Network Packet Management Optimisation for Business Forensic Readiness

Bryce Antony Coad

A thesis submitted to the graduate faculty of Design and Creative Technologies
Auckland University of Technology
in partial fulfilment of the
requirements for the degree of
Master of Information Security and Digital Forensics

School of Engineering, Computer and Mathematical Sciences

Auckland, New Zealand
2017

Declaration

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the qualification of any other degree or diploma of a University or other institution of higher learning, except where due acknowledgement is made in the acknowledgements.

....

Bryce Antony Coad

Acknowledgements

I would like to thank Auckland University of Technology and my supervisor Dr Brian Cusack for sharing his experience, along with continual support and advice. I have learned a great deal from Dr. Cusack. I am grateful for his supervision.

I also thank Dr. Alastair Nisbet, MISDF program leader at AUT, for accepting me into the brilliant MISDF course.

I would like to thank Ramon, BJ and Brian for their help with acquiring the hardware I used during the research and putting up with my constant sojourns into their workshop on the hunt for equipment.

I am indebted to Kirk Thomas at NetScanTools.com for supplying a full license to the NetScanTools Pro suite of network tools, which formed an integral component of this thesis.

Thank you to my friend Jason, I would not have been able to complete this thesis without your support.

Finally, and most importantly, I would like to thank my son, Samuel. His understanding and patience is beyond words.

Abstract

The acceptability of evidence in court is dependent upon stringent criteria for its admissibility. One of the leading guidelines is the Daubert criteria which asserts five requirements for admissibility. These criteria assert scientific principles that must be complied. The focus on the scientific testing of Digital Forensics tools is often glossed over by accepting commonly used proprietary tools without delving into their performance. In particular, Daubert states that the known error rates of a scientific procedure must be published, and that the scientific procedures must be independently tested. In this thesis, I am concerned about the satisfaction of Daubert's key points when collecting digital forensic information that is evidentially sound:

- The scientific procedure must be independently tested.
- The scientific procedure should be published and subjected to peer review.
- Are there standards and protocols for the execution of the methodology of the scientific procedure?
- Is the scientific procedure generally accepted by the relevant scientific communities
- Is there a known error rate or potential to know the error rate associated with the use of the scientific procedure (*Daubert v. Merrell Dow Pharmaceuticals*, 1993)

Without known error rates and falsifiability criterion, digital evidence should not be in courtrooms. In this thesis the relevant literature and conceptual scope of the problem is explored theoretically, and then network tools tested empirically. The thesis is an initial investigation into whether there are error rates above zero when a network tool is used for evidence collection. The contribution is to both the digital forensic investigation community as well as to the law practitioners and judicial profession by demonstrating a potential problem with network forensic data. The performance of two tools was tested by subjecting them to increasing packet loads and measuring their performance. The results were then measured against the baseline of expected outputs and the differences noted. These differences were then used to generate an error rate in the form of a percentage. As part of the methodology the assertion was that, the tools would perform without error. Therefore, the research question is:

Research Question: *Can the Network Management System and the Network Packet Capture tool, achieve zero errors for digital forensic purposes?*

Thus, the research goal of this thesis is to determine whether a computer system can capture relevant information systematically and comprehensively for post incident forensic presentation that complies with legal requirements of completeness. The findings demonstrated that the popular network tools selected had significant error rates that are not published. The error margins indicate that a large number of packets that are potentially evidential are lost as the work rate increases. This will in turn affect the validity of data presented as evidential.

I recommend that the concepts developed within this thesis be expanded and codified through future research. That a professionally accepted assertion test bench be developed and test case methodology procedures formulated. Thus, transference can be assured, where tools tested under many scenarios through many assertion tests can provide an acceptable level of assurance. Therefore, the desired outcome is that error rates can be determined by the forensic examiner as a standard part of digital network forensic readiness and provided along with evidential data.

Table of Contents

Table of Figures

List of Tables

Chapter 1
Introduction

1.0 INTRODUCTION

The purpose of the experimental research of this thesis is to test the capability of two selected network digital forensic tools. The experiment will compare a baseline of data generation to a received or captured data value upon an experimental networked hardware platform. The independent variable of the experiment will be defined as number of packets generated in the dependent variable will be defined as the number of packets received. The implication of these findings will be used to test for forensic readiness in terms of the Daubert criteria.

Section 1.1 will begin with a description of the motivation behind the research presented within this thesis. Then, Section 1.2 will address the research approach and the findings generated from the research. Finally, Section 1.3 will describe the structure of this thesis giving a description of the contents of each chapter.

1.1 MOTIVATION FOR RESEARCH

For the business environment, a computer workstation is no longer a separate and contained digital entity, but rather has linked to other entities in the form of a networked infrastructure. The field of digital forensics, however, has been focused upon static data collection. The development of the field of digital forensics has been unstructured when compared to the development of other forensic sciences or even scientific development as a whole. This brings into question the validity of digital forensic tools when they are adapted for use in the corporate networked environment as these tools have not been tested as part of a scientific development.

A fundamental precept of the philosophy of scientific development, is that the proposed scientific method be subjected to tests of falsifiability. This is where a scientific investigation makes repeated honest attempts to falsify a theory. Thus falsifiability can be considered the demarcation criteria to distinguish scientific from the unscientific process. Scientific method calls for a prediction to be made, and then observation and experimentation be taken that either proves the prediction to be false or not false. Digital network forensic tools in general have not been subject to this test of falsifiability. Therefore, this thesis will present an assertion statement of falsifiability.

Digital forensics is at the intersection of computing and Law. The objective of the forensic examiner is to generate a forensic report containing specialist expert evidence designed to be presented in a court of law. This means that the forensic examiner must continually consider whether the evidence collected meets standards of evidence admissibility. Standards for admissibility of scientific evidence have been developed, the most significant of which has been determined to be the Daubert standard. The standard sets out five points of consideration in which a judge can determine whether the evidence is truly scientific or to be considered junk science and therefore inadmissible.

This legal requirements present a problem to the network digital forensic examiner, as the Daubert standards include requirements for complete datasets. Also the results must be reproducible and must have published error rates. Therefore, the problem can be defined as a legal requirement to perform scientific methods of evaluation. For example, to generate an error value for a network digital forensic tool when utilised for evidence collection. This indicates the requirement for a taxonomical approach where many different digital forensic tools are tested and evaluated.

There is debate within the digital forensic community of how to determine the tools that might be appropriate for the moving data scenarios, of the network environment. The volatile nature of network data means that any data that is not captured is forever lost. The generation of a network data capture information file is presumed to consist of a complete dataset. Should the data set be incomplete, it is considered deficient evidence. The problem is how to assign evidential value to data gathered from a networked environment. Therefore, the presentation of the information generated by the forensic examiner should include an error rate as is expected by conformation to the Daubert standards. Thus, a logical and standardised digital tool testing system is required to assign error rates to Networked Digital Forensic Tools. This will assist in the determination of the most appropriate tool for the selected network environment.

A falsifiability statement is made as an assertion for testing. The research question is selected to guide the research and the methodology development to gain evidence for deciding outcomes. Two sub- questions, and two hypotheses are used to provide greater resolution of the context and evidential scope.

Therefore, the primary falsifiability prediction tested by the research methodology evidence is:

Falsifiability Statement: *The Network Management System and Network Packet Capture tool will achieve zero errors when tested within a networked digital environment.*

Thus the research question proposed in this thesis is:

Research Question: *Can the Network packet **Management** System and the Network Packet **Capture** tool, achieve zero errors for digital forensic purposes?*

Thus, the research goal of this thesis is to determine whether a computer system can capture relevant information systematically and comprehensively for post incident forensic presentation that complies with the legal requirements of completeness.

Sub-question 1: *Can a network Packet **Management** System and Network Packet **Capturing** tool provide legally acceptable forensic evidence?*

Sub-question2: *Will the selected Network Packet **Management** System and network Packet **Capturing** tool perform without error under load?*

The hypotheses generated through the appraisal of the research question proposed are:

Hypothesis 1: *That a commonly used, well established and professionally acceptable Packet **Capture** Tool will perform under high levels of network stress without error.*

Hypothesis 2: *That a commonly used, well established and professionally acceptable Packet **Management** tool will perform under high levels of network stress without error*

This research aims to provide a practical demonstration of the capabilities of a network packet Management system and a packet capture tool when presented with a known, predetermined dataset.

1.2 RESEARCH APPROACH AND FINDINGS

The research approach applied in this thesis was to conduct a search of professional publications which, upon perusal generated a literature review documenting information pertinent to the research topic. The literature review was then used to identify problems to be addressed by the research methodology. The research methodology was determined by identifying studies of a similar nature to the proposed investigation. Five studies were examined to determine processes and procedures relevant to the proposed investigation. The methodology involves several steps. First an assertion test bed was created based on an Ethernet segment, with standardised hardware, and standardised software. Then two selected applications, a network packet **Management** tool, in the form of an Intrusion Detection System (IDS), and a Packet **Capture** Tool, were evaluated. Several test cases were developed, each of which was used to evaluate a networked digital forensic tool in the following manner: A known dataset baseline was generated and each tool was set to either report on or capture the generated dataset. The resultant datasets generated by the networked digital forensic tool was then compared to the baseline. Nine test cases consisting of two iterations each were designed. The first set of three test cases evaluated the IDS by generating 45,000 ICMP echo request packets, transmitted at a rate of 1 ms 3 ms and 5 ms and comparing the Intrusion Detection System (IDS) statistics. A second set of three test cases tested the Packet Capture tool by generating 1024, 2048, and 4096 ARP packets and comparing the packet capture results. The third set of three test cases tested the packet capturing tool by generating 256, 512, 1024 ICM packets and comparing the packet capture results. It was expected that a complete dataset be generated by the network digital forensic tool under evaluation, would be complete and without errors.

Upon application of the methodology, the findings were then examined and applied to the sub-questions. The answer to these sub-questions were then used to answer the research question. The findings, when assessed and tabulated demonstrate a significant error rate for 16 out of the 18 test case iterations. The two, zero error rate, results were for single iterations of different test cases. Thus, it was determined that the findings resulted in a fail for all nine test cases evaluated.

Therefore:

Hypothesis 1: *That a commonly used, well established and professionally acceptable Packet* ***Capturing*** *Tool will perform under high levels of network stress without error.*

Result: *FAIL*

Hypothesis 2: *That a commonly used, well established and professionally acceptable Packet* ***Management*** *tool will perform under high levels of network stress without error*

Result: *FAIL*

Sub-question 1: *Can a Packet* ***Management*** *System and Network Packet* ***Capturing*** *tool provide legally acceptable forensic evidence?*

Result: *NO*

Sub-question2: *Will the selected Network Packet* ***Management*** *System and Packet* ***Capturing*** *tool perform without error under load?*

Result: *NO*

Research Question: *Can the Network* ***Management*** *System and the Network Packet* ***Capturing*** *tool achieve zero errors for digital forensic purposes?*

Result: *NO*

Falsifiability Statement: *The Network* ***Management*** *System and Network Packet* ***Capturing*** *tool will achieve zero errors when tested within a networked digital environment.*
Result: *FALSE*

The findings of the evaluation of the tools demonstrate the need for this research and enforceable criterion for tool testing. The assignation of the level of the errors is of concern, and the existence of any errors is disturbing. A continuation of this research with a taxonomical approach, where each potential network digital forensic tool is listed and evaluated upon a standardised assertion test bench will benefit all professionals within the network digital forensic community. This not only includes the forensic examiner but also practitioners of the law and those being judged. Thus it is a recommendation for future research that comprehensive tests of digital forensic tool be carried out in order to establish error rates which can be presented along with evidential datasets to a court of law.

Chapter 2

LITERATURE REVIEW

2.0 INTRODUCTION

The literature was selected primarily through conducting key word searches, using the Scholarly Commons interface on the Auckland University of Technology (AUT) Library portal. The initial term sets that were searched were generalized, such as ‘Forensic Readiness’ and ‘Digital Law.’ The search term sets were then refined to become more specifically related to the core topics of the research, such as ‘Ethernet Frames Forensics’ and ‘MAC Address Security.’ Many literature items were reviewed, with a number selected by topic relevancy and added to EndNote. The AUT library was then researched for relevant literature, with books selected for withdrawal and reading. The majority of the classical science development literature was downloaded as electronic books via the AUT Library web portal, by using a title search. Finally, a limited number of industry specific business websites, such as cisco.com and fireeye.com were visited and perused for relevant documentation.

Digital forensics as a discipline has encountered many technological changes since early 1980 and developed as a science in an unorganized and disorderly fashion. The works of Popper from 1934 and Kuhn from 1970 have influenced the rational development of scientific progress from conceptual and philosophical postulations that are still valid in the 21st century. Many of the tools and procedures accepted as valid within the field of digital forensics have not been subjected to tests of falsifiability nor have they been assessed for error rate determinations.

The technological changes since the 1970’s has been of such magnitude that new lexicons are being continually updated and paradigms are therefore shifting. Businesses are under legislative pressure to maintain a record of the events that occur within their networks, which is in a business’s best interest, and this forms the basis of being forensically ready. The literature review begins with philosophical points, and then moves to the specifics of packet management to optimise businesses network data collection. Effective forensic readiness, with results that are legally acceptable as evidence is also reviewed.

Section 2.1 reviews the formative philosophical views upon scientific development and then looks into investigation model development within the digital forensic discipline. Section 2.2 shows that the development of the legal system has proceeded in a rational manner, investigates the intersection of legal requirements and computing, the principles of digital evidence, and then discusses the Daubert standard and its importance in determining acceptable expert testimony. Section 2.3 explores the subject of forensic readiness, considering the objectives identified by Tan in 2001 and the necessity of including a pre-incident plan as the key concept of digital forensic readiness. Section 2.4 examines the business requirements for digital forensic readiness, identifying the lack of a single framework, and then reviews the ISO/IEC 30121 digital forensic risk mitigation standard, executive order 13636 aimed at improving the cyber-security of critical infrastructure, and the ACPO guidelines providing a view of the fundamental forensic focus for business. Section 2.5 inspects optimization aspects for business looking at the issues of rapid growth in the volume of data for pre-emptive collection. Section 2.6 network forensics, identifies the difference between information security and incident response, and then discusses live data investigations. Section 2.7 delves into the technicalities of business network infrastructure, with discussing the OSI reference model, how the model relates to Ethernet, data encapsulation, and address resolution protocol that shows the importance of layer 2, and the MAC address to the digital network forensic investigator. Section 2.8 looks at the challenges that face the forensic examiner, such as information reconstruction, and the necessity for running a hybrid investigation and then explore a potential solution then describe some security considerations. Section 2.9 summarises the issues and problems revealed from this investigation of current literature, and 2.10 concludes the review.

2.1 DIGITAL FORENSICS AS SCIENTIFIC KNOWLEDGE

Digital forensics has not developed as part of a scientific process, but has developed in an ad-hoc manner. Section 2.1 provides an overview of digital forensic development as contrasted to scientific theory development. Theoretical scientific development by Popper and Kuhn are discussed in the first two subsections, which then lead on to an exploration of digital forensic investigation model development.

2.1.1 Popper’s Scientific Theory

Popper’s scientific theory holds that scientific method is an iterative process of prediction, observation, analysis and model building to support the building of knowledge. Popper’s works on the methodology and philosophy of science were widely influential on determining the difference between science and non-science Popper postulated there is a clear criterion that provides a demarcation of scientific theories: the ability to be proven wrong by predicting future observations to be rendered true or false (Popper, 1959).

Popper postulates that scientific theory must be:

- Falsifiable
- Logically consistent
- At least as predictive as other theories
- Observation of tests of falsification must not falsify the predictions

He argues that non-scientific theories are distinguishable from scientific theories in that scientists investigating a theory make repeated, honest attempts to falsify a theory, in what amounts to a willingness to be wrong. This is different from pseudoscientific theories, which take measures to make the observed reality fit the predictions of the theory. Thus, falsifiability is the demarcation criterion that distinguishes the scientific from the unscientific and, it is only after numerous attempts of falsification that researchers can become confident in a theory.

Popper's theory views scientific progress to develop along an evolutionary model, where observations and experimental outcomes selecting against unfit theories by falsifying them. This allows for scientific progression and development that integrates new ideas and theories that incorporate technological developments.

An example of Popper's falsification theory is to consider the difference between the theories of psychoanalysis as postulated by Freud, and Einstein's theory of general relativity. A rough summary of these two theories are:

General relativity postulates that the observed speed of light in a vacuum will be the same for all observers, regardless of what velocity or which direction these observers are themselves moving. General relativity can be applied to cases where acceleration or gravity plays a role, specifically by treating gravity as a sort of distortion or bend in space-time created by massive objects.

Psychoanalysis theory holds that human behavior is driven at least in part by unconscious desires and motives. For example, Freud posited the existence of the id, an unconscious part of the human psyche that aims toward gratifying instinctive desires, regardless of whether this is rational. However, the desires of the id may be overridden in certain circumstances by its interaction with both the self-interested ego and the moral superego.

Both theories account for previously observed phenomena; for example, general relativity allows for an accurate description of the observed perihelion of Mercury, while psychoanalysis entails that it is possible for people to act in ways that are against their own long-term best interest.

Popper argues, however, that general relativity is scientific while psychoanalysis is not. The reason for this has to do with the falsifiability of Einstein's theory. In 1919, Eddington performed an experiment that observed the degree which light from distant starts shifted whilst passing the sun during a solar eclipse, the predictions of which differed between general relativity and the then dominant Newtonian mechanical theory. Whilst the observations by Eddington suggested that the Einsteinium postulation was not false, and that Newtonian mechanics were false in this instance, it was not the hypothesis being proven false or not-false, it was the fact that these theories could be tested for falsibility empirically that allowed general relativity theory to be labeled scientific.

This is in contrast to Freudian psychoanalysis theory, which does not make any predictions that allow for tests of falsification. The problem with theories such as these is they are compatible with every possible observation, being unable to rule out any possible human behavior. An example of which is two men, one who pushes a child into the water with the intention of drowning the child, and another who jumps into the water to save the child. Popper notes that both these seemingly contradictory actions can be explained by psychoanalysis, either as a repressed component of the id in the first case, or a successful sublimation of the same desire by the ego and superego in the second case. This shows how psychoanalysis can be used to explain any form of human behavior, and in turn, prevents the formulation of any crucial experiments that may falsify psychoanalysis.

This falsification process has been missing from the scientific development of digital forensics, where very few forensic processes, tools and procedures have been subjected to tests of falsification, which can be seen as a cornerstone of scientific rigor. Utilising tools and procedures to formulate expert opinions for digital forensic purposes must be based upon properly documented sources and are required to withstand judicial scrutiny. This means that the sources of information used to uncover evidence should have independent testing, be subject to peer review and, importantly, have standards, and error rates that are acceptable to the scientific community. Therefore, the development of digital forensic technology is in danger of becoming pseudoscientific or even metaphysical in theoretical development.

2.1.2 Kuhn’s Progress of Scientific Knowledge

Kuhn, in *Structure of Scientific Revolutions* (1970) presented a different viewpoint from Popper, introducing notions such as paradigms, scientific revolutions and incommensurability. Kuhn argued that the idea of science is an activity of puzzle solving, which operates under paradigms and these paradigms become discarded when they fail to respond appropriately to challenges of a rival paradigm. Thus, scientific progress can be seen as a series of paradigm shifts, rather than progressing along a gradual and cumulative path. Whilst this process is one of conceptual change, involving the philosophy of scientific development and lexical change, the progress of digital forensics can be seen to have moved in such a fashion. Therefore, progress of digital forensics as a science has taken the form of an ever-improving, technical puzzle-solving ability that is bounded by traditional standards of success and failure.

The development of this structural approach is that science is fundamentally a social undertaking, and when this is applied to digital forensics, individuals are able to arrive at different viewpoints concerning the level of difficulties collectively faced when working in a common research tradition. In terms of conceptual risk distribution, it is the development of possible alternatives, and even the possibility of radical conceptual revision that will provide a level of risk mitigation. This viewpoint of radical change is particularly applicable to the scientific development of digital forensics and is different from the majority of other scientific disciplines, where progress is based on normal cumulative approaches. Scientific advancement in digital forensics involves discoveries that are far more problematic, sometimes changing the field radically, resulting in what Kuhn termed as a paradigm shift (Kuhn, 1970).

Therefore, as postulated by Kuhn (1970), digital forensic as a science cannot develop by simply adding to what is already known. There are already difficulties describing new concepts in the vocabulary of the old lexicon, where digital forensics must embrace legal, social and structural requirements rather than just provide event detection and accountability. Digital forensic software tools are proprietary, vendor specific developments, and with technology outpacing development, the tools that have become widely used are accepted as sound merely because they still exist. Kuhn’s viewpoint is persuasive with respect to digital forensics developing as a science, where the development has incorporated complex technological and lexicological change. This can be seen as evolutionary. The tools that are accepted have been deemed to be fit, due to survival, rather than surviving because they are the fittest. This can be explained as having adapted to paradigm shifts caused by radical technological changes, which are the driving force behind developments, examples of which are the vast increases in data volumes and the transmission of data across interconnected computer systems.

2.1.3 Digital Forensic Investigation Model Development

As a point of reference, Lee (2001) presented *The crime scene investigation model*, which presented a scientific and methodical way of investigating a physical crime scene. The model consisted of the following four stages:

- Recognition: crime scene items or patterns are identified as being potential evidence.
- Identification: potential evidence is classified into forms such as physical, biological or chemical. No single standard of evidence comparison is conducted.
- Individualisation: evidence is examined for uniqueness in order to link the evidence to an individual or event.
- Reconstruction: the previous stages and other relevant information is evaluated, producing a detailed report about the events and actions discovered at the crime scene (Lee, 2001, p. 17).

Many aspects of this model, even though it refers only to a physical crime scene investigation, can be applied in the search for digital evidence at an electronic crime scene investigation. For example, whilst Lee's model refers to the physical element of the investigation and does not deal with preparation for collection of information, the first two stages are still relevant for the modern digital forensic investigator in terms of potential evidence recognition and classification. These are important aspects of optimisation, which is a key concern when preparing to collect electronic evidence from a source such as network activity which can consist of vast amounts of data .

Lee's model was further developed to include a digital forensics process in the *digital forensic research workshop* (Palmer, 2001). This model of Digital forensic investigation suggests a seven-step process with a number of actions to be performed on each step.

- Identification: Event detection, resolve signature, profile detection, anomaly detection, system monitoring, audit.
- Preservation: Case management, imaging, chain of custody, time synchronisation.
- Collection: Preservation, approved methods, approved software, approved hardware, legal authority, loss, loss, compression, sampling, data reduction, recovery techniques.

- Examination: Traceability, validation, filtering, pattern identification, hidden data discovery and extraction.
- Analysis: traceability, statistics, protocols, data mining, timeline, links.
- Presentation: Documentation, expert testimony, clarification, mission impact, recommended countermeasures, interpretation.
- Decision: Prosecution, employee termination, continued observation and analysis (Palmer, 2001, p. 17).

This model was presented at The Digital Forensic Work Shop in 2001, and has become a point of reference to the modern digital forensic examiner. The steps are included in many of the recent models and thus the model has become the basis of current work. As the model does not include any comprehensive explanation of the actions to be performed, but rather a list of techniques, it is difficult to use this model directly in a real world digital forensic investigation. The identification and collection stages of Palmer's model are of note as they pertain directly to the optomisation quandary, and the preservation stage is essential in order to maintain legal validity.

Rogers et al. (2006) further developed Palmer's model with an intention to identify, examine and interpret digital evidence immediately upon investigation. This model consists of six main phases.

- Planning: preparations made prior to an investigation.
- Triage: classification and identification of evidence, according to importance and volatility.
- User/usage profile: this concentrates on users activities and could also include observed physical activity that may link one individual to a single workstation.
- Chronology: this will determine a chronological sequence of crime events and may indicate modification, access and creation time of gathered data.
- Internet: examination of Internet services.
- Case specifics: Examination focus adjustment to align with the specifics of the case (Rogers, Goldman, Mislan, Wedge, & Debrota, 2006, p. 30).

This model was developed specifically for digital forensic investigation. It begins with the planning stage where preparation for an event is seen as naturally occurring pre-incident. This did not extend to collection of data in a networking infrastructure, but instead focuses on the immediate gathering and imaging of static data after an incident has been detected as being a priority. It is difficult to apply these stages to the forensic examination of a network infrastructure where the evidential information is volatile and certainly no longer attainable post-event. Yet the concept of triage is certainly important, and by identifying information that is acutely relevant and discarding any data deemed irrelevant, the volume of data collected can be reduced. It is important that triage be a core concept when developing a networked, forensically ready system for business use.

Yusof et al., (2011) examined existing models to determine common phases to develop a generic investigation model, which consists of the following five generic phases.

- Pre-process: gathering necessary approval from relevant authorities. Tool preparation, forensic examiner readiness.
- Acquisition and preservation: identification: collection: storage: transportation: and preservation of evidential data.
- Analysis: examination and identification with the goal to discover person or persons responsible for any criminal act.
- Presentation: report formation and presentation of findings.
- Post-process: utilising the results of the investigative process to improve system security as well is future investigations (Yusoff, Ismail, & Zainuddin, 2011, p. 29).

Although this is useful to the forensic examiner, this appears to be more of a framework. The model and each phase of the processes too generalised, making it difficult to implement as part of a real world process. The most significant contributions to the modern digital forensic examiner is the introduction of a pre-process stage, where the model notes that this phase relates to the task which need to be addressed prior to the actual investigation. The model viewed this phase relating to seeking authorisation, but the relevance to networked infrastructure can be seen if this stage is extended to include the second stage of this model. Identification and acquisition of data, when integrated into a pre-process stage, form the basis of digital forensic readiness.

Each of these models, whilst including important considerations to be taken by the forensic examiner, neglect to identify a substantial part that is essential to any networked digital forensic investigation, the gathering of potential evidence before any incident takes place. This is the main problem identified with any network forensics investigation, namely which data to gather, where to gather it from, and how much volume is necessary to provide the investigator with enough information to form legally acceptable conclusions.

2.2 DIGITAL FORENSICS

Digital forensics, as a discipline is moving out of infancy towards maturity, which is highlighted by changes in the industry. There is a call for reform within the discipline to align with more traditional forensic sciences and their associated levels of scientific rigor. There is a need for validation through accreditation to align this developing part of the digital domain with science and jurisprudence. Section 2.2 begins by discussing the intersection between technology and legal jurisprudence, then investigates admissibility standards for specialist scientific evidence, the principles of digital evidence and includes with a discussion of the Daubert test and how this relates to digital forensics (Caloyannides, 2006).

2.2.1 Intersection of Law and Computing

Digital forensics occupies the intersection between law and computer technology. Currently the most prevalent forensic objective of an organisation is one of regulatory compliance. At a business level this can actually be seen as non-compliance avoidance and is rarely seen as a combination of technical resource, and legal evidence management. The issue that an approach of mere compliance engenders is that regulatory compliance alone may defeat the purpose of digital forensics, by gathering data that contains deficiencies at either the network or legal level. These deficiencies present a problem at the start of an investigation where it is unknown which evidence will eventually be used and therefore what information is useful. This is a problem where the minimum information logging required to ensure regulatory compliance may produce little forensic data (Elyas, Maynard, Ahmad, & Lonie, 2014).

As digital evidence is required to satisfy exacting requirements if it is to be presented in court, a system must be in place to maximise the potential to produce forensically sound digital evidence. Therefore, the employment of qualified individuals, the utilisation of appropriate technologies and the identification of system architecture must be undertaken from both a technical as well as a legal viewpoint to maximise forensic evidence potential. This formula enables an organisation to be ready for legal proceedings in both criminal and civil courts as well as providing valuable technical networking information (Nikkel, 2005). The benefits of this approach, where digital information is collected in a structured and formalised evidential manner, are a reduction in investigation costs and minimisation of investigatory business disruption resulting in an improvement of policy enforcement. This means that business down-time can be reduced, which mitigates risk to the enterprises cash flow and provides the potential of policy reinforcement through successful prosecution. An added benefit is incident impact on the business concerned can be evaluated and security improvements implemented with minimal additional costs.

There are shortcomings with the process of digital forensic evidence collection where the evidence that is collected is binary data in the form of bits. This binary data, which consists of ones and zeros, can be altered by an expert to either change the evidence leaving no trace, or alter the identity of the owner, pointing the evidence in the wrong direction. This means that the forensic investigator can only determine the evidence that is available at the time that the evidence was acquired. This could cause legal issues with requirements of non-repudiation and completeness, causing vital evidence to be rejected in court. Therefore it is important that the information that is collected is not open to third party access by imposing security measures, that images, that are an exact copy of the captured data are produced and the original information is preserved as evidence (Caloyannides, 2006).

This data format has been traditionally associated with magnetic media, where the data is stored on some form of semi- permanent system such as a hard drive. The original evidence in this form can be easily imaged and kept securely, ready for presentation as legal evidence. Network digital forensics however can be shown to be more complicated as all the information is volatile in nature, and can be considered data in motion and evidence captured from a corporate network could be considered trace evidence. There are calls for a new legal approach to the analysis of forensic trace evidence, calling into question the validity of such evidence. There is increased emphasis on standardisation of scientific processes to meet new technological advances and the impact on traditional forensic analysis (Stoney & Stoney, 2015).

There are also legal standards that apply to the collection of digital evidence. In the USA this can include concepts such as the first amendment, the right to free speech, and the fourth amendment, the right to be free of unreasonable search and seisures. This must be determined at the IT governance level of a corporation's policy, whereby each employee is informed through the administration of an employment contract that any data transmitted from a computer connected to the corporate network is collected as a matter of course. Thus any employee can be seen to have consented to the seisure and search of any data located anywhere on that corporate network (Schwerha, 2004).

In New Zealand (NZ), it is an offense under section 252 of the NZ Crimes Act (Crimes Act 1961), for anyone to access any compute system without authorization. This ruling does not apply if the access is under the execution of an interception warrant or search warrant. Criminal law in NZ contains a moral element, where an offence is committed when criminal intent is present. Therefore it will always be prudent to gain clear permission from an appropriate authority in writing before collecting digital evidence from a computer system in NZ.

2.2.2 Admissibility Standards

A problem with the acquisition of digital forensics information is that such information needs to satisfy admissibility standards for evidence substantiation. The admissibility of this electronic evidence will often be left to the judge to determine in a court of law, therefore, many issues must be considered when giving digital evidence information. These range from the relevance, authenticity and completeness of the evidence, to the qualifications, knowledge, skill and experience of the expert witness presenting the evidence. Difficulties arise with the reconstruction of a multistep, multistage attack being presented as evidence that must be integrated with legal acceptability standards (Changwei, Singhal, & Wijesekera, 2014).

Evidential relevance and credibility can be called into question if there is evidence that has been intentionally destroyed or is missing. Therefore any forensic readiness policy must determine that the information collected is to comply with standards of completeness and non-repudiation. With the progressive nature of the threat landscape, where infractions are becoming increasingly more sophisticated, there is very little information to formalise legal acceptability of complex multistage attack evidence for prosecutorial use. Another issue is the complexity and multiplicity of different computer systems and attack techniques that creates difficulties in providing a universally accepted way to credibly and accurately assess digital evidential information. Practical methodology experimentation in different areas of digital forensic application is an important part of digital forensic research development rather than with the prevailing approach which views digital forensics information as only determined to be evidential in the final or reporting phase. It may not matter what the report indicates, if the information that the report is based upon cannot be presented along with the report (Karie & Venter, 2014).

Nikkel (2014) in a digital investigation editorial suggests developing threat intelligence as a useful way to identify new vulnerabilities, newly discovered threats and identifying stolen data. This includes investigating IT infrastructure and developing incident response and investigation processes in advance. An issue identified in this editorial is the difficulty in identifying which information needs to be captured proactively, and problems with how the digital information should be captured for post-mortem investigation and event reconstruction. Another problem is that this digital information must be captured before, during and after an event. The editorial also emphasises that staff, training, external support and tools all need to be in place before any incident, as part of a corporation’s security policy. Instead of focusing on evidential digital information, forensic readiness, as a whole, accentuates the entire digital forensic process through the addition of an anticipatory dimension. The importance of being comprehensively forensically ready is heightened with organisations being subject to increased levels of regulation.

Two levels of forensic readiness can be identified, operational readiness and infrastructural readiness, where training and the provision of forensic equipment can be determined as operational readiness, and infrastructural readiness ensures organisational data is preserved. Therefore, training, planning, policy and monitoring are necessary to improve forensic readiness which should also include cultural and governance aspect incorporation. There are problems with this approach in that whilst there are a multitude of incidents that affect all industries such as denial of service, malware, and spam attacks, there are many potential incidents that are industry specific. Examples of these industry specific attacks include copyright violation in the entertainment industry, phishing attacks in the banking industry and intellectual property theft in the pharmaceuticals industry (Elyas, Ahmad, Maynard, & Lonie, 2015; Nikkel, 2014).

2.2.3 Principles of Digital Evidence

Analysis of digital forensics information is very similar to traditional forensic analysis, for example, firearm analysis can prove that a bullet was fired from a specific weapon without the weapon itself being introduced as evidence. Thus data trails, meta-data, and time/dates stamps can establish timelines to correlate important events.

Some important aspects of a digital forensics report are:

- The conclusions formed within the report must be reproducible by independent third parties (Philipp, Cowen, & Davis, 2010).
- Opinions contained within the forensic report must be based on properly documented digital sources (Bates, 1998).

Therefore, it is important that independent third parties be able to replicate conclusions formed from analysis of the digital forensics information (Philipp et al., 2010). It is also important that any information must be documented outlining steps undertaken by the examiner, and raw data must be available for third-party analysis (Bates, 1998). It is therefore a problem if raw forensic data is not available or there is an inability to replicate conclusions formed within a digital forensic report. This may cause a report to be viewed as less dependable, or even inadmissible, as the accuracy or reliability of methodology is unable to be determined. It is also important that any digital forensics report not contain superfluous information and that data collected is limited by relevancy, budget and time constraints. These constraints, which can be significant, will have an impact on what data is found and the inferences that can be drawn.

2.2.4 The Daubert Test

Daubert versus Merrill Dow Pharmaceuticals Inc sets forth a five pronged standard for the admissibility of scientific evidence in a federal court. As the Daubert standard can be applied to any scientific procedure it is a particularly important step in the production of digital forensics evidence.

The Daubert standard applies whenever scientific procedure is used to prepare and uncover evidence and comprises the following five rules:

- The scientific procedure must be independently tested.
- The scientific procedure should be published and subjected to peer review.
- Is there a known error rate or potential to know the error rate associated with the use of the scientific procedure?
- Are there standards and protocols for the execution of the methodology of the scientific procedure?
- Is the scientific procedure generally accepted by the relevant scientific communities (*Daubert v. Merrell Dow Pharmaceuticals*, 1993)?

The standard provides judges a set of guidelines for the objective acceptance of scientific evidence yet there are no uniform sets of standards to gauge the competency of a digital forensics examiner or the effectiveness of digital forensic analysis tools. Limitations of these forensic analysis tools should be accurately qualified and industry-standard Best practices along with the application of current technologies should be addressed. This can prove problematic with the generation of an effective forensic report, which should be based on a logical and cohesive framework rather than relying on technical details alone.

Commercial software vendors of specialist digital forensic analysis tools have replaced the manual analysis process, resulting in reduced level of knowledge required for forensic examination which has lowered costs but increased the susceptibility of determining false conclusions. Therefore, an evaluation framework is required to produce an effective forensic report, the basis of which begins with the manner in which the evidence was acquired. In a business network environment this will be from live acquisition and therefore the digital forensic report must document the steps undertaken by the examiner with sufficient detail to allow an independent third party to replicate the conclusions. The problem with this approach is the provision of the forensic image formed from the capture of network data can be prohibitively large. Thus any report with a conclusion that is not reproducible utilising the captured network data may be granted little credence (Roussev, Quates, & Martell, 2013).

The scope of any investigation is generally limited by budgetary and time constraints, along with relevancy. This is a problem with business forensic readiness as it is difficult to determine what is relevant and what is not. There are also problems with information overload, which can reduce the provision of a cohesive and logical framework. Information overload is where the amount of information captured overwhelms the conclusion contained within the report, and this will be the rule rather than the exception where corporate network information is captured. Although there are commercially available tools that purport to handle issues of this nature, any tools that are used by the forensic examiner will also come under scrutiny. Therefore, the tools will not only need to provide replicable data but may be required to pass the items of the Daubert test, namely independent testing, peer review and error rates (Nikkel, 2005; Tan, 2001).

The requirement to provide a known error rate or potential to know the error rate associated with the use of the scientific procedure to comply with the Daubert test also complies with the requirements of falsifiability postulated by Popper as discussed in section 2.1.1. This requirement indicates that the principles of Popper from 1968 have been incorporated into the judicial standards to enable the elimination of pseudoscience from entering the courtroom as expert evidence therefore establishing that the evidence is scientifically valid.

2.3 FORENSIC READINESS

Section 2.3 provides an overview of digital forensic readiness, beginning with a definition of objectives and then investigates the necessity of implementing a pre-incident phase to be forensically ready.

2.3.1 Forensic Readiness Objectives

Tan (2001) identified two objectives for forensic readiness. These are to maximise incident evidence data usefulness and to minimise the cost required to produce the incident evidence data. In the document entitled forensic readiness, he indicates that the environmental complexities of business networks demand that the details be defined ahead of time. Whilst this may appear to be contra-indicative, the ability to determine evidential requirements pre-incident is at the heart of digital forensic readiness, maximising evidential digital information collection. Forensic readiness information, by its nature, is gathered as a pre-incidence phase rather than a post-incident response. The preservation of the data of attacking systems or malicious intervention means data must be collected in a timely manner before any volatile information is lost. Therefore, digital forensics focuses of the investigation of pre-incident gathering of digital information, after the occurrence of a security incident, to investigate suspected illegal or unauthorised activities (Tan, 2001).

2.3.2 Pre-Incident Plan

To be digitally forensic ready, it is necessary to have a pre-incident plan that determines digital evidence identification storage and preservation. As most security frameworks fail to prepare a pre-incident response phase, it is very difficult to research forensic readiness planning in depth. Forensic readiness can be seen as an essential part of an organisation's information security policy where the aim is to prepare for forensics capability by proactively extracting, collecting and maintaining digital evidence. Currently, there does not appear to be a universally accepted application methodology and yet there are many different approaches to Digital forensic investigation.

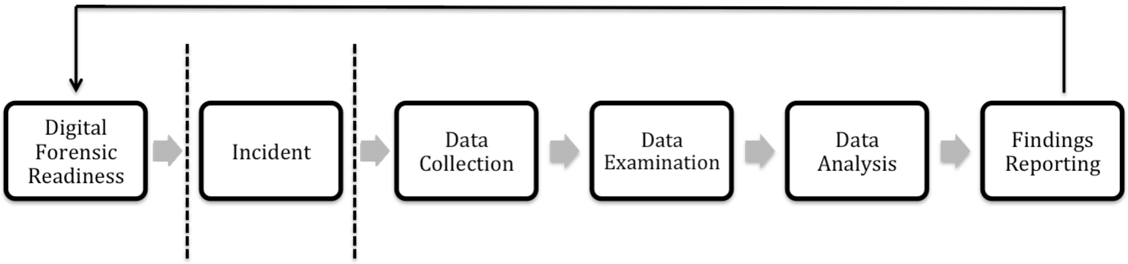


Figure 2.1: The digital forensic investigation lifecycle (Mouhtaropoulos, Li, & Grobler, 2014, p. 174).

It is this lack of commonality that adds to the complexity of any digital forensic investigation. As shown in figure 2.1 below, a digital forensic readiness policy must be in place before any incident occurs to provide an effective digital forensic investigation as an iterative process, where results from previous findings are integrated into future digital forensic readiness policy.

2.4 BUSINESS REQUIREMENTS

Section 2.4 investigates the digital forensic readiness requirements for business organisations. The section begins by looking at the background of proactive forensic capability, and then investigates ISO/IEC 30121:2014, a framework for governance of digital forensic risk, and then explores the NIST investigation into executive order 13636, aimed at improving the cyber-security of critical infrastructure in the USA. The APCO good practice guide from the UK is then examined, and the fundamental forensic focus for businesses is then investigated.

2.4.1 Business Forensic Capability

There has been little academic research into proactive business forensic capability and most digital forensic frameworks omit a pre-incident response phase. There is no single application or method that is universally accepted, and a business must choose between many different approaches to achieve a modicum of forensic readiness. Whilst there are frameworks and guidelines available from a multitude of sources, and there are few that cover the requirements for a pre-incident phase that emphasise data gathering designed for evidentiary purposes (Mouhtaropoulos, Dimotikalis, & Li, 2013).

2.4.2 ISO/IEC 30121:2014

The International Organisation for Standardisation (ISO) and the International Electro technical Commission (IEC) who form the worldwide standardisation specialised system have released international standard ISO/IEC 30121 which forms a framework which deals with governance of digital forensic risk. The framework is designed so that an organisation can strategically deploy their information technology systems to not only maximise evidential availability, cost effectiveness and accessibility effectiveness, but looks as well at prudent digital investigation preparation for organisations. The organisational governing body then establishes the framework as owners of risk, taking action to assure the strategic direction and the implementation of digital forensic risk governance within the organisation.

The digital forensic risk framework processes that are described within the standard consist of five categories:

- Archival strategy: comprehensive information property archival retention for maintenance of data integrity
- Discovery strategy: effective and efficient data retrieval capability for evidential presentation
- Disclosure strategy: establish information security and disclosure capability ensuring that disclosed information is auditable
- Digital forensic capability strategy: adoption of plans and policy to assure digital evidence preservation, along with access to appropriate digital forensic skills, assuring investigator integrity, expert independence and binary data as evidence
- Risk compliance strategy: adoption of strategic risk based on the application of risk criteria for digital evidence

As can be seen, the five main digital forensic process categories of the ISO standard 30121 are directed towards the maintenance and preservation of data as evidence. This directive shows the importance of regarding the digital forensic process for businesses and all information gathered as potential evidence, at every step of the process. The regard for evidential process and legal adherence must remain at the forefront of any business preparing for forensic readiness (ISO/IEC, 2014).

2.4.3 Executive Order 13636

In 2013, USA president Obama released an executive order, number 13636, aimed at improving the cyber-security of critical infrastructure. In particular, the executive order calls for the development of a framework that provides a flexible, performance based, cost effective approach. The framework will be designed to assist organisations mitigate and manage cyber-security risk. The executive order calls for a voluntary acceptance of the developed framework.

Whilst the directive is targeted on critical infrastructure that is vital to the United States, infrastructure that would have a debilitating societal impact on security, safety, national economic security or public health should it be compromised, the premises contained within can be adopted by any business, whatever the size of the organisation. The framework, which is being developed with industry collaboration, provides organisational risk management guidelines rather than a strict ‘follow the numbers,’ step-by-step checklist approach. Thus, it is upon existing best practices, standards, and guidance that the framework relies to provide assistance to cyber-security risk mitigation for any business (NIST, 2014). Attackers of Business network are tracked back through the application of forensic techniques where the ultimate goal is to gather enough evidential information to ensure successful prosecution. The focus of business forensic readiness should be upon the gathering of data that is evidentiary. (Pilli, Joshi, & Niyogi, 2010).

2.4.4 ACPO Principles

A computer-based digital evidence good practice guide has been developed by the Association of Chief Police Officers (ACPO) in the United Kingdom. ACPO determined that four principles are involved in the acquisition of electronic evidence:

- No data should be changed by the actions of the forensic investigator
- If original data is to be accessed, it must be performed by competent forensic investigators who must be able to explain the implications of their actions in evidence
- A complete record of all applied processes must be created, and this audit trail should enable a third party to achieve the same results
- The lead or charge forensic investigator must take overall responsibility for ensuring these principles as well as the appropriate legislative directives are followed

The ACPO guideline states that digital forensic evidence obeys the same rules and provisions that apply to documentary evidence. Thus the onus is on the forensic investigator as prosecutor to provide this evidence to court, presented exactly the same as it was collected before analysis. Care must be taken that any investigative analysis tools utilised do not alter any portion of the evidence. It may be acceptable to consider alternatives, such as selective copying, when data volumes prohibit a complete record being taken, but the forensic investigator must ensure that all relevant evidence is captured. It is also important that the evidence be preserved in such a way as to enable a third party to reach the same conclusions as the forensic investigator. The legal requirement demonstrates continuity and integrity of the evidence collected (Association of Chief Police Officers (ACPO), 2012).

2.4.5 Fundamental Forensic Focus for Business

The main driving force behind business network forensics is the increasing sophistication of cyber-attacks and the large number of security incidents that affect many organisations. Traditionally network security provides defensive approaches in the form of firewalls and intrusion detection systems, but these methods can only address attacks from a reactive perspective (Pilli et al., 2010). Criminal digital forensic investigations may however, be fundamentally different from business forensic investigations. The focus of a business investigation may be only to produce reports that are acceptable to the organisation, and the identification of an internal attacker may only result in employment termination rather than criminal charges (Lang, Bashir, Campbell, & DeStefano, 2014).

It would be unwise, however, to provide evidence that is any less rigorously tested for evidential acceptance in a court of law. The possibility of wrongly identifying an innocent employee precludes any relaxation of legal evidentiary processes. There is also the possibility that the wrongdoer may take the business organisation to task with a grievance of wrongful dismissal, resulting in a fine being laid if the digital information is not legally acceptable. Therefore, the goal is to provide an extended credible information security system that can be configured and prepared to produce evidentiary digital evidence at the same time as event monitoring without wasting valuable network resources. When designed correctly the business will benefit from gathering and preserving digital evidence as compared to a computer system that is not forensically prepared. It is time-consuming and expensive to trace and preserve digital evidence, which may prove impossible, post-event (Kazadi & Jazri, 2015).

2.5 OPTIMISATION

Section 2.5 considers the optimisation requirements of business forensic readiness, beginning with identifying objectives and constraints, the effects of the Sarbanes-Oxley Act and network volume growth on business network infrastructure.

2.5.1 Business Optimisation

As discussed in section 2.3.1, Tan (2001) identified two objectives for forensic readiness, maximising incident data usefulness and minimising cost, both of which are required components for business optimisation. Two additional requirements for business forensic readiness optimisation are minimising time expenditure, both in the pre-incident data gathering and the post incident analysis phases and minimising the overhead placed upon the business network infrastructure. The business forensic requirements of any data gathering system must include an investigative component, but in accordance with evidentiary procedures, the investigation must not change or alter the gathered data in any way, so the investigative component must work either on a verified copy or in read-only mode. There will be constraints on real-time data analysis as well as overheads on processors and bandwidth within any forensic readiness data gathering system. There is a need for data reduction and therefore a corresponding need to accurately analyse data collection needs for business requirements (Casey, Katz, & Lewthwaite, 2013; Quick & Choo, 2014).

It is expected that a company must provide information that proves the state of the network security as part of doing business on the Internet upon experiencing a security breach as part of compliance measures for the regulatory purposes. The Sarbanes-Oxley Act (2002) requires organisations to control the release of information to entities outside the company's network and to implement IT governance policies that define the manner of the businesses electronic communications. Financial institutions are required to develop, implement and maintain a comprehensive written information Security program to protect customer record privacy and integrity. The US Sarbanes-Oxley Act (2002) requires compliance by maintaining comprehensive data audit processes, along with strict security measures. These compliance requirements for organisations may become part of an integrated network forensics process. The data audit compliance information can be recorded in conjunction with a data capture system as part of Digital forensic readiness (Pilli et al., 2010).

2.5.2 Network Volume Growth

Each year the volume of data is increasing beyond the capacity of hardware and forensic tools and this on-going growth in volumes of data presents serious implications relating to delays and backlogs. In the past it was the rapid increase in the size of storage media hardware that presented the single greatest challenge to the digital forensic investigator, but it is the contemporary issue involving massive amount of transmitted data that needs to be addressed. As digital technology continues to change rapidly, presenting increases in the number of connected devices along with greater transmission speeds can result in diverse data sets that are increasing in size and complexity. The challenge in large datasets is that the digital forensic investigator must locate relevant pieces of information, much like finding one needle in many haystacks. (Quick & Choo, 2014).

It is not unusual for a network traffic evidence artefact to contain multiple terabytes of data collected from many different points on a network of unknown topology utilising diverse data gathering methods. It has become necessary to determine some basic aspects including whether the data gathered is from a useful time period, whether the data is uncorrupted, and what network segments are covered, before any forensic investigation begins. Analysing multiple terabytes of data is difficult for most network forensic analysis tools to perform efficiently and in a timely manner. This sheer volume of data is already incompatible with manual investigative techniques, with most digital forensic tools already unable to handle the load. (Bhoedjang et al., 2012).

2.6 NETWORK FORENSICS

Section 2.6 is a development of the above section, and addresses network forensics, beginning with a definition of digital network forensics, then moves on to differentiate between security and digital forensics, as a post incident response and then identifies the major difference between traditional investigation of dead forensics compared to the gathering of live network data for forensic purposes.

2.6.1 Definition

Digital network forensics is defined in Palmer (2001) as:

> "The use of scientifically proven techniques to collect, fuse, identify, examine, correlate, analyse, and document digital evidence from multiple, actively processing and transmitting digital sources for the purpose of uncovering facts related to the planned intent, or measured success of unauthorised activities meant to disrupt, corrupt, and or compromise system components as well as providing information to assist in response to or recovery from these activities" (Palmer, 2001, p. 27).

2.6.2 Information Security Vs. Incident Response

Network security or the protection of networks and their services from unauthorised access, as a discipline ends upon a security breach or incident event. It is in this post incident phase or incident response that digital forensics can be implemented. Thus, incident response is divorced from information security aspects and it is only after the analysis and reporting stage that information can be reintroduced as part of network security. It is very difficult to project potential threats and therefore risk assessments are essential and should be considered part of network security and form the basis of forensic readiness. Despite robust security measures as cybercrime becomes more sophisticated and technology evolves network systems become vulnerable over time. It is not a question of if a security breach will occur; rather it is a question of when (Mouhtaropoulos et al., 2014).

Digital forensic examination is no longer performed on single machines, but the domain has expanded to networks of interconnected computers and real-time monitoring of network traffic for anomaly detection and determination is a characteristic of digital network forensics. Upon detection of an anomaly the network forensic examiner must determine whether this indicates an attack. Upon determination of an attack investigators must have the ability to track back through a data collection to identify attackers or attacking systems and ultimately, the goal is to provide sufficient data to provide legally acceptable forensic evidence (Pooe & Labuschagne, 2012).

2.6.3 Bit-Stream Forensic Investigation as Live Data

As discussed in section 2.5.2, no longer are computer forensics performed on single machines with their relatively small data capacities. The scope of potential evidence now includes complex systems of interconnected computers. One of the major challenges of this form of computer forensics is in the provision of reliable and admissible evidence, selected from a vast amount of data that may or may not be useful to the forensic examiner. The standard evidence collection processes relying on conventional assumptions regarding storage media is no longer valid. These assumptions can be termed the collection of dead forensics where a bit-stream image can be created from entire original evidence such as a hard drive when a system is taken off-line. This involves consideration such as logical and physical safety of evidence and authentication can be provided with simple hashing tool. One problem occurs when each of the computers connected to network of interconnected computers have very large storage capacities containing potential legally relevant artifacts (Kiravuo, Sarela, & Manner, 2013).

However, as data contained upon the network infrastructure itself is continually in flow, the gathering of this information whilst it is in this moving state can be known as the collection of live forensics. Live forensic analysis techniques involve information gathered from the network infrastructure during the time an incidence is taking place. The reliance upon dead or static digital evidence gathering is no longer valid for risk mitigation for forensic information in a business environment (Roussev et al., 2013). Live forensics which concentrates upon the gathering of moving and therefore volatile information has its own challenges to provide effective, error-free data collection of bit-stream information. It is essential the collection of this live information be enacted in such a way no changes are made to the data gathered. Many current forensic tools concentrate upon information gathered from hardware in a static state and may therefore be detrimental to live forensic examination (Pooe & Labuschagne, 2012).

Forensic acquisition has failed to scale with this growth, especially in areas of IO rates, where the network transmission speed may overwhelm a disc write rate.

Whilst forensic imaging was defined over a decade ago, where linear and complete image integrity is protected by a hash function in the field this is not possible. At gigabit per second network transmission speed the disc write rate will be 128 MB, which is approaching current theoretical max. This causes latency during the acquisition of forensic data process, which can cause data loss through Ethernet frames being dropped. This causes potential problems of evidential acceptance due to nonlinear and potentially partial information gathering. This becomes a serious issue upon the collection of information from a business network containing hundreds of machines (Bertasi & Zago, 2013).

It is therefore extremely difficult to collect all evidence moving throughout a corporate network, and finding a single piece of evidential data is even more taxing. The vast amount of data flowing through a business network makes the creation and processing of a digital forensic image problematic. The plethora of pervasive technology, especially within business networks where 'Bring Your Own Device' (BYOD) is not only accepted but encouraged, means there is an ever increasing variety of devices adding to the volume of data. The investigative techniques must also adapt to enable the forensic examiner to search for evidence in new digital devices (Konstantinos, Emmanouil, & Vassileios, 2013).

As most corporate entities in the modern world employs these networks of interconnected computers, forensic investigations of digital evidence can no longer be viewed as a post-event response to an information security incident. Network forensic readiness provides an opportunity to collect potential evidence in the form of log files, network traffic records, emails and downloads as part of the network traffic between computers. This collection of information prior to a security incident can be considered part of a corporate's due diligence as it is impossible to collect network traffic data after the event. Therefore it is necessary to consider what evidence is important for a wide range of potential incidents.

Most network information is not considered relevant by a business and much potential evidence is simply ignored and discarded as part of normal practice. Incident preparedness must become a corporate goal focused upon maximising an organisation's ability to use digital evidence. An organisation's ability to utilise network data must become a focus of forensic readiness. Therefore, in the corporate environment digital evidence can no longer be passively used but instead must be actively sought from a wide range of potential evidence sources (Rowlingson, 2005).

2.7 BUSINESS NETWORK INFRASTRUCTURE

Section 2.7 investigates some technical details presented with business network infrastructure. This begins with a study of the OSI reference model as the primary description of the communication process of network data transfer, then describes the Ethernet family of network technologies, Ethernet data encapsulation and finally discusses the Address Resolution Protocol as important evidential data when presenting end-point connectivity and data transfer.

2.7.1 OSI Reference Model

The Open Systems Interconnection (OSI) reference model, which plays a key role in networks, consists of seven layers which perform different tasks and passes the packet to the next layer until it reaches the final destination layer, either application or physical. This reference model was developed by ISO in 1984 to provide a common model for this complex aspect when related to network communications. The purpose of the OSI reference model is to provide a common basis for the co-ordination of standards development for the purpose of system interconnection. Thus, the OSI framework, as an abstract model, assists the real world interconnection system standards.

The OSI reference model divides the different functions and services into seven layers. The top three layers, Application, Presentation, and Session layers are known as the upper layers and are implemented in software. The Transport and Network layers are mainly concerned with protocols for delivery and routing of packets to a destination and may be implemented in software. The Data-link layer is implemented in hard and firmware whilst the Physical layer is hardware only. It is only the last two, or Data-link and Physical layers that are defined in Ethernet LAN specifications (ISO/IEC 7498-1:1994(E)).

In a simplified description of data passing from host a to host b.:

- The application, presentation and session layer take the user input and convert it into data.
- The Transport layer adds a segment header and converts the data into segments
- The Network layer adds a network header and converts segments and to packets.
- The Data-link layer adds a frame header and converts the packets and frames.
- MAC sub-layer converts the frames into bits which the physical layer can then put on the wire or physical medium for transmission.

These five steps are known as the steps of data encapsulation. When the bit-stream arrives at the destination the physical layer removes it from the wire converting it into frames and each layer will remove their corresponding header while the data flows up the OSI model until it is converted back to data and presented to the user (Dye, McDonald, & Rufi, 2008).

2.7.2 Ethernet

Ethernet is a family of networking technologies that are defined in IEEE 802.2 and 802.3 standards. Ethernet provides unacknowledged connectionless service over a shared media utilising CSMA/CD as the media access control. Shared media requires the Ethernet packet header use a data-link layer address to identify the source and destination nodes, and this is referred to as the Media Access Control (MAC) address. Ethernet is best understood in reference to the OSI model because Ethernet functions at the OSI Physical and Data-Link layers. The OSI model separates addressing, framing and accessing the media from the Physical layer standards of the media. Even though Ethernet standards define both layer 2 protocols and layer 1 technologies supporting different media and bandwidths, the basic frame format and addressing schemes are the same for all Ethernet varieties (Dye, McDonald, & Antoon W Rufi, 2008; The Institute of Electrical and Electronic Engineers, 2012).

At the Data-Link layer, the encapsulation process consists of creating and adding headers and trailers to the Network layer packets.

This provides three primary functions:

- Frame delimiting
- Addressing
- Error detection

The data-link layer provides transparent network services to the network layer so the network layer can be ignorant about the physical network topology providing access, to the physical networking media. This is responsible for reassembling bits taken from the physical medium into frames and make sure the information is in the correct order and requests retransmission of frames if an error occurs. Therefore, the layer provides error checking capabilities by adding a cyclic redundancy check (CRC) to the frame (ISO/IEC 7498-1:1994(E)).

The data-link layer consists of two sub-layers, Logical Link Control (LLC) and Media Access Control. The MAC sub-layer is concerned with the physical components and the LLC sub-layer remains independent of the physical equipment. IEEE 802.2

standard describes the LLC sub-layer functions, and the IEEE 802.3 describes the MAC sub-layer and the physical layer functions. These two sets of standards, 802.2 and 802.3 describe the functions of Ethernet

The LLC sub-layer takes the network layer Protocol Data Unit (PDU), typically an IPv4 packet and adds control information to deliver the packet to the destination node as an Ethernet frame. Layer 2 communicates to the upper layers through the functions of the LLC sub-layer. The LLC can be considered the software driver for the Network Interface card (NIC) and acts directly with the hardware to pass data between the media and the MAC sub-layer.

Ethernet is designed for easy deployment and minimal administrative overhead which means anybody can connect to an Ethernet segment by gaining access to a vacant port on a switch. Ethernet architecture also allows unauthorised network expansion by the user installing their own switch or wireless access point, allowing other people to join the network.

2.7.3 Ethernet Data Encapsulation

Data encapsulation is the process of adding headers and trailers to the network layer Protocol Data units (PDU) at layer 2. Header, or control data is added to the data as it moves down the OSI layers, which contains control and addressing information. This information is very important to the forensic examiner. A PDU is a generic term for data at each level, but the PDU is distinctly different at each layer. At layer 2, data encapsulation provides three main functions: Frame delimiting, Addressing and Error detection. The encapsulation process frame assemble before transmission and frame examination upon reception. The MAC sublayer adds a header and trailer to the layer 3 PDU, provides delimiters which are used to identify the group of bits that make up a frame. This process provides synchronisation between the receiving and transmitting node points and provides delimiters to indicate the start and the end of a frame, therefore, all bits between these delimiters can be identified as being part of the same frame.

Data link layer addressing is also provided by the encapsulation process, where each Ethernet header contains the physical MAC address, providing the node information to which the packet is to be delivered, along with the source MAC address from which the Ethernet frame was received. This information gives definitive physical addressing information of the node that originated the frame onto the LAN and the node that accepted the frame from the LAN. An additional function of layer 2 data encapsulation is error detection consisting of a trailer that contains a Cyclic Redundancy Check (CRC) of the frame contents. The receiving node creates a CRC from the bits in the frame received and compares this with the CRC value contained within the frame trailer. If these two values match, then the frame can be trusted as being received with no errors and will then be processed, otherwise the frame will be discarded.

2.7.4 ARP

Although Address resolution protocol (ARP) is one of the oldest and most used of network protocols, it is also one of the most vulnerable. An ARP announcement is a type of request that is unsolicited, not intended to cause reply known as a gratuitous ARP request. The ARP is a simple protocol and as such does not have any type of security and in the process of mapping the IP address to the correct hardware may result in serious breaches of security (Oh, Kim, Hong, & Cha, 2012)

The ARP protocol provides essential services, allowing network devices to communicate with the TCP/IP protocol. When a frame is placed on the network, as it must have a destination MAC address it can be seen that there is no efficient method to build the datagram layer to destination address without the ARP protocol, which information is stored in an ARP cache table. The ARP cache table is maintained on every host, including gateways and contains information on all valid MAC and IP address peers within that network segment. Although these entries can normally be set manually in a static mode, this is really used and the automated operation requires regular updates or the information in the cache table becomes deleted (Pandey & Saini, 2012).

As defined in RFC 826, STD 37, Address Resolution Protocol (ARP) is a protocol that handles conversions from a network layer address to the appropriate hardware address. ARP enables devices to discover the connection between a hosts IP address and its MAC address. It is a method of automated discovery, which allows communication through devices such as firewalls, bridges and access servers along with communication between hosts on the same network. In a local area network, therefore, mapping an IP address to a MAC address is the function of the ARP. A source machine must obtain a destination MAC address from the target machine that utilises the destination IP address to construct and transmit the Ethernet frame in IP over Ethernet networks. This is achieved by the ARP by mapping the IP to the MAC address via a broadcast request and through stored reply from a dynamic memory area called an ARP cache table. When an application sends data to an IP address the IP packet is created by the network layer of the OSI stack, and then encapsulated into the Ethernet frame by the data-link layer of the OSI stack. At this point the destination MAC address is required to transmit frame. Therefore, the network stack verifies the IP address in the ARP cache table to locate the required destination MAC address. If the information is not available on the local cache table a broadcast art request is sent to the network. Each machine on that network segment examines the ARP request to check if it is the owner of the IP address contained within the request. The machine that owns the required IP will send an ARP reply which contains its MAC address. This is a unicast reply, which is sent to the originator of the ARP request. The originator then uses this information to populate the local ARP cache table in order to complete and transmit the Ethernet frame (Alzubaidi, Cai, & Alyawer, 2014).

The role of the ARP table and MAC addresses is to clearly inform the endpoint about the originator of the delivered frame. Therefore this information is essential to provide complete evidential proof of endpoint connection, rather than the focus upon an IP address currently utilised and, perhaps erroneously considered evidential. Thus, it is the lack of a proper authentication mechanism that causes ARP vulnerability through forged ARP requests and replies where ARP tables can be manipulated by a malicious host broadcasting ARP requests on the local network (Alzubaidi, Cai, Alyawer, & Siebert-Cole, 2015; Salim, Li, Tu, & Guo, 2012).

2.8 CHALLENGES

Section 2.8 considers the challenges presented to the forensic examiner when optimising business forensic readiness, beginning with difficulties when reconstructing event information. The issue of approching the ideal of a hybrid investigation where law enforcement is considered an integral part of digital business forensics is then explored along with the development of potential solutions and finally looks at network security considerations, which the digital forensic data gathering phase must present minimal impact.

2.8.1 Information Reconstruction

When an incident involves complex attack techniques, it becomes problematic when identifying incriminating evidence and reconstructing a multi-step attack scenario that will be admissible as evidentiary in a court of law. When evidence may be destroyed intentionally by the attacker or the data gathered in the pre-incident phase is incomplete then an attack scenario is part of a prosecutorial narrative. There are few publications that formalise quantification methods for legal acceptability when reconstructing attack information (Changwei et al., 2014).

The process that was followed to reconstruct attack information must not only be acceptable within the digital forensic community and rigorous, but be reproducible by independent third parties in order to be evidentiary. Thus the information presented to court must relate the extracted evidential digital information to established factual information for judicial review. There is a dearth of uniform approaches to information reconstruction presentation, and there are many different approaches which may prove ineffective when utilised for business digital forensics. Non-standard processes, multitudes of different network devices and high levels of scalability present challenges to the forensic examiner (Kohn, Eloff, & Eloff, 2013).

Reconstruction of a crime or an attack is the forensic investigator's prime responsibility. This must be reduced to facts as defined by analysis of the forensic data and reported as such (Konstantinos et al., 2013). Therefore, it is essential to understand which information needs to be collected to provide an accurate analysis, for any particular circumstance. This indicates that a process of not collecting all data but just focus on information required to provide an accurate analysis may be acceptable in some cases. This may cause difficulties to the forensic examiner in event reconstruction scenarios, especially when evidence needs to be presented in an evidentiary manner.

It is therefore a question of whether collecting data subset, which is potentially easily implemented and causes minimal changes to current systems and procedures, can be undertaken in a forensically sound manner and abide with common forensic principles.

It is proposed and network forensic data gathering that it be sufficient to locate necessary evidence to support an evidentiary forensic investigation without the need for the collection of full network data traffic (Quick & Choo, 2014).

2.8.2 Hybrid investigation

The collection of voluminous amounts of network forensic data makes the creation of an evidential image or even processing the data collected difficult. The digital forensic examiner may be called upon to provide digital evidence in support of a conventional criminal investigation, such as fraud, embezzlement, or intellectual property theft. Thus, there is a problem where law enforcement agencies are faced with the challenge of adopting new investigation methodologies, as too is the forensic examiner. This calls for a hybrid investigation approach, where both law enforcement and the forensic examiner approach the investigation process in the same manner (Konstantinos et al., 2013).

Key elements of robustness and resilience have been identified by law enforcement agencies in the context of sustainable digital forensic investigation. Recommendations include having a pro-active approach to organisational resilliance through the implimentation of threat analysis and risk management within a digital research laboratory. Knowledge management was isentified as a key part of the process, not only of intangible assets such as creative processes and operational routines, but of tangible processes involving hardware and applications and can be considered a crucial element of organisational policy. It is essential to maintain dynamic adaptive capabilities in order to face new technological challenges which improves the capability to adapt to the rapidly changing environments that face the forensic examiner (Amann & James, 2015).

2.8.3 Exploring potential solutions

A postulated solution to the issues addressed above is to combine continual evidence acquisition with an automated information extraction system. If a forensic acquisition system simultaneously collects data as an extraction process, whilst monitoring and evaluation is underway, the process will create a forensic duplicate as well as allowing continuous event monitoring.

Potentially, a combined system will allow continuous forensic information gathering with very little network overhead. A combined system may also allow forensic examination of Digital evidence without incurring a time lost due to creating a forensic evidence artifact. The solution should be investigated with possibility of providing a monitoring system to provide a baseline that may indicate potential misuse. A combined system may also provide useful indications that the network system may be under an Advanced Persistent Threat (APT) attack.

2.8.4 Network Security Considerations

Any digital forensic process applied to an organisation’s network must consider three essential security objectives in order to maintain a secure viable network, confidentiality, integrity and availability. These can be considered core security objectives and will differ in order of priority, depending on the organisation’s structure (Whitman & Mattord, 2016).

Confidentiality, the first of the three objectives with the objective to prevent detour and detect improper disclosure of information. Restricting access to components of the network system to only authorised parties is also a component of confidentiality. Confidentiality preservation in the real world is perhaps the best understood of the three security considerations and is easy to maintain as part of a forensic data gathering system. There are numerous techniques to protect the confidentiality of digital forensic data such as data encryption, but these may place substantial overheads on a live capture system (Balogun & Zhu, 2013).

Integrity, the second objective, whereby improper modification of information is guarded against by deterrence detection and prevention. Thus, information gathered must be guarded from modification including writing, changing of status deleting, creating or changing of data.

Availability, which is the third security objective, is the process of prevention, detection and deterrence of improper denial of access to services. Legitimate access should be provided to authorised parties, whenever requested. Another aspect of the security objective is holding authorised users accountable for their actions to the data and services. This not only includes providing the object or service and usable form, but to ensure capacity to meet service requirements and the provision of the service in a timely manner.

Any digital forensic data collection process must be careful to regard the above objectives so that any data gathered must remain confidential, be protected against alteration and modification, and must not impinge upon network resources more than is strictly necessary.

2.9 SUMMARY ISSUES AND PROBLEMS

Section 2.9 identifies several issues from the literature review, which are summarized. These issues include the lack of testing, especially testing for falsifiability and determination of error rates and no single point of reference when designing a forensically ready system (see Section 2.1.1). There is a need for forensically ready systems to adhere to legal admissibility standards at all times, as data presented at court may be disallowed, rendering the data collection moot (see Section 2.2.2). The considerations of cost and impact on a business are rarely studied within an academic setting and as such not studied as part of proposed digital forensic readiness models (see Section 2.3.1). Rates of growth in data volume and technological change effect the collection of live data and event reconstruction, which is of continual concern to the modern digital forensic examiner when considering forensic readiness for a business network (see Section 2.5.2).

2.9.1 Digital Forensic Falsifiability Testing

Very few digital forensic tools have been subjected to the Daubert standard, which determines the admissibility of scientific evidence in court. The tools and processes that are accepted by popular usage, have not been independently tested, the procedures are rarely published and subjected to peer review, are not part of standard protocols and are not generally accepted by the wider digital forensic scientific community. Most importantly, when applying Popper's scientific theory, there is no known error rate associated with digital forensic scientific procedure and tool utilisation.

The omission has resulted in the ad-hoc development that is evident in the literature that has been investigated. Digital forensic scientific theoretical development is in danger of being regarded as pseudo-science and even metaphysical without stringent testing and publication of results.

2.9.2 No single comprehensive approach

As shown in section 2.1.3, there are many forms of digital forensic models and frameworks, and no single form meets the requirements of digital forensic readiness for business purposes. This means that businesses are in a reactive situation rather than pro-active, where there is little thought to risk mitigation through establishing a pre-incident data collection phase as part of organisational policy. Thus a business is likely to recover form an attack badly, reacting to events in order to establish business functionality, rather than having policy in place designed with prosecution of offenders as a priority.

2.9.3 Presentation of deficient expert evidence

The lack of a pre-incident data collection policy creates an issue where the data that has been collected may be incomplete, allowing the defense to identify a lack of non-repudiation. This lack of completeness may also present problems for an independent third party to reach the same conclusions as the forensic examiner presenting expert evidence. An example of this is the reliance upon IP addressing to prove end point data reception, where in fact it is only the MAC address that can provide this evidence of Ethernet network data transfer conclusively.

The digital forensic expert when investigating incidents on a business network and live forensic data will apply various different techniques, tools and technologies. Relying on information contained within log files, for example, is only a small part of the completeness required for the presentation of expert evidence, and any such approach is in danger of being rejected as evidentiary. Table 2.1 shows evidence admissibility criteria and the corresponding principle and foundation.

Table 2.1: Evidence admissibility rubric

Principal	**Foundation**	**Principle Concept**	**Section**
Scientific procedure must be independently tested	Daubert	Testing	2.2.4
Scientific procedure should be published and subjected to peer review	Daubert	Peer review	2.2.4
Scientific procedure must be associated with a known error rate	Daubert	Error rate	2.2.4
Execution of Scientific procedure methodology, standards and protocols	Daubert	Standards	2.2.4
Scientific procedure generally accepted by relevant scientific communities	Daubert	Acceptance	2.2.4
Conclusions reproducible by independent third party	Davis, C. et al. (2009) ACPO	Replication	2.2.3 2.4.4(ACPO)
Based on properly documented digital sources	Bates, J. (1998)	Forensic imaging / completeness	2.2.3

2.9.4 Cost and Impact

Business organisational entities are effected by cost and impact, especially when the network infrastructure is affected. This is a major concern, where any digital forensic undertaking is required to minimise cost required to produce the maximal incident evidential data. Any forensic readiness implementation must present minimal impact on the business network infrastructure in terms of bandwidth overhead, information transmission delays or discarded information. Thus, any development of digital forensic readiness for business applications must keep cost and impact levels as low as possible by collecting the most relevant data only.

2.9.5 Volume growth and technological change

The single biggest issues that have been identified in the literature review are the rate of data volume growth and the influence of technological change on forensic readiness. This means that any forensic readiness approach must be resilient when adapting to technological change and robust when adapting to rapid growth in data transmission levels. A framework developed with this constraint in mind must be specific enough to delineate exactly which information should be gathered, and yet provide enough flexibility to allow for technological change.

2.9.6 Live data collection

The volatile nature of network data transmission is an issue with live data collection. If the system that is in place presents problems with processing and capturing forensic data, information may be dropped, which will not only impact upon any forensic investigation, but may render any information to be unacceptable as evidence through incompleteness. The gaps in information gathered may also effect identification of the guilty parties, or in a worst case scenario, may even mask the incident event itself.

2.9.7 Event reconstruction

Finally, the issue of event reconstruction is problematic without a comprehensive approach to forensic readiness. The focus of a digital forensic investigator is to provide credible information that reconstructs an incident and through this process, identifies any miscreant, and produces a report that is reliable and provides enough information to ensure successful prosecution in court. Therefore, any information gathered is required to contain the enough information to generate this report in a timely and efficient manner, yet only gather the minimum of extraneous information.

The following is a summary of the problems, issues and challenges dicussed:

- There is a challenge when encountering the continual need for a systematic arrangement and classification of forensic tools. For live network forensic readiness, the classification process would involve a rigorous and exhaustive organization to determine a set of agents that share the same ontology.
- The standard approach to the continually increasing data volume levels facing the forensic investigator is to extract and capture only relevant data. This in turn may have a negative effect on evidentiary standards of completeness
- The data to be collected to be forensically ready is volatile by nature, and any data that is not collected is lost. This presents potential problems if complete data-sets are not collected and filtering is not performed post-collection.
- There must be continual awareness of the legal and evidential ramifications of any decision that the forensic examiner makes. This requires adherence to directives such as are contained within the Daubert (1993) principles, such as establishing error rates when selecting forensic tools.
- There is an obligation to establish a test bench to provide a standardized platform to test forensic tools. The problem is how to determine evidentiary requirements such as error rates and levels of completeness.

2.10 CONCLUSION

The literature review conducted in this chapter provides a comprehensive outline for the requirements of an effective digital forensic readiness implementation for organisational use. The chapter began by identifying the lack of structured scientific development within the digital forensic discipline, and identified an absence of formal testing and discovery of error rates, which forms the basis of knowledge building as postulated by Popper (1959). The fact that development in this arena has moved through paradigm shifts through the advance of rapid technological change rather than gradual and progressive growth, indicates that Kuhn’s (1970) postulations of the progress of scientific knowledge is applicable. The inclusion of a legal element which is changing the lexicon of digital forensics shows where digital forensics is not just providing event detection but is embracing legal, social and structural inclusion. Therefore, there is a requirement for testing of forensic gathering tools and ascertaining levels of error in order to address the legal and structural inclusion.

The requirement for digital forensic readiness to address the intersecting nature of law and computers to maintain evidentiary admissibility standards were discussed.

The requirements, including standards, principles and acceptance through the Daubert test were analysed, indicating that there is a requirement that any data gathering process be undertaken with a view to providing acceptable evidentiary information. These requirements lead to the realisation that a pre-incident plan must be included in order to become forensically ready. The literature review then showed that there was no single, universally accepted forensic readiness framework, application or method that is accepted to provide forensic readiness specifically suited to business requirements and investigates the fundamental focus of these business requirements. Any development must include business optimisation conditions such as cost and impact minimisation whilst providing the maximal amount of useful information, whilst allowing for the rapid growth of data volumes as well as the collection of live network data.

Therefore, the proposed research will include forensic gathering tool testing, the development of a data gathering structure that collects the smallest volume of data whilst still providing evidentiary standards of completeness and non-repudiation and a system that delivers the maximum effect for the minimum effect on network infrastructure. Chapter 3 will compare several different forensic tool testing and laboratory methodologies, and the results will be used to design, implement and evaluate a framework that optimises forensic data readiness through network packet management.

Chapter 3
METHODOLOGY

3.0 INTRODUCTION

Chapter 2 has provided a review of relevant literature. It is critically evaluated to form a basis by which this study will proceed. Problems and issues have also been identified for further investigation. The purpose of Chapter 3 is to formulate a research question and an appropriate research methodology. The focus for research is derived from the problems and issues identified. Chapter 3 identifies the process that will be used to investigate the relationship between live data capture and legal requirements noted in Chapter 2. These legal requirements include concepts such as completeness and error rate determination (see Section 2.2.3 and Section 2.2.4).

Five similar studies are analysed in Section 3.1 to gain knowledge from previous researchers about research methods and methodology. The knowledge obtained will be used to develop a suitable research context and appropriate methodology for this thesis. The five similar studies are then summarized in section 3.2.1 and related to the problems identified in Section 2.10. The evaluation and selection of researchable problems is tabulated in Section 3.2.2. This is in order to identify the research problem and establish relevant research questions as shown in Section 3.2.3. Section 3.2.4 postulates the hypotheses which are used to shape the research design. The research phases are then adapted from guidelines provided by NIST (2001) and Wilsdon and Slay (2006) to form an empirical methodology and research phase map as shown in Section 3.2.5. A data map has then been formed to articulate the relationship between the research phases, research questions, sub-questions and hypotheses to be tested and is presented in Section 3.2.6.

Section 3.3 investigates and defines the data requirements for the proposed research, consisting of data generation, data collection, data processing and finally data analysis and presentation. Section 3.3 ensures a planned research design through identification of data required in order to test the hypotheses postulated and therefore to answer the proposed research question. Finally, Section 3.4 discusses the limitations of this research in terms of internal reliability and external validity which includes handover which may be incorporated into future research. Section 3.5 provides a conclusion, where the main points of Section 3 are summarized.

3.1 REVIEW OF SIMILAR STUDIES

Five studies that are relevant to the topic of this thesis have been analysed and critically reviewed in order to gauge existing research determinations and definitions. The analysis will then establish the direction of the research and determine the methodology to be applied. Thus, the task of this section is to define the conduct of the research for the topic area of business network forensic readiness.

3.1.1 Hoque, Bhuyan, Baishya, Bhattacharyya, & Kalita (2014)

Hoque, Bhuyan, Baishya, Bhattacharyya, & Kalita (2014) present a survey of network security tools. They assessed the various tools from both an offensive and a defensive viewpoint so that a taxonomy of network security tools could be developed. Hoque et al. (2014) categorised attacks into a series of distinct classes based upon severity of impact potential and behaviour characteristics. The premise of the categorical approach is that network attacks can be facilitated through the unscrupulous use of network security tools. The study of these tools can aid an attacker with identification of potential network vulnerabilities. In order to implement the research design the authors formulate a set of network security tools that can be utilised in the capture of live network traffic. They provide a level of pre-processing and feature extraction and vulnerability analysis. The study therefore provides a survey of network security tools and systems that is comprehensive and logically structured, designed as a basis for future network security research. The study then investigated network attack scenarios and concepts forming the basis of the study, which suggested that network tools may present risk mitigation against these attacks. The tools were then described and analysed for effectiveness and comparisons that were made against the completed list of tools (Hoque, Bhuyan, Baishya, Bhattacharyya, & Kalita, 2014).

Hoque et al. (2014) present a generalised four step attack launch execution plan: gathering information, vulnerability assessment, attack launch and finally an attack history clean-up stage. This series of steps was analysed by the authors and then appropriate network security tools that could be utilised to combat these attacks were placed into categories. The categories that the authors suggest in the study are: information gathering tools, sniffing tools, and scanning tools. The authors note that the tools selected for examination were not equally valuable at all times, or for all purposes. The tool’s value depended on the user’s purpose and requirements at certain points in time. The study provides a differentiation between live network and static data requirements. The research within this thesis concentrates upon live network data.

The information presented by Hoque et al. (2014) culminated in a two part delineation based upon the differentiation of use: sniffing tools and scanning tools. The paper suggests that sniffing tools are utilised to capture and analyse traffic, normally in the form of layer 2 frames, in motion across a networking infrastructure. Protocol information was determined to be utilised only in the presentation and visualisation of information. Scanning tools are utilised in the study and it suggests an overall report of the status of various network devices, ports and IP addresses are necessary in order to assess vulnerabilities in the network. The authors then analysed attack tools, and categorised these under the following headings: trojans, DoS, packet forging, application layer attack tools and fingerprinting attack tools. Thus it is apparent that the study relied upon an analysis of attack tools to generate a list of tools that had the potential to defend a network against tools used as a base for a network attack, such as network monitoring tools.

Hoque et al. (2014) presented information that is of interest to the digital forensic investigator when capturing networked data, which is therefore relevant to this research thesis. Network security tools were grouped by determining the security threat level and then categorised by the type of threat that was generated by the tool under test. This process suggests that security tools are subject to other categorisations than threat level, indicating a taxonomy that would be useful to a network forensic examiner. The ability to capture live data, for example, is a category that suggests a requirement for further research. This study by Hoque et al. (2014) therefore demonstrates that a network security tool may be used for purposes other than originally intended by the vendor. This indicates that additional research into the potential forensic abilities may be included as an additional categorisation of these security tools.

Into the proposed research methodology for my thesis the following points have been adopted:

- **Taxonomy:** A Classification scheme will be integrated into the proposed methodology. A classification of tools will encourage future research and ensure transference will be possible.
- **Forensic Tools:** A network security tool may be used for purposes other than originally intended by the vendor. Therefore, the tools selected for test in the proposed methodology of this thesis will be selected for their utility as a potential network digital forensic tool, even though they may not be considered to have been designed for that purpose.

- **Delineation:** A clear delineation between sniffing and scanning tools when used for networked digital forensicsis will be integrated into the proposed methodology. Therefore a packet sniffing tool and an intrusion detection system will be selected to be tested by the proposed methodology of this thesis

3.1.2 Pilli, Joshi, & Niyogi (2011)

Pilli, Joshi, & Niyogi (2011) investigate ways to reduce the large amount of data volumes network forensic investigators require when collecting live evidence. The study begins by discussing the difference between network security and network forensics. Network security attempts to control traffic by blocking access but network forensic processes collect evidence and do not block the criminal network activity. Pilli et al. (2011) identify attributes generated by network events that are of interest to a forensic investigator, and determine a minimum representative set that would be considered evidential. The network events are correlated with a particular attack format, and this correlation is then categorised and copied into a database, generating statistical threshold information. This information was then used to analyse captured data files for suspicious data, extracting information with the highest probable evidence level, which was then stored on a third database of evidentiary specific information.

Pilli et al. (2011) explore the TCP/IP protocol suite to identify security attacks, which were then categorised into the different protocol levels and divided into subsets of specific threat types. Each attack subtype was examined and unique characteristics were then determined. Attacks on the IP, TCP/IP, UDP and ICMP protocols were studied, and characteristic parameters were then determined by identifying manipulations within particular Ethernet fields of each protocol, causing malicious activities. Attack parameters were then compared to Ethernet field manipulation characteristics and the resulting correlation was then used to determine an attack detection process. The study then utilises a packet capture program to parse the attack field parameter information, and extract only the relevant packets from data captured on a network infrastructure.

The process suggested by Pilli et al. (2011) is important to the forensic investigator, as this system directly reduces the volume of data required to meet the parameters of the study. The minimum amount of information required to produce the maximum amount of evidence selected to be stored in an evidence data storage system. The process reduced the number of packets captured from a packet capture file, thereby reducing the total volume of data. The reduced data-set has an impact on the evidentiary standard of completeness. The proposed process will also only capture the packets that have been identified as suspicious after being parsed through a parameter list, in what amounts to an Intrusion Detection System. This is an example of a network security tool re-engineered as a forensic device, which is directly relevant to the topic of this thesis.

The TCP/IP attacks investigated by Pilli et al. (2011) involve layer 3, or network layer manipulation, where packets that are transmitted from router to router to form connection oriented services between source and destination. UDP which is an application interface and ICMP, an application within the TCP/IP suite, were investigated as part of the TCP/IP protocol suite. The selection process was determined by studying regular attacks on these upper layers of the TCP/IP suite, such as: spoofing for IP protocol, SYN flooding for TCP protocol, Port Loopback for UDP protocol and Smurf attack for ICMP protocol. Attack characteristics were then studied and parameters assigned so the proposed system would become effective. The parameter assignation is an issue for IDS systems, when they are used as security in a networked infrastructure. Therefore the researcher must be aware that issues discovered with a security tool may be inherited by a re-engineering of the tool for forensic purposes. The information collected for use in the study by Pilli et al. (2011) is a post attack response, and would not be useful for detecting a zero-day attack. When the characteristics of an attack are unknown at the time of the attack, the attack is known as a zero-day attack.

The reduced information was then collected in a forensic database, stored for future reference by a forensic investigator. The data was deemed to be useful to the forensic investigator within the study, signifying that the information was not considered evidential. A complete data-set was required to be maintained along with the reduced data-set database. The approach enabled a forensic investigator to highlight data considered important and then cross reference the data with the original, complete data-set. These outcomes imply that further research is required to determine the actual evidentiary value of the reduced data-set generated, where individual packets are captured based on field value characteristics. However, the study demonstrates that encapsulation data can be parsed and characterised data captured, reducing the volume of data dramatically, which is of direct concern to this thesis.

Upon evaluation of the study by Pilli et al. the following points will be integrated into the proposed research methodology for my thesis.

- **Ethernet Topology:** Ethernet has been selected as the network topology base for a proposed assertion test bench. This is because Ethernet is a widely used business infrastructure and characteristic parameters can be determined within the particular Ethernet fields of different protocols.

- **Datasets:** It is important to detect a minimum representative set that was still be considered Evidentially valid. Therefore, a zero reduction has been determined to be required for the proposed research methodology is the best representation of complete dataset.
- **ICMP:** The ICMP protocol has characteristics that can be readily identified. ICMP is identified as a significant attack vector threat, by such means as an ICMP address sweep. Therefore ICMP has been selected as a protocol to generate as a test of the chosen tools on the assertion test bench.
- **Wireshark:** Pcap Files generated by Wireshark have been identified by Pilley et al is a common packet capture file format. Therefore Wireshark has been selected as the packet capture tool for evaluation within the proposed methodology of this thesis.
- **SNORT:** SNORT has been selected as the IDS tool to be evaluated within the proposed methodology of this thesis.

3.1.3 Casey, Katz, & Lewthwaite (2013)

Casey, Katz, & Lewthwaite (2013) consider the effects of large volumes of data impinging upon forensically sound investigations in a digital forensic laboratory. The study looks at ways to process data from three sources; network traffic, static data and malware, efficiently and consistently, in a digital forensic laboratory. The study determined that improvements could be made by considering the forensic process rather than focusing on improving resource constraints. The study also established that it is not necessarily speeding up individual tasks, but generating useful information and effective case management that will engender timely results. The study therefore investigates the theoretical process of the forensic investigation, beginning by establishing two goals; increasing efficiency of the full process rather than the sub-parts and providing valid data that assists the forensic investigator in making effective decisions.

The study begins by looking at the process in general terms of forensic investigations: preservation, extraction, storage, examination, reporting and correlating of forensic data and seeks to remove process barriers. Casey et al. (2013) investigate bottlenecks within the process by first studying the complete process in a given context, using a methodical approach of capability implementation and requirement documentation. The study investigated the initial focus of forensic examiners, and established that automated forensic tools were employed and that it was disk IO speeds rather than processor capabilities that caused bottlenecks. This shows the importance of revealing impediments to the forensic investigation system that cannot be overcome, such as are hardware limitations. The converse to this outcome is that the investigation indicated that processor speed was being under-utilised, and therefore multiple processes could be running simultaneously without affecting the system output. The study then sought to establish process parts that were not scalable, such as extracting forensic duplicates, which is a serial process by nature. The approach was not scalable, as the continually increasing volume of storage media causes the process to become less efficient. The study suggests a potential solution, which is to combine sections of the forensic investigation process, such as combining automatic evidence extraction with acquisition and duplicate image creation. Therefore, the capabilities of the tool must be tested to identify issues to accurately assign cause and effect and thus determine mitigation strategies.

Casey et al. (2013) advocate that there are more ways to reduce process impact, than to increase speed or capacity when investigating bottlenecks within a forensic process. An example of this is to make sure that parts of the process are not idle, whilst waiting for a constrained process segment to become free, by creating a buffer or load shifting capabilities. The study then shows that it can be counterproductive to focus on one bottleneck, as changes made to the system can create choke points at different parts of the process. Therefore, the forensic process must be regarded as a whole when being re-engineered with efficiency increases as the goal and that monitoring and reviews are an important part of establishing the effectiveness of any changes. Finally, the changes must be useful to a wider range of organisational stakeholder than the forensic investigator, such as case managers and the prosecution and legal staff. Therefore, speed and efficiency metrics are found by investigating error rates.

The study by Casey et al. (2013) advised adding a section to the forensic investigative process concerning the ability to enable effective decisions. The study suggests that information extraction or triage stage of the forensic investigative process is primarily to support decisions about forensic processing. Triage can cause issues, however, when data may be collected that contains missing evidence or even the wrong information. If a high level overview is not conducted beforehand, problems can arise, such as identifying only one stage of a multi-stage attack as being relevant and discounting vital information that renders event reconstruction incomplete. When a reduced data-set is captured from live network information, which is volatile by nature, additional data cannot be recaptured post-event. The study terms this “information we could have had yesterday” and is lost data is considered to be an error or failure state during the methodological development of this thesis. (Casey et al., 2013, p. 141).

The study by Casey et al. (2013) showed that it was possible to integrate the preservation, extraction and storage stages of the digital forensic investigation process. The study also indicated that by extracting data to different points increased the efficiency of the total forensic investigation. The development of a high level overview suggests that the forensic examiner could work on a copy of data that has been subjected to a triage phase, inspecting a reduced data-set, reducing the time spent by only inspecting relevant data. Any information that is found to be significant can be cross-linked to an original, full data-set and therefore comply with legal evidential requirements for completeness.

Upon evaluation of the study by Casey et al. the following points will be integrated into the proposed research methodology for my thesis.

- **Tool Testing:** It is important to identify issues by testing a forensic tools capabilities. Therefore two forensic tools have been selected for test within the proposed research methodology.
- **PCAP Files:** PCAP files have been identified as most commonly used to capture network traffic information. Therefore PCAP file generation has been selected for use within the proposed research methodology.
- **Wireshark:** Wireshark has been identified as a tool commonly used for in-depth examination of generated peak Files. Therefore Wireshark has been selected as a tool for evaluation within the proposed research methodology.
- **Efficiency:** Efficiency rated by completeness has been selected as the baseline for use within the proposed methodology of this thesis. Thus the collection of a full dataset will indicate a zero error rate within the proposed methodology.

3.1.4 Changwei, Singhal, & Wijesekera (2014)

Changwei, Singhal, & Wijesekera (2014) study the effects that reconstruction of advanced network attacks have upon admissibility standards of digital information presented as evidence. The study acknowledges the difficulty that the forensic investigator faces when petitioning a Judge for admissibility of evidence gathered by combining reconstructed events from a multi-stage, multi-step advanced attack. The study attempts to incorporate evidentiary legal aspects, addressing admissibility standard requirements for substantiating evidence, with the technical aspects of digital forensics. The study is an indication that consideration of admissibility standards is vitally important to the digital forensic examiner.

The risk that a digital forensic investigator faces, with evidence admissibility rulings, is that charges may be dropped or the parties found not guilty, rendering the findings of a complex investigation ineffective. Changwei et al. (2014) recognise that reconstructed attack scenarios form a considerable portion of the prosecution presentation and must be accepted by a Judge as admissible evidence before a Jury can receive the evidence. The study also acknowledges that finding evidence that will assist reconstructing events is challenging when attackers use anti-forensic techniques and tools to avoid being tracked. The study suggests that even though using Intrusion Detection System (IDS) alerts as forensic evidence has been contested in court, the IDS logs do provide an initial focus of investigation to the digital forensic examiner. The study also proposes that combining and correlating alerts is required when reconstructing multi-stage multi step attack scenarios. This becomes even more problematic to the digital forensic examiner, the study suggests, as the reconstruction of these complex attack scenarios are not integrated with acceptable legal admissibility standards. The presentation of findings may lead to issues when the evidence path is broken through being intentionally destroyed and therefore missing which will call for an explanation justifying the use of incomplete evidence. The forensic examiner, the study submits, can therefore quantify assertions by integrating evidence relevance with credibility by determining levels of uncertainty.

Changwei et al. (2014) construct an experimental network as a test bench to be attacked, and formed attack scenarios to be analysed and the evidence reconstructed. The experimental network consisted of an IDS system, webserver and database server, all of which were able to detect the attack scenarios and generate log information and alerts. The study utilises time stamp information contained within IDS alert logs to group each separate attack scenario’s information, pre-determining a time window for each attack. The study also used the IP source/destination pair to determine evidence for the same attack class. The attack scenario generation procedure constructed rules from generic attack information to analyse the collected evidence and then constructed an attack tree that correlated the collected information. Therefore, the assertion test bed requires formation a standardised hardware platform to provide a baseline for the assertion testing procedure.

The study recognises that there are other constraints placed upon evidence, such as including chain of custody information. Two issues were addressed by Changwei et al. (2014). The first issue tackles the dynamic environment of a networked environment where evidence changes at a high rate, especially if the attacker is using anti-forensics to remove, hide or destroy evidence. The second issue the study investigated was that a single digital evidence collection tool may not suit all the needs of a particular digital forensic examination. The study recommends, in each case, that validation and substantiation through the use of additional constraints such as timestamps, relevance and testing for validatable hypothesis may mitigate the two issues investigated. Timestamp information was found useful when modelling legal admissibility in the study, as a time stamp on each piece of evidence will corroborate the chronological order of each attack step, and checking the end and start times of the attack step will provide a time frame within which relevant information should fall. The study accepts that timestamp values may not be exact due to system delays and nature of the logging systems themselves. Changwei et al. (2014) recommend that a distributed, synchronised time stamp is required.

The study found that there were situations where the same evidence could generate different explanations when generated between the same source-destination computers and postulated three different reasons. First, that the IDS may not have recognised that an attack was underway, especially if the attack was part of a larger coordinated incident. Second, that there may have been many different attacks launched, and only one succeeded and therefore generated an alert. Third, that different explanations may be provided by different experts for the same evidence. Changwei et al. (2014) determined that the Court would use Daubert (1993) principles to determine evidential validity, where the Court can judge the admissibility of evidence based upon the expert's principles and methodology, and whether these have been published, are widely accepted, or have been tested. Therefore, legal standards of acceptability are required as an integral component of forensic readiness.

Upon evaluation of the study by Changwei et al. the following points will be integrated into the proposed research methodology for my thesis.

- **Admissibility Standards:** It is important for the forensic investigator to continually consider admissibility standards. This includes reliance upon the Daubert standards which include error rate identification and complete dataset collection. Therefore the collection of a full dataset will be indicated by a zero error rate been generated by the tools evaluated within this proposed research methodology.

- **IDS:** IDS alert logs are identified as a commonly utilised tool for forensic examination. Therefore, an IDS has been selected as a tool to be evaluated within this proposed research methodology.
- **Networked Environment:** A dynamic networked environment has been identified as an experimental network test bench. Therefore a dynamic live network environment where evidence changes at a high rate has been chosen as the basis of the assertion test bench within the proposed research methodology of this thesis. The assertion test bench will be designed to contain a standardised hardware platform across all test scenarios.
- **Tool Testing:** A single digital evidential tool may not be suitable for the network forensic examiner as appropriate that the capabilities of the tool need to be tested. Therefore the testing of the capability of two tools have been selected for evaluation within the proposed research methodology. This is to determine whether either tool is appropriate for network forensic use by the forensic examiner.

3.1.5 Kiravuo, Sarela, & Manner (2013)

Kiravuo, Sarela, & Manner (2013) examine Ethernet LAN security, recognising that the Ethernet is the infrastructure that is ubiquitous and yet stands almost unnoticed by the user. The study focuses upon Ethernet related attacks and vulnerabilities, acknowledging that Ethernet is a fundamentally insecure technology, with no authentication required at layer 2, with MAC address filtering commonly used as a risk mitigation factor. Kiravuo et al. (2013) suggest that literature available in the form of network security texts focus on higher layer protocols to mitigate risk. With the goal of analysing Ethernet technology in order to understand the security issues, the paper outlines the protocol schema, hardware concepts and then investigates security solutions that are available.

Karavuo et al. (2013) considered the prevalent LAN technologies, and focused on pure Ethernet, when investigating security issues. They concluded that wireless technologies and layer 1, or physical issues to be out of scope for the investigation, concentrating therefore on layer 2. The study generalised the hardware technologies that together compromise a typical “Corporate Ethernet” (Kiravuo et al., 2013, p. 1478) and used the configuration to visualise, test, demonstrate and then report findings. The study set out a comprehensive display of MAC address utilisation and demonstrated this functionality on an Ethernet segment of a corporate network, and then systematically studies the effects of malicious behaviour on the network segment. Thus the study uses a standardised laboratory test environment of a single Ethernet segment to test the state of Ethernet security by investigating packet manipulation.

The study then investigated the Ethernet frame, discussing how the MAC address supplies the control information that determines where the frame will be sent, as well as identifying the sender. Thus it was determined in the study that each MAC address must be unique on the same Ethernet segment. The study also points out that control bits within the header of the Ethernet frame that effect delivery by indicating the frame is one of three types, unicast, multicast, or broadcast. Unicast for single, point-to-point delivery, broadcast for segment-wide transmission and multi-cast for frame distribution to select MAC addresses. The main component of an Ethernet frame, the payload, was not considered as relevant to the investigation of the study, even though the payload comprises the majority of the frame’s volume. The payload is encapsulated in higher order protocols, and therefore not germane to the study of an Ethernet networking laboratory. The Ethernet frame does support Ethernet handling, where a Cyclic Redundancy Check (CRC) value is generated before transmission and appended to the Ethernet frame before delivery. The CRC hash function runs upon reception, comparing the hash result to the value contained in the Ethernet frame trailer and the entire frame is discarded if the values do not match. Therefore it can be confirmed that a frame is complete and the data has not been lost or altered in transit by the frame having been accepted at the receiving node. The Ethernet handling process provides an indication of completeness, by showing that the frame was accepted.

The study then investigates the Ethernet switch, considered by Kiravuo et al. (2013) to be the most common, layer 2 LAN technology used within an Ethernet business network topology. The study then investigates how the MAC addressing functions within the switch via a learning function, which allows the switch to recognise incoming frame’s MAC addresses and add them to a table that matches the MAC address value to the incoming port. The switch builds up the table which maps the switch’s local and limited view of the network. The study then investigates the three structural layers of a switch, the data plane, the control plane, and the management plane. When investigating trunking and linking protocols such as Spanning Tree Protocol (STP) VLANS and Simple Network Management protocol (SNMP) for example. This indicates the importance of MAC addressing information to the Forensic Investigator as well as including common business hardware to an assertion test bed.

Kiravuo et al. (2013) found that self-configured Ethernet suffers from a lack of security against attacks and that ARP spoofing, and MAC flooding as well as Man-in-the-middle MAC based compromises. The study demonstrates the effectiveness of utilising a single Ethernet segment to conceptualise, test and trail security solutions through the development of an idealised corporate network topology. The conceptualised test bench single segment Ethernet can therefore be utilised to formulate an assertion test bed for research into networked digital forensics. Whilst the study is structured in a way that demonstrates a process to investigate the basics of Ethernet, in order to understand security ramifications, the process can, though be adapted to investigate the forensic ramifications of layer 2 frame transportation.

Upon evaluation of the study by kiravuo et al. the following points will be integrated into the proposed research methodology for my thesis.

- **Ethernet:** Ethernet is the most ubiquitous and common infrastructure within the corporate environment. Therefore Ethernet has been selected as the proposed environment of the assertion test bench utilised within the methodology of the thesis.
- **Typical Corporate Infrastructure:** The assertion test bench construction for use within the research methodology of this thesis, will generalise hardware technologies, ensuring a standard hardware platform.
- **ARP:** ARP spoofing has been identified as an attack vector so therefore ARP has been selected as a protocol to generate on the assertion test bench to evaluate a forensic tool.

3.2 RESEARCH DESIGN

Comparative analysis of the five studies presented in Section 3.1 provides guidance towards the research direction of this thesis. The following section evaluates the methodology followed in the five studies in Section 3.1, comparing and contrasting implementation along with the research direction of each paper. The scrutiny identifies areas where research previously implemented can be incorporated within the research design of this thesis, as well as identifying gaps that generate the research question, sub question and hypotheses addressed in Section 3.2. Data requirements are addressed in Section 3.3, where data generation, collection, processing, analysis and presentation are discussed. The limitations, reliability and validity of the research methodology are discussed in Section 3.4. Finally a conclusion is given in Section 3.5.

3.2.1 Summary of Similar Studies

Two of the research papers investigated in Section 3.1, Hoque et al. (2014) and Kiravuo et al. (2013) focus upon network security and these two research papers provide valuable insights regarding the catagorisation of security tools and Ethernet LAN security. This taxonomic approach has been used in the development of the research design presented in this section. Pilli et al. (2011) and Casey et al. (2013) investigate the effects of large volumes of data upon the provision of forensically sound investigations. Changwei et al. (2014) examine the effects of integrating evidence relevance and legal admissibility standards through a determination of uncertainty levels.

Hoque et al. (2014) determined that, based upon use, there were two categories of network security tools, sniffing tools and scanning tools. From the perspective of this research design, the sniffing tools are seen to be useful to the forensic investigator, especially when investigating digital network forensic readiness. Therefore a sniffing tool was selected to be tested. The scanning tools, however, provide a real time view of the network and whilst this is important for network security and anomaly detection which can identify breaches, these tools provide little use to the forensic investigator and were therefore disregarded in the experiment conducted in this research. This was adopted as part of the research methodology because sniffing tools are utilized to capture and analyse network traffic, which normally consists of layer 2 frames in motion across the network architecture.

The hardware test bench design of this thesis' is based upon Kiravuo et al. (2013) and Changwei et al. (2014) papers which utilized an Ethernet infrastructure in their investigations. Kiravuo et al. (2013) identified Ethernet as being both omnipresent and yet almost unnoticed by digital network users. Whilst the research by Kiravuo et al. (2013) focused upon Ethernet related attacks and vulnerabilities from a security viewpoint, for the purposes of this research it was seen that the lack of security within Ethernet is an advantage to the digital forensic investigator, and therefore Ethernet became the hardware infrastructure focus of this research. The assertion test bench utilized within the thesis (Figure 3.1) exploits the lack of security within the Ethernet structure, which allows network packet sniffers to operate in promiscuous mode, where all Ethernet traffic can be captured and stored for future analysis. Thus this research determined that an Ethernet setup was required in order to conduct the experiment. This focus on Ethernet is also supported by Hoque et al. (2014) whereupon network packet sniffing tools, which concentrate on layer 2, are separate to network scanning tools, which concentrate on layer 3. Therefore it was concluded that the experimental assertion test environment for this thesis would require a hardware node designed specifically to capture layer 2 information transmitted upon Ethernet.

The hardware setup that the research methodology utilizes, adopts the conclusion by Kiravuo et al. (2013) that wireless Ethernet technologies were beyond the scope of the paper's investigation, because the findings could be readily applied to wireless technologies without further complicating the assertion test environment infrastructure. Therefore the hardware component of the research methodology in this thesis focuses upon wired Ethernet. The study by Hoque et al. (2014) also determined that layer 1, or physical issues were also outside of the paper's scope, because an Ethernet frame only requires a recipient's Media Access Control (MAC) information for successful transmission. Hoque et al. (2014) and Kiravuo et al. (2013) state the hardware infrastructure generalizes the hardware technologies that compromise a corporate network, and therefore, an Ethernet configuration was designed as an assertion test environment. An Ethernet segment was configured, with two switches and various hosts as shown in Figure 3.1 below.

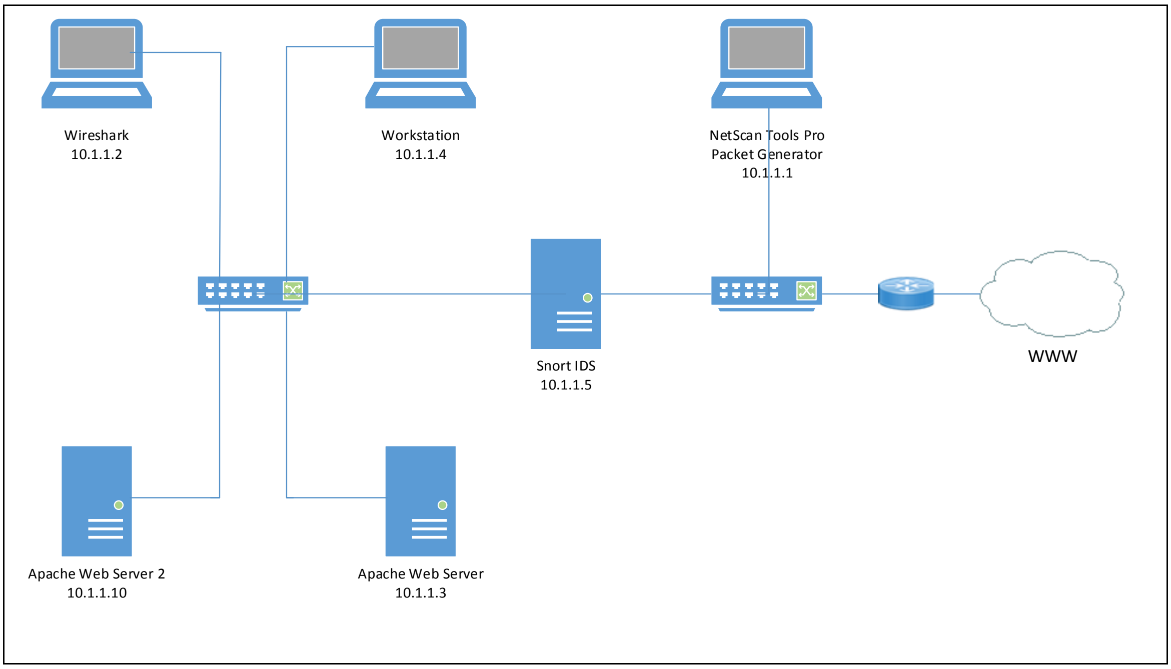


Figure 3.1: Assertion test environment

Pilli et al. (2011) established that there is a difference between network security and digital network forensics. The difference is adopted by the research methodology used in this experiment whereby network forensics do not attempt to block suspicious activity, unlike network security which attempts to control or even block information access. The model proposed by Pilli et al. (2011) suggests that a separate data capture node, specifically installed to capture data is required to be considered forensically ready. Therefore, a packet capture program with a filtering system will be implemented upon a dedicated hardware component. A laptop was found to be most suitable, because ease of mobility will allow the capture node to be placed at varying places within the assertion test environment. The experimental design of this thesis investigates whether the capturing process, under conditions of varying levels of network stress will have an impact upon data capture statistics. Therefore, it was important to generate a known number of packets and so a laptop will be configured with a programmable packet generator that will send packets to a specified node of the Ethernet segment test bench.

Many of the attacks investigated by Pilli et al. (2011) involved layer 3 or network layer manipulation, and the process proposed in the paper required a parameter list to be parsed in order to identify suspicious packets. The focus upon Layer 3 is in contrast to the findings of Kiravuo et al. (2013) which determined that layer 2 information provided the most complete information dataset to the forensic investigator seeking to establish a level of forensic readiness. Therefore, an Intrusion Detection System (IDS) was selected to be tested under varying levels of information stress. The inclusion of an IDS system to be tested was adopted from the identification of the issues that large dataset volumes presents to the forensic investigator considered by Casey et al. (2013)

Hunt and Zeadally (2012) provided the basis for the selection of the data capture tool and IDS system used to test the assertions developed. Wireshark was selected as an appropriate capture tool for testing, as Hunt et al. (2012) assert that Wireshark is a "Widely used network traffic analysis tool; forms basis of network forensic studies" (Hunt & Zeadally, 2012, p. 39). Snort was selected as an IDS for testing within the research assertion test environment, as Hunt et al. (2012) affirm that Snort is a "Widely used, popular tool for network intrusion detection and prevention, as well as for network forensic analysis" (Hunt & Zeadally, 2012, p. 39). Both tools were selected for their wide acceptance as forensic tools, which also maintains adherence to the Daubert (1993) principles of the procedure, or in this case the tools, being subject to peer review. The lack of independent testing and lack of known error rates direct this research methodology towards testing these two tools on the assertion test environment. Wireshark was therefore installed upon the data capture laptop, identified with the IP address 10.1.1.2 assigned to the Network Interface Card contained within the laptop, used to connect to the Ethernet test bench segment. Snort was installed upon a server, identified as 10.1.1.5, which is the IP address assigned to the NIC contained within the server, used to connect to the Ethernet test bench segment.

Kiravuo et al. (2013) and Pilli et al. (2011) Casey et al. (2013) identify that capturing a reduced data set, whilst increasing efficiency, may produce data with missing or erroneous information that may affect legal admissibility. Therefore, software tested upon the assertion test bench will utilise configuration settings with the minimum filtering parameters. Through the investigation of digital forensic process bottlenecks, Casey et al. (2013) identify that disk IO speeds cause the most problems, therefore the research is designed to show the potential effects of hardware limitations on digital forensic readiness and the error rates that may be produced yet have minimal effect on the node’s processing capabilities. Therefore, the assertion test environment will consider whether increasing data transmission rates will cause the digital forensic tool being tested to become less efficient.

Transmission rates, which regularly reach speeds of up to 1Gb per second on a single network segment, indicate that a fully functional forensic readiness system may require sustained hard disk write speed of 125MB, which is very close to the theoretical maximum for a mechanical hard drive. Theoretically, a Solid State Drive (SSD) can support burst speeds of up to 550MB/s on the SATA 3.0 platform, but the issue is that these theoretical read/write rates have not been subject to sustained testing in a forensically ready network scenario. Whilst the SATA 3.0 platform may be able to sustain the IO rate required to acquire live network evidence, the volume levels indicate that the SSD would soon become filled to capacity. Currently an SSD is 3 to 4 times the cost of a mechanical hard drive, and so an issue is the hardware cost determination compared to the volume of data to be captured. It is conceivable that a 1TB disk could become completely full in little less than 3 hours at a sustained capture rate of 1Gb/s, which also needs to be tested under controlled conditions in a standardized test-bench environment. Therefore, hard drive issue will be examined within the assertion test environment, where theoretical maximum data volume rates can be tested at sustained rates.

Finally, the research study addresses requirements for substantiating evidence. The basis of these requirements is formed by Changwei et al. (2014) investigating admissibility standards. The research explores the difficulties encountered in attempting to incorporate evidentiary legal aspects with the technical aspects of digital forensics. In order to establish legal credibility the study determines that it is only through establishing error rates, and therefore uncertainty levels, that have been incorporated into the research methodology in order to quantify assertions of process validity. The study by Changwei et al. (2014) used layer 3, IP source/destination information when determining which information to gather, which required a set of rules to be constructed from generic attack information. Aspects of that approach will be included in the assertions to be tested. Two major issues, as identified by Changwei et al. (2014) are to be explored: the collection of complete evidentiary data-sets, and the volatile nature of the dynamic networked environment. The assertions will be tested through controlled presentation of data onto the Ethernet test bench from a Network Packet Generator (NPG).

Therefore, packet generation delivery parameters of data volume and transmission speed have been selected to be the independent variables. Changes in the parameters of volume or speed will provide the basis of the cause and effect relationship tested within the assertion test environment. The dependent variable will be given as an error rating, calculated by determining the quantity of packets received compared to the number of packets generated. The assertion test environment will ensure that constant variables are controlled by calculating a baseline of network activity whilst the assertion test environment is at a quiescent state. Thus, error rates will be established through determination of received data packet amounts, upon changes in data transmission speed and volume levels.

3.2.2 Solution Selection

The five papers summarized in Section 3.2.1, indicate several solutions to problems described in Section 2.9 and issues that face the forensic investigator, especially when preparing proactively for a potential forensic investigation. The issues include lack of testing of digital forensic capture or sniffing tools, particularly testing for falsifiability and error rate determination. Resolving these issues requires subjecting sniffing tools and Intrusion Detection Systems to scientific rigor, as postulated by Popper (1959) and attention to legal principles such as Daubert (1993). Therefore, these tools are required to be independently tested, and error rates established, if the information is to be presented in court. There is an expectation that the use of forensic tools must produce data that is complete, accurate and able to be replicated by independent third parties.

As discussed in Section 2.10, data transmitted via Ethernet is volatile by nature, and this presents a problem with live data collection. If the data set is not acquired in a method that ensures completeness and should information be missed, then the missed information cannot be recovered after the fact. Therefore, there is a requirement for strenuous testing of network packet capture tools under various conditions to ascertain levels of functionality, not only to determine volume levels, but to establish the capabilities required to maintain levels of legal acceptability. Error rates that reveal the completeness of data captured are required to maintain acceptability in a legal environment. Evidentially sound qualities are essential to any capture system to become regarded as forensically ready in active business networking environments. There is little data available to verify levels of completeness or that show the error rates at any level. Therefore, Chapter 3 resolves to formulate a methodology that will resolve whether errors occur by generating empirical data through capturing and analyzing packet flow. The analysis will be performed on an assertion test environment to assess two commonly available, legally acceptable forensic tools.

Table 3.1: Solution selection overview

Issue from Section 2.9	Solution
Few digital forensic tools have been subjected to Daubert standards	Tools may be adapted from other intended use (see Section 3.1.1) Wireshark (see Section 3.1.2 and Section 3.1.3) SNORT (see Section 3.1.2 and Section 3.1.4) Complete datasets (see Section 3.1.2 and 3.1.4)
No comprehensive methodological approach	Taxonomy (see Section 3.1.1) Deliniation- Capture and Inspection(see Section 3.1.1 Tool testing (see Section 3.1.3 and section 3.1.4) Admissibility standards (see Section 3.1.4)
Incomplete dataset collection	SNORT (see Section 3.1.2 and Section 3.1.4) Test With protocol ICMP (see Section 3.1.2) -- Wireshark (see Section 3.1.2 and Section

	3.1.3) Test with ICMP (see Section 3.1.2) ARP (see Section 3.2.5)
Cost and Impact	Tool testing (see Section 3.1.3 and section 3.1.4)
Volume growth and technological change	Networked environment typical infrastructure (see Section 3.1.4 and Section 3.1.5) Ethernet (see Section 3.1.2 and Section 3.1.4)
Live data collection	Test bench infrastructure: Ethernet (see Section 3.1.2 and Section 3.1.4) Tools: Wireshark (see Section 3.1.2 and Section 3.1.3) SNORT (see Section 3.1.2 and Section 3.1.4)
Event reconstruction	Test protocols for completeness: ARP (see Section 3.2.5) ICMP (see Section 3.1.2)

3.2.3 Research Questions

The literature reviewed in Chapter 2 presents information that indicates there are several areas that require addressing in order to prepare a business for forensic readiness. The problems selected from the literature reviews in Section 2.10 were discussed and refined in Section 3.2.2 provide a basis for framing the research question context. The critical points identified are: that there has been little testing of falsifiability and error rates of digital forensic tools. The critical points suggest that there may be constraints that impact upon a business preparing to mitigate security incidents through establishing levels of forensic preparedness. A comprehensive Research Question has been selected that generates a problem context that may occur through large data-set sampling combined with evidential requirements of data completeness. The research question has then been broken down into several Sub-Questions to provide an in-depth analysis that will be addressed by the research direction of this thesis.

Research Question: (**RQ)** *Can the Network Management System and the Network Packet Capture tool, achieve zero errors for digital forensic purposes?*

Thus, the research goal of this thesis is to determine whether a computer system can capture relevant information systematically and comprehensively for post incident forensic presentation that complies with legal requirements of completeness.

Therefore, to satisfactorily answer the research question, a two sub-questions have been determined:

Sub-question 1: (**SQ1**)*Can a packet Management system and network packet capturing tool provide legally acceptable forensic evidence?*

Sub-question2: (**SQ2**) *Will the selected network packet Management system and packet capturing tool perform without error under load?*

3.2.4 Hypotheses

The hypotheses generated through the appraisal of the research question proposed in Section 3.2.3 are:

Hypothesis 1: (**H1**) *That a commonly used, well established and professionally acceptable packet capture tool will perform under high levels of network stress without error.*

Hypothesis 2*:* (**H2**) *That a commonly used, well established and professionally acceptable packet management tool will perform under high levels of network stress without error*

These hypotheses have been selected to define falsifiability by predicting an outcome, determined through empirical experimentation, thus complying with Popper (1959) postulation of scientific development (ref Section 2.1.1). The hypotheses are also designed to address technical versus legal requirements regarding forensic readiness as indicated by the Daubert (1993) decision that establish levels of evidentiary acceptability (ref Section 2.2.4).

3.2.5 Research Phases

The research phases selected for the investigation have been adapted from Wilsdon and Slay (2006) and the framework outlined within *general test methodology for computer forensic tools* presented by NIST (2001). The research plan consists of four phases as shown in Figure 3.2. The first phase is designed to identify software requirements and establish a standardised scenario that will model the software environment and simulate a simple business network. Phase 1 will also determine the physical requirements that will enable the testing of the software tools selected, on a hardware platform that is homogenous throughout the testing phase. Phase 1 establishes the network design that will include commonly utilised business hardware such as servers, workstations, switching and routing devices. Phase 1 will select the software that will be tested on the standardised network scenario. Phase 1 will also identify the software functionalities that will be tested upon the hardware such as webserver application and platform, routing software and platform, and workstation operating systems. Phase 1 will also determine which packet generation tool and packet management software that will be evaluated in this experiment along with the hardware and software platforms upon which these tools will be configured (NIST, 2001)

Phase 2 is designed to form testable assertions which will be tested upon the software selected in phase one, through the implementation of test cases. Each test assertion will provide a statement of behaviour that can be tested and measured and will therefore be an independent testable statement that will result in the realisation of a test plan. The creation of a test plan will then generate a test case which will apply the assertions to the environments which are to be tested. These environments will be based upon levels of usage as generated through an appropriately robust and industry standard packet generation tool that will simulate a business network that is operating at high, medium and low data utilisation levels. Thus the test case environment and the test case implementation will supply data that will furnish an acceptance spectrum as indicated by Guo, Slay and Beckett (2009) which will provide a level of acceptance rather than a binary pass or fail result (Guo, Slay, & Beckett, 2009).

Phase 3, implements the test plan upon the hardware platform to deliver generated data onto the assertion test bench network established in the first two phases in a uniform and consistent manner. Phase 3 will the test the packet capture tool and management software through comparing the packet generation volume against the packet capture volumes and the resulting empirical data will be attained.

Phase 4 will evaluate the results from the data acquired in phase three, in order to test the hypotheses postulated above. The data gathered will be used to calculate the levels of data captured as compared to the data transmitted which will test hypothesis one. The data collected will also be used to gauge whether the management system will function under stress without error, thus testing hypothesis two. The combination of this data will then be used to evaluate hypothesis three, which will assess whether there is a complete dataset captured, thus providing a level of legally acceptable forensic readiness (Wilsdon & Slay, 2006).

3.2.6 Data Map

The data map presented in Figure 3.3 shows the data interrelationships between the individual components of the proposed research methodology. The research question provided in Section 3.2.3 and associated research sub-questions listed in Section 3.2.3 are related to the research phases described in Section 3.2.5. The data collection and analysis phases are then allied with the associated research phases and to the hypotheses postulated in Section 3.2.4 of this research methodology.

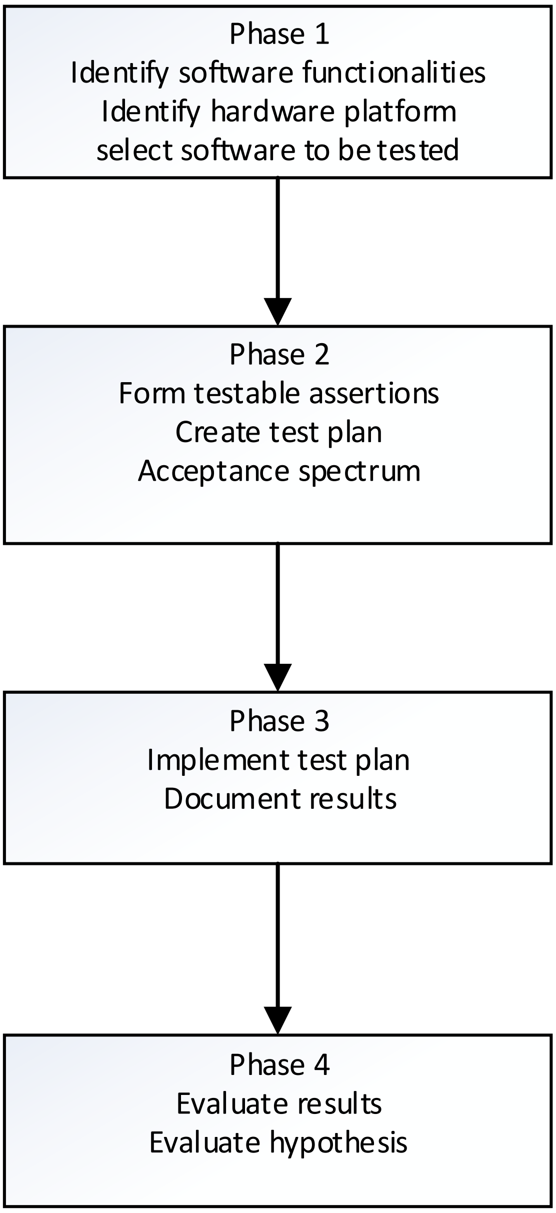


Figure 3.2: Research Phases adapted from Wildson & Slay, 2006.

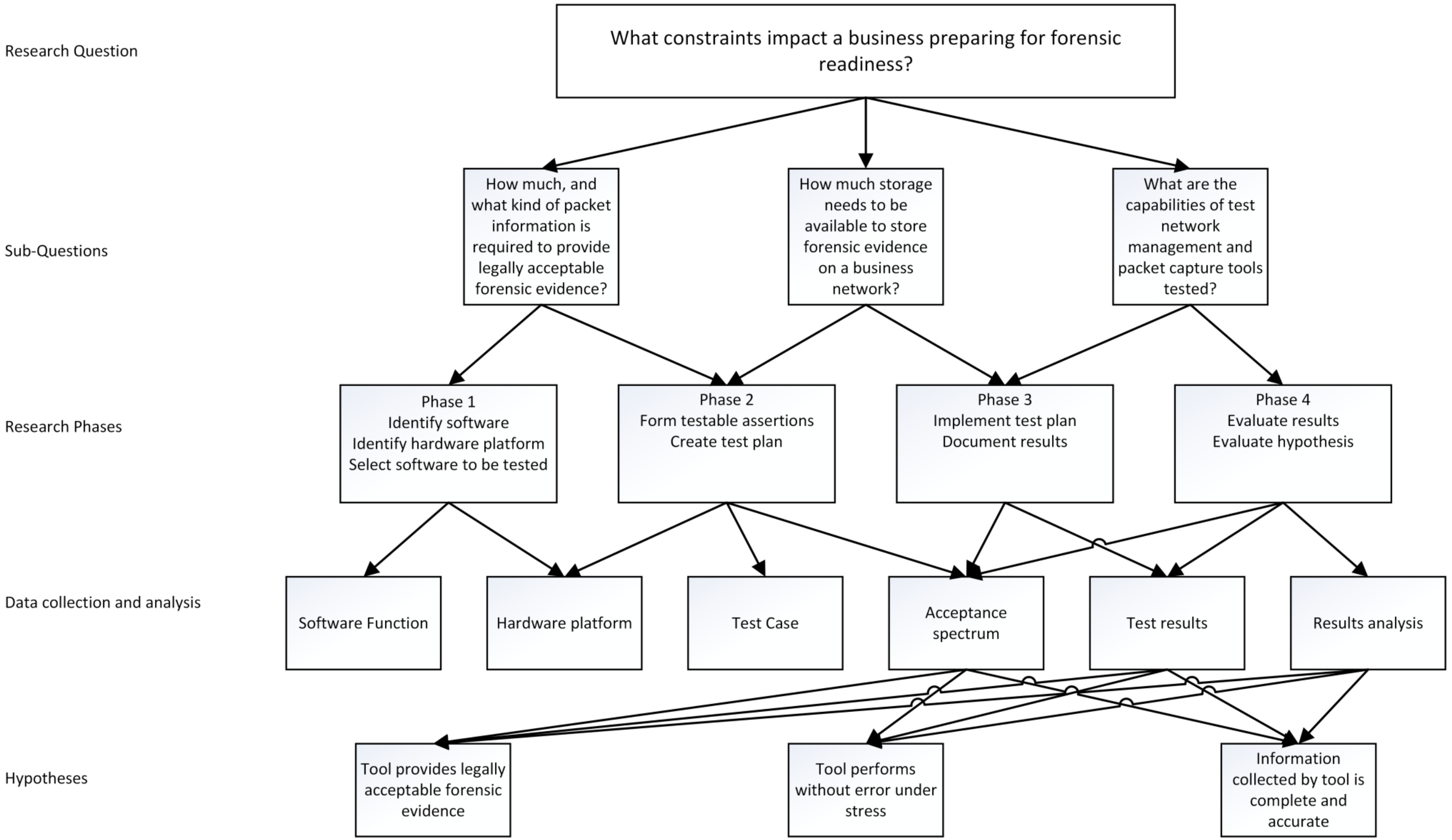

Research Question
What constraints impact a business preparing for forensic readiness?
Sub-Questions
How much, and what kind of packet information is required to provide legally acceptable forensic evidence?
How much storage needs to be available to store forensic evidence on a business network?
What are the capabilities of test network management and packet capture tools tested?
Research Phases
Phase 1 Identify software Identify hardware platform Select software to be tested
Phase 2 Form testable assertions Create test plan
Phase 3 Implement test plan Document results
Phase 4 Evaluate results Evaluate hypothesis
Data collection and analysis
Software Function
Hardware platform
Test Case
Acceptance spectrum
Test results
Results analysis
Hypotheses
Tool provides legally acceptable forensic evidence
Tool performs without error under stress
Information collected by tool is complete and accurate

Figure 3.3: Data Map

3.3 DATA REQUIREMENTS

Information determined in phase one will be used to select the software to be tested and the hardware platform upon which the test will run. Information from both phase 1 and phase 2 will be used to form testable assertions and which will then be used to create an actionable test plan. The test plan will then be implemented during phase three and the results documented. The results will then be analysed during phase four and the hypotheses will then be evaluated.

The implemented test plans will provide several different test cases that will alter variables encountered within business networks. The scenarios are to be tested upon networks that have consistent hardware and software platforms, as determined by the first two phases, but will be performing at different concentrations of network saturation. This is to simulate real world network examples, where some business networks run with very little network data transference on a day-to-day basis, and others run at an extremely high level of data transmission that approach full network saturation. Therefore the data generation, collection, processing, interpretation and presentation are discussed in greater detail within the following sub-sections.

3.3.1 Data Generation

The data generation process forms an essential component of the proposed research. NetscanTools Pro has been selected as it provides a comprehensive toolkit that includes many testing and exploration tools, not only for general network diagnostics but also for forensic use. NetscanTools Pro suite of packet generation tools will be utilised to provide a wide set of parameters that will be manipulated either manually or through a scripting process. The packet generator will be set to provide a network stream of packets that will be configured to simulate network conditions of varying levels of intensity or network saturation. Packets such as ICMP, ARP, and UDP are to be generated with specific sized payload configured to test the tools chosen. The parameters of this generated data will form the basis of a reference data set.

3.3.2 Data Collection

Initially, data will be collected to map functions of software tools that will be utilised to provide forensic readiness upon a test network in this research. The goal of this initial mapping process is to identify the requirements of legal acceptability and compare these requirements with available tools to determine if the selected tools will function as forensic information gathering tools. Once this initial mapping process has been completed, a hardware platform will be developed. This will ensure that the test will be homogenous and will ensure that the data collected will have only one factor under modification, network saturation, for each parameter under test.

Once the hardware platform and the software tools to be tested have been considered, data collected during phase two will be utilised to form testable assertions that will generate test cases. These test cases will generate data that will produce sources of information to provide the empirical data utilised in this research. The data generated will establish the data volume levels as a baseline, which will then be compared to the data volumes recorded by a packet capture tool and packet management system being tested. Thus, the requirements of this phase are to produce a circumstance that is tangible and therefore testable and each test case will investigate one testing requirement. The test cases will describe successive steps that will explain each stage of the process to be applied. The test plan developed will then be implemented through the application of the test cases, and the results will be collected and documented. These results will include data such as system configuration and procedures, and will then be analysed and evaluated.

3.3.3 Data Processing

The raw results will be generated by the implementation of the test cases. The raw results will be entered into Excel where the data will be processed to form a standardised result set. These results will then be compared to the reference data produced by the packet generation tool as a baseline. This comparison will be used to formulate an acceptance spectrum which will grade the results to produce data in the form of percentages.

3.3.4 Data Analysis and Presentation

The parameters to be tested are completeness of data collected and the amount of data processed by the tool being tested. This will be compared to network saturation levels and number of packets generated. Test volumes will be evaluated at three different levels of network saturation to provide hypothesis testing under varying intensities of network utilisation to provide a ranking of results comparing levels of network stress. Thus, the analysis of the data will focus on examining a comparative baseline of packets generated reflected upon a determination of packets gathered by the selected tools. These results will be presented in graphical and tabulated form designed to demonstrate the capability of the tool under test. It is through effective presentation of results that the greatest value from the comparative analysis will be delivered. Therefore the resultant graphical and tabulated information will present information in the form of different error rates, which will be based upon the calculations performed during the comparative analysis stage.

3.4 LIMITATIONS

The research aim of this thesis is to consider the constraints a business may encounter when becoming forensically ready. The scenarios and assertions tested present limitations that define the scope of this investigation, which is discussed in this section.

There are few tools that are specifically designed for business network forensic investigation. Therefore, the tools chosen to be investigated in this research have been adapted from information security toolsets, modified to provide business forensic readiness and therefore there are few case studies available. The tools that are to be examined have been chosen for their functionality, specifically the ability to capture streamed network traffic or to analyse network traffic and generate log information.

The hardware network test bench also presents limitations in terms of scope and size. The hardware setup was chosen to supply the minimum services to facilitate testing of a single broadcast domain with the minimum of network appliances, servers and workstations. This straightforward test bench, whilst being uncomplicated, is something that would rarely be encountered in a real-world situation.

The results that this research seeks may be seen to be limited in scope, but this has been influenced by the legal aspects of completeness and investigatiosn into error rate assignation. Legal advice has not been pursued; instead literature has been reviewed in Chapter 2 to provide an indication that data completeness and error rate investigation was germane to the topic of the research in this thesis.

3.4.1 Reliability

This sub-section examines whether these assessments produce stable and consistent results which may be affected by limitations. Thus referring to the extent that the extraneous variables are controlled within this research, it is essential to be able to determine that the changes measured are part of the methodology implementation. Therefore, any change in the independent variable must be shown to be responsible for the differences observed in the dependent variable. The primary mitigation of this reliability limitation to be employed within this research is to administer the same test twice over a period of time. Thus the results can be correlated in order to examine the results for stability over time as instituted by this form of test-retest reliability.

3.4.2 Validity

Whilst reliability is a research requirement within this thesis, it alone is not sufficient, as the test methodology must also be valid. Construct, criterion-related and sample validity are considered in this sub-section.

Construct validity, which ensures that the results actually measure what is intended to be measured are considered. Thus the measurements proposed in this research methodology are empirical, based upon quantitative measurements of data volume collection amounts contrasted to bandwidth saturation levels. This mitigates limitations that may be imposed if there were to be forensic analysis of the data collected, which would entail a qualitative aspect based upon the forensic examination.

The criterion-related validity of this proposed research may provide a level of transfer, whereupon the results may correlate with other criterion of interest in further research. Thus, the results sought through the application of this research methodology may form the basis of, or be compared to, the results pursued by future studies of network forensic readiness.

There are limitations in ensuring that a broad range of areas within the concepts of completeness and error rates under study in this research. This sampling validity limitation is to be mitigated by testing several Ethernet frame formats. This is acknowledged as a limitation due to the size and scope of this research.

3.5 CONCLUSION

Chapter 3 provides an overview of the proposed research design, including the research methodology, research question and sub-questions. Similar works from previous researchers have been studied to assess an appropriate methodology that can be adopted and integrated into the proposed research in Section 3.1. The research questions and sub-questions identified in Section 3.2.3 were derived from the selected challenges identified in Section 2.9. The challenges were extrapolated from the literature review carried out in Chapter 2. Section 3.2.4 presents the hypotheses developed through the appraisal of the research sub-questions. The hypotheses are selected to predict an outcome, therefore defining falsifiability.

The research model has been designed to demonstrate a logical progression through the four research phases, identified in Section 3.2.5 and the data map shown in

Section 3.2.6. The data requirements were presented in Section 3.3, where the data collection, analysis and presentation techniques are discussed. Finally, the research limitations, reliability and validity are discussed in Section 3.4 and potential transfer to future research is identified. Chapter 4 will present the results of the study, and the data formulated will be reviewed based upon the methodological framework developed in Chapter 3.

Chapter 4
RESEARCH FINDINGS

4.0 INTRODUCTION

Chapter 3 has defined the research methodology for investigating error rates in network management tools when applied to different intensities of packet rat In chapter 4 the findings of the application of the specified methodology from Chapter 3 is reported. The chapter is structured to acknowledge any variations that had to be made in practice while implementing the methodological specification. In Section 2 the assertions, test cases and the actual test-bed set up are reported. In Section 3 the data for each tool is reported starting with snort. And in Section 4 the wire shark results are reported. These results are discussed and further analysed in Chapter 5 where the research question is answered.

4.1 VARIATIONS IN DATA REQUIREMENTS

It is inevitable that a number of variations have been made during implementation of the original research methodology proposed in the data requirements section. The identification of these changes is significant, and it is important to describe the changes before the reporting of the findings from the research carried out in the testing phases. The following subsections outline these changes and variations to the original data requirements.

4.1.1 Data Generation.

There were few changes to this component of the experimental process. The packet generation software, NetScan Tools pro, was used to simulate network conditions of various levels of intensity up to and including network saturation. The change to the proposed outline was due to the difficulty in the formation of a reference dataset to be used as a baseline. The assertion test bench environment provided very little data, when running in a quiescent state. Therefore the data generated by the packet generator was determined to be the baseline for each of the test case scenarios that follow.

4.1.2 Data Processing

The analysis and presentation of the data gathered focused on examining an error rate based on a comparative baseline of generated packets against the amount of packets gathered by the selected tool. This comparison was to be formulated as an acceptance spectrum, which graded the results. It was upon reflection determined to be in keeping with the precepts of Popper, if this were a binary result of false or not false, or for the purposes of these test cases fail or not fail (see section 2.11).

4.1.3 Data Analysis and Presentation

The results were expressed as a percentage and presented in both tabular and graphical form. Whilst these results were originally to be presented in the form of an acceptance spectrum, when answering the research questions sub-questions and hypotheses, the binary result of fail or not fail, was used instead. This was upon consideration of the Daubert standards and the requirement for completeness, in which a non-zero error rate was determined to be significant in establishing an incomplete dataset (see Section 2.2.3 and Section 2.2.4)

4.2 ASSERTION TEST BENCH SETUP

Section 4.2 provides findings encompassing the four research phases identified in the previous section 3.2.5. Section 4.2.1 identifies the physical environment upon which the selected software is to be tested and is described in detail.

4.2.1 Physical Environment

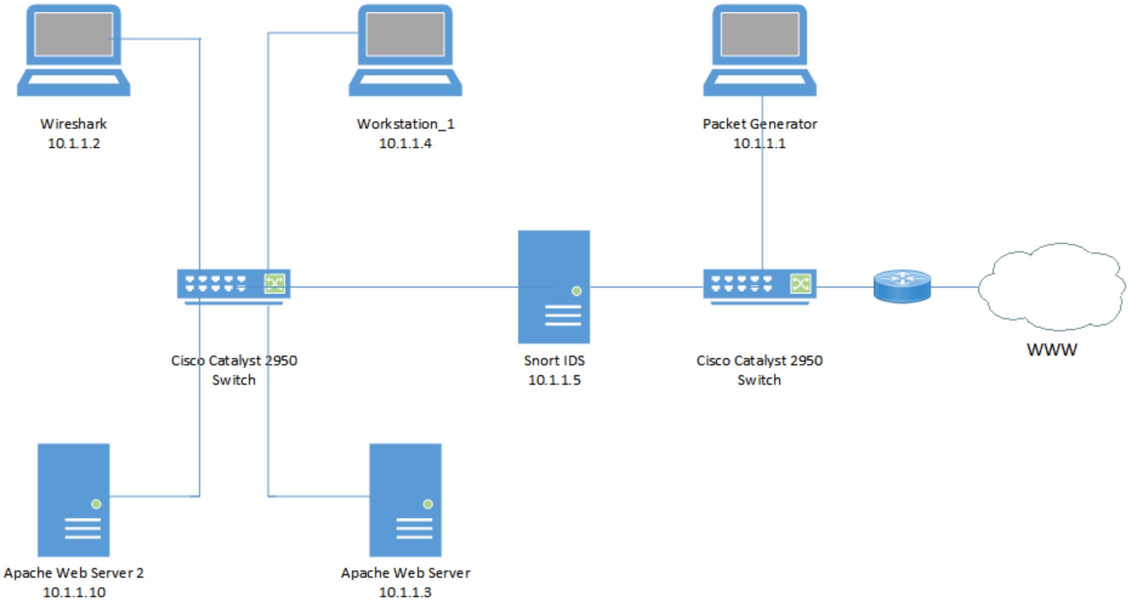


Figure 4.1: Assertion Test Bench

Section 4.2.2 then follows and identifies the software that has been selected to be tested. Section 4.2.3 then identifies the testable assertions that were formed. The testable assertions are then developed to provide requirements which generate a test plan. The test cases that are then developed are summarised in section 4.2.4 along with an identified acceptance spectrum, which is then used to deliver the final test results in section 4.2.5

The physical environment as shown in Figure 4.1 consists of the following equipment, connected via an Ethernet 10/100 network system. Each of the Linux Apache Web servers are based upon a hardware platform consisting of an Intel I5 processor running at 3.2 GHz, 8 GB of DDR3 RAM and a 1 TB hard drive. The software installed is a KALI 2.0 64-bit image dist-upgraded to 24/10/2016. The SNORT intrusion detection system is based upon hardware platform consisting of an Intel I5 processor running at 3.2 GHz, 8 GB of DDR3 RAM and a 1 TB hard drive. The SNORT software installed is version 2.9.7 .0 (build 149) utilising libpcap version 1.7.4 upon a KALI 2.0 64-bit image, dist-upgraded to 24/10/2016. The net scan tools pro packet generation software, version 2.62 is installed upon an HP 250 G3 notebook with an Intel I5/4210U processor, 4 GB DDR3 RAM, and a 500 GB hard disk drive running a 64-bit version of Windows 7 professional. Wireshark version 2.2.0–G5368C50 master 2.2 has been installed upon two HP 250 G3 notebook with an Intel I5/4210U processor, 4 GB DDR3 RAM, and a 500 GB hard disk drive running a 64-bit version of Windows 7 professional to be utilised as an Ethernet frame/packet sniffers. The workstation device consists of an HP 250 G3 notebook with an Intel I5/4210U processor, 4 GB DDR3 RAM, and a 500 GB hard disk drive running a 64-bit version of Windows 7 professional. The Ethernet network is controlled by two Cisco catalyst 2950 switches, one placed before the snort IDS and one after the snort IDS.

The 10/100 Ethernet network has been selected as an example of a common business networking infrastructure. Ethernet has become the standard for business networks as the Ethernet infrastructure is designed for easy deployment with minimal administrative overhead, as discussed in Section 2.7.3.

4.2.2 Software Tools Selected For Testing.

As discussed in section 3.2.1 above, wire shark was selected as the capture tool for testing due to wire shark being widely used as a network traffic analysis tool which forms the basis of network forensics studies (Hunt & Zeadally, 2012). The version of wire shark selected for testing is version 2.2.0–G5368C50 master 2.2. SNORT has been selected as appropriate IDS tool to be evaluated as it has also been identified as a widely used popular network intrusion detection tool.

4.2.3 Testable Assertion

As identified by the National Institute of Standards and Technology (NIST) General Test Methodology for Computer Forensics Tools, a test assertion is a testable statement that is independent, and complete that will result in a test case whereupon a statement of behaviour can be tested or measured (NIST, 2001). The following principles have been followed in creating the testable assertions identified in this subsection:

- The testable assertions address one feature at a time.
- The testable assertions be kept as simple as possible.
- The testable assertions describe features and attributes that can be measured to supply indications of success or failure.

The indications of success and failure follow the premise regarding error rates as defined by the Daubert standard, and are discussed in section 2.2.4 above. This premise, which forms a principle of Digital evidence, is to be applied whenever scientific procedure is used to prepare and uncover evidence. Thus, the development of an error rate measurement is determined to be the overarching indication of success or failure within the following assertions. Therefore the dependent variable in this instance is the error rate determination and the independent variables are the various data rates and Ethernet bandwidth saturation levels.

The following statement has been identified as the testable assertion for the research:

The Intrusion Detection and Packet Sniffing Software under test will perform with zero errors, independent on the levels of data throughput.

Therefore, it is predicted that the various rates of layer2 Ethernet frames generated will not influence the effectiveness of the software under test, generating a zero error rate from each test sample. Thus, the assertion is that the software under test will produce no errors, with the expected result being 100% complete capture.

4.2.4 Test Cases

The following test cases apply the assertions stated in section 4.2.3 above, to the environments to be tested and to specify what is to be tested. The test scenarios will envelop three requirements for each of the two software tools under test:

- The tool under test should successfully, completely and accurately process Ethernet frames.
- The tool under test should process Ethernet Frames at different data frame rates.
- That the results generated by the tools under test generate zero errors.

These requirements, therefore, are ustilised to generate the following test plan, the scope of which involves a test protocol and outlines the steps followed throughout the testing process. The test plan is then developed into test cases, which will describe the process that will test each scenario established. The test plan is therefore used to document the detailed intent and the assertion test bench requirements that will link to the functional coverage of specific tests in order to test the assertions and acquire data to establish levels of error, if any. The test plan is shown in Table 4.1.

Table 4.1: Test Plan

Test Case	Protocol	Tool	Iterations	Expected Results
001	ICMP	SNORT	2	Zero Errors
002	ICMP	SNORT	2	Zero Errors
003	ICMP	SNORT	2	Zero Errors

Test Number	Protocol	Tool	Iterations	Expected Results
004	ARP	Wireshark	2	Zero Errors
005	ARP	Wireshark	2	Zero Errors
006	ARP	Wireshark	2	Zero Errors

Test Number	Protocol	Tool	Iterations	Expected Results
007	ICMP	Wireshark	2	Zero Errors
008	ICMP	Wireshark	2	Zero Errors
009	ICMP	Wireshark	2	Zero Errors

4.3 TEST RESULTS: SNORT

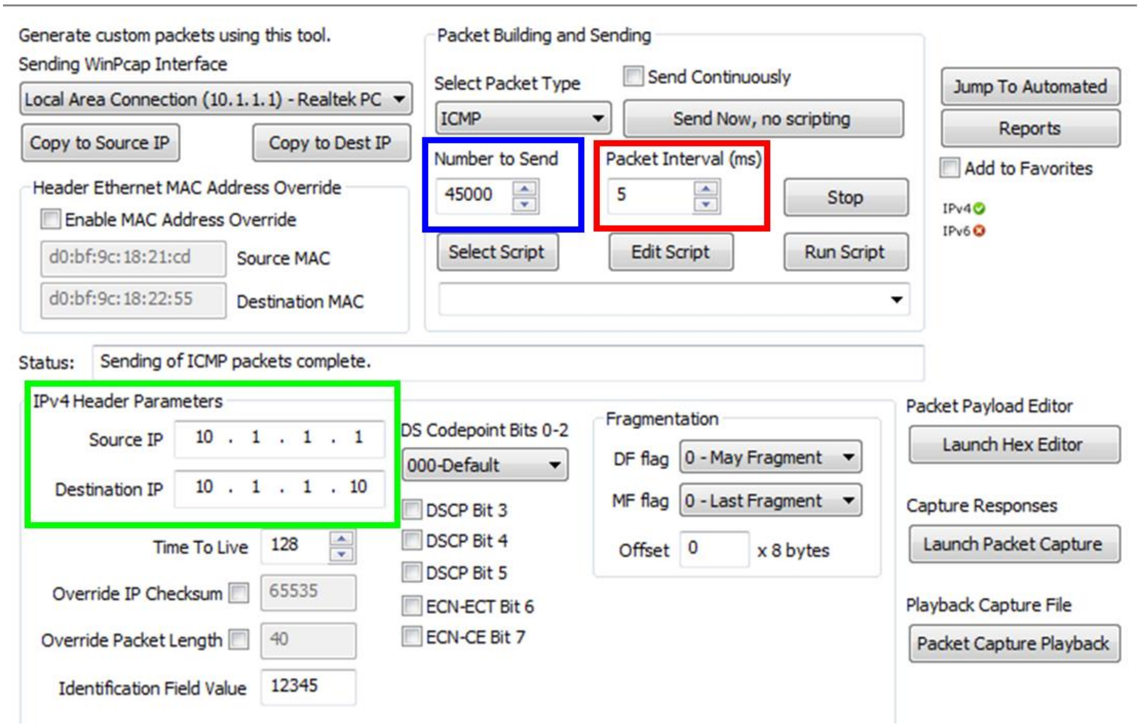
SNORT was the initial software to be tested. As discussed in Section 4.2.2, SNORT was selected as appropriate IDS tool for testing as it has also been identified as a widely used popular network intrusion detection tool. SNORT is readily available as a free download, and is included in most distributions of the Linux operating system. Snort offers Intrusion Detection Capabilities and also functions as a packet sniffer and packet logger. Thus snort generated an output file that contains details that was evaluated as discussed below in the following Sub-section.

4.3.1 SNORT

The following test cases were designed to assess the capabilities of the SNORT IDS. The NetScan Tools Pro packet generator was configured to send ICMP packets to the Apache webserver at 10.1.1.3 under various conditions of speed and with uniform data size. Each set consisted of 45,000 ICMP packets, of 1,460 Bytes. The packets were sent at 5ms, 3ms, and 1ms intervals. SNORT then generated a report file that presented packet Input / Output (I/O) information regarding details of packets received, analysed and dropped. This information was then used to generate error rates.

4.3.2 Test Case TC001

The first test case was designed to set NetScan Tools Pro to generate 45,000 ICMP packets, which were directed towards the Apache webserver, which has been assigned the IP address of 10.1.1.10. The packets were transmitted at 5ms intervals. Figure 4.2 shows the baseline configuration of the packet generator. The number of packets to be sent is shown in the blue rectangle, the IP addresses of the packet generator and the target are shown in the green rectangle and the packet interval selection is shown in the red rectangle.



Generate custom packets using this tool.

Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override

□ Enable MAC Address Override

d0:bf:9c:18:21:cd Source MAC

d0:bf:9c:18:22:55 Destination MAC

Packet Building and Sending

Select Packet Type: ICMP

□ Send Continuously

Number to Send: 45000

Packet Interval (ms): 5

Status: Sending of ICMP packets complete.

IPv4 Header Parameters

Source IP: 10 . 1 . 1 . 1

Destination IP: 10 . 1 . 1 . 10

Time To Live: 128

Override IP Checksum □ 65535

Override Packet Length □ 40

Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default

□ DSCP Bit 3

□ DSCP Bit 4

□ DSCP Bit 5

□ ECN-ECT Bit 6

□ ECN-CE Bit 7

Fragmentation

DF flag: 0 - May Fragment

MF flag: 0 - Last Fragment

Offset: 0 x 8 bytes

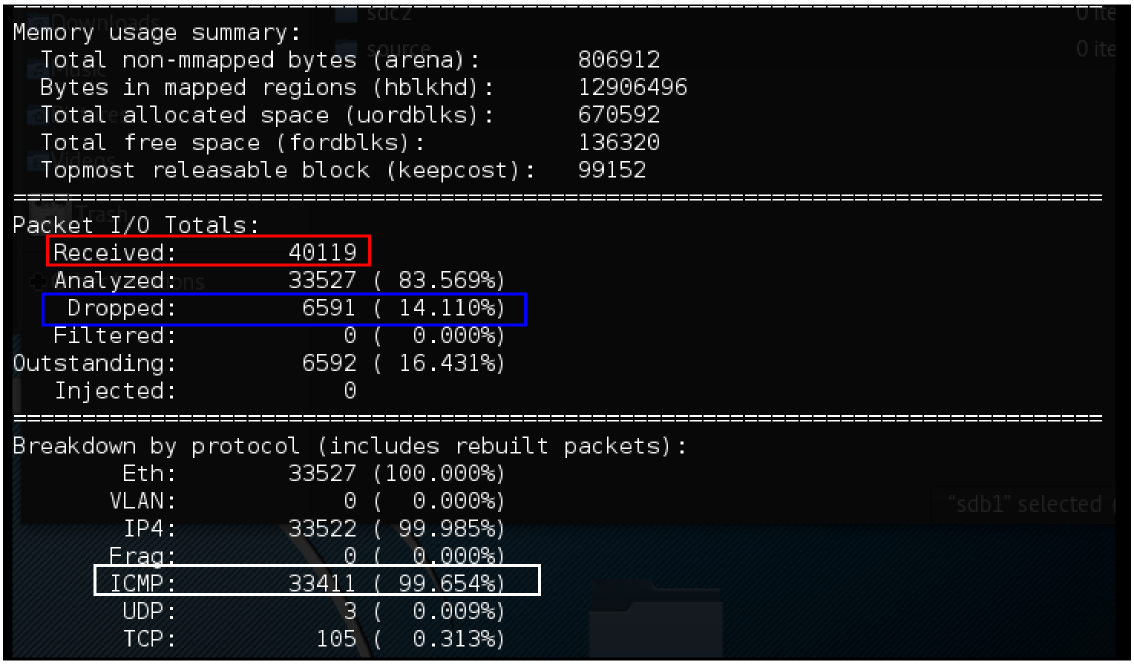
Figure 4.2: TC001 baseline

The results shown in figure 4.3 result from 45.000 ICMP packets sent at 5ms intervals. The IDS report states that 40,119 packets of the 45,000 packets sent were received. The IDS was able to analyse 33,527 packets of 40,119 packets received. The IDS dropped, or disregarded 6,591 packets and shows an outstanding amount of 6,592 packets. It is interesting to note that there is a discrepancy in the figures reported. The log information value presented for the Number of Packets Dropped is shown to be 14.11% The correct value should be 16.43% as shown in Eqn 4.1

$$^{6,591}/_{40,119} \times {}^{100}/_{1} = 16.43\% \qquad \text{(Eqn 4.1)}$$

However, the number of packets generated and sent through the IDS was 45,000, which produces the following error result (see Eqn 4.2)

$$1 - ({}^{33,527}/_{45,000}) \times {}^{100}/_{1} = 25.50\% \qquad \text{(Eqn 4.2)}$$



```
===============================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       806912
  Bytes in mapped regions (hblkhd):      12906496
  Total allocated space (uordblks):      670592
  Total free space (fordblks):           136320
  Topmost releasable block (keepcost):   99152
===============================================================================
Packet I/O Totals:
   Received:          40119
   Analyzed:          33527 ( 83.569%)
    Dropped:           6591 ( 14.110%)
   Filtered:              0 (  0.000%)
Outstanding:           6592 ( 16.431%)
   Injected:              0
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:          33527 (100.000%)
       VLAN:              0 (  0.000%)
        IP4:          33522 ( 99.985%)
       Frag:              0 (  0.000%)
       ICMP:          33411 ( 99.654%)
        UDP:              3 (  0.009%)
        TCP:            105 (  0.313%)
```

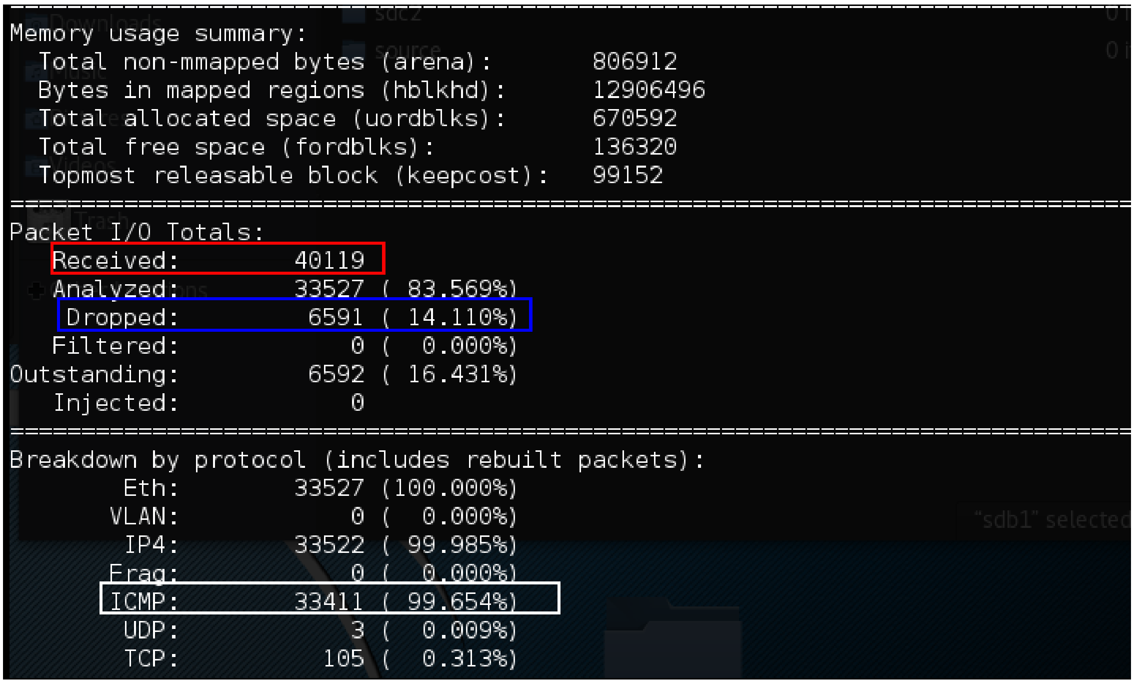
Figure 4.3: TC001-1

The results shown in Figure 4.4 result from 45,,000 ICMP packets sent at 5ms intervals. The IDS report states that 40,119 packets of the 45,000 packets sent were received. From the received packets the IDS was able to analyse 33,527 packets of 40,119 packets received. The IDS dropped, or disregarded 6,591 packets and shows an outstanding amount of 6,592 packets. It is interesting to note that there is a discrepancy in the figures reported. The log information value presented for the Number of Packets Dropped is shown to be 14.11% The correct value should be 16.43% (see Eqn 4.3).

$$^{6,591}\!/_{40,119} \times {}^{100}\!/_{1} = 16.43\% \qquad \text{(Eqn 4.3)}$$

However, the number of packets generated and sent through the IDS was 45,000, which produces an error rate of 25.50% (see Eqn 4.4).

$$1 - (^{33,527}\!/_{45,000}) \times {}^{100}\!/_{1} = 25.50\% \qquad \text{(Eqn 4.4)}$$



```
===============================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       806912
  Bytes in mapped regions (hblkhd):      12906496
  Total allocated space (uordblks):      670592
  Total free space (fordblks):           136320
  Topmost releasable block (keepcost):   99152
===============================================================================
Packet I/O Totals:
   Received:         40119
   Analyzed:         33527 ( 83.569%)
    Dropped:          6591 ( 14.110%)
   Filtered:             0 (  0.000%)
Outstanding:          6592 ( 16.431%)
   Injected:             0
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:         33527 (100.000%)
       VLAN:             0 (  0.000%)
        IP4:         33522 ( 99.985%)
       Frag:             0 (  0.000%)
       ICMP:         33411 ( 99.654%)
        UDP:             3 (  0.009%)
        TCP:           105 (  0.313%)
```

Figure 4.4: TC001-2

Table 4.2 shows the output from test cases TC001-1 and TC001-2, and displays the error rates calculated. (see Eqn 4.2 and Eqn 4.4).

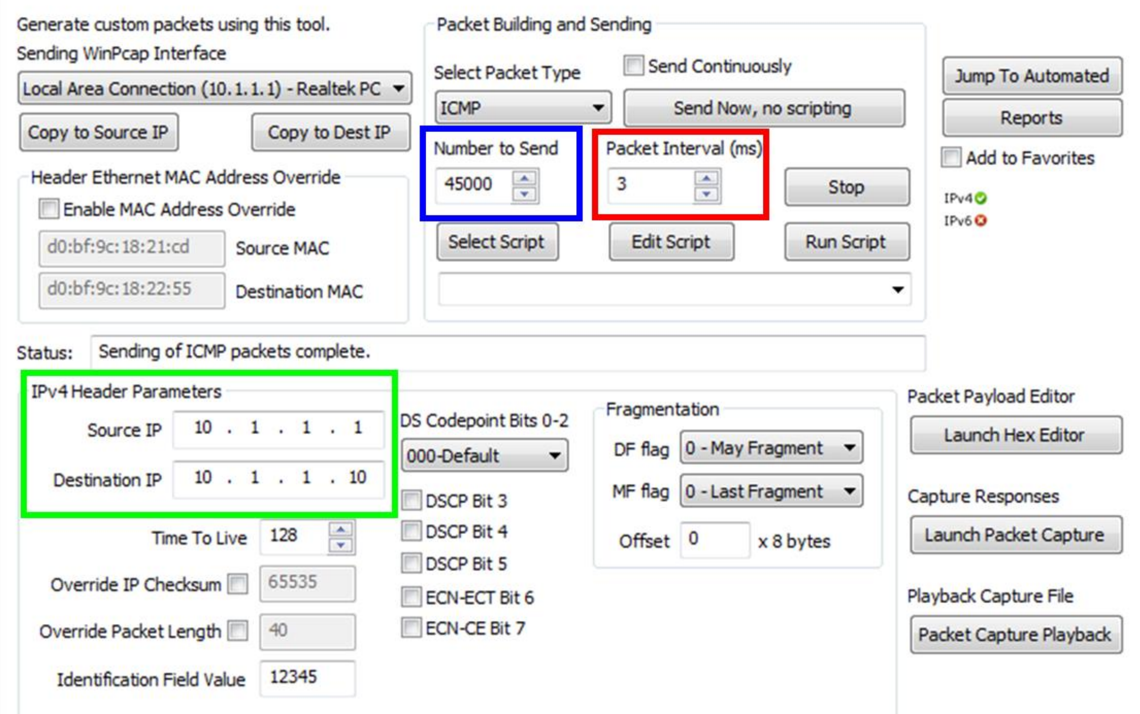
It is interesting to note, that at both iterations of this test case, TC001-1 and TC001-2 achieved the same results, with the same number of packets received and the same number of packets dropped.

Table 4.2: TC001 findings

Test Case	Packets Received	Packets Analysed	Packets Dropped	Error Rate	Error Rate	Packets Generated
TC001-1	40,119	33,527	6,591	0.2550	25.50%	45,000
TC001-2	40,119	33,527	6,591	0.2550	25.50%	45,000

4.3.3 Test Case TC002

The second test case in test case series 002 was designed to utilise NetScan Tools Pro configured to generate 45,000 ICMP packets directed towards the Apache webserver, which has been assigned the IP address of 10.1.1.10. The packets were transmitted at 3ms intervals. Figure 4.5 shows the baseline configuration of the packet generator. The number of packets to be sent is shown in the blue rectangle, the IP addresses of the packet generator and the target are shown in the green rectangle and the packet interval selection is shown in the red rectangle.



Generate custom packets using this tool.
Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override
□ Enable MAC Address Override
Source MAC: d0:bf:9c:18:21:cd
Destination MAC: d0:bf:9c:18:22:55

Packet Building and Sending
Select Packet Type: ICMP
□ Send Continuously
Number to Send: 45000
Packet Interval (ms): 3

Status: Sending of ICMP packets complete.

IPv4 Header Parameters
Source IP: 10 . 1 . 1 . 1
Destination IP: 10 . 1 . 1 . 10
Time To Live: 128
Override IP Checksum □ 65535
Override Packet Length □ 40
Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default
□ DSCP Bit 3
□ DSCP Bit 4
□ DSCP Bit 5
□ ECN-ECT Bit 6
□ ECN-CE Bit 7

Fragmentation
DF flag: 0 - May Fragment
MF flag: 0 - Last Fragment
Offset: 0 x 8 bytes

□ Add to Favorites

Figure 4.5: TC002 baseline

The results shown in Figure 4.6 result from 45,000 ICMP packets sent at 3ms intervals. The IDS report states that 40,543 packets of the 45,000 packets sent were received. From the received packets the IDS was able to analyse 15,125 packets of 40,543 packets received. The IDS dropped, or disregarded 25,416 packets and shows an outstanding amount of 25,418 packets. It is interesting to note that there is a discrepancy in the figures reported. The log information value presented for the Number of Packets

Dropped is shown to be 38.53%. The correct value should be 62.69% as given in Eqn 4.5:

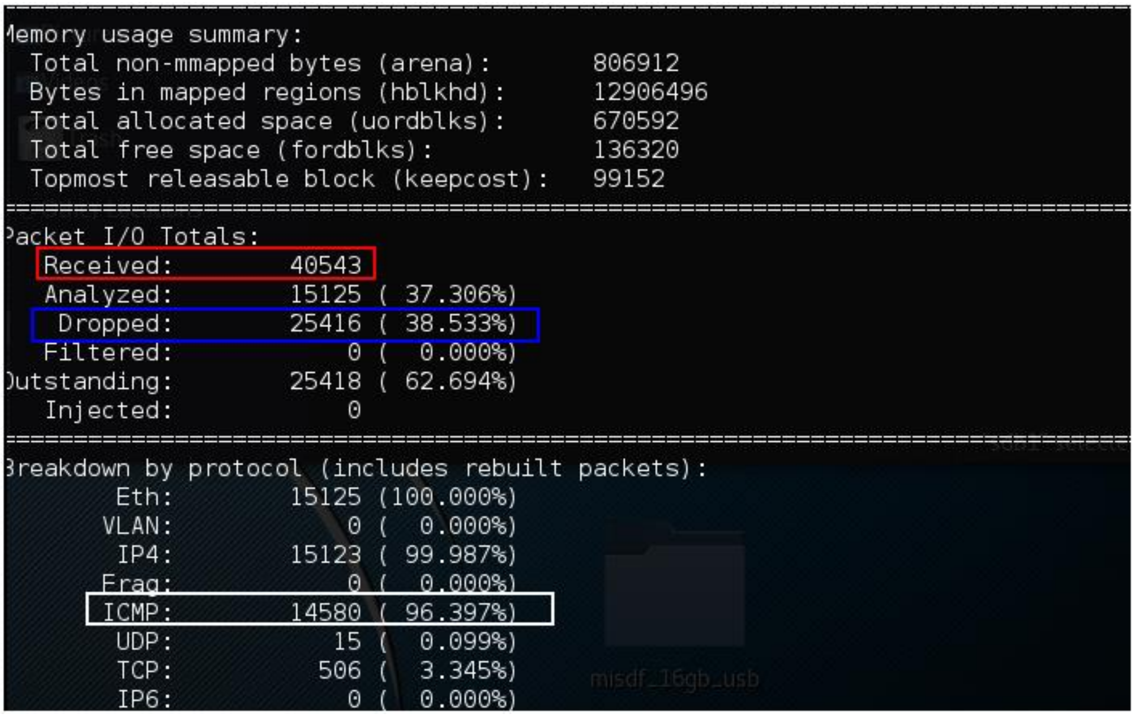
$$ {}^{25,416}\!/_{40,543} \times {}^{100}\!/_{1} = 62.69\% \tag{Eqn 4.5} $$

However, the number of packets generated and sent through the IDS was 45,000, which alters the error result to the 66.39% (see Eqn 4.6):

$$ 1 - ({}^{15,125}\!/_{45,000}) \times {}^{100}\!/_{1} = 66.39\% \tag{Eqn 4.6} $$

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.



```
===============================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       806912
  Bytes in mapped regions (hblkhd):      12906496
  Total allocated space (uordblks):      670592
  Total free space (fordblks):           136320
  Topmost releasable block (keepcost):   99152
===============================================================================
Packet I/O Totals:
   Received:          40543
   Analyzed:          15125 ( 37.306%)
    Dropped:          25416 ( 38.533%)
   Filtered:              0 (  0.000%)
Outstanding:          25418 ( 62.694%)
   Injected:              0
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:          15125 (100.000%)
       VLAN:              0 (  0.000%)
        IP4:          15123 ( 99.987%)
       Frag:              0 (  0.000%)
       ICMP:          14580 ( 96.397%)
        UDP:             15 (  0.099%)
        TCP:            506 (  3.345%)
        IP6:              0 (  0.000%)
```

Figure 4.6: TC002-1

The results shown in Figure 4.7 result from 45,000 ICMP packets sent at 3ms intervals. The IDS report states that 40,177 packets of the 45,000 packets sent were received. From the received packets the IDS was able to analyse 11,764 packets of 40,177 packets received. The IDS dropped, or disregarded 28,412 packets and shows an outstanding amount of 28,413 packets. It is interesting to note that there is a discrepancy in the figures reported. The log information value presented for the Number of Packets Dropped is shown to be 41.42%. The correct value should be 70.71% as given in Eqn 4.7:

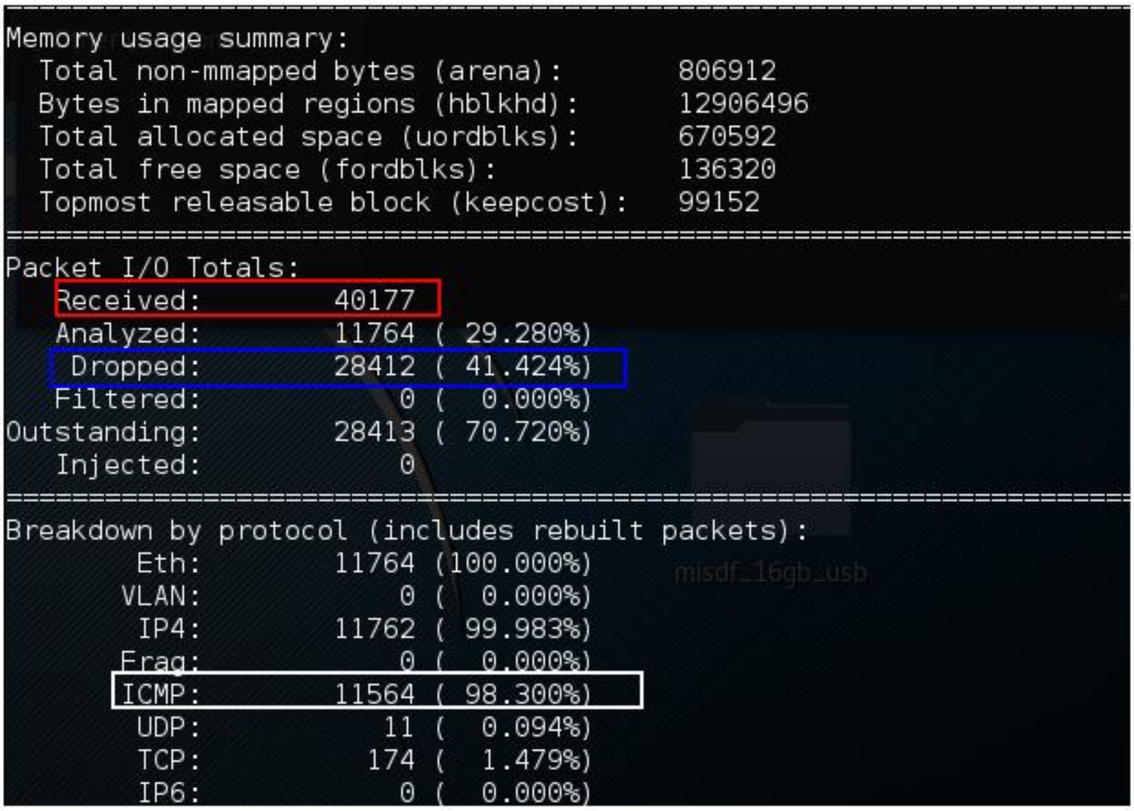
$$ {}^{28{,}412}\!/_{40{,}177} \times {}^{100}\!/_{1} = 70.71\% \qquad \text{(Eqn 4.7)} $$

However, the number of packets generated and sent through the IDS was 45,000, which alters the error result to 73.86.% (see Eqn 4.8)

$$ 1 - ({}^{11{,}764}\!/_{45{,}000}) \times {}^{100}\!/_{1} = 73.86\% \qquad \text{(Eqn 4.8)} $$

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.



```
===============================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       806912
  Bytes in mapped regions (hblkhd):      12906496
  Total allocated space (uordblks):      670592
  Total free space (fordblks):           136320
  Topmost releasable block (keepcost):   99152
===============================================================================
Packet I/O Totals:
   Received:          40177
   Analyzed:          11764 ( 29.280%)
    Dropped:          28412 ( 41.424%)
   Filtered:              0 (  0.000%)
Outstanding:          28413 ( 70.720%)
   Injected:              0
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:          11764 (100.000%)
       VLAN:              0 (  0.000%)
        IP4:          11762 ( 99.983%)
       Frag:              0 (  0.000%)
       ICMP:          11564 ( 98.300%)
        UDP:             11 (  0.094%)
        TCP:            174 (  1.479%)
        IP6:              0 (  0.000%)
```

Figure 4.7: TC002-2

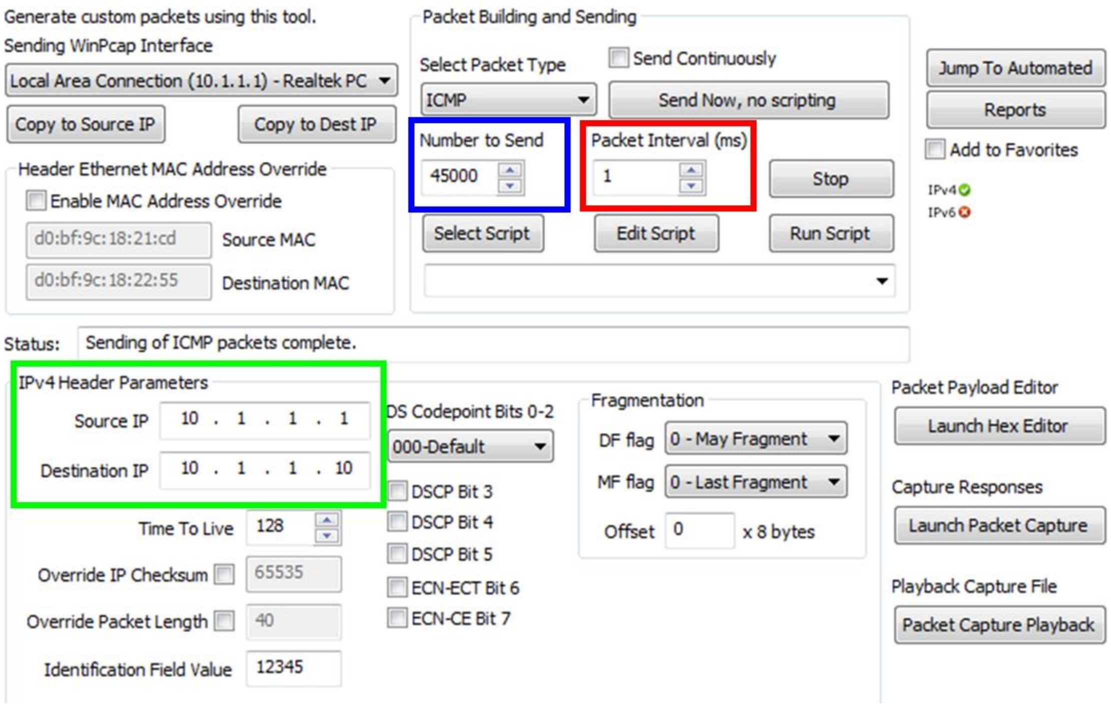
Table 4.3 shows the output from test cases TC002-1 and TC001-2, and displays the error rates calculated by Eqn 4.6 and Eqn 4.8.

Table 4.3 TC002 findings

Test Case	Packets Received	Packets Analysed	Packets Dropped	Error Rate	Error Rate	Packets Generated
TC002-1	40,543	15,125	25,416	0.6639	66.39%	45000
TC002-2	40,177	11,764	28,412	0.7386	73.86%	45000

4.3.4 Test Case TC003

The final test case, test case series TC003 was designed to utilise NetScan Tools Pro configured to generate 45,000 ICMP packets directed towards the Apache webserver, which has been assigned the IP address of 10.1.1.10. The packets were transmitted at 1ms intervals. Figure 4.8 below shows the configuration of the packet generator. The number of packets to be sent is shown in the blue rectangle, the IP addresses of the packet generator and the target are shown in the green rectangle and the packet interval selection is shown in the red rectangle.



Generate custom packets using this tool.
Sending WinPcap Interface
Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override
□ Enable MAC Address Override
d0:bf:9c:18:21:cd Source MAC
d0:bf:9c:18:22:55 Destination MAC

Packet Building and Sending
Select Packet Type: ICMP
□ Send Continuously
Number to Send: 45000
Packet Interval (ms): 1

Status: Sending of ICMP packets complete.

IPv4 Header Parameters
Source IP 10 . 1 . 1 . 1
Destination IP 10 . 1 . 1 . 10
Time To Live 128
Override IP Checksum □ 65535
Override Packet Length □ 40
Identification Field Value 12345

DS Codepoint Bits 0-2: 000-Default
□ DSCP Bit 3
□ DSCP Bit 4
□ DSCP Bit 5
□ ECN-ECT Bit 6
□ ECN-CE Bit 7

Fragmentation
DF flag: 0 - May Fragment
MF flag: 0 - Last Fragment
Offset 0 x 8 bytes

□ Add to Favorites

Figure 4.8: TC003 baseline

The results shown in Figure 4.9 below result from 45,000 ICMP packets sent at 1ms intervals. The IDS report states that 39,967 packets of the 45,000 packets sent were received. From the received packets the IDS was able to analyse 9,353 packets of 39,967 packets received. The IDS dropped, or disregarded 30,612 packets and shows an outstanding amount of 30,614 packets. It is interesting to note that there is a discrepancy in the figures reported. The log information value presented for the Number of Packets Dropped is shown to be 43.37% The correct value should be 76.60% as given in Eqn 4.9.

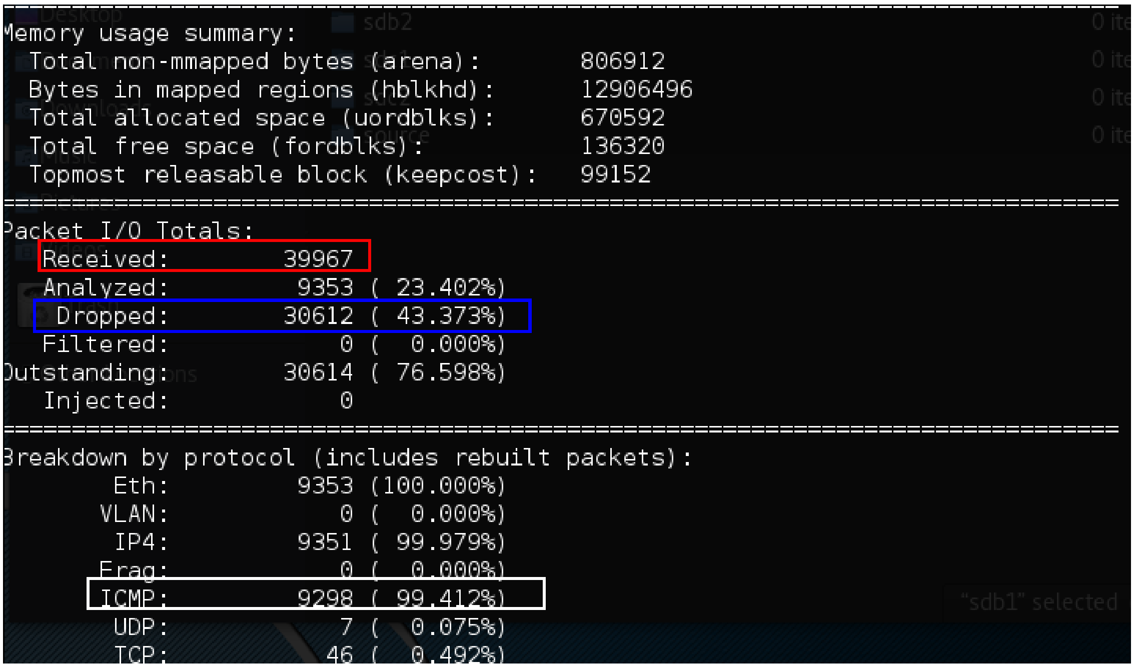
$$ {}^{30,612}\!/_{39,967} \times {}^{100}\!/_{1} = 76.60\% \qquad \text{(Eqn 4.9)} $$

However, the number of packets generated and sent through the IDS was 45,000, which alters the error result to the 79.22% (see Eqn 4.10)

$$ 1 - ({}^{9,353}\!/_{45,000}) \times {}^{100}\!/_{1} = 79.22\% \qquad \text{(Eqn 4.10)} $$

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.



```
=====================================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       806912
  Bytes in mapped regions (hblkhd):      12906496
  Total allocated space (uordblks):      670592
  Total free space (fordblks):           136320
  Topmost releasable block (keepcost):   99152
=====================================================================================
Packet I/O Totals:
   Received:          39967
   Analyzed:           9353 ( 23.402%)
    Dropped:          30612 ( 43.373%)
   Filtered:              0 (  0.000%)
Outstanding:          30614 ( 76.598%)
   Injected:              0
=====================================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:           9353 (100.000%)
       VLAN:              0 (  0.000%)
        IP4:           9351 ( 99.979%)
       Frag:              0 (  0.000%)
       ICMP:           9298 ( 99.412%)
        UDP:              7 (  0.075%)
        TCP:             46 (  0.492%)
```

Figure 4.9: TC003-1

The results shown in Figure 4.10 result from 4,.000 ICMP packets sent at 1ms intervals. The IDS report states that 40,176 packets of the 45,000 packets sent were received. From the received packets the IDS was able to analyse 9,397 packets of 40,176 packets received. The IDS dropped, or disregarded 30,778 packets and shows an outstanding amount of 30,779 packets. It is interesting to note that there is a discrepancy in the figures reported. The log information value presented for the Number of Packets Dropped is shown to be 43.38%. The correct value should be 76.61% as given in Eqn 4.11.

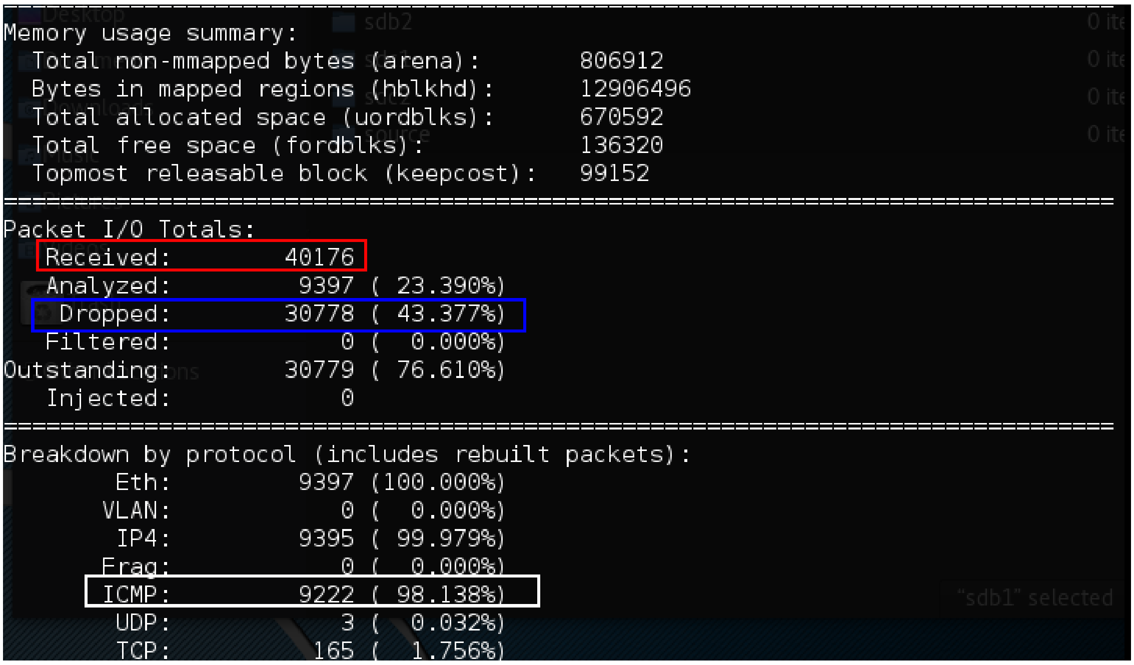
$$ {}^{30{,}778}\!/_{40{,}176} \times {}^{100}\!/_{1} \ = 76.61\% \qquad \text{(Eqn 4.11)} $$

However, the number of packets generated and sent through the IDS was 45,000, which alters the error result to the 79.12 (see Eqn 4.12).

$$ 1 - ({}^{9{,}397}\!/_{45{,}000}) \times {}^{100}\!/_{1} \ = 79.12\% \qquad \text{(Eqn 4.12)} $$

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.



```
===============================================================================
Memory usage summary:
  Total non-mmapped bytes (arena):       806912
  Bytes in mapped regions (hblkhd):      12906496
  Total allocated space (uordblks):      670592
  Total free space (fordblks):           136320
  Topmost releasable block (keepcost):   99152
===============================================================================
Packet I/O Totals:
   Received:          40176
   Analyzed:           9397 ( 23.390%)
    Dropped:          30778 ( 43.377%)
   Filtered:              0 (  0.000%)
Outstanding:          30779 ( 76.610%)
   Injected:              0
===============================================================================
Breakdown by protocol (includes rebuilt packets):
        Eth:           9397 (100.000%)
       VLAN:              0 (  0.000%)
        IP4:           9395 ( 99.979%)
       Frag:              0 (  0.000%)
       ICMP:           9222 ( 98.138%)
        UDP:              3 (  0.032%)
        TCP:            165 (  1.756%)
```

Figure 4.10: TC003-2

Table 4.4 shows the output from test cases TC003-1 and TC003-2, and displays the error rates calculated by Eqn 4.10 and Eqn 4.12.

Table 4.4: TC003 findings

Test Case	Packets Received	Packets Analysed	Packets Dropped	Error Rate	Error Rate	Packets Generated
TC003-1	39,967	9,353	30,612	0.7922	79.22%	45000
TC003-2	40,176	9,397	30,778	0.7912	79.12%	45000

The results as shown in figure 4 above indicate that at 1ms intervals, the IDS was able to analyse 23.40% of the total number of packets received, which was 39,967 packets of 45,000 total sent. The IDS dropped, or disregarded 30,612 packets out of an outstanding 30,614 packets.

4.3.5 Analysis

Table 4.2 shows the results generated from each of the test cases in the series performed above.

Table 4.5: SNORT IDS findings

Test Case	Packets Received	Packets Analysed	Packets Dropped	Error Rate	Error Rate %	Time Interval
TC001-1	40,119	33,527	6,591	0.2550	25.50%	5ms
TC001-2	40,119	33,527	6,591	0.2550	25.50%	5ms
TC002-1	40,543	15,125	25,416	0.6639	66.39%	3ms
TC002-2	40,177	11,764	28,412	0.7386	73.86%	3ms
TC003-1	39,967	9,353	30,612	0.7922	79.22%	1ms
TC003-2	40,176	9,397	30,778	0.7912	79.12%	1ms

The error rate generated was calculated by comparing the number of packets analysed to the number of packets generated. This calculation was performed for each of the test cases. The results are displayed graphically in Figure 4.11, which shows the number of packets generated and the corresponding number of packets received. Figure 4.12 shows the error rate derived from each test case in the series.

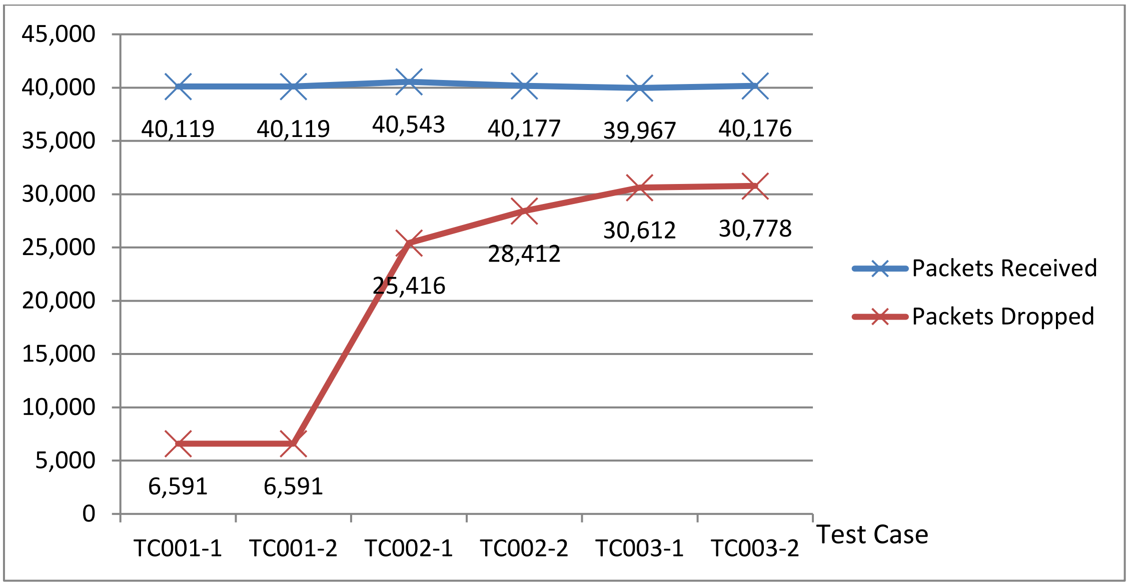


Figure 4.11: Packets dropped SNORT

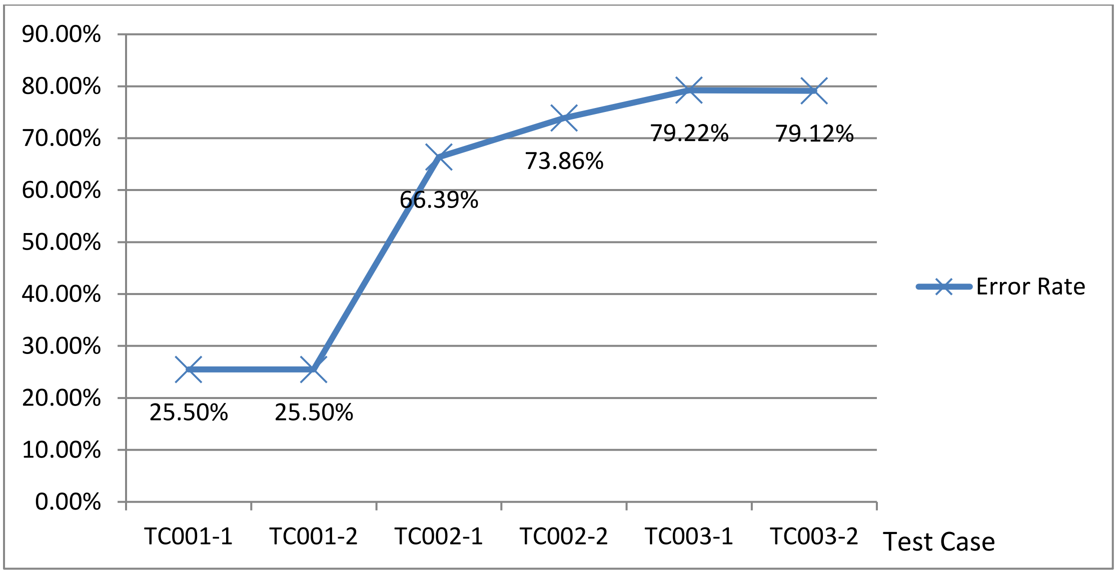


Figure 4.12: Error Rate SNORT IDS

4.4 TEST RESULTS: WIRESHARK

The second tool tested, was a popular Packet Sniffing capture tool, Wireshark. A laptop, configured with IP address 10.1.1.2 with the LAN interface set to promiscuous mode was used to capture packets using Wierhark. NetScan Tools Pro packet generator was set to broadcast two test series. The first series gathered data from a generated broadcast of 1024, 2048 and 4196 ARP requests for 10.1.1.5 onto the network. The second series gathered data from a series of 256, 512, and 1024 ICMP requests sent directly to the

Wireshark capture interface. The Wireshark interface was assigned the network address 10.1.1.2. Each capture exercise was repeated twice.

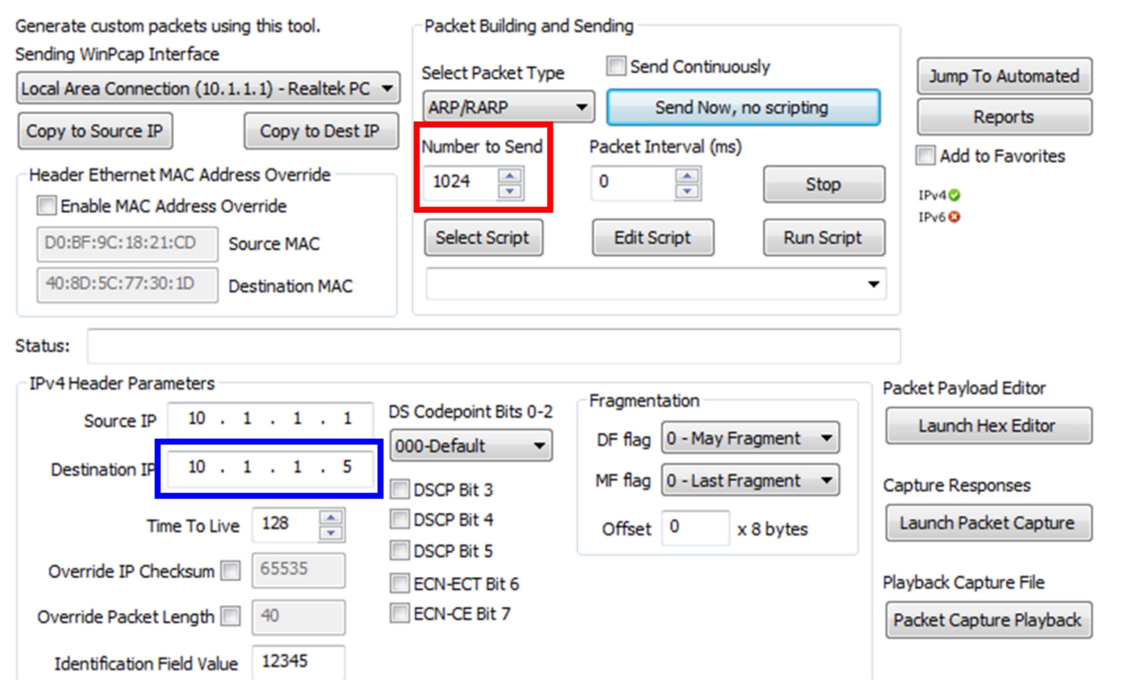
4.4.1 Wireshark ARP Request Series

The test case series TC004, TC005 and TC006 will test Wireshark with ARP packet datasets. TC004 will test Wireshark with 1,024 packets, TC005 will test Wireshark with 2,048 packets and TC005 will test Wireshark with 4,096 packets. The testable assertion in each test case is:

- The tool under test should successfully, completely and accurately process Ethernet frames.

4.4.2 Test Case 004

The first test case in this series was designed to set NetScan Tools Pro to generate 1,024 ARP packets, which were broadcast upon the LAN. Figure X below shows the configuration of the packet generator. The number of packets to be sent is shown in the red rectangle, the IP addresses of the request target of the ARP broadcast is shown in the blue rectangle. The two iterations of Test Case 004 were carried out with the same packet generation parameters.



Generate custom packets using this tool.

Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override

□ Enable MAC Address Override

Source MAC: D0:BF:9C:18:21:CD

Destination MAC: 40:8D:5C:77:30:1D

Packet Building and Sending

Select Packet Type: ARP/RARP

□ Send Continuously

Number to Send: 1024

Packet Interval (ms): 0

Status:

IPv4 Header Parameters

Source IP: 10 . 1 . 1 . 1

Destination IP: 10 . 1 . 1 . 5

Time To Live: 128

Override IP Checksum □ 65535

Override Packet Length □ 40

Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default

□ DSCP Bit 3

□ DSCP Bit 4

□ DSCP Bit 5

□ ECN-ECT Bit 6

□ ECN-CE Bit 7

Fragmentation

DF flag: 0 - May Fragment

MF flag: 0 - Last Fragment

Offset: 0 x 8 bytes

Figure 4.13: TC004 baseline

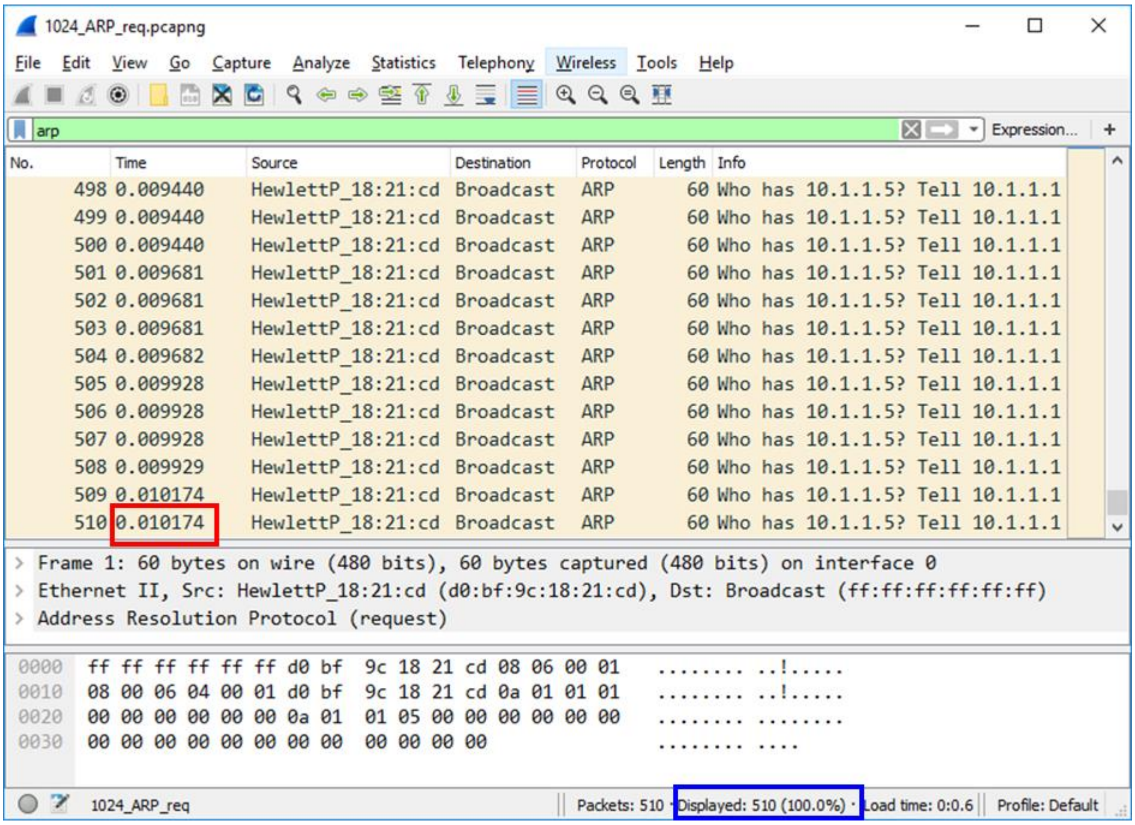
The results shown in Figure 4.14 are found by processing test case 004-1. 1,024 ARP requests were broadcast on the Ethernet LAN. The Wireshark report states that 510 packets were received of the 1,024 packets sent. This indicates that 514 packets were dropped. The results as shown in Eqn 4.13 indicate that Wireshark dropped 50.20% of the ARP request packets when 1024 ARP request packet were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC004-1 provides an error rate of 50.20% from Eqn 4.13

$$ {}^{514}\!/_{1,024} \times {}^{100}\!/_{1} = 50.20\% \qquad \text{(Eqn 4.13)} $$



```
No.      Time      Source              Destination  Protocol  Length  Info
     498 0.009440  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     499 0.009440  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     500 0.009440  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     501 0.009681  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     502 0.009681  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     503 0.009681  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     504 0.009682  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     505 0.009928  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     506 0.009928  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     507 0.009928  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     508 0.009929  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     509 0.010174  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
     510 0.010174  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)

0000  ff ff ff ff ff ff d0 bf  9c 18 21 cd 08 06 00 01   ........ ..!.....
0010  08 00 06 04 00 01 d0 bf  9c 18 21 cd 0a 01 01 01   ........ ..!.....
0020  00 00 00 00 00 00 0a 01  01 05 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....

1024_ARP_req    Packets: 510 · Displayed: 510 (100.0%) · Load time: 0:0.6    Profile: Default
```

Figure 4.14: TC004-1

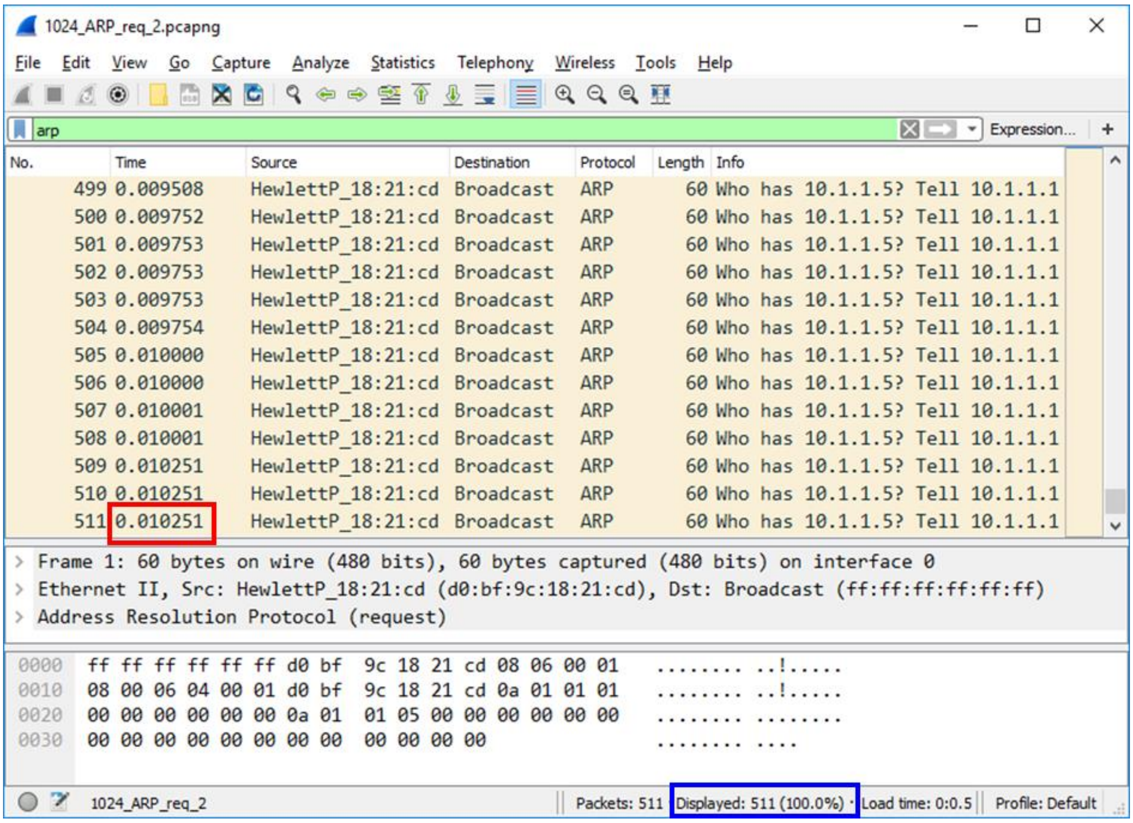
The results shown in Figure 4.15 are found by processing test case 004-2. 1,024 ARP requests were broadcast on the Ethernet LAN. The Wireshark report states that 511 packets were received of the 1,024 packets sent. This indicates that 513 packets were dropped. The results as shown in Eqn 4.14 indicate that Wireshark dropped 50.10% of the ARP request packets when 1,024 ARP request packet were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC004-2 provides an error rate of 50.10% from Eqn 4.14:

$$^{513}/_{1,024} \times {}^{100}/_{1} = 50.10\% \tag{Eqn 4.14}$$


1024_ARP_req_2.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
arp
Expression...
+

```
No.   Time      Source             Destination  Protocol  Length  Info
  499 0.009508  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  500 0.009752  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  501 0.009753  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  502 0.009753  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  503 0.009753  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  504 0.009754  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  505 0.010000  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  506 0.010000  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  507 0.010001  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  508 0.010001  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  509 0.010251  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  510 0.010251  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  511 0.010251  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)

0000  ff ff ff ff ff ff d0 bf  9c 18 21 cd 08 06 00 01   ........ ..!.....
0010  08 00 06 04 00 01 d0 bf  9c 18 21 cd 0a 01 01 01   ........ ..!.....
0020  00 00 00 00 00 00 0a 01  01 05 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....

1024_ARP_req_2 | Packets: 511 | Displayed: 511 (100.0%) | Load time: 0:0.5 | Profile: Default
```

Figure 4.15: TC004-2

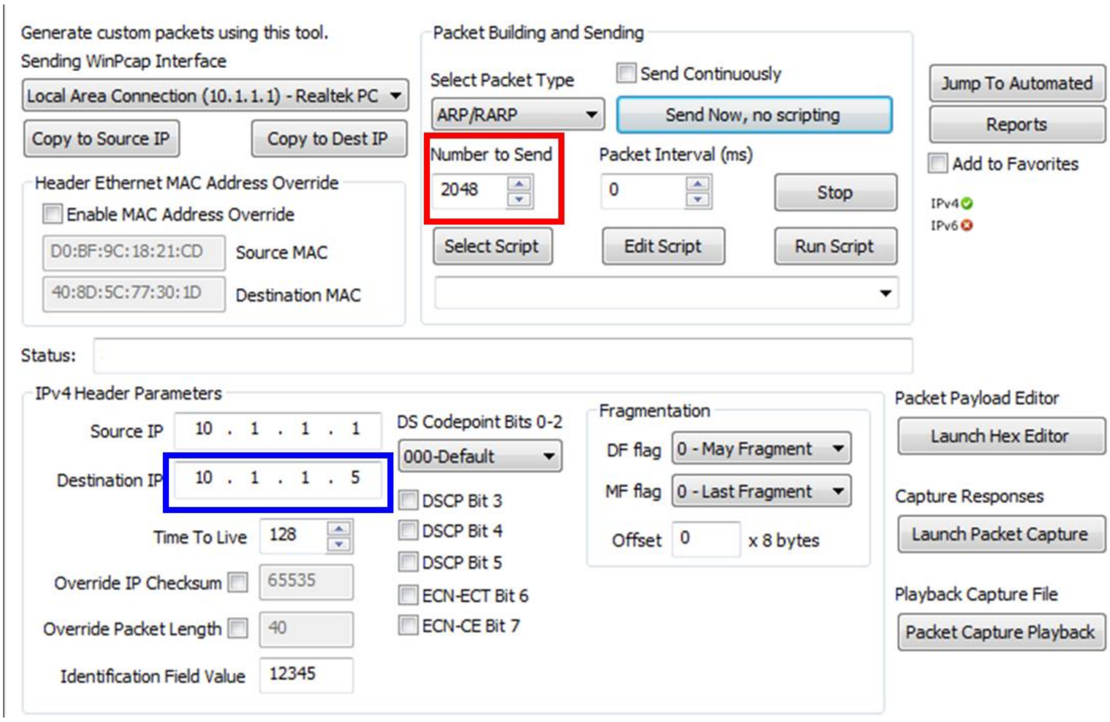
Table 4.6 shows the output from test cases TC004-1 and TC004-2, and displays the error rates calculated by Eqn 4.13 and Eqn 4.14.

Table 4.6: TC004 findings

Test Case	ARP Packets Generated	ARP Packets Captured	ARP Packets Dropped	Error Rate
TC004-1	1,024	510	514	50.20%
TC004-2	1,024	511	513	50.10%

4.4.3 Test Case 005

The second test case in this series was designed to set NetScan Tools Pro to generate 2,048 ARP packets, which were broadcast upon the LAN. Figure X below shows the configuration of the packet generator. The number of packets to be sent is shown in the red rectangle, the IP addresses of the request target of the ARP broadcast is shown in the blue rectangle. The two iterations of Test Case TC005 were carried out with the same packet generation parameters.



Generate custom packets using this tool.
Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override
□ Enable MAC Address Override
Source MAC: D0:BF:9C:18:21:CD
Destination MAC: 40:8D:5C:77:30:1D

Packet Building and Sending
Select Packet Type: ARP/RARP
□ Send Continuously
Number to Send: 2048
Packet Interval (ms): 0

Status:

IPv4 Header Parameters
Source IP: 10 . 1 . 1 . 1
Destination IP: 10 . 1 . 1 . 5
Time To Live: 128
Override IP Checksum □ 65535
Override Packet Length □ 40
Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default
□ DSCP Bit 3
□ DSCP Bit 4
□ DSCP Bit 5
□ ECN-ECT Bit 6
□ ECN-CE Bit 7

Fragmentation
DF flag: 0 - May Fragment
MF flag: 0 - Last Fragment
Offset: 0 x 8 bytes

□ Add to Favorites

Figure 4.16: TC005 baseline

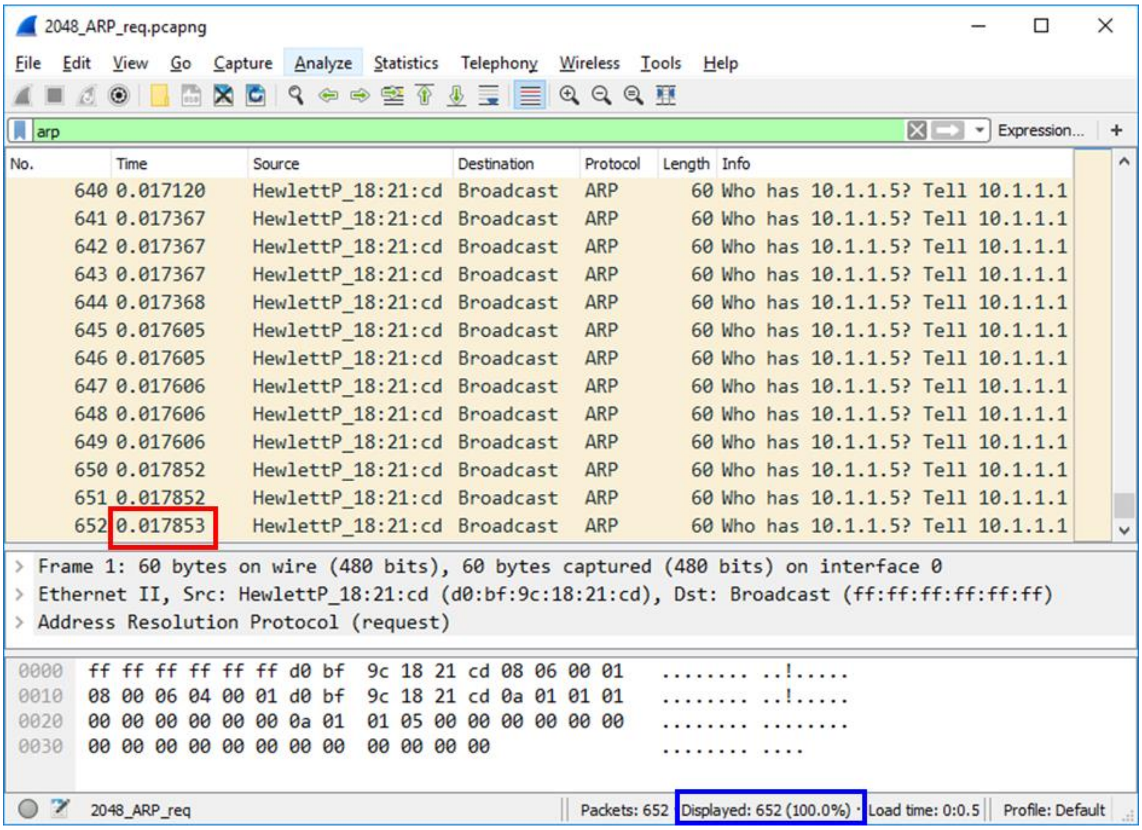
The results shown in Figure 4.17 are found by processing test case TC005-1. 2,048 ARP requests were broadcast on the Ethernet LAN. The Wireshark report states that 652 packets were received of the 2,048 packets sent. This indicates that 1,396 packets were dropped. The results as shown in Eqn 4.15 indicate that Wireshark dropped 68.16% of the ARP request packets when 2,048 ARP request packet were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC005-2 provides an error rate of 68.16% from Eqn 4.15:

$$ {}^{1{,}396}/_{2{,}048} \times {}^{100}/_{1} = 68.16\% \qquad \text{(Eqn 4.15)} $$


2048_ARP_req.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...
+

arp

```
No.   Time      Source             Destination  Protocol  Length  Info
  640 0.017120  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  641 0.017367  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  642 0.017367  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  643 0.017367  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  644 0.017368  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  645 0.017605  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  646 0.017605  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  647 0.017606  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  648 0.017606  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  649 0.017606  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  650 0.017852  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  651 0.017852  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
  652 0.017853  HewlettP_18:21:cd  Broadcast    ARP           60  Who has 10.1.1.5? Tell 10.1.1.1
```

```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)
```

```
0000  ff ff ff ff ff ff d0 bf  9c 18 21 cd 08 06 00 01   ........ ..!.....
0010  08 00 06 04 00 01 d0 bf  9c 18 21 cd 0a 01 01 01   ........ ..!.....
0020  00 00 00 00 00 00 0a 01  01 05 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

2048_ARP_req | Packets: 652 | Displayed: 652 (100.0%) | Load time: 0:0.5 | Profile: Default

Figure 4.17: TC005-1

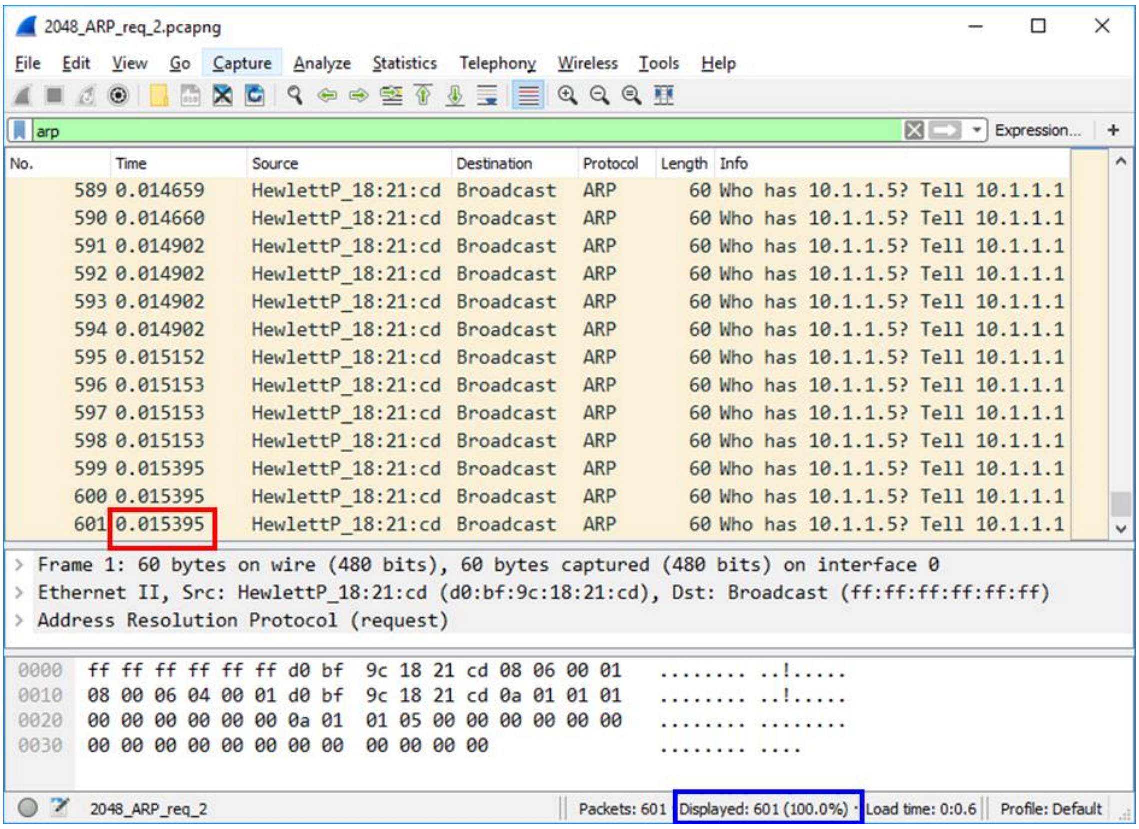
The results shown in Figure 4.18 are found by processing test case TC005-2. 2,048 ARP requests were broadcast on the Ethernet LAN. The Wireshark report states that 601 packets were received of the 2,048 packets sent. This indicates that 1,447 packets were dropped. The results as shown in Eqn 4.16 show that Wireshark dropped 70.65% of the ARP request packets when 2,048 ARP request packet were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC005-2 provides an error rate of 70.65% from Eqn 4.16:

$$ {}^{1,447}/_{2,048} \times {}^{100}/_{1} = 70.65\% \qquad \text{(Eqn 4.16)} $$



arp

```
No.   Time       Source              Destination  Protocol  Length Info
  589 0.014659   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  590 0.014660   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  591 0.014902   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  592 0.014902   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  593 0.014902   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  594 0.014902   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  595 0.015152   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  596 0.015153   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  597 0.015153   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  598 0.015153   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  599 0.015395   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  600 0.015395   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
  601 0.015395   HewlettP_18:21:cd   Broadcast    ARP       60 Who has 10.1.1.5? Tell 10.1.1.1
```

```
> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)
```

```
0000  ff ff ff ff ff ff d0 bf  9c 18 21 cd 08 06 00 01   ........ ..!.....
0010  08 00 06 04 00 01 d0 bf  9c 18 21 cd 0a 01 01 01   ........ ..!.....
0020  00 00 00 00 00 00 0a 01  01 05 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....
```

2048_ARP_req_2 | Packets: 601 · Displayed: 601 (100.0%) · Load time: 0:0.6 | Profile: Default

Figure 4.18: TC 005-2

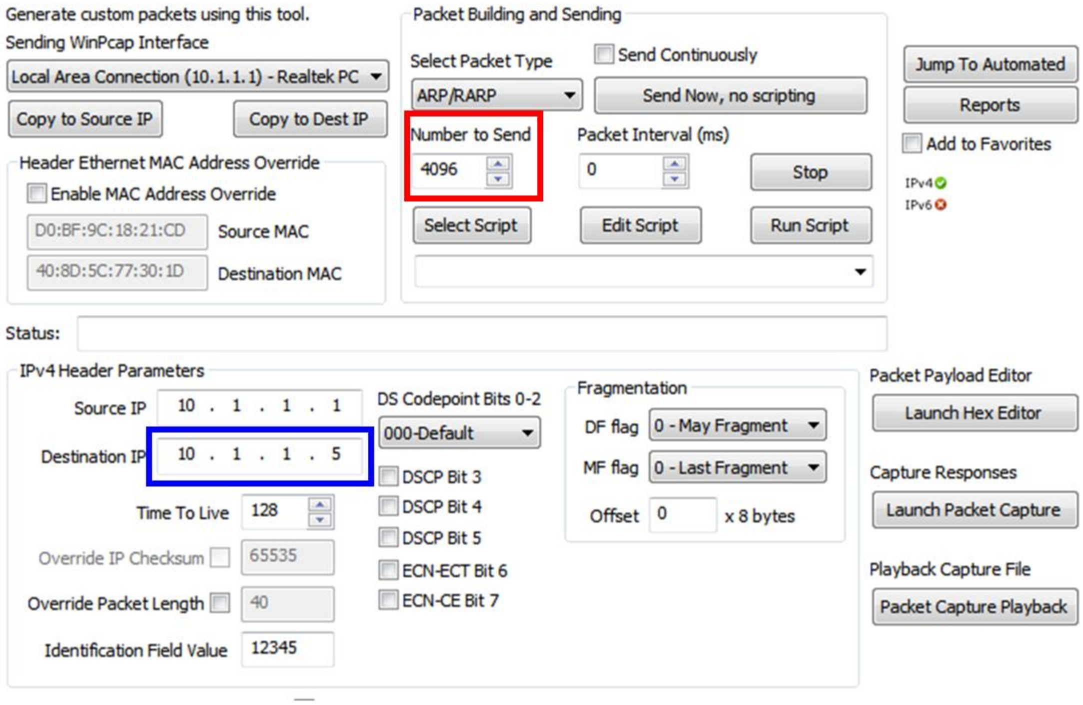
Table 4.7 shows the output from test cases TC005-1 and TC005-2, and displays the error rates calculated by Eqn 4.15 and Eqn 4.16.

Table 4.7: TC005 findings

Test Case	ARP Packets Generated	ARP Packets Captured	ARP Packets Dropped	Error Rate
TC005-1	2,048	652	1,396	68.16%
TC005-2	2,048	601	1,447	70.65%

4.4.4 Test Case 006

The final test case in this series was designed to set NetScan Tools Pro to generate 4,096 ARP packets, which were broadcast upon the LAN. Figure X below shows the configuration of the packet generator. The number of packets to be sent is shown in the red rectangle, the IP addresses of the request target of the ARP broadcast is shown in the blue rectangle. The two iterations of Test Case 006 were performed with the same packet generation parameters.



Generate custom packets using this tool.

Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override

□ Enable MAC Address Override

D0:BF:9C:18:21:CD Source MAC

40:8D:5C:77:30:1D Destination MAC

Packet Building and Sending

Select Packet Type: ARP/RARP

□ Send Continuously

Number to Send: 4096

Packet Interval (ms): 0

□ Add to Favorites

IPv4 ✓

IPv6 ✗

Status:

IPv4 Header Parameters

Source IP: 10 . 1 . 1 . 1

Destination IP: 10 . 1 . 1 . 5

Time To Live: 128

Override IP Checksum □ 65535

Override Packet Length □ 40

Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default

□ DSCP Bit 3

□ DSCP Bit 4

□ DSCP Bit 5

□ ECN-ECT Bit 6

□ ECN-CE Bit 7

Fragmentation

DF flag: 0 - May Fragment

MF flag: 0 - Last Fragment

Offset: 0 x 8 bytes

Packet Payload Editor

Capture Responses

Playback Capture File

Figure 4.19: TC006 baseline

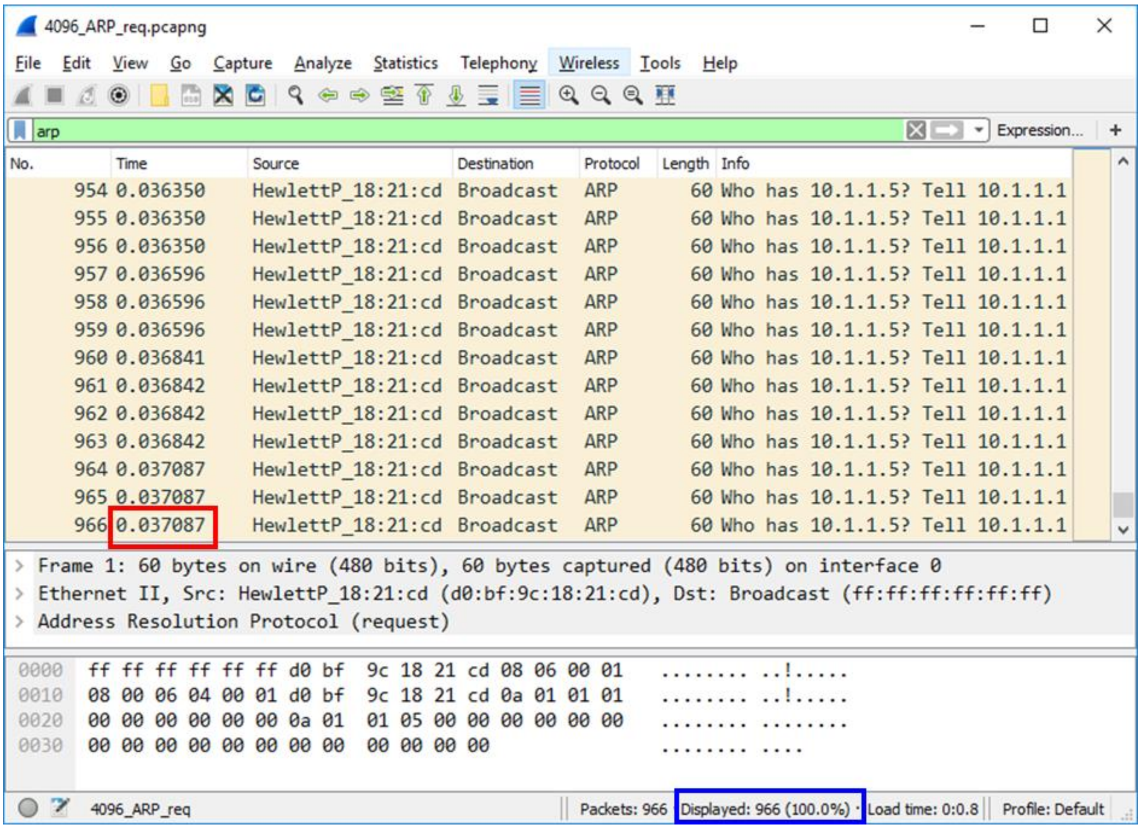
The results shown in Figure 4.20 are found by processing test case TC006-1. 4,096 ARP requests were broadcast on the Ethernet LAN. The Wireshark report states that 966 packets were received of the 4,096 packets sent. This indicates that 3,130 packets were dropped. The results as shown in Eqn 4.17 indicate that Wireshark dropped 76.42% of the ARP request packets when 4,096 ARP request packet were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC006-1 provides an error rate of 76.42% from Eqn 4.17:

$$ {}^{3{,}130}/_{4{,}096} \times {}^{100}/_{1} = 76.42\% \qquad \text{(Eqn 4.17)} $$



arp

```
No.   Time      Source             Destination  Protocol  Length Info
  954 0.036350  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  955 0.036350  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  956 0.036350  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  957 0.036596  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  958 0.036596  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  959 0.036596  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  960 0.036841  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  961 0.036842  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  962 0.036842  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  963 0.036842  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  964 0.037087  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  965 0.037087  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1
  966 0.037087  HewlettP_18:21:cd  Broadcast    ARP           60 Who has 10.1.1.5? Tell 10.1.1.1

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)

0000  ff ff ff ff ff ff d0 bf  9c 18 21 cd 08 06 00 01   ........ ..!.....
0010  08 00 06 04 00 01 d0 bf  9c 18 21 cd 0a 01 01 01   ........ ..!.....
0020  00 00 00 00 00 00 0a 01  01 05 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....

4096_ARP_req    Packets: 966 · Displayed: 966 (100.0%) · Load time: 0:0.8    Profile: Default
```

Figure 4.20: TC006-1

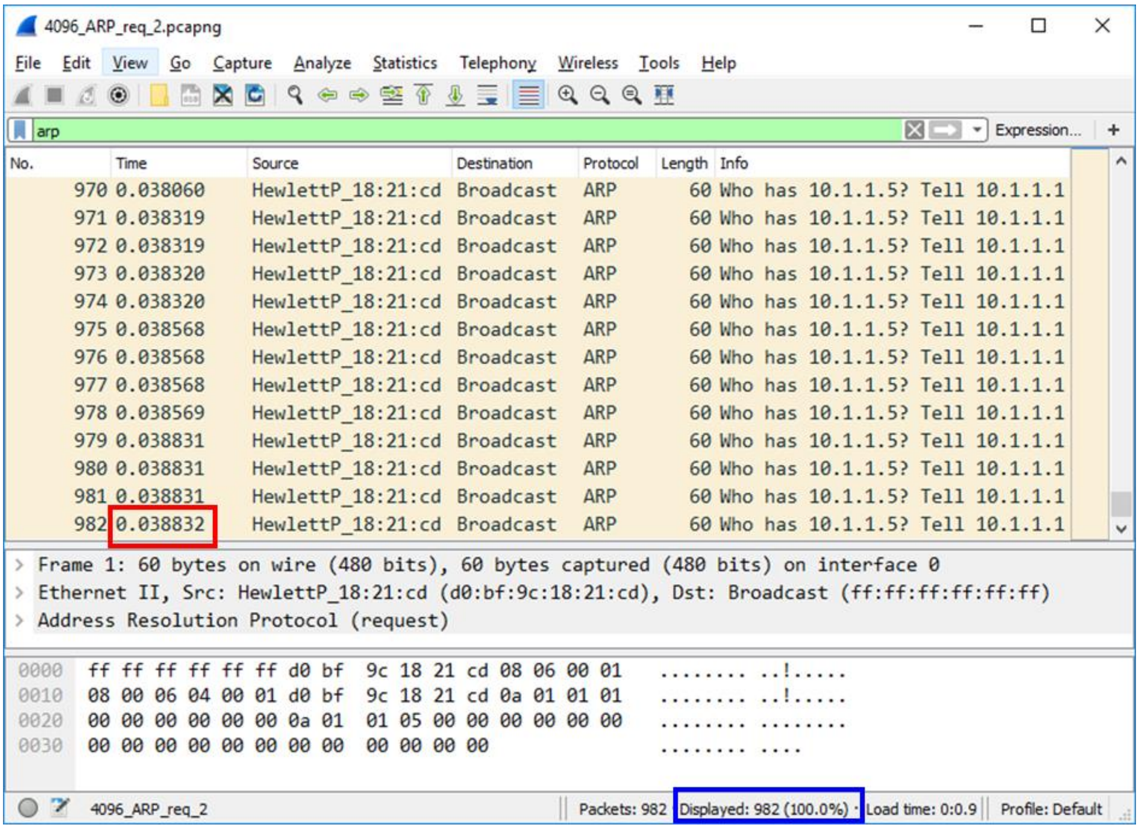
The results shown in Figure 4.21 are found by processing test case TC006-2. 4,096 ARP requests were broadcast on the Ethernet LAN. The Wireshark report states that 982 packets were received of the 4,096 packets sent. This indicates that 3,114 packets were dropped. The results as shown in Eqn 4.18 indicate that Wireshark dropped 76.03% of the ARP request packets when 4,096 ARP request packet were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC006-2 provides an error rate of 76.03% from Eqn 4.18:

$$^{3,114}/_{4,096} \times {}^{100}/_{1} = 76.03\% \qquad \text{(Eqn 4.18)}$$


4096_ARP_req_2.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
arp
Expression...
+

```
No.  Time      Source             Destination Protocol Length Info
 970 0.038060  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 971 0.038319  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 972 0.038319  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 973 0.038320  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 974 0.038320  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 975 0.038568  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 976 0.038568  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 977 0.038568  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 978 0.038569  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 979 0.038831  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 980 0.038831  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 981 0.038831  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1
 982 0.038832  HewlettP_18:21:cd  Broadcast   ARP      60 Who has 10.1.1.5? Tell 10.1.1.1

> Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits) on interface 0
> Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
> Address Resolution Protocol (request)

0000  ff ff ff ff ff ff d0 bf  9c 18 21 cd 08 06 00 01   ........ ..!.....
0010  08 00 06 04 00 01 d0 bf  9c 18 21 cd 0a 01 01 01   ........ ..!.....
0020  00 00 00 00 00 00 0a 01  01 05 00 00 00 00 00 00   ........ ........
0030  00 00 00 00 00 00 00 00  00 00 00 00               ........ ....

4096_ARP_req_2    Packets: 982 · Displayed: 982 (100.0%) · Load time: 0:0.9    Profile: Default
```

Figure 4.21: TC006-2

Table 4.8 shows the output from test cases TC006-1 and TC006-2, and displays the error rates calculated by Eqn 4.17 and Eqn 4.18.

Table 4.8: TC006 findings

Test Case	ARP Packets Generated	ARP Packets Captured	ARP Packets Dropped	Error Rate
TC006-1	4,096	966	3,130	76.42%
TC006-2	4,096	982	3,114	76.03%

4.4.5 Analysis

The error rate generated was calculated as a percentage of number of packets dropped as compared to the number of packets generated for each of the packet generation levels tested. The complete data set from the 6 tests in the series are given in Table 4.9.

Table 4.9: Wireshark ARP findings

Test Case	ARP Packets Generated	ARP Packets Captured	ARP Packets Dropped	Error Rate
TC004-1	1,024	510	514	50.20%
TC004-2	1,024	511	513	50.10%
TC005-1	2,048	652	1,396	68.16%
TC005-2	2,048	601	1,447	70.65%
TC006-1	4,096	966	3,230	76.42%
TC006-2	4,096	982	3,114	76.03%

The results are displayed graphically in Figure 4.22, which shows the number of packets generated and the corresponding number of packets received. Figure 4.23 shows the error rate derived from each test case in the series.

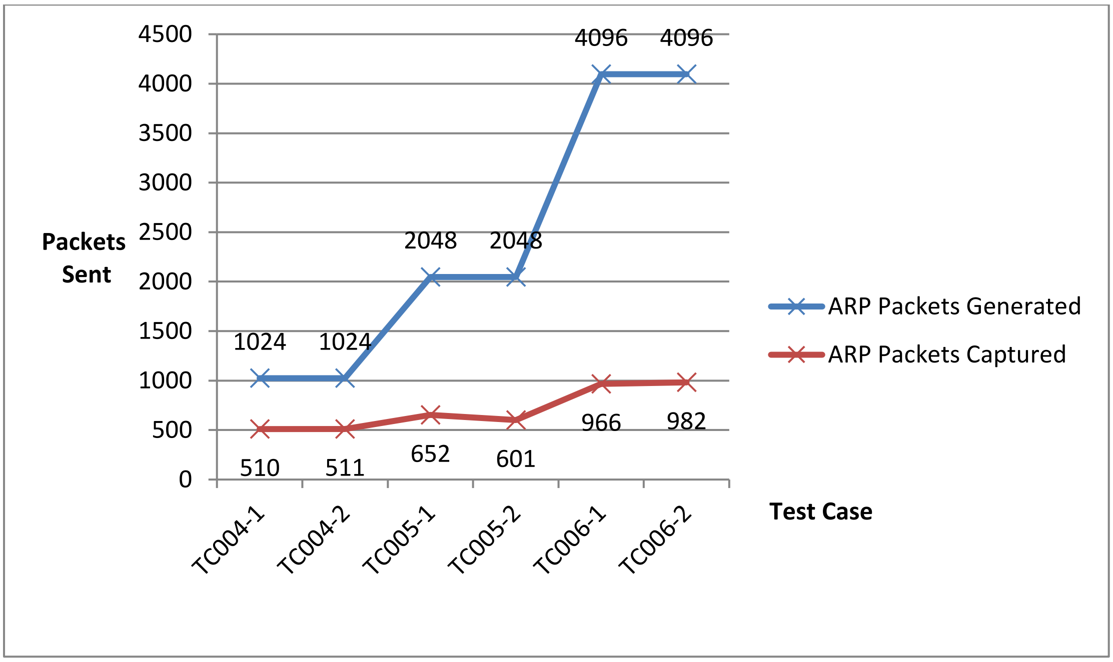


Figure 4.22: Number of packets dropped Wireshark ARP

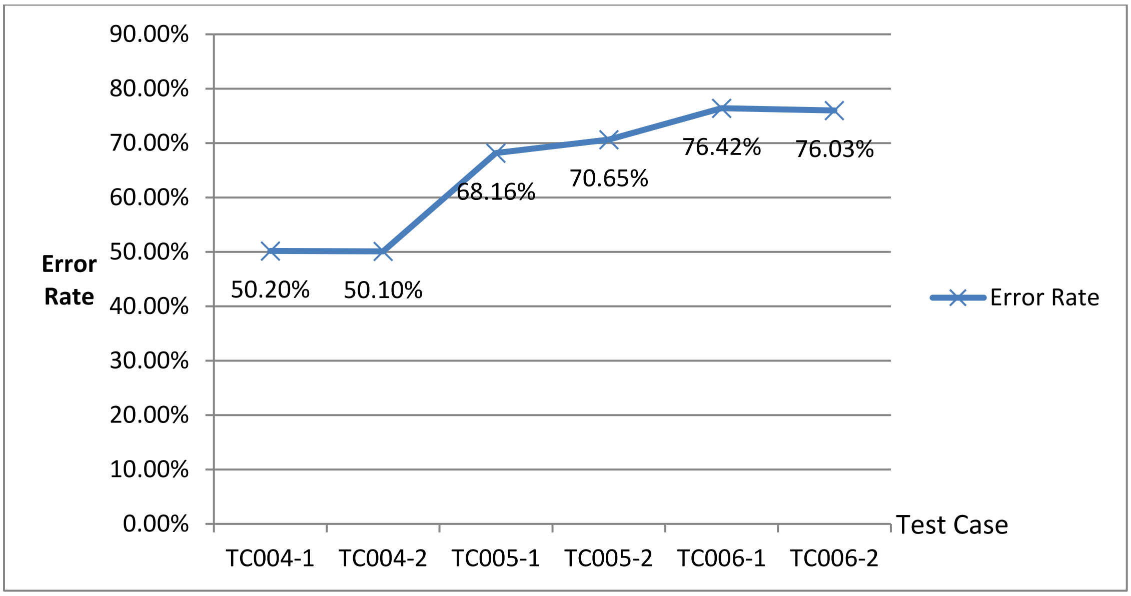


Figure 4.23: Error Rate Wireshark ARP

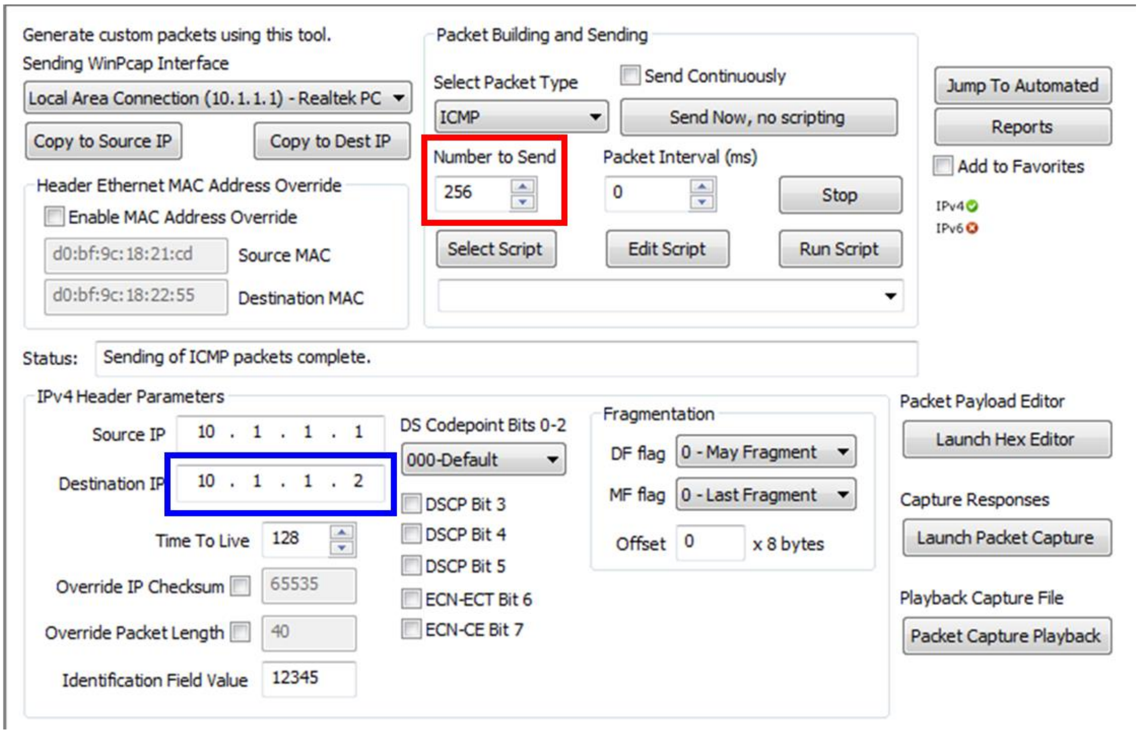
4.4.6 Wireshark ICMP Request Series

The test case series TC007, TC008 and TC009 will test Wireshark with ICMP echo request packet datasets. TC007 will test Wireshark with 256 packets, TC008 will test Wireshark with 512 packets and TC009 will test Wireshark with 1,024 packets. The testable assertion in each test case is:

- The tool under test should successfully, completely and accurately process Ethernet frames.

4.4.7 Test Case 007

The first test case in this series was designed to set NetScan Tools Pro to generate 256 ICMP echo requests. Each request was sent directly to the Wireshark capture device with the IP address of 10.1.1.2 upon the Assertion Test LAN. Figure X below shows the configuration of the packet generator. The number of packets to be sent is shown in the red rectangle, the IP addresses of the target of the ICMP request is shown in the blue rectangle. The two iterations of Test Case 007 were performed with the same packet generation parameters.



Generate custom packets using this tool.
Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override
□ Enable MAC Address Override
Source MAC: d0:bf:9c:18:21:cd
Destination MAC: d0:bf:9c:18:22:55

Packet Building and Sending
Select Packet Type: ICMP
□ Send Continuously
Number to Send: 256
Packet Interval (ms): 0

Status: Sending of ICMP packets complete.

IPv4 Header Parameters
Source IP: 10 . 1 . 1 . 1
Destination IP: 10 . 1 . 1 . 2
Time To Live: 128
Override IP Checksum □ 65535
Override Packet Length □ 40
Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default
□ DSCP Bit 3
□ DSCP Bit 4
□ DSCP Bit 5
□ ECN-ECT Bit 6
□ ECN-CE Bit 7

Fragmentation
DF flag: 0 - May Fragment
MF flag: 0 - Last Fragment
Offset: 0 x 8 bytes

□ Add to Favorites

Figure 4.24: TC007 baseline

The results shown in Figure 4.25 are found by processing test case TC007-1. 256 ICMP echo requests were broadcast on the Ethernet LAN. The Wireshark report states that 214 packets were received of the 256 packets sent. This indicates that 42 packets were dropped. The results as shown in Eqn 4.19 indicate that Wireshark dropped 16.41% of the ICMP echo request packets when 256 ICMP echo request packets were sent onto the network.

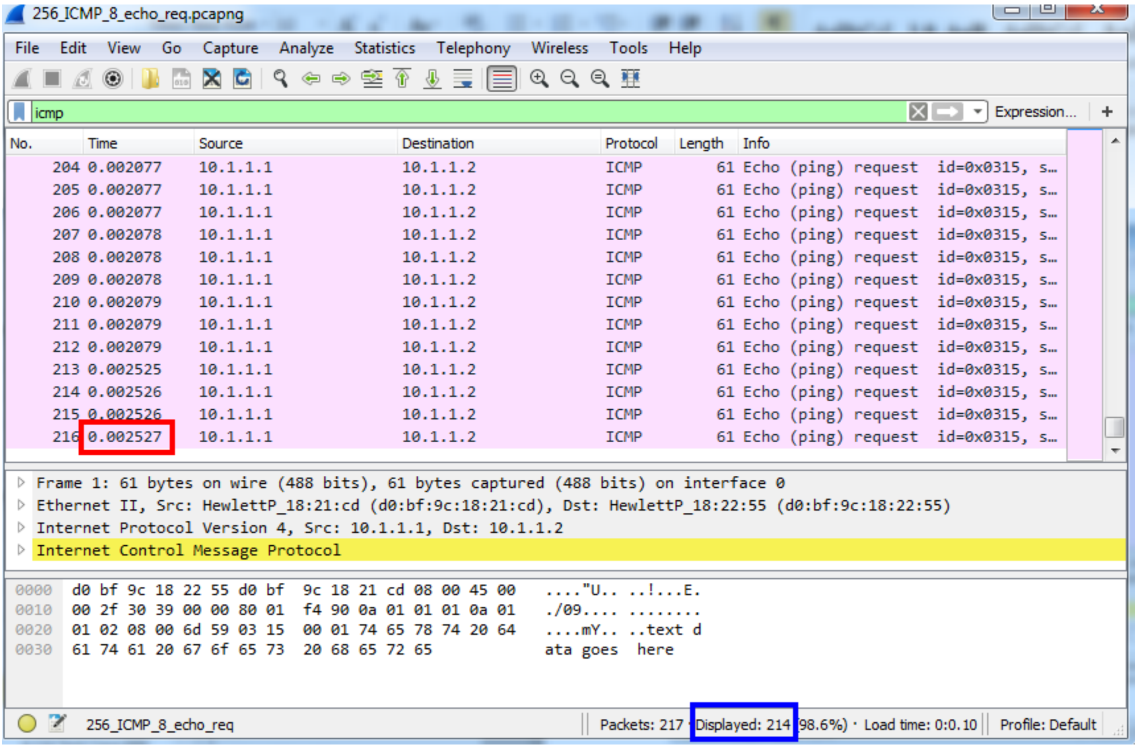
Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC007-1 provides an error rate of 16.41% from Eqn 4.19:

$$^{42}/_{256} \times {}^{100}/_{1} = 16.41\% \qquad \text{(Eqn 4.19)}$$

.



256_ICMP_8_echo_req.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression…
+

```
No.   Time      Source     Destination  Protocol  Length  Info
  204 0.002077  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  205 0.002077  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  206 0.002077  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  207 0.002078  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  208 0.002078  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  209 0.002078  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  210 0.002079  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  211 0.002079  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  212 0.002079  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  213 0.002525  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  214 0.002526  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  215 0.002526  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
  216 0.002527  10.1.1.1   10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
```

```
▷ Frame 1: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface 0
▷ Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: HewlettP_18:22:55 (d0:bf:9c:18:22:55)
▷ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2
▷ Internet Control Message Protocol
```

```
0000  d0 bf 9c 18 22 55 d0 bf  9c 18 21 cd 08 00 45 00   ...."U.. ..!...E.
0010  00 2f 30 39 00 00 80 01  f4 90 0a 01 01 01 0a 01   ./09.... ........
0020  01 02 08 00 6d 59 03 15  00 01 74 65 78 74 20 64   ....mY.. ..text d
0030  61 74 61 20 67 6f 65 73  20 68 65 72 65            ata goes  here
```

```
256_ICMP_8_echo_req    Packets: 217 · Displayed: 214 (98.6%) · Load time: 0:0.10    Profile: Default
```

Figure 4.25:TC007-1

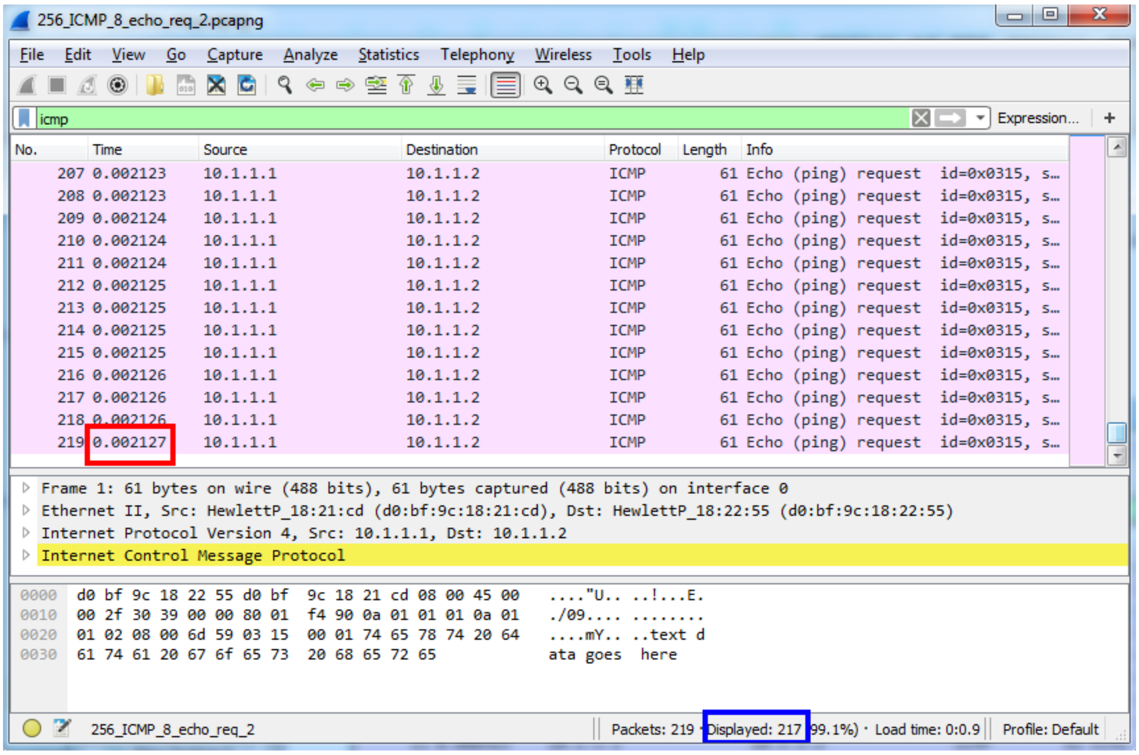
The results shown in Figure 4.26 are found by processing test case TC007-2. 256 ICMP echo requests were broadcast on the Ethernet LAN. The Wireshark report states that 217 packets were received of the 256 packets sent. This indicates that 39 packets were dropped. The results as shown in Eqn 4.20 indicate that Wireshark dropped 15.23% of the ICMP echo request packets when 256 ICMP echo request packets were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC007-2 provides an error rate of 15.23% from Eqn 4.20:

$$^{39}/_{256} \times ^{100}/_{1} = 15.23\%$$ (Eqn 4.20)



256_ICMP_8_echo_req_2.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression…
+

```
No.  Time      Source    Destination Protocol Length Info
 207 0.002123  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 208 0.002123  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 209 0.002124  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 210 0.002124  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 211 0.002124  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 212 0.002125  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 213 0.002125  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 214 0.002125  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 215 0.002125  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 216 0.002126  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 217 0.002126  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 218 0.002126  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
 219 0.002127  10.1.1.1  10.1.1.2    ICMP     61 Echo (ping) request  id=0x0315, s…
```

```
▷ Frame 1: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface 0
▷ Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: HewlettP_18:22:55 (d0:bf:9c:18:22:55)
▷ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2
▷ Internet Control Message Protocol
```

```
0000  d0 bf 9c 18 22 55 d0 bf  9c 18 21 cd 08 00 45 00   ...."U.. ..!...E.
0010  00 2f 30 39 00 00 80 01  f4 90 0a 01 01 01 0a 01   ./09.... ........
0020  01 02 08 00 6d 59 03 15  00 01 74 65 78 74 20 64   ....mY.. ..text d
0030  61 74 61 20 67 6f 65 73  20 68 65 72 65            ata goes  here
```

256_ICMP_8_echo_req_2 | Packets: 219 · Displayed: 217 (99.1%) · Load time: 0:0.9 | Profile: Default

Figure 4.26 :TC007-2

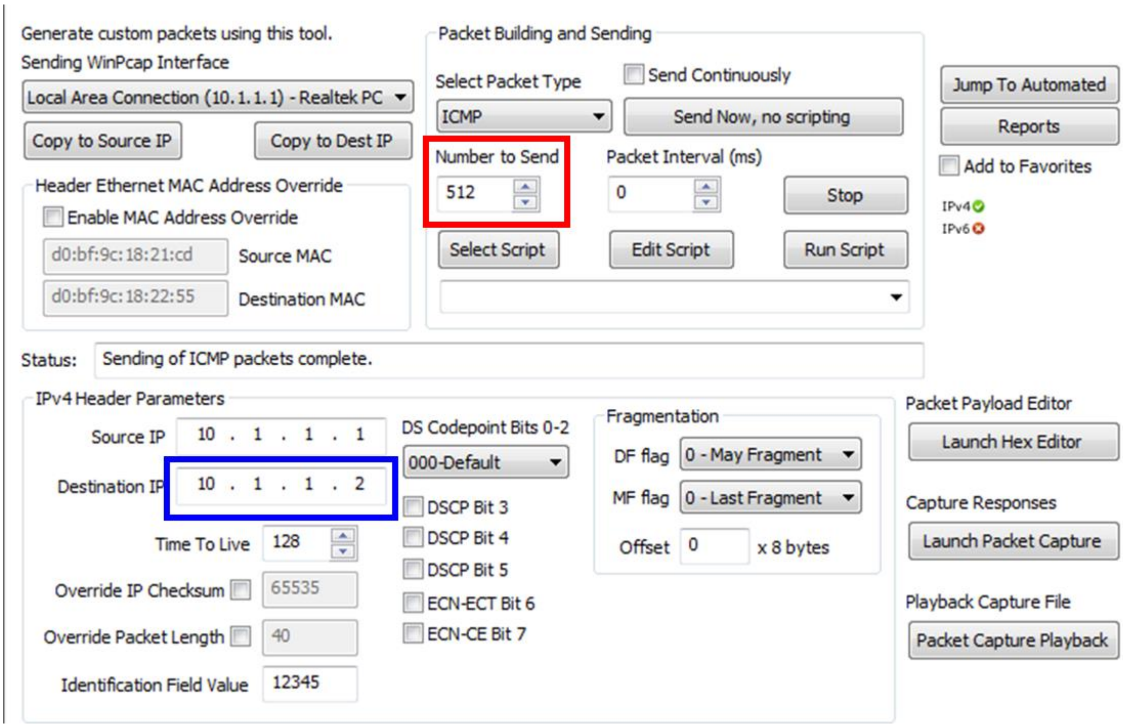
Table 4.10 shows the output from test cases TC007-1 and TC007-2, and displays the error rates calculated by Eqn 4.19 and Eqn 4.20.

Table 4.10: TC007 findings

Test Case	ICMP Packets Generated	ICMP Packets Received	ICMP Packets Dropped	Error Rate
007-1	256	214	42	16.41%
007-2	256	217	39	15.23%

4.4.8 Test Case 008

The second test case in this series was designed to set NetScan Tools Pro to generate 512 ICMP echo requests. Each request was sent directly to the Wireshark capture device with the IP address of 10.1.1.2 upon the LAN. Figure X below shows the configuration of the packet generator. The number of packets to be sent is shown in the red rectangle, the IP addresses of the target of the ICMP request is shown in the blue rectangle. The two iterations of Test Case 008 were carried out with the same packet generation parameters.



Generate custom packets using this tool.

Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override

□ Enable MAC Address Override

Source MAC: d0:bf:9c:18:21:cd

Destination MAC: d0:bf:9c:18:22:55

Packet Building and Sending

Select Packet Type: ICMP

□ Send Continuously

Number to Send: 512

Packet Interval (ms): 0

□ Add to Favorites

Status: Sending of ICMP packets complete.

IPv4 Header Parameters

Source IP: 10 . 1 . 1 . 1

Destination IP: 10 . 1 . 1 . 2

Time To Live: 128

Override IP Checksum □ 65535

Override Packet Length □ 40

Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default

□ DSCP Bit 3

□ DSCP Bit 4

□ DSCP Bit 5

□ ECN-ECT Bit 6

□ ECN-CE Bit 7

Fragmentation

DF flag: 0 - May Fragment

MF flag: 0 - Last Fragment

Offset: 0 x 8 bytes

Figure 4.27: TC008 baseline

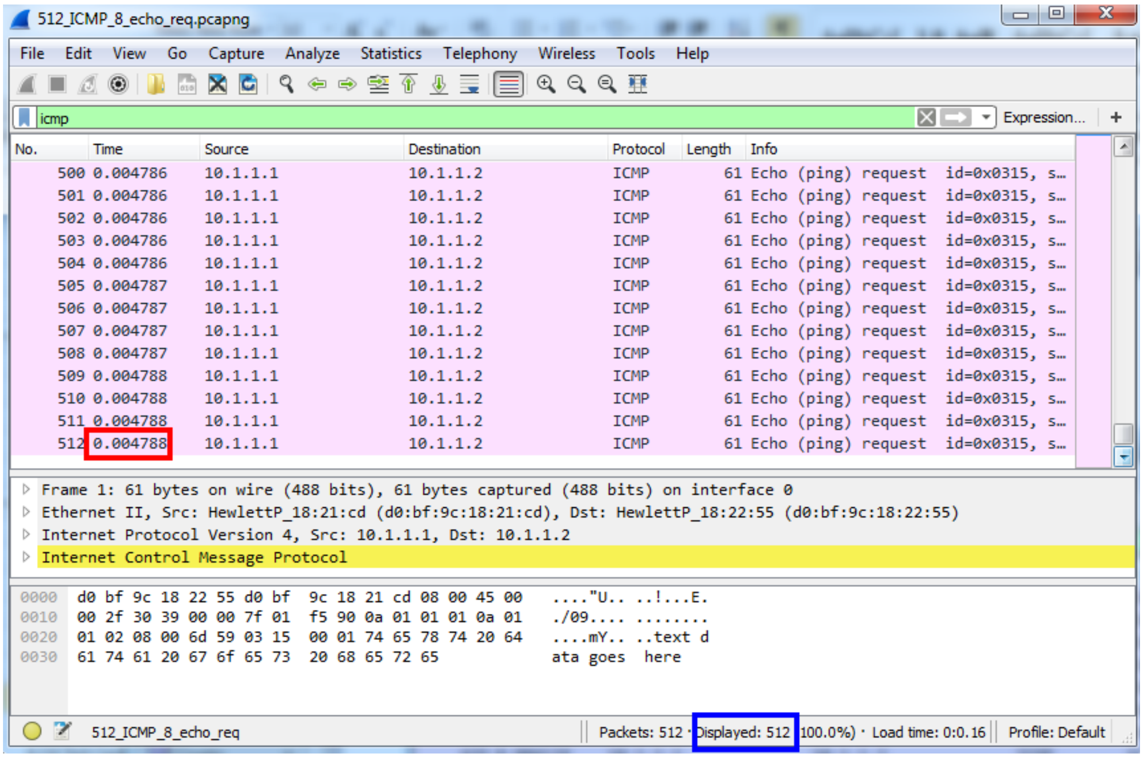
The results shown in Figure 4.28 were generated by processing test case TC008-1. 512 ICMP echo requests were broadcast on the Ethernet LAN. The Wireshark report states that 512 packets were received of the 512 packets sent. This indicates that 0 packets were dropped. The results as shown in Eqn 4.21 indicate that Wireshark dropped 0.0% of the ICMP echo request packets when 512 ICMP echo request packets were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC008-1 provides an error rate of 0% from Eqn 4.21:

$$^{0}/_{512} \times {}^{100}/_{1} = 0\% \qquad \text{(Eqn 4.21)}$$



512_ICMP_8_echo_req.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression…
+

```
No.  Time      Source    Destination  Protocol  Length  Info
 500 0.004786  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 501 0.004786  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 502 0.004786  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 503 0.004786  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 504 0.004786  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 505 0.004787  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 506 0.004787  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 507 0.004787  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 508 0.004787  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 509 0.004788  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 510 0.004788  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 511 0.004788  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
 512 0.004788  10.1.1.1  10.1.1.2     ICMP          61  Echo (ping) request  id=0x0315, s…
```

```
▷ Frame 1: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface 0
▷ Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: HewlettP_18:22:55 (d0:bf:9c:18:22:55)
▷ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2
▷ Internet Control Message Protocol
```

```
0000  d0 bf 9c 18 22 55 d0 bf  9c 18 21 cd 08 00 45 00   ...."U.. ..!...E.
0010  00 2f 30 39 00 00 7f 01  f5 90 0a 01 01 01 0a 01   ./09.... ........
0020  01 02 08 00 6d 59 03 15  00 01 74 65 78 74 20 64   ....mY.. ..text d
0030  61 74 61 20 67 6f 65 73  20 68 65 72 65            ata goes  here
```

```
512_ICMP_8_echo_req    Packets: 512 · Displayed: 512 (100.0%) · Load time: 0:0.16    Profile: Default
```

Figure4.28: TC008-1

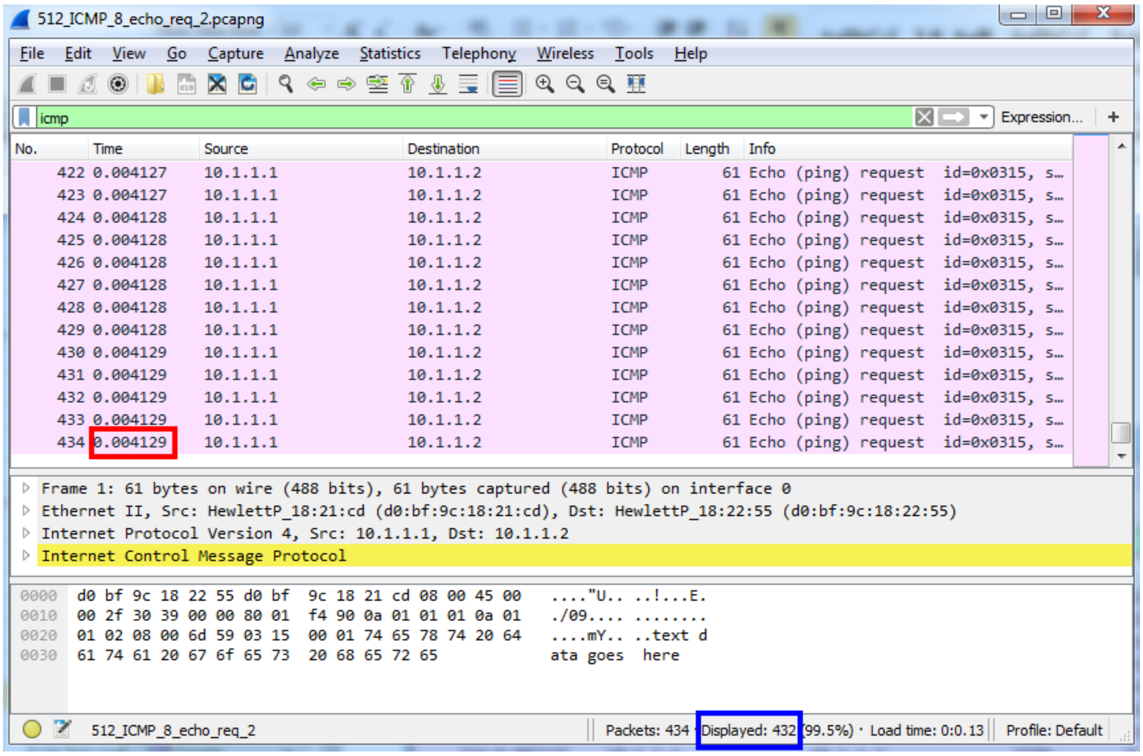
The results shown in Figure 4.29 were generated by processing test case TC008-2. 512 ICMP echo requests were broadcast on the Ethernet LAN. The Wireshark report states that 432 packets were received of the 512 packets sent. This indicates that 80 packets were dropped. The results as shown in Eqn 4.22 indicate that Wireshark dropped 15.63% of the ICMP echo request packets when 512 ICMP echo request packets were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC008-2 provides an error rate of 15.63% from Eqn 4.22:

$$^{80}/_{512} \times {}^{100}/_{1} = 15.63\% \qquad \text{(Eqn 4.22)}$$



512_ICMP_8_echo_req_2.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression...

```
No.     Time       Source     Destination  Protocol  Length Info
    422 0.004127   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    423 0.004127   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    424 0.004128   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    425 0.004128   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    426 0.004128   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    427 0.004128   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    428 0.004128   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    429 0.004128   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    430 0.004129   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    431 0.004129   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    432 0.004129   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    433 0.004129   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…
    434 0.004129   10.1.1.1   10.1.1.2     ICMP          61 Echo (ping) request  id=0x0315, s…

▷ Frame 1: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface 0
▷ Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: HewlettP_18:22:55 (d0:bf:9c:18:22:55)
▷ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2
▷ Internet Control Message Protocol

0000  d0 bf 9c 18 22 55 d0 bf  9c 18 21 cd 08 00 45 00   ...."U.. ..!...E.
0010  00 2f 30 39 00 00 80 01  f4 90 0a 01 01 01 0a 01   ./09.... ........
0020  01 02 08 00 6d 59 03 15  00 01 74 65 78 74 20 64   ....mY.. ..text d
0030  61 74 61 20 67 6f 65 73  20 68 65 72 65            ata goes  here

512_ICMP_8_echo_req_2        Packets: 434 · Displayed: 432 (99.5%) · Load time: 0:0.13    Profile: Default
```

Figure 4.29: TC008-2

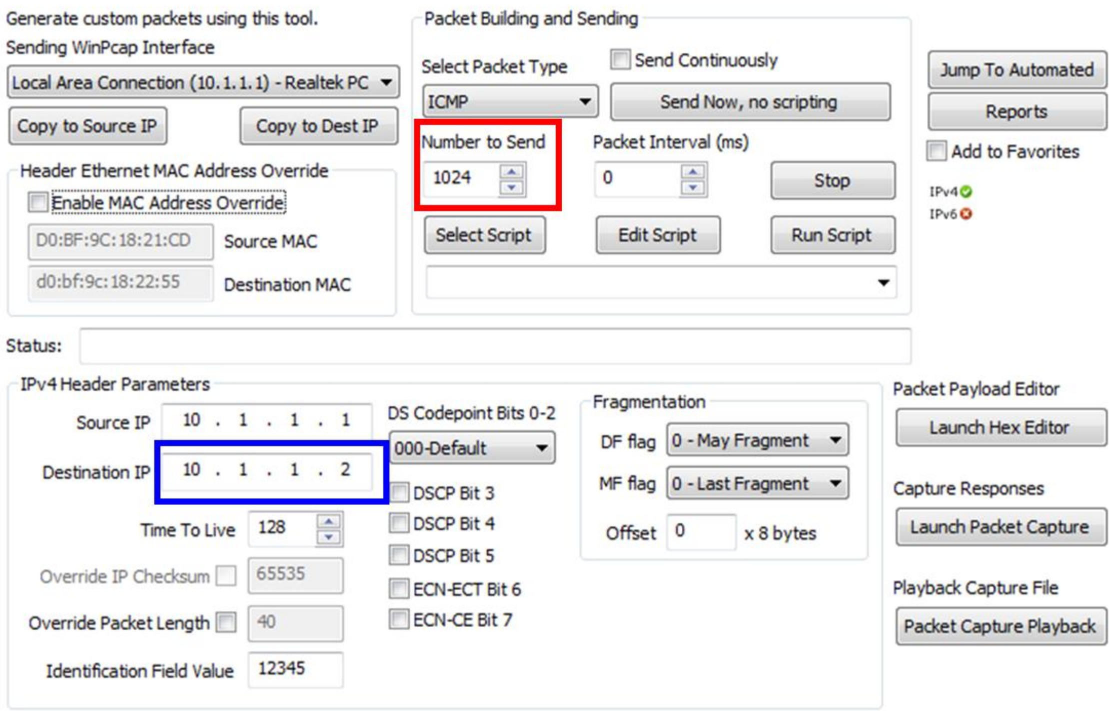
Table 4.11 shows the output from test cases TC008-1 and TC008-2, and displays the error rates calculated by Eqn 4.21 and Eqn 4.22.

Table 4.11: TC008 findings

Test Case	ICMP Packets Generated	ICMP Packets Received	ICMP Packets Dropped	Error Rate
008-1	512	512	0	0.00%
008-2	512	432	80	15.63%

4.4.9 Test Case 009

The final test case in this series was designed to set NetScan Tools Pro to generate 1,024 ICMP echo requests. Each request was sent directly to the Wireshark capture device with the IP address of 10.1.1.2 upon the LAN. Figure X below shows the configuration of the packet generator. The number of packets to be sent is shown in the red rectangle, the IP addresses of the target of the ICMP request is shown in the blue rectangle. The two iterations of Test Case 009 were performed with the same packet generation parameters.



Generate custom packets using this tool.

Sending WinPcap Interface: Local Area Connection (10.1.1.1) - Realtek PC

Header Ethernet MAC Address Override

□ Enable MAC Address Override

Source MAC: D0:BF:9C:18:21:CD

Destination MAC: d0:bf:9c:18:22:55

Packet Building and Sending

Select Packet Type: ICMP

□ Send Continuously

Number to Send: 1024

Packet Interval (ms): 0

□ Add to Favorites

Status:

IPv4 Header Parameters

Source IP: 10 . 1 . 1 . 1

Destination IP: 10 . 1 . 1 . 2

Time To Live: 128

Override IP Checksum □ 65535

Override Packet Length □ 40

Identification Field Value: 12345

DS Codepoint Bits 0-2: 000-Default

□ DSCP Bit 3

□ DSCP Bit 4

□ DSCP Bit 5

□ ECN-ECT Bit 6

□ ECN-CE Bit 7

Fragmentation

DF flag: 0 - May Fragment

MF flag: 0 - Last Fragment

Offset: 0 x 8 bytes

Figure 4.30: TC009 baseline

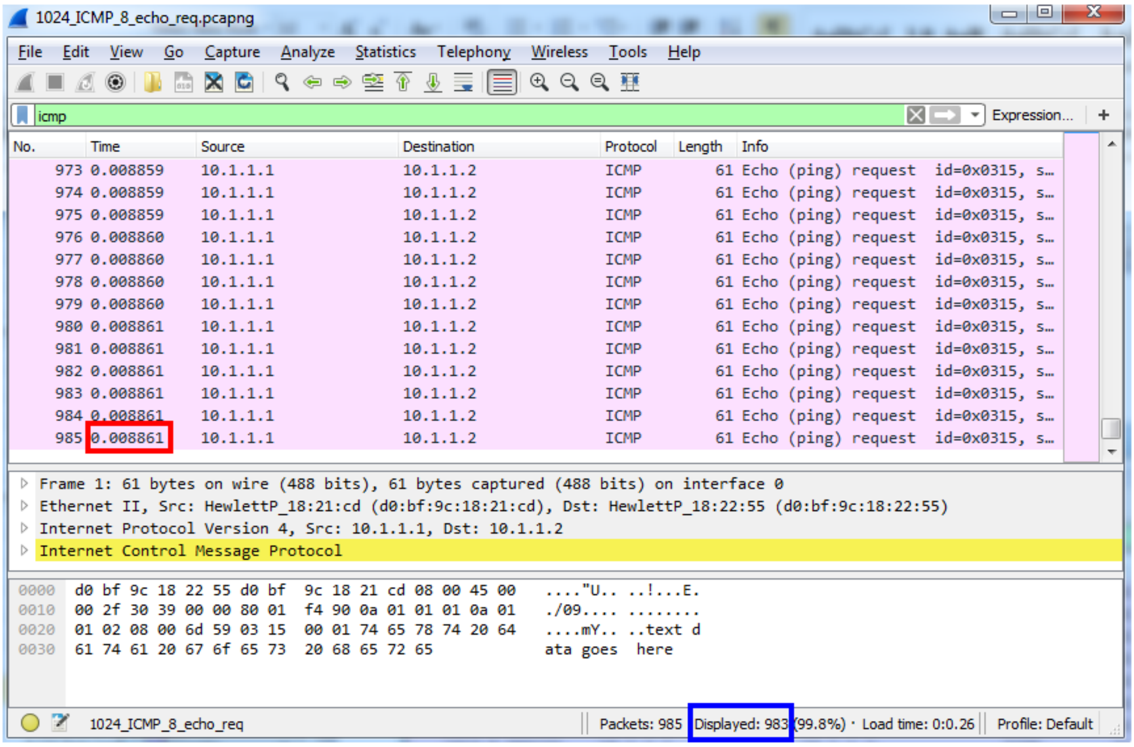
The results shown in Figure 4.31 were generated by processing test case TC009-1. 1,024 ICMP echo requests were broadcast on the Ethernet LAN. The Wireshark report states that 983 packets were received of the 1024 packets sent. This indicates that 41 packets were dropped. The results as shown in Eqn 4.23 indicate that Wireshark dropped 4.00% of the ICMP echo request packets when 1,024 ICMP echo request packets were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC009-1 provides an error rate of 4% from Eqn 4.23:

$$ {}^{41}\!/_{1,024} \times {}^{100}\!/_{1} = 4.00\% \qquad \text{(Eqn 4.23)} $$



1024_ICMP_8_echo_req.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression…
+

```
No.   Time       Source     Destination  Protocol  Length  Info
  973 0.008859   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  974 0.008859   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  975 0.008859   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  976 0.008860   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  977 0.008860   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  978 0.008860   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  979 0.008860   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  980 0.008861   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  981 0.008861   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  982 0.008861   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  983 0.008861   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  984 0.008861   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…
  985 0.008861   10.1.1.1   10.1.1.2     ICMP         61 Echo (ping) request  id=0x0315, s…

▷ Frame 1: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface 0
▷ Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: HewlettP_18:22:55 (d0:bf:9c:18:22:55)
▷ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2
▷ Internet Control Message Protocol

0000  d0 bf 9c 18 22 55 d0 bf  9c 18 21 cd 08 00 45 00   ...."U.. ..!...E.
0010  00 2f 30 39 00 00 80 01  f4 90 0a 01 01 01 0a 01   ./09.... ........
0020  01 02 08 00 6d 59 03 15  00 01 74 65 78 74 20 64   ....mY.. ..text d
0030  61 74 61 20 67 6f 65 73  20 68 65 72 65            ata goes  here

1024_ICMP_8_echo_req    Packets: 985 Displayed: 983 (99.8%) · Load time: 0:0.26    Profile: Default
```

Figure 4.31: TC009-1

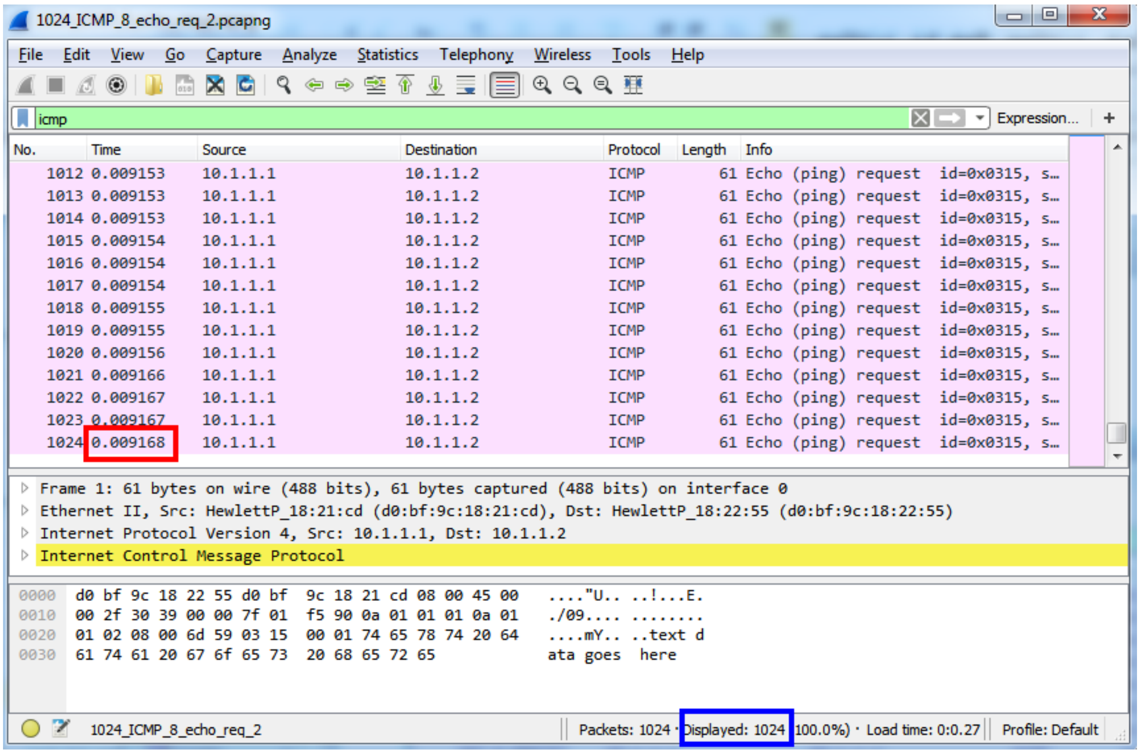
The results shown in Figure 4.32 were generated by processing test case TC009-2. 1,024 ICMP echo requests were broadcast on the Ethernet LAN. The Wireshark report states that 1,024 packets were received of the 1,024 packets sent. This indicates that 0 packets were dropped. The results as shown in Eqn 4.24 indicate that Wireshark dropped 0.0% of the ICMP echo request packets when 1024 ICMP echo request packets were sent onto the network.

Test assertion:

- The tool under test should successfully, completely and accurately process Ethernet frames.

TC009-2 provides an error rate of 0.0% from Eqn 4.24:

$$ {}^{0}/_{1{,}024} \times {}^{100}/_{1} = 0\% \qquad \text{(Eqn 4.24)} $$


1024_ICMP_8_echo_req_2.pcapng
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
icmp
Expression...
+

```
No.   Time      Source     Destination  Protocol  Length  Info
 1012 0.009153  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1013 0.009153  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1014 0.009153  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1015 0.009154  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1016 0.009154  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1017 0.009154  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1018 0.009155  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1019 0.009155  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1020 0.009156  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1021 0.009166  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1022 0.009167  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1023 0.009167  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
 1024 0.009168  10.1.1.1   10.1.1.2     ICMP        61 Echo (ping) request  id=0x0315, s…
```

```
▷ Frame 1: 61 bytes on wire (488 bits), 61 bytes captured (488 bits) on interface 0
▷ Ethernet II, Src: HewlettP_18:21:cd (d0:bf:9c:18:21:cd), Dst: HewlettP_18:22:55 (d0:bf:9c:18:22:55)
▷ Internet Protocol Version 4, Src: 10.1.1.1, Dst: 10.1.1.2
▷ Internet Control Message Protocol
```

```
0000  d0 bf 9c 18 22 55 d0 bf  9c 18 21 cd 08 00 45 00   ...."U.. ..!...E.
0010  00 2f 30 39 00 00 7f 01  f5 90 0a 01 01 01 0a 01   ./09.... ........
0020  01 02 08 00 6d 59 03 15  00 01 74 65 78 74 20 64   ....mY.. ..text d
0030  61 74 61 20 67 6f 65 73  20 68 65 72 65            ata goes  here
```

```
1024_ICMP_8_echo_req_2    Packets: 1024 · Displayed: 1024 (100.0%) · Load time: 0:0.27    Profile: Default
```

Figure 4.32: TC009-2

Table 4.12 shows the output from test cases TC009-1 and TC009-2, and displays the error rates calculated by Eqn 4.23 and Eqn 4.24.

Table 4.12: TC009 findings

Test Case	ICMP Packets Generated	ICMP Packets Received	ICMP Packets Dropped	Error Rate
009-1	1,024	983	41	4.00%
009-2	1,024	1024	0	0.00%

4.4.10 Analysis

The error rate generated was calculated as a percentage of number of packets dropped as compared to the number of packets generated for each of the packet generation levels tested. The complete data set from the 6 tests in the series are given in Table 4.13.

Table 4.13: Wireshark ICMP findings

Test Case	ICMP Packets Generated	ICMP Packets Received	ICMP Packets Dropped	Error Rate
007-1	256	214	42	16.41%
007-2	256	217	39	15.23%
008-1	512	512	0	0.00%
008-2	512	432	80	15.63%
009-1	1024	983	41	4.00%
009-2	1024	1024	0	0.00%

The results are displayed graphically in Figure 4.33, which shows the number of packets generated and the corresponding number of packets received. Figure 4.34 shows the error rate derived from each test case in the series.

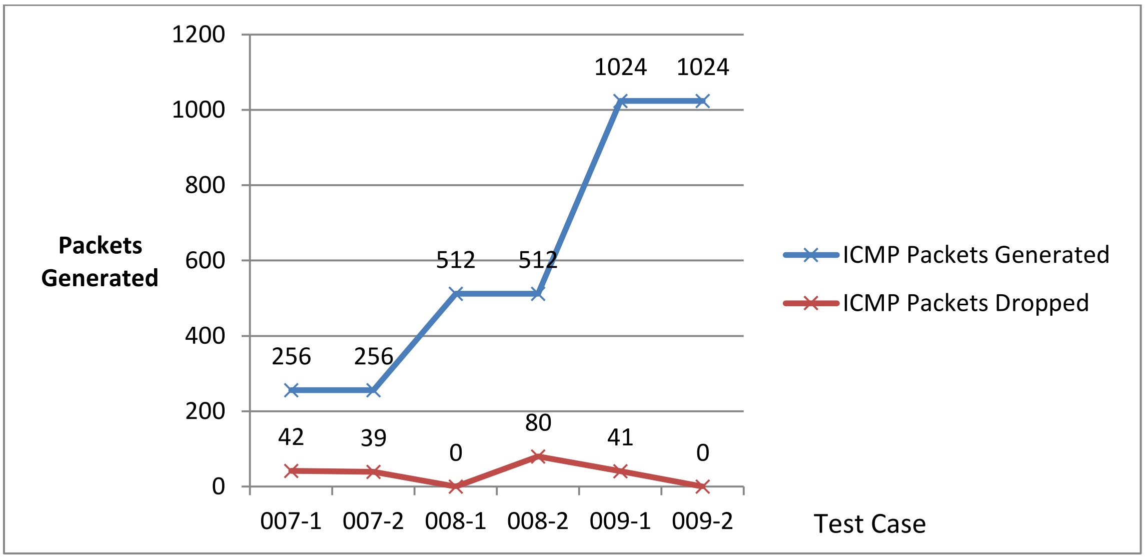


Figure 4.33: Packets dropped Wireshark ICMP

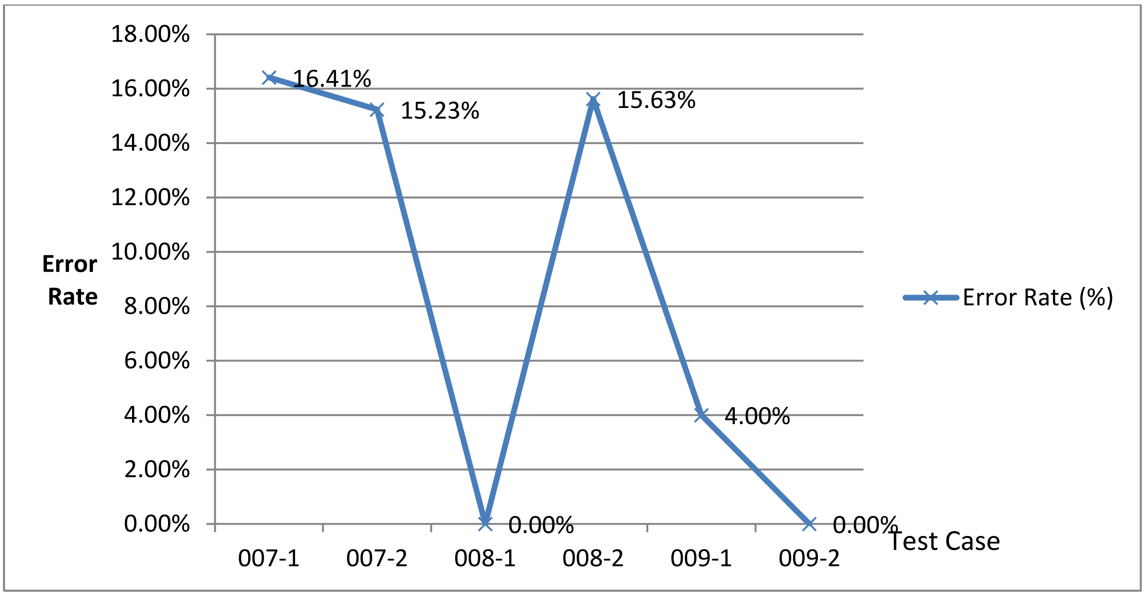


Figure 4.34: Error rate Wireshark ICMP

4.5 CONCLUSION

Chapter 4 has reported the results of the experimental testing and shown that the network tools used are vulnerable to failure. The error rates are considerable for each tool which raises issues regarding their usefulness in managing network traffic, providing information for security decision-making, and forensic investigation. In Chapter 5 a full investigation of these findings will be made. The research question, sub- questions and hypotheses are to be examined and the results reported based on the evidence presented in this chapter.

Chapter 5

DISCUSSION

5.0 INTRODUCTION

The object of Chapter 5 is to develop and discuss with analytical and critical thinking the theoretical arguments arising from the literature review contained in Chapter 2 and the empirical results gained in Chapter 4. In particular, it was the issues and problems identified in section 2.10, such as the lack of Digital forensics falsifiability testing and the presentation of deficient expert evidence, coupled with the issues of live data collection from a network environment, that formed the basis of the purpose of the study. The review of similar studies contained in Chapter 3, provided a methodological guide for data collection and process management (see Section 3.2.3). The research design that was applied was also set out in chapter 3, along with a list of data requirements. Chapter 4 has presented the results gained from applying the methodology and in Chapter 5 an in-depth discussion of these findings based on an evaluation of the various claims and issues that helped developed the specific set of research questions is made. Therefore, the purpose of chapter 5 is to move beyond the facts presented in chapter 4, and to engage in productive speculation. Thus, the discussion of the results contained in chapter 4 will be presented with analysis and interpretation from the literature review carried out in chapter 2 and the links to other research as discussed in chapter 3.

Section 5.1 and 5.2 discuss the findings generated by testing each Hypothesis. These 8 pages of Section 5.1 follow the same format. The Hypothsis is restated, and the result of the Hypothesis testing is given. Each test case is identified with a result given. The findings for each test case are identified, and then the findings are tested against the stated test assertion and the result is identified. Section 5.1.1 discusses the findings in relation to Hypothesis H1, *That a commonly used well-established and professionally acceptable Network Packet **Management** tool will perform under various levels of network stress without error.* Section 5.1.2 discusses the findings in relation to Hypothesis H2, *That a commonly used well-established and professionally acceptable Network Packet **Capture** tool will perform under various levels of network stress without error* Section 5.2 answers the sub questions: **SQ1**, *Does a network packet **Management** system and network packet **Capturing** tool provide legally acceptable forensic evidence?* and **SQ2***: Does the selected network packet **Management** system* *and packet **Capture** tool perform without error under load?* Section 5.3 answers the Research Question: **RQ:** *Does the Network **Management** System and the Network Packet **Capture** tool, achieve zero errors for digital forensic purposes?* Section 5.4 presents a discussion of the findings in relation to the literature reviewed in Chapter 2 and reviews digital forensic scientific development, forensic readiness criteria. Section 5.4.3 discusses the 2 results that are evidential exceptions generated from this research. Section 5.4.4 forms an opinion from the discussions. Section 5.5 describes my personal journey in review and Section 5.6 concludes Chapter 5.

5.1 HYPOTHESES TESTING

In section 3.2.4 the hypotheses were generated through the appraisal of the research sub-questions outlined in section 3.2.3. These research questions and sub-questions were derived from the problems identified in the review of relevant literature carried out in chapter 2 and these problems were identified in section 2.10. The hypotheses along with the result in an explanation will be presented in the following two sub-sections.

5.1.1 Hypothesis H1

Hypothesis H1: That a commonly used well-established and professionally acceptable Network Packet **Management** tool will perform under various levels of network stress without error (see Section 3.2.4).
H1 Result: = FALSE

TC001-1 Result = FALSE
Evidence: The IDS reported that 40,119 packets were received (see Figure 4.3, line 9, Red box). Of the 40,119 packets received, 33,527 packets were analysed (see Figure 4.3, Line 10). Therefore, the IDS reported that 6,591 packets were dropped from the 40,119 packets received (see Figure 4.3, Line 11, Blue box). However, when the number of packets analysed by the IDS is compared to 45,000 packets generated, this produced an error rate of 25.50% (see Eqn 4.2).

The test assertion for TC001-1 is the IDS will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC001-1 when related to the predictions of the test assertion and the test assertion requirements fails to accurately and completely process the test data resulting in a FALSE outcome.

TC001-2 Result = FALSE

Evidence: The IDS reported that 40,119 packets were received (see Figure 4.4, line 9, Red box). Of the 40,119 packets received, 33,527 packets were analysed (see Figure 4.4, Line 10). Therefore, the IDS reported that 6,591 packets were dropped from the 40,119 packets received (see Figure 4.4, Line 11, Blue box). However, when the number of packets analysed by the IDS is compared to 45,000 packets generated, this produced an error rate of 25.50% (see Eqn 4.4).

The test assertion for TC001-2 is the IDS will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC001-2 when related to the predictions of the test assertion and the test assertion requirements fails to accurately and completely process the test data resulting in a FALSE outcome.

TC002-1: Result = FALSE

Evidence: The IDS reported that 40,543 packets were received (see Figure 4.6, Line 9, Red box). Of the 40,543 packets received, 15,125 packets were analysed (see Figure 4.6, Line 10). Therefore, the IDS reported that 25,416 packets were dropped from the 40,543 packets received (see Figure 4.6, Line 11, Blue box). However, when the number of packets analysed by the IDS is compared to 45,000 packets generated, this produced an error rate of 66.39% (see Eqn 4.6).

The test assertion for TC002-1 is the IDS will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC002-1 when related to the predictions of the test assertion and the test assertion requirements fails to accurately and completely process the test data resulting in a FALSE outcome.

TC002-2: Result = FALSE

Evidence: The IDS reported that 40,177 packets were received (see Figure 4.7, Line 9, Red box). Of the 40,177 packets received, 11,764 packets were analysed (see Figure 4.7, Line 10). Therefore, the IDS reported that 28,412 packets were dropped from the 40,177 packets received (see Figure 4.7, Line 11, Blue box). However, when the number of packets analysed by the IDS is compared to 45,000 packets generated, this produced an error rate of 73.86% (see Eqn 4.8).

The test assertion for TC002-2 is the IDS will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC002-2 when related to the predictions of the test assertion and the test assertion requirements fails to accurately and completely process the test data resulting in a FALSE outcome.

TC 003-1: Result = FALSE

Evidence: The IDS reported that 39,967 packets were received (see Figure 4.9, Line 9, Red box). Of the 39,967 packets received, 9,353 packets were analysed (see Figure 4.9, Line 10). Therefore, the IDS reported that 30,612 packets were dropped from the 39,967 packets received (see Figure 4.9, Line 11, Blue box). However, when the number of packets analysed by the IDS is compared to 45,000 packets generated, this produced an error rate of 79.22% (see Eqn 4.10).

The test assertion for TC003-1 is the IDS will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC003-1 when related to the predictions of the test assertion and the test assertion requirements fails to accurately and completely process the test data resulting in a FALSE outcome.

TC 003-2: Result = FALSE

Evidence: The IDS reported that 40,176 packets were received (see Figure 4.10, Line 9, Red box). Of the 40,176 packets received, 9,397 packets were analysed (see Figure 4.10, Line 10). Therefore, the IDS reported that 30,778 packets were dropped from the 40,176 packets received (see Figure 4.10, Line 11, Blue box). However, when the number of packets analysed by the IDS is compared to 45,000 packets generated, this produced an error rate of 79.12% (see Eqn 4.12).

The test assertion for TC003-2 is the IDS will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC003-2 when related to the predictions of the test assertion and the test assertion requirements fails to accurately and completely process the test data resulting in a FALSE outcome.

Each Test Case in this series produced errors, caused by incomplete data collection, where packets were dropped (See Table 4.5, Rows TC001-1, TC001-2, TC002-1, TC002-2, TC003-1 and TC003-2). The graphical results show an increasing error rate that is not zero, at any transmission rate (see Figure 4.11 and 4.12). Thus the Test Case requirements for the IDS to consistently process Ethernet frames at different data rates with zero errors have not been met. Therefore, the Hypothesis **H1** has been proven to be **FALSE**.

5.1.2 Hypothesis H2

Hypothesis H2: That a commonly used well-established and professionally acceptable Network Packet **Capture** tool will perform under various levels of network stress without error (see Section 3.2.4).

H2 Result: = FALSE

TC004-1 Result = FALSE

Evidence: 1024 ARP request packets were generated by NetScan Tools Pro (see Figure 4.13, Red Box). Each ARP packet requested information regarding IP address 10.1.15 (see Figure 4.13, Blue Box). Wireshark reported that 510 ARP packets were received (see Figure 4.14, Blue Box). Therefore, 514 ARP packets were dropped from the 1024 packets generated (see Table 4.6, Line 1, Cell 4). When the number of packets captured by Wireshark is compared to the 1024 ARP packets generated, this produced an error rate of 50.20% (see Eqn 4.13).

The test assertion for TC004-1 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC004-1, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome.

TC004-2 Result = FALSE

Evidence: 1024 ARP request packets were generated by NetScan Tools Pro (see Figure 4.13, Red Box). Each ARP packet requested information regarding IP address 10.1.15

(see Figure 4.13, Blue Box). Wireshark reported that 511 ARP packets were received (see Figure 4.15, Blue Box). Therefore, 513 ARP packets were dropped from the 1024 packets generated (see Table 4.6, Line 2, Cell 4). When the number of ARP packets captured by Wireshark is compared to the 1024 ARP packets generated, this produced an error rate of 50.10% (see Eqn 4.14).

The test assertion for TC004-2 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC004-2, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome.

TC004 Result = FALSE

Evidence: The test assertion for the complete test case, TC004, is the packet capture tool under test will perform 100% successfully independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion, that the tool under test should successfully completely and accurately process Ethernet frames, has not been met (see Section 4.2.4). TC004-1 and TC004-2 were both tests of the same level of data, and both iterations were shown to have errors (see Table 4.6, Rows TC004-1 and TC004-2). Therefore, upon analysis of the results of each iteration of test TC004 when related to the predictions of the test assertion and the test assertion requirement, TC004 fails to accurately and completely process the test data resulting in a FALSE outcome.

TC005-1 Result = FALSE

Evidence: 2048 ARP request packets were generated by NetScan Tools Pro (see Figure 4.16, Red Box). Each ARP packet requested information regarding IP address 10.1.15 (see Figure 4.16, Blue Box). Wireshark reported that 652 ARP packets were captured (see Figure 4.17, Blue Box). Therefore, 1,396 ARP packets were dropped from the 2048 packets generated (see Table 4.7, Line 1, Cell 4). When 652 ARP packets captured by Wireshark is compared to the 2048 ARP packets generated, this produced an error rate of 68.16% (see Eqn 4.15).

The test assertion for TC005-1 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC005-1, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.7, Row TC005-1).

TC005-2 Result = FALSE

Evidence: 2048 ARP request packets were generated by NetScan Tools Pro (see Figure 4.16, Red Box). Each ARP packet requested information regarding IP address 10.1.15 (see Figure 4.16, Blue Box). Wireshark reported that 601 ARP packets were received (see Figure 4.18, Blue Box). Therefore, 1,447 ARP packets were dropped from the 2048 packets generated (see Table 4.7, Line 2, Cell 4). When the number of ARP packets captured by Wireshark is compared to the 2048 ARP packets generated, this produced an error rate of 70.65% (see Eqn 4.16).

The test assertion for TC005-2 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC005-2, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome.

TC005: Result = FALSE

Evidence: The test assertion for the complete test case, TC 005, is the packet capture tool under test will perform 100% successfully independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion, that the tool under test should successfully completely and accurately process Ethernet frames, has not been met (see Section 4.2.4). TC005-1 and TC 005-2 were both tests of the same level of data, and both iterations were shown to have errors (see Table 4.7, Rows TC005-1 and TC005-2). Therefore, upon analysis of the results of each iteration of test TC005 when related to the predictions of the test assertion and the test assertion requirement, TC005 fails to accurately and completely process the test data resulting in a FALSE outcome.

TC006-1 Result = FALSE

Evidence: 4096 ARP request packets were generated by NetScan Tools Pro (see Figure 4.19, Red Box). Each ARP packet requested information regarding IP address 10.1.15 (see Figure 4.19, Blue Box). Wireshark reported that 966 ARP packets were captured (see Figure 4.20, Blue Box). Therefore, 3,130 ARP packets were dropped from the 4096 packets generated (see Table 4.8, Line 1, Cell 4). When the 966 ARP packets captured by Wireshark is compared to the 4096 ARP packets generated, this produced an error rate of 76.42% (see Eqn 4.17).

The test assertion for TC006-1 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC006-1, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.8, Row TC006-1).

TC006-2 Result = FALSE

Evidence: 4096 ARP request packets were generated by NetScan Tools Pro (see Figure 4.19, Red Box). Each ARP packet requested information regarding IP address 10.1.15 (see Figure 4.19, Blue Box). Wireshark reported that 982 ARP packets were captured (see Figure 4.21, Blue Box). Therefore, 3,114 ARP packets were dropped from the 4096 packets generated (see Table 4.8, Line 2, Cell 4). When 982 ARP packets captured by Wireshark is compared to the 4096 ARP packets generated, this produced an error rate of 76.03% (see Eqn 4.18).

The test assertion for TC006-2 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC006-2, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.8, Row TC006-2).

TC006: Result = FALSE

The test assertion for the complete test case, TC006, is the packet capture tool under test will perform 100% successfully independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion, that the packet capture tool should successfully completely and accurately process Ethernet frames, has not been met (see Section 4.2.4). TC006-1 and TC006-2 were both tests of the same level of data, and both iterations were shown to have errors (see Table 4.8, Rows TC006-1 and TC006-2). Therefore, upon analysis of the results of each iteration of test TC006 when related to the predictions of the test assertion and the test assertion requirement, TC006 fails to accurately and completely process the test data resulting in a FALSE outcome.

TC007-1 Result = FALSE

Evidence: 256 ICMP echo packets were generated by NetScan Tools Pro (see Figure 4.24, Red Box). Each ICMP echo packet was directed to IP address 10.1.1.2 (see Figure 4.24, Blue Box). Wireshark reported that 214 ICMP echo packets were captured (see Figure 4.25, Blue Box). Therefore, 42 ICMP echo packets were dropped from the 256 packets generated (see Table 4.10, Line 1, Cell 4). When the 214 ICMP echo packets captured by Wireshark is compared to the 256 ARP packets generated, this produced an error rate of 16.41% (see Eqn 4.19).

The test assertion for TC007-1 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC007-1, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.10, Row TC07-1).

TC007-2 Result = FALSE

Evidence: 256 ICMP echo packets were generated by NetScan Tools Pro (see Figure 4.24, Red Box). Each ICMP echo packet was directed to IP address 10.1.1.2 (see Figure 4.24, Blue Box). Wireshark reported that 217 ICMP echo packets were captured (see Figure 4.26, Blue Box). Therefore, 39 ICMP echo packets were dropped from the 256 packets generated (see Table 4.10, Line 2, Cell 4). When the 217 ICMP echo packets captured by Wireshark is compared to the 256 ARP packets generated, this produced an error rate of 15.23% (see Eqn 4.20).

The test assertion for TC007-2 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC007-2, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.10, Row TC007-2).

TC007: Result = FALSE

The test assertion for the complete test case, TC007, is the packet capture tool under test will perform 100% successfully independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion, that the packet capture tool should successfully completely and accurately process Ethernet frames, has not been met (see Section 4.2.4). TC007-1 and TC007-2 were both tests of the same level of data, and both iterations were shown to have errors (see Table 4.10, Rows TC007-1 and TC007-2). Therefore, upon analysis of the results of each iteration of test TC007 when related to the predictions of the test assertion and the test assertion requirement, TC007 fails to accurately and completely process the test data resulting in a FALSE outcome.

TC008-1 Result = NOT FALSE

Evidence: 512 ICMP echo packets were generated by NetScan Tools Pro (see Figure 4.27, Red Box). Each ICMP echo packet was directed to IP address 10.1.1.2 (see Figure 4.27, Blue Box). Wireshark reported that 512 ICMP echo packets were captured (see Figure 4.28, Blue Box). Therefore, NO ICMP echo packets were dropped from the 512 packets generated (see Table 4.11, Line 1, Cell 4). When the 256 ICMP echo packets captured by Wireshark is compared to the 256 ICMP echo packets generated, this produced a ZERO error rate (see Eqn 4.21).

The test assertion for TC008-1 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC008-1, when related to the predictions of the test assertion and the test assertion requirements, accurately and completely processed the test data resulting in a NOT FALSE outcome (see Table 4.11, Row TC08-1).

TC008-2 Result = FALSE

Evidence: 512 ICMP echo packets were generated by NetScan Tools Pro (see Figure 4.27, Red Box). Each ICMP echo packet was directed to IP address 10.1.1.2 (see Figure 4.27, Blue Box). Wireshark reported that 432 ICMP echo packets were captured (see Figure 4.29, Blue Box). Therefore, 80 ICMP echo packets were dropped from the 512 packets generated (see Table 4.11, Line 2, Cell 4). When the 432 ICMP echo packets captured by Wireshark is compared to the 512 ARP packets generated, this produced an error rate of 15.23% (see Eqn 4.22).

The test assertion for TC008-2 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC008-2, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.11, Row TC008-2).

TC008: Result = FALSE

The test assertion for the complete test case, TC008, is the packet capture tool under test will perform 100% successfully independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion, that the packet capture tool should successfully completely and accurately process Ethernet frames, has not been met (see Section 4.2.4). TC008-1 and TC008-2 were both tests of the same level of data. TC008-1 was shown to capture the complete dataset without errors (see Table 4.11, Row TC008-1). However, TC008-2 was shown to contain errors (see Table 4.11, Row TC008-2) Therefore, upon analysis of the results of each iteration of test TC008 when related to the predictions of the test assertion and the test assertion requirement, TC008 fails to accurately and completely process the test data resulting in a FALSE outcome.

TC009-1 Result = FALSE

Evidence: 1024 ICMP echo packets were generated by NetScan Tools Pro (see Figure 4.30, Red Box). Each ICMP echo packet was directed to IP address 10.1.1.2 (see Figure 4.30, Blue Box). Wireshark reported that 983 ICMP echo packets were captured (see Figure 4.31, Blue Box). Therefore, 41 ICMP echo packets were dropped from the 1024 packets generated (see Table 4.12, Line 1, Cell 4). When the 983 ICMP echo packets captured by Wireshark is compared to the 1024 ICMP echo packets generated, this produced an error rate of 4.00% (see Eqn 4.23).

The test assertion for TC009-1 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion are complete processing with zero error rates consistently, has not been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC009-1, when related to the predictions of the test assertion and the test assertion requirements, fails to accurately and completely process the test data resulting in a FALSE outcome (see Table 4.12, Row TC009-1).

TC009-2 Result = NOT FALSE

Evidence: 1024 ICMP echo packets were generated by NetScan Tools Pro (see Figure 4.30, Red Box). Each ICMP echo packet was directed to IP address 10.1.1.2 (see Figure 4.30, Blue Box). Wireshark reported that 1024 ICMP echo packets were captured (see Figure 4.32, Blue Box). Therefore, NO ICMP echo packets were dropped from the 1024 packets generated (see Table 4.12, Line 2, Cell 4). When the 1024 ICMP echo packets captured by Wireshark is compared to the 1024 ICMP echo packets generated, this produced a ZERO error rate (see Eqn 4.24).

The test assertion for TC009-2 is Wireshark will successfully capture 100% of the packets generated, independent of the levels of data throughput (see Section 4.2.3).

The requirements for testing this assertion are complete processing with zero error rates consistently, has been met (see Section 4.2.4). Therefore, upon analysis of the results of test TC009-2, when related to the predictions of the test assertion and the test assertion requirements, accurately and completely processed the test data resulting in a NOT FALSE outcome (see Table 4.12, Row TC009-2).

TC009: Result = FALSE

The test assertion for the complete test case, TC009, is the packet capture tool under test will perform 100% successfully independent of the levels of data throughput (see Section 4.2.3). The requirements for testing this assertion, that the packet capture tool should successfully completely and accurately process Ethernet frames, has not been met (see Section 4.2.4). TC009-1 and TC009-2 were both tests of the same level of data. TC009-1 was shown to contain errors (see Table 4.12, Row TC009-1) However, TC009-2 was shown to capture a complete dataset (see Table 4.12, Row TC009-2). Therefore, upon analysis of the results of each iteration of test TC009 when related to the predictions of the test assertion and the test assertion requirement, TC009 fails to accurately and completely process the test data resulting in a FALSE outcome.

5.2 SUB-QUESTIONS

Whilst the objective of the research question is to define the scope of this research, The purpose of the sub-questions are to ensure the main question is answered thoroughly
2 sub-questions have been determined to be appropriate to satisfactorily answer the research question. These are:

Subquestion SQ1: *Does a network packet **Management** system and network packet **Capturing** tool provide legally acceptable forensic evidence?*

SQ1 Answer: NO

Evidence: The creation of an evaluation framework is required as part of producing an effective forensic report where the manner in which the evidence was acquired forms the basis of the evaluation framework (see Section 2.4 and Section 3.3). Importantly, there is a requirement to assign an error rate to a scientific procedural tool to comply with the Daubert test (see Section 2.2.4). This is also an important part of scientific development as postulated by Popper (see Section 2.1.1). The requirements of falsifiability have been incorporated into the judicial standards as demonstrated by the Daubert test, which is designed to eliminate pseudoscience from entering the courtroom disguised as expert evidence (see Section 2.1.1). Therefore, it is the establishment of error rates that confirm the evidence gathered is scientifically valid.

When packet Management Systems and network packet capturing tool is a subject to these guidelines, frameworks and principles, any error rate produced demonstrates the capture of a data set that is incomplete. It is expected that any expert evidence will report must include the information the report is based upon and the presentation of an incomplete dataset may render the evidence in admissible (see Section 2.2.2 and Section 2.2.4). Therefore, when combining the requirements for the evidence to be scientifically valid and the requirements for completeness show that there is an expectation of zero errors, when expert evidence is presented in a court.

Thus, in all test cases performed the expectation is that there be zero errors (see Section 4.2. and, Table 4.1).

Upon evaluation of the data provided by the 18 iterations of the 9 test cases, 2 test case iterations were found to produce zero errors (see Table 4.13, row TC008-1 and row TC009-2). These results however, were found to be inconsistent. The experiment failed to generate a zero error rate with the next test case iteration using the same parameters (see Table 4.13 row TC008-2 and row TC 009-1). As the results were not replicable over both iterations, and it is an evidential requirement that results deriving from a scientific process must be replicable by a third party performing the same function, these two results are deemed inconclusive (see Section 2.2.3). Therefore, the six test cases, ranging from TC 004 TC 009 all consistently produced non-zero errors and the answer to Subquestion 1 is NO (see Table 4.9 and Table 4.13).

Subquestion SQ2*: Does the selected network packet* ***Management*** *system and packet* ***Capture*** *tool perform without error under load?*
SQ2 Answer: NO

Evidence: The creation of an evaluation framework is required as part of producing an effective forensic report where the manner in which the evidence was acquired forms the basis of the evaluation framework (see Section 2.4 and Section 3.3). Importantly, there is a requirement to assign an error rate to a scientific procedural tool to comply with the Daubert test (see Section 2.2.4). This is also an important part of scientific development as postulated by Popper (see Section 2.1.1). The requirements of falsifiability have been incorporated into the judicial standards as demonstrated by the Daubert test, which is designed to eliminate pseudoscience from entering the courtroom disguised as expert evidence (see Section 2.1.1). Therefore, it is the establishment of error rates that confirm the evidence gathered is scientifically valid.

When packet Management Systems and network packet capturing tool is a subject to these guidelines, frameworks and principles, any error rate produced demonstrates the capture of a data set that is incomplete. It is expected that any expert evidence will report must include the information the report is based upon and the presentation of an incomplete dataset may render the evidence in admissible (see Section 2.2.2). Therefore, when combining the requirements for the evidence to be scientifically valid and the requirements for completeness show that there is an expectation of zero errors, when expert evidence is presented in a court. Thus, in all test cases performed the expectation is that there be zero errors (see Section 4.2.4, and Table 4.1).

Upon evaluation of the data provided by the 18 iterations of the 9 test cases, 2 test case iterations were found to produce zero errors (see Table 4.13, row TC008-1 and row TC009-2). These results however, were found to be inconsistent. The experiment failed to generate a zero error rate with the next test case iteration using the same parameters (see Table 4.13 row TC008-2 and row TC 009-1). The six test cases, ranging from TC 4 to TC 009 all consistently produced non-zero errors (see Table 4.5 and Table 4.13). As the results were not replicable over both iterations, and it is an evidential requirement that results deriving from a scientific process must be replicable by a third party performing the same function, these two results are deemed inconclusive (see Section 2.2.3).

5.3 RESEARCH QUESTION

In section 2.9, there is a summary of problems to be addressed in order to be forensically prepared when a business network is considered. These problems were derived through an evaluation of a comprehensive literature review, which was carried out in chapter 2. The literature review highlighted the following key issues facing an implementation of network forensic readiness:

- Few digital forensic tools have been subjected to Daubert standards (see Section 2.9.1 and Section 2.2.4).
- No comprehensive methodological approach (see Section 2.9.2 and Section 2.1.3).
- Incomplete dataset collection (see Section 2.9.3 and Section 2.3.2).
- Cost and Impact (see Section 2.9.4 and Section 2.3.1).

- Volume growth and technological change (see Section 2.9.5 and Section 2.5.2).
- Live data collection (see Section 2.9.6 and Section 2.6.2).
- Event reconstruction (see Section 2.9.7 and Section 2.8.1).

Section 3.2.3 presents the research question which addresses the critical point of testing packet Management and packets nothing tools falsifiability and error rates. The research question itself is comprehensive and designed to generate a problem context that may occur, and is therefore broken down into several sub questions as identified in section 5.2.

The research question being answered by this methodological research is:

Research Question RQ: *Does the Network **Management** System and the Network Packet **Capture** tool, achieve zero errors for digital forensic purposes?*

RQ Result: NO

Evidence: the results for test case, TC001 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC001 (see Table 4.2 Rows TC001-1 and TC001-2). The results for test case, TC 002 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC002 (see Table 4.3 Rows TC002-1 and TC002-2). The results for test case, TC 003 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC003 (see Table 4.4 Rows TC003-1 and TC003-2).Thus, the Packet Management Software under test did not achieve a zero rate for digital forensic purposes. The results for test case, TC004 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC004 (see Table 4.6 Rows TC004-1 and TC004-2). The results for test case, TC 005 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC005 (see Table 4.7 Rows TC005-1 and TC005-2). The results for test case, TC006 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC006 (see Table 4.8 Rows TC006-1 and TC006-2). The results for test case, TC007 show that both iterations of this test produced a nonzero error rate and therefore a FAIL result for test case TC007 (see Table 4.10 Rows TC007-1 and TC007-2).

The results for test case, TC 008 show that one iteration of this test produced a nonzero error rate and therefore a FAIL result for test case TC008-2 (see Table 4.11

Rows TC008-2) Iteration TC008-1 did, however, record a zero error rate(see Table 4.11 Row TC008-1). As the non-zero result was not repeated in the next iteration of the test case scenario, the result overall for TC008 was determined to be FAIL. The results for test case, TC 009 show that one iteration of this test produced a nonzero error rate and therefore a FAIL result for test case TC009-1 (see Table 4.12 Rows TC009-1) Iteration TC009-2 did, however, record a zero error rate (see Table 4.12 Row TC009-2). As the non-zero result was not repeated in the previous iteration of the test case scenario, the result overall for TC009 was determined to be FAIL.

Therefore the answer to the Research Question RQ1 : *Does the Network Management System and the Network Packet Capture tool, achieve zero errors for digital forensic purposes?* Is negative, and therefore is determined to be NO

5.4 DISCUSSION

Digital forensics is a science, and not metaphysics, as the science can be subjected to tests of falsifiability, thus satisfying Popper’s postulations (see Section 2.1.1). Digital forensics has also been subjected to paradigm shifts in technology that can be considered scientific revolutions. An example of this is computers shifting from individual separate instances to networked systems encompassing many thousands of computers. Thus the field of digital forensics cannot advance by simply adding to what is already know. An example of this is the difficulty in adapting static network forensic techniques such as hard drive acquisition, to the more difficult techniques involved in capturing live data. This requires an evolutionary change, as postulated by Kuhn (see Section 2.1.2). There is a problem with the evolutionary concept in terms of networked digital forensic tools. The survivors, in terms of digital forensic tools, are seen to be fit by the nature of survival alone. For example, in the vendor driven marketplace the tools existence can be as simple as the fact that the vendor still exists after 10 years, rather than the vendor providing the best tool for the job. Therefore, the digital forensic tools that have survived, require testing for error rates. Thus fitness in terms of digital forensic science and in keeping with the testing that has been carried out within this research thesis.

A study of the digital forensic investigation model development from 2001 to 2011 showed the development of the static digital forensic investigation from a methodical investigation of a physical crime scene through acquisition and preservation before investigation through to triage in chronological considerations to finally, delivering a generic investigation model (see Section 2.1.3). In the light of the philosophical scientific developmental processes, the live-networked digital forensic investigation will develop from chaos towards order. Therefore, it is the basis, and a firm foundation for the precepts of the methodological component of this thesis. Thus, the deliverable is to test with falsifiability an evolved tool with respect to digital forensic investigation development. The deliverable directly relates to the overarching research assertion that the tool under test will produce zero errors. Both of the tools tested within this research fail this test and the outcome is determined to be false. The purpose of the research as demonstrated in the findings along with the subsequent analysis is not to proclaim that networked digital forensic investigation has been determined to be false, but that the tool itself generated errors upon use in a specific scenario under certain conditions. Benefits to the digital forensic community and the legal profession will be through access to published results of multiple tests of many tools.

Networked digital forensics is facing a requirement to be aligned with the more traditional forensic sciences and face increasing levels of scientific rigour. As digital forensics is at the intersection between law and computer technology, the technological issues only form one part of the digital forensic investigators cohesive whole. Digital evidence is required to satisfy exacting requirements when presented to court, and the desired outcome of forensic readiness is for an organisation to be ready for legal proceedings as well as providing technical networking information (see Section 2.2.1). Admissibility standards ensure evidential relevance and credibility and any forensic readiness policy must ensure the information gathered complies with standards of completeness and non-repudiation. The standards have been established in legal decisions and determinations such as the Delbert test, which can be applied to any scientific procedure as a basis for admissibility of scientific evidence in court. A very important standard within these tests is that there be a known error rate or the potential to know the error rate associated with the use of the scientific procedure (see Section 2.2.4). Thus, the outcome of the research is the ability to produce a known error rate.

The error rate determined the answer to the research question: *Does the selected network packet **Management** system and packet **Capture** tool perform without error under load?* The answer to which is determined to be negative in 16 out of 18 test case iterations, where error rates were generated that exceeded zero.

It is important to note in this discussion that a forensic tool which generates a non-zero error rate, will still be considered as valid scientific expert evidence, if the error rate is able to be determined and is delivered, along with the scientific evidence.

As there is little if any publication of known error rates of network digital forensic tools, the research investigation was focused on whether any errors occurred, rather than a determination of error rates generated.

5.4.1 Defining Forensic Science Development

Section 2.1.identified Digital forensics as a science developeding in a rather ad hoc, chaotic manner which has not conformed to well-known models of scientific development. The reading of Popper's book '*The logic of scientific discovery'* show the distinguishing of a set of methodological rules which he termed falsificationism. Methodological scientific development as postulated by Popper consists of an iterative process of model building that includes prediction, observation and analysis (see Section 2.1.1). Popper's criterion provide the demarcation between science and metaphysics, specifying that scientific theory must be falsifiable, consistent and predictive. This forms the basis of modern scientific development, where only hypotheses containing a form of prediction are then tested against observation reports, count as scientific progress. In a later publication 'C*onjectures and refutations'* Popper expanded this theme to include the precept that we can learn from our mistakes and through this knowledge science can progress. Therefore Popper's postulation is one where scientific determination is directed and must therefore always be testable. Popper's theory of falsifiability argues that scientific development is not an inductive process, but rather deductive, where science develops through analysis of problems attacked by bold conjecture rather than observations proceeding to generalisations (see Section 2.1.1).

Thomas Kuhn, (1970) introduced notions such as paradigms and scientific revolutions, where scientific advancement involves discoveries that may change the field radically. The radical change is demonstrated in the development of digital forensics from an investigation into static data, to network digital forensics, where the data is moving and volatile. As predicted by Kuhn, the development of digital forensics has shown a lexicological change that has incorporate complex technological advancement. Examples of which are increases in data volumes and transmission of data across interconnected computer systems. The proprietary, vendor driven development can be seen as evolutionary, which was predicted by Kuhn as the logical result of rapid technological revolution.

5.4.2 Forensic Readiness Criteria

The term forensic readiness in the research refers to criteria relating to evidence admissibility. The focus of the digital forensic investigator is two-fold, to collect data that can establish culpability, and also that the data be collected in a manner that is acceptable as expert scientific evidence to a court. Thus the digital forensic role intersects with both computers and law, and the digital forensic investigator should always be aware of admissibility criteria and the ramifications of the tools that the investigator is utilizing.

- The conclusions formed must be reproducible by independent third parties (see Section 2.2.3)
- Opinions contained within the forensic report must be based on properly documented digital sources (see Section 2.2.3)
- The scientific procedure must be independently tested (see Section 2.2.4).
- The scientific procedure should be published and subjected to peer review (see Section 2.2.4).
- Is there a known error rate or potential to know the error rate associated with the use of the scientific procedure? (see Section 2.2.4)
- Are there standards and protocols for the execution of the methodology of the scientific procedure? (see Section 2.2.4)
- Is the scientific procedure generally accepted by the relevant scientific communities? (see Section 2.2.4)
- No data should be changed by the actions of the forensic investigator (see Section 2.4.4).
- If original data is to be accessed, it must be performed by competent forensic investigators who must be able to explain the implications of their actions in evidence (see Section 2.4.4).
- A complete record of all applied processes must be created, and this audit trail should enable a third party to achieve the same results (see Section 2.4.4).
- The lead or charge forensic investigator must take overall responsibility for ensuring these principles as well as the appropriate legislative directives are followed (see Section 2.4.4).

5.4.3 Evidential Exceptions

The results of test Case TC008-1 (see Section 4.4.8) and test case TC009-2 (see Section 4.4.9) are exceptions. Noting the exceptions is important, because the circumstances which generated the capture of a complete dataset could be identified. This information may then be published as a method for capturing datasets with a zero error rate.

TC008-1: This is the first iteration of test case scenario TC008, where 512 ICMP echo requests were generated and sent directly to the Wireshark capture device. The resultant network artifact captured by the Wireshark tool in this case performed flawlessly and captured all 512 of the packets sent. This result would have determined a not fail result if this test case iteration was taken as a complete set. As part of the consideration of the limitations of the research methodology, it was however determined that there was a reliability requirement for these assessments to produce stable and consistent results. Thus, the primary mitigation of the reliability limitation was to administer the same test twice in the results to be correlated in order to examine stability (see Section3.4.1).

Thus, when the result from TC008-1 is compared to the second iteration in this test case scenario, TC008-2, the zero error rate result is not repeated as TC008-2 produced an error rate of 15.63% (See Table 4.8 Row TC008-2). In this iteration, the same number of 512 ICMP echo requests were generated and sent directly to the Wireshark capture device, thus generating an equivalent baseline as TC008-1. TC009-2: This is the second iteration of test case scenario TC009, where 1024 ICMP echo requests were generated and sent directly to the Wireshark capture device. The resultant network artifact captured by the Wireshark tool in this case performed flawlessly and captured all 1024 of the packets sent. This result would have determined a not fail result if this test case iteration was taken as a complete set. As part of the consideration of the limitations of the research methodology, however, it was determined that there was a reliability requirement for these assessments to produce stable and consistent results. Thus, the primary mitigation of the reliability limitation was to administer the same test twice in the results to be correlated in order to examine stability (see Section3.4.1).

Thus, when the result from TC009-2 is compared to the first iteration in this test case scenario, TC008-1, the zero error rate result is not repeated as TC009-1 produced an error rate of 4.00% (See Table 4.9 Row TC009-1). In the TC009-1 iteration, the same number of 1024 ICMP echo requests were generated and sent directly to the Wireshark capture device, thus generating an equivalent baseline as TC009-2. The results of test cases TC008 and TC009 show the effectiveness of integrating procedures that mitigated reliability limitations by administering the same test multiple times. The two results TC008-1 and TC009-2, indicate that a baseline test of a tool being utilised in a digital network forensics arena should be tested more than once in order to determine a calibrated level of error.

5.4.4 Forming an Opinion

The research has been designed with the precepts established by the Daubert standards and the principles of Digital evidence such as reproducibility and complete documentation of digital sources. When these precepts are combined with the fact that there are very few if any tools designed specifically for Digital network forensics it becomes problematic when determining an acceptance spectrum. As the acceptance spectrum concept for the research is a binary state of fail/non-fail, the research contained within this thesis is aligned with Popper and his views on falsifiability and the development of scientific knowledge.

The research within this thesis became, therefore, focused on whether the tools being tested contained any errors. The importance is, when considering requirements for expert evidence in court, completeness and reproducibility. When considering the field of digital forensics as an intersection between law and computing, the gathering of evidential data is only the first part of the digital forensic investigators task. The culmination of the role of the forensic investigator is in the generation of a forensic report, with the expectation of the report being legally acceptable. If the focus of the entire forensic investigation loses sight of these precepts, the report is in danger of becoming inadmissible in court. The admissibility of any evidence is ultimately determined by the Judge presiding, and as one of the particular important steps of producing of digital forensic evidence is the determination of whether there is a known error rate associated with the use of scientific procedure, without this expression of error rate, the evidence may become inadmissible.

The outcome of these experimental test cases were not in any way to be used as a criticism of the tool being tested, but rather as a determination that there are significant error rates. The IDS under test was not designed as a forensic tool, but it does however generate a log which can be used as evidence. The IDS tool was not configured to take full advantage of the processing power of the system, instead was utilised in an ’out of box,’ standard configuration. The packet capture tool was also used without specific configuration. Whilst the packet capture tool is a network specific tool, and is used as a forensic tool by forensic investigators, the IDS packet management system is not designed specifically for forensic investigation.

It is, however, the opinion of this researcher that any expert evidence presented in a court of law must be accompanied by an error rate. Collected data must be compared to a baseline, which should be generated in a systematic manner and regularly applied. Comparisons will generate error rates. Error rates will allow an adjudicator to form an opinion on admissibility with pertinent and accurate data. Thus, it makes sense to the researcher that a known packet structure is generated, injected onto the network, and the resulting capture be compared to the generated data volumes. The error rate thus determined can form the basis of a known error rate and this process could be used in order to calibrate any digital network forensic tool.

5.5 PERSONAL JOURNEY IN REVIEW

My personal journey begins with a research idea, to test for the best network data gathering forensic tools. My literature review process began by searching for broadly related terms on Scholarly Commons, such as ‘Forensic Readiness,’ ‘Ethernet Frames Forensics’ and ‘Daubert Forensics’ as search terms. I read that the role of digital forensic analysis is very similar to other forms of forensic analysis such as ballistics and firearm forensics. It became very clear at the beginning of the research into the literature review that digital forensics has developed in a manner that is different from development of scientific knowledge as a whole. This lead to a discourse on forensic science that linked the Daubert principles with Karl Popper (Crispino, Ribaux, Houck, & Margot, 2011). Having already read Popper’s book ‘*The logic of scientific discovery’ (Popper, 1959)* and understood the set of methodological rules which he termed falsificationism, I formed the opinion that Popper’s philisopy of scientific development through the concept of falsifiability had not been applied to the digital forensic domain. Upon discussion, my supervisor directed me towards Thomas Kuhn so I could gain an alternative viewpoint. I read ‘*The structure of scientific revolutions*’ (Kuhn, 1970) and I found that Kuhn’s philosophy of evolution and paradigm shift could also apply to the development of digital network communications, and therefore, network digital forensics. I learned that discussing my readings with my supervisor was, as always, beneficial and the introduction to Kuhn added a further dimension to my interpretation of the scientific development of digital forensics.

After muttering for a while that digital forensics was metaphysics at best, I saw that my focus, and thus the direction of my research, had to change. When the lack of structured scientific development as postulated by Popper was combined with the principles of the Daubert test, I determined that it was the lack of error rate testing that was missing. I found that there was little published literature when ‘Digital Forensic Error Rates’ was searched on Scholarly Commons. The first, and most pertinent paper shown in the search results, questioned why, if error rates is such a simple concept, aren’t there error rates for forensic tools (Lyle, 2010). Thus, the research direction shifted towards the investigation into whether there were any error when collecting a known dataset from a network environment. The new research direction satisfied a primary test of falsifiability, that the forensic tool would capture a complete dataset without errors. The new research direction also conformed with the Daubert principle of determining if there was an error rate associated with the scientific process, or tool (see Section 2.2.3). I learned from this part of the research process, that the literature review provided the primary influence upon my research direction, which changed upon reading about Popper and Kuhn and led me to the investigation of legal admissibility (see Section 2.2.2).

The literature review was then used to identify existing gaps and weaknesses of the topic of digital network forensic error checking. The choice of literature I chose for inclusion in this thesis influenced the questions and hypotheses and formed the direction of the methodology. The primary falsifiability or assertion statement this thesis addresses became: ‘That the tools under test will generate zero errors regardless of throughput’ (see Section 3.3.3 and Section 4.1.3). When the review of the data collection and analysis phase of the data map (see Figure 3.3) was complete, it became clear that the acceptance spectrum was not suitable. Thus, the experimental phases of this thesis was designed to produce findings that were binary in concept, where the results were either false or not false. The binary findings were used to determine whether there were any error rates at all which became the overarching premise of the research. Thus, the objective was to design a methodology that would produce results that would not only satisfy Popper’s determination of scientific development, but also the criteria for legal scientific evidence admissibility (see Section 2.1.1 and Section 2.2.2). Investigating similar studies and scrutinising the methodological approach allowed me to develop the methodology used to test the hypotheses formed. The studies also directed me towards the development of an assertion test bench, or a standardized hardware platform to ensure network hardware variables were controlled when individual test cases were performed. Thus, I learned to not only satisfy my research objectives, but also to create transference, where the findings of the research could be added to by being adapted for future researchers.

After completion of the experimental phase of the research and documenting the findings (see Chapter 4) an analysis showed that errors were detected in each test case, with two noted exceptions (see Section 5.4.3). The analysis shows the need for further research on the error rates associated with the use of network digital forensic tools and the codification and publication of the results which will benefit both the digital forensic community and the legal profession. Therefore, I have learned that a study of forensic tool error rates will have an impact on disciplines outside of the digital forensics and that continual appreciation of legal ramifications is an important part of the digital forensic process.

In conclusion, my personal journey involved change and adaptation, such as Kuhn philosophised as a facet of scientific development. I focused the intention of the research to an assertion based test, which conformed with Popper’s concept of falsifiability. The assertion that the forensic tools would perform without errors was tested on 18 Test Cases, from which the finding show significant errors. My view of these findings changed during the research process, where the implication was about the detection of an error, rather than the level of errors. A major influence on my research was the continual awareness of the legal implications of being forensically ready. This meant a continual awareness and consideration for the criteria of legally acceptable scientific evidence (see Section 2.2.2 Section 2.2.3 Section 2.2.4 and Section 5.4.2) I have also learned the importance of preliminary, exploritive results that will allow further research to build upon the methodology, producing a codified, tabulated findings list. From these findings, digital forensics as a discipline may continue to develop scientifically, with continual testing and the publication of the capabilities of network digital tools and their error rates.

5.6 CONCLUSION

Chapter 5 has discussed the findings reported in Chapter 4 where the evidence generated was tested against the Hypotheses. The results showed that Hypothesis H1 and H2 both failed. The information was then used to answer the sub-questions, the answer was NO to both SQ1 and SQ2. Finally, the Research Question: RQ *Can the Network Management System and the Network Packet Capture tool, achieve zero errors for digital forensic purposes?* Resulted in a NO. Chapter 6 will conclude the research by investigating possible limitations to the research in Section 6.1 and then suggest avenues for future research in Section 6.2 and a final conclusion will be formed in Section 6.3.

Chapter 6
CONCLUSION

6.0 INTRODUCTION

The research has evaluated two network digital forensic tools, a packet **Capture** tool and a packet **Management** system. The literature review formed the basis of the research question and testable assertions (see Chapter 2 and Section 4.2.3). A review of similar studies formed the basis of the methodology used to test the assertions (see Section 3.1). The findings from the tests were then used to test the Hypotheses and then to answer the research question. The findings showed that the Hypotheses FAILED and the answer to the research question: Can the Network Management System and the Network Packet Capture tool, achieve zero errors for digital forensic purposes? Was NO (see Section 5.3). Chapter 6 presents a final conclusion to the research. Section 6.1 discusses the limitations of the research, in terms of scope, perception and generalization. Section 6.2 suggests areas that would benefit from future research, such as live testing, tools, test frameworks and the establishment of error rates. Finally, Section 6.3 discusses a final conclusion to the research.

6.1 LIMITATIONS OF RESEARCH

It is important to note the possible methodological limitations of the test cases used in the following sub-Sections, and identify those limitations that had the greatest impact on transfer. It is difficult to determine the degree to which these different factors have limited the findings and thus, the extent to which the research question and hypotheses are adequately addressed. The following potential limitations are outlined: scope, perception, and generalisation.

6.1.1 Scope

Assessing the findings as a binary result was a change from the original acceptance spectrum (see Section 4.1.4) The binary results were set as a response to the Daubert standard that asked whether there were error rates as the first part of the statement. The second part of the statement is, if so what are the reported error rates. (see Section 2.2.4) The acceptance spectrum and consequent establishment of error rates is a suggestion for future research The scope of this research, and the extent of the development of the test cases were not only relevant to the legal standards of admissibility, but contained a falsifiability prediction in the form of a testable assertion (see Section 4.2.3)

The assertion test bench was designed to be as simple as possible. The rationale behind the decision for a simple hardware setup, was to mitigate hardware changes influencing the data gathered. The straightforward design may also assist knowledge transfer for future tool testing using a common test platform. The tools tested were left in their most basic set up, installed and updated to the latest release. This was in order to simulate a business installation of the tool, and with minimal configuration (see Section 3.2.1).

The tools selected for testing within the research were selected for their inclusion in one of the two delineations of packet capture and packet management. The tool's popularity, and general acceptance within the digital forensic profession was also considered. A comprehensive packet generation tool suite, that is industry recognised and reliable, performed the packet delivery. The packet generator is effective in use, as can be seen in the screenshots for each test case scenario. Each new test case changes only one variable. This is in keeping with the NIST principles and the testable assertions (see Section 4.2.3).

6.1.2 Perception

During the research phase, it was found that there were very few, if any, digital network tools that specialised in forensic applications. The process of tool selection may have potential limitations due to the perception of the tool selected not being valid for previous forensic investigation use. However, the consideration of log files and .pcap files is an essential part of the forensic investigation process (see Section 3.1.2 and Section 3.1.3 and Section 3.1.4). A series of log file outputs from the IDS and .pcap files from the packet capture tool were used to generate a fail/not fail result in order to test the hypotheses. As part of a continuing process of development within the field of networked digital forensics there will be a continual adaptation of tools to cope with the problems associated with large volumes of volatile data, each of which will need to be tested and included in a taxonomical database of forensic tools for network situations (see Section 2.9.5 and Section 2.9.6).

6.1.3 Generalisation

There is an inherent generalisation contained within the methodology where two types of tool were tested: a packet **Capture** tool and a packet **Management** system (IDS). The generalization of the two types of tool was described in Section 3.1.1 which provided a two part delineation based upon a differentiation of use. The delineation is defined as sniffing tools and scanning tools (see Section 3.1.1). These tools have the ability to continuously record information, which can be considered by the Forensic investigator post-incidence. These two types of tools were therefore tested for forensic readiness requirements of complete data sets and potential error rates, when capturing live, volatile data (see Section 2.2.4 and Section 5.4.2) Thus the selected tools represented one of the two delineations of network forensic capabilities: packet **Capturing** and packet **Managing** (see Section 3.1.1).

6.2 FUTURE RESEARCH

The limitations discussed in section 6.1 may be mitigated through future research. The following subsections discuss areas for potential future research, live testing, tools, test framework, and finally establish error rates.

6.2.1 Live Test

Future research would benefit from the gathering of data from a live network situation and comparing it to information gathered on the assertion test bench. The information gathered will help determine volume levels and data throughput in a real-world scenario, which could then in turn be injected into the assertion test bench as background noise. A live test will also help determine the configuration requirements for the tool testing. A baseline configuration may then be developed that will optimise the tool across networks, designed for forensic readiness.

6.2.2 Tools

The investigation of many tools provides data that would benefit the network digital forensic community. Future research may then determine the tools providing the minimum, or even zero errors for a certain task, such as packet capture. Future research may also eliminate the forensic use of tools that are unsuitable by testing for excessive errors. Future research may provide a taxonomical database of network digital tools and their associated rates of error. Such a database would assist the legal profession and the digital forensic community to establish informed decisions of which tool will provide the best form of forensic readiness.

6.2.3 Test Framework

The development of future research into a test framework of multiple tools will assist forensic investigators to decide which tool is best for the specific requirements, prior to use. Potentially including such concepts as a live network calibration guide, an assertion test bench hardware setup and step-by-step forensic data collection instructions. Each of these concepts would formulate a standardised approach to testing networked digital forensic tools. A test framework will also assist researchers confirming the results from previous research, adding to the network digital forensic tool knowledge base in a scientific manner.

6.2.4 Establish Error Rates

The establishment of known error rates is a significant criterion for evidence admissibility and therefore forensic readiness. Digital forensic information produced as scientific evidence must be reproducible by independent third parties and the sources must be complete (see Section 2.2.3). Should the forensic information be incomplete, the Daubert standard states that the scientific information must include an error rate (see Section 2.2.4). Utilising tools and procedures to formulate expert opinions for digital forensic purposes are required to withstand judicial scrutiny. This means that the sources of information used to uncover evidence should have independent testing to determine error rates that are acceptable to the scientific and legal community. Therefore, it is suggested that future research be undertaken to establish error rates of tools that can be adapted to deliver network digital forensic readiness and to maintain legal credibility.

6.3 PERSONAL JOURNEY: CONCLUSION AND DIRECTION

There are indications that Australia are adopting similar response to data breaches as have been enforced in the United Kingdom and the United States. A new law has been passed in Australia, the Notifiable Data Breaches Bill, on 13th February 2017. The new law states that any organization that is accountable to the privacy act will be required to inform the Australian Information Commission as well as members of the public if the organization has been subjected to a data breach. Whilst this is 15 years after similar law was passed in the United States and 3 months after the United Kingdom also passed similar legislation. The onus of this legislative focus is upon the organisation to provide evidence of any data breaches, maintained in a data breach incident log.

The ramifications of such legislative direction are that an organization will soon be required to become forensically ready, as described in Section 2.2 of my thesis. Complete datasets are required, and there will be little pre-emptive triage opportunities available. As my research has shown, there are errors with digital forensic tools capturing datasets from a networked environment such as is common across most organisations. It would be a rare business organization that runs with disparate and disconnected computers. Therefore, the only fail-safe method, when providing business forensic readiness is to capture the data-stream in its entirety. As identified in Section 2.9.5 and Section 2.9.6 volume growth and collection of live data is an issue to the collection of forensic data. I have discussed the potential issues of this compliance regulation in Section 2.5.1.

This will place the technological challenge of collecting complete data stream information directly in the laps of businesses, and present a challenge for future investigation. The amount of data storage required to capture all packets, incoming and outgoing is daunting. Forensic acquisition has failed to scale with this growth, especially in areas of IO rates, where the network transmission speed may overwhelm a disc write rate. Whilst forensic imaging was defined over a decade ago, where linear and complete image integrity is protected by a hash function in the field this is not possible. At gigabit per second network transmission speed the disc write rate will be 128 MB, which is approaching current theoretical max. This causes latency during the acquisition of forensic data process, which can cause data loss through Ethernet frames being dropped. This causes potential problems of evidential acceptance due to nonlinear and potentially partial information gathering. This becomes a serious issue upon the collection of information from a business network containing hundreds of machines where many terabytes of data per hour may need to be recorded.

Thus, there can be seen to be an urgent need for network digital forensic tools to be tested, and for a cohesive and integrated dataset capturing process to become optimized for business forensic readiness.

6.4 CONCLUSION

It is not the level of errors of the tools that were under investigation in this thesis that is important. It is the fact that the tools produce errors at all. The protocols behind tool testing for static digital forensics is well established but there is very little transfer from these tools to the networked digital domain. These protocols are designed to establish error rates in order to qualify the information ready for reporting as expert evidence in the legal domain. Therefore, there is a requirement for any tool that cannot demonstrate a zero error rate for that rate to be expressed and with the presentation of expert evidence gathered from the use of the tool. Technology is only one part of the realm of digital forensics, where the second part is the legal realm. The ultimate test for the digital forensic investigator is whether a judge will allow the investigators report to be introduced as expert evidence. An incomplete dataset is in danger of being considered deficient evidence in terms of expectations of replication and completeness, when considered under the Daubert requirements. These incomplete datsets can be mitigated when an error rate is determined and declared. The research presents results that show there are errors when using two popular network tools in a network forensic readiness application.

REFERENCES

Alzubaidi, W. K., Cai, L., & Alyawer, S. A. (2014, 3-5 June 2014). Enhance the performance of ICMP protocol by reduction the IP over ethernet naming architecture Symposium conducted at the meeting of the Computer and Information Sciences (ICCOINS), 2014 International Conference on doi:10.1109/ICCOINS.2014.6868392

Alzubaidi, W. K., Cai, L., Alyawer, S. A., & Siebert-Cole, E. (2015). Visibility for Network Security Enhancement in Internet Protocol Over Ethernet Networks. *Advanced Computer & Communication Engineering Technology*, 277.

Amann, P., & James, J. I. (2015). Designing robustness and resilience in digital investigation laboratories. *Digital Investigation, 12, Supplement 1*, S111-S120. doi:http://dx.doi.org/10.1016/j.diin.2015.01.015

Association of Chief Police Officers (ACPO). (2012). Good practice guide for computer based electronic evidence. Retrieved from http://www.7safe.com/electronic_evidence/ACPO_guidelines_computer_evidence.pdf

Balogun, A. M., & Zhu, S. Y. (2013). Privacy Impacts of Data Encryption on the Efficiency of Digital Forensics Technology.

Bates, J. (1998). Fundamentals of computer forensics. *Information Security Technical Report, 3*(4), 75-78. doi:http://dx.doi.org/10.1016/S1363-4127(98)80040-X

Bertasi, P., & Zago, N. (2013, 2-6 Sept. 2013). FASTDD: An Open Source Forensic Imaging Tool Symposium conducted at the meeting of the Availability, Reliability and Security (ARES), 2013 Eighth International Conference on doi:10.1109/ARES.2013.63

Bhoedjang, R. A. F., van Ballegooij, A. R., van Beek, H. M. A., van Schie, J. C., Dillema, F. W., van Baar, R. B., . . . Streppel, M. (2012). Engineering an online computer forensic service. *Digital Investigation, 9*(2), 96-108. doi:http://dx.doi.org/10.1016/j.diin.2012.10.001

Caloyannides, M. (2006). Digital "Evidence" is Often Evidence of Nothing. In *Digital Crime and Forensic Science in Cyberspace* (pp. 334-339). Hershey, PA, USA: IGI Global. Retrieved from http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-59140-872-7.ch015. doi:10.4018/978-1-59140-872-7.ch015

Casey, E., Katz, G., & Lewthwaite, J. (2013). Honing digital forensic processes. *Digital Investigation, 10*(2), 138-147. doi:http://dx.doi.org/10.1016/j.diin.2013.07.002

Changwei, L., Singhal, A., & Wijesekera, D. (2014). Relating admissibility standards for digital evidence to attack scenario reconstruction. *Journal of Digital Forensics, Security & Law, 9*(2), 181.

Crispino, F., Ribaux, O., Houck, M., & Margot, P. (2011). Forensic science - A true science? [Article]. *Australian Journal of Forensic Sciences, 43*(2/3), 157-176. doi:10.1080/00450618.2011.555416

Daubert v. Merrell Dow Pharmaceuticals [1993] 509 U.S. 579 (1993).

Dye, M. A., McDonald, R., & Antoon W Rufi. (2008). *Network fundamentals, CCNA exploration companion guide*. Indianapolis: Cisco Press.

Dye, M. A., McDonald, R., & Rufi, A. W. (2008). *Network fundamentals : CCNA exploration companion guide* [Non-fiction

Computer File]: Indianapolis, Ind. : Cisco Press, [2008]. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=cat05020a&AN=aut.b11428594&site=eds-live. Retrieved from cat05020a database.

Elyas, M., Ahmad, A., Maynard, S. B., & Lonie, A. (2015). Digital forensic readiness: Expert perspectives on a theoretical framework. *Computers & Security, 52*, 70-89. doi:http://dx.doi.org/10.1016/j.cose.2015.04.003

Elyas, M., Maynard, S. B., Ahmad, A., & Lonie, A. (2014). Towards a systemic framework for digital forensic readiness [Article]. *Journal of Computer Information Systems, 54*(3), 97-105.

Guo, Y., Slay, J., & Beckett, J. (2009). Validation and verification of computer forensic software tools—Searching Function. *Digital Investigation, 6, Supplement*, S12-S22. doi:http://dx.doi.org/10.1016/j.diin.2009.06.015

Hoque, N., Bhuyan, M. H., Baishya, R. C., Bhattacharyya, D. K., & Kalita, J. K. (2014). Review: Network attacks: Taxonomy, tools and systems [Review Article]. *Journal of Network and Computer Applications, 40*, 307-324. doi:10.1016/j.jnca.2013.08.001

Hunt, R., & Zeadally, S. (2012). Network forensics: An analysis of techniques, tools, and trends. *IEEE Computer, 45*(12), 36-43.

ISO/IEC 7498-1:1994(E). *Information technology - Open systems interconnection - Basic reference model: The basic model*: ISO copyright office. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx?

ISO/IEC. (2014). *ISO/IEC 30121:2014 Information technology - Governance of digital forensik risk framework* [Standard]: ISO copyright office. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx? Retrieved from edsbsi database.

Karie, N. M., & Venter, H. S. (2014). Toward a General Ontology for Digital Forensic Disciplines. *Journal of Forensic Sciences (Wiley-Blackwell), 59*(5), 1231.

Kazadi, J. M., & Jazri, H. (2015). Using digital forensic readiness model to increase the forensic readiness of a computer system. *2015 International Conference on Emerging Trends in Networks & Computer Communications (ETNCC)*, 131.

Kiravuo, T., Sarela, M., & Manner, J. (2013). A Survey of Ethernet LAN Security. *IEEE Communications Surveys & Tutorials, 15*(3), 1477.

Kohn, M. D., Eloff, M. M., & Eloff, J. H. P. (2013). Integrated digital forensic process model. *Computers & Security, 38,* 103-115. doi:http://dx.doi.org/10.1016/j.cose.2013.05.001

Konstantinos, V., Emmanouil, M., & Vassileios, C. (2013). A Model for Hybrid Evidence Investigation. In *Emerging Digital Forensics Applications for Crime Detection, Prevention, and Security* (pp. 150-165). Hershey, PA, USA: IGI Global. Retrieved from http://services.igi-global.com/resolvedoi/resolve.aspx?doi=10.4018/978-1-4666-4006-1.ch011. doi:10.4018/978-1-4666-4006-1.ch011

Kuhn, T. S. (1970). *The Structure of Scinetific Revolutions*. Retrieved from https://books.google.co.nz/books?id=tAakQwAACAAJ

Lang, A., Bashir, M., Campbell, R., & DeStefano, L. (2014). Developing a new digital forensics curriculum. *Digital Investigation, 11, Supplement 2*, S76-S84. doi:http://dx.doi.org/10.1016/j.diin.2014.05.008

Lee, H. (2001). *Henry lee's Crime Scene Handbook*. NY: Academic Press.

Lyle, J. R. (2010). If error rate is such a simple concept, why don't I have one for my forensic tool yet? [Article]. *Digital Investigation, 7*(Supplement), S135-S139. doi:10.1016/j.diin.2010.05.017

Mouhtaropoulos, A., Dimotikalis, P., & Li, C.-T. (2013). Applying a Digital forensic readiness framework: Three case studies. *2013 IEEE International Conference on Technologies for Homeland Security (HST),* 217.

Mouhtaropoulos, A., Li, C.-T., & Grobler, M. (2014). Digital Forensic Readiness: Are We There Yet [article] [article]. 173. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edshol&AN=hein.journals.jcolate9.20&site=eds-live&scope=site

Nikkel, B. J. (2005). Generalizing sources of live network evidence [Article]. *Digital Investigation, 2,* 193-200. doi:10.1016/j.diin.2005.08.001

Nikkel, B. J. (2014). Fostering incident response and digital forensics research. *Digital Investigation, 11*(4), 249-251. doi:http://dx.doi.org/10.1016/j.diin.2014.09.004

NIST. (2001). General test methodology for computer forensic tools. Retrieved from www.cftt.nist.gov/Test%20Methodology%207.doc

NIST. (2014). *Improving critical infrasructure cybersecurity executive order 13636*. Retrieved from www.nist.gov/itl/upload/preliminary-cybersecurity-framework.pdf

Oh, M., Kim, Y. G., Hong, S., & Cha, S. (2012). ASA: agent-based secure ARP cache management. *IET Communications, 6*(7), 685.

Palmer, G. (2001). A road map for digital forensic research Symposium conducted at the meeting of the First digital forensic research workshop (DFRWS 2001), Utica, NY.

Pandey, A., & Saini, J. R. (2012). Counter Measures to Combat Misuses of MAC Address Spoofing Techniques. *International Journal of Advanced Networking & Applications, 3*(5), 1358.

Philipp, A., Cowen, D., & Davis, C. (2010). *Hacking Exposed Computer Forensics, Second Edition*: McGraw-Hill, Inc.

Pilli, E. S., Joshi, R. C., & Niyogi, R. (2010). Network forensic frameworks: Survey and research challenges. *Digital Investigation, 7*(1–2), 14-27. doi:http://dx.doi.org/10.1016/j.diin.2010.02.003

Pooe, A., & Labuschagne, L. (2012). A conceptual model for digital forensic readiness. *2012 Information Security for South Africa*, 1. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=edb&AN=86577324&site=eds-live&scope=site

Popper, K. (1959). *The logic of scientific discovery*. London: Hutchinson.

Quick, D., & Choo, K.-K. R. (2014). Impacts of increasing volume of digital forensic data: A survey and future research challenges. *Digital Investigation, 11*(4), 273-294. doi:http://dx.doi.org/10.1016/j.diin.2014.09.002

Rogers, M. K., Goldman, J., Mislan, R., Wedge, T., & Debrota, S. (2006). Computer Forensics Field Triage Process Model. *Proceedings of the Conference on Digital Forensics, Security and Law*, 27-40.

Roussev, V., Quates, C., & Martell, R. (2013). Real-time digital forensics and triage. *Digital Investigation, 10*(2), 158-167. doi:http://dx.doi.org/10.1016/j.diin.2013.02.001

Rowlingson, R. (2005). A ten step process for forensic readiness. *International Journal of Digital Evidence*(2), 1-28.

Salim, H., Li, Z., Tu, H., & Guo, Z. (2012). Preventing ARP Spoofing Attacks through Gratuitous Decision Packet. *2012 11th International Symposium on Distributed Computing & Applications to Business, Engineering & Science*, 295.

Schwerha, J. J. (2004). Cybercrime: Legal Standards Governing the Collection of Digital Evidence. *Information Systems Frontiers, 6*(2), 133-151. doi:10.1023/B:ISFI.0000025782.13582.87

Stoney, D. A., & Stoney, P. L. (2015). Critical review of forensic trace evidence analysis and the need for a new approach. *Forensic Science International, 251*, 159-170. doi:http://dx.doi.org/10.1016/j.forsciint.2015.03.022

Tan, J. (2001). Forensic readiness. *Cambridge, MA:@ Stake*, 1-23.

The Institute of Electrical and Electronic Engineers, I. (2012). IEEE 802 Local and metropolitan area network standards. Retrieved from http://standards.ieee.org/getieee802/802.3.html

Whitman, M. E., & Mattord, H. J. (2016). *Principles of information security* [BibliographiesNon-fiction]: Australia : CengageLearning, [2016]Fifth edition. Retrieved from http://ezproxy.aut.ac.nz/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=cat05020a&AN=aut.b14502987&site=eds-live.

Wilsdon, T., & Slay, J. (2006). *Validation of Forensic Computing Software Utilizing Black Box Testing Techniques*. presented at the meeting of the Australian Digital Forensics Conference, Retrieved from http://ro.ecu.edu.au/cgi/viewcontent.cgi?article=1036&context=adf

Yusoff, Y., Ismail, R., & Zainuddin, H. (2011). Common phases of computer forensics investigation models. *International journal of computer science and information technology, 3*(3), 17-31. doi:10.5121/ijcsit.2011.3302