,43,44,9,16,17,28
Shellcode	6,9,10,12,13,14,15,16,17,18,23
Worms	41,37,9,11,10,46,23,17,14,5,13
Common	6,9,10,11,12,13,14,15,16.17,36,37,41,42,44,45

The researcher computed the final results depending on the highest repeated features with at least three times. The feature vectors and flow network identifiers such as source IP (srcip), source port (sport), destination IP (dstip), destination source (dsport) and protocol types (proto) were selected using the simple random sampling technique in order to remove repeated instances or missing values, improving the overall performance of network forensic examination phase model for managing network security incidents features service (Saputra, 2022), sbytes, and sttl are from the Basic Feature category. Feature smean is from the Content Features category and feature ct_dst_sport_ltm is from Additional Generated Features category.

The origins of attack instances can be easily tracked via correlating their flow identifiers with their estimated RL. This way will help to define the risk level of those instances. If the RL value equals one, this means that type of attacks constitutes the highest risk to an organization as it sends many flows to a specific destination such as events of DDoS

attacks. But. If the RL value equals zero, this indicates this type of attacks makes the lowest risk to that organization as shown in Table 12.

Table 12

Selected vectors with Risk Level (RL)

srcip	sport	dstip	dsport	Protocol	Label	RL
192.168.168.1	179	239.255.255.250	33159	TCP	0	0.23
192.168.1.6	15982	156.67.212.136	5060	TCP	0	0.11
175.45.176.3	63888	149.171.126.14	179	TCP	0	0.25
175.45.176.2	7434	149.171.126.16	80	TCP	1	0.83
175.45.176.0	15558	149.171.126.13	179	TCP	1	0.72

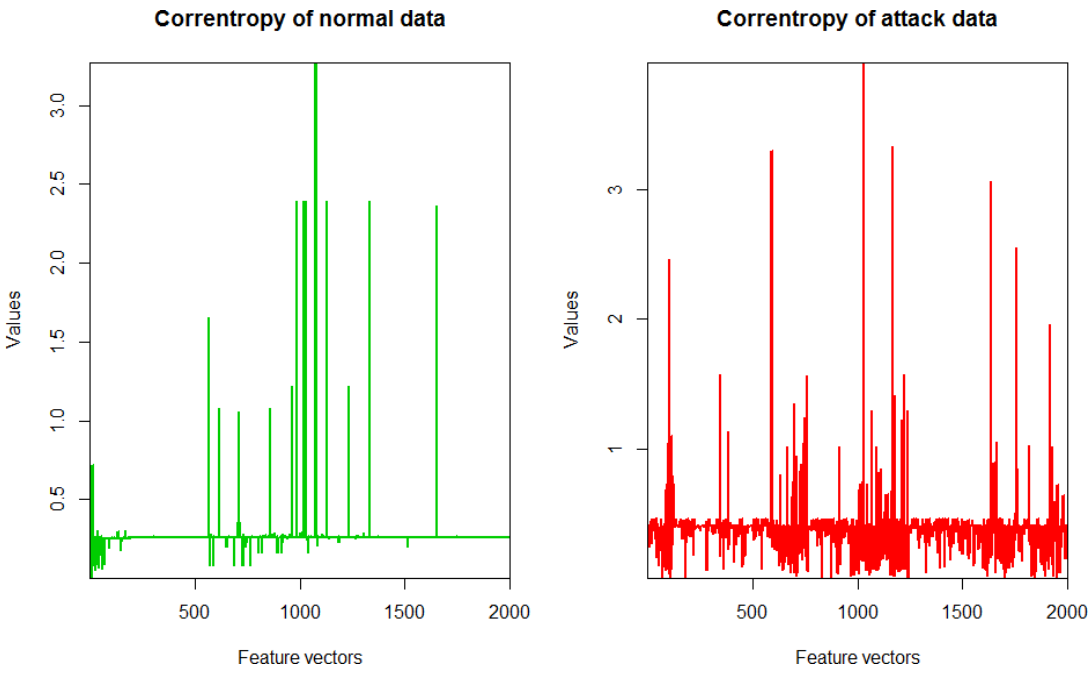
The observation was the abnormal records have higher RL (more than 0.5) than normal activities (less than 0.5). Ultimately, the proposed network forensic framework for managing network security incidents examination phase defines attack activities and their risk level, helping network administrators to track and report bad events that try to penetrate their network events. As established from table 11 from the example of select five vectors from the USNW-NB15 dataset was designed to demonstrate how the risk level was computed based on these levels were connected with their network traffic flow identifiers for examining the evidence of attack events.

The simple random sampling and chi-square techniques confirm selection of the significant observations and features. It replicates the patterns of appropriate and suspicious occurrences while running the proposed network examination phase forensic model for managing network security incidents as from the 2000 correntropy features USNW-NB15 dataset as shown in Figure 40.

4.4.2 Correntropy of Some Normal and Attack Samples

Figure 40

Correntropy of some normal and attack samples



The correntropy illustration graph evidently demonstrate authentic difference among normal data features and attack feature vectors as shown in figure 40 as it approximate the relationships between these nonlinear vectors. The normal samples of 2000 correntropy features flow data values clearly demonstrates difference from attack ones as illustrated in figure 40. The correntropy of normal data contains even clear graphical representation as opposed to correntropy of attacks data which contain distorted randomly graphical representation. The correntropy sample between normal and attack data assist the forensic investigator to proceed in examining the attack data. As a result, the various form of network security incidents were substantially examined and explored the five flow captured identifiers which includes source IP address, source port destination IP address, destination port and type of version protocol associated with particular risk level as shown in Table 12.

4.4.3 Evaluation of Analysis Phase based on based on Derived Metrics and Computer Simulations

The two algorithms were developed to handle the redundancy and timestamp issues. The algorithm in figure 41 in subsequent section decides which files are to be integrated directly and which files are to be checked for redundancy issues before integration. The developed algorithm identified related packets field of all transmitted network traffic examined after being integrated from the timestamp selected within the dataset range. The algorithm enables and identifies the specific dataset similarity by selecting the packet header and payload information. One of the similar packets is included and the redundant packets are ignored. The algorithm does not drop duplicate packets within the same packet capture file and ensures a fair treatment to crafty packets created by the attacker with similar payload.

Algorithm implements the rules for forming groups of hosts from which the files must be integrated, only after removing redundant packets. The packets from hosts at different router level need not be in the same group. Level represents all hosts who are in same subnet or domain. The packets collected by agents residing in different subnet or under a different NAT address need not be in the same group. Packet logs existing in a particular group are only integrated after removing the redundancy.

Algorithm Handling Packet Redundancy

Figure 41

Algorithm Handling Packet Redundancy

```
program PacketLogging (Output)
begin
   for i:= 1 to k do    # For all levels, 'k' is the last
      if agent(host logging packet)is at level = i then
           collect packet logs only for all level i+1 routers
           by filtering out their IP/mask into a group
           # Only packet logs in level 'i' will be in a group
      end if
   end for
end.
```

The packet capture files are grouped according to the above rules and the program for integration is executed on files in the same group. Packet captures in one particular group will need a check for redundancy before integration.

All the' m' pcap files are made into n blocks each according to a fixed timestamp range. The blocks of packets with similar specific range timestamp are considered for examination and each block of packets are stored before presenting to analysis phase in forensic database server. The packets from the first are copied into a 'temp' database. The packets from second to the m^{th} pcap are compared with packets in 'temp' and inserted if they do not already exist. Once all the pcap files are compared, the 'temp' database in moved to a 'final' database. The algorithm in figure 42 below demonstrated how the implementation process takes place.

Though all the packets in each timestamp range are checked for redundant packets, there may be a possibility to miss out packets lying in the border ranges. The process of leaving similar of packets might lead to losses of information but this value very negligible compared to the strength obtained because of data reduction.

Algorithm Handling Packet Integration

Figure 42

Algorithm for Managing File Integration After Checking Redundancy Results

```
program Integration (Output)
   temp: temporary database
   final: final database
   var  m: int; # number of pcaps to be integrated
        n: int; # number of timestamp ranges in each file
        TotalNoPackets [1..m]: int; # Total number of
                  # packets in each of the m pcap files
begin
   for i = 0 to n-1 do
        INSERT all packets of the first pcap into temp
        for j = 1 to m-1 do
             for k = 0 to TotalNoPacket[j] do
                  INSERT pcap[k] IF NOT EXISTS in temp
             end for
        end for
        SELECT all packets from temp and INSERT into final
        EMPTY temp
   end for
```

An attack dataset was created using three hosts in a LAN in our research laboratory. Two systems had Ubuntu v 20.10 and one system had Windows 10 (SP4) as operating systems. Three systems recorded packet captured using tcpdump. Normal internet browsing and downloading of various files was carried on these three systems. The two systems installed with different operating systems initiated port scan and DDoS intrusions. Packet captured collected from the three hosts were 183.0 MB, 328.2 MB and 200.5 MB in sizes respectively and integrated into single file using the integration algorithm. Simply combining the files resulted in a file size of 711.7 MB. Using algorithms described in section 4.4.2.3, avoiding the redundant information the combined file resultant to 622.7 MB as shown in Table 13 below.

Table 13

Reduction by Data Integration

Description	File 1	File 2	File 3	Combined File	Integrated File
Initial Packet Size(MB)	181	328.2	200.5	711.7	622.7
Number of Packet	405956	574945	551051	1531952	651357

File size reduction in (MB) % = Combined file size (MB)- Integrated file size (MB)

Combined file size (MB)

= 711.7-622.7x100% =12.5 %

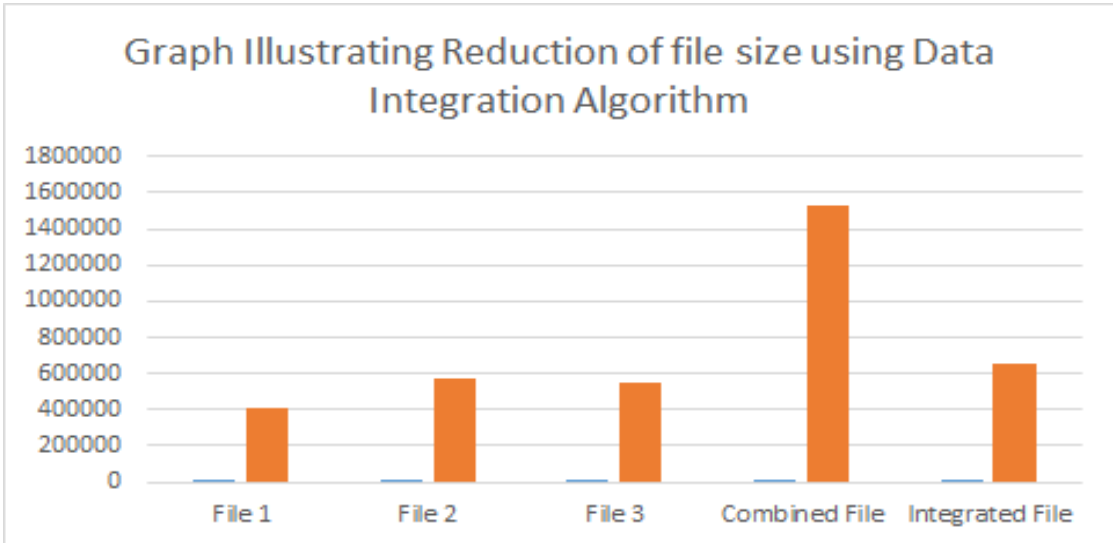
711.7

% reduction in number of packets =15311952-651357 x100% = 95.7%

15311952

Figure 43

Graph Illustrating Reduction of File Size in MB



The results shown in Table13 and graph in Figure 43 demonstrates a reduction of 12.5 %, which is not very significant. The reduction of 95.7% in packet capture file size is

less but the number of redundant or duplicate packets dropped is considerable. This is due to many attack packets being captured by all files but the effective size being very small (i. e. TCP and UDP Flood packets). The reduction saves the system storage by the same percentage. The reduction of packets dropped make the layer 3 network switches more effective and also exclude traffic captured in promiscuous mode hence the redundant or duplicate packets are dropped considerable. The reduction of packets may be very small or worthless the effort in many cases. It is advantageous when multiple packets during analysis phase, evaluated and when there are constraints in time and forensic investigators personnel.

4.4.4 Multi Sensor Data Fusion Results

The tools used are Snort, Wireshark, TCPstat, Bro and BASE. We use the dataset created and use the above tools on the packet capture files collected by tcpdump while the attacks were in progress. Screenshots of various alerts as indicated by the security sensors for port scans & DDoS attacks are given below. Figures 44 and 45 show the alerts as indicated by BASE of port scan traffic and DDoS attacks respectively.

Figure 44

Port Scan traffic

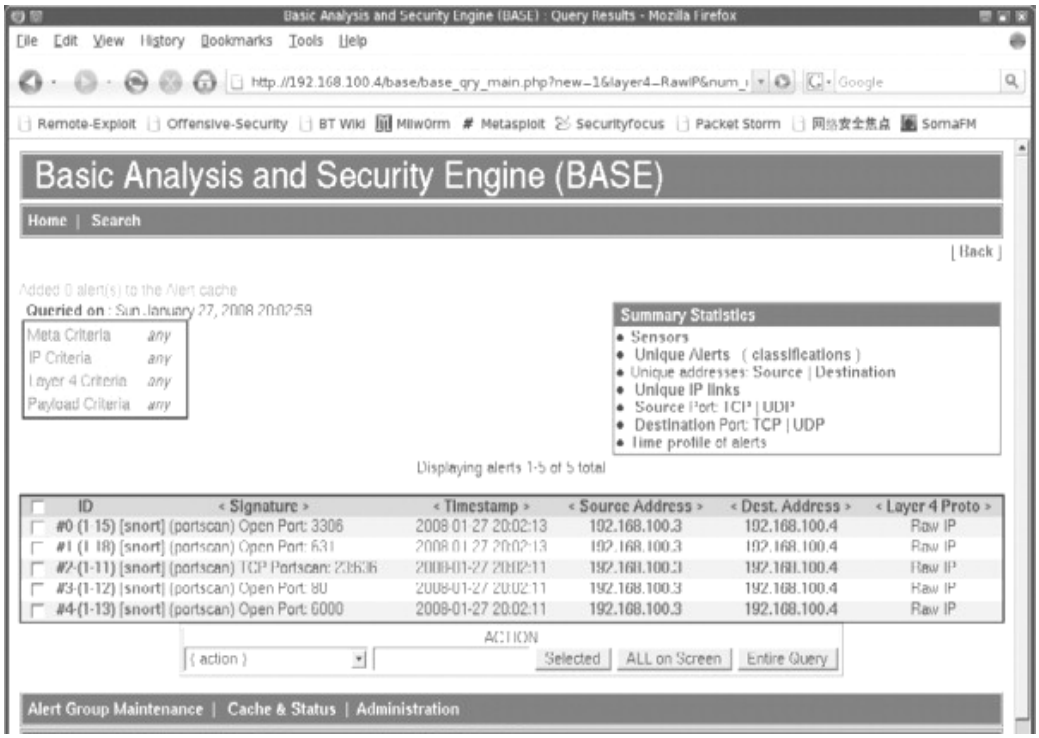
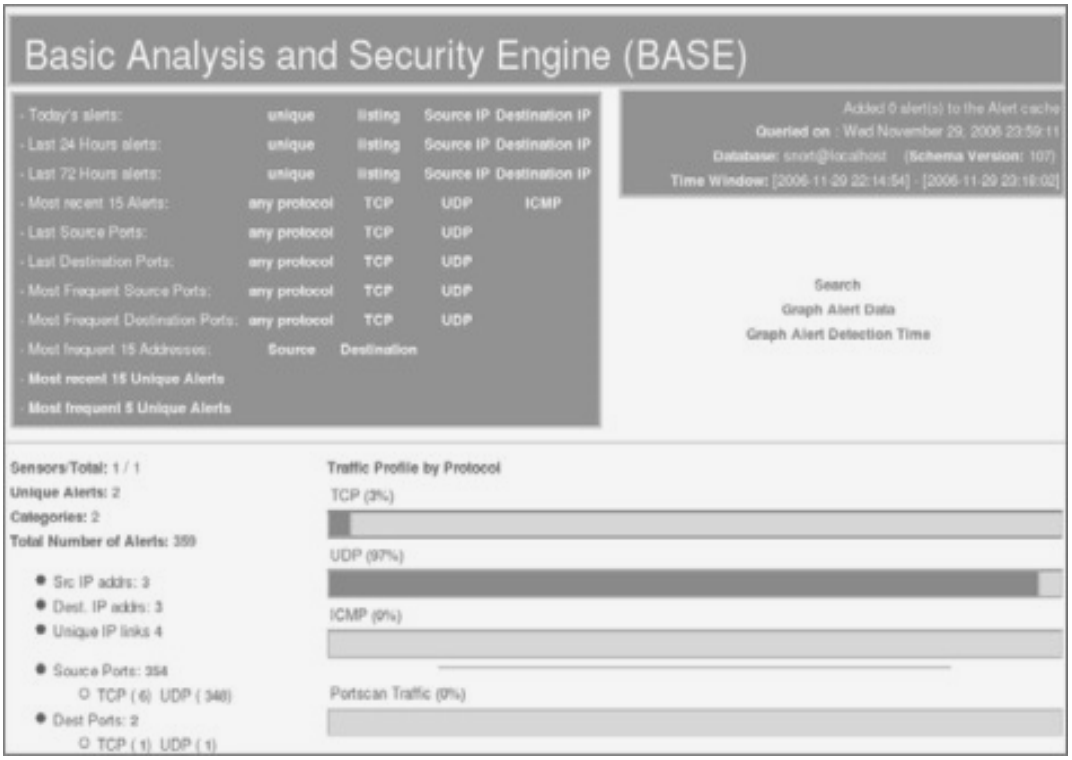


Figure 45

DDoS Traffic



Figures 46 and 47 shows the information provided by Wireshark analysing packet captures for Port Scan (TCP Connect) and DDoS Attack (NewTear) as shown IN Figure 46.

Figure 46

Port Scan (TCP Connect)

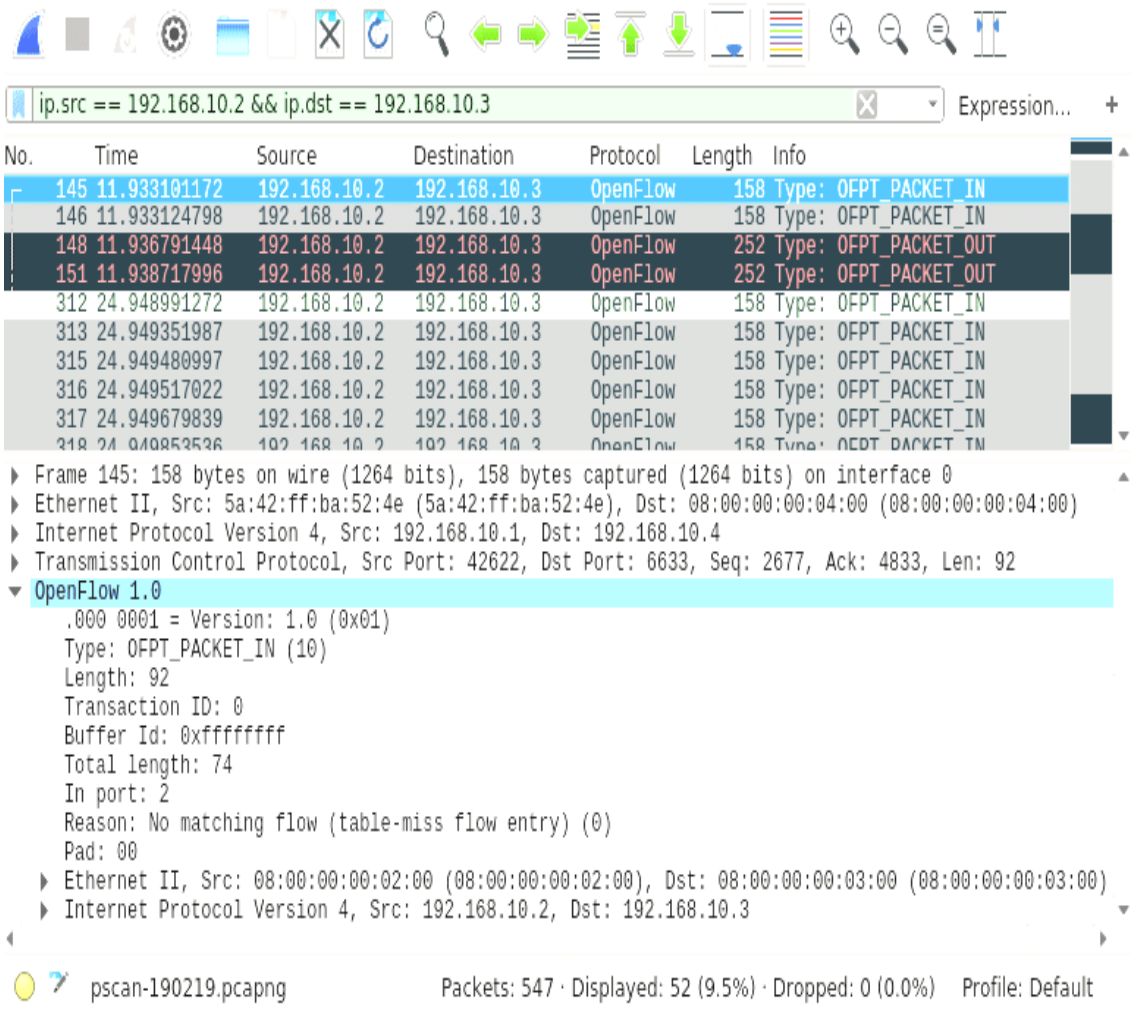
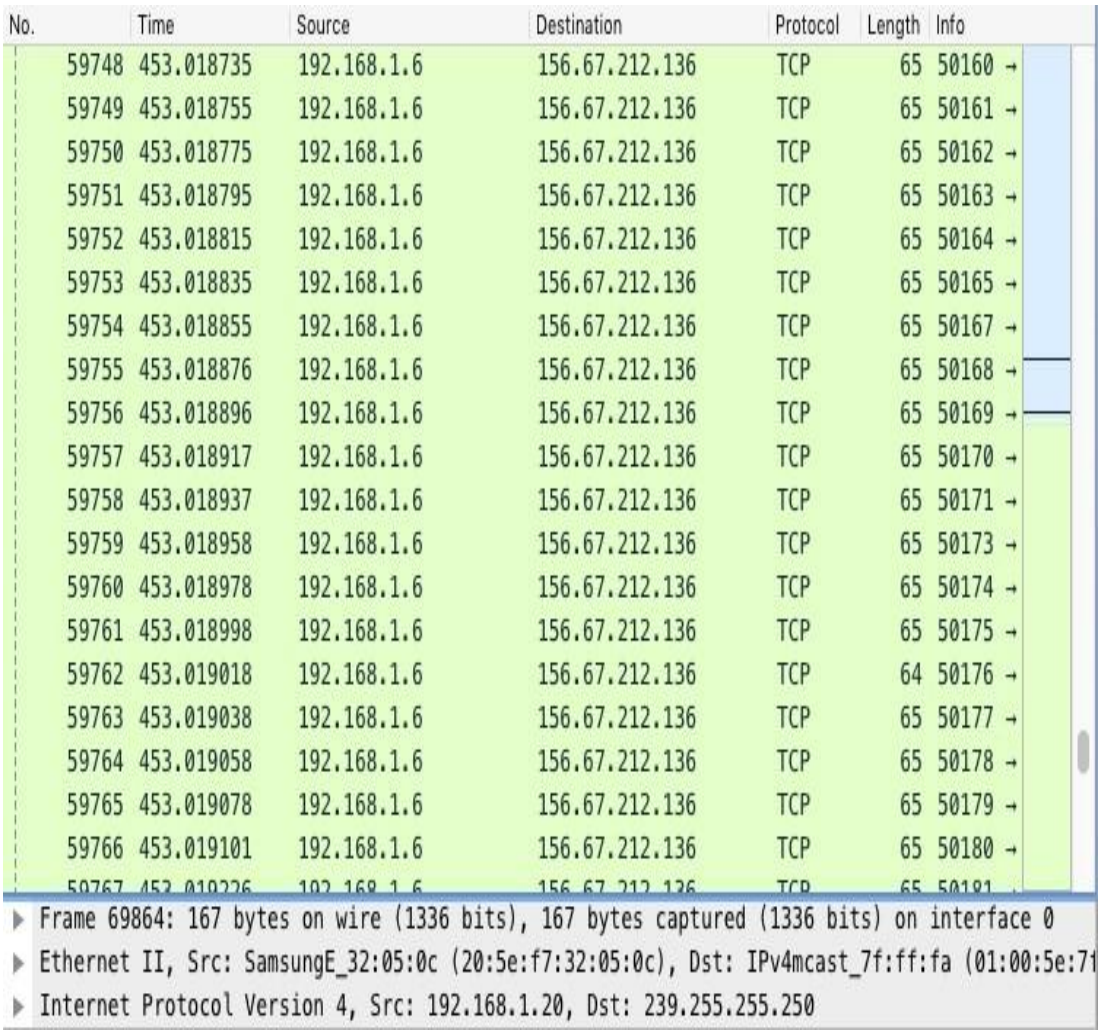


Figure 47

DDoS Attack (NewTear)



No.	Time	Source	Destination	Protocol	Length	Info
59748	453.018735	192.168.1.6	156.67.212.136	TCP	65	50160 →
59749	453.018755	192.168.1.6	156.67.212.136	TCP	65	50161 →
59750	453.018775	192.168.1.6	156.67.212.136	TCP	65	50162 →
59751	453.018795	192.168.1.6	156.67.212.136	TCP	65	50163 →
59752	453.018815	192.168.1.6	156.67.212.136	TCP	65	50164 →
59753	453.018835	192.168.1.6	156.67.212.136	TCP	65	50165 →
59754	453.018855	192.168.1.6	156.67.212.136	TCP	65	50167 →
59755	453.018876	192.168.1.6	156.67.212.136	TCP	65	50168 →
59756	453.018896	192.168.1.6	156.67.212.136	TCP	65	50169 →
59757	453.018917	192.168.1.6	156.67.212.136	TCP	65	50170 →
59758	453.018937	192.168.1.6	156.67.212.136	TCP	65	50171 →
59759	453.018958	192.168.1.6	156.67.212.136	TCP	65	50173 →
59760	453.018978	192.168.1.6	156.67.212.136	TCP	65	50174 →
59761	453.018998	192.168.1.6	156.67.212.136	TCP	65	50175 →
59762	453.019018	192.168.1.6	156.67.212.136	TCP	64	50176 →
59763	453.019038	192.168.1.6	156.67.212.136	TCP	65	50177 →
59764	453.019058	192.168.1.6	156.67.212.136	TCP	65	50178 →
59765	453.019078	192.168.1.6	156.67.212.136	TCP	65	50179 →
59766	453.019101	192.168.1.6	156.67.212.136	TCP	65	50180 →
59767	453.019226	192.168.1.6	156.67.212.136	TCP	65	50181 →

Snort alerts for some of the port scan and DDoS attacks were as follows:-

Figure 48

Port Scan and DDoS Attacks alert Detected by Snorts Sensor

- *UDP Scan:*

```
04/21-14:38:34.420257  [**] [1:100000160:2] COMMUNITY SIP
TCP/IP message flooding directed to SIP proxy [**]
[Classification: Attempted Denial of Service] [Priority:
2] {UDP} 192.168.111.214:34251 -> 192.168.111.209: 18994
Run time for packet processing was 0.41368 seconds
```

- *Xmas Scan:*

```
04/21-14:20:15.287561 [**] [1:100000160:2] COMMUNITY SIP
TCP/IP message flooding directed to SIP proxy [**]
[Classification: Attempted Denial of Service] [Priority:
2] {TCP} 192.168.111.209:1087 -> 192.168.111.214: 59592
04/21-14:20:28.443260  [**]  [122:1:0]  (portscan)  TCP
Portscan [**] [Priority: 3] {PROTO:255} 192.168.111.214 -
> 192.168.111.209
04/21-14:20:28.518435  [**]  [116:59:1]  (snort_decoder):
Tcp Window Scale Option found with length > 14 [**]
[Priority:  3]  {TCP}  192.168.111.214:  59864  ->
192.168.111.209:1
Run time for packet processing was 0.10479 seconds
```

- *Jolt Attack:*

```
04/21-15:27:01.213385  [**]  [123:3:1]  (spp_frag3)  Short
fragment, possible DoS attempt [**] [Priority: 3] [20]
192.168.111.220 -> 192.168.111.209
```

- *Smurf Attack:*

```
04/21-16:01:12.311008 [**] [1:100000160:2] COMMUNITY SIP
TCP/IP message flooding directed to SIP proxy [**]
      [Classification: Attempted Denial of Service] [Priority:
      2] {ICMP} 192.168.111.209 -> 192.168.111.120
      Run time for packet processing was 0.96950 seconds
```

- *SynDrop Attack:*

```
      04/21-15:58:02.944992 [**] [123:5:1] (spp_frag3) Zero-
      byte fragment packet [**] [Priority: 3] {TCP}
      192.168.111.220 -> 192.168.111.209
      04/21-15:58:02.945564 [**] [123:4:1] (spp_frag3) Fragment
      packet ends after defragmented packet [**] [Priority: 3]
      {TCP} 192.168.111.220 -> 192.168.111.209
```

4.4.5 Discussion of Multi Sensor Data Fusion Results

The screen shot of some of tools detected attacks alerts and the alert logs were as shown in figure47. The confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors of evidence for identification of the suspicious addresses detected by the multi fusion sensors tools applied to test the admissibility and validate the attacks evidence. Some of the terms used in confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors and applied in calculations of the combination rule are as follows:

i. *The frame of discernment (FOD) 0*comprises of entire hypotheses for which sources of information canoffer the evidence. The setof evidence is finite and comprises of mutually exclusive propositions that span the hypotheses space.

ii. *A basic probability assignment (bpa)* over a FOD 0 is defined as a mass function m, which assigns beliefs in a hypothesis and defines a mapping of the power set to the interval between 0 and 1. The sum of all bpa is equal to 1 as shown in equation (1).

The *bpa* are assigned based on the information collected from the sensors. The bpa values are decided based on information specific to the environment in which the attack data was collected, type of attacks, correlation between attack and response generated by a particular tool and similar factors. Basic probability assignments given for various attack alerts generated by Bro sensor are shown in table 14. Similarly, bpa values were assigned based on the alert information collected from the Snort, Tcpstat, Base and Wireshark sensor tools as shown in table 15 below.

Basic probability assignments given for various attack alerts generated by Bro are shown in Table 14.

Table 14

Statistics Information Generated by BroSensor

Attack	Alert Information	bpa
UDP Scan	1303376290.650025 weird: bad_UDP_checksum	0.06
Xmas Scan	1303375815.316225 weird: spontaneous_FIN 1303375825.534776 weird: bad_TCP_checksum 1303375826.286681 weird: baroque_SYN	0.27
Jolt	weird: 1303379824.209294 excessively_large_fragment weird: 1303379824.209294 fragment_overlap	0.25
Smurf	1303381865.190161 weird: bad_UDP_checksum	0.16
SynDrop	weird: 1303381689.530709 excessively_small_fragment weird: 1303381689.530721 fragment_inconsistency weird: 1303381689.530721 fragment_size_inconsistency 1303381689.530721 weird: bad_TCP_header_len	0.26

Bpa's given for various statistic information generated by snort, tcpstat, Base and Wireshark are shown in Table 15.

Table 15

Statistics Information Generated by Snort, Tcpstat, Base and Wireshark

Attack	Alert Information	bpa
UDP Scan	Time: 1303376039 No of pkts:119 IPV4:109 TCP:0 UDP=104 ICMP:5	0.10
Xmas Scan	Time: 1303375812 No of Packets:754 IPV4:746 TCP:746 UDP=0 ICMP:0	0.19
Jolt	1303379823 No of pkts:6366 IPV4:6360 TCP:0 UDP=3 ICMP:6357	0.28
Smurf	Time: 1303381872 No of pkts:10001 IPV4:9990 TCP:0 UDP=0 ICMP:9990	0.23
SynDrop	Time: 1303381684 No of pkts:11250 IPV4:11237 TCP:11237 UDP=0 ICMP:0	0.20

The probability of believing that network attack has been carried out and occurred is based on the degree of alert credible evidence used to launch attack. The evidence belongs to same set *A* of attack information but they are considered to be special subset of ***A*** information. The measure plausibility degree of alert information to ascertain the evidence belongs to set *A* of attack information or to any of its subsets or to any set that overlaps with *A*. Based on several combination of network sensors used in multidata fusion to analyse the evidences information captured to calculate the belief functions measure using the confusion matrix according to (Heydarian, 2020)multi-label confusion matrix rule of combination as given in equation (i).

Equation (i)

$$\mathrm{M(A)} = \frac{\sum B \cap C = AM_1(B)M_2(C)}{1 - \sum B \cap C = \emptyset M_1(B)M_2(C)}$$

The numerator represents the accumulated evidence for the attack information sets B and C, which ascertain the attack evidence hypothesis A and the denominator sum measures the quantity of conflict between the B and C sets. Equation (i) can as well represented as shown in equation (ii).

Equation(ii).

$$\mathrm{M(A)} = \frac{product\ of\ bpa's\ of\ the\ attacks\ given\ by\ all\ sensors}{summation\ of\ the\ product\ of\ bpa's\ of\ the\ all\ individual\ attacks\ given\ by\ all\ sensors}$$

Or

$$FAR = \frac{FP + FN}{TP + TN + FP + FN}$$

Where TP represents true positive which denotes a number of the correctly attack classified, TN represents true negative expressing a number of the correctly normal classified, FP represents false positive the number of the misclassified attacks and FN

represents false negative the number of the misclassified normal records according to (Heydarian, 2020).

We include the bpa's or FAR for the five attacks and using the five tools. The values are shown in Table 16 as given below.

Table 16

Combined Sensors Statistics information generated by snort, Bro, tcpstat, Wireshark and Base

	Snort	Bro	tcpstat	Wireshark	BASE	Product
UDP Scan	0.09	0.06	0.10	0.12	0.15	0.0000097200
Xmas Scan	0.24	0.27	0.19	0.21	0.22	0.0005688144
Jolt	0.27	0.25	0.28	0.26	0.23	0.0011302200
Smurf	0.18	0.16	0.23	0.22	0.19	0.0002768832
SynDrop	0.22	0.26	0.20	0.19	0.21	0.0004564560

The calculations for a two attacks, UDP Scan and Jolt are shown in the table 16above. The numerators are the product of all the bpa's for the Jolt and UDP Scan respectively. The denominators are the summation of product of all bpa's of the individual sensors.

The value of m for UDP Scan was calculated as shown below:

$$\text{Bro UDP Scan (bpa or m)} = \frac{0.06}{0.06 + 0.10}$$

$$= \frac{0.06}{0.16} = 0.375$$

The value of bpa or m for (combined security sensors) UDP Scan was calculated as shown below:

$$\mathrm{m(combined\ )} = \frac{0.0000097200 + 0.06}{00000972 + 0.09 + 0.06 + 0.1 + 0.12} = 0.11538$$

$$Ratio = \frac{Bro\ scan\ (bra\ or\ m)}{m(combined\ Scan\ UDP)} = \frac{0.375}{0.11538} = 3.2$$

The values for bra or m in both sensors seem to be very low small when compared to the assigned values of bpa since five sensors tools were considered for attack evidence data fusion.

The value of m would be less than individual bpa if only individual tools were put into consideration. Nevertheless, it is very insignificant from the calculation to put into consideration from the values of m that the UDP scan attack has transpired. The probability of m value of combined attack is three times less than m value of UDP scan attack based on Bro sensors. This implies that results attained from the calculation of m's values above illustrate that combination from combining many network sensors gives a strong confidence belief that the attack has taken place as compared to the belief of using only one individual network sensor. The same procedure or steps applied for other types of network attacks. From calculation above it illustrates that it is more accurate to prove and validate that an attack has taken place when attack evidence is fused and subjected to combination of network sensors tools.

Based on evaluation criteria of accuracy or False Alarm Rate (FAR) metric applied to measure the performance of the proposed network forensic framework for managing security incidents in examination phase while identifying and tracking security incidents. The beliefs of bpa or FAR metrics are dispersed by a forensic investigator manually stepwise and the values are determined based on the years of expertise of how to monitor the network traffic. This depends on evidences generated by the specific network sensor tool ability or specific attack occurrence security incidents launch by an attacker. The type of security sensors determines the value of "m"or threshold"τ" that is calculated

using same procedure. When the value of “bra or m” is greater than the value of"τ” for a given combination of sensors used, then we can prove that indeed attack has occurred based on evidence. It the value of “m” is less than the value of"τ”for a given set of network sensors then we can prove that an attack has not occurred.

This evidence validate that attack has taken place based on this information which can be relied upon and strengthen the forensic investigation before making informed decisions if an attack to place or not. We can use same network tools in ensuring there no data versatility and in cases of redundancy. This kindly of combining number of network sensors for data fusing analysing based evidence captured determines strong evidences of proving that an attack has been launched. This kind of open source network tools can established strong belief and evidences for analysing post mortem network attacks based on captured network traffic and files. The sensors security tools are setup and configured to capture the network logs traffic or plain text files for output analysis purposes.

The logs or plain files are captured by sensors tools automatically whenever there is alert of attack traces information and validation of information proved by using combination confusion matrix metric theory criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors of evidence. The additional evidences are negligible which is taken as overhead due to time spent during data fusion and time taken when combination of security sensors are capturing the attack information or giving alerts as compared to time taken when individual security sensors are used. The results are given in Table 17.

Table 17

Reduction of File Size through combination of Security Sensors Multidata Fusion

Description	Integrated file
Integrated packets file size(MB)	622.7
Suspicious packets file size (MB)	622.7x (3.2) =19.92
% of reduction packet file size	$\frac{602.9}{622.7} x100\%$ = 96.8%

From the Table 17, it illustrates that two main essential information. The first one is the reduction of file size by sufficient size from integrated file 622.7 MB to 19.9MBwhich translates to 96.8% reduction of file size. The file reduction of file size can be valuable when analysis the evidence of attacked information and save storage system. The second point is it gives details information about the source of attacked information validating the captured evidence. The information analysed is based on identified, generated reports from sensors tools alerts of suspicious hosts and connected IP addresses. This information reveals evidence consisting of all successful sources of attacks containing network communication, suspicious inbound and outbound traffic.

We observe that the abnormal records have higher “m” (more than “r”) than normal activities (less than “r”). Ultimately, the proposed network forensic framework for managing network security incidents defines attack activities and their risk level, helping network administrators to track and report bad events that try to penetrate the network.

4.4.6 Evaluation of Investigation Phase based on Derived Metrics and Computer Simulations

We present a detailed study of evaluation against various performance metrics, in comparison with other related techniques, validation of the trace back and attribution packet marking approaches. Evaluation performance and observations between ASDPM

and DRIM are as shown in Table 18 after conducting simulation between the two proposed techniques approaches.

Table 18

Evaluation Performance and Observations between ASDPM and DRIM

ASDPM	DRIM
First internal router interface marks of every single packet using 12-bit hashed of its address and first (AS) boundary router marks each packet existing the first AS with 16 bits a unique AS number.	First ingress (AS) edge router marks interface each packet with the 16-bit hash of its address and 12-bit interface number through which it encounter the edge router.
Out of 32 bits router addresses 12 bits hash are stored.	Out of 32 bits router addresses 16 bits hash are stored.
The source AS is identified by traceback technique using single packet entering the first internal router.	The first internal router and associated interface is identified by traceback technique using the single packet that enters the router.
Internal attacks is easily identifiable if 16 bits identifier field AS number is not marked	It is very easy to identify the interface which allow the packet entry and exit even if the networks allow NAT applications. This helps when search specific packet details.
There is high chances of packet spoofing by intruders if there are aware of traceback scheme mechanism where they can set flags to high with marking information.	Packet spoofing is impossible where the first internal router identifies and enforces the entire packet marking both for all internal interfaces even when the intruders set the flags to high with marked information.

4.4.7 Evaluation performance and observations of ASDPM and DRIM with other Techniques

The comparison between ASDPM and DRIM with other existing techniques in terms of specific evaluation metrics with various traceback approaches. The evaluation and simulation were done based on set by ISP in terms of packets required for traceback after an attack, deployment effect, overhead processing, overhead bandwidth, required storage memory, prevention of attack evasion, safeguard, scale of attackers accommodation, and number of implementable network devices, capability to handle major DDoS attackers and capability to traceback altered packets. Table 19 gives a summary comparison between ASDPM technique with DPM, ASEM and ASSPT techniques respectively. Table20also gives a summary comparison between DRIM technique with RIM,DPM and DPMLS techniques respectively.

Table 19

Evaluation Comparison between ASDPM with DPM, ASEM and ASSPT Techniques

Evaluation Metric	Deterministic Packet Marking (DPM)	AS based Edge Marking (ASEM)	AS level Single Packet Traceback (ASSPT)	AS based DPM (ASDPM)
Mechanism	Packet marking is purely deterministic	AS marking is based on PPM	AS marking is based on single logged packet	AS marking is based on DPM
Packets for traceback	7 marked packets	68 marked packets	Only one marked packet	Only one marked packet
Marking field length	2 successive packets out of 32 bits	32 bits	packets marking not applicable since packets are logged in,	32 bits
Processing Overhead	Every packet marked with first or last IP address of 16 bits of the edge router. The two successive packets is used to transfer the total IP address.	The AS inbound edge router updates and calculates the AS path information. The first marking router does the three marks updates, calculated the path and marking)	All the queries can be done on the edge router since all the logs of packets are captured during the traceback process	The first internal router and AS edge router does the packet marking consecutive.
Storage Overhead	The inbound address can be used to determine the victim by matching with stored in routing table.	The hashed IP address can be calculated and determine in advance and stored from edged router	AS edge routers requires enough storage space for logging purposes.	The 12 bits hash of AS router address is calculated, determined and stored in advance.
Overhead Traceback	In bound table can be used to identify source address	Hashed IP existing from ID field can be used to calculated source address	The attacked packet can be identified in the outbound router after querying ASes upstream from AS recursive traceback server	Hashed value existing from the offset field and the ASN in ID field can be used to calculated source address

Changes In Infrastructure	The network devices can add one additional network function	The network devices can add two extra network functions	AS boundary router implements logging function mechanism	AS boundary and internal routers each adds one additional network function
Changes In Infrastructure	The network devices can add one additional network function	The network devices can add two extra network functions	AS boundary router implements logging function mechanism	AS boundary and internal routers each adds one additional network function
Scalability	Numerous simultaneous intruders can be traced	Numerous intruders can be traced and handled	Small scale number of intruders from the packet logs due to small storage capacity.	It can handle any number of intruders
Involvement of ISP	Less charges leading to limited packet marking	The calculation of PATH information deployment of AS level is a requirement leading to limited packet marking	Deployment information of AS level is a requirement for logging and querying leading to limited packet marking	Less charges leading to limited packet marking
False Positives	Minimum false positives	Minimum positives are efficiently suppressed	Storage of packet logs lead to false positives due to active blooms filters	Few false positives from the address of hashing routers'

Table 20

Evaluation Summary Comparison between DRIM with RIM, DPM and DPMLS Techniques

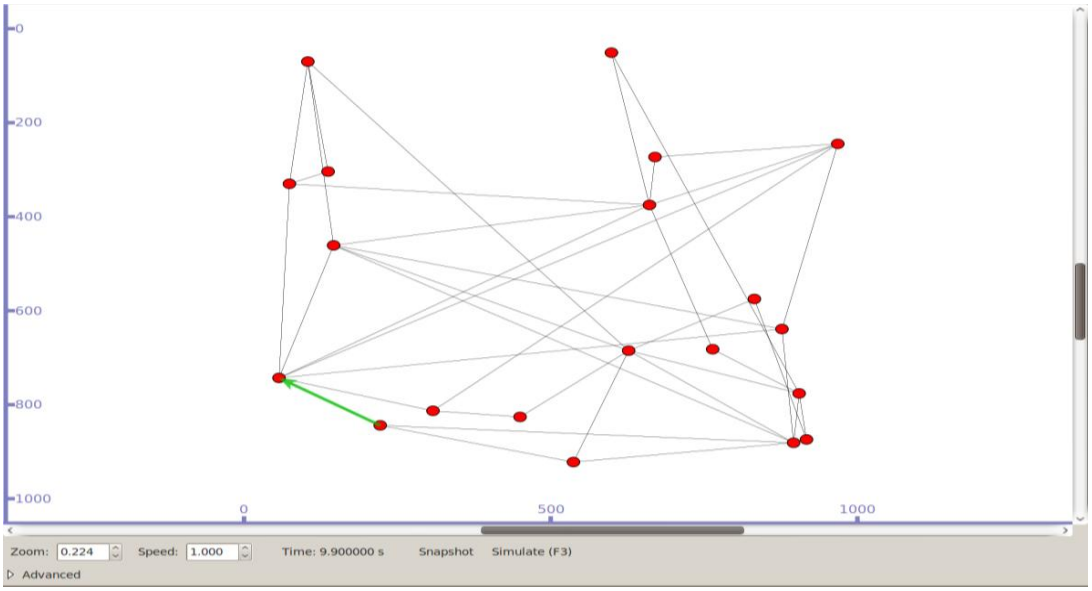
Evaluation Metric	Router Interface Marking (RIM)	Deterministic Packet Marking (DPM)	DPM with Link Signatures (DPMLS)	Deterministic Router and Interface Marking (DRIM)
Mechanism	PPM based on the interface link	Based on purely deterministic marking	DPM based on link signatures	DPM based on Interface link
Traceback packets	Attack path reconstructed from single packet	Attack path reconstructed from seven packet	Attack path reconstructed from single packet	Attack path reconstructed from single packet
Marking field length	Uses seventeen bits	Uses two bits from thirty two consecutive packets	Uses sixteen bits	Uses thirty one bits
Processing overhead	Probabilistic marked packets with IID and XOR values. The XOR field values are updated instantly	Every packet is marked with edge router address using the first or last bits from 16 bits.	Router marks every packet entering or exiting the network	Only the first ingress edge router placed two marks for every packet.
Storage overhead	The hop counts, interface ID and XOR value are stored and maintained in trace table database	The victim addresses are marched together with the source address from trace table of ingress router	All adjacent links signatures are stores and maintained by the every individual router	The router address is calculated and stores in advance 16 bits hashed address ‘

ASDPM Simulation

The brite Ns-4simulator was used to simulate and generate simple topology comprising of two hierarchy level routers and Autonomous system. The topology comprises of five routers and four ASes in each Autonomous system. The planes length set to 10 for both specified Low Set (LS) and High Set (HS) specified number of five routers, four ASes nodes and links per node set to default of value of two. The topology was linked and jointed together with incremental growth type of node facility. The interlinking of router within the topology was done using the random edge connections method. The bandwidth distance between the intra and inter was set to the constant default value. The topology was generated and simulated then imported from Ns-4format file to tcl file format. The network animator (nam) snapshot of the Ns-4is shown in figure 49as generated by the segment code illustrated in appendix III Figure 13.

Figure 49

NAM Output of Topology Generate by NS-4Brite Simulator



The two agents were created using Brite generator and placed in Ns-4application file format. One of the agents was UDP which main responsibility was connected to node attached to main intruder and generates the packets. The other UDP Sink agents' main

responsibility was connected to node attached to main victim node and receives the generated packets from the intruder side. The two algorithms in figure 32 and figure 33 were used for marking the packets at the AS boundary router and internal router respectively. The classifier is able to receive the packet accordingly depend on received code function of every router. The file extension of the Ns-4node.cc was modified in order to distinguish between the node of inter router and node of ASes router.

Every packet are marked by the first router using the node ID number and the other routers within the topology confirm the marking to ensure the packet just traverse without marking it again. The packet existing the topology through the ASes router are marked the same way as the first internal router and the other AS boundary routers does not marked the packet again as well. The victim node receives the attacked packets and the algorithm 37 main function is to extract the attacked packets information value details. ASDPM performs the traceback and capable of giving all packet information as per simulation Ns-4generator even with just single packet.

DRIM Simulation

The Ns-4topology in figure 43 was generated and simulated using Ns-4manually. The two new agents were created the same way as discussed in the previous section. The code of algorithm 6.4 was configured and applied in the recv function of the classifier where every individual packet received or leaving the manually generated Ns-4topologyis marked. The packets received by the first router are marked with a unique node ID number and proceeding node ID.

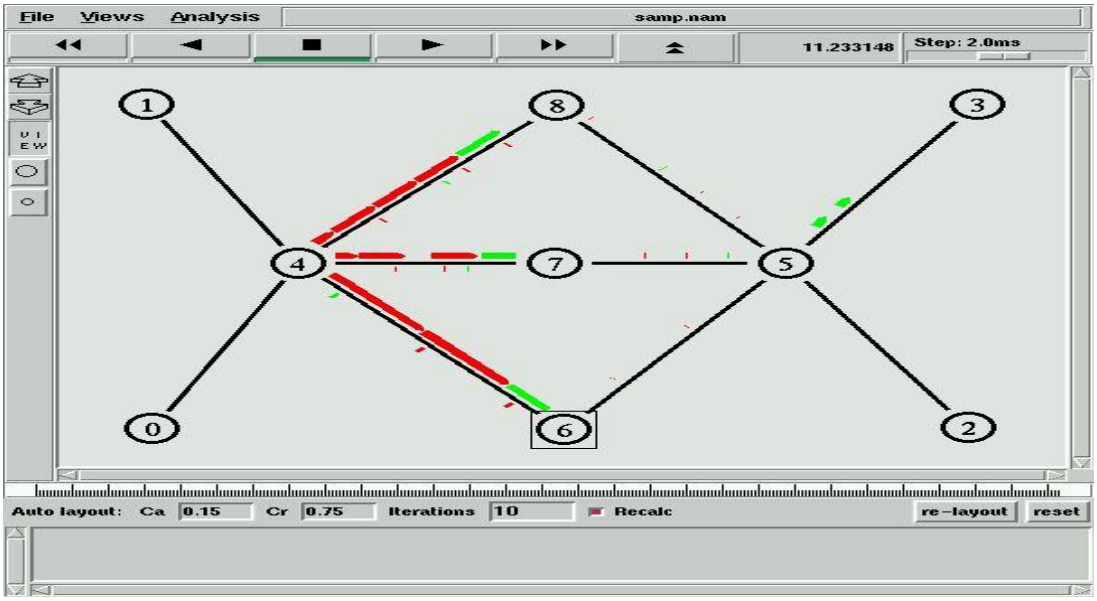
The proceeding node ID identifies arrival of packet direction where the router interface does not specifies the packet direction directly. The first router does the marking and the packet traverse the other router without marking the packet gain but just confirm the marked placed on the packet. The value of the packet received by the victims was

extracted using redundancy algorithm figure 41 after applying the source code shown in the appendix III Figure 14.

The simulation shows that DRIM performs the traceback effectively for each packet. The network animator (nam) screenshot of the Ns-4is shown in figure 50created by segment code in appendix III figure 1.4.The diagram illustrates the network topology attack path where node 4 launch an attack. The attack is propagated to node 6, 7, 8. Node 7 further propagates packets attacks towards node 5then towards node 3. The attack path is traceback and attributed to main node 4 using the DRIM techniques based on proposed algorithm from node 3 as per NAM Output Topology Generated by NS-4Simulator below leading to source of attack.

Figure 50

NAM Output Topology Generated by NS-4Simulator



Some of the Practical Considerations

The two approaches ASDPM and DRIM when compared to other existing approaches were found more advantageous in various metrics used during evaluation process. The

two main essential practical considerations noted during ASDPM and DRIM approach evaluation with other existing approaches were as follows.

Evasion Protection Simplicity

An attacker can alter a marked packet with information by inserting spoof packet stream. In case an attacker is aware and knowledge of marking approaches, they can alter storage encoded marks packet field by inserting false packets spoofed marked packets. The ASDPM and DRIM proposed an approach detects and traceback spoofed marked packet. As discuss in the previous subsequent sections, ASDPM approach, every packet is marked by the first internal router and set the first bit to high or1 deterministically. If an attacker makes an effort or attempted to alter the marked packet or spoofed the marking then the ASDPM approach overwrites the spoofed marking with accurate mark.

Subsequently the DRIM approach, every packet reaching the first ingress router marks is marking with hashed IP address, identification of router interface alongside interface number using the encoded mark. The ingress router eliminates the spoofed packet by ensuring that the all the packet reaching the interface are uniquely marked, hashed and encoded. If an intruder attempts attack a victim using spoofed packet, then DRIM approach is capable to detect, traceback and overwrite the spoofed packet in totally. There makes the approach 100% guarantee from spoofed marked packet evasion protection simplicity by intruders.

Steady Deployment

The ASDPM and DRIM proposed approaches are prospective in terms of steady deployment where only the first internal router and ingress AS boundary router (ASBR) does the path enforcement packet marking mechanism. There no marking that takes

place by any other router within the set and active topology path connection and any encoded field marked packet information cannot be overwritten.

This makes the proposed approaches 100% guarantees where all packets are marked and encoded once by first internal router and AS boundary router. The approaches ease and facilitate the traceback investigation process since there is no routers within the communicating routers which do the packet marking only the two router are designated to mark the packets. Any network forensic investigator will definitely traceback and attribute an attack to legitimate user if at all the intruder cannot be traced or accessed.

CHAPTER FIVE

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

5.1 Summary of the findings

This study was set out to enhance the performance of network forensic investigation by developing a network forensic framework for managing security incidents. The proposed framework enhances the challenges inherent in examination, analysis and investigation phases. New security techniques are proposed to accomplish the philosophy of providing more efficiency and optimal network forensic framework for managing security incidents. The specific contributions are:

5.1.1 To investigate the challenges of the examination, analysis and investigation phases of network forensic frameworks in managing security incidents

Network forensics is a science of determining and retrieving evidential information in a networked environment about a criminality in such a way as to make it acceptable. This information are collected from multiple, actively processing sources and security sensors. Intrusion or misbehaviour features were categorised in order to determine the source of security intrusion using scientifically proven techniques. This leads to detecting and characterizing unauthorised network events meant to compromise system services, components, assists in incident response and recovery of affected systems. The established computer networks forensic field lays a strong foundation for network forensics as standard procedures and tools are in place for detecting, collecting, preserving and presenting evidence. However, little has been done to address the challenges in examination, analysis and investigation phases. The examination phase lacks effective mechanism to identify, correlate and validate the features of attacked packets manipulated by attackers. The analysis phase presents attacked packets and alerts, which are not admissible since no reconnaissance performed from various

security sensors, while the investigation phase has a challenge in determining the source traceback and hence unable to attribute attacker to a particular host within a network. These challenges deter the principles of evidence and the criteria for admissibility of scientific evidence that states evidence should be admissible, authentic, complete, reliable and believable.

5.1.2 To analysis the Network Forensic Security Techniques used to address the challenges of examination, analysis and investigation phases of Network Forensic Frameworks

The forensic security sensors technique that addresses the challenges of examination phase was identification and correlation of network events using network security protocols open source applications that open the file, read the file contents, extract the contents and encode from several protocols features specifying the every self-explanatory field with unique attribute name. Identifying important sessions of suspicious activity will reduce the data to be analysed. The correlation of events validates the occurrence of the malicious incident and guide in decision making to proceed with the investigation phase.

The main security sensors techniques that address the challenges of analysis phase was implementation of data fusion that alert and attack information generated by network security sensors so that the attack evidence were more accurate to ascertain the validity of the attack occurrence. The main security sensors techniques that address the challenges of investigation phase was identification of traceback and attribution technique based on packet marking, packet logging or hybrid approaches. The attack traceback and attribution technique analysing the data packets transmitted, applications being run, traffic patterns observed and protocols violated. It also traceback and attribution the source of attack to particular system.

5.1.3 To develop a Network Forensic Framework that Incorporates Network Forensic Security Techniques that addresses the challenges of examination, Analysis and Investigation Phases

This framework is built over the existing network framework considering phases specific to each individual network forensic framework phases. The proposed generic network forensic framework for managing security incidents has potential to handle both active and post intrusion. The proposed framework investigation starts from the examination phase which specifically identifies the intrusion using protocols features implemented by attackers for both active and post intrusion by capturing, copying the packet file (libcap) format and the evidence indicators correlated. The specific network attacks examine, identified and correlated includes the port scan, cross site scripting and distributed denial of service.

The analysis phase role fuses the identified and validated multi data evidence from multiple security sensors. It also classifies attacked patterns using data mining, soft computing or statistical approaches. A technique for performing data fusion of information from multiple security sensors was identified, implemented; tools with complementary and contradictory functions were also identified. Data fusion was performed on the alert and attack information generated by these network sensors so that the attack evidence eliminates the redundancy and ascertain the accuracy of captured attacked evidence. Confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied were used to measure the performance of the proposed scheme for analysing and tracing the attacked vectors evidence was used to perform fusion of the alert information and the validity of the attack occurrence was ascertained.

The investigation phase involves traceback and attribution. A practical approach was presented to identify network events at application; transport and network layer of the TCP/IP protocol stack and correlate them with attacks. The two proposed techniques includes ASDPM and DRIM capable of validating the attacked evidence from the tested dataset generated from the laboratory over a period of time.

ASDPM role implements the marking of every packet entering and existing the network with capability of hashing the IP addresses by the first internal router using the AS number (ASN) within the AS. The DRIM role implements the marking of every packet deterministically with hashing value using the IP address and interface number of the first ingress router within AS reaching router. The deterministically marked packet by the router simplifies process of traceback, the source of attacked packet. The validation of the two-approach techniques attacked evidence was performed using the ns-2 simulator.

5.1.4 To evaluate the Effectiveness of the developed Network Forensic Framework in addressing the Challenges of Examination, Analysis and Investigation Phases Based on Derived Metrics and Computer Simulations

In examination phase, identification and correlation of protocol attributes, which are manipulated by the attackers, is a continuous process. The monitoring, detection and capturing of attacked evidence network traffic files is a continuous process. The common techniques and behavioural patterns of the hackers can be examined and correlated for different types of intrusion.

The analysis phase, packets captures were collected from multiple sources and integrated to hold the entire information across the network. The integration process helps in elimination of redundant or duplicate and ascertains validity of attacked

evidence. The fusion of alert information and statistical values from various tools and application of confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors evidence for fusion increased the confidence level. The attack information detected and validated where crucial decision shall be to proceed with the investigation phase. The suspicious attacked packets identified and effects information reduction with increased acceptance validity. The proposed data integration and fusion models were validated on sample data sets with attack traffic generated in our research. The results achieved improve the expectation and reliability as compared to existing network forensic framework infrastructure. These frameworks have to be tested against large datasets comprising of real time attack data.

In the investigative phase, the main role is network forensic examination, implementation and resolving the problem of IP traceback. Two approaches based on deterministic packet marking were proposed; the main objective was accessing the point as close as possible in the source of attack as well as recording the evidence. The ASDPM is the first approach was where the first internal router traceback the attacker within the confine of AS possibility.

The DRIM is the second approach, which identifies the packets that attacker uses to launch the intrusion specifically the one which reaches the router interface. This is one-step closer in identification of source of attack traceback and attribution in network forensic. This meets the requirement of network forensics, where the investigation of attacks may involve only few packets.

5.2 Conclusion

This study proposes network forensic framework for managing security incidents. The study has reviewed related literature and identified existing challenges in network

forensic frameworks examination, analysis and investigation phases. The study propose network framework with solutions for examination, analysis and investigation phases that strengthen the network forensics and makes the investigation admissible. The study proposed identification and correlation of network traffic security techniques in examination phase. In analysis the study proposes multidata fusion security technique using combined open source security sensors. In investigation the study proposes traceback and attribution techniques in identification of source of attack. The study used interrogative, quantitative and evaluation based on Design Science Research Process (DSRP) methodology approach. The proposed technique in examination phase was identification and correlation.

The identification provided attempts made in compromising a system and assist during reconstruction of intruded information. The correlation validated the particular intrusion and guide in decision to proceed with investigation. The techniques resulted in confirmation of DDoS, Portscan and XSS attacks dataset. The proposed techniques in analysis phase was combination of multidata fusion security sensors and integration algorithm. Sensors relies alerts attacked network events evidence which was subjected to confusion matrix and FAR metrics to validate the evidence accuracy. Algorithm resulted in minimizing evidence file size from 100% to 92.96% saving the system storage by 7.04%. The proposed techniques in investigation phase were traceback and attribution techniques based on ASDPM, DIRM and marking algorithm. The techniques resulted in marking and logging of attacked packets or hybrid both towards particular source of attack and recorded accurate attached evidence based on evaluation metrics set by ISP.

5.3 Recommendations

The main recommendations from this research thesis are classified into:

i. The challenges identified with existing network frameworks for managing security network incidents associated with examination phase was identification and correlation of attacks using networks protocols, the challenges associated with analysis phase was multi-sensor data fusion of various network traffic and the challenges associated with investigation phase was traceback and attribution of network packets to ascertain specific source of intrusion.

ii. The two algorithms were developed to handle the redundancy and timestamp issues. One of the developed algorithm enabled and identify specific dataset similarity by selecting the packet header and payload information. The other developed algorithm identified related packets field of all transmitted network traffic examined and integrate from the timestamp selected within the dataset range to manageable reduce size.

iii. The main security techniques identified that addresses the challenges of examination phase was identification and correlation of network events by incorporating several open source applications network security sensors that examine, capture suspicious network events, read the contents, extract the contents and encode from several protocols features specifying self-explanatory field with unique attribute name. The capture evidence subjected to criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors evidence for fusion increased the evidence prove and confidence level.

iv. The main components identified and address the challenges of analysis phase was implementation of data fusion that had capability to rely alerts and attack network events generated by several network security sensors and output accurate evidence that are valid to ascertain attack occurrence. The fusion of alert information and statistical values from various tools and application of confusion matrix theory or criteria of accuracy and confusion or False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors evidence for fusion increased the evidence prove and confidence level

v. The main components identified and addresses the challenges of investigation phase was identification of ASDPM and DIRM traceback and attribution technique based on packet marking, packet logging or hybrid approaches. The two approaches were based on deterministic packet marking capable of accessing the point as close as possible the source of attack and record the evidence based on evaluation metric determination.

The availability of a framework with practical solutions in examination, data analysis and investigation phases strengthen the investigation in network forensics by networks forensic practitioners, lawyers, computer forensics investigators, computer ethnical hackers and security agencies.

5.4 Recommendation for Further Research

Further research is recommended in correlation of attack features needs to be done using classification techniques in data mining and soft computing techniques. Improvement in data fusion of attack alerts and logs should be automated and

automation of generating FAR values should be put into consideration in future research.

Suitability of alternative statistical methods similar to confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance of the proposed scheme for analysing and tracing the attacked vectors of evidence need to be investigated for validating the attack. For more reliable and accurate evidence, other combination of network sensors tools than the ones used should be considered in future. Fragmentation of internet traffic also needs to be facilitated while marking packets in order to account for many attack strategies.

Important task that remains unresolved is NAT for network forensic hindrance complicate the traceback and attribution techniques making forensic more difficult to move closer to the attacker. The data sets used were locally generated in the academic environment validated on real world online datasets over a specified period of time in future research in order to achieve accurate forensic validated evidence.

REFERENCE

Abdullahi, A. R. D. (2020). Detecting Ransomware Using Process Behavior Analysis. *Procedia Computer Science*, pp 289-296.

Abdullah, A. K. (2022). Digital forensics and cyber forensics investigation: security challenges, limitations, open issues, and future direction. International Journal of Electronic Security and Digital Forensics, pp124–150.

Abdullah, A. K. S. A (2021) "Network Forensics Investigation: Behaviour Analysis of Distinct Operating Systems to Detect and Identify the Host in IPV6 Network", *International Journal of Electronic Security and Digital Forensics*,pp 600-611.

Abdullah, H. I.(2022). Digital Forensics Investigation Procedures of Smart Grid Environment. *International Journal of Computing and Digital Systems*, pp1071-1082.

Abdullah, Y. N.(2021).Single Packet AS Traceback against DoS Attacks,*2021 IEEE International Systems Conference (SysCon)*, 2021, pp. 1-8.

Abdulalem, A. S. A. D. (2021). Validating Mobile Forensic Metamodel Using Tracing Method. *Advances on Intelligent Informatics and Computing ,pp*473-482.

Andreas H.&Jürgen C. (2023). Getting pwn'd by AI: Penetration Testing with Large Language Models. Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations, pp2082–2086.

Aladaileh, M. A.A. M. W. (2020). Detection Techniques of Distributed Denial of Service Attacks on Software-Defined Networking Controller. *IEEE Access*, pp143985-143995.

Al-Dhaqm, A.S. (2020). Towards the Development of an Integrated Incident Response Model for Database Forensic Investigation Field, in *IEEE Access*, pp145018-145032.

Alharbi, S., Jens, W. J.& Traore, I. (2019). The Proactive and Reactive Digital Forensics Investigation Process: A Systematic Literature Review. *International Conference on Information Security and Assurance,pp*87-100.

Ali A. & Yousaf M. M. (2020). Novel Three-Tier Intrusion Detection and Prevention System in Software Defined Network, in *IEEE Access*, 8, pp109662-109676.

Amal, P. M. V. (2022). Review of Cyber Attack Detection: Honeypot System. *Webology*,pp5497-5514.

Anita, P. S. B. (2021). Roadmap of Digital Forensics Investigation. *Cyber Security and Digital Forensics*, pp241-269.

Anita P. S. B. (2022). *Roadmap of Digital Forensics Investigation Process with Discovery of Tools.* Wisley.

Antonia, N. G. L. (2021). Data-Driven Decision Support for Optimizing Cyber Forensic Investigations. *IEEE Transactions on Information Forensics and Security* , pp2397 - 2412.

Athanasios, D. N. I. (2019). Digital forensics framework for reviewing and investigating cyber attacks. *Elsevier Inc. T under open access article under the CC BY-NC-ND license,* pp1-8.

Atonu, G. K. M. (2020). A Systematic Review of Digital, Cloud and IoT Forensics. *The "Essence" of Network Security: An End-to-End Panorama* , pp31-74.

Arulanand, M. K. (2021). Network Support for IP Traceback Model in Wireless Sensor. *Wireless Personal Communications,* 3808-3821.

Aswanandini, R. &Deepa, C. (2023). Comparison of Advanced Classification Algorithms Based Intrusion Detection from Real-Time Dataset. *Aut. Control Comp. Sci.*57, 287–295.

Ashwini, D. A. (2024). Collecting and analyzing network-based evidence. *Computer Science and Information Technologies*, 1-6.

Ashutosh P. I. R. (2020). Hybrid Planning Using Learning and Model Checking for Autonomous Systems. *IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS)*, pp20-26.

Ates, C. (2020). Graph-based Fuzzy Approach against DDoS Attacks. *Journal of Intelligent &Fuzzy Systems, pp*6315 – 6324.

Aymen, A. M. F. S. (2020). Ontology-Based Smart Sound Digital Forensics Analysis for Web Services. *Digital Forensics and Forensic Investigations: Breakthroughs in Research and Practice*, 24.

Babulak, E. (2023). *AI Tools for Protecting and Preventing Sophisticated Cyber Attacks.* IGI Global's InfoSc.

Bijalwan, A. (2021). *Network Forensics :The Privacy and Security*. New York: Chapman and Hall/CRC.

Blasch, E. (2021). "Machine Learning/Artificial Intelligence for Sensor Data Fusion–Opportunities and Challenges," in *IEEE Aerospace and Electronic Systems Magazine*, pp80-93

Bouyeddou, B. H. (2020). Detecting network cyber-attacks using an integrated statistical approach. Cluster Comput 24. *Cluster Computing*, pp1435–1453.

Bro Network Security Monitor. (2022). Retrieved from The Bro Network Security Monitor Web site: http://www.bro.org.

Carvalho B. (2021). An End-to-End Framework for Machine Learning-Based Network Intrusion Detection System,. *in IEEE Access,*, 106790-106805.

Chourasiya, S. (2024). Categorizing Tracing Techniques for Network Forensics. *Cyber Security and Digital Forensics*, pp978-981.

Dalal, M., Juneja, M. (2021). Steganography and Steganalysis (in digital forensics): a Cyber security guide. *Multimed Tools Appl*,pp5723–5771,

DCI (2020). *Directorate of criminal investigation*. Retrieved from Directorate of criminal investigation: www.cid.go.ke.

Dumitrache, C. P.G. (2022). Comparative analysis of routing protocols using GNS3, Wireshark and IPerf3, International *Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, pp1-6.

Fadel , M. M. (2021). Hybrid Distributed Single-packet Low-storage IP Traceback Framework. *Mansoura Engineering Journal, (ME)*, pp75-89.

Ferguson, R. I. K. R. (2020). PRECEPT: A framework for ethical digital forensics investigations. *Journal of Intellectual Capital*,1469-1930.

Francesco, F. X. D. (2021). pcapindex: An index for network packet traces with legacy compatibility. *SIGCOMM Computer Communications Review*,47–53.

Francesco, F. M. P. (2020). NET-FLi: On-the-fly Compression, Archiving and Indexing of Streaming Network Traffic. *Proceedings of the VLDB Endowment*,1382–1393).

Frank, B. J. N. H. (2021). DFRWS EU 10-year review and future directions in Digital Forensic Research. *Forensic Science International: Digital Investigation*, 2666-2817.

Fornasini, F. (2019). Observability and Reconstructibility of Probabilistic Boolean Networks, in *IEEE Control Systems Letters*, pp 319-324,

Gebrye, H., Wang, Y. & Li, F. (2023). Traffic data extraction and labelling for machine learning based attack detection in IoT networks. *Int. J. Mach. Learn. & Cyber.*14, 2317–2332.

Ghabban, F. M. D. K. (2021). Comparative Analysis of Network Forensic Tools and Network Forensics Processes. *International Conference on Smart Computing and Electronic Enterprise (ICSCEE)* ,pp78-83.

Giovanna, C. G. N. (2021). The ISO/IEC 27001 security management standard: literature review and theory-based. *Polytechnic Department of Engineering and Architecture, University of Udine*, 77-105.

Gregor M. R. S. (2020). Enriching Network Security Analysis with Time Travel. *Applications, Technologies, Architectures, and Protocols For Computer Communications* , 17-25.

Godwin,B. W. K. (2020). Comparative Analysis of the Performance of Network. *International Journal of Computer Applications*, 39-44.

Gupta, A. (2020). A categorical survey of state-of-the-art intrusion detection system-Snort. *International Journal of Information and Computer Security*, 337-356.

Gupta, R. F. S.S. (2021). "Grid-Aware Distributed Model Predictive Control of Heterogeneous Resources in a Distribution Network: Theory and Experimental Validation," in *IEEE Transactions on Energy Conversion*, 1392-1402.

Haddadi, M. Y. O. (2022). A Single-Packet IP Traceback: Combating DoS-DDoS Attacks. *EDPACS*, 1-12.

Haider, W., Oleiw, N. S. T. (2022). An Enhanced Interface Selectivity Technique to Improve the QoS for the Multi-homed Node. *Engineering and Technology Journal*, 1006- 1013.

Hamza A. M. I. (2023). Cybercrime Unmasked: Investigating Cases and Digital. *International Journal of Emerging Multidisciplinaries*, 1-31.

Hemdan, E.E. D., M.D. (2021). An efficient digital forensic model for cybercrimes investigation in cloud computing. *Multimed Tools Appl*, 14255–14282.

Heydarian, M. T. E. (2022). "MLCM: Multi-Label Confusion Matrix," in *IEEE Access*, 10, pp. 19083-19095.

Himanshu, S. A. (2022). Digital Forensics Techniques and Trends: A Review. *The International Arab Journal of Information Technology, 20, (4)*, 644-657.

Humaira, A. A. J. (2020). Formal knowledge model for online social network forensics. *Computers & Security*, 234-247.

Humayun, M. N. (2020). Cyber Security Threats and Vulnerabilities: A Systematic Mapping Study. *Computer Engineering and Computer Science*, 3171–3189.

Hussein, E. (2021). Proposed intelligence systems based on digital Forensics:. *International Conference on Nanoelectronics, Nanophotonics, Nanomaterials, Nanobioscience & Nanotechnology.* (pp. 32-42). Malaysia: Elsevier Ltd.

Iftikhar, A Q. E. (2022). An Efficient Network Intrusion Detection and Classification System. *Applied Mathematics for 5th Generation (5G) and beyond Communication Systems)*, 1-15.

Ilyas, M. U. & Alharbi, S. A. (2022). *Machine learning approaches to network intrusion detection for contemporary internet traffic.* Retrieved from https://orcid.org/0000-0002-5694-1569:https://doi.org/10.1007/s00607021010 50 -5

Jan P. F. B. (2020). Netfox detective: A novel open-source network forensics analysis tool. *Forensic Science International: Digital Investigation*, 1-10.

Javed, A. R., W. A. M. A (2022). A Comprehensive Survey on Computer Forensics: State-of-the-Art, Tools, Techniques, Challenges, and Future Directions, in *IEEE Access*, 11065-11089.

Jihyung L. S. L. (2021). FloSIS: A Highly Scalable Network Flow Capture System for Fast Retrieval and Storage Efficiency. *USENIX Annual Technical Conference*, 56-64.

Johnson, K. (2021). *Base:Basic Analysis and Security Engine.* base.secure.net: http:/base.secureideas.net/

Kalangi, R. R., P. S. (2021). A Hybrid IP Traceback Mechanism To Pinpoint The Attacker.*2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, 1613-1618.

Kotha, S.K., Rani, M.S., Subedi, B. (2022). A Comprehensive Review on Secure Data Sharing in Cloud Environment. *Wireless Personal Communications*, 2161–2188.

Koroniotis, N. N. M. (2020). A new network forensic framework based on deep learning for Internet of Things networks: A particle deep framework. *Future Generation Computer Systems*, 91-106.

Kulandaivel, M. A. (2022). Network Support for IP Traceback Model in Wireless Sensor Networks Using Quantum Annealing Method. *Wireless Personal Communications* , 3807–3821.

Kumar, K. M. K. (2021). Architecture of Digital Twin for Network Forensic Analysis Using Nmap and Wireshark.*Digital Twin Technology* (p.22 eBook ISBN 9781003132868). UK: CRC Press.

Mario, D. M. (2020). Experimental Review of Neural-Based Approaches for Network Intrusion Management. *IEEE Transactions on Network and Service Management 17(4),* 2480 - 2495.

Marlien, H. A. B. (2020). Applying Design Science research as a methodology in post graduate studies:. *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology* , 251–258.

Matthias, V. R. S. (2020). Scalably Stateful Network Intrusion Detection on Commodity Hardware. *The NIDS Cluster,* 223-230.

Mauro Tropea, M. V. (2020). On packet marking and Markov modeling for IP Traceback: A deep. *Computer Networks,* 1-14.

Medina, E. F. (2020). Rectification and Super-Resolution Enhancements for Forensic Text Recognition. *International Conference on Imaging for Crime Detection and Prevention (ICDP-19),* 2345-2352.

Mei Y., H. W. (2024). A Novel Network Forensic Framework for Advanced Persistent Threat Attack Attribution Through Deep Learning,. *IEEE Transactions on Intelligent Transportation Systems,,* 1-10.

Memon, P. G. (2019). NetStore: An Efficient Storage Infrastructure for Network Forensics and Monitoring. *International Symposium on Recent Advances in Intrusion Detection (RAID),* 277–296.

Mingxing L. Y. L. (2021). A Traffic-Aware Probabilistic Packet Marking for Collaborative DDoS Mitigation. *2021 17th International Conference on Mobility, Sensing and Networking (MSN)* (pp. 25-34). Exeter, United Kingdom : IEEE.

Mohammad, R. A.M. (2021). Generic Proactive IoT Cybercrime Evidence Analysis Model for Digital Forensics. *International Conference on Information Technology (ICIT),* 654-659.

Mohammad, S. O, P. V. (2019). *Social Network Forensics, Cyber Security, and Machine Learning.* Amazon: Springer.

Moustafa, N. A. (2019). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set).*Military Communications and Information Systems(MilCIS), IEEE,* 67-74.

Muniu, H. (2020). *PwC 2020 Global Economic Crime and Fraud Survey Kenyaeport.* Kenya: Fraud The overlooked competitor.

Nagendra, K. N. A. P. (2023). *Wireshark for Network Forensics.* Springer , 1-27.

Nejma, M., & Cherkaoui, A. (2020). Enabling multi-tier collaboration between supply chain dyads: a conceptual modelling framework. *Supply Chain Forum: An International Journal, 21*(1), 35–52.

Nickolaos. N. (2022). A new Intelligent Satellite Deep Learning Network Forensic framework for smart satellite networks. *Computers and Electrical Engineering,* 1-10.

Ordabayeva G. K. O. M. (2020). A Systematic Review of Transition from IPV4 To IPV6. *ICEMIS'20: Proceedings of the 6th International Conference on Engineering & MIS,* 1–15.

Paulo, V. C. S. A. J. (2020). An Intelligent Hybrid Model for the Construction of Expert Systems in Malware Detection. C*onference on Evolving and Adaptive Intelligent Systems (EAIS),* 1-8.

Prabhjot, K. A. B. (2019). Network Forensic Process Model and Framework: An Alternative Scenario. *Intelligent Communication, Control and Devices pp 493–502*, 493–502.

Qadeer,H. (2020). Towards an Efficient Intrusion Detection System for High Speed Networks, *2020 17th International Bhurban Conference on Applied Sciences and Technology (IBCAST)*, Islamabad, Pakistan, pp. 428-433.

Qadir, S.& Noor B. *(*2021).Applications of Machine Learning in Digital Forensics, International *Conference on Digital Futures and Transformative Technologies (ICoDT2)* Islamabad, Pakistan, pp. 1-8

Rachana P. Y. H. (2022). A Hybrid Traceback based Network Forensic Technique to Identifying Origin of. *Journal of Engineering Science and Technology Review*, 28 - 34.

Rashmi M.& Jain, A. (2021). EnNetForens: An Efficient Proactive Approach For Network Forensic. *nternational Conference on Communication, Control and Information Sciences (ICCISc)*, 152476-152502,.

Rachana, Y.&Patil, S. R. (2022). Network Forensic Investigation Protocol to Identify True Origin of Cyber Crime. *journal of King Saud University - Computer and Information Sciences*, 2031-2044.

Rachana Y. &Patil M.A.R. (2021). A New Network Forensic Investigation Process Model. *Mobile Computing and Sustainable Informatics, 139–146*.

Rajput, B. (2020). Integrated Cyber Crime and Cyber Security Model. In: Cyber Economic Crime in India. Springer Series on Asian Criminology and Criminal Justice Research, 227–260.

Reith, C. G. (2013). An Examination of Digital Forensic Models. *International Journal of Digital Evidence,* 1(3)12-16.

Ricci Joseph, I. B. (2019). Blockchain-Based Distributed Cloud Storage Digital Forensics. *IEEE Security & Privacy*, 34- 42.

Richard Hill, V. C. F. (2019). The Standardised Digital Forensic Investigation Process Model (SDFIPM). *Advanced Sciences and Technologies for Security Applications* , 169–209.

Rusydi Umar, I. R. (2021). Analysis of Conti Ransomware Attack on Computer Network with Live Forensic Method. *IJID (International Journal on Informatics for Development),* 2549-744.

Saeed, S., (2023). Digital Transformation and Cybersecurity Challenges for Businesses Resilience: Issues and Recommendations . *Sensors MDPI*, 1-23.

Sadegh, T. E. B. H. (2020). A Scalable Platform for Enabling the Forensic Investigation of Exploited IoT Devices and Their Generated Unsolicited Activities. *Forensic Science International: Digital Investigation*, 1-20.

Sahu, T. (2021). Multi-Source Multi-Domain Data Fusion for Cyber-attack Detection in Power Systems, in *IEEE Access.* 119118-119138.

Samuel Chng, H. Y. (2022). Hacker types, motivations and strategies: A comprehensive framework. *Computers in Human Behavior Reports 5*, 1-8.

Sandeep Kaur, M. S. (2023). *Network Forensics* . Cyber Security Using Modern Technologies CRC Press.

Saputra, R.Z.(2022). Feature Selection using Chi Square to Improve Attack Detection Classification in IoT Network:. *International Conference on Electrical Engineering, Computer Science and Informatics (EECSI), Jakarta, Indonesia,*, 226-232.

Sheikhalishahi, M. S. (2022). Privacy preserving data sharing and analysis for edge-based architectures. Int. J. Inf. Secur. . *International Journal of Information Security*, 79–101.

Sikos, L. F. (2020). Packet analysis for network forensics : A Comprehensive survey. *Forensic Science International : Digital Investigation* , 1-12.

Singh, K. S. (2019). Cyber Forensics and Comparative Analysis of Digital Forensic Investigation Frameworks, *International Conference on Information Systems and Computer Networks (ISCON)*, 584-590.

Sirajuddin Q. J. H. (2021). Network Forensics: A Comprehensive Review of.Tools and Techniques. *International Journal of Advanced Computer Science and Applications*, 879-887.

Soliman A. E. S. E. (2023). A framework for digital forensics of encrypted real-time network traffic, instant messaging, and VoIP application case study. *Ain Shams Engineering Journal*, 1-18.

Steffen, H. F. W. (2019). Efficient Attack Correlation and Identification of Attack Scenarios based on Network-Motifs. *2019 IEEE 38th International Performance Computing and Communications Conference (IPCCC)*, 1-11.

Sunde, G. H. (2021). Unboxing the digital forensic investigation process. *Science and Justice* , 171-180.

Suresh, S. S. R, N. (2020). Feasible DDoS attack source traceback scheme by deterministic multiple packet marking mechanism. *J Supercomput*, 4232–4246.

Talib, M. J. A. (2021). Identifying Digital Forensic Frameworks Based on Processes Models. *Iraqi Journal of Science,* , 249–258.

Thomas J. & Holt, A. M. S. (2022). *Cybercrime and Digital Forensics.* London: Taylor Francis Group.

Thomas. M. M. B. (2023). A Brief Review of Network Forensics Process Models and a Proposed Systematic Model for Investigation, *Intelligent Cyber Physical Systems and Internet of Things*, 599–627.

Tiffanie, E. S. M. (2021). On Exploring the Sub-domain of Artificial Intelligence (AI) Model Forensics. *Digital Forensics and Cyber Crime* , 35–51.

Tinubu, C. S. (2022). DT-Model: a classification model for distributed denial of service attacks and flash events. *International Journal of Information Technology,* 3077–3087.

Tiwari, A. V. M. S.G. (2021). Developing Trends and Challenges of Digital Forensics,*2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, Mathura, India, 2021, pp. 1-5

Tobias, L. F. H. (2021). Bringing Forensic Readiness to Modern Computer Firmware. *DFRWS*, 20-28.

Toraskar, T. S.(2019). Efficient Computer Forensic Analysis Using Machine Learning Approaches. *IEEE Bombay Section Signature Conference (IBSSC)*, 189-199.

Victor R. K, H. S. (2020). A comparative analysis of digital forensic readiness models using CFRaaS as a baseline. *Wires Forensic Science* , 1-11.

Victor R.K, P.P. (2020). Holistic digital forensic readiness framework for IoT-enabled organizations. *Forensic Science International:*, 78-98

Vishwakarma, R. J. (2020). A survey of DDoS attacking techniques and defence mechanisms in the network. *Telecommunication Systems*, 3-25.

Wang R., Ji .W. (2020) "Computational Intelligence for Information Security: A Survey," in *IEEE Transactions on Emerging Topics in Computational Intelligence*, 616-629.

Wang, X.-J. a.-y. (2021). "Topology-assisted deterministic packet marking for IP traceback." . *The Journal of China Universities of Posts and Telecommunications* , 116-121.

Warusia Y, M. F. (2020). Cloud Forensic Challenges and Recommendations: A Review. *OIC-CERT Journal of Cyber Security*, 19- 29.

Yang, M..H (2020) "Hybrid Multilayer Network Traceback to the Real Sources of Attack Devices," in *IEEE Access*, vol. 8, 201087-201097

Yonghui, L. e. (2020). Deterministic packet marking based on the coordination of border gateways. *Education Technology and Computer (ICETC),*, 1234-1240.

Yuan Zuo, X. Z. (2022). Heterogeneous big data fusion in distributed networking systems for anomaly detection and localisation. *International Journal of Security and Networks*, 220-229.

Yunfei Z, B. C. (2021). Broad Learning System Based on Maximum Correntropy Criterion. *IEEE Transactions on Neural Networks and Learning Systems* , 3083 - 3097.

Yusof M. H. M., Z. A. M. (2021). Network-Level Behavioral Malware Analysis Model based on Bayesian Network," . *International Conference on Computer & Information Sciences (ICCOINS)*, 316-321.

APPENDICES

Appendix I: Research Instruments

Description of Network Forensic Analysis Tools (NFATs)	
NFAT	**Description**
Infinistream	Exploits intellectual Deep Packet Capture (iDPC) expertise and implements active examination. It sort out extraordinary rate internment of numerous packet particulars, numerical examination of packet or stream established information and identifies hundreds of submissions. It practices refined optimization SRDM and indexing
Network Miner	Identifies network traffic by active sniffing, implements client detection, reconstructs transmitted data, capture intruder peers and evaluates how greatly an intruder exaggerated information escape.
Xplico	Examine Internet traffic, dissevers information at the networks protocol level, restructures and regulates it aimed to be use by exploiters. The exploiters recodes, connect and combined records for examination and existent the outcomes in an envisaged form
PyFlag	A progressive forensic tool to examine network internments in libpcap arrangement while associating several network protocols used. It has the capability repeat, observe records at numerous levels and preferably appropriate network protocols that are classically layered. PyFlag analyses the pcap files, excerpts the packets and dissevers them at physical, data-link and network layers respectively including IP, TCP or UDP network protocols. Associated packets are gather into streams by means of reconstruction. This traffic are then dissevered with advanced upper protocols like HTTP, IRC, etc for dissection, Himanshu(2022)
SilentRunner	Gathers, examines and envisages network events by discovering incident challenges, anomalous practice, exploitation and inconsistencies. It creates collaborative visual images of the sequences actions and associates genuine network stream.

	Likewise shows back and rebuilds security events in their particular order.
Solera DS 5150	DS 5150 is an application for very high rate records internment, comprehensive indexed logs of network traffic, filter, restoration and replay. DS forensic collection has three software namely Reports for reconstruction, Sonar for indexing and Search for searching all network traffic.
OmniPeek	Delivers active reflectivity into each fragment of the network. It has high identify abilities, central console, dispersed engines and proficient examination.Omnipliance is a network logging application with a multi-terabyte disk farm and high rate internment interfaces. Omni-Engine software internments and stores network traffic. Omni-Peek interface searches, mines and detect data for particular evidence
NetIntercept	Collects network traffic and preserves in pcap design, reconstructs single files traffic, evaluates them by analysing to identify the protocol, discovers spoofing and produces a diversity of evidence from the outcomes.
NetWitness	Identify entirely network traffic and recreates the network gatherings to the application layer for automatic notifying, checking, collaborating examination and assessment
NetDetector	Collects interventions, incorporates mark established abnormality discovery, recreates application sessions and accomplishes various scale examination on dissimilar applications and protocols. It has spontaneous managing console and complete standards centred recording tools. It ingresses and disseminates data in a diversity of setups.
Iris	Assembles network traffic and reconstructs it in its intuitive assembly centred fragments, recreates concrete scripts of the assembly, repetitions of traffic for examination test of mistrustful actions, delivers a diversity of statistical depths and has progressive examination and filtering tools for speedy discovery of evidence.

Description of Network Security and Monitoring (NSM) Tools Table	
NSM tool	**Description**
TCPDump	A common packet sniffer and analyser, runs in command line, intercepts and displays packets being transmitted over a network. It captures, displays, and stores all forms of network traffic in a variety of output formats. It will print packet data like timestamp, protocol, source and destination hosts and ports, flags, options, and sequence numbers.
Wireshark	Most popular network protocol analyser. It can perform live capture in libpcap format, inspect and dissect hundreds of protocols, do offline analysis, and work on multiple platforms. It can read and write files in different file formats of other tools.
TCPFlow	Captures data transmitted as part of TCP connections (flows) and stores it for protocol analysis. It reconstructs actual data streams and stores in a separate file. TCPFlow understands sequence numbers and will correctly reconstruct data streams regardless of retransmissions or out-of-order delivery.
NfDump:	NfDump: A suite of tools working with NetFlow format: nfcapd – NetFlow capture daemon reads the NetFlow data from the network and stores it. NfDump – NetFlow dump reads the NetFlow data from these files displays them and creates statistics of flows, IP addresses, ports etc. nfprofile – NetFlow profiler filters the NetFlow data according to the specified filter sets and stores the filtered data. nfreplay – NetFlow rerun leads data with the entire network to clients.
PADS	PADS is a transferable, trivial and intellectual network sniffer. It is a signature-based discovery engine used to inactively discover network properties. It can snort TCP, ARP and ICMP traffic stream. It can transfer evidence around exclusive properties and facilities appreciated on the network into perpetual storage, production it in CSV or MySQL set-up or current user approachable report.
Nessus	Susceptibility scanner presenting high-speed detection, configuration examining, ability profiling, penetrating data detection and susceptibility examination

Sebek	Kernel founded data internment tool aimed to detention all action on a Honeypot. It registers keystrokes of a gathering that is expending encryption mend files copied with SCP, internment passwords used to record in to isolated system and complete numerous additional forensics associated responsibilities.
TCPTrace	Produce dissimilar varieties of production comprising of information, such as intervened time, bytes/fragments which are guided and established, round trip times, window announcements throughput and retransmissions
Ntop	Used for traffic amount, network traffic checking, optimization, scheduling and discovery of network safety damages. It offers provision for both tracing current intrusions and detecting possible security weakness comprising denial of service intrusions, port scan intrusions and IP spoofing.
TCPStat	Captures network interface indicators like bandwidth, amount of packets, rate of packets, mean of packet magnitude, and typical deviation size of packet and load of interface by checking an interface or analysing from libpcap file.
IOSNetFlow	Gathers and amount of IP packet features of every packet accelerated through switches or routers, collect flow of related packets, to aid to determine traffic flow and recognise who, what, when, where and how the traffic pattern. Similarly, it identifies network abnormalities and security weaknesses.
TCPDstat	Outputs each protocols analysis of traffic, for a specified libpcap file, including quantity of packets, packet speed, standard deviation, quantity of distinctive source and destination IP address. Likewise, it is essential in attaining high-elevated interpretation of traffic outlines.
Ngrep	Captures records from network traffic established on file marks. Similarly, it captured logs transferred through the networks.
SiLK	Internment, storage and inspects network flow data founded on Cisco Net Flow. The suite tool comprising of gathering and examination tools, provides experts in forensics with the ways in understanding, probing and summarizing together present and past traffic information

	in network flow logs. Help in network forensics detecting articles of attacks, susceptibility activities, worm actions etc. It performs a key constituent and accomplishes the bulky capacity of traffic storage security associated information.
TCP Replay	Group of tools with capability to categorize earlier apprehended traffic as peer or server; modify data link, network and transport layers headers respectively and restate the traffic back onto the network. Multi-pass pcap file pre-processor regulates packets as peer or server. TCP replay is a pcap file editor that modify packet headers, sample randomly the speeds of the network and segment two network.
Pof	Passive OS finger reproduces by apprehending traffic imminent from a peer to the network. Likewise it identify use of NAT, presence of firewall, presence of load balance outfit, its uptime and reserve to the isolated system.
Nmap	Nmap is useful for network examination and security examining. It consists of numerous varieties of port scans and can be utilise as on OS finger examination tool. It practices unprocessed IP packets in unique techniques to regulate peers accessible on the network, facilities existence obtainable, success of operating systems, installed firewalls and numerous additional features.
Bro	NIDS that inactively screens network traffic for apprehensive actions. It identifies interferences by first analysing network traffic to abstract its application level semantics and then perform incident focused on analysers that associate this events with outlines troublesome. It is predominantly a research policy for IDS, traffic investigation as well as network forensics.
Snort	Network attacks anticipation or discovery system accomplished by executing packet recording, detecting and active traffic examination. It implement protocol examination, details leads probes, similar and application level investigation. It seizures the traffic in libpcap layout that is can be utilised in network forensic.

Appendix II: Network Forensic Principles and Procedures

The following network forensic principles and procedures must be followed by the forensic expert during analysing of network security incidents.

i.) The activities of the network forensic practitioner should not alter the original data. If the requirements of the work mean that this is not possible then the effect of the expert's actions on the original data should be clearly identified and the process that caused any changes justified.

ii.) A complete record of all activities associated with the identification, mapping, analysing and managing network security incidents of the unique information and several replicas of the unique information necessity to be preserved. This comprises submission with the suitable procedures of substantiation, such as preserving a sequence of protection evidence, and confirmation procedures like hashing.

iii.) The forensic expert necessity not assumes actions, which are outside their capability or awareness.

iv.) The forensic expert necessity takes into respect all features of individual and tools protection while carrying out all action of their work.

v.) Altogether, times the authorized privileges of anybody affected by your activities must be well thought and put into consideration.

vi.) The expert necessity awareness of entire logistic strategies and processes linking to their actions.

vii.) Message necessity preservation suitable with the user, authorized experts, superiors and other group associates.

Appendix III: Packets Structure and Marking Approaches

Packet Structure

IPv6 Packet structure

IPv6 headers have one Fixed Header and zero or more Optional (Extension) Headers. All the necessary information that is essential for a router is kept in the Fixed Header. The Extension Header contains optional information that helps routers to understand how to handle a packet/flow.

- **Version** (4-bits): It represents the version of Internet Protocol, i.e. 0110.
- **Traffic Class** (8-bits): These 8 bits are divided into two parts. The most significant 6 bits are used for Type of Service to let the Router Known what services should be provided to this packet. The least significant 2 bits are used for Explicit Congestion Notification (ECN).
- **Flow Label** (20-bits): This label is used to maintain the sequential flow of the packets belonging to a communication. The source labels the sequence to help the router identify that a particular packet belongs to a specific flow of information. This field helps avoid re-ordering of data packets. It is designed for streaming/real-time media.
- **Payload Length** (16-bits): This field is used to tell the routers how much information a particular packet contains in its payload. Payload is composed of Extension Headers and Upper Layer data. With 16 bits, up to 65535 bytes can be indicated; but if the Extension Headers contain Hop-by-Hop Extension Header, then the payload may exceed 65535 bytes and this field is set to 0.

TCP Header

A TCP packet consists of two sections, header and data. All fields may not be used in every transmission. A flag field is used to indicate the type of transmission the packet represents and
how the packet should be interpreted

- **Source port**—identifies the sending application.
- **Destination port**- Identifies the destination application.
- **Sequence number**—Used for assembling segmented data in the proper order at the receiving end

- **Acknowledgement number**—The sequence number the sender (the receiving end) expects next
- **Data offset** or header length —The size of the TCP header, it is also the offset from the start of the TCP packet to the data portion.
- **Reserved**—Reserved for future use, should be set to zero.
- **Flags**(also known as control bits)—contains 6 1-bit flags
    - **URG-**Urgent pointer field is significant
    - **ACK**-Acknowledgement field is significant.
    - **PSH**-Push function.
    - **RST**-Reset the connection.
    - **SYN**-Synchronize sequence numbers
    - **FIN**-No more data from sender.

Window—the number of bytes the sender is willing to receive starting from the acknowledgement field value.

Checksum—used for error-checking of the header and data

Urgent pointer—if the URG flag is set, then this 16-bit field is an offset from the sequence number indicating the last urgent data byte

Options—Additional header fields (called options) may follow the urgent pointer

The UDP Packet Structure

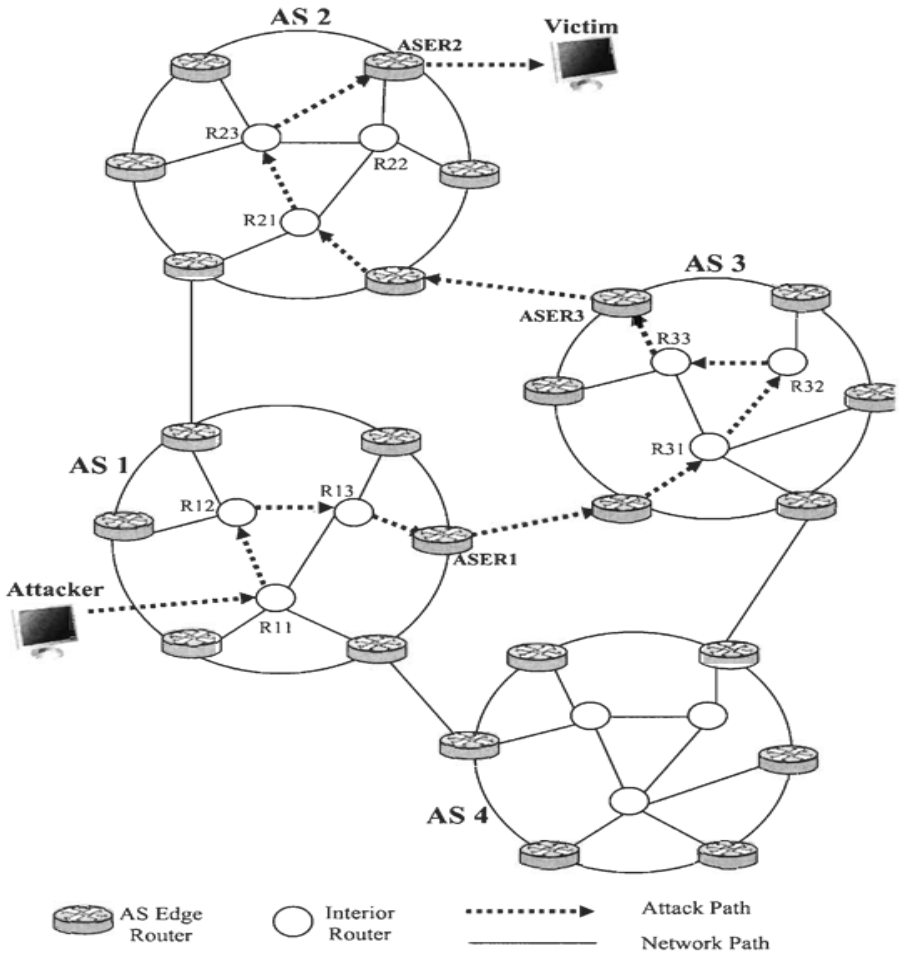
The UDP packet structure consists of 5 fields, some of which are optional:

- **Source Port-**The sending application. This is an optional field
- **Destination Port-**The target application at the receiving end.
- **Length**-The length of the entire packet.
- **Checksum-**Optional field used to perform basic error correction on the packet.
- **Data**-The user data to be transmitted.

Packet Marking Approaches, Tracing And Attribution Simulation Codes

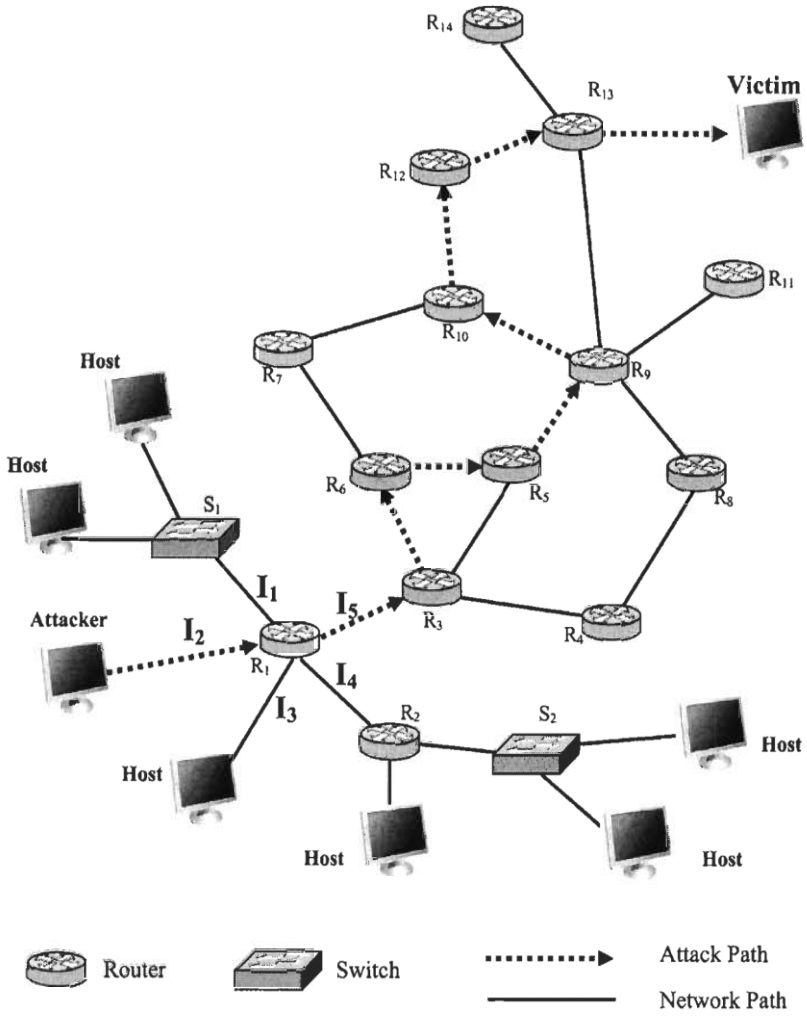
Autonomous System based Deterministic Packet Marking (ASDPM)

The two level traceback mechanism implemented where the first level deterministic mark each packet by the first internal router within AS. The second router mark each packet by ASBR marks the with 16 bits hash of the router address as in DERM which reduces the number of false positive due to hash collisions as also in the as shown in the figure below.



Deterministic Router and Interface Marking (DR1M)

Deterministic marking of the address of the router and the interface number through which the packet enters the network. The packet is marked by the ingress interfaces edge router, which places both the marks. Every outbound packet is marked and inbound packets are not marked.



ns 3: Attaching Sources segment code

The data source which were feed in bytes to our agent simulator. It generated a constant bit rate source (CBR) initialize by the following source code to trace the packet path.

```
# Create Source
set cbr1 [new Application/Traffic/CBR]

# Configure Source
$cbr1 set packetSize_ 500
$cbr1 set interval_ 0.005

# Attach source to agent
$cbr1 attach-agent $udp1

# Schedule cbr on
$ns at 0.5 "$cbr1 start"
# Schedule cbr off
$ns at 4.5 "$cbr1 stop"
```

ns 3: Tracing and attribution code

While our current simulation created sources and generate traffic which travelled over our link. The simulator traced network events for us.

Note in the full code file that the trace-all and nam-trace-all commands must be run before nodes and agents are created, but the variable trace commands was run afterwards initialized by the following source code to trace the packet path.

```
# all packet trace:
$ns trace-all [open out.tr w]

# animator trace:
$ns namtrace-all [open out.nam w]

# variable trace:
set tf [open "cwnd.tr" w]
set tracer [new Trace/Var]
$tracer attach $tf
$tcp trace cwnd_ $tracer
```

Appendix IV: Description of NSL KDD_Data Set

NO	Feature	Description	Type
1	Duration	Length of the connection.	Basic Features
2	protocol type	Connection protocol (e.g. tcp, udp)	Basic Features
3	Service	Destination service (e.g. telnet, ftp)	Basic Features
4	Flag.	Normal or error status of the connection	Basic Features
5	source byte	Bytes sent from source to destination	Basic Features
6	destination bytes	Bytes sent from destination to source	Basic Features
7	Land	1 - Connection is from/to the same host/port; 0 – otherwise.	Basic Features
8	Wrong Fragment	Number of "wrong" fragments	Basic Features
9	Urgent	Number of urgent packets	Content Features
10	Hot	number of "hot indicators".	Content Features
12	num_failed_logins	number of failed login attempts	Content Features
13	logged in	1 - successfully logged in; 0 – otherwise	Content Features
14	num_compromise	number of "compromised" conditions.	Content Features
15	root shell	number of "compromised" conditions.	Content Features
16	su_attempted	1 - root shell is obtained; 0 – otherwise.	Content Features
17	num_root	number of "root" accesses.	Content Features
18	num_file_creations	number file creation operations	Content Features
19	num_shells	number of shell prompts	Content Features
20	Num_access_files	number of operations on access control files	Content Features
21	Num_outbound_cmds	number of outbound commands in a ftp session	Content Features
22	is_hot_login	1 - the login belongs to the "hot" list; 0 – otherwise.	Content Features
23	is_guest_login	1 - the login is a "guest"login; 0 - otherwise	Content Features
24	Count	number of connections to the same host as the current connection in the past 2 seconds	Time-based Traffic Feature
25	srv_count	number of connections to the same	Time-based

		service as the current connection in the past 2 seconds	Traffic Features
26	serror_rate	% of connections that have “SYN” error	Time-based Traffic Features
27	rerror_rate	% of connections that have “REJ” errors	Time-based Traffic
28	same srv rate	% of connections to the same service	Time-based Traffic
29	diff srv rate	% of connections to different services	Time-based Traffic
30	srv_serror_rate	% of connections that have “SYN” errors	Time-based Traffic
31	srv_rerror_rate	% of connections that have “REJ” errors	Time-based Traffic
32	srv_diff_host_rate	% of connections to different hosts	Time-based Traffic
33	Dst_host_count	count of connections having the same destination host	Host-based Traffic Feature
34	dst_host_srv_count	count of connections having the same destination host and using the same service	Host-based Traffic Feature
35	dst_host_same_srv_ rate	% of connections having the same destination host and using the same service	Host-based Traffic Feature
36	dst_host_diff_srv_rate	% of different services on the current host	Host-based Traffic Feature
37	dst_host_same_src_port_rat	% of connections to the current host having the same src port	Host-based Traffic Feature
38	Dst_host_ srv_diff_host_rate	% of connections to the same service coming from different hosts	Host-based Traffic Feature
39	Dst_host_srv_rerror_rate	% of connections to the current host and specified service that have an S0 error	Host-based Traffic Feature
40	dst_host_serror_rate	% of connections to the current host that have an S0 error	Host-based Traffic Feature
41	dst_host_srv_serror_rate	% of connections to the current host and specified service that have an S0 error	Host-based Traffic Feature

Appendix V: Description of USW-NB15_Features

Feature #	Name	Description
	1	Flow features
1	srcip	Source IP address
2	sport	Source port number
3	dstip	Destinations IP address
4	dsport	Destination port number
5	proto	Protocol type, such as TCP, UDP
	2	Basic features
6	state	The states and its dependent protocol (e.g., CON)
7	dur	Row total duration
8	sbytes	Source to destination bytes
9	dbytes	Destination to source bytes
10	sttl	Source to destination time to live
11	dttl	Destination to source time to live
12	sloss	Source packets retransmitted or dropped
13	dloss	Destination packets retransmitted or dropped
14	service	Such as http, ftp, smtp, ssh, dns and ftp-data
15	sload	Source bits per second
16	dload	Destination bits per second
17	spkts	Source to destination packet count
18	dpkts	Destination to source packet count
	3	Content features
19	swin	Source TCP Window advertisement value
20	dwin	Destination TCP Window advertisement value
21	Stcpb	Source TCP base sequence number
22	dtcpb	Destination TCP base sequence number
23	smeansz	Mean of the packet size transmitted by srcip
24	dmeansz	Mean of the packet size transmitted by dstip
25	trans-depth	The connection of http request/response transaction
26	res_bdy_len	The content size of data transferred from http
	4	Time features
27	sjit	Source jitter
28	djit	Destination jitter
29	stime	Row start time
30	ltime	Row last time
31	sintpkt	Source interpacket arrival time
32	dintpkt	Destination interpacket arrival time
33	tcprtt	Setup round-trip-time, the sum of synack and ackdat
34	synack	The time between the SYN and the SYN_ACK packets
35	ackdat	The time between the ACK and the ACK_DAT packets
36	is_sm_ips_ports	If srcip(1) = dstip(3) and sportC(2) = dsport(4), assign 1

		else 0.
T ABLE 1	Continued	
Feature #	Name	Description
	5	Additional generated features
37	ct_state_ttl	No. of each state(6) according to values of sttl(10) and dttl(11)
38	ct_flw_http_method	No. of methods such as Get and Post in http service
39	is_ftp_login	If ftp session is accessed by userid and password, then 1 else 0
40	ct_ftp_cmd	No. of flows that have command in ftp session
41a	ct_srv_src	No. of rows of the same service(14) and srcip(1) in 100 rows
42a	ct_srv_dst	No. of rows of the same service(14) and dstip(3) in 100 rows
43a	ct_dst_ltm	No. of rows of the same dstip(3) in 100 rows
44a	ct_src_ltm	No. of rows of the same srcip(1) in 100 rows
45a	ct_src_dport_ltm	No. of rows of the same srcip(1) and dsport(4) in 100 rows
46a	ct_dst_sport_ltm	No. of rows of the same dstip(3) and sport(2) in 100 rows
47a	ct_dst_src_ltm	No. of rows of the same srcip(1) and dstip(3) in 100 rows

Appendix VI: Algorithm: Best Features Selection

1. Input: Datasets with Common Reduced Features
2. Output: A set of most relevant features
3. /*Stage 4.1: Gradually Delete Phase*/
4. Starting from the common features set CS[i]
5. Rank the CS[i], U2R[i], R2L [i], PROBE[i], and DOS[i] based on
6. The importance of the feature to the attack type (relevance value)
7. How many attack type the feature can detect
8. How many algorithms select this feature for each attack type
9. For j = 1 to i
10. If a feature is (used to detect ONLY DOS) AND it is (in the lowest ranked list of DOS)
11. Else if a feature is (used to detect ONLY PROBE) AND it is (in the lowest ranked list of PROBE)
12. Else if a feature is (used to detect ONLY R2L) AND it is (in the lowest ranked list of R2L)
13. Else if a feature is (used to detect ONLY U2R) AND it is (in the lowest ranked list of U2R)
14. Else if a feature is (used to detect DOS and PROBE) AND it is (in the lowest ranked list of DOS and PROBE)
15. Delete this feature
16. Update the CS[j]
17. Evaluate performance of the updated CS[j]
18. If better performance for U2R, R2L, and PROBE
19. Confirm feature deletion
20. Update CS[j]
21. Update BSA
22. Else
23. keep this feature
24. Update CS[j]
25. Update BSA
26. Next j
27. /*End of Gradually Delete Phase*/
28. /* Stage 4.2: Gradually Add Phase*/
29. Start by a common selected set CF(i) of features that are:
30. Selected as important for all attack types
31. Selected by all algorithms with high relevance value
32. Evaluate the performance of CF(i) Æ BSA
33. Do until Max BSA
34. Add the top ranked feature form the U2R(j) set to CF(i)
35. Evaluate the performance of CF(i)
36. If performance > BSA
37. Confirm adding this feature

38. Update CF(i)
39. Update U2R(j)
40. Update BSA
41. Else
42. Change the feature importance to lowest rank
43. Update U2R(j)
44. End if
45. Add the top ranked feature form the R2L(j) set to CF(i)
46. Evaluate the performance of CF(i)
47. If performance > BSA
48. Confirm adding this feature
49. Update CF(i) 50: Update R2L(j)
50. Update BSA
51. Else
52. Change the feature importance to lowest rank
53. Update R2L(j)
54. End if
55. Add the top ranked feature form the PROBE(j) set to CF(i)
56. Evaluate the performance of CF(i)
57. If performance > BSA
58. Confirm adding this feature
59. Update CF(i)
60. Update PROBE(j)
61. Update BSA
62. Else
63. Change the feature importance to lowest rank
64. Update PROBE(j)
65. End if
66. Add the top ranked feature form the DOS(j) set to CF(i)
67. Evaluate the performance of CF(i)
68. If performance > BSA
69. Confirm adding this feature
70. Update CF(i)
71. Update DOS(j)
72. Update BSA
73. Else
74. Change the feature importance to lowest rank
75. Update DOS(j)
76. End if
77. Repeat
78. Return BSA and CF(i)
79. /*End of Gradually
80. Add Phase*/

Appendix VII: University Approval Letter

KABARAK UNIVERSITY

BOARD OF POST GRADUATE STUDIES

Private Bag - 20157
KABARAK, KENYA
http://kabarak.ac.ke/institute-postgraduate-studies

Tel: 0773265999
E-mail: directorpostgraduate@kabarak.ac.ke

12/11/2020

The Director General
National Commission for Science, Technology & Innovation (NACOSTI)
P.O. Box 30623 – 00100
NAIROBI

Dear Sir/Madam,

RE: PETER KIPRONO KEMEI – GDI/M/1199/09/13

The above named is a candidate at Kabarak University pursuing PhD in Information Technology Security and Audit. He is carrying out a research entitled "***Network Forensic Framework for Managing Security Incidents***". He has defended his proposal and has been authorised to proceed with field research.

The information obtained in the course of this research will be used for academic purposes only and will be treated with utmost confidentiality.

Please provide the student with a research permit to enable her to undertake the research.

Thank you.

Yours faithfully,

KABARAK UNIVERSITY DIRECTOR POST GRADUATE STUDIES 03 DEC 2020 PRIVATE BAG - 20157, KABARAK

Dr. Wilson O. Shitandi
DIRECTOR, INSTITUTE OF POST GRADUATE STUDIES

Appendix VIII: NACOSTI Research Permit

REPUBLIC OF KENYA

NATIONAL COMMISSION FOR SCIENCE,TECHNOLOGY & INNOVATION

Ref No: **822655**

Date of Issue: **15/December/2020**

RESEARCH LICENSE

This is to Certify that Mr.. Peter Kiprono Kemei of Kabarak University, has been licensed to conduct research in Nakuru on the topic: NETWORK FORENSIC FRAMEWORK FOR MANAGING SECURITY INCIDENTS for the period ending : 15/December/2021.

License No: **NACOSTI/P/20/8247**

822655

Applicant Identification Number

Director General
NATIONAL COMMISSION FOR SCIENCE,TECHNOLOGY & INNOVATION

Verification QR Code

NOTE: This is a computer generated License. To verify the authenticity of this document, Scan the QR Code using QR scanner application.

Appendix IX: Evidence of Conference Participation

TENP/RPND/2024/006

The Eldoret National Polytechnic
P. O. BOX 4461 - 30100 Eldoret Tel: 0738092126/0797222666
Email: info@tenp.ac.ke

Certificate of Participation

is awarded to

Peter K. Kemei

for presenting a paper entitled: -

Network forensic Framework for Managing Security Incidents

During the 13TH Annual International Conference on ***'Competences In Applied Research, Science And Innovation For Sustainable Development'*** Held at The Eldoret National Polytechnic on **16th to 17th October 2024**

Principal

Coordinator Research & Development

This certificate is issued without any alteration whatsoever

Appendix X: List of Publications

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:08/August-2024 Impact Factor- 7.868 www.irjmets.com

EXAMINATION PHASE NETWORK FORENSIC FRAMEWORK FOR IDENTIFICATION AND CORRELATION OF ATTACK ATTRIBUTES

Peter. K. Kemei*1, Dr. Moses Thiga*2, Dr. Joel Cherus*3

*1Graduate Student, Kabarak University Information Technology School Of Engineering And Technology, Kenya.

*2Senior Lecturer, Kabarak University Information Technology School Of Engineering And Technology, Kenya.

*3Forensic Security Consultant Expert Researcher, Kenya.

DOI : https://www.doi.org/10.56726/IRJMETS61000

ABSTRACT

Network forensics is a science of determining and retrieving evidential information in a computer networked environment about a criminality in such a way as to make it admissible. The established computer networks forensic field lays a strong foundation for network forensics as standard security frameworks, tools and techniques are in place for phase detection, collection, preservation and presentation of evidence. However, little has been done to address phase examination. The main challenge identified on this phase is identification and correlation. The objectives of the study were to; analyse, investigate, identify, develop and evaluate a network forensic framework which addresses the challenge in examination. A methodology was specifically formalized on real time and post attacked network traffic investigation based on datasets prototype implementation. The proposed technique in examination phase is identification and correlation of traced datasets. The identification provided attempts made in compromising a system and assist during reconstruction of intruded information. The correlation validated the particular intrusion and guide in decision to proceed with investigation. The techniques resulted in confirmation of DDoS, Portscan and cross-site scripting attacks dataset.

Keywords: Network, Forensic, Framework, Examination, Identification, Correlation.

I. INTRODUCTION

Network forensics deals with determination and retrieval of evidential information in a networked environment about a criminality in such a way as to make it acceptable. It attempts to capture traffic data logged through ingress and egress network devices. It also involves evaluation, analyses of traceable, log data of network intrusions from the current network security products, and provides information to characterize intrusion or misbehavior features. However, as we advance network in home and business, there was a need to advance network forensic view from local host to the network and web application level. There was necessity to consider these transitions into concepts, designs, models, frameworks and prototypes capabilities and implementation.

Background to the Study

Network forensics is a scientific determination and recovery of evidential information in a networked setup about an attack using an approach that makes it permissible. It has evolved to an almost established investigation process in reaction to the intruder communal and includes capturing, recording and analysing of network actions in order to determine the source of attacks according to (Sirajuddin, 2021).. The network data traffic was identified using intrusion detection systems collected from existing network security sensors tools. This data was examined for attack classification and scrutinised to trace back the intruders. The procedure can convey out insufficiencies in security products, which can be applied to guide placement and enhancement of these tools. The methodology gathers the required evidence for incident response and investigation of the crime. Network forensics expedites recording evidence for examination and assist in understanding the intruder's methodology (Mei, 2024).. It provides to understand the tools used by the intruder and new techniques in which edge defenses were evaded. (Dalal, 2021), network forensic information can also convey insufficiencies in the current sensors network security tools. These tools can be hardened to become robust

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:06/Issue:10/October-2024 Impact Factor- 8.187 www.irjmets.com

COMPARATIVE MULTIDATA FUSION NETWORK FORENSIC ANALYSIS PHASE FRAMEWORK FOR MANAGING SECURITY INCIDENTS

Mr. Peter Kiprono Kemei[*1], Dr. Joel Cherus[*2], Dr. Moses Thiga[*3]

[*1,3]Kabarak University Information Technology School Of Engineering And Technology, Kenya.

[*2]Forensic Security Consultant Expert Researcher, Kenya.

DOI : https://www.doi.org/10.56726/IRJMETS62234

ABSTRACT

Network forensics determines and retrieval of evidential evidence in a computer networked environs about a criminal activities which is admissible by grieved party. Computer forensic and data science field lays a robust foundation for network forensics as security frameworks, tools and techniques are in place for detecting, collecting, preserving and presenting breached information. Nevertheless, less has been done in mitigating phase analysis challenges from existing network forensic framework. The multidata fusion, data redundancy and integration evidences from various network sensors tools is the main challenge in analysis phase. The objectives of the study were to; analyse, investigate, identify, develop and evaluate a network forensic framework which addresses the multidata fusion, data redundancy and integration. A methodology was specifically formalized on real time and post attacked network traffic investigation based on datasets prototype implementation. The proposed technique in analysis phase is multidata fusion, data redundancy and integration traced datasets. The multidata fusion frameworks consolidates captured evidences from various network security sensors. The data redundancy algorithm eliminates data duplication and integration algorithm consolidate various attacked evidences into single entity attacks dataset.

Keywords: Network, Forensic, Framework, Analysis, Multidata.

I. INTRODUCTION

The analysis framework built by aggregating the strengths of open source tools in accomplishing the task of collection and analysis. Network security sensors with self-contradictory and corresponding utilities are used Security tools with similarity build the redundancy and reliability of attack information. Diversity among the tools will ensure versatility. Data fusion performed on the alert and attack information generated by these sensors so that the decision to ascertain the accuracy of the intruder data. In case suspicious packets are detected security sensors gives notifications inform of alerts. Packet capture and analysis tools or sniffers identify sessions or connections with anomalies in network traffic. Traffic statistics are read from packet captures or from Netflow records taken from the network connection through a monitored device. Network security sensors analyses the packet attacks in order to increase the confidence level of evidence. The redundancy and data integration algorithm validates sample data set with attack packets generated in desk check test in order to eliminate data duplication and consolidate into single entity. The accuracy of these tools is validated using confusion matrix theory or criteria of accuracy and False Alarm Rate (FAR) metric applied to measure the performance validation before making crucial decision to proceed with investigation.

Background to the Study

Analysis phase determine significance, reconstruct fragments of data and draw conclusions based on evidence found. There are numerous forms of exploitation that analysis phases (Aymen, 2020). (Dalal, 2021), network forensic information can also convey insufficiencies in the current sensors network security tools. These challenges makes existing analysis phase network forensic framework not able to present and provide admissible evidence. Network security tools can be applied to network traffic and data fusion performed on various output values generated. Many models have been proposed for data fusion of intrusion detection data. They are mostly based on confusion matrix theory of evidence. The models are surveyed to obtain the direction for data fusion in the context of network forensic analysis. In the same area, (Tiffanie,2021) Machine learning or artificial intelligence for sensor data fusion model. Pieces of evidence from heterogeneous defence systems are fused or combined to detect the attacks for anomaly detection and localization. Yuan (2021). Data fusion techniques effectively combine evidence from multiple sensors or a single sensor used in multiple places. A