A Synchronized Log Cloud Forensic Framework

Sean Thorpe[1], Indrajit Ray[2], Tyrone Grandison[3]

[1]Faculty of Engineering and Computing
University of Technology, Kingston, Jamaica
sthorpe@utech.edu.jm

[2]Department of Computer Science
Colorado State University, Fort Collins, CO, USA
indrajit@cs.colostate.edu

[3]IBM Research
Yorktown Heights, NY, USA
tyroneg@us.ibm.com

Abstract. In this paper, we present a framework for compute cloud forensics that includes an investigation process model based on physical crime scene procedures. The work is motivated by work done by Carrier and Spafford [3]. We posit the concepts in [3] for articulation within a distributed service oriented virtualization environment aptly described as a utility compute cloud. In this model, each digital device is a connection of virtual cloud servers mapped to a physical disk cluster situated within the storage area network (SAN). Each digital device is considered a part of the physical crime scene where it is located. The investigation includes the preservation of the Virtual Machine (VM) Networks, the search for digital evidence, and the reconstruction of digital events. Our contribution applies a case study evaluation of our design of a VM log auditor software application within the University environment to corroborate the arguments presented within this suggested framework. The focus of the cloud forensics investigation is on the reconstruction of events using the synchronized VM and physical disk log files so that hypotheses can be developed and tested. The paper also includes definitions and descriptions of the basic and core concepts the framework uses.

Keywords: Cloud, Digital Forensics, Log Synchronization

1 Introduction

Cloud Computing and virtualization are terms used interchangeably and date back from work in the areas of both Grid Computing, it's predecessor, and Mainframe Systems the pioneering technology from IBM. As a basis for establishing relevance to the field of digital forensics, cloud computing is still an immature discipline with very little evidence of proven research results. However, as industry and academia adopt

virtual clouds as a network provision that purport economies of scale for optimizing centralized system hardware, the opportunities for crime present a clear, if not one of the biggest, opportunities for miscreant communities.

For the benefit of the reader, utility compute clouds represent by definition mapping a file to a binding IP and MAC (machine) address within the physical data center which may be geographically situated in remote locations, for which the users of such devices may not have direct control o r ownership of the said resources [2]. Simply put the Virtual Machine user rents space within physical data center for hosting his services. Although the proponent argument for compute clouds is predicated on cost savings for the System users of these services, the clear danger of security of the data manipulated within these abstract domains presents challenges for auditing and forensic investigations. This represents our primary concern as researchers at this time.

At the Digital Forensics Research Workshop (DFRWS) in 2001 the need for a standard framework was acknowledged; however, progress has been slow. By application, the need for digital forensics for a compute cloud equally presents similar if not new challenges within this realm. A framework for digital forensics with a compute cloud (more popularly referred to as Cloud forensics today) needs to be flexible enough so that it can support future technologies and different types of incidents. Therefore, it needs to be simple although it's abstract. On the one hand, if it is too simple and abstract then it is difficult to create tool requirements and test procedures for each phase.

For this paper, we have examined the concept of an investigation to determine what is required. The result is a Virtual Machine (VM) log event based framework that can be used to develop hypotheses and answer questions about an incident or a crime. Hypotheses are developed by collecting VM objects that may have played a role in an event that was related to an incident. Once the objects are collected as evidence, the investigator/administrator can develop hypotheses about previous events at the crime scene.

The framework is based on the well-established process model that is used at physical police crime scenes, that has been refined through repeated use over many years, and that has been accepted in many court cases. Using this model we articulate that investigating a Virtual Machine is similar to investigating a physical crime. The log cloud forensic framework adopts clear goals for each phase. In a future work, we plan to propose requirements for each phase.

The rest of the paper is organized as follows. Section 2 of this paper describes the basic concepts that are used within our compute cloud forensic framework. Section 3 describes the big picture of the framework and Section 4 focuses on the digital analysis types. Section 5 deals with an evaluative case study. Of how we apply the cloud forensic framework, Section 6 describes the Virtual Machine Digital Investigation Process Model. Section 7 deals with System Preservation and Documentation. Section 8 focuses on Digital Evidence Searching and Documentation Section 9 gives information on Digital Event Reconstruction and Documentation while Section 10 compares this model to other existing models and Section 11 concludes the paper.

2 General Concepts and Definitions

We shall borrow some of the basic and traditional definitions that are used in the area of digital forensics research. Digital data are data represented in a numerical form. With modern computers, it is common for the data to be internally represented in a binary encoding, but this is not a requirement. A digital object is a discrete collection of digital data, which could be a file, a hard disk sector, a network IP address packet, a machine (MAC) address, a memory page, or even a process. By analogy the Digital Object is also equivalent to that of our VM Data Object. And hence the VM data object is nothing more than a meta digital data object.

In addition to its numerical representation, digital data has a physical representation. For example, the bits in a hard disk are magnetic impulses on platters that can be read with analog sensors. Network wires contain electric signals that represent network packets and keyboard cables contain electric signals to a digital representation. Things like Digital photography and video are a digital representation of the light associated with physical objects. Digital data can be stored on many mediums and each has different properties that determine how long data will reside. For example, data will reside on a keyboard cable for a fraction of a second, but it may reside on a hard disk for years.

We posit that the VM Digital Objects, which are merely meta-data sets of an existing Digital Object, has unique features based on the creator/owner and function(s) [2]. For example, the characteristics of a hard disk sector will be different when it is used to store contents of an ASCII text document versus a JPEG image. We can use the characteristics to identify data. The state of an object is the value of its characteristics. If a letter were changed in an ASCII text document corresponding to the file, the file would have a new state. Similarly the state of a running computer process changes every time data is written to its memory.

We attribute to the authors in [3] that a Digital event is an occurrence that changes the state of one or more digital data object. Hence if the state of a VM object changes as a result of an event, then it is an effect of an event. Some types of objects have the ability to cause events and they are called causes. Note that because digital objects are stored in a physical form, then their state can be altered by both physical and digital events. An object is evidence of an event if the event changed the object's state. This means that the object can be examined for information about past events. Every object is at least evidence of at least one event, because there has to be an event that created the VM object.

Some environments have developed policies and laws that forbid certain events occurring. An incident is an event or sequence of events that violate a policy. A crime is an event or sequence of events that violate a law. Particularly, a digital incident comprises of one or more digital events that violate a policy. In response to an incident or crime, an investigation may begin. An investigation is a process that develops and test hypotheses about events that occurred. Example questions include "what caused the incident to occur", "when did the incident occur", and "where did the incident occur".

To develop and test hypotheses about the VM events that occurred before, during and after the incident, we need to determine what actually happened. The only proof that an event may have occurred is if the *logged evidence* of the event exist. If the

object whose state was changed by the event still exists, then we examine it for information about the event and about other VM objects that were causes or effects of the event. Hence we could argue with some specificity that an auditable VM or compute cloud object is a log file or a set of log files of evidence of an incident if its state was used to cause an event related to the incident or if its state was changed by an event that was related to the incident.

For this framework, we will use the following definitions of evidence, which are a little more general and do not focus on the cause and effect relationship. Physical evidence of an incident is any physical object that contains reliable information that supports or refutes a hypothesis about the incident and digital evidence of an incident is any digital data that contains reliable information that supports or refutes reliable information about the incident [3]. It is understood that a VM object has information about an incident because it was a cause or effect in an event related to the incident.

Since digital data has a physical form resident on the SAN disks, then physical evidence can contain digital evidence. Using this definition, a hard disk in our SAN cluster to which the VM Objects are mapped via the VM system raw device mapping processes. A hard disk is physical evidence and the sectors and files that contain information about the incident are digital evidence. The Electronic Crime Scene Investigation guide [4] describes the recognition and collection of a hard disk or other storage device as the collection of electronic, or digital evidence.

In our synchronized log framework, the collection of the SAN hard disk(s) is the collection of digital evidence and the collection of a digital object from each such possible disk as a basis of a chain of digital evidence. Equally we should remind the reader that for VM objects, this is really meta-data or an instance about an existing digital object stored within the SAN data centre. Also, note that the difference between physical and digital evidence is their format and nothing to do with the type of incident. For the purposes of our own work we adopt the well established Advanced Digital Forensic format (ADDF). Therefore, we can have digital evidence for a physical incident or crime. For example at our labs, a digital video camera will create a digital representation of a physical event and the resulting file will be digital evidence of the event. We can also have physical evidence for a digital crime.

3 Virtual Machine Digital Forensic Investigation

The previous section discussed the basic concepts of an investigation without mention of the term forensic. To determine whether or not the term forensic can be applied we will first consult its definition. The American Heritage Dictionary [5] defines forensic as an adjective and "relating to science or technology in the investigation and establishment of facts of evidence in a court of law." Therefore to be considered forensic, a process must use science and technology and the results for the purposes of a court law.

With digital evidence, technology is always needed to process the digital data and therefore the only difference between a forensic and a non-forensic investigation of VM digital data is whether or not the evidence can be used in a court of law. We argue within the context of our own framework definitions that compute cloud

forensics is predicated on this traditional principle of digital forensic investigation. A forensic investigation is a process that uses science and technology to develop and test theories, which can be entered into a court of law, to answer questions about events that occurred. In particular, a digital forensic investigation is a process that uses science and technology to examine digital objects and that develops and tests theories, which can be entered into a court of law, to answer questions about events that occurred. The requirements to enter digital evidence into a court of law are specific to the jurisdiction of the case in question. This however is not the focus of our current paper.

4 Digital Analysis Types

A Digital investigation may encounter many formats of digital data including the formats that store the logs and therefore there exist several types of analysis that can be done on such data. The different analysis types are based on interpretation, or abstraction, layers, which are generally a part of the data's design [3][4]. For example consider the data on a hard disk, which has been designed with several interpretation layers. The lowest layer may contain partitions or other containers that are used for volume management. Inside of each partition is data that has been organized into a file system or database. The data in a file system is interpreted to create files that contain data in an application specific format. Each of these layers has its own analysis techniques and requirements. Examples of common digital analysis types include:

Media Analysis: The analysis of the data from a storage device. This analysis does not consider any partitions or other operating system specific data structures. If the storage device uses a fixed size unit, such as a sector, then it can be used for this analysis.

Media Management Analysis: This analysis of the management system used to organize the media. This typically involves partitions and may include volume management or RAID system that merges data from multiple storage devices into a single virtual storage device.

File System Analysis: The analysis of the file system data inside of a partition or a disk. This typically involves processing the data to extract the contents of a file or to recover the contents of a deleted file.

Application Analysis: The analysis of the data inside of a file. Files are created by users and applications and the content are application specific.

Network Analysis: The analysis of the data on a communication network. Network packets can be examined using the OSI model to interpret the raw data into an application level stream.

5 An Evaluative Case Study of the Cloud Forensic Framework

At the University of Technology (UTECH) Jamaica, we have designed a software application called a log auditor that synchronizes the log events of VM environment.

We run a VMware essx3i host that connects 12 VM server instances on that host. The events being mapped records our payroll systems, and student registration. The VM's physical disk is located on a 40 Terabyte hard disk Storage Area Network (SAN). The log auditor currently runs on a separate SAN disk domain. An ftp session that reads the existing production VM disk logs is written to the auditor's back end Oracle 11g database from which forensic log reports are generated for the VM system administrators. The log auditor is used to corroborate the forensic framework we described in this paper. Ongoing work on the auditor is still being done to perform dynamic mapping of the native VM environments. The internal operational tasks of the log auditor are excluded from this paper.

6 The Virtual Machine Design Investigation Process Model

Based on ideas articulated in[3] we will now describe the proposed process model. The model is based on the phases that are documented for investigating physical crime scenes. The phases are applied to a Virtual Machine digital crime scene, where we consider the digital crime scene investigation to occur as a subset of a physical crime scene data center investigation. It is organized into five categories of phases as shown in Fig. 1.

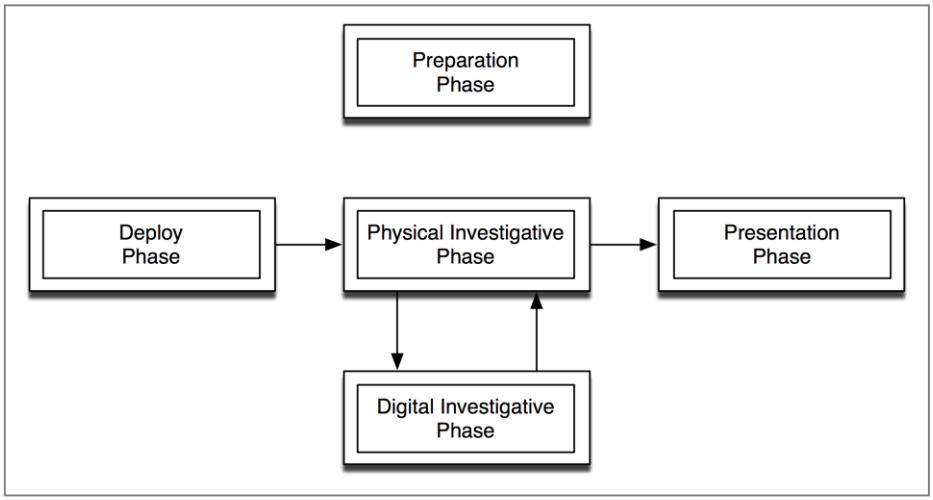

Fig. 1. Graphical representation of major categories of phases in an adopted cloud forensic framework [3]

Preparation Phase: Includes the operations readiness, which allows for training the appropriate people and tests the tools that will be used to investigate a system. The infrastructure readiness phase configures the equipment to help ensure that the needed data exists when an incident occurs. For example, in a corporate or military

environment this could include adding network monitoring tools to increase logging levels.

Deploy Phase: Includes detection and notification where the victim or another party detects the incident and the investigators are alerted. For example, a network intrusion could be detected by an intrusion detection system and a contraband incident could be detected using logs. Our ongoing work on the design of a VM log synchronization tool auditor currently explores the effort in this phase. This phase also includes the confirmation and authorization where the investigators/ system administrators receive authorization to conduct an investigation. In a corporate environment, this could include the incident response team doing a brief analysis of a system to confirm that it has indeed been compromised. If it is a critical system, additional permission may be needed before a full analysis can be conducted. In a law enforcement environment, the officer may need to obtain a search warrant before the investigation can progress.

Physical Investigative Phase: Includes the operations readiness, which allows for training the appropriate people and tests the tools that will be used to investigate a system. The infrastructure readiness phase configures the equipment to help ensure that the needed data exists when an incident occurs. For example, in a corporate or military environment this could include adding network monitoring tools to increase logging levels.

Digital Investigative Phase: This is the phase that examines the digital data for evidence. This set of phases is actually a subset of the physical investigate phase for the crime scene investigations and the conclusions that are made from such digital investigations. The results will be used in the investigate phase of the investigation.

Presentation Phase: After theories have been developed and tested about the events related to the incident, the results can be presented to either a corporate audience or a court of law.

Although all the phases are important, our focus in this work is the digital investigation phase. For the purpose of our Virtual Machine environment, we decompose this phase into three sub phases as shown in Fig. 2. In the next section, we discuss each of these sub phases in more detail.

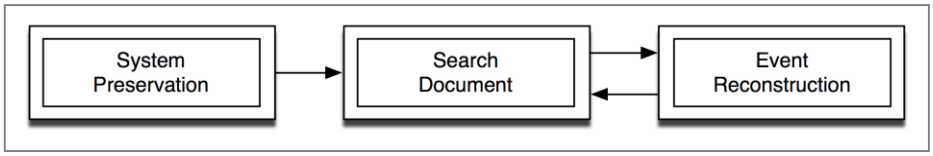

Fig. 2. Graphical representation of the three sub phases in a digital investigation[3]

7 System Preservation and Documentation

The first phase of a digital investigation is to preserve the crime scene. In the case of a virtual machine environment, this arguably has to be mapped to the physical data center SAN disk, or co-located data centers over which the attack had spawned. We

take into account that this digital crime must account for all associated hardware and software where the digital evidence about the crime or incident exists. Recall that a Cloud forensic investigation is looking for VM objects whose state is an effect of a VM event that was related to an incident. Therefore, we want to preserve the state of as many digital VM objects by reducing the number of additional events that may occur. Any event could modify the state of a piece of evidence and destroy the relevant information. The goal of this phase is to preserve the state of the crime scene. Equally important is the need to document the state of the digital crime scene as a source of ongoing reference for the current and future related incidents of the same.

By analogy to a physical crime scene, this phase occurs when a first responder shows up at the scene and assists an injured, detains a suspect, and limits unwanted traffic in the specific area of the Data Centre. After the area is secured, only authorized investigators are allowed on site. Documentation is achieved through video, photography and sketching.

At a digital crime scene, the actions are dependent on the goals and details of the incident. At one extreme is imaging and at the other the system could be powered off and mirror copies of the hard disk are made. Copies of the memory may also be made using software or hardware. This type of preservation can create an exact copy of the system so that it can be recreated later. This is analogous to making a copy of a building and taking it in to the police for analysis. When imaging occurs, the investigator has full documentation of the crime scene and as long as the integrity of the image is maintained then no additional steps are needed. In practice, a backup copy of the image is recommended as a safeguard.

On the other extreme you may seemingly be not able to take steps to preserve the crime scene. This could be scenario on a critical server where it cannot be turned off and no process can be killed. For this scenario, the crime scene could be documented by running data collection tools from a CD, and writing the tool output to dedicated and secure network drives. Somewhere in between these two extremes is the process of containment, which frequently occurs when responding to a system intrusion. Suspect processes and non-critical processes are killed so that they cannot overwrite evidence. Network filtering is applied or the network cable is plugged into an empty hub so that malicious sessions cannot occur. Copies of critical synchronized VM log files are copied from the system via a third party log auditor, so that data is not lost. An image of the system could be made while it is still running. The goal is to keep the system running, but minimize the amount of data that is being changed. This is similar to reducing the number of unauthorized people who can access a crime scene.

From a non forensic point of view, this phase is sometimes not seen as a requirement. An investigation could occur on a live system and accurate conclusions could be made. This phase is frequently performed because it improves the probability of finding reliable and relevant evidence. But the same evidence may still exist even if preservation steps are not taken. From a forensic investigation point of view, then this may be required. Some courts may require that the suspect be able to analyze an image of the system and therefore an image of some sort is required. General technology based requirements can be developed for this phase, but it will be up to the courts to determine what preservation steps are required for any digital evidence to be entered into a court of law.

We have become used to the idea that an image of the SAN disk exists, but this may need to be reconsidered as disk sizes get larger and it becomes infeasible to make copies of every disk. The new real time challenges of digital forensics and uptime requirements of servers may necessitate that evidence be entered into court without the complete system image being created. Unfortunately, this type of scenario can lead to inconclusive and falsification of results.

8 Digital Evidence Searching and Documentation

After steps have been taken to preserve the state of the digital VM objects at the digital crime scene, such a crime scene is searched for evidence. The primary goal is to recognize the VM digital objects that may contain information about an incident. As shown in Fig. 3, searching is a four phase process.

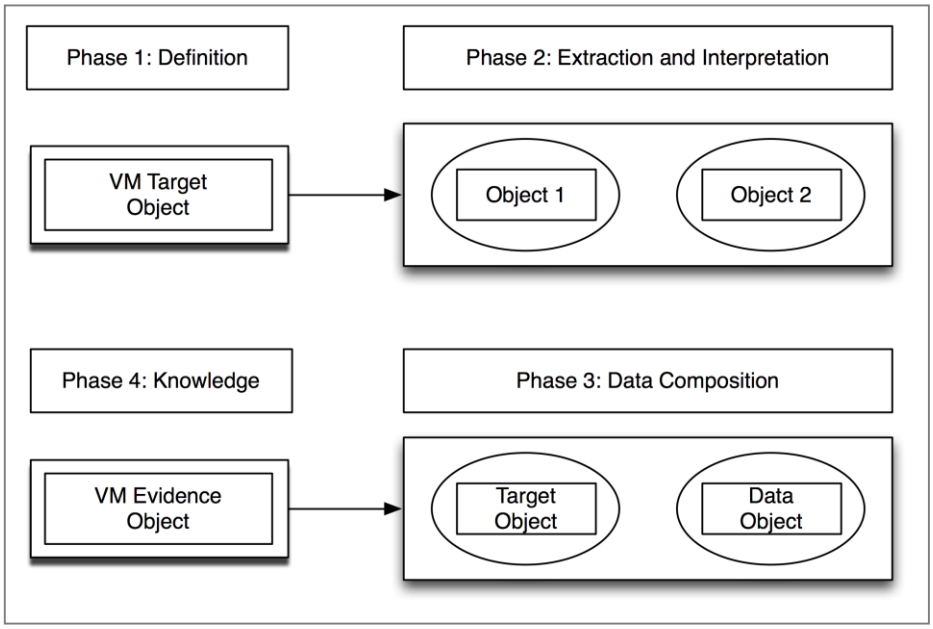

Fig. 3. Graphical representation of the evidence search phase (adopted from [3])

The first phase is to define the target that will be used to locate the evidence. For example, if you are looking for a file named test.txt then the target would have a name called test.txt . If the file you are looking for is called .doc then the target would have ".doc" in the content. The second phase is to extract data from the crime scene in some search pattern and the third phase is to compare the extracted data with the target. After new evidence is found, the fourth phase updates the general knowledge about the investigation so that more targets can be defined.

The target definition process is probably the most challenging of the search phase. Targets are defined either by experience or from existing evidence. Experience from similar cases will help the VM investigator to determine what types of evidence should exist and targets can be defined for them. In a digital crime scene, examples include hidden files and the evidence that such files may contain useful information related to the case.

Targets can also be defined based on the evidence that has already been found. For example if evidence is found in a stored VM log file, then a search may be conducted to find every associated reference of files from which the logs had indexed. If we find keywords within a log file, then such keywords can be used to conduct related searches to find other similar files.

Searches are conducted in an ordered pattern. By analogy for a physical crime scene such searches may reflect geometric shapes and occur in lines or circles. For the virtual machine environment digital crime scene we seek to conduct our searches using interpretation or abstraction layers. Common examples include looking at each file, each sector, or looking at a network packet. Searches that are conducted at different abstraction layers will provide different amounts of data that can be used to recognize the evidence.

If a common class characteristic is used to define the target, then many results may be found and data reduction techniques are needed to determine which is the actual evidence. For example if a search for all files with a ".jpg" extension is performed, then thousands of files may be found, but only a few may be contraband or evidence of an incident.

In the current investigation, the target object is defined as the synchronized VM log file auditor, which in our work this is the analogue to the VM investigator's/system administrator database memory. The VM log auditor browses the heterogeneous file database system that comprises several hybrid domains of virtual servers running over your network data centers. One additional consideration for the log auditor is to perform VM network packet tracing, however this is the subject of independent inquiry. We use these investigations to enable signature analysis against the target. The comparison is done using visualization techniques. Some examples are, looking for a log file that has a certain name or looking for a log file that was modified at a given time. Keyword searches are highly automated tasks and are frequently done to find files and sectors that have a specific value in their content. Hash databases are also used in an automated fashion and can be used to search for files whose content has a specific value.

After a VM object has been identified as evidence, then it must be documented and preserved. The requirements for documentation and preservation will heavily depend on the legal requirements for the court. Even investigations that are conducted without a legal expectation will need basic documentation so that the evidence can be used in the next phase of the investigation. Digital evidence is typically preserved by calculating its MD5 or SHA-1 hash, and making copies of it.

9 Digital Event Reconstruction and Documentation

Having a VM object that has characteristics that reflect possible evidence does not help to answer questions about the incident. To answer questions about the incident, we need to convert the state of the objects into the events that caused the state. For example, if a suspect file is found in the cache of a web browser then we need to develop hypotheses for how it got there. Was it copied there from the command line? Did the web browser place it there when a user viewed a web site? Do other web browsers use the directory? How many persons share the same IP and CPUID address for which the virtual machine in question is connected to? To make conclusions about the suspect file, we need to develop and test hypotheses about the events that it was an effect of, and when applicable, determine what events it could have been a cause of. The goals of this phase are to examine each piece of VM collected evidence and determine what VM events it was involved in so that we can determine which events occurred at the crime scene.

VM digital event reconstruction is a five-phase process as shown in Fig. 4. In the VM evidence examination phase, each of the digital evidence VM objects are examined and identified using their class characteristics and individualized using their unique characteristics. In many cases, the evidence was examined when it was recognized during the search process, but this phase conducts any additional analysis that was not previously done. The second phase, role classification, examines the characteristics of each object and creates hypotheses about what roles the object could have played. For example, an investigator could examine an executable file and conclude that it could be the cause of an event to create a specific file or the cause of an event to open a network connection.

Other files may have evidence that they were the effect of an object, even if the exact cause of the event is unknown. The process was likely conducted on a smaller scale during the search phase.

After all of the VM objects have been examined and their possible roles defined, the third phase, ***event construction and testing*** groups the roles together to form events. Cause and effect roles are grouped together and if other objects must exist for the event to occur then they are searched for. The search may involve the objects that have been collected or it may involve a new search of the crime scene, if it is still available. After possible events have been constructed there may be VM objects that should exist, but could not be found. Hypotheses about the location of these VM objects should be created and justified. Once the objects have been identified, the event is tested. This will determine if the event could have occurred as expected and if the needed evidence exist to show that the event occurred.

After the discrete VM events have been created and tested, the fourth phase, *event-sequencing* orders the events based on their occurrence. Some of these VM objects contain temporal information that can be used to sequence and synchronize two events and other objects contain functional information that can be used to test an incident hypothesis.

The final phase is the hypothesis testing phase where the hypotheses about the entire VM incident are tested using what is known about the events that occurred. The hypotheses must not contradict the VM events for which evidence exist to support it. If a VM hypothesis relies on VM events for which not all VM objects can be found,

then confidence in it must be less than the confidence in the event for which all cause and effect objects could be established.

Digital event reconstruction has not been the focus of traditional forensics let alone consideration for compute cloud forensics. But the issue is becoming increasingly important as apart of most corporate environment security policy. Consider the case in the UK where suspects have used Trojan defense, which is a defense that claims that the computer contains evidence because it was placed there by a Trojan horse and not a suspect [4]. It is no longer acceptable to just identify the existence of a file; rather the source and events that created such a file must also be determined.

After the events have been tested and sequenced, then they must be properly documented so that the final hypothesis and testing can properly represent them. Event chains can be documented by describing the cause and effect VM objects and what hypotheses were needed to describe why some objects were missing. The final point of our theory is used by the physical crime scene investigation, which occurs within the suspected data center(s), which we use to integrate and formulate the results about multiple locations and hence draw final conclusion, even if the results convey the need for yet further investigations.

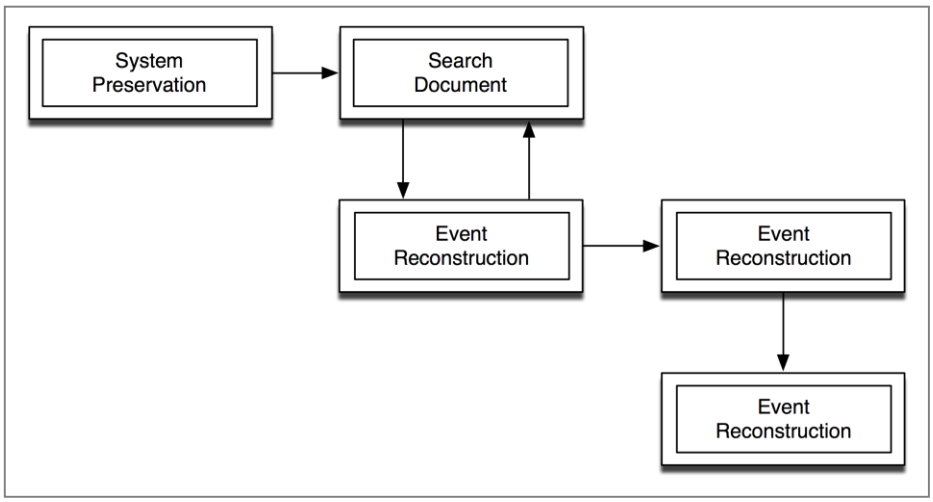

Fig. 4. Graphical representation of the five phases that occur during the event reconstruction phase[3]

10 Comparison to Existing Models

This process model reflects the process that has been used and tested by physical crime investigators. When the area of “Compute Cloud Forensics ” is compared to digital forensics and other traditional forensic sciences, then there are many similarities. Typical forensic science areas answer comparison questions [11]. An

unknown VM object is compared to a standard reference and the scientist determines if they are the same. A VM object is identified by comparing it to several references. The process that occurs in digital forensics involves searching for evidence, identifying it, and reconstructing events. The identification and comparison process is only one part of the big picture and therefore leveraging the physical crime scene investigation procedures instead of forensic procedures seems more logical.

The model proposed in [7] and later by Reith et al. [12] contains many of the same ideas as this model, but in different categories. Our model uses these existing models as templates for building an intuitive VM forensic model that could be adopted for the cloud-computing environment as a part of a general digital investigative framework.

11 Conclusion

In this paper, we have presented a simple framework for the pursuit of a cloud computing digital investigation environment that is based on the causes and effects of VM events. The phases have been organized into the basic requirements of an investigation: namely we need to search for evidence that shows the causes and effects of a VM event and we need to develop hypotheses about such events that occurred at the crime scene. Each phase has a clear goal and requirements, and procedures can be developed accordingly. We have applied the tenets of our model as apart of our case study VM log auditor prototype to corroborate the concerns presented in this framework. We have also presented a clear and detailed set of definitions and concepts that are used in our framework.

References

1. Brian Carrier. Defining Digital Forensics Examination and Analysis Tool Abstraction Layers. International Journal of Digital Evidence, Winter 2003.
2. Tyrone Grandison, E. Michael Maxmillen, Sean Thorpe, and Alfredo Alba. Towards a Formal Model of Cloud Computing, Proceedings of 6th IEEE World Congress on Services, Miami, FL, 2010.
3. Brian Carrier and Eugene H. Spafford. Getting Physical with the Digital Investigation Process. International Journal of Digital Evidence. Fall 2003.
4. Esther George. UK Computer Misuse Act - Trojan Virus Defense. Journal of Digital Investigations, 2, 2004.
5. Houghton Mifflin Company. The American Heritage Dictionary, 4th edition, 2000.
6. Stuart James and John Nordby, editors. Forensic Science: An Introduction to Scientific and Investigative Techniques. CRC Press, 2003.
7. Gary Palmer. A Road Map for Digital Forensic Research. Technical Report DTR-T001-01, DFRWS, November 2001. Report from the First Digital Forensic Workshop (DFRWS).
8. Joseph Rynearson. Evidence and Crime Scene Reconstruction. National Crime Investigation and Training, 6th Edition, 2002.
9. Peter Stephenson. Modeling of Post Incident Root Cause Analysis, International Journal of Digital Evidence, Fall 2003.
10. Technical Working Group for Electronic Crime Scene Investigation. Electronic Crime Scene Investigation: A Guide for First Responders, July 2001.

11. Richard Saferstein. Criminalistics: An Introduction to Forensic Science. Pearson, 7th edition, 2000.
12. Mark Reith, Clint Carr, and Gregg Gunsch. An Examination of Digital Forensic Models. International Journal of Digital Evidence, Fall 2002.