

Locating and Extracting Digital Evidence from Hosted virtual desktop Infrastructures: Cloud Context

NIRBHAY JAWALE
B.C.I.S (AUT University) New Zealand

A thesis submitted to the graduate faculty of Design and Creative Technologies
AUT University
in partial fulfilment of the
Requirements for the degree of
Master of Forensic Information Technology

School of Computing and Mathematical Sciences

Auckland, New Zealand
2010

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.

.....
Signature

III

Acknowledgements

This thesis was conducted at the Faculty of Design and Creative Technologies in the school of Computing and Mathematical Sciences at AUT University, New Zealand. During the course of writing this thesis, I have received a valuable support from many people. Firstly, I would like to thank my family for their blessings, encouragement and believing that I could get through this study.

Secondly, I am deeply thankful and honored by having Professor Ajit Narayanan as my thesis supervisor. This thesis would not have been possible without his guidance, encouragement and interest; he took to supervise my thesis topic. I very much enjoyed the critical discussions and comments on my research topic with Professor Ajit Narayanan, this thesis would not have been in the present form without his valuable contribution. I would also like to thank Dr Brian Cusack for his ongoing support during the course of Masters in Forensic IT. He has been extremely helpful during my PG1 approval, without his input this research topic would not have been approved in the first place.

Lastly, I would like to thank everyone once again from the bottom of my heart for their support, encouragement and enthusiasm that has kept me going throughout this period. I also greatly appreciate all the MFIT staff and lecturers for sharing their knowledge and concepts; this has deepened my knowledge at a critical point of my life.

Abstract

The development of virtualization started in 1960, when VMware introduced partitioning of large mainframes for better hardware utilization. (Virtualization History, 2010) Since then virtualization has matured and been adopted to a wide extent in the industry. Recent developments include branching into areas of server virtualization, storage and application virtualization and, very recently, desktop virtualization. Desktop virtualization has so far been through two models: the Client hosted model, which is typically operated from the user's workstation using Windows Virtual PC; and the VMware workstation or Java Virtual Machine (VM). However, recently a third model has emerged, called the server hosted model or Hosted virtual desktop (HVD), which is a virtualized form of desktop (VM) delivered to users from the cloud infrastructure. In other words virtualization in computing has progressed to an extent where desktops can be virtualized and accessed from anywhere. The server hosted model has already surpassed 1% market share of the worldwide professional PC market, with estimates indicating that this is a rapidly growing area.

This study investigates the adequacy of current digital forensic procedures on hosted virtual desktops (HVDs) as there does not appear to be specific methods of locating and extracting evidences from this infrastructure. Using the Forensic Iterative Development Model (FIDM), HVDs deployed in private cloud were simulated to reflect three different computer crime (quasi-experimental) scenarios. It was found that current digital forensic procedures may not be adequate for locating and extracting evidence, since the infrastructure in scenario 2 and 3 introduces complications such as non-persistent disk modes and segregating data in a multi-tenant environment. However in scenario 1, findings illustrate that all standard investigation techniques can be followed as a result of the persistent user environment. Furthermore, suggestions are made to extend the current research in the areas of techniques to acquire virtual machines from hypervisors, hashing evidence and forensic readiness in environments consisting HVDs.

Table of Contents

Declaration.....	ii
Acknowledgements.....	iii
Abstract.....	iv
List of Figures.....	viii
List of Tables	ix
List of Abbreviations	x

Chapter 1 - Introduction

1.0 INTRODUCTION	1
1.1 PROBLEM DEFINITION.....	2
1.2 MOTIVATION.....	3
1.2.1 Hosted virtual desktops and Forensics	3
1.2.2 Forensic Challenges in Virtualized Environments	4
1.2.3 Thinking of What the Future Holds.....	5
1.3 CONCLUSION - THESIS STRUCTURE	5

Chapter 2 - Literature Review

2.0 INTRODUCTION	8
2.1 THE CURRENT STATE OF DIGITAL FORENSICS.....	8
2.1.1 Disciplines of Digital Forensics	9
2.1.2 Investigation frameworks	11
2.1.3 Capability of Current Tools.....	14
2.1.3.1 Encase v7	14
2.1.3.2 Paraben and FTK	15
2.2 CURRENT STATE OF DESKTOP COMPUTING	15
2.2.1 Virtual Desktops.....	17
2.3 CURRENT STATE OF DISTRIBUTED COMPUTING (CLOUD CONTEXT)	19
2.3.1 Service Models	20
2.3.2 Cloud Deployment Methods.....	21
2.3.3 Virtual Desktops in Clouds	23
2.3.4 Security and Forensic Challenges.....	26
2.4 CURRENT FORENSIC CAPABILITIES.....	28
2.4.1 Hiding in a Virtual World.....	29
2.4.2 Live Digital Forensics in the Virtual World.....	31

2.4.3	Virtualization and Forensics	34
2.5	CONCLUSION.....	38

Chapter 3 - Research Methodology

3.0	INTRODUCTION	40
3.1	WHAT IS METHODOLOGY	41
3.2	RANGE OF METHODOLOGY'S.....	42
3.2.1	Grounded Theory.....	42
3.2.2	Descriptive Research Methodology.....	45
3.2.3	Case Study	46
3.2.4	Scientific Methodology (SM) - Research.....	48
3.3	RESEARCH DESIGN	50
3.4	HYPOTHESIS/PROBLEM STATEMENT	51
3.5	EXPERIMENTS AND METHODS	54
3.5.1	Scenario 1 - Persistence VM	55
3.5.2	Scenario 2 - NON-Persistence VM	55
3.5.3	Scenario 3 - Multi-Tenant - Persistent VM	56
3.5.4	Setup of Simulation	56
3.5.5	Modes of Virtual HDD	57
3.5.6	Pool Type.....	57
3.5.7	VM Allocation.....	58
3.5.8	Event Database	58
3.6	CONCLUSION.....	58

Chapter 4 - Research Findings

4.0	INTRODUCTION	59
4.1	Variations in Data requirements.....	59
4.2	System Design.....	60
4.3	SCENARIO 1 - PERSISTENT VIRTUAL ENVIRONMENT.	62
4.3.1	Preparation - General Findings.....	62
4.3.2	Search & Recognition.....	63
4.3.3	Collection.....	63
4.3.4	Examination & Analysis.....	69
4.4	SCENARIO 2 - NON-PERSISTENT VIRTUAL ENVIRONMENT.....	73
4.4.1	Preparations - General Findings	73

VII

4.4.2	Search and Recognition	73
4.4.3	Collection.....	74
4.4.4	Examination & Analysis:.....	75
4.5	SCENARIO 3 - INVESTIGATING CRIMES IN A MULTI-TENANT ENVIRONMENT	80
4.5.1	Preparations - General Findings	80
4.6	CONCLUSION.....	84

Chapter 5 - Research Discussion & Recommendation

5.0	INTRODUCTION	85
5.1	DISCUSSION OF FINDINGS	86
5.1.1	Case Scenario 1	86
5.1.2	Case Scenario 2	87
5.1.3	Case Scenario 3	88
5.2	Unforeseen Scenarios	89
5.3	RECOMMENDATIONS AND BEST PRACTICES.....	90
5.4	FIDM FEEDBACK	93
5.5	RESEARCH PROBLEM AND HYPOTHESIS.....	93
5.6	CONCLUSION.....	94

Chapter 6 - Conclusion

6	INTRODUCTION	95
6.1	LIMITATIONS.....	96
6.2	FUTURE RESEARCH	97
6.3	CONCLUSION.....	98
	APPENDIX A:.....	108

List of Figures

Figure 2.1: The NIJ vs. the DFRWS Model	12
Figure 2.2: Cloud forensics - FORZA Model	13
Figure 2.3: Two Modes of Operation for Virtual Desktops.....	17
Figure 2.4: Type 1 vs. Type 2 Hypervisor	18
Figure 2.5: Multi-Tier and Client Server Model.....	19
Figure 2.6: Visual Model of Cloud Computing	22
Figure 2.7: Basic Architecture of VDI.....	24
Figure 2.8: Multi-Tenanting.....	28
Figure 2.9: Order of tests	29
Figure 2.10: Results of Set 1	30
Figure 2.11: Results of set 2	31
Figure 2.12: Virtual Machine Applications and associated files	33
Figure 2.13: Files associated with VMware VM	33
Figure 2.14: VMware Server files	35
Figure 2.15: Registry entries by VMware application.....	36
Figure 2.16: VMware Log Locations.....	37
Figure 2.17: Mandiant Audit View	38
Figure 3.1: Iterative Cycle used in GTM	43
Figure 3.2: Interview Notes Format.....	44
Figure 3.3: Forensic Iterative Development Model (FIDM)	53
Figure 3.4: Infrastructure in VMware Workstation 7	54
Figure 4.1: VMware View Client - Connection timeout error.....	61
Figure 4.2: VMware Workstation: LAN Segment Configuration	62
Figure 4.3: Command Helix 2.0	63
Figure 4.4: Helix 2.0 - Configure netCat Listener	64
Figure 4.5: NetCat listening.....	65
Figure 4.6: Prepare suspect's HVD for Helix	66
Figure 4.7: Command Helix 2.0	66
Figure 4.8: Prepare suspects HVD for acquisition.....	67

Figure 4.9: Preparing vCenter Standalone Converter	68
Figure 4.10: Locations of VMware View logs.	70
Figure 4.11: Event Monitor Log	71
Figure 4.12: View Client Log	71
Figure 4.13: Security Logs showing accessed object	75
Figure 4.14: Disk Mode Configuration.....	77
Figure 4.15: (.vmex) file	78
Figure 4.16: Blind search on ESX host.....	81
Figure 4.17: Imaging ESX	82

List of Tables

Table 3.1: HVD configuration per scenario.....	56
Table 4.1: Scenario 1 Investigation Results.....	72
Table 4.2: Scenario 2 Investigation Results.....	79
Table 4.3: Scenario 3 Investigation Results.....	83
Table 4.4: Overview of scenarios	84
Table 5.1: Crimes in concealed case Scenarios	89

List of Abbreviations

HVD	Hosted virtual desktop
VM	Virtual Machine
CHVD	Client Hosted virtual desktop
VDI	Virtual Desktop Infrastructure
RFID	Radio Frequency Identification
ROI	Return on Investment
FTK	Forensic Toolkit
VDDK	Virtual Machine Development Kit
VMDK	Virtual Machine Disk
ESX	Enterprise Server X
API	Application Programming Interface
FIDM	Forensic Iterative Development Model
CART	Computer Analysis and Response Team
CCT	Computer Crime Team
NIJ	National Institute of Justice
ATM	Automatic Teller Machine
ACL	Access Control List
RTP	Real Time Protocol Packets
VoIP	Voice Over Internet Protocol
DFRWS	Digital Forensic Research Workshop
SWGDE	Scientific Working Group on Digital Evidence
TWGDE	Technical Working Group on Digital Evidence
DOJ	United States Department of Justice
FORZA	Digital Forensic Investigation Framework
VHD	Virtual Hard Disk
VMEM	VMware virtual machine paging file.
VMM	Virtual Machine Monitor
NIST	National Institute of Standards Technology

XI

SaaS	Software as a Service
PaaS	Platform as a Service
IaaS	Infrastructure as a Service
IRD	Inland Revenue Department
VPC	Virtual PC
SLA	Service Level Agreement
RDP	Remote Desktop Protocol
PCoIP	Personal Computer Over Internet Protocol
CSP	Cloud Service Provider
GTM	Grounded Theory Methodology
CSM	Case Study Methodology
SM	Scientific Methodology
LAN	Local Area Network
WAN	Wide Area Network
VP	Vice President
HDD	Hard Disk Drive
USB	Universal Serial Bus
E-SATA	External Serial ATA
VMFS	Virtual Machine File System
DHCP	Dynamic Host Configuration Protocol
GPO	Group Policy Object
AD	Active Directory
LUN	Logical Unit Number

Chapter 1 - Introduction

1. INTRODUCTION

In computing, virtualization has advanced in such a way that desktops can now be virtualized and accessed from anywhere. It has been established that the server hosted type desktops is already gone beyond 1 percent share of the market of the global PC market. It is estimated that this is a rapidly growing area. Virtualization development began in 1960 (Rutkowski, 2004), when VMware initiated large mainframes partitioning for enhanced hardware use. From that time on virtualization has developed and it has been utilized immensely in the industry. Current developments consist of branching into server area virtualization, application virtualization, storage and desktop virtualization (Dodge, 2006).

It is worth mentioning that desktop virtualization has successfully introduced two models. The Client-Hosted Model, which includes Virtual PC and VMware workstation, is also commonly referred to as Java Virtual Machine. The second model is the Hosted virtual desktop (HVD) or server hosted model. HVD is a desktop (VM) virtualized form that is delivered to users from cloud infrastructure. This means that the users are able to see a desktop that is identical to a standard local desktop; however, the desktop is operated and virtualized from a remote server. Both the Client-Hosted virtual desktop (CHVD) and the Hosted virtual desktop (HVD) are VDI's mode of operation (Jump & Gammage, 2009).

Businesses are increasingly considering the use of HVD solutions as a way of controlling hardware costs. It also allows them to reduce upgrading and maintenance costs: a HVD is remotely updated automatically. One is only required to have a local screen, internet connection and enough screen memory in order to run HVD. Gartner (2009) contends that HVD has already scooped 1% market share of the global professional PC market, and estimates that 16 percent of present international professional desktop PCs might migrate to HVD by 2015, equal to around 67 million related devices. It is contended that the demand to be able to access one's desktop from anywhere continues to grow significantly. Similarly the utilization of this model for private cloud computing is projected to grow as firms choose lower cost options especially with vendors who offering products that allow firms to establish their own in-house cloud.

As far as digital forensics is concerned, digital evidence is usually collected from numerous sources. The most common sources include computers, digital cameras, cell phones, hard drives, USB memory devices and CD-ROMs. There are less common sources such as black boxes, stationed inside automobiles, and digital thermometers settings that must be conserved as they are subject to change, and data retention devices such as Internet usage meters or mobile network and RFID tags (Taylor et al., 2010). This, therefore, raises questions about the way HVDs will have an effect on digital forensics especially in locating and extracting evidence (McLaughlin, 2007).

1.1 PROBLEM DEFINITION

Jump & Gammage (2009) argue that forensic experts are able to carry out procedures on the physical machines with great confidence due to the presence of physical hardware in the forensic laboratory. This means that if a system is fully virtualized, then the experts may not confidently be in a position to conduct thorough investigation. This is due to different elements of the environment, for instance, the storage of data in the cloud environment may be in a state of instability and the custody of data may not be guaranteed. Additionally, the data may not be segregated from other data as a result of resource sharing by users who may be in a multi-tenant setting (Gartner, 2009).

We are currently unsure, whether existing forensic methods are adequate for investigating crimes occurring within Hosted virtual desktop based in private clouds. It is worth noting that there is no specific method or even a set of approaches for extracting evidence especially from cloud systems. Even if previously there were several techniques used to investigate the VM presence or utilize a VM as a forensic instrument, there are few techniques that are used to investigate the cloud as an infrastructure. There is a possibility that the present digital forensic methods are adequate to investigate crimes committed within a cloud, however, up to now there seems to be no adequate systematic investigation of the digital forensic techniques and approaches.

This thesis examines the sufficiency of existing digital forensic measures on hosted virtual desktops (HVDs) as there seems to be no systematic approach for locating and extracting evidence from this particular infrastructure. The aim of this study is to investigate whether it would be possible to locate and extract digital evidence in hosted virtual desktop infrastructure

with new digital forensic techniques for investigating security breaches as well as crimes committed using HVDs.

1.2 MOTIVATION

The interest of the research is in digital forensics with regard to virtualization in computing. Secondly, HVD digital forensic is a new area that has not been exhaustively discussed, yet it seems to present difficulties especially in the location and extraction of evidence.

1.2.1 Hosted virtual desktops and Forensics

Scholars had predicted that 2010 would be a good year for hosted virtual desktops (HVD). (Messmer, 2009) predicted that there would be around 67 million hosted virtual desktops by 2015. The researcher has noted that this is currently becoming a reality. To date, utilization of desktop virtualization technology has significantly increased and the server hosted model is being employed by a growing number of customers. As Ruan & Carthy (2011) have forecast, the Return On Investment (ROI) for unrelenting hosted virtual desktops is nearly nonexistent for the majority of cases, and so untimely adopters in various industries are now looking to maximize their ROI through adopting desktop virtualization (Rutkowski, 2004).

In many circumstance during forensic investigations, the analyst often virtualized an acquired raw disk image that is available to them as a way of assisting in the investigation in order to detect and extract malware within VM's. They therefore not only use instruments such as FTK, PyFlag sleuthkit and EnCase but they also practically reconstruct the image through dd2vmdk, Liveview, raw2vmdk or pro/VFC. In August 2011, there was an upgrade of the Virtual Disk Development Kit (VDDK) 1.2.1 to 5.0 versions in order to offer support for Windows, ESXi 5.0, vSphere 5, and a huge amount of Unix-based platforms. VMware's Virtual Disk Development Kit (VDDK) is an absolute set of tools that work and manipulate VMware VMDK images, thus allows forensic analysts to code their own applications which gives a chance to directly access the VM disk (Gartner, 2009).

It is worth noting that beside VMware, other vendors such as Neocleus and Virtual Computer which are currently enjoying limited success but which might experience success to the magnitude enjoyed by VMware and Citrix when they entered the market space, might pose a

problem in the location and extraction of evidence in the same way. However, VMware is so far the most widely used vendor. VDDK essentially encompasses three major utilities: virtual machine disk (VMDK) management utility which enables growth and reduction of the VMDK image, VMDK disk mount utility that enhance VMDK images mounting and the interior Virtual disk API's that allow programmatic interface to contact VMDK image. However, from a forensic point of view, during VMDK volume mounting, the VDDK enables explicit read-only permission; therefore, it does not enable the analyst to access the VMDK disk (Henry, 2009). Conversely, pre and post hashing need to be executed in order to establish whether VMDK have been customized after the VDDK API use or not. Generally, with the VDDK APIs, one is able to get a direct access to VMDK files, thus it provide wide-ranging functionality which includes permitting arbitrary read or write access to data everywhere in a VMDK file, creation and management of redo logs, reading and writing of VMDK disk metadata, the capacity to erase VMDK files, automation and customization that allows one to decide how to one access the VMDK image instead of relying on existing utilities.

1.2.2 Forensic Challenges in Virtualized Environments

Virtualized technologies often make forensic investigation intricate. Technological advances in cloud computing, portable environments and virtualization often make it difficult to effectively trace the host system. Unlike the past, all evidence is not confined in the local hard drive and therefore, only a few traces of evidence are left in this technology. This makes the researcher wonder whether there could be effective approaches for virtualized settings which would make forensic investigation easy (Spruill & Pavan, 2007).

Basically, techniques are required to examine the implications that non-persistent environments and multi-tenanted architectures have for forensic IT procedures. It is apparent that the use of multi-tenanted architecture would have digital forensic consequences especially in protecting the privacy of those who are not involved in objectionable behaviors. This means that firms which adopt multi-tenanted architectures would be required to obtain the written employees' written consent to use their private data that is in the digital forensic investigations scope although they themselves may be not alleged to be doing something wrong. In a similar manner, the non-persistent setting environments could have advantages; nevertheless, it could affect accessibility of potential evidence (Ruan & Carthy, 2011).

1.2.3 Thinking of What the Future Holds

It seems that virtualization has an real hold on the market. Additionally, organizations are competing keenly to extend and implement products suitable for such environments. Even though this is a good move, challenges are inevitable. Numerous court systems have begun to experience difficulties due to cybercrime. It becomes extremely difficult to locate and extract evidence especially when using HVDs even though it is marvelous to utilize these new e-discovery models and products, analysts will continue miss necessary information related to virtual environments and they may end up offering unintended data. This means that there is a real possibility of challenge during investigation regarding data on virtual technologies. What would happen when people have kiosks where there are internet cafe's or other public use computers a user downloads virtual settings via a certain browser, commits a crime and afterward deletes that virtual machine. This can happen anywhere making location and extraction of evidence more difficult (Taylor et al., 2010).

1.3 CONCLUSION - THESIS STRUCTURE

The paper begins with the thesis abstract, an introductory part that includes acknowledgements, a table of contents, the list of the abbreviations, a list of figures and tables and an abstract.

1.3.1 Chapter One

In brief, chapter one lays out background information on desktop virtualization, specifically focusing on Hosted virtual desktops (HVD). It also offers some statistical facts on HVDs, the reason their use is expected to grow and a clear elaboration of digital evidence. The motivation for the research and this thesis project is also introduced and finally the structure of the entire thesis is presented in this chapter.

1.3.2 Chapter Two

This chapter offers a critical literature review of the selected research areas mainly focusing on the current state of digital forensics. The first section concerns the disciplines of digital forensics. In this section, the paper clearly elaborates on device forensics, database forensics and network forensics and it addresses the challenges of each of these elements. This is followed by an outline of various investigation frameworks. In this section, active research groups such as the National

Institute of Justice (NIJ), the Technical Working Group on Digital Evidence (TWGDE), the Computer Analysis and Response Team (CART), the Scientific Working Group on Digital Evidence (SWGDE), the Digital Forensic Research Workshop (DFRWS), and the American Department of Justice (DOJ) (Jump & Gammage, 2009) will be explained into detail. The section also compares investigative models NIJ and DFRWS. It will also examine the FORZA model which is the proposed cloud investigation forensic model. In this chapter, the capability of current tools such as encase v7, paraben f2 commander, FTK are discussed as well as their ability to investigate cases concerning desktop virtualization. The current state of desktop computing and current state of distributed computing have been discussed in this chapter. Finally, the chapter analyses the current forensic capabilities by making reviews of three journal articles. They are 'Hiding in a Virtual World', 'Live Digital Forensics in Virtual World' and 'Virtualization and Forensics' (McLaughlin, 2007).

1.3.3 Chapter Three

This chapter critically evaluates the proposed research methodologies for this thesis project. The chapter starts with a section that explains the importance of methodology in research. Problem statement and hypotheses is formed. The chapter covers vital methodology concepts such as ground theory, case study and a description of the key methods used in this study. Research methodologies and research methods proposed here are constructed in a way that will provide response to the research statement. The final research design is stipulated. Due to the fact that the area under study is so new, there is no suitable methodology and therefore the researcher will utilize action research together with scientific methodology (Avison et al., 1999). The null hypothesis will be tested and observed using the scientific observational method by implementing and running a problem based simulation of HVD and investigating it by applying the existing digital forensic procedures.

A Forensic Iterative Development Model (FIDM) that demonstrates the use of a scientific research process with action research is included in this chapter. Additionally, three case scenarios are developed in this section. The scenarios are designed to target three primary HVD challenges in private clouds. These scenarios are fictional. After the development of these scenarios, they are simulated. Current digital forensic procedures and tools are used to conduct a forensic investigation on these scenarios.

1.3.4 Chapter Four

The results for each of the simulated scenarios are presented in this chapter.

1.3.5 Chapter five

This chapter discusses the findings from the previous chapters stating the advantages and disadvantages of each digital forensic procedure used. The chapter will elaborate on the unforeseen scenarios. Thereafter, recommendations and best practices shall be incorporated in order to provide guidance to organizations using or planning to implement desktop virtualization technology, so their IT infrastructure can be planned accordingly. A FIDM feedback section will be considered in this chapter in order to suggest changes to the FIDM model and more specifically on the operational changes. The last section in this chapter shall be revisiting the problem statement and the hypothesis.

1.3.6 Chapter six

This chapter will provide a conclusion based on the research findings. The chapter summarizes every chapter's contents, outline the limitations of the study and offer suggestions for future research.

Chapter 2 - Literature Review

2. INTRODUCTION

The main research objective of chapter two is to explore the background knowledge and contextual relationship between Digital Forensics, Computer Virtualization and Cloud Computing. In order to achieve this objective, section 2.1 identifies the current state of the digital forensic field by initially outlining the history of the field, followed by the disciplines of digital forensics; which aims to outline existing specialized sub-fields of digital forensics, currently used investigation frameworks and relevant software tools commonly used in this field. In section 2.2 Current State of Desktop Computing covers the expedition between ordinary desktop computing and virtual desktops, also stating the technical aspects of virtual desktops. Section 2.3 Current State of Distributed Computing aims to review the shift in distributed computing that leads to discuss architectural, security and forensic aspects of cloud computing. Section 2.4 Current Forensic Capabilities, targets to review three types of literature and determine the depth of existing research in the areas of digital forensics, computer virtualization and cloud computing. Finally in section 2.5 chapter two is summarized and concluded.

2.1 THE CURRENT STATE OF DIGITAL FORENICS

Steadily growing technology has benefited users by allowing them to browse the internet on their mobile devices or control their workstations from remote locations, on the other hand it has also benefited criminal minds by allowing them to commit computer crimes using devices such as PDA's, mobile phones, digital cameras or ATM machines which we use in our day to day life. The importance of digital forensics has therefore grown over the years as law enforcement has demanded the gathering of digital evidence in this digital era. The digital forensic field has been developing over the last forty years and is known by various special units organized by police; such as the Computer Analysis and Response Team (CART) in the USA and UK, or the Computer Crime Team (CCT) in Norway. Simon (2010) states that in the 1970's digital forensic techniques were first used to recover data from highly fragmented database files unintentionally erased by careless research. In the following years (1980's) several software utilities were made available which had the ability to diagnose and remedy cases involving recovery of data. In

today's times one may argue and refer to this practice as data recovery. A universally understood definition is "a methodical series of techniques and procedures for gathering evidence, from computing equipment and various storage devices and digital media, that can be presented in a court of law in a coherent and meaningful format" (Wolfe, 2007, p. 3). It is methodically performed while maintaining a documented chain of evidence to solve a criminal case.

2.1.1 Disciplines of Digital Forensics

This section will introduce the three specialized fields recognized as the sub-fields of digital forensics; mobile device forensics, database forensics and network forensics. With the increased demand for mobility in computing, users are increasingly relying upon mobile devices and there are countless types of devices manufactured by numerous manufacturers and to add further complication, each mobile device manufacturer is likely to have their own proprietary technology and storage formats (Martin, 2008, p. 4). Mobile device forensics is relied upon to carry out critical security investigations to locate and extract data from mobile electronic devices, the methodologies employed to do so are similar to digital forensics. As part of the ongoing research in this field, new techniques and procedures to locate and extract evidence are regularly updated as new brands, models, and types of mobile device arise in the market. As well as this, Raghav & Saxena (2009) state, constant research is also conducted on challenges such as identifying hardware from Chinese made phones, recovery and preservation of data when a device is found in an uncommon state i.e. liquidated, password protected or scrapped.

Database forensics is the field of extracting artifacts from databases which largely depends on the metadata that contains key information about stored data and programs using the database (Olivier, 2009). A typical scenario where database forensic techniques are needed and used is where an e-commerce website is compromised where the hacker has managed to change pricing of advertised products by modifying related tables/columns, this leads to a significant loss for the owner. This is when a forensic examination will be undertaken, checking the metadata, audit logs or any other backend utility useful to find further leads. Database forensic investigation is preferably performed in live state as "the data ostensibly retrieved from the system may not be what is really on the system, but what the rootkit (or other malware) prefers to present" (Olivier, 2009, p. 116). Software tools such as ACL, Idea, logMiner and Arbutus are often found

necessary to analyze and document evidence in a database forensic investigation. However these tools are not officially forensic tools, they are simply used to assist a database forensic investigation. As the field of database forensics matures, reliable and precise tools are going to be necessary as database software utilities cannot be always be depended on in a forensic investigation (Wright, 2005).

Network forensics is the third specialized field of computer forensics, which is used to precisely capture and examine network traffic as a way of collecting artifacts for forensic investigations, hence it is also defined as the field of "sniffing, recording, and analysis of network" (Wiles & Reyes, 2007, p. 588). Network forensics is commonly used in investigations regarding policy violation, data breaches, tracing email spam, network intrusions, denial-of-service attacks and router attacks. In network forensics there are several methods used to investigate crimes using such attacks. Wiles & Reyes (2007) describe the different types of router attacks i.e. destructive bandwidth flooding, DoS attack, routing table poisoning attack and Hit-And-Run attack are the three types of router attacks which are investigated by collecting artifacts from analyzing log files and a series of built-in commands oriented towards the nodes and IP routes of a network. The authors also mention the increased difficulty of tracing connectionless nodes as compared to connection-oriented nodes is far complex. In (Casey, 2005) a network intrusion case is discussed where a continuous operating research laboratory was shut down for 7 days. The servers were hacked as the intruder managed to replace a standard version of telnet with a backdoor, neighboring computers in the network were further compromised as the intruder also managed to sniff their login details. As part of forensic procedures, a backdoor code was searched and identified from the audit logs. This code was used as a keyword to search and analyze for additional leads. In the end all the artifacts were used to reconstruct the crime in order to document the case for law enforcement. It is interesting to note the methods used to search, seize analyze and present the evidence in such cases. Although the depth of methods used in (Casey, 2005) and (Wiles & Reyes, 2007) were restricted to basic system audit logs and a keyword search, depending on the nature of the attack the components of the network involved in the attack, there are numerous other methods employed in network forensics to search, seize and analyze evidence. Such methods involve capturing live traffic of packets and analyzing them using deep packet inspection. A comparable method is discussed in Slay & Simon (2008), where drug dealers were using Skype to communicate with their colleagues. Law enforcement bodies

wanted to tap into the conversations and collect evidence, but the traditional wiretapping methods were not applicable because there are "no wires to tap, no call logs and no ability to tie a person to a specific geographic location" (Slay & Simon, p. 4). Alternate methods discussed involve; imaging suspects' machines, finding real time protocol packets (RTP) within the image, extracting a payload from RTP and recreating audio files containing the conversation. As easy and straight forward as this method may sound, in reality the method relies on the access to the suspects' machines and availability of RTP packets. What's more the results of this method are not comparable to results of traditional wiretapping as it doesn't have the ability to intercept VoIP conversations due to encryption.

The level of sophistication of digital crimes involving computer networks has advanced, resulting in the development of sub-disciplines of network forensics. With the most commonly known sub-disciplines of network forensics i.e. cyber forensic, wireless forensics, router and VoIP forensics, comes the addition of virtualization forensics to the discipline of network forensics. Yet there is no fixed definition of this discipline due to two reasons. Firstly, virtualization is widely used, in digital forensics it can be used to build a virtual forensic lab and practice forensic applications or solve a criminal case by reconstructing a malicious environment or, on the other hand it can be utilized as a disposable operating system to avoid forensic discovery or recovery of evidence by criminals (Shavers, 2008). Secondly, crimes involving virtualization are mostly discussed hypothetically as cases where virtual computing is exploited are yet to occur, although this situation is likely to change as up to 49 million PC's worldwide will be virtualized by the end of 2013, according to research conducted by (Gartner, 2009). Consequently, virtualized environments are listed among the recent challenges to digital forensics because of the "shifts in evidence location" described by Nikkel (2010) where less evidence is found on a client PC disk and is increasingly found on the external infrastructure, adding further complexity in locating and extracting evidence. Further challenges specific to external infrastructure (Cloud computing) and virtualized PC (Desktop virtualization) are discussed in section 2.3.4.

2.1.2 Investigation frameworks

Regardless of which sub-discipline is in practice, digital forensic procedures are sustained by following investigation frameworks. This section aims to provide a concise description of the

major investigative frameworks recognized. The series of procedures is often driven according to the forensic investigation model in use. The foundation of every framework consists of four elements i.e. preservation, collection, analysis and reporting, Ever since practitioners have been to solve the arising e-crimes, several research groups have formed to discuss a digital forensic approach and standardize its procedures. Current active research groups are the National Institute of Justice (NIJ), the Digital Forensic Research Workshop (DFRWS), the Computer Analysis and Response Team (CART), the Scientific Working Group on Digital Evidence (SWGDE), the Technical Working Group on Digital Evidence (TWGDE) and the American department of Justice (DOJ) (Agarwal et al., 2011). Each research group proposes personalized investigation methodologies based on the four elements e.g. As shown in Figure 2.1 below, DFRWS introduces two additional elements (identification and examination) to support investigations involving digital systems as well as computer networks, where "identification" is referred to as understanding the scope of the "investigation such as type of case, subjects involved, and systems involved" (Harrell, 2010); whereas "Examination" is the process of identifying evidence relevant to the case prior to analysis.

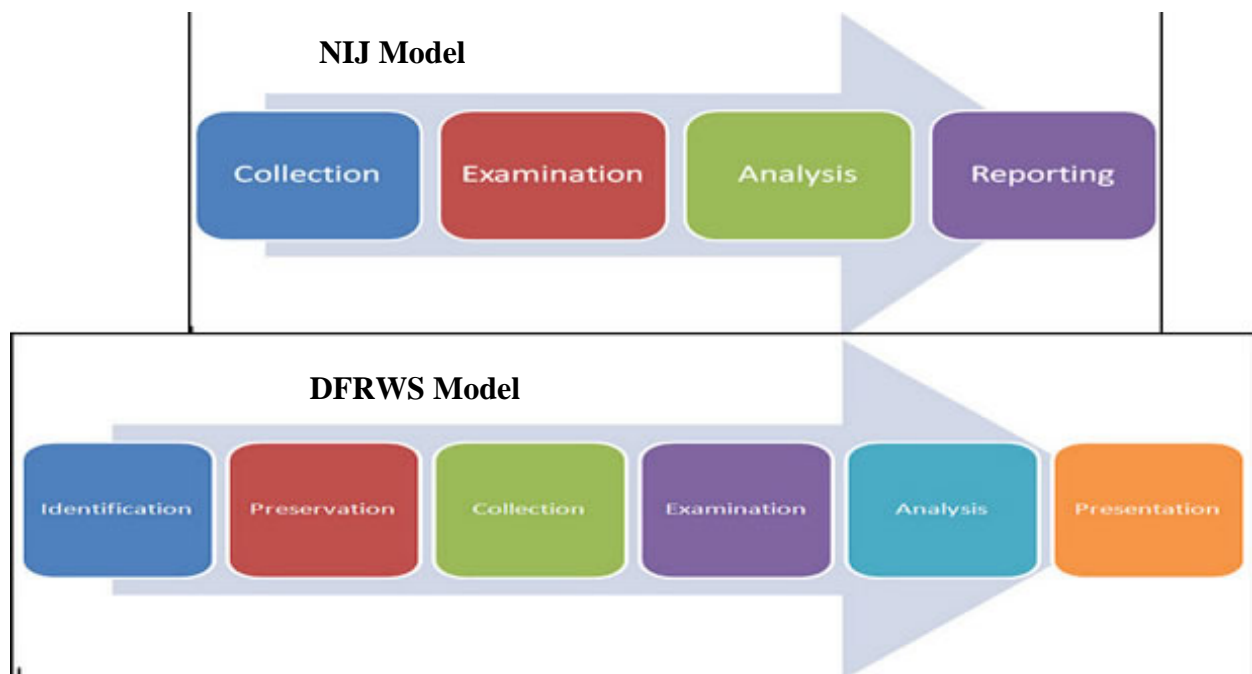


Figure 2.1: The NIJ vs. the DFRWS Model (Source: Harrell, 2010)

Figure 2.1 above only shows a visual comparison between the NIJ and DFRWS models, there are several other frameworks that could be compared in order to understand variations in them;

however, currently, digital forensic investigations do not always follow a particular framework consistently, mainly because there is no consistency in the type of e-crimes occurring. Instead, the type of framework is selected on a case by case basis, where the methodology could be revised and elements could be restructured to fit the type of case (Sanya-Isijola, 2009). On the other hand, the suitability of the existing frameworks for cloud forensics is unknown as sophisticated crimes in cloud computing are yet to occur. Despite of this, (Leong, 2006) has proposed a basic cloud forensic model illustrated below in figure 1.2.

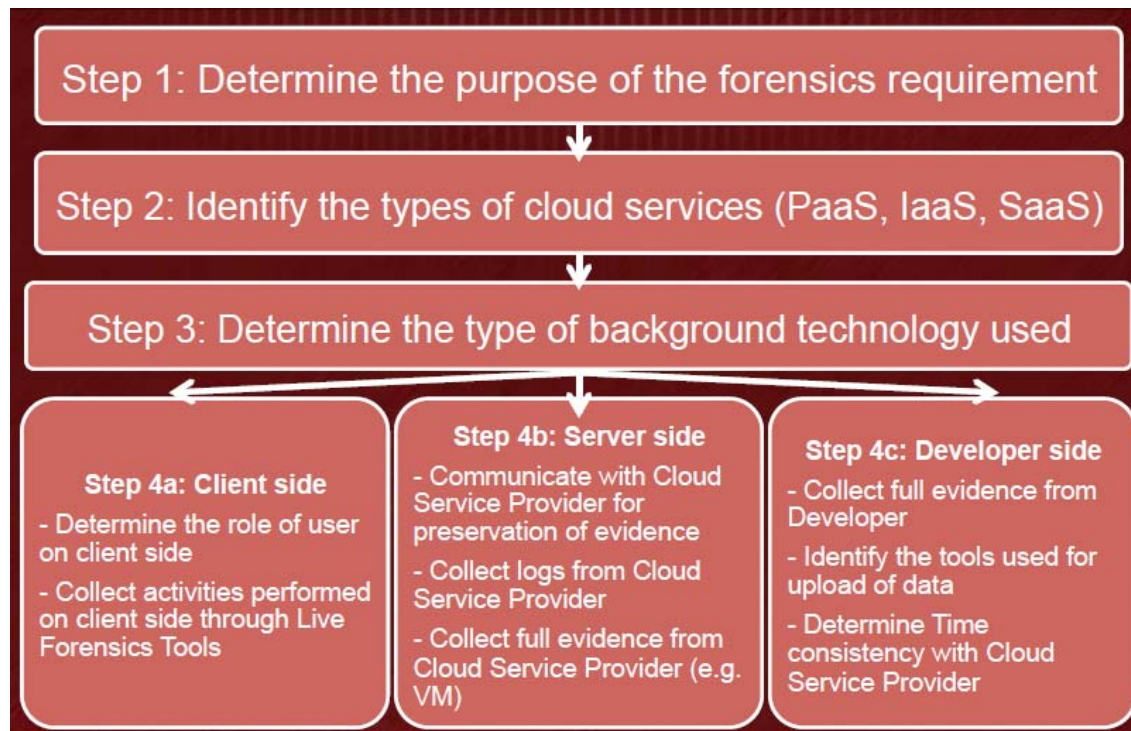


Figure 2.2: Cloud forensics - FORZA Model (Source: Leong, 2006)

It is interesting to note the framework in Figure 2.2 above has no resemblance to the typical digital forensic investigation frameworks, because the FORZA framework is intentionally made least technologist focused as often many digital forensics practitioners only follow technical procedures and overlook the actual purpose and core concept of the investigation. Leong (2006) states the framework is designed to break the technical barrier between investigators and legal practitioners so corresponding tasks during an investigation can be accomplished together using this technical-independent framework. Although the cloud computing focused framework provides a general flow of how a forensic practitioner should plan an investigation; it has not considered the different cloud deployments i.e. private, public, hybrid and community, as this

would largely impact the procedures for client, server and development side. Even though the FORZA model is basic, it is certainly a good start considering other research groups have no specific framework for cloud forensics.

2.1.3 Capability of Current Tools

Digital forensic tools are essential for a forensic practitioner as they are used to assist investigators in the course of acquiring, examining and analyzing evidence. The following section aims to discuss the capability of the three most widely used forensic tools; Encase v7, Paraben F2 and FTK in the context of cloud computing.

Digital forensic tools range over the sub-disciplines of the field e.g. mobile forensics or network forensics; however tools are mostly classified as hardware or software tools in a digital forensic toolkit. The range of hardware tools consists of drive erasers, bridges/write blockers and most importantly a workstation to copy and analyze data. within the software tools range, there are a large number of tools which have limited scope and focus on a sole function, such as password cracking, Email viewing or data integrity tools. There are also forensic tool suites which are virtually all-in-one solutions, by and large an enterprise type of applications. Among the many widely used commercially licensed multipurpose forensic applications are Encase, Forensic Tool Kit (FTK) and Paraben P2 Commander, which have the capability to capture and analyze digital evidence using distributed processing (Golden et al., ACM New York/2006). (Reilly et al., 2010) believe, digital forensic tools give a reasonable support for investigating conventional localized systems including the support virtualized environment, where virtualization on standalone systems is utilized by (Type 2) hypervisor applications like VMware workstation or Microsoft Virtual PC. The following quotes are captured from the release notes of selected forensic tools; to help understand the level of support provided for virtualized environment.

2.1.3.1 Encase v7

"EnCase has native ability to analyze VMware .vmdx data files and VMware snapshot files..... EnCase can interpret the data files that compose the physical and logical structure of the virtual hard drive, including unallocated space, and allow for quick and thorough analysis of VMware. EnCase Enterprise can also do live system analysis of running VMware instances with machines that have the servlet installed" (Encase forensic - Detailed product description, 2010, p. 5).

2.1.3.2 Paraben and FTK

"Added support for Virtual HD and VMware disk image. Supports disk images from the most popular forensic imaging software and virtual image formats...VMware disk images, Virtual PC Virtual HD disk image" (Paraben's P2 Commander 2.0 - Release notes).

The release notes of Encase and Paraben suggests the support for localized virtualized environment is acceptable. However there is still a lack of tool support for extracting and examining cloud deployment. For example, the ability to perform acquisitions on ESX servers and analyze artefacts in virtual infrastructure as Haletky (2008) mentions, "Today it is possible for AccessData's FTK and Encase tools to read virtual machine disk files (VMDK) for further forensic study, but how do you get this information off a VMware ESX server's VMFS in a forensically sound manner? Is the VMDK all you need to grab?" FTK, Encase and Paraben do not currently support acquisitions in such a manner or recognizes virtual file systems such as a Virtual Machine File System (VMFS), a file system used by VMware ESX servers.

2.2 CURRENT STATE OF DESKTOP COMPUTING

This section aims to identify the evolution of desktop computing from the past to present. As past, the development of personal computers is discussed and as present, client hosted desktop virtualization is explained. A personal computer (PC) in today's time is defined as an inexpensive, general purpose computer, directly intended for an end-user. "Whose size, capabilities, and original sales price make it useful for individuals" (Enderle, 2011), as oppose to the models known as time-sharing or batch processors operated by full time computer operators, which allowed many end-users to work on a large, expensive mainframe system at the same time. It is obvious to say that the definition of a computer has notably evolved over time, as the concept began to take shape in the period between 1970's and 80's where end-users started using PC's for automating calculations and general purposes such as typing letters. Within a short time personal computers were seen as an appliance, networked to an organization's mainframe and present in every departmental office. As their use became common, PC's were used in every small to medium sized company, "evolving into an indispensable appliance in almost every home in the developed world, no single technology has impacted more people than the personal computer" ("Personal Computer Timeline," 2000). In recent times, a computer or Laptop are

generally identified as PC, even though they differ from form factors, size, price and targeted purpose.

Ubiquitous computing, computers being everywhere has been on rise ever since. In these times of the increasing application of computers, desktop computing technology continues to progress as the demanded for mobility, manageability, versatility, legacy software support and most importantly cost have been the developing factors. In an attempt to develop the above factors, virtualization in computing was introduced. The term virtualization in computing is generally defined as "the creation of a virtual (rather than actual) version of something, such as a hardware platform, operating system, a storage device or network resources" (Turban et al., 2008, p. 27). Virtualization has been development commercially since 1964 by IBM, but it was first introduced as an end user product called "Virtual PC" in 1997 by a company called Connectix, later VMware later filed a patent for their techniques and went on developing the first virtual platform (*Timeline of virtualization development*, 2011). Since then, virtualization has evolved under sub-divisions such as; hardware, software and desktop. Hardware virtualization refers to a virtualized hardware platform that allows the software implementation of a real computer (virtual machine) on an operating system (Turban et al., 2008). Hardware virtualization also breaks down into different types, such as full/partial virtualization, paravirtualization and hardware-assisted virtualization. Among the different types; hardware-assisted virtualization, which imitates parts of a computer's hardware or the whole computer altogether (hypervisor) is most commonly used. The software virtualization concept is similar to hardware virtualization which allows a soft implementation of a physical machine, whereas software virtualization implements "a virtual layer or virtual hard drive space where applications can be installed. From this virtual space, applications can then be run as though they have been installed onto host OS" (Menken & Blokdijs, 2008, p. 24). Software virtualization's commonly known benefits are running applications without registry changes (portable apps, ThinApps) or running multiple versions of software on the same platform. Desktop virtualization is also understood as the separation of logical desktop from the physical machine. The concept of the virtualized desktop is a combination of hardware and software, hence the possibility of this concept is strongly dependent on the abilities of hardware and software virtualization. Besides hardware, software and desktop, there are various other types of virtualization such as network, data and storage. However the scope of this research is restricted to virtual desktops and the technologies directly

in relation to them. So the following section will continue to explain the current state of desktop computing in relation to desktop virtualization.

2.2.1 Virtual Desktops

As mentioned earlier desktop virtualization is the functioning representation of a physical desktop. According to (Intelligroup, 2009) the concept of virtual desktop was brought forward as the growth in computer users increased, introducing various management challenges such as software and hardware refresh cycles, power and space consumption, all together costing organizations financially. Desktop virtualization operates in two main modes, remote hosted and client hosted. The remote hosted virtual desktops (HVD), also known as virtual desktop infrastructure (VDI) "is an integrated solution of hardware, software and management tools to provide a replacement for standard desktop deployments" (Intelligroup, 2009, p. 3) i.e. a user's entire desktop is located on a centralized server, this mode of operation will be explored in section 2.3.3 with the reference to the cloud context.

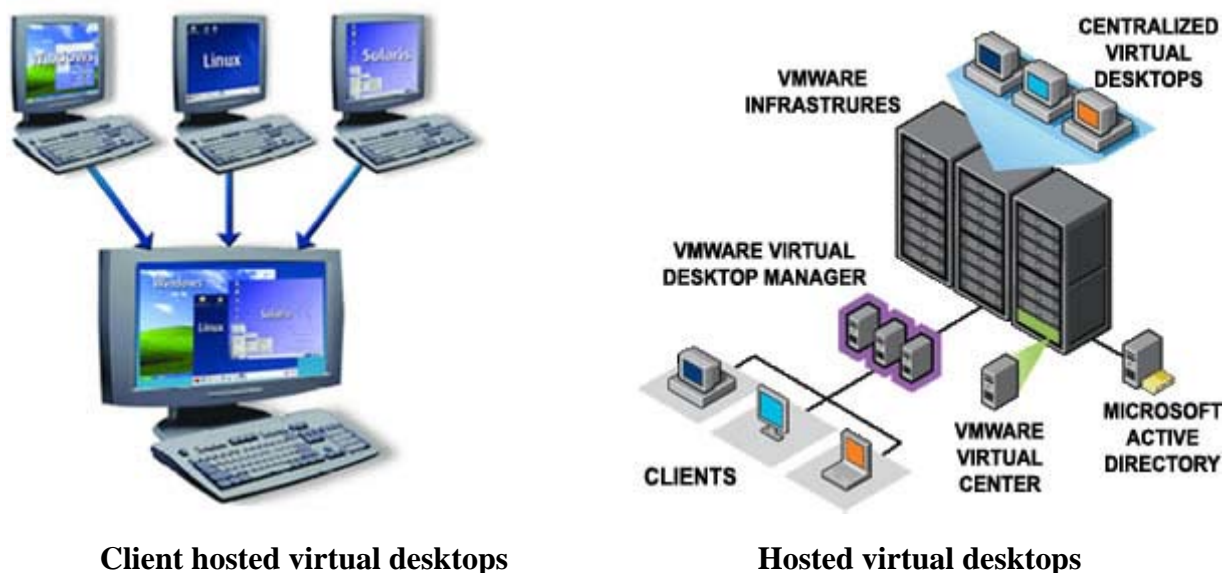


Figure 2.3: Two Modes of Operation for Virtual Desktops.

Client hosted desktop's on the other hand is the implementation of an end-user environment (Virtual desktop) hosted on a local client PC, as illustrated in Figure 2.3. In some cases the end-users interact with the virtual desktop as they would an ordinary application with the help of workstation programs (also known as hypervisor software) like VMware Workstation, Microsoft

Virtual PC, Java VirtualBox or Parallels Desktops for Mac. The most common uses of client hosted virtual desktops include, software testing for developers, studying the nature of malware infection, general demonstrations, running multiple OS side by side or parallel to the host OS (Windows on a Mac OS). The capabilities of the above workstation software also allows emulation of hard disk drives (virtual HDD or VHD), CD/DVD drives, memory (VMEM), switch/network adapters for bridging and segmenting multiple OS running simultaneously.

Between the two variants of desktop virtualization, the virtual machine (VM) is the common element that plays the biggest role in delivering the end-user environment. Whether the user is accessing the desktop remotely or locally, Windows, Linux or Mac OS, it is the virtual machine that executes the environment as if it were a physical machine (Smith & Nair, 2005). The implementation of a VM is possible as "developers add a software layer to a real machine to support the desired architecture, By doing so, a VM can circumvent real machine compatibility and hardware resource constraints" (Smith & Nair, p. 33). This software layer is also known as the function of virtual machine monitor (VMM) or a Hypervisor. Though remote hosted and client hosted virtual desktop both make use of a VM and VMM/Hypervisor, the arrangement of the abstraction layer differs. This can be understood from the illustration in Figure 2.4 below.

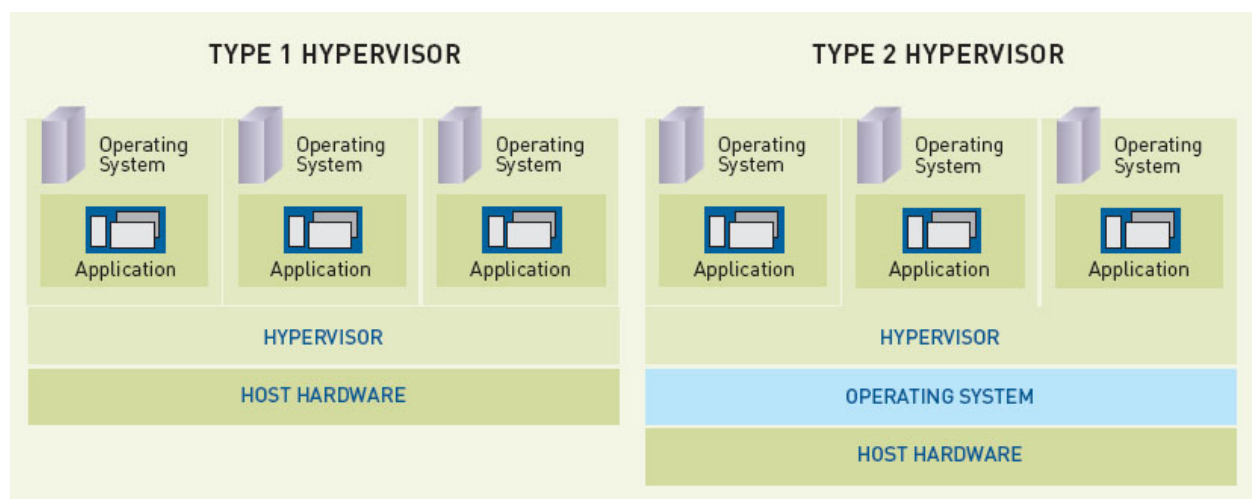


Figure 2.4: Type 1 vs. Type 2 Hypervisor (Source: Brown, 2009)

As shown in figure 2.4 the comparison between two types of hypervisors. The Type 2 hypervisor resides above the host machines hardware and software (OS) which explains why Type 2

hypervisor is used for client hosted virtual desktops, whereas the Type 1 hypervisor can run on hardware itself (bare metal) as it can form a layer of abstraction as well as host VM's at the same time, hence the Type 2 hypervisor is used in hosted virtual desktop infrastructures.

2.3 CURRENT STATE OF DISTRIBUTED COMPUTING (CLOUD CONTEXT)

This section aims to review the current state of distributed computing, in regard to cloud computing so the various methods of cloud deployment, service models and security risks associated with cloud computing can be understood. In a computing environment, task and data intensive applications are managed by properties such as load sharing, fault tolerance, node expandability and resource sharing. These properties were part of distributed computing design. A distributed computing environment is understood to be a system utilizing multiple computers to host multiple software components and function as a single system (IBM, 2009). The distributed system consists of two main operational models. The client/server model; where users and service providers are divided, the user of a service is called a client and the service providers are called servers. The client/server model was later developed into a multi-tier architecture (three tier client/server model) which logically divided presentation, application processing and data management into separate processes. The principal is illustrated in Figure 2.5 below.

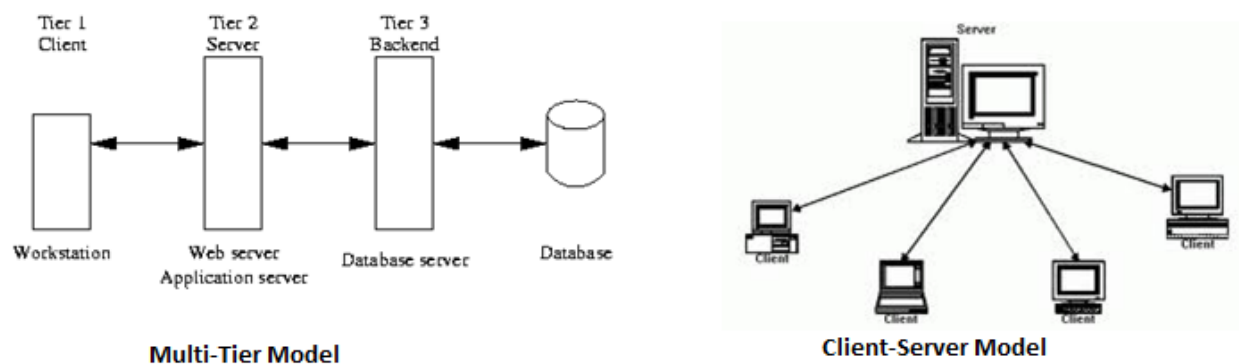


Figure 2.5: Multi-Tier and Client Server Model

In the recent times distributed computing has been gradually moving towards cloud computing. According to (Junjie et al., 2009, p. 93) "Cloud computing is a complete new technique put forward from industry circle, it is the development of parallel computing, distributed computing and grid computing". The concept of cloud computing is very similar to distributed computing in the way it delivers services to clients. However the involvement of virtualization on the cloud

platform permits computation and provision of services over the network (LAN/WAN); offering further benefits, such as infrastructure scalability allowing organizations to accommodate their growing needs for IT resources, virtual delivery allowing greater availability of resources and the ability to allocate, relocate and deploy IT resources dynamically, which all together makes cost savings for an organization.

2.3.1 Service Models

According to the National Institute of Standards Technology (NIST) report on cloud computing (Mell & Grance, 2011), the platform is operational under three services models; software as a service (SaaS), platform as a service (PaaS) and infrastructure as a services (IaaS) . In the (SaaS) model, applications are hosted in the cloud, where service providers are responsible for maintaining and providing access to its subscribers. Users/subscribers on the other end don't need to install, manage or buy relevant hardware to run the applications "since service provided to client is the applications running on the cloud computing infrastructure provided by the service providers" (Junjie et al., 2009, p. 23), which can be accessed using various devices through a thin client interface like web browsers. Typically business oriented applications such as (CRM), (ERP) or (HRM) are hosted on a cloud SaaS model. The PaaS model in clouds is used by the subscribers to complete the entire life cycle (develop, test, deploy) of software development including "application services such as team collaboration, web service integration and marshaling, database integration, security, scalability, storage, persistence, state management, application versioning, application instrumentation and developer community facilitation" (Roebuck, 2011, p. 155). All the services are provisioned as an integrated solution over the web without the need to build platform locally. Paas solutions provided by companies include Microsoft's -Azure or Google's - App Engine.

The cloud IaaS model allows subscribers to lease the entire computer infrastructure to deploy applications, or platforms to run operating systems. As the whole infrastructure is leased, the user has control over processing power, network, storage and any other basic computing resources (Junjie et al.). This type of service model is used by organizations to reduce the costs involved with purchasing new servers and networking equipment to build a data center. The most popular use of the IaaS model is hosting virtualized servers which can be used for web hosting, email exchange and the emerging virtual desktop technology, also called hosted virtual

desktop (HVD). The most popular IaaS cloud providers of public clouds (as explained in section 2.3.2) are Amazon's Elastic Compute Cloud (EC2), VMware and tuClouds.

2.3.2 Cloud Deployment Methods

Proceeding with an understanding of the different cloud based service models, from the deployment point of view; software-as-a-service (SaaS), platform-as-a-service (PaaS) and infrastructure-as-a-services (IaaS) are deployed through four variations; private (internal) cloud, public cloud, community cloud and hybrid cloud. The private cloud infrastructure is exclusively operated and managed by an organization, but sometimes the management of the infrastructure can be leased to a third party and as a result, it may exist on or off premises (Mell & Grance, 2011). Having a in-house cloud allows an organization to utilize the highest level of virtualization in order to provide services internally with a higher level of management control than the other deployment methods. This also gives the organization an opportunity to isolate its data even though it is hosted in the cloud. The Public cloud infrastructure is solely owned by organizations which provide cloud services to the general public or other organizations. Public cloud infrastructure is popular among small to medium sized companies wanting to offload their IT management by having their IT services hosted instead of managing them locally (Bakshi, 2011). Community cloud infrastructure is jointly formed by several organizations that share common interests and "concerns (e.g., mission, security requirements, policy, and compliance considerations)" (2011, p. 3). As with private clouds, community infrastructure can be managed by the owners or leased to a third party. A realistic example of a community cloud could be government agencies such as IRD, Police or Customs sharing, a cloud to maintain records of all the citizens. NIST definition of a hybrid cloud infrastructure (2011, p. 3) is "a composition of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds)". For example, organization may use the hybrid deployment method to sort their applications from most frequently used to least frequently used, and the least frequently used applications can be stored on a public cloud whereas the frequently used applications can be stored on the private cloud.



Figure 2.6: Visual Model of Cloud Computing (Source: NIST, 2011)

Having mentioned about the four major methods of deploying cloud services, Figure 2.6 above provides a snapshot of cloud computing. In addition to cloud computing, recently developed virtual private clouds and inter-clouds are two emerging deployment models to be introduced in the near future, according to (Bakshi, 2011). A virtual private cloud (VPC) is an extension private cloud, allowing organizations to form a private isolated sector on a public cloud. The concept of VPC has emerged to extend the trust boundaries (SLA and Compliance) of a public cloud. An inter-cloud decouples organizations from cloud service providers, giving the flexibility to access resources on demand without any agreements with the providers. The emerging inter-cloud deployment would be an "enhancement and extension of the Internet itself" (Bakshi, 2011, p. 2).

Upon reviewing the different deployment techniques and cloud service models it is understood that the flexibility allowed by virtualization in cloud computing permits various cloud models to host a variety of services. In sense, one type of service is not restricted to a specific type of cloud deployment method, we can see an entire email exchange hosted internationally on a public cloud, where clients operate it as if it is locally available. As the primary focus of this

thesis is on the hosted virtual desktop (HVD) in private clouds, the research will continue with a technical review of HVD architecture and involving the technologies involved in the following section.

2.3.3 Virtual Desktops in Clouds

The discussion from the literature's reviewed until now has provided an overview of the current state of desktop computing and its direction in distributed computing as it shifts towards clouds. The term "Virtual Desktop" is a highly generalized term in the field of computing as seen in previous sections, it can refer to a VM hosted on a user's workstation or a VM acting as a virtual desktop hosted in different cloud types. The following section will discuss the architecture of virtual desktop deployed in clouds.

A virtual desktop is known as a hosted virtual desktop (HVD) or virtual desktop infrastructure (VDI) when referring to cloud computing. "VDI was VMware's original acronym for the hosting of desktop operating systems on their virtualization platform, but it is now acknowledged as an industry-wide term" (Buckle, 2009). The architecture of HVD differs from the client hosted virtual desktop in that its virtual existence resides remotely. While HVD integrates hardware and software as part of its infrastructure, it also requires five mandatory components to function fully. The five components are; the end user, connection broker, virtualization platform, directory service and management tools. While these are the core set of components common to every VDI solution regardless of the vendor, there may be additional components as "the components used in a VDI solution are dependent on the functional requirements that it is intended to fulfill" (Larson & Carbon, 2009). Figure 2.7 illustrates the arrangement of VDI components.

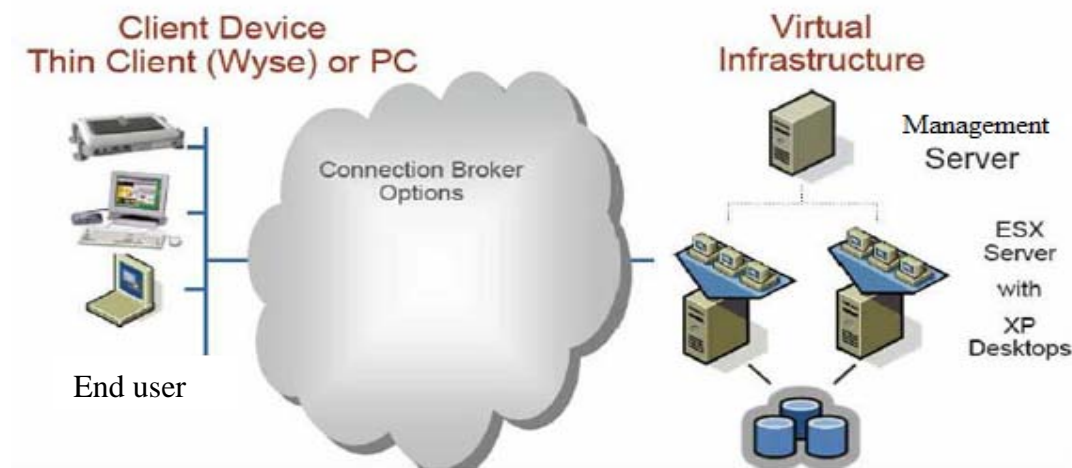


Figure 2.7: Basic Architecture of VDI

To be able to access anything from anywhere end users within the infrastructures are able to access their virtual desktops using devices like personal computers, tablets and thin clients. The type of client access device would depend on usage (Intelligroup, 2009) e.g. a regular user of HVD is more likely to use a regular PC or thin client for better performance, whereas mobile devices like tablets, will mostly be used for emergency or quick access to a user's personal space. Every client access device displays the users virtual desktop using a display protocol. There is a range of connection protocols used in HVD infrastructures, mostly dependent on the vendor of the HVD solution used e.g. VMware uses Teradici PC-over-IP (PCoIP), Citrix HDX 3D and Microsoft's Windows Remote Desktop Protocol (RDP) (Vanover, 2010).

The connection broker is a software component of the HVD infrastructure installed on a server operating system and is used to maintain the connection between the user and the designated virtualized desktop. The connection broker is most often used when there are more than 100 managed users. While the connection between users and their virtualized desktop is maintained, the connection broker also validates users' and permissions, manages users access to multiple VM pools, reassigns users to designated VMs in case of connection loss and monitors the user status based on usage (active/inactive).

The virtualization platform as a VDI component is a Type 1 hypervisor which also hosts virtual machines containing client operating systems. Every VM created within the VDI is held on a virtualization platform in order to function properly. The hypervisor embedded within the virtualization platform also allows resource sharing off a single hardware platform so multiple VM's can share CPU, memory, storage and network bandwidth as mentioned in (Bernard, 2009). The most well known virtualization platforms include VMware ESX, Microsoft Hyper-V and Citrix Xen Server. In addition to the platforms, a cluster file system is used to store disk images and snapshots formed by a functional VM. A cluster file system is often proprietary and heavily encrypted.

An active directory service is an essential component of a VDI's architecture. The function of Authenticating and authorizing users and computers in a traditional network is no different in VDI. The use of a group policy object (GPO) in a VDI solution is considered best practice according to (Maltz, 2010) because there is no other way to apply policies to Windows users of different levels e.g. Restrict access to the control panel for lower level users and grant full access to higher level users. The author also refers to the dynamic user environment in VDI where roaming profiles can be useful for users to keep application settings consistent regardless of the environment's state.

The management platform of VDI is considered as the most vital component of the architecture. Its primarily used to quickly and efficiently provide VMs according to the infrastructure's demand. The provisioning of VM can be automated using VM templates or manually setup by loading client OS. Depending on the implementation of VDI solution and the solution provider (Vendor), the management platform can also include monitoring capabilities such as monitoring user activity to observe applications in use or resource monitoring to warn about any imminent bottlenecks. Popular management platforms include VMware VirtualCenter or vSphere, Microsoft System Center and Leostream VDI management.

The architectural components discussed above are the basic requirement to form VDI or HVD in any cloud deployment method. All the related products i.e. display protocols, virtualization and management platforms are normally packaged under one VDI/HVD solution. VMware's View, Microsoft's RDS and Citrix XenDesktop are the main products and providers of VDI solutions in cloud computing.

2.3.4 Security and Forensic Challenges

As Gartner (2009) research suggests "the worldwide hosted virtual desktop (HVD) market will accelerate through 2013 to reach 49 million units, up from more than 500,000 units in 2009", and further estimates "approximately 15 percent of current worldwide traditional professional desktop PCs will migrate to HVDs by 2014, equal to about 66 million connected devices." There is no doubt HVD implementations will raise as organizations discover an alternative computing experience which is cost effective and consolidates and reduces major costs related to maintenance and support. On the other hand, with the new paradigm shift in distributed computing, cloud computing has also introduced various security concerns in relation to protecting data and the overall security infrastructure of the enterprise. This section aims to highlight the common security concerns of cloud computing and further discusses its implications on digital forensics.

According to Sabahi (2011) security is the most controversial topic in the field of cloud computing. The author outlines three major kinds of security issues in regards to cloud computing infrastructure. Data location is considered as the first concern because of the lack of control clients have regarding the location of their hosted data. Data located in clouds does is not always located on a single server as it can be distributed between many data centers, this could lead towards legal issues such as privacy, as the data protection laws for every country vary. For example, a client's desktop is being hosted in a country with no data protection laws, this could potentially mean the desktop could be monitored for any given reason. The result of improper custody of data can prevent a digital forensic investigation as it results in multi-jurisdiction issues, Taylor et al., (2010) state that if the dispersal of data within the cloud resides in a country where privacy laws are non-existent or not steadily enforced, this could prevent to forming of chain of custody for the data i.e. seizing and preserving evidence for analysis.

Compliance generally refers to stating or meeting rules or standards, but in cloud computing compliance is also refers to security concern because clients could face serious legal compliance challenges if they don't make sure the accepted service agreements and terms of service are comprehensive and balanced enough to ensure necessary regulatory compliance (Matsuura, 2011). For example, a client decides to unsubscribe to services provided by a cloud service provider (CSP), and move to another CSP. While the SLA with the first provider was only in

effect while the client was currently subscribed, it could mean the first provider could retain statistical data like browsing habits etc and trade it with a third party.

From the digital forensic standpoint, compliancy/SLA presents similar digital forensic challenges to custody of data. However (Taylor et al., 2010, p. 305) mention, in a instance where gathering information such as system logs is required "a public cloud (internet based) managed by another organization that provides cloud computing services is likely to be more difficult to investigate than a private cloud". CSP clients lack awareness about cybercrime and the added effects of forensic investigation in cloud computing, which is why conditions related to forensic investigations are not included in SLA's by most of the CSP. Ruan & Carthy (2011) state that there are instances where user data can be compromised without the user knowing anything about it. The uncertainty is caused because CSP "either do not know how to investigate cloud crimes themselves or the methods and techniques they are using are likely to be problematic in the highly complex and dynamic multi-jurisdiction and multi-tenancy cloud environment" (Ruan & Carthy, 2011, p. 14).

Multiple tenancy in cloud computing refers to multiple virtual machine co-located on the same physical server, where underlying physical resources are shared transparently between the multiple virtual machines, as shown in Figure 2.8. Although, typically, this has the advantage of consolidating many physical machines, it also introduces security risks where, a "malicious user having control of a VM can try to gain control over other VM's resources or utilize all system resources leading to denial of resource attack over other VM users" (Jasti et al., 2010). The attacker could gain further control by attacking the hypervisor file, allowing access to other users' data on the same physical server.

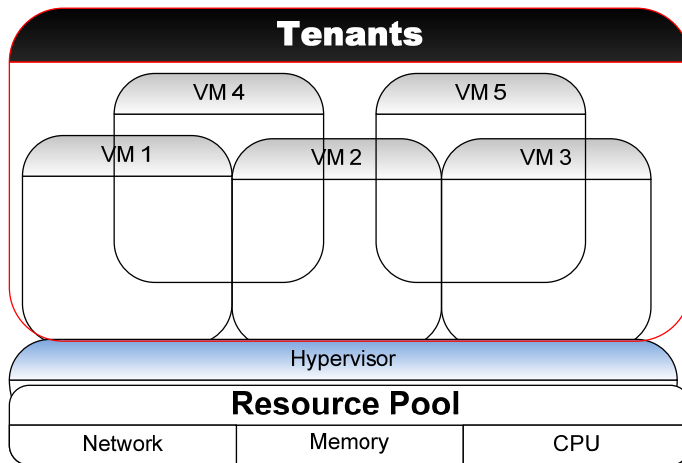


Figure 2.8: Multi-Tenancing

Data segregation is a particular security concern in cloud computing. Virtualization in cloud computing permits the hosting of several instances of user environments on the same physical machine, whereby one customer's data is stored alongside another customer's data and having no assurance the data is separated. This issue is also known as data leakage prevention (DLP) (Sabahi, 2011). Multi-tenancy is also regarded as a digital forensic challenge and according to Broadhurst (2006) it has been identified as the top legal concern among digital forensic experts. The challenge forensic experts of CSP and law enforcement authorities face are due to the lack of provisioning and de-provision technologies which segregate forensic data and system level audit logs, while maintaining the confidentiality of "other tenants sharing the same infrastructure and ensure the admissibility of the evidence" (Ruan & Carthy, 2011, p. 11)

2.4 CURRENT FORENSIC CAPABILITIES

The previous section discussed the three types of security issues that are of universal concern in universally in concern of cloud computing, no matter what type of service or deployment method. How the outlined threats impact digital forensic investigation was also discussed. As the focus of this study is to investigate the depth of digital forensic procedures in the context of HVD infrastructure, the review of security threats and challenges shows that hosted virtual desktops in relation to digital forensics is a fairly new topic as yet rarely studied. Hence a aide of review of any currently available forensic procedures on HVD architecture may not yet be possible. However, similar studies on components that make up the HVD architecture can be sourced and will be reviewed in section 2.4. The purpose of reviewing similar studies is to

determine the depth of the existing research by finding the strengths and weaknesses of currently used digital forensic methods and techniques to investigate crimes in virtual desktop environment, and further identify any gaps and inconsistencies whether they are directly or indirectly related.

2.4.1 Hiding in a Virtual World

Bares (2009) believes unconventionally installed operating systems such as virtual machines or operating systems installed on a removable disk are possible ways to hide in a virtual world. Although unconventionally installed operating systems are one of the most widely used recent innovations among developers and network administrators, it can also be used to mask user activities. This paper presents how these innovations can be used to hide users' illegal activities from forensic tools and how one can detect their traces using forensic tools and other tracking methods. Bares (2009) claims this will be beneficial to law enforcement agencies and industry experts that deal with virtual environments or unconventionally installed operating systems so they can be prepared to overcome any related security incidents.

(Bares, 2009) conducted experiments for a range of unconventionally installed operating systems i.e. bootable external storage, removable media. However the type that is of interest in this research is a Virtual Machine. (Bares, 2009, p. 276) experiment on a virtual machine as an unconventional operating system was conducted by installing four testing elements, email communication, IM traffic, word file and web traffic, on a virtual machine. The sequence of tests is shown below in Figure 2.9.

Test 1	Test 2	Test 3	Test 4
Email	Web Traffic	IM	Web Traffic
IM	Word	Email	Word
Word	IM	Web Traffic	Email
Web Traffic	Email	Word	IM

Figure 2.9: Order of tests (Source: Bares, 2009, p. 279)

The researcher recorded unique testing element identifiers, while every test was acquired as a separate image the FTK search function was run against the images using the unique identifiers.

In terms of evidence gathering it was interesting to notice the researcher mentioned detecting the file paths as the second metric e.g. files moved locally, files downloaded off the internet, as this is not a very commonly used metric. During the experiment, (Bares, 2009, p. 280) had unexpectedly discovered that a second run of tests needed to be done as the findings from the first round showed that data was easily recovered. This was due to the fact that virtual machines were shutdown improperly, leaving virtual memory files which also kept the recoverable data intact. Figure 2.10 below of the first round shows the elements discovered (email communication, IM traffic, word file and web traffic). The results state that there is no consistency between the numbers of files found, whether it is IM communication or Web traffic while the four tests were run. For instance, Test 1 for email shows only a 44% hit rate while Test 4 shows 26%.

	256 Test 1	256 Test 2	256 Test 3	256 Test 4
Email	18 / 8	169 / 50	128 / 34	128 / 34
IM	120 / 42	72 / 8	89 / 12	89 / 13
Word	293 / 12	521 / 18	602 / 21	607 / 21
Web Traffic	1260 / 86	913 / 119	1905 / 123	1886 / 119
Graphic Files	272	468	689	689
Driver Address	YES	YES	YES	YES
Save to Desktop	B D N Z	D N Z	D N Z	D N Z
Save to USB	B N	NONE	NONE	NONE
Load File Path	B Z	B Z	B Z	B Z
VMEM	YES	NO	NO	NO
Account Name	NO	YES	YES	YES

Legend: Numbers of Hits / Number of Files B = BMP, D = DOC, N = Driver, Z = ZIP, YES DEL = Found as a deleted file

Figure 2.10: Results of Set 1 (Source: Bares, 2009, p. 281)

A similar regularity of search hits was discovered in Set 2 as results show (Figure 2.11), however, a smaller number of files were found in the second round was run if compared with the first round. The results of both rounds proved that discovery of any activities or traces to the activities heavily depend on the state of the virtual machine seized.

	256 Test 1	256 Test 2	256 Test 3	256 Test 4
Email	0 / 0	10 / 52	344 / 51	0 / 0
IM	38 / 10	124 / 41	374 / 60	2 / 2
Word	121 / 12	22 / 4	20 / 2	30 / 2
Web Traffic	3464 / 155	1262 / 127	1677 / 37	1328 / 111
Graphic Files	615	427	406	434
Driver Address	YES	YES	NO	YES
Save to Desktop	D Z	D Z	Z	NONE
Save to USB	N	NONE	NONE	NONE
Load File Path	Z	B Z	B Z	NONE
VMEM	NO	NO	NO	NO
Account Name	YES	NO	NO	YES

Figure 2.11: Results of set 2 (Source: Bares, 2009, p. 282)

Furthermore, the researcher also mentions "The biggest factor in determining how many and what files are recoverable is whether the VM was shut down" (Bares, 2009, p. 282). The overall conclusion also suggests that user data can be recovered or traced from unconventionally installed operating systems, however, the amount of data may vary depending on several factors. The researchers overall conclusion suggests, conventional techniques "should be tried first and then the steps taken in the experiments should be conducted prior to seizing the suspect's computer to help build a stronger case if the later examination bears little or no evidence" (Bares, 2009, p. 283).

2.4.2 Live Digital Forensics in the Virtual World

The paper written by Wang (2010) discusses the booming world of virtualization technology with the popularity of Virtual Machines increasing as end users utilize it daily for work. With it, the use of virtual machines for malicious purposes is also growing (Zhang et al., 2010, p. 328). Digital forensics has caught the attention as a result of the state of this technology. This issue is addressed by demonstrating a method of collecting data from a physical machine containing an implementation of a virtual machine. According to Wang (2010) performing forensic procedures using forensic tools towards acquiring disk images and memory dumps from a physical machine is different to performing forensic procedures on a virtual machine. On the other hand, a use of virtual machines as forensic tool is also discussed, giving investigators the ability to investigate evidence in a live state.

(Bares, 2009) discusses the use of virtual machine for masking user activities, (Zhang et al., 2010) particularly focuses on the entity of virtual machine's from the standpoint of three major virtual machine applications i.e. VMware Workstation, Sun VirtualBox and Windows Virtual PC. The point of reviewing (Zhang et al., 2010) is because the methods that exist to investigate a virtual machine or use a virtual machine as a forensic tool can both be applied to this thesis, although this thesis looks into investigating an infrastructure consisting virtual machines, the straightforward techniques used in (Zhang et al., 2010) can be used in scenario's where virtual machine are found segregated.

As a stated method in (Zhang et al., 2010, p. 328) a virtual machine can also be used as a forensic tool by mounting a forensically acquired image and further booting the image using a virtual machine application. "By this way, we generate a "live" and "restored" subject system, and then we can explore evidence more easily and observe application software's behavior in real time" (Zhang et al., 2010, p. 330). This method can be used if the infrastructure of the hosted virtual desktop can be acquired as a whole and emulated in a virtual application, although this is not as straightforward as it sounds. Concerns such as tenant privacy, technical complications and incompatibility may threaten the feasibility of using this method in a hosted virtual desktop infrastructure. In order to investigate a physical machine containing a virtual machine remainders, Wang (2010) suggests the use of static forensics for the first attempt as opposed to live forensics, i.e. powering down the physical machine and cloning the targeted disk. However if the disk sizes are too large and forensic investigators already know the area of exploration, the most effective and feasible way is to acquire the necessary files only (Zhang et al., 2010). Figure 2.12 outlines the files generated by virtual machine applications and Figure 2.13 explains the purpose of each file type for VMware VM application.

Virtual Machine Applications	Virtual Machine Information	
	Virtual Machine Files	File Locations in a Windows 7 host system
VMware Workstation	.vmdk, .log, .vmem, .vmsn, .vmx, .vmxs, .nvram, .vmtm, .vmxf	1. User defined virtual machine folder (Like "D:\My Virtual Machine\Windows XP"). 2. \Users\ "User name" \VMware 3. \ProgramData\VMware
Sun VirtualBox	.vdi, .sav, .log, .xml	1. User defined virtual machine folder. 2. \VirtualBox 3. \VirtualBox\Machines\ "VM name" 4. \VirtualBox\Machines\ "VM name" \Snapshots 5. \VirtualBox\Machines\ "VM name" \Logs
Windows Virtual PC	.vhd, .vsv, .vmc, .vmcx, .xml	1. User defined virtual machine folder. 2. \Users\ "User name" \AppData\Local\Microsoft\Windows Virtual PC 3. \Users\ "User name" \AppData\Local\Microsoft\Windows Virtual PC \Virtual machine

Figure 2.12: Virtual Machine Applications and associated files (Source: Zhang et al., 2010, p. 329)

VMware Virtual Machine Files	Meanings
.vmdk	Virtual hard disk of the virtual guest operating system, may be dynamic or fixed sizes.
.log	The virtual machine's log file.
.vmem	A virtual machine's memory file which only exists when the VM is running or a snapshot has created.
.vmsn	VMware snapshot file which stores the state of the virtual machine when the snapshot is created.
.vmx	A text file contains hardware and operating system configurations of the virtual machine.
.vmxs	Metadata of the snapshot.
.vmss	Suspended state file, storing the state of a suspended virtual machine.
.nvram	The virtual machine's BIOS information.
.vmtm	Team configuration data file.
.vmxf	Remaining file while a virtual machine is removed from a team.

Figure 2.13: Files associated with VMware VM

Although these are the files generated by a virtual machine application while a VM is in function, it is worth noting a hosted virtual machine will also generate associated files but it is yet to be discover if they are any different to the workstation application and if not are there any additional files for a hosted infrastructure. In addition to this, (Zhang et al., 2010) states the difficulties in forensically acquiring large disk sizes or storage systems like RAID, SANs or RAID and hence suggests the live forensics method. This statement is particularly important to note as hosted virtual desktop are commonly implemented on such storage systems.

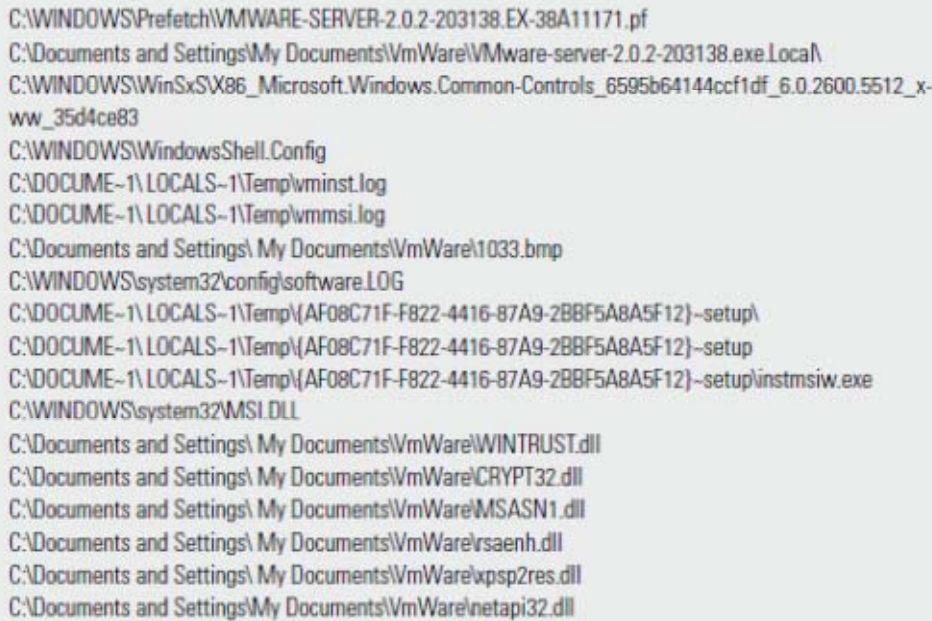
The author concludes the paper by suggesting the use of live forensics and static (traditional forensics) techniques should be balanced for efficiency, data quantity and usefulness because the traditional methods and tools will not be very efficient as analysis of the collected evidence will start from a binary level (Zhang et al., 2010).

2.4.3 Virtualization and Forensics

In (Barrett & Kipper, 2010) major virtualization technologies are discussed, And end user technologies such as server, desktop and portable virtualization are covered with solutions from vendors like Microsoft, VMware and Citrix. The book is dedicated to providing a complete understanding of virtualization technology and its forensic challenges and how to overcome them. (Barrett & Kipper, 2010) believe security implementation is often put into practice in a reactive manner, waiting for the problem to occur so an action can be taken, and hence security implementation often turns out to be expensive. Likewise, experts can perform procedures on physical machines in full confidence because it is more likely to be a straightforward task for them. However if a system is fully virtualized, the confidence may not be at the same level. The authors intention is to educate individuals or forensic experts in the proactive methods of investigating and analyzing virtual applications, in case of any incidents, allowing them to "quickly perform the forensics and minimize the damage to your systems" (Barrett & Kipper, 2010, p. 5). The first four chapters of the book discuss each type of virtualization technology individually, describing its uses, vendor offerings and security challenges.

The area that is of interest to this thesis is related to virtualized distributed computing, specifically hosted virtual desktops but establishing a direct link is not straightforward because similar studies reviewed so far are all based on virtual machines with very little or no relation to distributed computing. But the use of various methods to acquire, analyze evidence based in virtual machines is relevant. In this same way, sections from chapters in Barrett and Kipper's book explore the relation between digital forensics and virtualization. In chapter five - Investigating Dead Virtual Environments the authors main concern is regarded to the inability to examine a image containing virtual applications using traditional methods (Barrett & Kipper, 2010).

The authors' aim of the experiment is to investigate the presence of virtual machines. The suggested method to investigate virtual environment is to monitor notable installation files, registry entries, artifacts and remnants which apply only to individual virtual environments. Figure 2.14 shows the locations of files created by a VMware server which are also monitored for changes.



```
C:\WINDOWS\Prefetch\VMWARE-SERVER-2.0.2-203138.EX-38A11171.pf
C:\Documents and Settings\My Documents\VmWare\VMware-server-2.0.2-203138.exe.Loca\
C:\WINDOWS\WinSxS\X86_Microsoft.Windows.Common-Controls_6595b64144ccf1df_6.0.2600.5512_x-
ww_35d4ce83
C:\WINDOWS\WindowsShell.Config
C:\DOCUME~1\LOCALS~1\Temp\vmnst.log
C:\DOCUME~1\LOCALS~1\Temp\vmmsi.log
C:\Documents and Settings\My Documents\VmWare\1033.bmp
C:\WINDOWS\system32\config\software.LOG
C:\DOCUME~1\LOCALS~1\Temp\{AF08C71F-F822-4416-87A9-2BBF5A8A5F12}~setup\
C:\DOCUME~1\LOCALS~1\Temp\{AF08C71F-F822-4416-87A9-2BBF5A8A5F12}~setup
C:\DOCUME~1\LOCALS~1\Temp\{AF08C71F-F822-4416-87A9-2BBF5A8A5F12}~setup\instmsiw.exe
C:\WINDOWS\system32\MSI.DLL
C:\Documents and Settings\My Documents\VmWare\WINTRUST.dll
C:\Documents and Settings\My Documents\VmWare\CRYPT32.dll
C:\Documents and Settings\My Documents\VmWare\MSASN1.dll
C:\Documents and Settings\My Documents\VmWare\rsaenh.dll
C:\Documents and Settings\My Documents\VmWare\xp2res.dll
C:\Documents and Settings\My Documents\VmWare\netapi32.dll
```

Figure 2.14: VMware Server files (Source: Barrett & Kipper, 2010, p. 85)

As figure 2.14 illustrates, the files monitored for changes, it is yet to be seen how the files are being monitored for changes and also the purpose of each file so artifacts can be extracted.

In order to determine the registry keys in effect after the use of virtual applications, (Barrett & Kipper, 2010) make use of the virtual application to boot Knoppix as a virtual machine and, with the help of RegRipper, registry hives (on the host machine) are examined. As a result, it was noticed that "file associations maintained in the registry will indicate which program will be started based upon a specific file being selected" (Barrett & Kipper, 2010, p. 101). Some of the many registry keys that indicate the presence of a virtual machine on the host system are illustrated in Figure 2.15.

```

MUICache
Software\Microsoft\Windows\ShellNoRoam\MUICache
LastWrite Time Mon Jan 4 22:42:25 2010 (UTC)
C:\Program Files\VMware\VMware Server\tomcat\bin\tomcat6w
.exe (Procrun Service Manager)
Mon Jan 4 22:03:42 2010 (UTC)
UEME_RUNPIDL:%csidl2%\VMware (3)
UEME_RUNPIDL:%csidl2%\VMware\VMware Web Access (2)
UEME_RUNPIDL:%csidl2%\VMware\VMware Web Access\Tomcat 6.0
Program Directory.lnk (1)
Mon Jan 4 22:02:36 2010 (UTC)
UEME_RUNPIDL:%csidl2%\VMware\VMware Web Access\Configure
Tomcat.lnk (1)
UEME_RUNPATH:C:\Program Files\VMware\VMware Server\tomcat\
bin\tomcat6w.exe (1)
Mon Jan 4 21:51:19 2010 (UTC)

```

Figure 2.15: Registry entries by VMware application (Source: Barrett & Kipper, 2010, p. 85)

Yet again the function and the use of these registry entries remain unknown. This makes it hard to understand how one can extract artifacts through identifying registry entries in use. As a conclusion to the chapter, the authors suggest imaging the host machine; generating keywords and using FTK to launch a keyword search. This technique may still prove handy as even search hits on keywords could be enough to provide leads for further analysis.

In the following chapter - Investigating Live Virtual Environments (Barrett & Kipper, 2010) puts emphasis on server products like VMware ESXi and Microsoft Virtual Server 2005. The author believe the use of digital forensic procedures are very commonly used to investigate in environment consisting of static or dead drives, by static or dead the authors mean functional/non-functional drives. But with technology advancing, challenges such as encryption, large drive capacity and complex networking can limit the outcomes of existing procedures. As previously mentioned in (Zhang et al., 2010) large disk capacity and network complexity are among many concerns that forensic practitioners have over existing acquisition methods. "As organizations move to a virtual environment for servers and desktops, there is a good chance of running into a virtual environment when conducting a live investigation" (Barrett & Kipper, 2010). Although live forensics looks to be more acceptable in such situations, issues still exist with this type of acquisition, as live investigation may change the state of the targeted system leaving inconsistencies while verifying the evidence. In the past, attempts have been made in order to avoid this issue by pre-installing monitoring programs on users' workstations; this allowed the acquiring of the user machine over the network in case of an incident. Similarly, the

ability to perform live acquisition over the web browser was enabled by installing applets. But upon a test conducted in 2007 on various commercial tools, "Most of the tools were unsuccessful in being pushed to the virtual environment when given their own IP address" (Barrett & Kipper, 2010, p. 112). Problems, such as Microsoft's "blue screen of death" and compatibility issues with the virtual environment were apparent during the tests.

Furthermore, the authors' state types of evidence such as currently logged user, open ports, running processes, registry information and attached devices are still vital whether it is a virtual or physical environment. These types of evidence are mainly collected via logs, as Figure 2.16 illustrates, giving the location of log files found on the VMware infrastructure.

Log	Location
Virtual machines	vmware.log
Web access	/var/log/vmware/webAccess
Authentication log	/var/log/secure
VMkernel	/var/log/vmkernel
ESX server host agent log	/var/log/vmware/hostd.log
VirtualCenter agent	/var/log/vmware/vpx
System events	/var/log/messages

Figure 2.16: VMware Log Locations (Source: Barrett & Kipper, 2010, p. 85)

Likewise, in the analysis section, the authors base their findings on a theory shared in an article of *Digital Forensic Magazine* (Fitterman & Durick, 2010), tools and options for examining virtual data is limited. However, best practices and response methodology should maintain the integrity and fidelity of the acquired files and also maintain the proper chain of custody. One possible way this can be achieved is by collecting a snapshot of the VM and the .vmsn file, followed by copying (.vmx), (.vmdk) files using an ESX service console. It is interesting to see chain of custody mentioned, as this would typically refer to physical devices, but a command "esxcfg-info" maps storage volumes to physical devices and displays recordable information such as serial numbers of the hard disks connected. As part of analyzing the evidence, the authors' mention that it is relatively easy to extract information from a virtual machine's memory based on a virtual server. Extracting the virtual memory file (.vmem) can be done using a VMware server console and analyzed using tools like Mandiant Audit by simply mounting the virtual memory file. Figure 2.17 gives a screenshot of the application reviewing running processes within a VM.

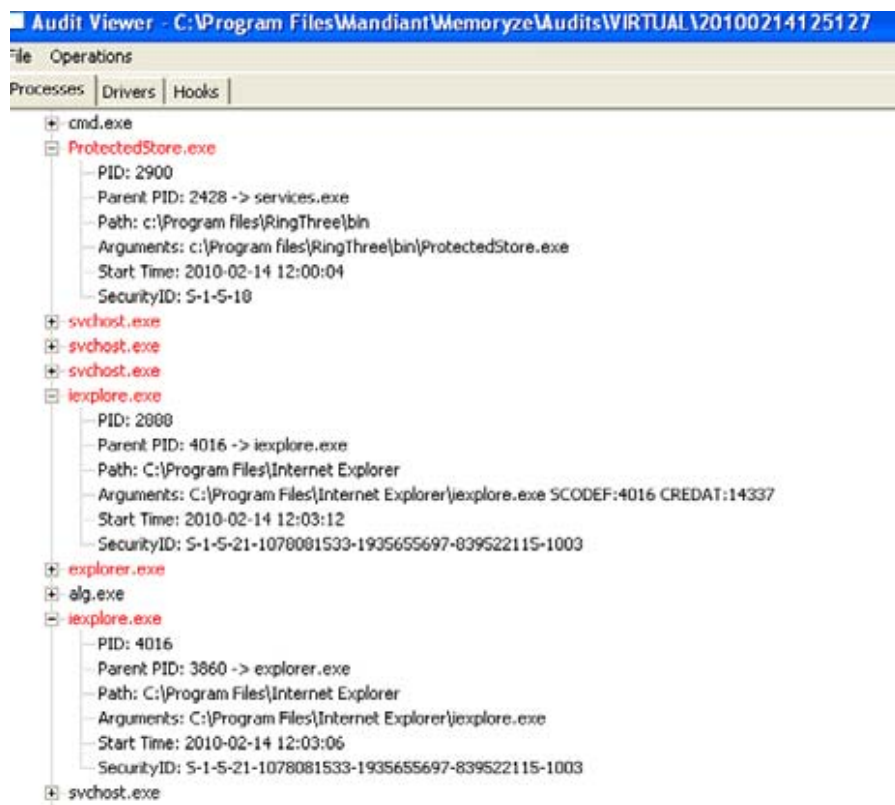


Figure 2.17: Mandiant Audit View (Source: Barrett & Kipper, 2010, p. 85)

2.5 CONCLUSION

In this chapter, the current state of digital forensics (Section 2.1), the current state of desktop computing (Section 2.2) and the current state of distributed computing (Section 2.3) were reviewed in relation to cloud computing. The structure of this chapter was arranged in this manner, so the progression of conventional computing toward cloud computing could be understood; in addition to this, security risk in relation to hosted virtual desktops based in private clouds were also discussed. This was to identify the various technical or non-technical challenges forensic practitioners could face while investigating crimes in a virtualized environment.

Furthermore, in Section 2.4 "Current forensic capabilities", three academic studies were reviewed to understand the misuses of desktop virtualization and successfulness of proposed methods used to conduct a digital forensic investigation. After reviewing these studies it was implied that, the forensic methods identified in the previous studies mainly demonstrate ways to trace the existence of a virtual machine and malicious activities originating from a rogue virtual machine. In most cases, these methods were practical for client hosted (Type 2 Hypervisor)

virtual machines as the source of the virtual machine was known. In a HVD infrastructure, these techniques may not be adequate as several virtual machines are hosted on a common platform i.e. on a (Type 1 Hypervisor), where searching, locating and analyzing the evidence will not be as straightforward.

Chapter 3 - Methodology

3. INTRODUCTION

In chapter 2, literature was reviewed based on topics surrounding desktop computing technology. The knowledge from Chapter 2 was used to create familiarity with the current state of desktop computing in respect to distributed computing and its effect on the field of digital forensics. In addition to this, existing studies on virtualization forensics were also reviewed.

As a result, it was understood that the client side of any form of computer system will always exist likewise desktops will always be in the reach of users either delivered physically or virtually. To cater to the demand for virtual delivery of desktops, the use of hosted virtual desktops is growing. In parallel to this, a lack of literature on the forensically readiness of this cloud architecture makes it hard to understand the suitability of current digital forensic practices on hosted virtual desktops. However a number of techniques exist to investigate client hosted virtual desktops such as VMware workstation, Virtual Box or Virtual PC which were reviewed in existing studies on virtualization forensics. The techniques identified in the existing studies are mainly targeted towards tracing the existence of a virtual machine or ways used by a suspect to mask activities using a virtual machine followed by methods to analyze them using the available tools. Most of the techniques discussed in the studies were seen practical as a result of knowing where the evidence existed. But in a hosted virtual desktop infrastructure, without knowing the how, the when, or the where, the current procedures may not be practical.

This chapter includes a definition of methodology along with an explanation of the difference between methodology and methods. Upon an understanding of the two terms, Section 3.2 will summarize the three commonly used methodologies in this type of research i.e. grounded theory, descriptive research methodology and case study methodology, to show the process of elimination and the need to form a custom made methodology. Finally, in Section 3.6, the hypothesis and the custom made methodology will be introduced.

3.1 WHAT IS METHODOLOGY

A methodology is defined as a structure of methods, rules, procedures or set of procedures employed by a discipline (*Methodology*). The application of methodology to research can solve research problems systematically. While designing a research methodology for the targeted study, it is important to differentiate the research methods/techniques and research methodology. Research methods are understood as the use of various methods and techniques to conduct the research, and is often used to describe experimental processes such as collecting data, making observations or evaluate results obtained (Ross & Morrison, 2004). The most commonly used research methods are direct observations, personal/group interview, surveys and case studies. While methods are more general, it is the methodology that triggers the use of the method. During a study the researcher uses methodology to set criteria for applying appropriate methods towards an applicable problem within the scope of study, as methods/techniques may differ according to the nature of the identified problems in the research. Hence we "say that research methodology has many dimensions and research methods do constitute a part of the research methodology. The scope of research methodology is wider than that of research methods" (Kothari, 2004, p. 8).

Prior to adapting a research methodology for the chosen study, the type of research and a suitable research approach needs to be considered in order to select an accurate method. The two basic types of research approach that exist are qualitative and quantitative. The qualitative research approach was developed to research on topics related to social science and natural science, allowing the researchers to study natural, cultural as well as social phenomena (Myers, 1997). The qualitative approach is used in disciplines where there is an existence of previous data or studied cases as the conclusions are only propositions (informed assertions)(Tashakkori & Teddlie, 2003). Typical data collection methods employed in qualitative research are narratology, grounded theory, storytelling and ethnography. On the other hand, the quantitative "research approach is based on the measurement of quantity or amount. It is applicable to phenomena that can be expressed in terms of quantity" (Kothari, 2004, p. 3). In other words it has the ability to quantify relationship between variables like weight, time or performance. In the conventional view, the quantitative approach is aimed at an experimental or descriptive type of research. In an attempt to use a quantitative approach during experimental research,

measurements are taken while interventions are in place, as where in descriptive research measurements are taken on the existing behavior or condition (Hopkins, 2008). Quantitative research approach is also sub-classified into experimental and simulation research. "The experimental approach is characterized by much greater control over the research environment and in this case some variables are manipulated to observe their effect on other variables" (Kothari, 2004, p. 5). While the simulation research approach is understood to be an artificially created replica of a system, used to generate relevant data under specific conditions and to observe any dynamic behavior shown by the system.

3.2 RANGE OF METHODOLOGY'S

There is range of methodologies available that could possibly suit the nature of this study. The methodologies chosen for discussion in this section are grounded theory, case study and descriptive. These were shortlisted as these methodologies are used in various forms to conduct research in the field of digital forensics and computer science. The following section will discuss the essential background and fundamental understanding of each methodology.

3.2.1 Grounded Theory

The Grounded Theory Methodology (GTM) is defined as the discovery of theory from data systematically obtained from social research (Glaser & Strauss, 1967). The methodology was originally coined constant comparison and later became known as 'grounded theory' as a result of the book *Awareness of Dying* by Barney Glaser and Anselm Strauss in 1967. Glaser and Strauss created this methodology to conduct research on terminally ill hospital patients. Grounded theory is also commonly understood as examining the experience of participants within a research context. As stated by Strauss and Corbin (1998) "if a researcher is interested in knowing what it is like to be a participant in a drug study [...] then he or she might sensibly engage in a grounded theory project or some other type of qualitative study" (p.40). Grounded theory differs from other research methodologies as it has the explicit sense of nature. It aims to study the notion of an existing research situation (as it is) rather than starting with the intention of testing a specific hypothesis (Dick, 2000). As such, grounded theory methodology is popular among investigative research types who rely on empirical data.

GTM uses theoretical sampling. Selection criteria are defined by the researchers and are specified according to initial findings. Participants are then selected using these specified criteria. The sampling process is directed by the on-going development of the theory with early analysis of data directing researchers towards concerns that need further study.

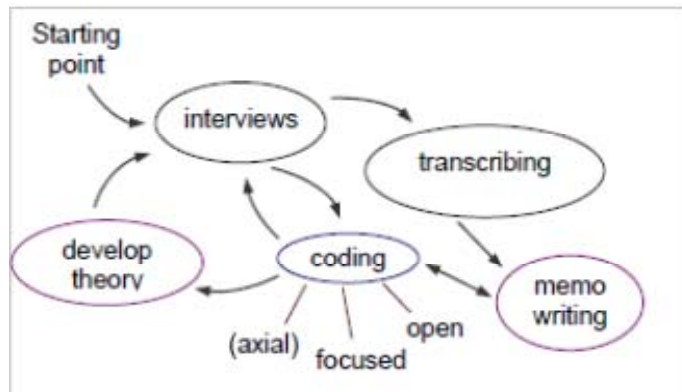


Figure 3.1: Iterative Cycle used in GTM (Source: Gorra, p. 87)

Figure 3.1 illustrates the iterative cycle of introduction and elimination that takes place in studies that use GTM. The cycle works to guide further sampling by continuously comparing current results with new findings (Strauss & Corbin, 1998). As such, the interviewees initially define the variables of the research. Past this, researchers use the cycle to further develop the variables through their own findings and research. This continues until no new or relevant data can be found, which is referred to as *theoretical saturation*. Once achieved, relationships are made within the categories of data. Though any method of data collection may be undertaken, interviews are often the primary source of data that theories will be developed from. Other activities can be used to generate data including colloquial conversation, group feedback analysis and even focus groups. Codes are then used to label experiences significant to the participants. Codes can be divided into conceptual categories that are used as a basis for the theory being developed (Dick, 2000).

Glaser was against the idea of note taking during data collection, including recording interviews. Below is an example of how interview notes would be set out, with ‘notes’ to the left-hand side of the page, and ‘coding’ to the right. Any relevant bio-data about the interviewee will also be noted, which can help to identify participants (Gorra).

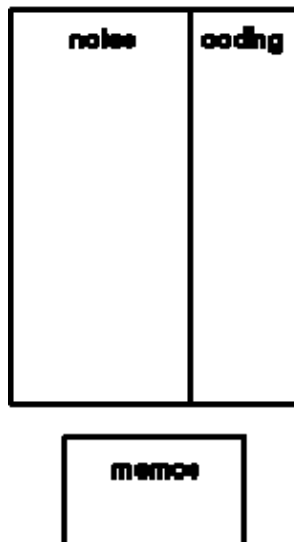


Figure 3.2: Interview Notes Format (Source: Dick, 2000, p. 59)

Ultimately, the choice of research will guide your initial sample entering into the iterative cycle. Though Glaser seems to offer no clear definition for an original sample, it can be assumed, for example, that if your situation involves a multitude of people, your initial sample would include a group as diverse as possible. Categories will arise from the data and will add the need for additions to the sample in order to increase relevant diversity. Defining the characteristics of and relationships between your categories seeks to strengthen the theory.

A memo in GTM refers to some note you make to yourself in regards to a hypothesis behind a category, or a relationship(s) between two categories. At some point, your primary category and its relational categories would have reached saturation. At this point it can be assumed that you would have written a considerable number of memos. These memos would outline the different aspects and theories within your data as it has evolved.

GTM works off the idea that your theory is hidden within data and the situations surrounding it, for you to extract. Coding allows researchers to label experiences and categorize them, whilst making memos allows you to build relationships between the categories themselves and extract the relevant theories behind them.

3.2.2 Descriptive Research Methodology

"Descriptive research involves gathering data that describe events and then organizes, tabulates, depicts, and describes the data collection" (Glass & Hopkins, 1984). It can be categorized into two types of information; qualitative or quantitative. Quantitative information is numerical in nature (i.e. the number of people that fit into a certain category) and is often aided by graphs and charts to give a visual representation of the data for ease of understanding; whilst qualitative information refers to the categories themselves (i.e. the qualities that divide the quantitative information). Descriptive research aims to describe and explain findings in an attempt to validate them. "Description emerges following creative exploration, and serves to organize the findings in order to fit them with explanations, and then test or validate those explanations" (Krathwohl, 1993). Research programs describe the specific characteristics of natural or man-made situations in order to extract knowledge from these situations that we might not otherwise understand.

Large amounts of data can be increasingly difficult to interpret; hence transforming the data into a form that is easier to understand is of utmost important. As such, descriptive narratives are generally used to help in the understanding of a study that is qualitative in nature, and the implications it holds. The narratives help to outline patterns that emerge during the analysis of the data at hand. To gather data, descriptive methodology is known to commonly use the following methods.

3.2.2.1 Surveys

Kothari (2004) states that primary data can be collected in a number of ways, particularly in surveys or descriptive research. For experimental studies, data is collected by observing the experiment itself. For descriptive studies data can often be obtained or recorded through observation of the circumstance in question or direct communication with respondents through personal interview or other forms

3.2.2.2 Observations

For descriptive research, observation is used most commonly, particularly for behavioral studies. Everyday observation by the average person is usually not considered to be scientifically inclined. However, it becomes a tool for researchers when used to accumulate relevant data. Structured observation works through succinct definition of the elements being observed,

consistent conditions and the noting of relevant data. If these conditions are not met prior to the observation, the observation is called unstructured observation. Structured observation is deemed appropriate in descriptive research (Kothari, 2004).

3.2.2.3 Interviews

For study purposes, data collection through interviewing is achieved by the recording of oral-verbal responses to oral-verbal stimuli (i.e. questions or statements) (Kothari, 2004). Interviewing can be done in person, or when possible, over the phone. Structured interviews are often used for descriptive studies as it is economical, provides a base for generalization and on the interviewer's part, requires lesser skill.

In addition to the above methods for data gathering, case studies are also used to collect data of individuals or organizations. All these methods collectively are the biggest strength of this methodology as they remove any strict academic barriers, making it easier for researchers to provide an insight into real life experiences or how others may experience an event, which other methods fail to do. However, the use descriptive research methodology can also conclude fictitious or inaccurate findings as, during the collection of data, an interviewee may not answer questions properly. During an observation, the subject may behave or act differently upon knowing that it is being observed.

3.2.3 Case Study

A case study is defined as "an intensive analysis of an individual unit (as a person or community) stressing developmental factors in relation to the environment" (case study, 2011). The Case Study Methodology (CSM) has been used in a variety of studies, and is well suited to a detailed and comprehensive investigation (Feagin et al., 1991). (Yin, 2003), and other researchers have developed sound procedures that are as developed and durable as any in the field of science. Stake (1995) has suggested that data collection and analysis tends to hide details, whereas case studies are structured in such a way that they bring out aspects from the standpoint of the participants by using multiple source of data.

Specific types of case studies include exploratory, explanatory, and descriptive. Stake (1995) went on further to define three more, *Intrinsic* (the researchers have an interest in the case), *instrumental* (used to understand what is not obvious) and *collective* (a group of case studies).

Exploratory cases are sometimes said to be preliminary to social research. Explanatory case studies are suited to studies aiming to attribute cause and effect. Descriptive cases require a descriptive theory to be developed prior to the case study. Single-case or multiple case applications are able to be applied to all of the above methodologies.

Stake (1995) state that the design chosen for a specific case study is to be taken into account particularly for that studies whose:

- Participants you cannot manipulate (i.e. change their behavior)
- Focus is the 'how' and 'why'
- Context you consider to be highly relevant to the area of study
- The area of study and context are not clearly divided.

CSM combines various methods in order to offer several viewpoints to a case. Careful organization of data is critical when looking at case study research as the amount of data accumulates as it comes from multiple sources. This ensures the researcher does not become so engulfed in the data that he/she is diverted from the original focus of the study.

As previously mentioned, CSM uses multiple data sources. Stake (1995) state that this increases the credibility and basis the data gives to a study. Researchers have a excess of data sources to choose from, many of which are easily accessible, including interviews, physical artifacts, observations. In relation to other types of qualitative study, case studies are able to help researchers gain a holistic view of the area of study by being able to accumulate and integrate data from quantitative survey data. As opposed to being handled on a case-by-case basis, the data from these varying sources is combined during their analysis. By bringing these sources together, researchers are able to increase the credibility and foundation on which their theory stands as it gives greater backing and understanding of the case.

Effectively organizing data has been seen to be of utmost importance (Yin, 2003). By organizing your data, easy retrieval of specific information is possible. One of the most effective ways to achieve this is by using a database in which the information is placed. A database allows for easy retrieval, particularly when attempting to access more specific information such as notes, key documents, photographs, interview files etc.

Data collection and analysis takes place simultaneously, much like other qualitative studies. The form of analysis used is highly dependent on the type of case study taking place. Analysis can be approached in five ways; pattern matching, linking data to propositions, explanation building, time-series analysis, logic models and cross case synthesis (Yin, 2003).

Because the approach of the methodology can often be highly complex, case study reporting can be a daunting responsibility. To succinctly report findings in an easy-to-read format is difficult at best. A report aims to give readers a detailed and holistic view of the case study in order to let themselves be engulfed by the study itself, allowing them to decide if the study can be applied on a personal level. Though there is no 'right' way to report a case study, it is important that the researcher details the context and the subject of the study.

CSM, whilst considered highly robust, has its limitations. Without an appropriate number of sources through which to accumulate data, results can be seen as insufficient or weak – something CSM aims to eliminate. In addition to this, behavior changes in participants (due to being closely observed) can cause results to become skewed in relation to the reality of the phenomenon that is being studied. Other limitations include a reader's personal relevance to the situation and the complexity of studies using CSM that makes them unsuitable for rudimentary level readers. Case studies are an effective method to research a phenomenon of interest. It is a robust methodology, drawing its data from various sources to produce sufficient amounts of data to provide a holistic view on the phenomenon. It has been used in varying disciplines for a plethora of reasons; strengthening or supporting a theory, providing basis for a new theory, attributing cause and effect, and even simply to describing a phenomenon. CSM allows for realistic application to human situations and provides the public with easily accessible information via reports. When used correctly, CSM aims to facilitate studies so they may provide relevance to the layman's everyday experience and aid in the explanation of situations that might otherwise seem difficult to understand.

3.2.4 Scientific Methodology (SM) - Research

"The scientific method is the process by which scientists, collectively and over time, endeavor to construct an accurate (that is, reliable, consistent and non-arbitrary) representation of the world" (Wilson, 1952) SM is the methodology used to try and link cause and effect in nature. This is

done by changing variables in one circumstance of the experiment and predicting what impact it will have on the specific subject. The key concepts of scientific methodology consist of observation, hypotheses and deduction, which are used by scientists to draw conclusions from the experimental data. Thinking of various ways to use SM, scientists attempt to answer the question(s) at hand.

Observations done on a one-off basis have little or no value in terms of the SM. Additional observations are done deliberately in controlled or non-controlled environments in order to confirm or argue against the first observation. Observations can be made to answer or solve a question or problem (Wilson, 1952). Such an example might be 'If X is present, will Y occur?' where X may be a condition or stimulus and Y may be a response. Research is a suitable synonym for 'observation' in terms of the SM. It is used as a tool to understand the problem at hand in order to answer the question being asked.

Once the observations have been completed, the next stage in the SM is the formulation of a hypothesis. The hypothesis is a statement that outlines a scientists reasoning around the observations in an attempt to explain the phenomena being observed.

The hypothesis is a more holistic statement in regards to explaining the phenomena at hand. Predictions allow scientists to more specifically pin point what they are aiming to experiment on, and as such help them prove or disprove the hypothesis (Shuttleworth, 2009). Predictions are made by assuming the hypothesis is true – and as such aid in proving/disproving it.

After the predictions have been set, experiments are used to collect further observational data in order to prove or nullify the prediction, and in turn the hypothesis. Circumstances and events are used and altered depending on what scientists believe is the best way to test the prediction. More succinctly and arguably more simply put the experiment is designed to observe responses based on certain stimuli or environmental characteristics. Wilson (1952) also mentions that the experiment is the pillar of the SM and, as such is to be treated with utmost care. Due diligence is taken which include static and changing variables (explained below), as well as controls to ensure the observations being made are in direct relation to the change in the environment. Static variables refer to parts of the experimental environment that do not change. Changing variables refer to parts of the experimental environment that can change (either by choice, or naturally).

Using the observations taken during the experiment, scientists go back to their predictions and hypotheses to conclude their research. Careful calculation and linking of ideas help to prove or disprove the hypothesis. Careful calculation must be done in a number of ways. These include linking the variables and perhaps finding other reasons for the results. In addition to this, statistical significance plays a large part in determining whether the experiment at hand is to be taken as a reliable source for cause and effect in its relevant area of study.

3.3 RESEARCH DESIGN

In order to ensure a systematic research flow, the following section will define the use of the research methodology chosen for this research, which will be a mixture of the scientific research methodology but adopting observational research methods as applied to "quasi-experimental case scenarios". Subsequent to studying the previous literature within the area of virtual desktops in private clouds, research methods and gaps were identified. This guided the researcher to construct a problem statement on the adequacy of current digital forensic procedures for investigating crimes involving hosted virtual desktops. To answer this question various research methodologies were short listed. The descriptive research methodology was most commonly found in used in the existing studies. In the context of desktop virtualization this methodology was mostly used to describe VDI. The focus of a descriptive study is entirely describing the target, and its predictions are based on statistical information rather than factual proofs (Kothari, 2004). In which case using the descriptive method would have partly succeeded, but any investigative work where experiments are required would not have been possible. Initially, the case study method was also considered for this study, as case studies are typically used to discover the interrelationship between the process and the ability to observe factors held accountable for different behavioral patterns. Similarly, it was expected that discovering the interrelationship between digital forensics and hosted desktop virtualization would rely on building a case study. However, due to a lack of technical expertise and the availability of non-fictional cases on HVD infrastructure, the appropriate depth required could not be reached to build a robust case study. Grounded theory methodology (GTM) was also reviewed for this study, however, it was eliminated because GTM uses theoretical sampling and develops a theory by collecting data from interviews. Yet again, due to hosted virtual desktop being introduced only in the recent times, many practitioners lack experience in virtualization as they have yet to

investigate a case related to HVD, and so interviews would prove fruitless as a means of seeking answers.

None of the above commonly used methodologies would have been appropriate to study desktop virtualization in a private cloud infrastructure, although this technology has been around for a while, it has most commonly been utilized in public clouds as Platform- as-a -Service (PaaS), which is outside the scope of this study. But the use of hosted desktop virtualization has been more recently introduced in private clouds and so the occurrence of criminal activity may not yet exist. As a result, the overall impression is that hosted virtual desktop infrastructure has not yet been contemplated in the context of digital forensics, even though it is more than or equally vulnerable than traditional desktop computers. As the possibility of observing a case study based on non-fictional case scenarios is rare, this study will utilize a scientific observation method to observe quasi-experimental scenarios, which are discussed further on.

3.4 HYPOTHESIS/PROBLEM STATEMENT

It is understood that hosted virtual desktops in private clouds are a recently introduced technology. In cases where HVD is compromised, locating and extracting evidence in a forensically sound manner will be necessary. Hence the problem statement identified for this study is as follows:

Problem statement: *It is currently unknown whether existing forensic methods are adequate for investigating crimes occurring within Hosted virtual desktops, based in private clouds.*

Based upon the literature review and the collection of facts concerning the gap between hosted virtual desktops and digital forensics, a research hypothesis for this thesis is identified below:

Null Hypothesis: *Current digital forensic procedures are adequate for investigations involving Hosted virtual desktops based in private clouds.*

Falsification of the null hypothesis will take place if a new digital forensic method is found to be required to deal with quasi-experimental scenarios introduced in this thesis. In saying that, due to the futuristic nature of this research chance of making errors during the experimentation are also possible. However these will be recognized by the courtesy of FIDM in the early stages. Possible occurrences of errors will be determined if during the iteration, FIDM fails to provide feedback

(negative or positive) for the next scenario to progress. The failure to provide feedback could possibly be due to software bugs, unsuitable tools or improbable results.

Due to the lack of previous work in this area, the choice of research methodology is important for generating confidence that the approach adopted for investigating the main problem in this study is soundly based. Hence the action research methodology was preferred over the other methodologies. Action research methodology is defined as the combination of theory and practice interleaved in an iterative process, where practitioners and researchers act together on problem diagnosis by combining action, and reflective learning (Avison et al., 1999). This means that, action research will be used in conjunction with scientific research methodology, while the null hypothesis for this thesis; Current digital forensic procedures are adequate for investigations involving hosted virtual desktops based in private clouds, will be tested and observed using the scientific observational method by implementing and running a problem based simulation of HVD and investigating it by applying the existing digital forensic procedures.

Like the name suggests, it is understood that outputs are gained by taking actions. In order to put theories into action, the research methodology was combined with the elements of the scientific research process. The methodology was customized to accommodate the process of designing and developing plausible case scenarios and simulating these case scenarios on a suitable platform and then analyzing the findings of the simulations in a scientifically sound manner. The simulation of hosted virtual desktops will be used to run three scenarios based on cloud security challenges. Each scenario will be investigated in a forensically sound manner, and best practices will be followed as far as possible.

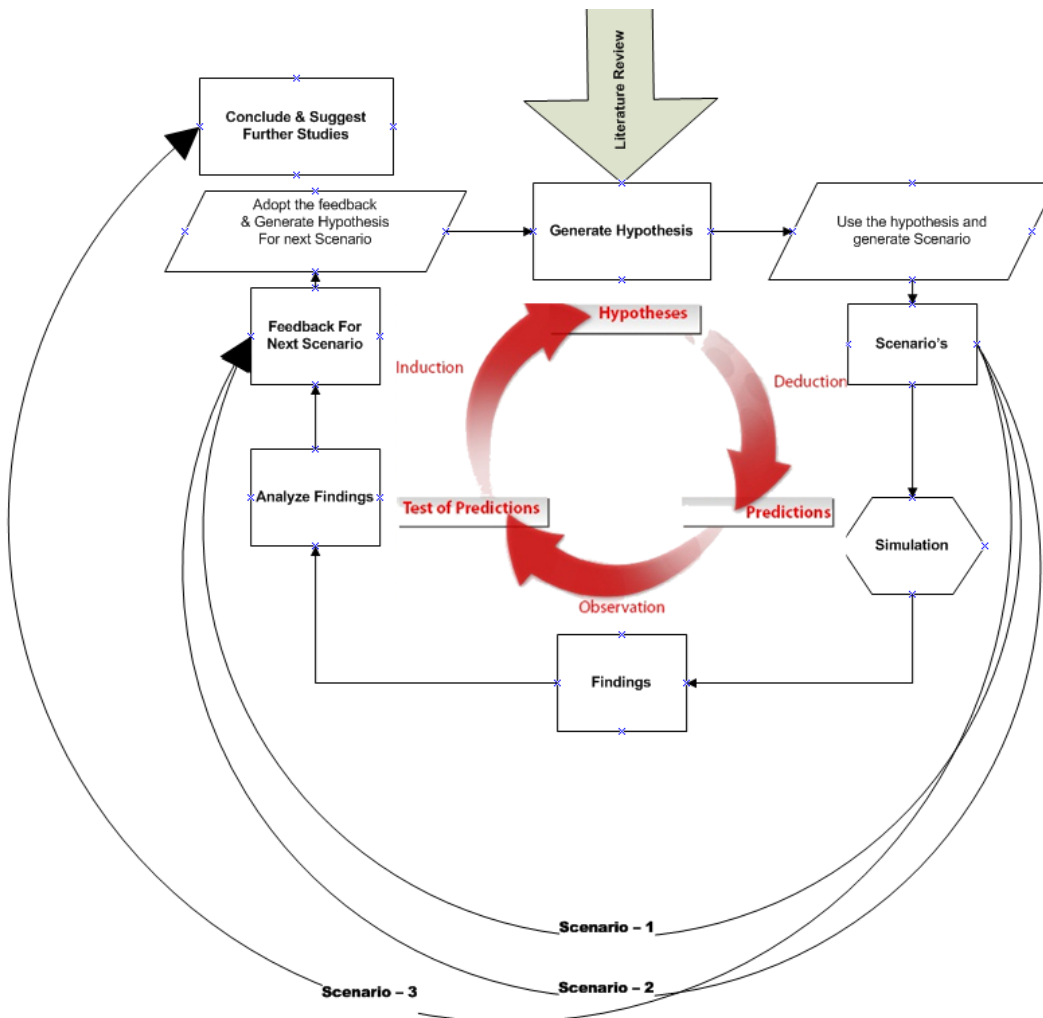


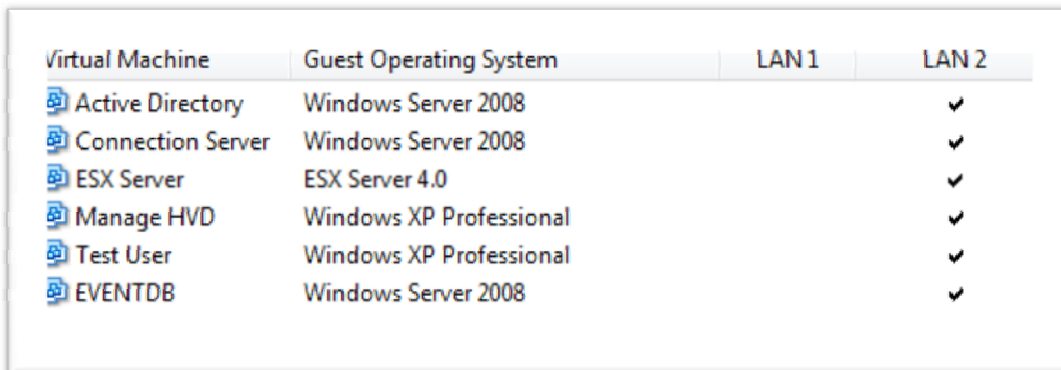
Figure 3.3: Forensic Iterative Development Model (FIDM)

In Figure 3.3 above, the Forensic Iterative Development Model (FIDM) demonstrates the use of the scientific research process with action research. The null hypothesis, from a science perspective, is: “Current digital forensic procedures are adequate for investigations involving hosted virtual desktops based in private clouds”, three case scenarios (to be described below) are designed and hypotheses developed to generate suitable simulations of these case scenarios to produce maximum data. With the simulation data, measurements are made and interim conclusions reached. These interim conclusions feed into the design and simulation of the second and third case scenarios, hopefully to produce better data and improved conclusions. At the end of the third iteration through the methodology, the conclusions reached are evaluated against the null hypothesis. The research concludes with suggestions for future research.

3.5 EXPERIMENTS AND METHODS

The experiments are conducted through simulations of three case studies scenario's. The aim of the experiments is to locate and extract evidence within the HVD infrastructure. The case studies used to generate the experiments are based on plausible scenarios typically found in standard forensic IT investigations but modified to be applicable to a HVD infrastructure and environment. The methods adopted are taken from standard digital forensic procedures with the aim of identifying their appropriateness and adequacy for analyzing the data and results of the experiments.

With regard to the simulation infrastructure, the simulated model of a hosted virtual desktop in a private cloud will be constructed on a workstation consisting of VMware Workstation 7 and the use of VMware 4.5 evaluation version. In order to deploy HVD successfully, five major components are required: an active directory, an ESX Server, a connection server, an infrastructure tool and an event database.



Virtual Machine	Guest Operating System	LAN1	LAN 2
Active Directory	Windows Server 2008		✓
Connection Server	Windows Server 2008		✓
ESX Server	ESX Server 4.0		✓
Manage HVD	Windows XP Professional		✓
Test User	Windows XP Professional		✓
EVENTDB	Windows Server 2008		✓

Figure 3.4: Infrastructure in VMware Workstation 7

To accommodate the essential components of the infrastructure, five virtual machines were created with relevant operating systems, as shown in Figure 3.4. Every component of the infrastructure is on the same LAN segment (hence ticks under LAN 2) because every component needs to be part of the same domain with a static IP address.

The purpose of an active directory in this infrastructure is to manage users, groups and apply control using group policy while the virtual desktops are in operation. An ESX Server is a bare metal hypervisor which stores every virtual machine created. The connection server is the key component that is responsible for maintaining the connection between the VM stored on the

hypervisor and the clients, to delivering desktops virtually. It can also be used to group virtual machines in pools and apply permissions. VMware vSphere is used to implement and maintain virtual machines (Manage HVD). It can also be used to maintain hardware allocations for every virtual machine. The event database (EVENTDB) maintains records related to the virtual machines and changes to the infrastructure. Data such as time and date of virtual machine creation, shutdown, and restart and by which user, is held in this database. Users (Test User) within the infrastructure are generated as required by creating a number of virtual machines configured with Windows XP. Every virtual machine needs to be implemented with View Agent software and also configured on the infrastructure's domain to be visible in the infrastructure. Similarly, Client Agent needs to be installed on every client machine trying to access their VM.

As mentioned earlier, three case scenarios were designed and developed to produce simulations that, in turn, lead to data and results for evaluation against currently used forensic IT tools, methods and techniques.

3.5.1 Scenario 1 - Persistence VM

In a company, the vice president (VP) is suspected of surfing pornographic websites during office hours. An anonymous complaint has been lodged with the company's director of HR that the VP's desktop screen had been briefly observed displaying pornographic material. Due to the high standing of the suspect, the company's HR department wants to gather solid evidence of the VP's activities before confronting the VP. The director of HR approaches a computer forensic team to investigate whether evidence exists concerning the VP's alleged activities. In the company every employee has either a designated workstation or a laptop connected to the company's in-house cloud infrastructure. The VP was using his company laptop while surfing and could access external sites through the company's in-house cloud.

3.5.2 Scenario 2 - NON-Persistence VM

It is reported that sensitive data has been accessed by a user during office hours, which is not consistent with his role in the company. The data contains the company's future development plans and client information, which makes it very to competitors. The login used to access the sensitive information belongs to a genuine trust worthy employee, with moderate computer skills. In addition to this, it was discovered that the user had misplaced his flash drive and

reported it to reception. The flash drive contained a hidden file containing login information along with other data. It is suspected that user login details were stolen from the flash drive and used to access the sensitive information. The question's are; Who?, When?, How?.

As part of cost cutting, the company has recently adapted desktop virtualization to cater for task workers and visitors. The local administrator has not been thoroughly trained to manage the newly introduced system; hence the HR manager thinks it's best to seek advice from digital forensic experts.

3.5.3 Scenario 3 - Multi-Tenant - Persistent VM

A school's ISP has detected and blocked the attempt of a user in the school to surf an offshore network containing objectionable material. The ISP wants to warn the school's head teacher for everyone's safety.

The school's administrator has limited knowledge about the recently deployed virtual desktops, which are linked to an in-house cloud. The head teacher doesn't want to risk the school's reputation, so a private computer forensic team is hired to investigate this matter and report their findings.

3.5.4 Setup of Simulation

The three case scenarios were accordingly implemented using the simulated model of HVD. Every case scenario contained the mandatory components outlined earlier in figure 3.4. The variations in numbers users, virtual HDD configuration and pooling configurations were controlled according to the objective of every case scenario.

Scenario	Virtual HDD Configuration	Pool Type	VM Allocation (User Assignment)	Event Database Status
1	Persistent	Manual	Dedicated	On
2	Non-Persistent	Automatic	Floating	On
3	Persistent	Manual	Dedicated	Off

Table 3.1: HVD configuration per scenario

As Table 3.1 above shows, the intention of scenario 1 is to test the adequacy of the current forensic procedures on crimes occurring in persistent (static) virtual desktops, hence a suitable environment is configured. Likewise Scenarios 2 and 3 intends to test the adequacy of the current procedures against non-persistent (Dynamic) and multi-tenant environment and so the appropriate settings are applied. The meaning of different configurations used to control each scenario is as follows:-

3.5.5 Modes of Virtual HDD

3.5.5.1 Persistent

Persistent setting is a disk mode that allows the virtual disk to behave like a conventional disk drive on a physical computer. The data is written to the disk and stays intact until deleted by the user. This disk mode is the simplest to use according to VMware (VMware, 2009a).

3.5.5.2 Non - Persistent

Non-Persistent setting is a disk mode that saves changes per session, as all the user data/settings are lost once the virtual machine is powered off or reset. The non-persistent disk mode is designed to suit for purposes such as software testing, technical support staff, software demonstration or a guest environment. Changes to the disk are written to and read from a redo log file that is deleted when you power off or reset (VMware, 2009a, p. 27).

3.5.6 Pool Type

3.5.6.1 Manual

The manual pool type can contain physical machines, VMs and VMs on different virtualized platforms. Desktops are provisioned according to end users demand.

Automatic - The automatic pool type utilizes VM templates to generate a new virtual machine each time an end user request it. VM templates are created with custom settings prior to this process.

3.5.7 VM Allocation

3.5.7.1 Dedicated

Dedicated allocation assigns a single desktop per user. The user is guaranteed to get the same desktop every time they login, if a user is explicitly assigned a desktop.

3.5.7.2 Floating

The user is not guaranteed to get the same desktop every time they login. Floating allocation assigns desktops according to the availability of resources.

3.5.8 Event Database

The event database is a feature built into VMware view 4.6 to monitor and store events in the HVD infrastructure, the event database is setup on the view connection server. Microsoft SQL & Oracle databases are supported.

3.6 CONCLUSION

Upon reviewing previous literature, different research methods and methodologies used in similar studies were examined. From the forensic point of view, previous literature suggests that it is possible to investigate cases involving client hosted virtual desktops, but cloud hosted virtual desktops are rarely discussed in the context of e-crime and digital forensics. Hence, the adequacy of current digital forensic procedures on crimes related to cloud hosted virtual desktops is unknown. In Order to develop a research methodology, various methodologies were discussed to find one that best fits. An approach consisting of a mixture of traditional scientific and action based research methodology using observational research methods to observe quasi-experimental scenarios was preferred. In addition to this, a forensic iterative development model (FIDM) was also created as a result of this approach. In this chapter, sections 3.5.1 to 3.5.3 also introduced three fictional case scenarios which were based on the top three concerns for both cloud security and digital forensics. The technical setup of these scenarios was also outlined in sections 3.5.4 to 3.5.8.

Chapter 4 - Findings

4.0 INTRODUCTION

In chapter three, a problem statement was generated based on the issues concerning the adequacy of current digital forensic procedures in hosted virtual desktop infrastructures. In order to address this research problem systematically, a research methodology was formulated which resulted in a scientific iterative model named the "Forensic Iterative Development Model" (FIDM) .Three quasi experimental case scenarios were also generated to run on the hosted virtual desktops simulation.

Research in the targeted field has been completed by following FIDM, a research methodology proposed in Chapter 3. Chapter Four will now report the findings from the simulation of the three case scenarios discussed in Chapter 3. Firstly, Section 4.1 will report any variations in the experiments while following the methodology. Subsequently, in Section 4.2; problems encountered in the system design will be highlighted and discussed. Then findings of each case scenario will be discussed according to the primary phases of a digital forensic investigation i.e. first responder findings, preparation, investigation and outcomes.

4.1 Variations in Data requirements

Using the hypothesis, suitable simulations were generated for the case scenarios, so maximum data could be produced. Although this was accomplished by following FIDM closely, minor variations were encountered during the actual experiments.

According to the iteration in FIDM, after every case scenario is simulated and the findings are analyzed; based on the analysis of findings from the current case scenario a feedback related to the design and simulation of the next case scenario needed to be outlined, thus better data and improved conclusions could be produced. However in actuality this phase could not be followed, largely because every scenario was underlining a different security area of desktop virtualization. Hence the underlying design of the simulation was setup accordingly. Therefore any feedback produced would have been irrelevant for the next scenario as the design of the simulation would be different. As a result of this glitch, the iteration was sustained by generating hypothesis for

the next scenario after analyzing findings from the previous one. Although this phase turned out to be irrelevant to this study, it cannot be considered as a flaw of this model because in instances where an experiment is repeatedly performed for observing variations in results, feedback can be a vital phase of this iterative model.

4.2 System Design

The most important part of the simulation was the base of virtual infrastructure consisting of five major components of the system as discussed in chapter 3 section 3.5. The success of every simulated scenario depended on the reliability of the virtual infrastructure. During the attempt to internetwork these components, various technical challenges were encountered.

Initially the plans were to lay out the HVD infrastructure on physical servers and populate clients on desktop computers and partly on virtualized platforms. However there were issues due to the lack of suitable equipments, like the VMware ESX (Hypervisor) was incompatible with the HP x345 eServer that was available. The server was too old to be compatible with any version of VMware hypervisor software. Likewise, a foreseen issue was the lack of suitable routers/switches to handle the expected bandwidth.

Even if the routers/switches functioned to a certain degree, there was a high chances of connection timeout between the virtual desktops and the hypervisor due to high latency and low bandwidth (Figure 4.1).

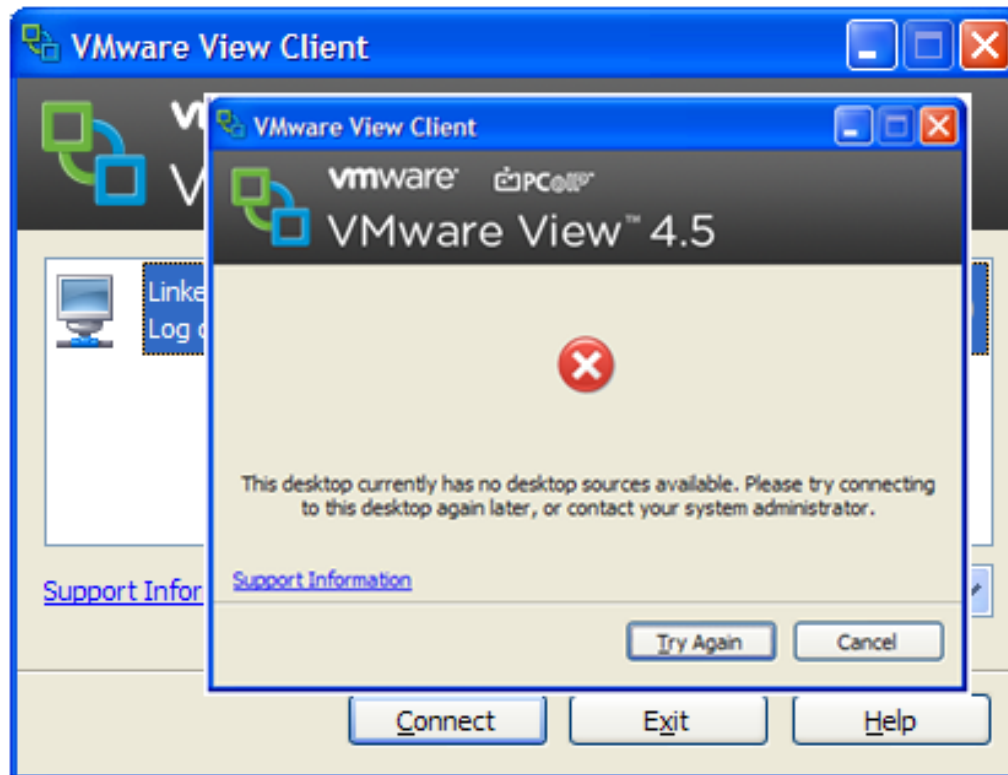


Figure 4.1: VMware View Client - Connection timeout error

Taking these concerns into account, the system design was modified by shifting the entire infrastructure onto client hosted virtual machines (Type 2 Hypervisor). By shifting the entire infrastructure it is meant that the VMware workstation was used to dedicate a virtual machine to every component of the infrastructure i.e. Active Directory, VMware ESX, vSphere, View connection server and Event database, then all the virtual machines were teamed, which allowed me to build a private network (LAN segment) to be built and the ability to control bandwidth within the LAN segment, as shown in Figure 4.2 below.

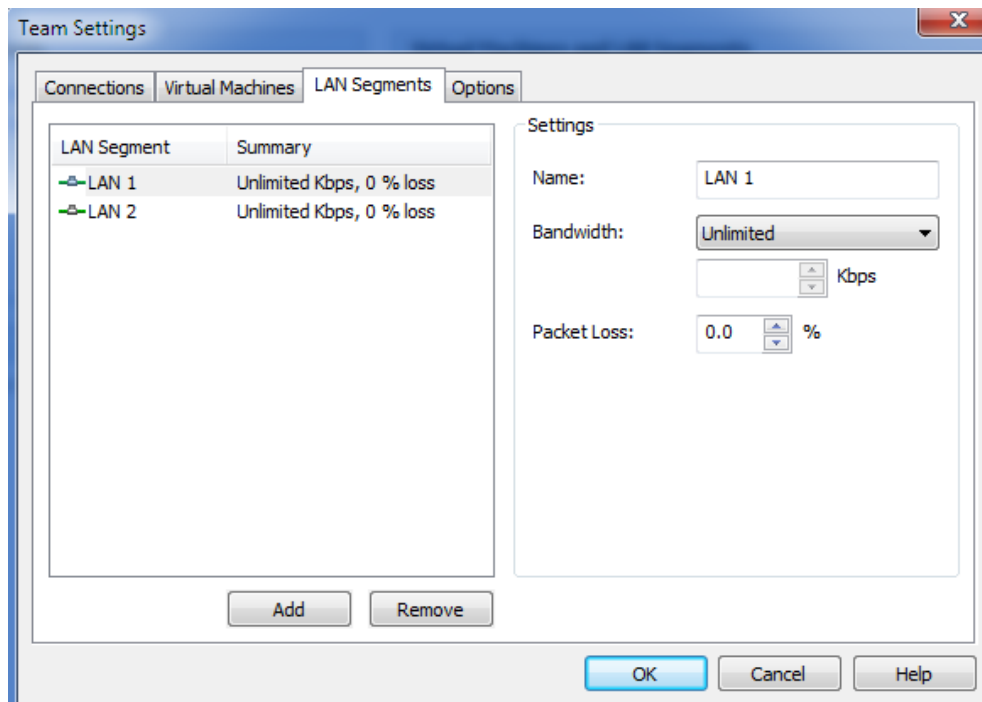


Figure 4.2: VMware Workstation: LAN Segment Configuration

Having resolved the issue of hardware compatibility, shifting the entire infrastructure also gave the ability to control and view results from a central point.

4.3 Scenario 1 - Persistent virtual environment.

Perform forensically sound data acquisition in a persistent virtual environment, and find evidence useful for proving that the suspect was surfing inappropriate websites during office hours.

4.3.1 Preparation - General Findings

- The company has recently deployed their own private cloud and hosted desktops for their employees.
- The infrastructure consists of 100 virtual machines organized in three pools. Task workers, power users and kiosk users.
- The suspect is part of the power user pool, which consists of stateful (Persistent) virtual machines.
- The reported date and time of the alleged incident was on 22/07/11 around 2:50pm, according to the anonymous report.

4.3.2 Search & Recognition

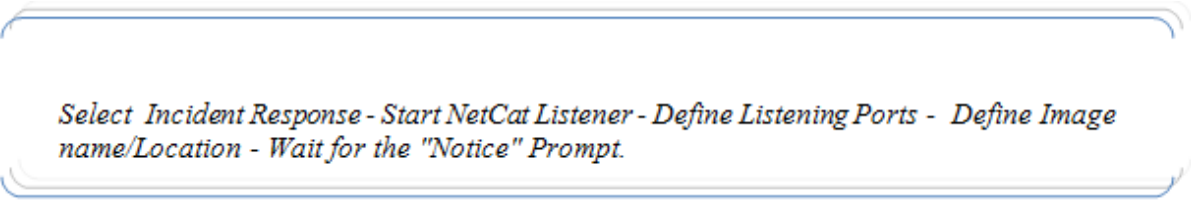
The suspected location of potential evidence is the VP's laptop; there is no other workstation or mobile device in the office as far as it can be ascertained. To avoid evidence tampering and alterations, all the network applications local to the VP's laptop were paused.

4.3.3 Collection

The VP's laptop was discovered in a state where a live acquisition procedure on the laptop's local disk could be performed. Although this was straightforward, the procedure for performing live acquisition on the virtual desktop was slightly more complex than performing it on a local machine. Acquiring a live virtual machine hosted on an ESX server was possible through two techniques.

Collection technique one involved the use of the e-fense Helix CD to obtain an image of the live system. e-fense Helix is a CD packaged with tools for forensics and incident respondents. The CD can be functional on a booted Windows operating system to perform live acquisitions or the bootable side, which creates a derivative forensic environment to perform imaging and basic analysis. Collection technique one for this scenario will use Helix while the Windows is booted, requiring an independent collection machine connected to the domain. So, Helix can perform a live acquisition of the memory/HDD belonging to the VP's laptop and output the results to the collection machine.

To prepare the collection machine, once the CD prompted the auto play menu following steps were taken to prepare the collection machine:



Select Incident Response - Start NetCat Listener - Define Listening Ports - Define Image name/Location - Wait for the "Notice" Prompt.

Figure 4.3: Command Helix 2.0

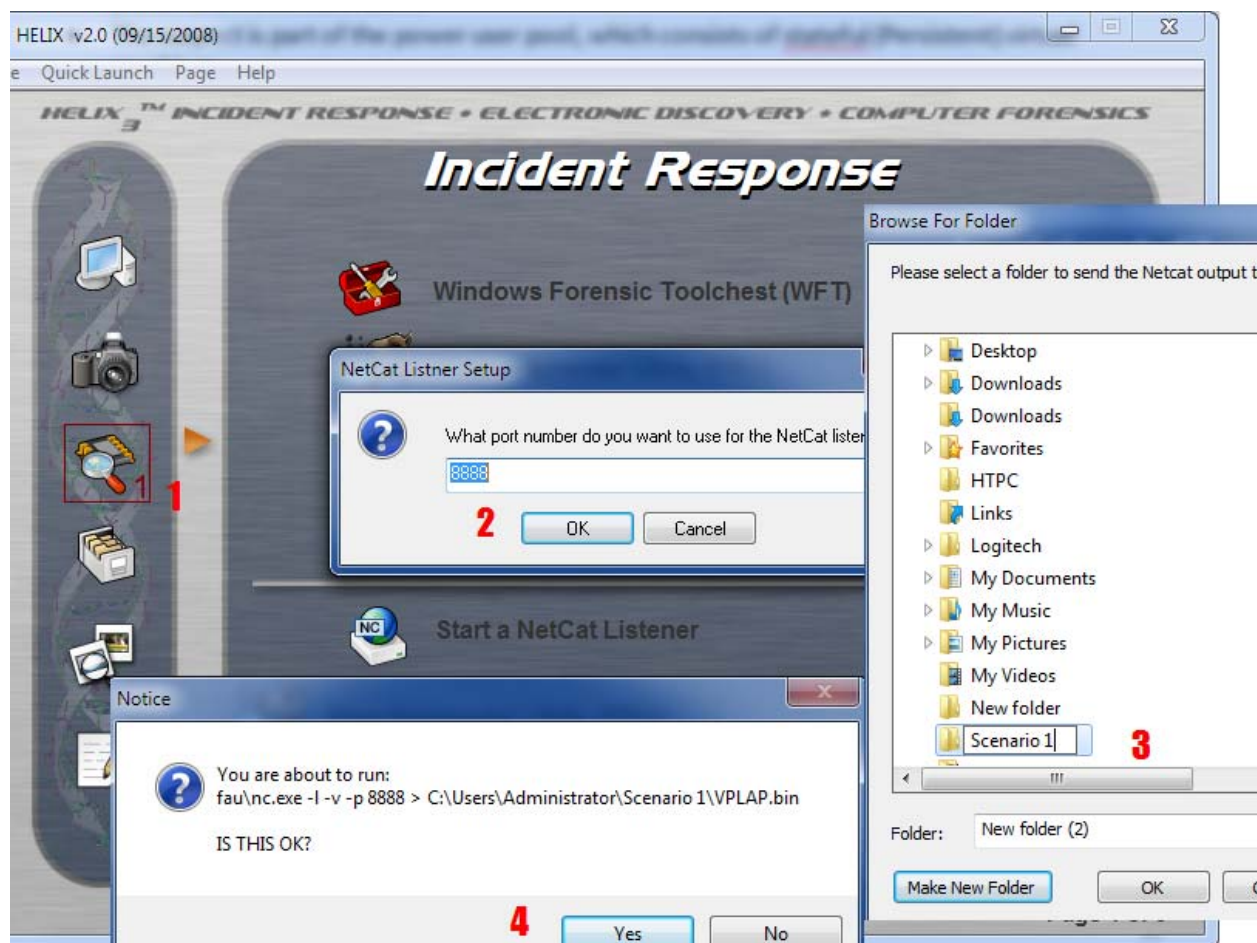
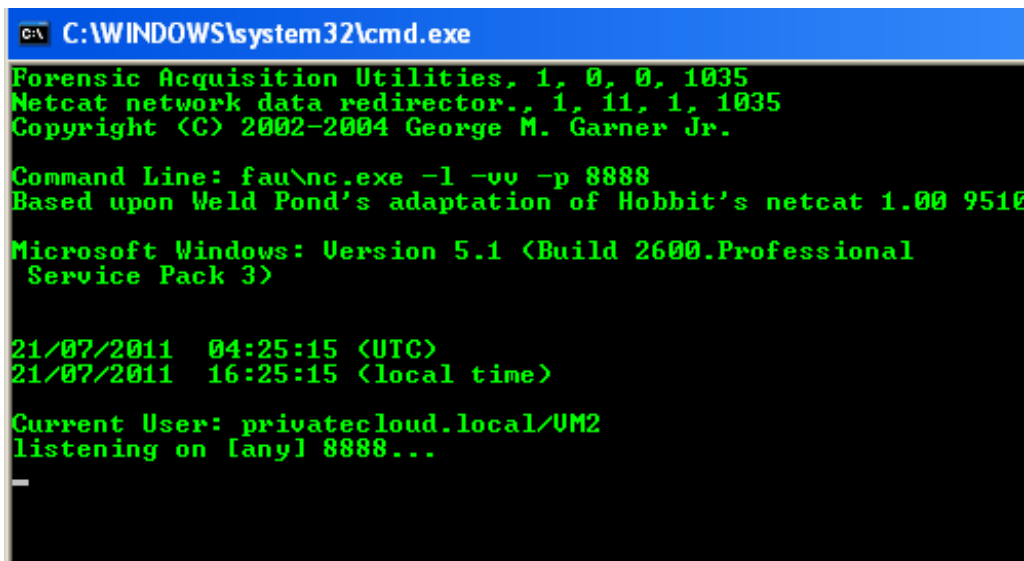


Figure 4.4: Helix 2.0 - Configure netCat Listener

Finally, a notice dialogue will prompt confirmation of the NetCat command, upon the confirmation, a command prompt will display the port number and the user being listened to (Figure 4.5).

A screenshot of a Windows command prompt window. The title bar is blue and contains the text "C:\WINDOWS\system32\cmd.exe". The command prompt itself has a black background with green text. The text displayed is as follows:
Forensic Acquisition Utilities, 1, 0, 0, 1035
Netcat network data redirector., 1, 11, 1, 1035
Copyright (C) 2002-2004 George M. Garner Jr.

Command Line: fau\nc.exe -l -vv -p 8888
Based upon Weld Pond's adaptation of Hobbit's netcat 1.00 9510

Microsoft Windows: Version 5.1 (Build 2600.Professional
Service Pack 3)

21/07/2011 04:25:15 (UTC)
21/07/2011 16:25:15 (local time)

Current User: privatecloud.local\UM2
listening on [any] 8888...
_

Figure 4.5: NetCat listening

When the collection machine was ready, the suspect's virtual desktop was prepared for e-fense Helix CD, so the VP's virtual desktop environment could be captured in a live state. While the virtual desktop was also found in live state, Helix CD image was launched into the HVD via VMware vSphere (Management console) as shown in Figure 4.6.

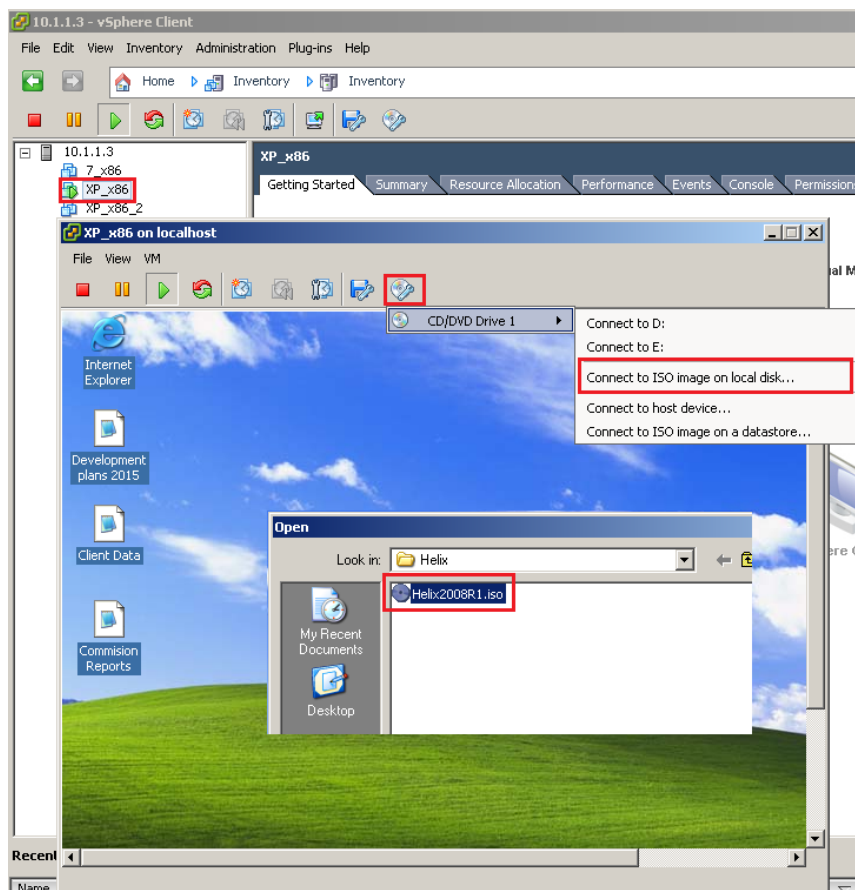


Figure 4.6: Prepare suspect's HVD for Helix

This mounted the Helix CD image on to the suspect's virtual desktop and the auto play menu prompted as usual. Next, in attempt to prepare Helix to acquire the memory/HDD of th suspects HVD, commands below were followed to navigate to the menu shown in Figure 4.7.

Select Acquisition - Source: HDD or Memory - Location: NetCat - Destination: collection machine IP - Image name: default - Block size: default - Conv: noerror - Acquire!

Figure 4.7: Command Helix 2.0

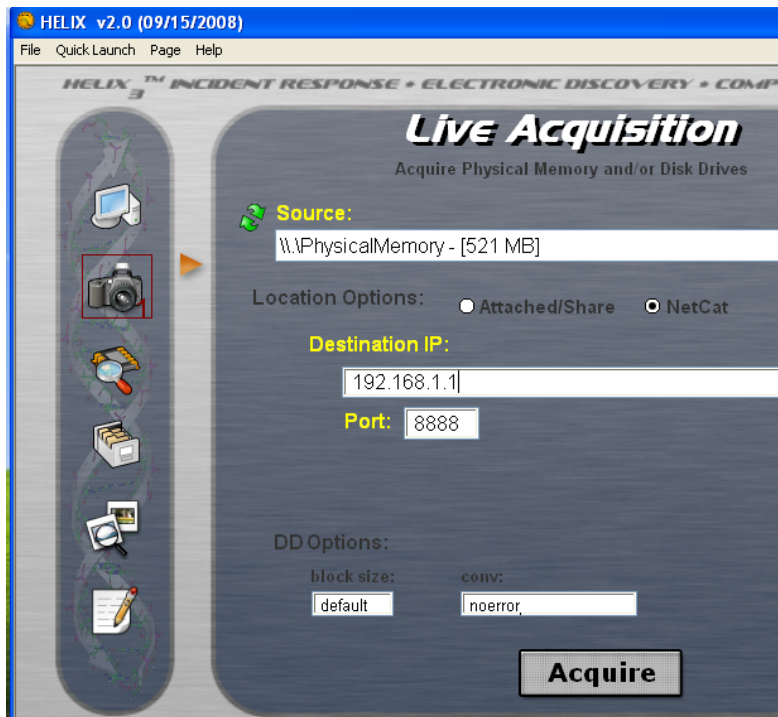


Figure 4.8: Prepare suspects HVD for acquisition.

This acquired the HDD/memory of the suspects virtual desktop hosted on the hypervisor within the private cloud, By using a NetCat/e-fense Helix interface, the output was written to an image created on the collection machine. All the activity logs were also saved and documented as far as good practice is concerned.

The second collection technique involves the use of a vCenter Standalone converter; a management tool which is used to convert virtual and physical machines to VMware standardized virtual machines (I. VMware, 2010). The standalone converter is also used to convert between Type 1 and Type 2 hypervisor based virtual machines. Hence this tool was used to capture suspects running HVD to a desired location. In order to perform this procedure, the suspect's virtual desktop was suspended via VMware vSphere (management console). Suspending suspects HVD will freeze it at the current state, enabling the forensic team to identify actual running processes and potential information in the volatile memory during the analysis stage.

Once the virtual desktop was suspended, the vCenter Standalone converter was prepared to begin the copy process by selecting the suspect's HVD using its IP address (Figure 4.8). This process was run from the collection machine mentioned in technique one, which was configured to be on

the same domain as the suspect's HVD. If the company was using vCenter server to manage their private cloud, this process is as straightforward as selecting the suspect's HVD, right click and copy

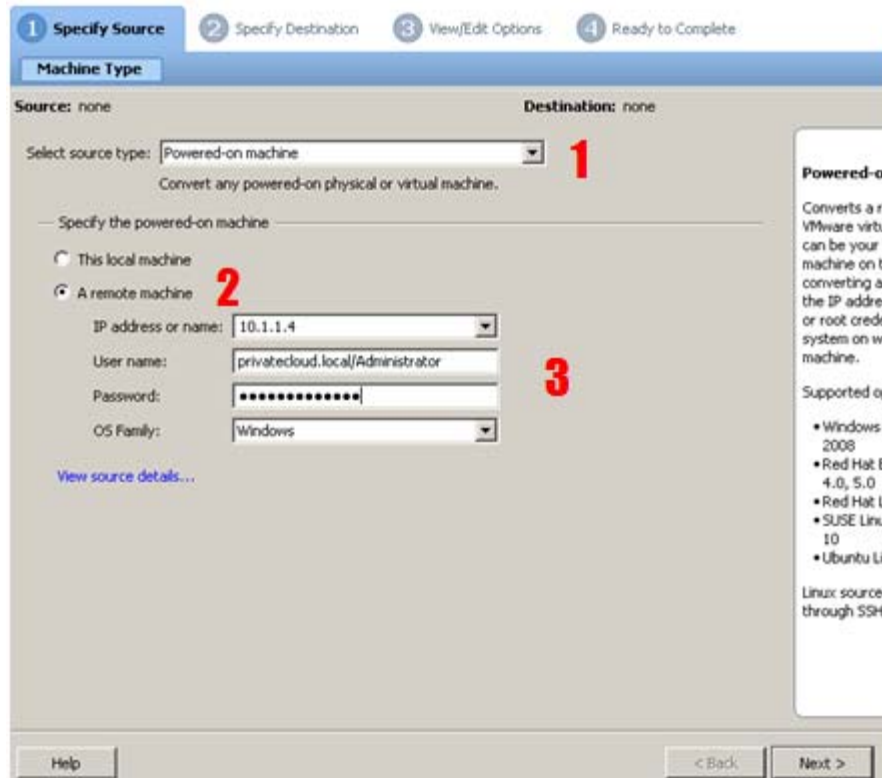


Figure 4.9: Preparing vCenter Standalone Converter

As part of the comprehensive collection phase along with acquiring the user environment, system logs were also collected from the following location in the VMware View log files:

Event Data base server

View Admin Console \ Monitoring \ Events - Filter by suspects VM - Export.

Connection Server and Security Server logs

<DriveLetter>:ProgramData\Application Data\VMware\VDM\logs

View Agent & Client Logs

Suspects VM\<DriveLetter>:\Documents and Settings\All

Users\Application Data\VMware\VDM\logs.

Suspects Laptop\C:\Documents and Settings\%username%\Local

Settings\Application Data\VMware\VDM\Logs

Figure 4.10: Locations of VMware View logs.

In this case scenario, both techniques were used to acquire the evidence to maximize the measurements made during the investigation analysis phase of the captured artifacts. However, both techniques have technical constraints making them less forensically sound. These constraints are discussed in next chapter.

4.3.4 Examination & Analysis

Examination and analysis was performed on the following evidence collected:-

- The image of the suspect's physical machine ("VPLaptop.dd").
- The image of the suspect's HVD (VPHVD.ovf, VPHVD.dd).
- The logs from the event database, connection server security server and view agent.

The image of suspect's laptop "VPLaptop.dd" was imported into forensic software as a raw image, and standard examination and analysis procedures were followed; the results show no traces of access to any inappropriate websites. Likewise, using techniques discussed in (Shavers, 2008) an image of the suspect's virtual desktop "VPHVD.ova" was imported into the VMware workstation for examination and analysis which recovered pornographic images in temporary internet files and also attempts to visit pornographic websites made during office hours were also found logged in the internet history.

Although it is possible that the suspect's VM may have been logged into by another user in the network, the View Client logs found on the suspect's physical machine and event logs on the

server show the time stamps of the user's login/log off times. These were found to match the timestamps of internet history and the objectionable images written to disk, as illustrated in Figure 4.9 and 4.10. This strongly suggests that the user who was engaged in objectionable activity on the targeted VM was logged in via the VP's laptop.









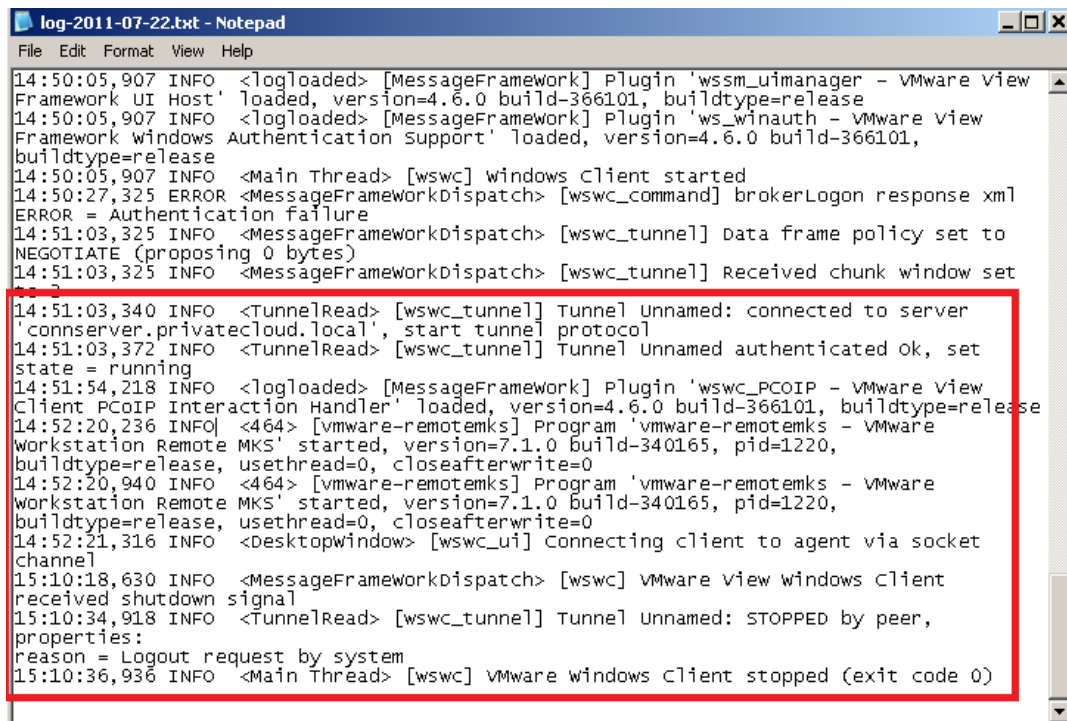
User	Severity	Time	Module	Message
PRIVATECLOUD\vice.president	 Audit success	7/22/11 3:10:39	Connection Server	User PRIVATECLOUD\vice.president has logged out
PRIVATECLOUD\vice.president	 Info	7/22/11 2:52:43	Agent	User PRIVATECLOUD\vice.president has logged in to a new session on
PRIVATECLOUD\vice.president	 Info	7/22/11 2:51:46	Agent	The agent running on machine XPVM has accepted an allocated session for user 
PRIVATECLOUD\vice.president	 Info	7/22/11 2:51:29	Connection Server	User PRIVATECLOUD\vice.president requested Pool Scenario_1, 
PRIVATECLOUD\vice.president	 Info	7/22/11 2:51:29	Connection Server	User PRIVATECLOUD\vice.president requested Pool Scenario_1
PRIVATECLOUD\vice.president	 Audit success	7/22/11 2:51:08	Connection Server	User PRIVATECLOUD\vice.president has logged in

Figure 4.11: Event Monitor Log

The snapshot above shows two logged events that suggests the vice president was logged into his virtual desktop for approximately 14 minutes 47 seconds.



```
log-2011-07-22.txt - Notepad
File Edit Format View Help
14:50:05,907 INFO <logloaded> [MessageFramework] Plugin 'wssm_uiframework - VMware view
Framework UI Host' loaded, version=4.6.0 build-366101, buildtype=release
14:50:05,907 INFO <logloaded> [MessageFramework] Plugin 'ws_winauth - VMware view
Framework windows Authentication Support' loaded, version=4.6.0 build-366101,
buildtype=release
14:50:05,907 INFO <Main Thread> [wswc] windows Client started
14:50:27,325 ERROR <MessageFrameworkDispatch> [wswc_command] brokerLogon response xml
ERROR = Authentication failure
14:51:03,325 INFO <MessageFrameworkDispatch> [wswc_tunnel] Data frame policy set to
NEGOTIATE (proposing 0 bytes)
14:51:03,325 INFO <MessageFrameworkDispatch> [wswc_tunnel] Received chunk window set
14:51:03,340 INFO <TunnelRead> [wswc_tunnel] Tunnel Unnamed: connected to server
'connserver.privatecloud.local', start tunnel protocol
14:51:03,372 INFO <TunnelRead> [wswc_tunnel] Tunnel Unnamed authenticated ok, set
state = running
14:51:54,218 INFO <logloaded> [MessageFramework] Plugin 'wswc_PCOIP - VMware view
Client PCoIP Interaction Handler' loaded, version=4.6.0 build-366101, buildtype=release
14:52:20,236 INFO <464> [vmware-remotemks] Program 'vmware-remotemks - VMware
Workstation Remote MKS' started, version=7.1.0 build-340165, pid=1220,
buildtype=release, usethread=0, closeafterwrite=0
14:52:20,940 INFO <464> [vmware-remotemks] Program 'vmware-remotemks - VMware
Workstation Remote MKS' started, version=7.1.0 build-340165, pid=1220,
buildtype=release, usethread=0, closeafterwrite=0
14:52:21,316 INFO <Desktopwindow> [wswc_ui] Connecting client to agent via socket
channel
15:10:18,630 INFO <MessageFrameworkDispatch> [wswc] VMware view windows Client
received shutdown signal
15:10:34,918 INFO <TunnelRead> [wswc_tunnel] Tunnel Unnamed: STOPPED by peer,
properties:
reason = Logout request by system
15:10:36,936 INFO <Main Thread> [wswc] VMware windows Client stopped (exit code 0)
```

Figure 4.12: View Client Log

The snapshot above proves that the VP used his laptop to join his virtual desktop as the view client logs obtained from the VP's laptop and the time of the event in this log correlates with the events in the event monitor logs.

Investigation Process	
Preparation	✓
Search	✓
Recognition	✓
Collection	✓
Investigation	
Examination	✓
Analysis	✓
Presentation	✓
Documentation	✓

Table 4.1: Scenario 1 Investigation Results

As a result, this scenario was successfully investigated using current digital forensic procedures. Though it wasn't straightforward, the forensic team were successfully able to investigate because the user's environment was known and secondly, the suspect's virtual machine was configured with persistent disk mode, which left the internet temporary files behind. Table 4.1 shows that all standard investigation techniques can be followed.

4.4 SCENARIO 2 - NON-PERSISTENT VIRTUAL ENVIRONMENT.

Perform a forensically sound investigation and find evidence concerning the leaked company data. Furthermore, investigate the incident so strict policies can be enforced to avoid any future leakage.

4.4.1 Preparations - General Findings

- Object audit logs (Windows security auditing) show the link between domain user and accessed data.
- The computer name belonged to a virtual desktop available for visitors.
- Company's virtual infrastructure consists of 50 active virtual machines, out of which 5 are assigned to the guests and visitor pool.
- The guest and visitor pool is part of a floating pool, meaning a user is not guaranteed to get the same virtual desktop every time they log in.
- HVDs within the guests/visitor pools are set to use a non-persistent disk, meaning anything written on the disk will be wiped off once restarted, this is to avoid junk from web surfing and corruption to system files caused by malware/virus .
- Virtual desktops are provided to visitors via a low-powered machine which has no USB/E-SATA socket for security purposes.
- Event reported on 25/07/2011 at 3:10pm, forensic investigation began on 25/07/2011 at 5:00pm.

4.4.2 Search and Recognition

After verifying the audit logs (Figure 4.13) produced by the file server it was verified that the suspect had logged into the system from the guest machine near the reception, therefore the evidence was potentially present within the physical machine, the virtual desktop itself or the security logs generated by the HVD infrastructure and Windows server. Since the machine used by the suspect had no way of being connected to external storage devices it is highly likely he/she may have stored it in the clouds or emailed themselves a copy of the stolen files.

4.4.3 Collection

The physical machine was found in a state where it was powered on and logged by a guest user. Although the physical machine was not directly suspected of having any potential evidence, it was still imaged using standard live acquisition procedures. In an attempt to identify the line of attack used to leak the sensitive data, security logs from VMware event database, view client, view agent and Windows server 2008 audits were collected. Security logs available by default were monitoring the attributes of the "Sensitive Data" folder. Logged events included failed and successful attempts to Read/Write/Delete attribute to the folder and files within. A custom view of the logs was created according to the targeted username, the audited object and exported for further analysis.

As the guest HVD belonged to a floating pool of five virtual desktops, it was critical to identify the exact VM used by the suspect before acquisition could take place. However this was easily traced by correlating information such as account and domain name between object audit logs and event database featured in VMware view version 4.6 and Windows server 2008.

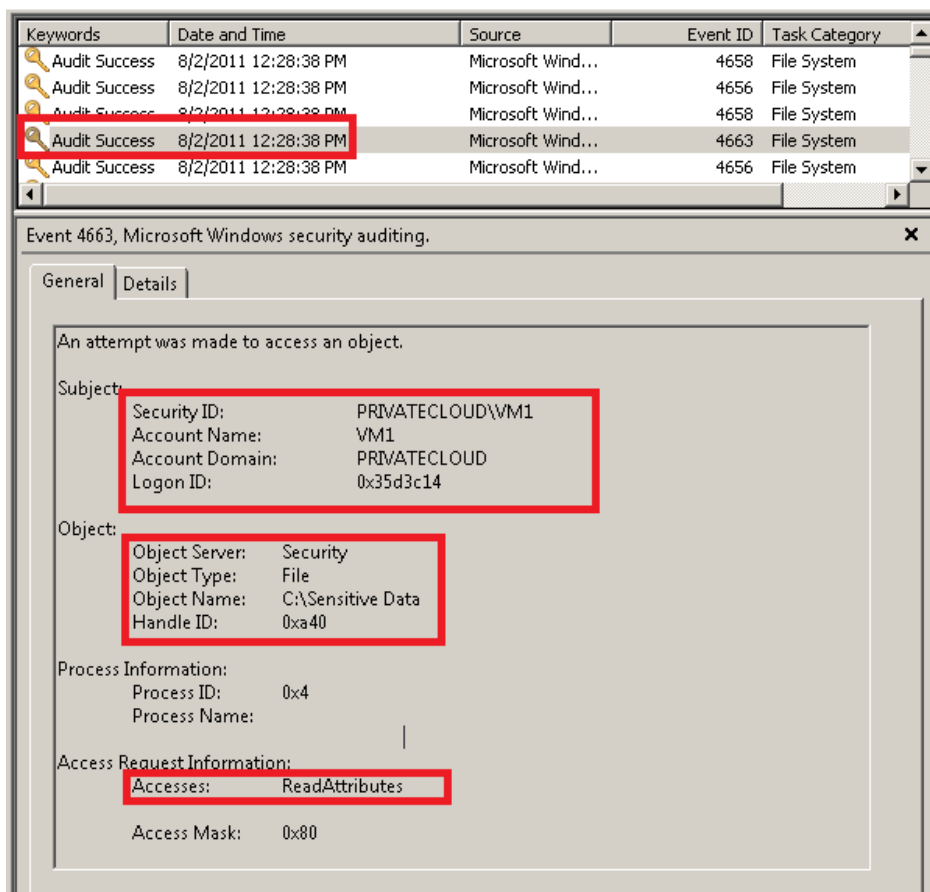


Figure 4.13: Security Logs showing accessed object

The identified HVD used by the suspect was found in standby mode waiting to be assigned. The only possible way to acquire the virtual desktop in this state was to suspend it and copy it over to the collection machine using vCenter converter; as described in the previous scenario. As a result, vCenter standalone converter was used to capture and copy the virtual desktop to the collection machine.

4.4.4 Examination & Analysis:

Authenticity of the collected data needed to be examined by creating checksum. This was possible for the image created of the physical machine the suspect used to access the virtual desktop. However checksum was not able to be used on the virtual desktop as it was copied over in a suspended state. As for the logs, since custom view was created by the forensic team during the collection phase, no further examination was required.

As part of the analysis stage, the image of the physical machine was imported into forensic software for further examination, mainly to find any traces of the suspect attempting to transfer any sensitive data between the virtual and physical desktop. Although no such traces were found, except the View Client logs found in "C:\Documents and Settings\All Users\Application Data\VMware\VDM\logs" show, the suspect logged into a virtual desktop at the same time as shown in the security logs.

At this stage, by examining object audit logs, VMware View client logs and VMware event logs, it can be determined that the suspect attempted to access sensitive files present in the "Sensitive Data" folder via a virtual desktop. To determine whether any information was leaked, i.e. copied or transferred somewhere, and by who can be traced by analyzing the hosted virtual machine. By following procedures in (Shavers, 2008), a copy of the acquired virtual machine was emulated in VMware Workstation 6.5. This resumed the virtual machine and enabled a live view of the suspect's environment. The objective was to find any traces that lead to the suspect trying to upload sensitive files on the internet. But no relevant traces were present. This confirms that the suspect may have restarted the virtual desktop, wiping all the potential traces as the HVDs in the guest pool were configured with non-persistent disk mode, as shown in figure 4.14. This makes the analysis stage very challenging as the key evidence has possibly been deleted.

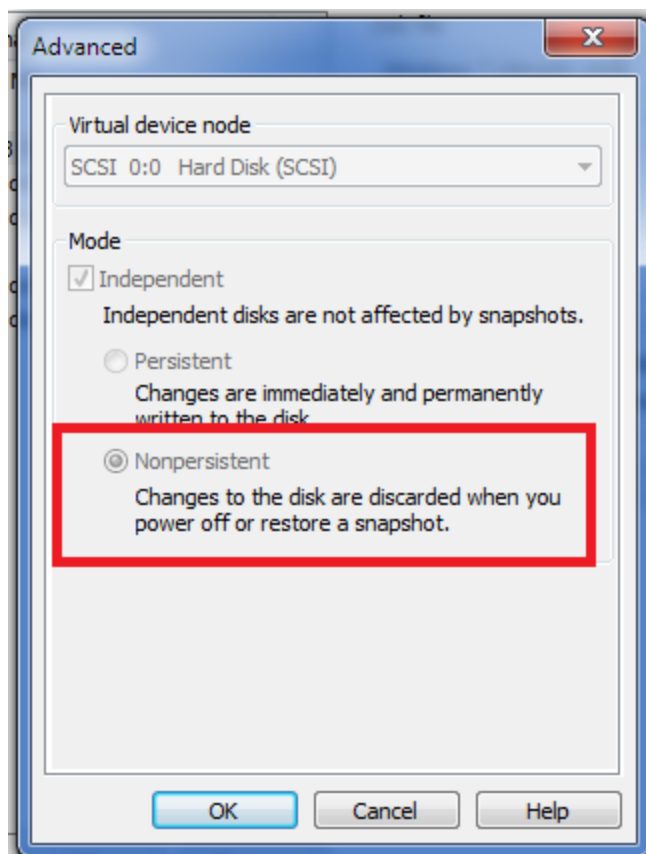
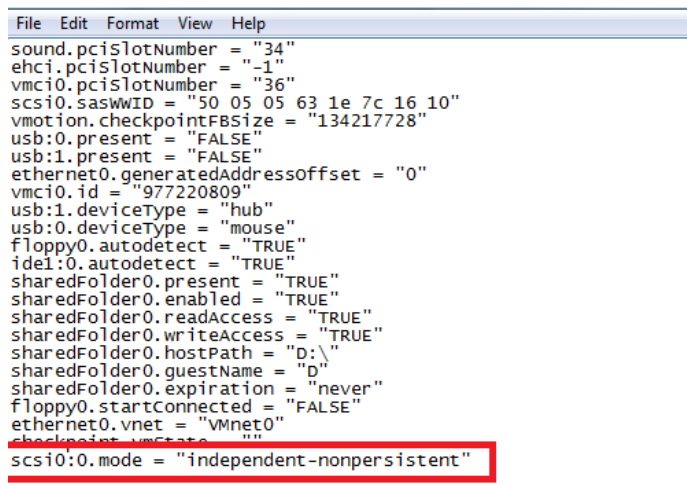


Figure 4.14: Disk Mode Configuration

Further attempts can be made by checking to see if any snapshots were created at the time of the breach. According to VMware knowledgebase: 1015180 (*Understanding virtual machine snapshots in VMware ESX*, 2010) snapshots can be taken while the virtual machine is powered-on, powered-off or suspended, preserving data and the state of the virtual machine providing its specified as a system policy. Preserved data includes physical memory and the network device cache. Another component of the snapshot that cannot be overlooked is a redo file. Redo files hold all the changes that have occurred on a virtual machine since last booted, this is redirected to a redo file instead of the virtual machine's disk because redo files are only used for non-persistent, undoable and appended disk modes.

Examining virtual machine files may also assist in uncovering further leads. Various files are generated when a new virtual machine is created, amongst the various files are some that are relevant to a forensic analysis of a virtual machine, like (.vmx), (.vmdk) and (.vmem). The (.vmx) files can be used to determine the overall configuration of a virtual machine like size, connection method and the encoding method of the virtual disk. It also has the configuration of

devices connected to the virtual machine. In this scenario checking this file can confirm whether the suspect tried to manipulate any configurations. This can be done by comparing it to the (.vmx) file belonging to non-compromised virtual machines within the pool. The (.vmdk) Virtual Machine Disk is the actual disk of the virtual machine, there can be several 2 GB chunks of this file. Although this file may potentially contain most of the evidence, it was found to be useless in this scenario because of the non-persistent configuration as figure 4.15 shows.



```
File Edit Format View Help
sound.pciSlotNumber = "34"
ehci.pciSlotNumber = "-1"
vmci0.pciSlotNumber = "36"
scsi0.saswwID = "50 05 05 63 1e 7c 16 10"
vmotion.checkpointFBSize = "134217728"
usb:0.present = "FALSE"
usb:1.present = "FALSE"
ethernet0.generatedAddressoffset = "0"
vmci0.id = "977220809"
usb:1.deviceType = "hub"
usb:0.deviceType = "mouse"
floppy0.autodetect = "TRUE"
ide1:0.autodetect = "TRUE"
sharedFolder0.present = "TRUE"
sharedFolder0.enabled = "TRUE"
sharedFolder0.readAccess = "TRUE"
sharedFolder0.writeAccess = "TRUE"
sharedFolder0.hostPath = "D:\\"
sharedFolder0.guestName = "D"
sharedFolder0.expiration = "never"
floppy0.startConnected = "FALSE"
ethernet0.vnet = "VMnet0"
checkpoint.vmState = ""
scsi0:0.mode = "independent-nonpersistent"
```

Figure 4.15: (.vmex) file

The (.vmdk) file can usually be mounted and explored using popular disk mounting tools like Mount Image Pro, but if the virtual machine is found in any state other than powered off, the disk will be compressed and encrypted by default, making it almost impossible to work with. The (.vmem) file is the extension of a snapshot file, it primarily acts as the virtual memory of a virtual machine, but it is only generated while the machine is running or crashed. .vmem files can be analyzed using memory analysis tools, e.g. SANS SIFT Workstation or Python based volatility framework. The virtual machine in this scenario was found in power on state, and Figure 4.15 proves that virtual machine was configured for non-persistent disk mode, even though (.vmem) file was available; it would not contain any user critical information.

In the examination and analysis phase of this case scenario, a snapshot, redo or other system file belonging to the suspect's virtual machine were not present. If either of the files were present, determining who and where the data was transported could have been possibly traced. The browser cache could have held the traces as to by whom, when and where the sensitive data was transported. However, knowing the virtual machine discarded all the user data written to disk

halts the investigation and no further conclusion can be drawn other than that someone accessed sensitive information but whether it was simply viewed, copied or transferred to another network cannot be determined.

In cases where persistent or non-persistent disk modes are used, snapshots are not taken as part of default policy (VMware, Inc, 2009b). Saying that, in an HVD environment; redo files can grow in gigabytes affecting the performance of the whole virtual desktop network (Laverick, 2005). From the forensic point of view having snapshots taken whenever there are changes in the state of a machine would make investigation less complex. Crime can take place anywhere, and it is often committed in the least obvious location. In this scenario the suspect may have chosen a guest machine knowing it is the least probed, as it is only a guest machine, and any user populated data will not be backed up at the end of the day. It has been said before that auditing is only as good as you have prepared for. In cases similar to this case scenario, investigators may only be able to proceed if all the physical and virtual desktops are auditable.

Investigation Process	
Preparation	
Search	✓
Recognition	✓
Collection	X
Investigation	
Examination	X
Analysis	X
Presentation	
Documentation	X

Table 4.2: Scenario 2 Investigation Results

Enforcing policies such as setting user entitlements for virtual desktop pools so only staff members can log into a staff pool, performing backups for data on virtual desktops especially for virtual machines configured for independent disk mode and also restricting users from restarting or shutting down virtual desktops, will insure an HVD infrastructure in case of digital

forensic investigation. As Table 4.2 above shows, traditional forensic methods were not adequate in investigating this kind of case scenario completely. Using the current digital forensic procedures, the investigation process could only progress up to search and recognition.

4.5 SCENARIO 3 - INVESTIGATING CRIMES IN A MULTI-TENANT ENVIRONMENT

Perform a forensically sound investigation on the school's multi-tenant - hosted virtual desktop environment, to locate and extract relevant evidence in solving this case.

4.5.1 Preparations - General Findings

- ISP has leased 3 IP addresses to the school which have been further distributed so every computer can be uniquely identified on the school's network.
- School has a hosted virtual desktop infrastructure in place.
- All the virtual machines are distributed among three virtual servers i.e. teachers, students and staff. Each virtual server is using 1 of 3 ISP provided IPs.
- All virtual desktops are part of a dedicated pool and every virtual machine is configured for persistent disk mode.
- Schools internet filtering system is very basic and shows it failed to detect attempts made to connect a network containing objectionable material.
- School had no WIFI to restrict students using smart devices during school hours

The potential locations of the evidence were either in the virtual desktop or in backups which are performed every day at 12:00 midnight. Possible traces that may show users attempts to connect to the offshore network can be found in the browser cache, internet history or DNS cache. But in this scenario, the forensic team only has reports provided by the ISP, which include address of the offshore network and timestamps confirming the activity occurred via the IP range belonging to the school. With this information, it is hard to narrow down to a user, as this could potentially be any user or group of users within the school since the IP range is using NAT to accommodate every computer in the campus. It is also possible that a rogue user may be accessing the school's network to gain access to the external network containing objectionable material.

Possible ways to start the forensic procedure:-

4.5.1.1 Forensic Technique 1

Create a list of keywords from information provided by the ISP and perform a search using the keywords on every virtual server i.e. teachers, staff, student, assuming the search will go through the file system of every virtual machine present on the server.

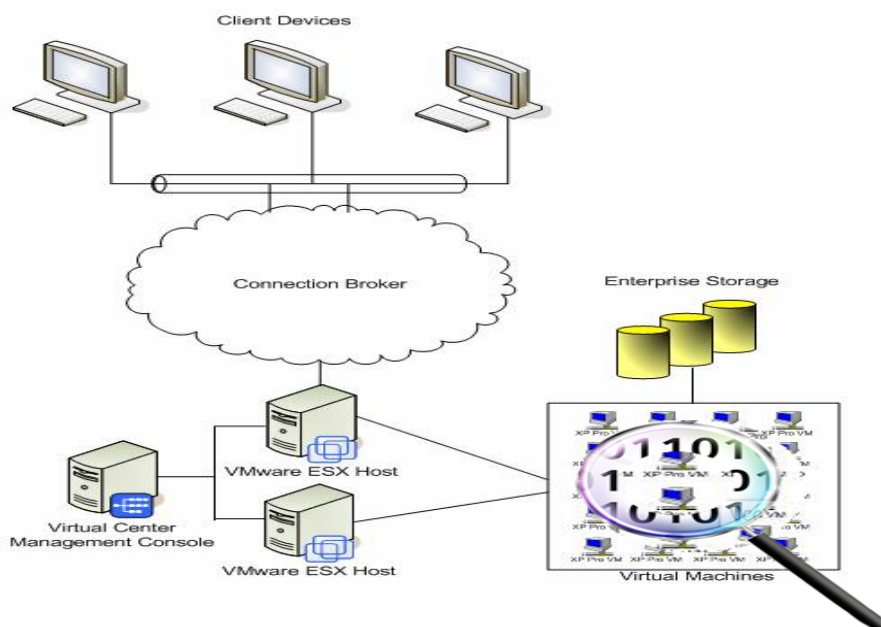


Figure 4.16: Blind search on ESX host

Possible results can be internet temp files or visited URL's. This may mean taking the servers offline but the biggest challenge is to perform a search on the entire server containing numerous virtual machines. Not only is this time consuming but also the ability to perform keyword search in such a manner is very limited unless supported by third party tools. Currently none of the forensic suites are able to read, analyze or perform such search on ESX file system (VMFS) (Haletky, 2008). This technique also violates best practice as the search will be performed on a live system. The advantage of using this technique is that it avoids intrusion into a user's personal space, as the search will only show results based on the information provided in response to the key words.

4.5.1.2 Forensic Technique 2

Second method could be seizing every virtual server and imaging them so they can be imported into appropriate forensic software for further analysis. The time taken to image every virtual server will result in downtime of school work as all the academic applications available to users are via their virtual desktops. More importantly, due to the nature of the analysis stage the likelihood of browsing through other users' personal space is very high, as multiple tenants are hosted on the same server.

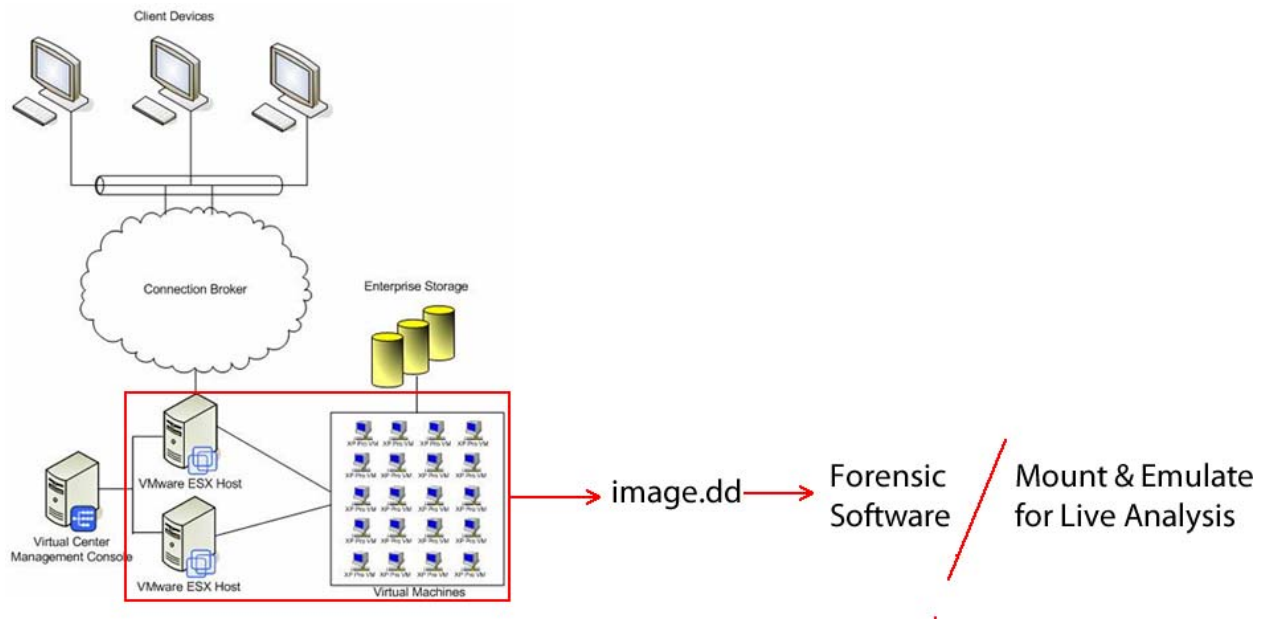


Figure 4.17: Imaging ESX

4.5.1.3 Forensic Technique 3

Attempt to identify the user rather than looking for traces i.e. examine the record of previously occurring incidents involving the school's computers and internet. This depends on the school's records, and if such campus incidents are logged it will help to narrow it down to a number of users who may be involved.

4.5.1.4 Forensic Technique 4

It is also possible to audit firewall logs. Depending on the type of firewall, it is likely the logs show visited URLs with timestamps and user/s.

This scenario is set in a school context. In reality, a school is likely to have an internet traffic monitoring system that will detect such activity. But if a similar incident occurs in an organization, the ability to identify a single user can be very complex. An Internet user does not have a unique identity, in contrast to an internal network. The primary concern of this scenario is respecting the privacy of other users' personal space during the investigation. In cases where techniques 1, 2, 3 are not practical, investigators are left with option two, where data of all users is forcefully exposed rather than a specific users' data. Trawling through the resulting data can be very difficult without knowing what to look for. This clearly shows typical forensic procedures fail on multi-tenant environments because the industry lacks tools and procedures to segregate forensic data from multiple tenants (Ruan & Carthy, 2011) Table 4.3 summarizes the investigation process.

Investigation Process	
Preparation	
Search	X
Recognition	X
Collection	X
Investigation	
Examination	X
Analysis	X
Presentation	
Documentation	X

Table 4.3: Scenario 3 Investigation Results

4.6 CONCLUSION

As seen in this chapter, by simulating a working HVD model, three quasi-experimental case scenarios were designed, developed and implemented. This permitted us to observe the adequacy of current digital forensic procedures in such cases.

	Persistent Disk Mode	Non-Persistent Disk Mode	Multi-Tenant Hosting
Scenario 1	●		○
Scenario 2		●	○
Scenario 3			●
● - Targeted issue in the scenario ○ - Issue present in every scenario			

Table 4.4: Overview of scenarios

As overviewed in Table 4.4, each scenario was designed to target a specific security issue which may also create concerns for digital forensic investigation in future. The aim was to exercise traditional forensic procedures containing the key elements of preparation, investigation and presentation to locate and extract evidence within a hosted desktop virtualization infrastructure. The success of every scenario was recorded in an "Investigation Process" table at the end of every scenario. The findings show that locating and extracting evidence from Scenario 1 was possible but not straight forward. In scenario 2 complexity increased and traditional forensic procedures were only useful to locate the evidence. Scenario 3 proved to be too complex for current digital procedures to handle easily, with neither location nor extraction of evidence being possible.

In the next chapter, discussion of the results of the three quasi-experimental case scenarios and the overall methodology employed for this research will be outlined.

Chapter 5- Research Discussion & Recommendation

5.0 Introduction

In this study to determine the adequacy of current digital forensic procedures on recently introduced cloud computing technologies such as HVD, FIDM was used to implement three quasi-experimental case scenarios on a developed simulation of HVD in a private cloud deployment. The three case scenarios were simulated and findings were generated as of Chapter 4.

The aim of this chapter is to summarize and discuss findings from the previous chapter and also provide recommendations based on the practical implications of this thesis. In Section 5.1, findings of the experiments outlined in the previous chapter will be discussed in relation to the adequacy of methods and techniques used while investigating each case scenario. In Section 5.2, possible recommendations and best practices will be outlined for forensic practitioners and organizations wanting to adopt HVD. Then, Section 5.3 will introduce possible unseen scenarios the field of digital forensics could face during an investigation which are to some extent, technically related to the case scenarios used in this study. In section 5.4, the research methodology developed for this thesis (FIDM) will be revisited to gather feedback and suggest changes for individuals wanting to adapt FIDM for research of a similar nature. Finally in Section 5.5, the chapter will conclude by summarizing this chapter and briefly outlining the contents of the subsequent part of this thesis.

5.1 DISCUSSION OF FINDINGS

This section aims to discuss the outcomes of the three case scenarios in regards to the success of the current digital forensic procedures applied.

5.1.1 Case Scenario 1

The purpose of this case scenario was to test the adequacy of current digital forensic procedures and tools to locate and extract evidence from a pool of virtual desktops, hosted on the company's in-house private cloud, where the user environment was configured with persistent disk mode and dedicatedly assigned to a user from the pool of VM's.

In an attempt to acquire the evidence, live acquisition procedures were performed on the physical machine used to access the virtual desktop and the virtual desktop itself. It was fairly straightforward to image the physical machine but the virtual instance of the desktop had to be acquired in an unusual manner, both techniques had technical constraints making them less forensically sound. As illustrated in Section 4.3 of chapter 4, the first acquisition technique was successfully able to acquire the virtual HDD and memory of the suspects HVD using an e-fense CD. In the background this procedure was utilizing dd tools to capture and netCat to transfer the images to a collection machine. While this worked seamlessly in this scenario, in a network where the function of netCat is restricted, this procedure cannot be employed. The second technique used to acquire evidence relied on using vCenter standalone converter. This converter has the ability to copy the entire virtual desktop to a desired location. Though using this procedure; the suspect's HVD was copied over to a collection machine, but this was less forensically sound method. Firstly, vCenter standalone converter can only communicate with the virtual machine if the virtual machine has "converter standalone agent" software installed. The agent nevertheless can be installed later but may alter system files at the same time. Secondly, the standalone converter cannot be used if the virtual machine is running; however it can function if the virtual machine is suspended/paused. Even though this mode is supposed to save the current state of a virtual machine there is no guarantee that all system files/configurations remained unchanged. Finally, the second acquisition technique cannot generate hash values at given stage. Hence there is no assurance of the originality of the evidence.

An image of the physical machine was analyzed and resulted in no evidence, during the analysis of the acquired images of the HVD; VPHVD.dd (generated by e-fence Helix) was recognized and successfully analyzed to locate the potential evidence. But VPHVD.ovf (generated by standalone converter) could not be opened in Encase for analysis. Even though there was the option of converting VPHVD.ovf to VPHVD.vmdk so it could be recognized by Encase, the live analysis techniques described in (Shavers, 2008) were favored, due to the level of sophistication of this scenario.

5.1.2 Case Scenario 2

The purpose of the second scenario was to test the adequacy of current digital forensic procedures and tools, while locating and extracting evidence from a non-persistent user environment.

It was suspected that an existing employee's credentials were used to steal company's valuable data, and so digital forensic services were called upon to investigate this matter in concern to data theft. Usual forensic methodology was applied to investigate the individuals responsible for stealing the company's valuable data. One approach used in an attempt of identify the individuals was to trace the machine used to steal the data and find out where it was transferred afterwards. With the help of "Windows security auditing" (object) logs, link between the stolen credentials and the valuable data was determined; this also revealed the physical machine and the HVD as MAC times (last access time stamp) for the files/folder were viewed. Furthermore, the physical machine was analyzed but there were no traces of valuable data present. It was determined that the suspect made use of the HVD assigned for the company's guests. Further examination revealed the suspect had re-started his/her HVD in order to wipe any traces left behind.

The forensic investigation could not progress beyond this point because the HVD used by the suspect was part of the guest pool, configured with non-persistent disk mode. There were currently no forensic procedures at hand to investigate systems configured in such manner. While existing methods (audit log inspection and MAC times) were partially helpful in determining the "Where" and "What" i.e. the physical/Virtual desktop used and the stolen data, these are unreliable and volatile because this data is often treated as metadata that is overwritten by newer activity, to keep down increasing overheads in a typical distributed system.

5.1.3 Case Scenario 3

The purpose of the third scenario was to test the adequacy of current digital forensic procedures and tools while investigating crimes occurring in a multi-tenant environment. In Scenario three, the school's internet filtering system failed to detect an attempt to access an inappropriate offshore network containing child pornography, as reported by the school's internet service provider. And so, digital forensic investigation procedures were required to locate and extract evidence from a multi-tenant - HVD infrastructure, however the case was fairly complex to begin for two reasons. Firstly, very limited information about the incident was provided by the ISP. The only information provided was, the public IP used, the time and date of the attempt and the blacklisted address of the offshore network. Secondly, the three IP addresses leased to the school were distributed (NAT) to accommodate every virtual desktop in the infrastructure. Though the ISP reported the IP involved in the incident, it was not possible to translate back to a single HVD internally using that public IP. Nevertheless, four techniques based on traditional forensic procedures were suggested to acquire and analyze evidence from the schools multi-tenant - HVD infrastructure. As discussed in chapter 4, section 4.5, the first technique involved creating a keyword list based on the information provided and performing a blind keyword search on the targeted virtual server, containing all the VM's while they are running. While the technique assures no downtime and maintains the privacy of other tenants it is not forensically sound and only a theoretical idea applied to HVD infrastructure. The second procedure was proposed was to image the entire virtual server using ddtool and input the results (image.dd) into forensic software for further analysis. Though this technique sounded like a typical imaging and analysis process, it wasn't very forensically sound when applied to this scenario. Technically, a image created by dd tool containing an ESX server on a VMFS partition cannot be recognized by major forensic software as yet. This left no other option but to mount the image and emulate the infrastructure to perform a live analysis procedure, bringing static data to life. However this means the privacy of other tenants is not maintained as live analysis procedures involve manually interacting with user data/environments to find answers. Thus the likelihood of browsing through other user's personal space is very high, as multiple tenants are hosted on the same server. Techniques three and four purely relied on the availability of system logs from routers DHCP tables, tcpdumps, firewalls or squid logs if proxy network is implemented. Not

knowing what to look for was one issue and the lack of ability of the current digital forensic methods and tools to investigate in a multi-tenant environment was another.

5.2 Unforeseen Scenarios

The effectiveness of traditional forensic procedures and tools greatly depended on the scenario's architecture. It is possible to say if scenarios were designed to favor digital forensics, i.e. forensically ready; the procedures would have been more effective. The scenario designs were mostly guided using vendors' manuals, but as desktop virtualization and consolidation become common, organizations are likely to implement architectures to suit their needs. Best practices may not always be followed, as was discovered in a survey performed while the quarterly security and compliance training by employees of major organizations (West, 2011). Nevertheless, this will introduce unforeseen HVD cases to the field of digital forensics. As table 2 illustrates, forensic investigators will have to perform forensically sound investigation on unforeseen HVD infrastructure.

	Persistent Disk Mode	Non-Persistent Disk Mode	Pool Assignment	Multi-Tenant Hosting
Scenario 1				
Scenario 2			Dedicated	
Scenario 3			Floating	

Table 5.1: Crimes in concealed case scenarios

As shown in Table 5.1, Scenario 1 contains a mixture of virtual machines with persistent or non-persistent disk modes, which are assigned to users manually. Investigators may come across a similar infrastructure, where the location of the user and data is easy to discover, although there is no guarantee, the required data is available. Likewise, in cases like Scenarios 2 and 3, where the HVD infrastructure consists of multiple virtual machines configured with a mixture of persistent and non-persistent disk modes belonging to a dedicated/floating pool, investigators may find it challenging to identify the suspect on the network. If the suspect is not identified from among the other tenants, the investigation can head towards multi-tenancy issues threatening other tenants' privacy. At the same time investigators may need to restrict users restarting or shutting down their virtual machine, as a result, data may not be present by the time the tenant is identified. A Mixture of aspects within the HVD infrastructure identified above are

likely to be seen in the future. The findings of the experiments and potential unforeseen scenarios indicate clearly that practicing current digital forensic procedures may not be sufficient to carry out a forensically sound investigation within the HVD environment.

5.3 RECOMMENDATIONS AND BEST PRACTICES

The purpose of this section is to outline possible recommendations and best practices to provide guidance for organizations using or planning to implement desktop virtualization technology, so their IT infrastructure can be planned accordingly.

The scenarios discussed in this thesis were designed to demonstrate various aspects of the hosted virtual desktop infrastructure, aspects that question technology from the security and digital forensic point of view. The three scenarios were quasi-experimental scenarios, while there may not be any solid non-fictional cases in existence yet, it's only a matter of time until the desktop virtualization technology is widely adopted throughout the industry, and these issues will concern digital forensics. To reduce the potential security risks identified in this study, forensic readiness needs to be adopted within HVD infrastructure.

In the first scenario, where a crime was being investigated in a persistent environment the investigation was able to succeed even though the methods used to investigate were not entirely forensically sound. The success of this scenario was largely dependent on two factors; the evidence was located in a persistent environment and the events within the infrastructure were being logged. Due to the persistent environment, the evidence remained intact for investigation. Being able to verify the suspect's activity was possible by associating the event logs generated by the VMware view's event database. As stated in Chapter 3, an event database is an infrastructure level logging component, responsible for recording events from end-user actions (logging in and starting a desktop session), administrative actions (adding entitlements and creating desktop pools) and general alerts caused by system failures or errors. In the VMware view event database, activities were only logged for a maximum period of three months, as per the default options available in the monitoring tab. NIST recommends "infrastructure-level administrators need to configure log sources so that they capture the needed information in the desired format and locations, as well as retain the information for the appropriate period of time" (Kent & Souppaya, 2006, p. 5). Taking this advice into account, VMware view event database will only display

events up to three months but stores up to six month old events for external queries up to six months. In regards to retained information, the monitoring tab in view administration only shows basic types of events, but several other types of events can be displayed if an external query is run. See appendix A for a full list of events.

Virtual machine template design can also play a huge role in a digital forensic investigation therefore best practices should be considered. As discussed in Section 4.3.3 the ability to acquire an HVD using VMware standalone converter depended on the "converter standalone agent". The agent may or may not be installed by default on the suspect's VM, depending on the template used to create the VM in the first place. Although including the agent in the template is not part of the practice guide (Dodge, 2006), it is highly recommended for administrators and to ensure forensic readiness. Currently there are no official guidelines for designing VM templates, but an organization must integrate necessary pre configuration policies, settings and software agents, ensuring that HVDs within the organizations private infrastructure can be ready for a digital forensic investigation.

In the second scenario, data theft was made obvious by stating that an employee's login credentials were stolen and been the suspect gained access to steal the company's valuable data. The point was to highlight the consequences of poorly planned and configured HVD infrastructure. During the investigation, the HVD used by the suspect and the stolen data was identified with the help of server audit logs. But the stolen data could not be traced any further. The suspect was able to restart the HVD which caused it to refresh, deleting all the user data produced while he/she was logged in. This was the danger of a non-persistent environment and a lack of monitoring support.

As described in Chapter Two, Section 2.3.4, the non-persistent environment is highly vulnerable to digital forensic as it suggests anti-forensics, as seen in Scenario two. Therefore, if a group of users/employees require a non-persistent HVD for any given reason, it is recommended that such an environment is isolated from the main intranet and internet access, if not fully auditable and monitored. By fully auditable and monitored it is meant, creating an audit policy plan to collect and archive security logs across the network so an audit trail of users' interactions with the organization's data can be logged. Audit policies can be defined for objects (user/s, folders, and files) to log event information such as failed and successful attempts to authenticate an object,

changes to security policy, deletion and creation of objects. (2005) mentions that a combination of success and failure object audits is typically used during a forensic investigation. Furthermore, if an organization is using Microsoft Active Directory (AD), Group Policy Objects (GPO) can also be used to enforce restrictions on users' privileges in a HVD. This way, policies can be set to restrict users being able to log off, restart, shut down etc.

In case Scenario three, a school's internet traffic monitoring system failed to detect an attempt to access an offshore network containing inappropriate content. In reality, the likelihood of a school's internet monitoring system failing to detect such activity is much less. But the goal of this scenario was to highlight the difficulty of investigating crimes of such a nature in a multi-tenant environment, particularly if you want to narrow it down to the users and the corresponding HVDs. Although multi-tenant issues are mostly discussed in the public cloud context, private clouds also facing the same issue. Not so much from the compliance point of view, but more to do with the privacy of other tenants. Therefore, to ensure forensic readiness in a multi-tenant environment, data segregation techniques should be applied.

In a virtualized environment such as the HVD infrastructure, tenants (VM's) are organized on a common domain as they are hosted on a common hypervisor for the purpose of consolidation. For this reason digital forensics is challenged because tenants in the infrastructure share a common storage platform, application database (common tables) and system/infrastructure level logs. Potential methods of segregating data are to configure storage LUN's or user profile management. A storage LUN (logical unit number) is a virtual layer that partitions a physical disk/disk volume into smaller segments for the provision of tenants in the infrastructure; each segment having a target ID address with each partition being a unique LUN (Schulz, 2011). This would make it easier to uniquely identify data belonging to different tenants and offer better traceability assuming audit logs are in place.

Profile management solutions can also be considered as a data segregation method. Profile management can be employed to define separate security policies for each tenant, separate audit logging and organize data/personalized settings according to a user's profile. During a virtualization implementation in an organization, the profile management capability may vary depending on whether the organization adopts a third party application or decides to use the infrastructure itself (Davis, 2011).

5.4 FIDM FEEDBACK

The research methodology for this research was modeled through the formation of FIDM. This systematic research model was built on the basis of the scientific research process and action research methodology. Combining the two methodologies, three quasi-experimental case scenarios were simulated and tested using the scientific observational method. Upon using FIDM to complete the defined research, this section aims to provide operational feedback of the model by suggesting changes for future research.

The first suggested change to this model is to consider digital forensic methodology in the scope of each iteration because, currently, it is assumed that the researcher would be using some sort of forensic framework. As per the existing model, a hypothesis is generated, from the hypothesis case scenarios are generated and simulated; subsequent to simulation, a forensic framework should be confirmed before findings are produced. The inclusion of forensic methodology into FIDM will support the findings produced from the case scenarios and give the researcher an opportunity to define the type of forensic framework used to ensure it makes apparent whether or not the type of forensic framework reflected upon the final results.

If the first suggested change is adopted, the second suggested change to the FIDM is to add the ability to perform a sub-loop between the simulation phase and analysis. As the quasi-experimental case scenarios were forensically investigated, it was found that there are many methods to acquire and analyze evidence with certain strengths and weaknesses. The sub-loop can allow future researchers to either, loop until the best methods are used to produce the purest findings or loop until every possible method is applied and note their strengths and weaknesses, as it is possible a method is applicable to one crime scene while it may not be reasonable for another.

5.5 RESEARCH PROBLEM AND HYPOTHESIS

The main research problem understood after closely reviewing the literature and the existence of current studies in the field of cloud forensics was:

It is currently unknown whether existing forensic methods are adequate for investigating crimes occurred within Hosted virtual desktops, based in private clouds.

Subsequently, the null hypothesis below was also generated.

Current digital forensic procedures are adequate for investigations involving hosted virtual desktops based in private clouds.

To investigate the problem above, three quasi-experimental case scenarios were designed, simulated and scientifically observed within Chapters 3 and 4. Findings suggest that current digital forensic procedures and tools cannot cope with the dynamic and consolidated environment of cloud computing in general. The findings suggests the same for private cloud computing. However if an organization follows and adopts the best practices and recommendations, HVD in private clouds could possibly be forensically investigated in private clouds. Falsification of the null hypothesis has taken place as new digital forensic methods are found to be necessary to deal with the quasi-experimental scenarios introduced in this thesis.

5.6 CONCLUSION

This chapter gives a detailed discussion of the research findings for each quasi-experimental case scenario investigated in chapter four. The findings were restated and asserted with discussion on the successfulness of current digital forensic procedures while investigating possible crimes that could occur in a hosted virtual desktop environment hosted in the private cloud infrastructure of an organization. In subsequent to this discussion, possible unforeseen scenarios were produced to highlight the forensic implications of investigating cases if the current case scenarios were configured differently. Recommendations and best practices were also discussed in the context of security and implementing a forensically ready HVD infrastructure. Finally the main research problem was answered and the null hypothesis was invalidated, as it was found that new digital forensic techniques were needed to be developed to cope with private cloud computing.

The next chapter aims to conclude this thesis by summarizing the research conducted on this topic and also the major outcomes that were revealed. Chapter Six will also cover a section on the limitations encountered during the course of this thesis and offer ideas in regard to future research.

Chapter 6 - Conclusion

6.0 INTRODUCTION

In this thesis, Chapter One introduced the research topic and the motivations behind conducting this type of research. Cloud computing was introduced in the context of its application within organizations during recent times. Applications such as storage virtualization, server virtualization and desktop virtualization were briefly introduced. On the basis of statistical data and the success stories of the applications, the rapid growth and demand for cloud applications is on the rise. On the other hand, the depth of digital forensic tools and procedures in regards to cloud computing was unknown; hence it was important to know if any special forensic procedures or tools are in existence, or, if current digital forensic practices are adequate for investigating crimes in cloud computing.

In Chapter Two, literature surrounding the research area was chosen and reviewed which provided an opportunity to expand the understanding on the area of research. This was accomplished by substantially looking at the current states of the digital forensic field and the development of desktop computing from the past to the present. The most relevant findings in this area were also identified by reviewing existing studies in relation to desktop virtualization and digital forensic capability.

Chapter Three - methodology, is where the researcher's understanding of methodology was laid out and from which research methodologies and methods were reviewed, primarily to evaluate and select the best possible methodology for this thesis. To do so, the main elements of methodologies such as grounded theory, descriptive, case study, scientific research and action research were discussed. Moreover, a hypothesis and a problem statement were established followed by, a finalized research methodology and a working model called the "Forensic Iterative Development Model" (FIDM) was produced on the basis of scientific research elements and action research.

According to the working methodology, three quasi-experimental case scenarios were introduced. The case scenarios were geared towards three primary concerns regarding desktop

virtualization based in private cloud computing from the digital forensic point of view. The scenarios were later generated and produced results relevant to the research area.

The results of forensically investigating three case scenarios were outlined in Chapter Four. The outcomes of forensically investigating each case scenario were documented in the manner of a traditional digital forensic investigation framework i.e. preparation, investigation, presentation. The results were used to determine the success of current digital forensic procedures and tools in a HVD infrastructure based in private clouds.

Chapter Five discussed the findings of the three case scenario's in regards to the success of current digital forensic procedures applied, including the strength and weakness of each technique applied or proposed. Additionally, best practices and recommendations for organizations were outlined, which were mostly implementation practices from the security and digital forensic point of view. The working model of the methodology, FIDM, was re-visited and functional changes were suggested for future researchers. Finally the chapter was concluded by presenting answers to the problem statement and nullifying the hypothesis.

As the forensic iterative model suggests (Figure 3.3), once the three scenarios have been observed and the findings have been analyzed, researcher must exit by concluding and suggesting further studies. This, chapter aims to conclude the research by summarizing the content of every chapter, outline any research limitations and suggest further work.

6.1 LIMITATIONS

The purpose of this study was to explore the possibility of locating and extracting evidence from a HVD infrastructure using the current digital forensic procedures. This was performed using an experimental setup of a an HVD product provided by VMware called View v4.6. Although the test bench was setup and experiments were successfully conducted there were a few limitations identified, for this reason the outcomes may vary if these limitations are acknowledged.

As the experiments were concluded and results were produced, first obvious limitation noted was experimenting with HVD solutions from just one vendor, when the major players in this industry are VMware, Microsoft and Citrix. Although, statistically, most popular solutions in private or

hybrid clouds are VMware's virtual infrastructure and Microsoft's MED-V. Experimenting with solutions from different vendors, would have provided a wide variety of outcomes.

Secondly, the HVD infrastructure was fully simulated on a VMware workstation and, although the simulated model was designed using best practices as given in the vendor manuals the limitation was that it was not an industrial setup. For example, commonly used features like backup systems or roaming profiles were not exercised in the experiment; this was due to the limited functionality of the demo version. Even though this did not impact heavily on the outcomes, it would've introduced new issues or possibilities in regards to digital forensic investigations.

The fact that the scenarios were fictional was another limitation of this study. This was due to the lack of actual cases where an e-crime concerning a virtualized desktop environment has occurred. With this limitation in place, there was the inability to observe an HVD infrastructure from variety of aspects which are only likely to be seen in real cases.

6.2 FUTURE RESEARCH

The research conducted on this topic provided reasonable depth with regard to digital forensics and desktop virtualization. During the experiments, a number of aspects were noted for further research. From the tools point of view, developing a tool that will have the ability to acquire virtual machines from ESX in any state, and check the integrity of the virtual machine on the ESX server against the acquired image using md5 hash values is necessary. It would also be helpful to have a mapping tool that would map the HVD infrastructure with details like; number of users, virtual machines, the various pools, network sharing etc; as this would give an investigators better understanding of the what, where and how within an infrastructure and reduce time wasted during the investigation. Another important aspect that needs attention is the implementation of security policies within the HVD infrastructure. Additional research needs to be done on efficiency of the current security policies and requirements for any additional policies, particularly the audit log policies. It's often found that audit log policies are not always enforced on critical objects within an infrastructure, as logs can often prove to be a key when it comes to investigating distributed systems. Hence, having appropriate logs in place will make the infrastructure one step closer of being forensically ready. From a security point of view, in

order to make sure consistency is maintained, further research also needs to be conducted in standardizing the implementation of hosted virtual desktop infrastructure, the necessity of this is illustrated in Table 5.1. Unforeseen cases could prove to be very destructive as they start to appear in the future.

Therefore further research can be undertaken in standardizing the implementation of the hosted virtual desktop and developing specialized tools and procedures in parallel, making the infrastructure more auditable and forensically ready. Similarly tools and procedures to carry out efficient forensic investigation need to be developed.

6.3 CONCLUSION

In this research, by simulating a working HVD model various case scenarios were designed, developed and implemented. This permitted the researcher to observe the adequacy of current digital forensic procedures in such cases. The findings show that locating and extracting evidence in cases like scenario 1 was possible but not straightforward or completely forensically sound. Cases like in scenario 2 and 3 proved to be too complex for current digital procedures to handle easily, with neither location nor extraction of evidence being possible. This problem is not caused by the actual virtualization suite used, but is inherent in the simulated cloud architecture.

In other words, the differences between the outcomes of the three case scenarios are the result of the three independently-configured, private cloud computing HVD infrastructures. In the case of Scenario 1, the cloud computing model consists of VM's configured with a persistent virtual HDD, in the case of Scenario 2 the VM's are configured with a non-persistent virtual HDD and, finally, in Scenario 3, the cloud computing model is multi-tenanted. While these results are preliminary and more work is required to investigate the implications of multi-tenanted architectures and non-persistent environments for forensic IT procedures, it is clear that using a multi-tenanted architecture could have digital forensic consequences in regard to protecting the privacy of individuals not involved in any objectionable behavior. While the non-persistent environment also poses a risk of anti-forensic and presents the inability to retrieve user data if the virtual desktop has been found refreshed. In other words, it is possible that companies and organizations adopting multi-tenanted architectures may need to get the written agreement of

employees and users to access their private data in the scope of a digital forensic investigation even if they are not suspected of doing anything wrong. While, the non-persistent (stateless) environments may have advantages, yet it affects the availability of potential evidence. In conclusion, these experiments indicate that organizations considering the deployment of HVDs will need to review their digital forensic preparedness to ensure that their auditing and investigative procedures are placed on as sound a footing as in non-HVD environments.

References

- Agarwal, A., Gupta, M., Gupta, S., & Gupta, S. C. (2011). Systematic Digital Forensic Investigation Model. *International Journal of Computer Science and Security*, 5(1), 118-131.
- Avison, D. E., Lau, F., Myers, M. D., & Nielsen, P. A. (1999). Action research. *Communications of the ACM*, 42(1), 94-97. doi:10.1145/291469.291479
- Bakshi, K. (2011). Considerations for cloud data centers: Framework, architecture and adoption *IEEE Computer Society Washington*. Symposium conducted at the meeting of the Aerospace Conference, 2011 IEEE
- Bares, R. A. (2009). Hiding in a virtual world: Using unconventionally installed operating systems *IEEE Press Piscataway*. Symposium conducted at the meeting of the Intelligence and Security Informatics, Dallas, Texas, USA. Retrieved from http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5137326
doi:10.1109/ISI.2009.5137326
- Barrett, D., & Kipper, G. (2010). *Virtualization and Forensics*. Retrieved from <http://my.safaribooksonline.com/book/networking/forensic-analysis/9781597495578>
- Bernard, G. (2009). *Virtualization For Dummies*: John Wiley & Sons. Retrieved from <http://my.safaribooksonline.com/book/operating-systems-and-server-administration/virtualization/9780470148310>
- Broadhurst, R. (2006). Developments in the global law enforcement of cyber-crime. *Policing: An International Journal of Police Strategies & Management*, 29(3), 408-433.
- Brown, M. (2009). White Paper: Virtual Machines. Retrieved from http://www.maximumpc.com/article/features/white_paper_virtual_machines
- Buckle, P. (2009). *VDI Connection Broker Overview – featuring Leostream*. Retrieved from <http://xtravirt.com/xd10105>
- . case study. (2011). Retrieved from <http://www.merriam-webster.com/dictionary/case%20study>

- Casey, E. (2005). Case study: Network intrusion investigation – lessons in forensic preparation. *Digital Investigation*, 2(4), 254-260. doi:10.1016/j.diin.2005.11.007
- Davis, Z. (2011). *Profile management best practices in a desktop virtualization environment*: Ziff davis enterprises. Retrieved from <http://citrix.com/site/jumpPage.asp?pageID=1453077>
- Dick, B. (2000). *Grounded theory*. University of Queensland. Retrieved from http://www.uq.net.au/~zzbdick/dlitt/DLitt_P59ground.pdf
- Dodge, J. (2006). *VirtualCenter 2: Template Usage and Best Practices*: Foedus. Retrieved from http://www.vmware.com/pdf/vc_2_templates_usage_best_practices_wp.pdf
- Encase forensic - Detailed product description*. (2010). Retrieved from http://www.mediarecovery.pl/doc/encase-forensic/Detailed_Product_Description.pdf
- Enderle, R. (2011). *Is an iPad a PC?* Retrieved from <http://technologypundits.com/2011/03/is-an-ipad-a-pc/>
- Feagin, J. R., Orum, A. M., & Sjoberg, G. (1991). *A Case for the Case Study*. Retrieved from <http://www.questia.com/PM.qst?a=o&d=54443373>
- Fitterman, E., & Durick, J. D. (2010). Ghost in the machine. *Digital Forensics* 73-77. Retrieved from <http://www.vmforensics.org/files/Ghost%20in%20the%20Machine.pdf>
- Garfinkel, S. (2010). Digital forensics research: The next 10 years. *Digital Investigation*, 7, Supplement(0), S64-S73. doi:10.1016/j.diin.2010.05.009
- Gartner. (2009). Gartner says Worldwide Hosted virtual desktop Market to Surpass \$65 Billion in 2013. Retrieved from <http://www.gartner.com/it/page.jsp?id=920814>
- Glaser, B., & Strauss, A. (1967). *The discovery of grounded theory: Strategies for qualitative research*
- Glass, G. V., & Hopkins, K. D. (Eds.). (1984). *Statistical Methods in Education and Psychology* (2nd ed.)

- Golden, G., Richard, I., & Vassil, R. (2006). Next-generation digital forensics. *Communications of the ACM*, 49(2), 76-80. doi:10.1145/1113034.1113074 (ACM New York)
- Gorra, A. *An analysis of the relationship between individuals' perceptions of privacy and mobile phone location data - a grounded theory study* (PhD). Leeds Metropolitan University. Retrieved from http://www.leedsmet.ac.uk/inn/alic/agorra/3_Chapter3_Methodology_AndreaGorra.pdf
- Haletky, E. (2008). Thoughts on Forensics. Retrieved from <http://www.itworld.com/edward-haletky>
- Harrell, C. (2010). Overall DF Investigation Process. Retrieved from <http://journeyintoit.blogspot.com/2010/10/overall-df-investigation-process.html>
- Henry, P. (2009). *Best Practices In Digital Evidence Collection*. Retrieved 18/09/2011, 2011, from <http://computer-forensics.sans.org/blog/2009/09/12/best-practices-in-digital-evidence-collection/>
- Hopkins, W. G. (2008). Quantitative Research Design. *SPORTSCIENCE*(12).
- IBM. (2009, 25/11/2010). *TXSeries for Multiplatforms - Concepts and Planning*. Retrieved 10/10/2011, from <http://publib.boulder.ibm.com/infocenter/txformp/v6r0m0/index.jsp?topic=%2Fcom.ibm.cics.te.doc%2Ferziaz0015.htm>
- Ieong, R. S. C. (2006). FORZA – Digital forensics investigation framework that incorporate legal issues. *Digital Investigation*, 3, 29-36. doi:10.1016/j.diin.2006.06.004
- Intelligroup. (2009). *Whitepaper - Desktop Virtualization*. Retrieved from www.intelligroup.com/ppt/WhitepaperDesktopVirtualization.pdf
- Jasti, A., Shah, P., Nagaraj, R., & Pendse, R. (2010). Security in multi-tenancy cloud Symposium conducted at the meeting of the Security Technology (ICCST), 2010, Carnahan Retrieved from http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5678682
- Jump, A., & Gammage, B. (2009). *Emerging Technology Analysis: Hosted virtual desktops*.

- Junjie, P., Xuejun, Z., Zhou, L., Bofeng, Z., Wu, Z., & Qing, L. (2009). Comparison of Several Cloud Computing Platforms Symposium conducted at the meeting of the Information Science and Engineering (ISISE), 2009 Second International Symposium on Retrieved from 10.1109/ISISE.2009.94
- Kent, K., & Souppaya, M. (2006). *Guide to computer security log management*: National institute of standards and technology. Retrieved from csrc.nist.gov/publications/nistpubs/800-92/SP800-92.pdf
- Kothari, D. C. R. (2004). *Research methodology: methods and techniques* (Second ed.)[Internet]. Retrieved from http://books.google.com/books?id=8c6gkbKi-F4C&printsec=frontcover&source=gbs_ge_summary_r&cad=0#v=onepage&q&f=false. Retrieved from Google Books database.
- Krathwohl, D. R. (1993). *Methods of Educational and Social Science Research*
- Larson, R., & Carbon, J. (2009). *Windows Server® 2008 Hyper-V™ Resource Kit* Retrieved from <http://books.google.co.nz/books?id=sGk5AgB3ttYC&lpg=PP1&dq=Windows%20Server%202008%20Hyper-V%20Resource&pg=PP1#v=onepage&q&f=false>
- Laverick, M. (2005). *VMware ESX REDO Demystified*. Retrieved from <http://www.rtfm-ed.co.uk/docs/vmwdocs/whitepaper-vmware-esx2.x-redo-demystified.pdf>
- Leong, R. (2006). Challenges to digital forensics from cloud computing Symposium conducted at the meeting of the DFRWS, Hong Kong. Retrieved from www.dfrws.org/2006/proceedings/4-Ieong.pdf
- Maltz, B. (2010). *Configuring Active Directory GPOs in a VDI environment*. Retrieved from <http://searchvirtualdesktop.techtarget.com/tip/Configuring-Active-Directory-GPOs-in-a-VDI-environment>
- Martin, A. (2008). *Mobile Device Forensics*: SANS. Retrieved from http://www.sans.org/reading_room/whitepapers/forensics/mobile-device-forensics_32888
- Matsuura, J. (2011). *Opinion: Regulatory compliance in the cloud*. Retrieved from <http://computerworld.co.nz/news.nsf/news/opinion-regulatory-compliance-in-the-cloud>

- McLaughlin, L. (2007). *How to Find and Fix 10 Real Security Threats on Your Virtual Servers*. Retrieved from http://www.cio.com/article/154950/How_to_Find_and_Fix_10_Real_Security_Threats_on_Your_Virtual_Servers
- Mell, P., & Grance, T. (2011). *The NIST Definition of cloud computing*. Retrieved from www.nist.gov/itl/cloud/upload/cloud-def-v15.pdf
- Menken, I., & Blokdijk, G. (2008). *The Complete Cornerstone Guide to Virtualization Best Practices*: Emereo Pty Limited.
- Messmer, E. (2009). *Gartner: Server virtualization now at 18% of server workload*. Retrieved from <http://www.networkworld.com/news/2009/102009-gartner-server-virtualization.html>
- . *Methodology*. Retrieved from <http://www.merriam-webster.com/thesaurus/methodology>
- Microsoft. (2005). *Auditing security events best practices*. Retrieved from <http://technet.microsoft.com/en-us/library/cc778162%28WS.10%29.aspx>
- Myers, M. D. (1997). *Qualitative Research in Information Systems*. Retrieved from <http://www.qual.auckland.ac.nz/#Overview%20of%20Qualitative%20Research>
- Nikkel, B. (2010). *Corporate IT forensics in the new decade, Hong Kong*. Retrieved from http://www.ar.admin.ch/internet/armasuisse/en/home/themen/wissenschaft/technologie/veranstaltungen/Digital_Forensics_Research_Network.parsys.80503.DownloadFile.tmp/corporateitforensicsinthenewdecade.pdf
- NIST, & NIST_Visual_Model_of_Cloud_Computing_Definition.jpg. (2011). *Visual model of NIST working definition of cloud computing* [jpg].
- Olivier, M. S. (2009). On metadata context in Database Forensics. *Digital Investigation*, 5(3-4), 115-123. doi:10.1016/j.diin.2008.10.001
- . *Paraben's P2 Commander 2.0 - Release notes*. Retrieved from www.paraben.com/downloads/p2c20.pdf

- Personal Computer Timeline. (2000). *PCMAG.com*. Retrieved from http://www.pcmag.com/encyclopedia_term/0,2542,t=personal+computer&i=49133,00.asp#fbid=hkENiMcooBo
- Raghav, S., & Saxena, A. K. (2009). Mobile forensics: Guidelines and challenges in data preservation and acquisition Symposium conducted at the meeting of the IEEE Student Conference on research and development, 2009
- Reilly, D., Wren, C., & Berry, T. (2010). Cloud computing: Forensic challenges for law enforcement Symposium conducted at the meeting of the Internet Technology and Secured Transactions, 2010, London.
- Roebuck, K. (2011). *Platform as a Service (PaaS): High-impact Emerging Technology*. Retrieved from <http://www.scribd.com/doc/58603927/Platform-as-a-Service-PaaS-High-impact-Emerging-Technology-What-You-Need-to-Know-Definitions-Adoptions-Impact-Benefits-Maturity-Vendors>
- Ross, S. M., & Morrison, G. R. (2004). Experimental Research Methods.
- Ruan, K., & Carthy, J. (2011). Cloud forensics: An overview.
- Rutkowski, J. (2004). *Red Pill or how to detect WMM using (almost) one CPU instruction*. Retrieved from <http://invisiblethings.org/papers/redpill.html>
- Sabahi, F. (2011). Cloud Computing Security Threats and Responses Symposium conducted at the meeting of the IEEE 3rd International Conference on Communication Software and Networks (ICCSN), 2011
- Sanya-Isijola, A. (2009). *Models of Digital Forensic Investigation*: University of East London.
- Schulz, G. (2011). *What is a LUN, and why do we need one?* Retrieved from <http://searchstorage.techtarget.com/answer/What-is-a-LUN-and-why-do-we-need-one>
- Shavers, B. (2008). Virtual Forensics.
- Shuttleworth, M. (2009). *What is the Scientific Method?* Retrieved from <http://www.experiment-resources.com/what-is-the-scientific-method.html>

- Slay, J., & Simon, M. (2008). *Voice over IP forensics*. presented at the meeting of the Proceedings of the 1st international conference on Forensic applications and techniques in telecommunications, information, and multimedia and workshop, Adelaide, Australia.
- Smith, J. E., & Nair, R. (2005). The architecture of virtual machines *Computer*, 38(5), 32-38.
- Spruill, A., & Pavan, C. (2007). Tackling the U3 trend with computer forensics. *Digital Investigation*, 4, 7-12.
- Stake, R. E. (1995). The art of case study research. Retrieved from http://www.personal.psu.edu/rsw136/blogs/rebecca_west_burns/2009/07/stake-r-1995-the-art-of-case-study-research-thousand-oaks-ca-sage-publications-chapter-4.html
- Strauss, A., & Corbin, J. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory* (Second ed.). Retrieved from http://books.google.com/books/about/Basics_of_qualitative_research.html?id=wTwYUnHYsmMC
- Tashakkori, A., & Teddlie, C. (2003). *Handbook of Mixed Methods in Social & Behavioral Research* (First ed.): Sage Publications.
- Taylor, M., Haggerty, J., Gresty, D., & Hegarty, R. (2010). Digital evidence in cloud computing systems. *Computer Law & Security Review*, 26(3), 304-308.
doi:10.1016/j.clsr.2010.03.002
- Timeline of virtualization development*. (2011). Retrieved from http://en.wikipedia.org/wiki/Timeline_of_virtualization_development
- Turban, E., King, D., Lee, J., & Viehland, D. (2008). Building E-Commerce Applications and Infrastructure. In *Electronic Commerce A Managerial Perspective* (5th ed.): Prentice-Hall.
- Understanding virtual machine snapshots in VMware ESX, & VMware*. (2010). [VMware Knowledge Base (KB)].
- Vanover, R. (2010). *VDI Display Protocols -- Why They're Important*. Retrieved 20/10/11, 2011, from <http://virtualizationreview.com/articles/2010/05/01/vdi-display-protocols.aspx>

- VMware. (2009a). *Workstation User's Manual* (Vol. EN-000168-00)
- VMware, Inc. (2009b). *Performance best practices for VMware vSphere® 4.0*. Retrieved from www.vmware.com/pdf/Perf_Best_Practices_vSphere4.0.pdf
- VMware, I., &. (2010). *VMware vCenter Converter Standalone User's Guide* 3401 Hillview Ave Palo Alto, CA 94304.
- West, B. (2011). *Most organizations do not follow security best practices, survey finds*. Retrieved from <http://www.infosecurity-magazine.com/view/19737/most-organizations-do-not-follow-security-best-practices-survey-finds/>
- Wiles, J., & Reyes, A. (2007). *The best damn cybercrime and digital forensics book period: Syngress*. Retrieved from <http://my.safaribooksonline.com/book/networking/forensic-analysis/9781597492287>
- Wilson, E. B. (1952). Courier Dover Publications.
- Wolfe, H. B. (2007). What is forensic computing? In *The best damn cybercrime and digital forensics book period: Syngress Publishing*.
- Wright, P. M. (2005). *Oracle Database Forensics using LogMiner*. London: Global Information Assurance Certification Paper. Retrieved from <http://www.giac.org/paper/gcfa/159/oracle-database-forensics-logminer/105140>
- Yin, R. K. (2003). *Case study research: design and methods*. Retrieved from <http://www.amazon.com/s?search-alias=stripbooks&field-isbn=0761925538>
- Zhang, L., Zhang, D., & Wang, L. (2010). Live digital forensics in a virtual machine *Computer Application and System Modeling (ICCASM)*, V4-328 - V324-332. doi:10.1109/ICCASM.2010.5620364

Appendix

APPENDIX A:

Events generated by VMware View connection broker

EventType	Severity	ModuleAndEventText
BROKER_AGENT_OFFLINE	WARNING	The agent running on machine \${MachineName} has not responded to queries, marking it as offline
BROKER_AGENT_ONLINE	WARNING	The agent running on machine \${MachineName} is responding again, but did not send a startup message
BROKER_DAILY_MAX_USERS	INFO	\$(Time): Over the past 24 hours, the maximum number of users with concurrent desktop sessions was \${UserCount}
BROKER_DESKTOP_LAUNCH_FAILURE	ERROR	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: The broker encountered an error while processing the request, please contact support for assistance
BROKER_DESKTOP_NOT_ENTITLED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: User is not entitled to this Pool

EventType	Severity	ModuleAndEventText
BROKER_DESKTOP_PROTOCOL_NOT_SUPPORTED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Requested protocol \${ProtocolId} is not supported
BROKER_DESKTOP_REQUEST	INFO	User \${UserDisplayName} requested Pool \${DesktopId}
BROKER_EVENT_HANDLING_STARTED	INFO	Broker \${BrokerName} has started handling events
BROKER_EVENT_HANDLING_STOPPED	INFO	\${BrokerName} has stopped handling events
BROKER_LOCALMODE_OLD_ANCHOR_DELETE_FAILURE	WARNING	Old anchor snapshot deletion task failed for Machine \${MachineName}. Error message: \${LocalModeMessage}
BROKER_LOCALMODE_OPERATION_AUDIT_FAILURE	AUDIT_FAIL	Local Mode Operation \${LocalModeOperation} failed for Desktop \${MachineName}, ID: \${DesktopId}. Error message: \${LocalModeMessage}
BROKER_LOCALMODE_OPERATION_FAILURE	ERROR	Local Mode Operation \${LocalModeOperation} failed for Desktop \${MachineName}, ID: \${DesktopId}. Error message: \${LocalModeMessage}
BROKER_LOCALMODE_OPERATION_SUCCESS	AUDIT_SUCCESS	\${LocalModeOperation} has completed for Desktop \${MachineName}, ID: \${DesktopId}.
BROKER_MACHINE_ALLOCATED	INFO	User \${UserDisplayName} requested Pool \${DesktopId}, allocated machine \${MachineName}
BROKER_MACHINE_ASSIGNED_UNAVAILABLE	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Assigned machine \${MachineName} is unavailable

BROKER_MACHINE_CANNOT_CONNECT	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Failed to connect to Machine \${MachineName} using \${ProtocolId}
BROKER_MACHINE_CHECKEDOUT	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: VM \${MachineName} is currently checked out on a client device
BROKER_MACHINE_CONFIGURED_VIDEO_SETTINGS	INFO	Successfully configured video settings for Machine VM \${MachineName} in Pool \${DesktopId}
BROKER_MACHINE_NOT_READY	WARNING	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} is not ready to accept connections
BROKER_MACHINE_OPERATION_DELETED	INFO	machine \${MachineName} has been deleted
BROKER_MACHINE_PROTOCOL_NOT_SUPPORTED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} does not support protocol \${ProtocolId}
BROKER_MACHINE_PROTOCOL_UNAVAILABLE	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} did not report protocol \${ProtocolId} as ready

EventType	Severity	ModuleAndEventText
BROKER_MACHINE_REJECTED_SESSION	WARNING	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Machine \${MachineName} rejected the start session request
BROKER_MACHINE_SESSION_TIMEDOUT	WARNING	Session for user \${UserDisplayName} timed out
BROKER_POOL_CANNOT_ASSIGN	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: There are no machines available to assign the user to
BROKER_POOL_COMANAGER_REQUIRED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: No co-management availability for protocol \${ProtocolId}
BROKER_POOL_EMPTY	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: The Desktop Pool is empty
BROKER_POOL_NO_MACHINE_ASSIGNED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: No machine assigned to this user
BROKER_POOL_NO_RESPONSES	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: No machines in the Desktop Pool are responsive
BROKER_POOL_OVERLOADED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: All responding machines are currently in use

BROKER_POOL_POLICY_VIOLATION	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: This Desktop Pool does not allow online sessions
BROKER_POOL_PROTOCOL_NOT_SUPPORTED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: There were no machines available that support protocol \${ProtocolId}
BROKER_POOL_PROTOCOL_UNAVAILABLE	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: There were no machines available that reported protocol \${ProtocolId} as ready
BROKER_POOL_TUNNEL_NOT_SUPPORTED	AUDIT_FAIL	Unable to launch from Pool \${DesktopId} for user \${UserDisplayName}: Tunneling is not supported for protocol \${ProtocolId}
BROKER_PROVISIONING_ERROR_CONFIG_CLEARED	INFO	The previously reported configuration problem is no longer present on Pool \${DesktopId}
BROKER_PROVISIONING_ERROR_CONFIG_SET	ERROR	Provisioning error occurred on Pool \${DesktopId} because of a configuration problem
BROKER_PROVISIONING_ERROR_DISK_CLEARED	INFO	The previously reported disk problem is no longer present on Pool \${DesktopId}
BROKER_PROVISIONING_ERROR_DISK_LC_RESERVATION_CLEARED	INFO	The previously reported error due to available free disk space reserved for linked clones is no longer present on Pool \${DesktopId}

BROKER_SVI_ARCHIVE_UDD_FAILED	AUDIT_FAIL	Failed to archive user data disk \${UserDiskName} to location \${SVIPath}
BROKER_SVI_ARCHIVE_UDD_SUCCEEDED	AUDIT_SUCCESS	Archived user data disk \${UserDiskName} to location \${SVIPath}
BROKER_SVI_ATTACH_UDD_FAILED	AUDIT_FAIL	Failed to attach user data disk \${UserDiskName} to VM \${SVTVMID}
BROKER_SVI_ATTACH_UDD_SUCCEEDED	AUDIT_SUCCESS	Attached user data disk \${UserDiskName} to VM \${SVTVMID}
BROKER_SVI_DETACH_UDD_FAILED	AUDIT_FAIL	Failed to detach user data disk \${UserDiskName} from VM \${SVTVMID}
BROKER_SVI_DETACH_UDD_SUCCEEDED	AUDIT_SUCCESS	Detached user data disk \${UserDiskName} from VM \${SVTVMID}
BROKER_USER_AUTHFAILED_ACCOUNT_DISABLED	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because the account is disabled
BROKER_USER_AUTHFAILED_ACCOUNT_EXPIRED	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because the account has expired
BROKER_USER_AUTHFAILED_ACCOUNT_LOCKED_OUT	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because the account is locked out
BROKER_USER_AUTHFAILED_ACCOUNT_RESTRICTION	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because of an account restriction
BROKER_USER_AUTHFAILED_BAD_USER_PASSWORD	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because of a bad username or password
BROKER_USER_AUTHFAILED_GENERAL	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate

EventType	Severity	ModuleAndEventText
BROKER_USER_AUTHFAILED_NO_LOGON_SERVERS	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because there are no logon servers
BROKER_USER_AUTHFAILED_PASSWORD_EXPIRED	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because the password has expired
BROKER_USER_AUTHFAILED_PASSWORD_MUST_CHANGE	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because the password must change
BROKER_USER_AUTHFAILED_SECUREID_ACCESS_DENIED	AUDIT_FAIL	SecurID access denied for user \${UserDisplayName}
BROKER_USER_AUTHFAILED_SECUREID_NEWPIN_REJECTED	AUDIT_FAIL	SecurID access denied for user \${UserDisplayName} because new pin was rejected
BROKER_USER_AUTHFAILED_SECUREID_WRONG_NEXTTOKEN	AUDIT_FAIL	SecurID access denied for user \${UserDisplayName} because wrong next token entered
BROKER_USER_AUTHFAILED_SECUREID_WRONG_STATE	AUDIT_FAIL	SecurID access denied for user \${UserDisplayName} because of incorrect state
BROKER_USER_AUTHFAILED_TIME_RESTRICTION	AUDIT_FAIL	User \${UserDisplayName} failed to authenticate because of a time restriction
BROKER_USER_NOT_AUTHORIZED	AUDIT_FAIL	User \${UserDisplayName} has authenticated, but is not authorized to perform the operation
BROKER_USER_NOT_ENTITLED	AUDIT_FAIL	User \${UserDisplayName} has authenticated, but is not entitled to any Pools
BROKER_USERCHANGEDPASSWORD	AUDIT_SUCCESS	Password for \${UserDisplayName} has been changed by the user
BROKER_USERLOGGEDIN	AUDIT_SUCCESS	User \${UserDisplayName} has logged in
BROKER_USERLOGGEDOUT	AUDIT_SUCCESS	User \${UserDisplayName} has logged out
BROKER_VC_DISABLED	INFO	vCenter at address \${VCAddress} has been temporarily disabled
BROKER_VC_ENABLED	INFO	vCenter at address \${VCAddress} has been enabled
BROKER_VC_STATUS_CHANGED_CANNOT_LOGIN	WARNING	Cannot log in to vCenter at address \${VCAddress}
BROKER_VC_STATUS_CHANGED_DOWN	INFO	vCenter at address \${VCAddress} is down
BROKER_VC_STATUS_CHANGED_INVALID_CREDENTIALS	WARNING	vCenter at address \${VCAddress} has invalid credentials
BROKER_VC_STATUS_CHANGED_NOT_YET_CONNECTED	INFO	Not yet connected to vCenter at address \${VCAddress}
BROKER_VC_STATUS_CHANGED_RECONNECTING	INFO	Reconnecting to vCenter at address \${VCAddress}
BROKER_VC_STATUS_CHANGED_UNKNOWN	WARNING	The status of vCenter at address \${VCAddress} is unknown
BROKER_VC_STATUS_CHANGED_UP	INFO	vCenter at address \${VCAddress} is up
MULTIPLE_DESKTOPS_FOR_KIOSK_USER	WARNING	User \${UserDisplayName} is entitled to multiple desktop pools

Events generated by VMware View agent

EventType	Severity	ModuleAndEventText
AGENT_CONNECTED	INFO	User \${UserDisplayName} has logged in to a new session on machine \${MachineName}
AGENT_DISCONNECTED	INFO	User \${UserDisplayName} has disconnected from machine \${MachineName}
AGENT_ENDED	INFO	User \${UserDisplayName} has logged off machine \${MachineName}
AGENT_PENDING	INFO	The agent running on machine \${MachineName} has accepted an allocated session for user \${UserDisplayName}
AGENT_PENDING_EXPIRED	WARNING	The pending session on machine \${MachineName} for user \${UserDisplayName} has expired
AGENT_RECONFIGURED	INFO	Machine \${MachineName} has been successfully reconfigured
AGENT_RECONNECTED	INFO	User \${UserDisplayName} has reconnected to machine \${MachineName}
AGENT_RESUME	INFO	The agent on machine \${MachineName} sent a resume message
AGENT_SHUTDOWN	INFO	The agent running on machine \${MachineName} has shut down, this machine will be unavailable
AGENT_STARTUP	INFO	The agent running on machine \${MachineName} has contacted the connection server and sent a startup message
AGENT_SUSPEND	INFO	The agent on machine \${MachineName} sent a suspend message

