African Journal of Science, Technology, Innovation and Development, 2019
Vol. 11, No. 6, 749–769, https://doi.org/10.1080/20421338.2019.1585675
© 2019 The Authors

CFRaaS: Architectural design of a Cloud Forensic Readiness as-a-Service Model using NMB solution as a forensic agent

Victor R. Kebande [1]* and H. S. Venter [2]

[1] *Department of Computer Science and Media Technology, Malmö University, Malmö, Sweden*
[2] *Department of Computer Science, University of Pretoria, Pretoria, South Africa*
*Corresponding author email: *vitor.kebande@mau.se*

The proliferation of cloud resources among organizations has had numerous benefits with regard to how business processes are conducted. However, despite the benefits, the cloud has not been very resilient due to how it is distributed and its open nature. Due to this, there have been numerous reports on how the security of organizational information has been compromised. In any organization, Digital Forensic Readiness (DFR) is employed as a pre-incident phase whose aim is to maximize the use of Potential Digital Evidence (PDE) while minimizing the cost of performing a Digital Forensic Investigation (DFI). Therefore, it is on this premise that this paper makes a contribution to the architectural design of a Cloud Forensic Readiness as-a-Service (CFRaaS) that uses a Non-Malicious Botnet (NMB) solution as a forensic agent. The authors argue that the architectural design of a CFRaaS is an important aspect, which brings out the requirements that are needed in order for the cloud to be forensically ready for digital investigations when a modified NMB acting as an Agent-Based Solution (ABS) is used. To support this claim, the authors have identified important dependencies and indicators that will provide a synergistic relationship while coming up with CFRaaS design decisions. The main objective of this paper is to present the requirements, design and implementation for achieving DFR in the cloud using a CFRaaS. This study complies with the ISO/IEC 27043: 2015 international standard which presents guidelines for Information Technology, Security Techniques and Incident Investigation Principles and Processes. The result of the study has indicated that it is possible to achieve DFR in the cloud environment using a botnet with modified functionalities.

Keywords: digital, forensic, readiness, architectural, design, requirements, cloud, botnet, NMB

Introduction

The emergence of cloud computing infrastructure has led to dramatic advances and development of powerful networking, storage and information being disseminated across individuals. Due to this, commercial and academic organizations have taken these novel approaches to building their systems under cloud-based technologies. A 2015–2017 forecast in cloud computing highlights five predictions as follows: 35% of new applications will be cloud enabled by 2017, 50% of IT organizations will enforce cloud-management solutions by 2016 and 65% of IT organizations will migrate to hybrid clouds while 11% will move to new delivery models. Lastly, 65% of cloud workloads will comply with data privacy by 2015 (IDC 2015). All these developments are aimed at digitizing and transforming cloud services to virtualized environments.

Despite offering high economic benefits, the security risks that are associated with the cloud are very high because of how open the cloud environment is. In addition, issues like management and control, legislation, regulations, disaster recovery and lack of standardization are some of the concerns within the cloud. Normally, these concerns arise because the Cloud Service Providers (CSPs) control the IT infrastructure and the cloud clients do not have direct access to the resources in the cloud. This has prompted many concerns with regard to how sensitive data is handled for fears of leakage and breach.

Digital Forensics (DF) is a field of science that deals with the process of conducting investigations by using excavated Potential Digital Evidence (PDE) to develop a hypothesis that can be used in legal proceedings to prove or disprove whether a security incident occurred. Due to lack of accepted Standardized Operating Procedures (SOPs) and processes, the cloud is yet to adapt to conventional DF processes. Consequently, conducting digital investigations in the cloud was still compelling at the time of writing this paper.

Discounting that, the problem that this paper investigates is how we can formulate an architectural design of Cloud Forensic Readiness as-a-Service (CFRaaS) that is aimed at making the cloud forensically ready when a Non-Malicious Botnet (NMB) is used as an Agent-Based Solution (ABS) to collect forensic evidence. In this context, the NMB is a modified form of a botnet that acts like a forensic agent that is implemented in the cloud environment on a Software as-a-Service (SaaS) platform.

The main objective of this paper is to propose the design and implementation of a CFRaaS that uses an NMB to gather digital forensic information from a constantly changing cloud environment and digitally preserve it in a forensic database for Digital Forensic Readiness (DFR) purposes. The focus of the paper is on designing the CFRaaS architecture in the best way possible in order for it to meet its main functional purpose. This allows ease of use and proper interoperability of functions, proper integration and communication with stakeholders. Each primary functioning process of the CFRaaS architectural design is identified and each required input and output of the respective CFRaaS model is also represented. To highlight the problem addressed by this

African Journal of Science, Technology, Innovation and Development is co-published by NISC Pty (Ltd) and Informa Limited (trading as Taylor & Francis Group)

paper, the authors consider a hypothetical case scenario on intrusion, information theft, information tampering and framing. The hypothetical case scenario involves a situation whereby a disgruntled employee hacks into a computer security administrator's system and steals confidential information and thereafter wipes traces of evidence. Later on, the disgruntled employee was able to frame the administrator in what led to the arrest of an innocent administrator.

> August 5th was Anthony P. Hopkins first day at work as a computer security administrator for company PQR. PQR dealt with digital electronic supply chain systems that had different trading partners who could conduct e-commerce and B2B under some Service Level Agreements (SLAs). As a security administrator, he had to manage information security aspects of all the retailers' confidential information and transactions and to keep track of possible vulnerabilities that could attack the supply chain process.
>
> Anthony gave his job top-level devotion because of the privacy and security of the IT system's trading secrets, able to connect different trading partners, financial institutions, manufacturers, vendors, associates and other customers. In November, Anthony P got wind of what was a possible security intrusion while performing a system upgrade where a new program called "iscanned" had altered a number of files. The altered files had changed the file formats and the size of the files had increased rapidly. He was concerned because he had kept track of all the programs and the running processes that had been installed in the system on that particular day.
>
> Before Anthony P reported this matter to Manager M, he decided to do a preliminary scrutiny of the system to try and see if he could find any traces. After performing a series of tests, he discovered something very interesting: some critical information had been deleted and he could not trace the origin of the suspect's IP address. By not being able to find any further trace, Anthony P decided to ask Alice who was working as his immediate supervisor. Unfortunately, Alice was a disgruntled employee who had accessed the system remotely and installed a malware that was able to delete, modify, and steal confidential information and thereafter she was able to cover her traces; however, she was not able to cover traces of her IP address entirely. When Anthony P reported the matter to Alice, Alice told him that they could run a scan in order to check for possible causes. Anthony agreed and Alice instead installed another stealth program that wiped her remaining traces and showed the attack might have originated from Anthony's system.
>
> The pilot investigation by the two employees found that there was no PDE that could link the perpetrator to the crime and knowing the obligations involved in consumers' confidential information, Anthony and Alice decided to report the matter to M. M triggered a digital investigation immediately by informing digital forensic investigators and the LEA. A preliminary examination on Anthony's' system revealed that the source was from Anthony's system and he was responsible. Anthony was arrested immediately as digital investigators continued with further forensic investigations. Company PQR is not sure whether their security administrator was responsible for the intrusion and tampering or not.
>
> Why was Alice very comfortable in stealing very confidential information from her employer? In this instance, there was no any forensic process that could collect any valuable information in real time or remotely while the intruder was planting the malware. In any case, was company PQR prepared forensically for any of these incidents? Is it that Anthony could be charged and serve time for a digital crime that he did not commit?

Based on the aforementioned case, it is evident that the security incidents that were pointed out may warrant a Digital Forensic Investigation (DFI). What this paper wishes to determine is how the digital forensic process could have been conducted both with and without the presence of DFR. As a result, the contribution of this paper is presented in three phases. Firstly, we present the proactive approaches for achieving DFR; next, we provide a highlight of the requirements; and thirdly, we present a design of CFRaaS that helps the cloud to be forensically ready for potential security incidents.

The rest of the sections in the paper are structured as follows. The next section describes the background of cloud computing, DF, DFR, and related and previous work. The section thereafter discusses the proactive forensic monitoring approach in the cloud environment. The proposed requirements in order to achieve DFR in the cloud are presented in the next section. This is followed by the section that presents the CFRaaS architectural design. Then comes a section that presents a prototype of CFRaaS implementation, followed by a section on the critical evaluation of the propositions. The final section states a conclusion and suggests future work.

Background

This section gives an overview of the following topics: DF and DFR, cloud computing including related work and previous work. DF is discussed because the entire research is focused on the scientific process of digital investigation. On the other hand, DFR which is a proactive process is discussed to show the need for pre-incident preparation and planning in the DF domain. The cloud is discussed because the whole process occurs within the cloud environment. Related work is also discussed to show the approaches that have previously been applied in DFR.

Digital forensics

Palmer (2001) at the first Digital Forensics Research Workshop (DFRWS) in 2001, defined DF science as

> the use of scientifically derived and proven methods toward the preservation, collection, validation, identification, analysis, interpretation, documentation and presentation of digital evidence derived from digital sources for the purpose of facilitating or furthering the reconstruction of events found to be criminal, or helping to anticipate unauthorised actions shown to be disruptive to planned operations.

This, among other definitions shows that DF is used to prove or disprove a fact/hypothesis in a court of law during litigation, civil or criminal proceedings.

On digital forensic readiness

Digital Forensic Readiness (DFR) concepts were first defined by Tan (2001) as having the objective of maximizing the environment's capability of collecting digital forensic information, while minimizing the cost of the forensic investigation during an incident response. Additionally, ISO/IEC 27043: 2015 defines DFR as a proactive process that precedes *incident detection* and involves *pre-incident planning* within the DF circuit. Figure 1 shows the multi-

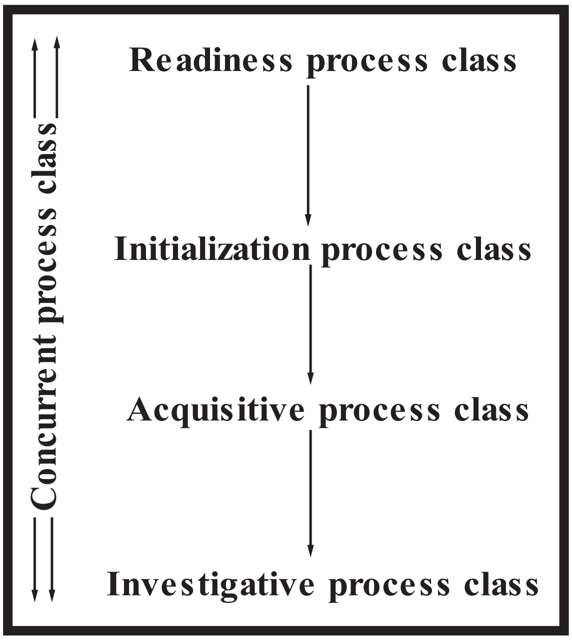

Readiness process class
Initialization process class
Acquisitive process class
Investigative process class
Concurrent process class

Figure 1: Classes of digital investigation processes (ISO/IEC 27043:2015).

tier layer in ISO/IEC 27043 that represents various classes of Digital Investigation Processes (DIP) with *readiness process class* presented using the uppermost process. ISO/IEC 27043 presents readiness as a process that deals with *pre-incident investigation* processes as shown in Figure 1. In this context, readiness as a class is used to define the strategies that need to be employed prior to occurrence of a potential security incident. On the same note, Yasinsac and Manzano (2001) proposed the following: information retention; planning of the response; training; investigation acceleration; prevention of anonymous activities, and protecting the evidence.

Initialization processes trigger the commencement of a digital investigation process, the acquisitive process tackles the physical investigation with availability of PDE while the investigative process uncovers the existence of PDE. Consequently, the concurrent processes happen in line with other processes. Readiness in this context has thus been presented as a group that will be able to maximize the use of potential evidence whilst minimizing the cost of performing a DFI. This includes planning, implementation and assessment processes. To sum up, DFR depicts a process of planning and preparing before potential security incidents can occur. Therefore, it is worth noting that this study is inclined towards the objectives on forensic readiness that have been highlighted by Rowlingson (2004) and Tan (2001).

On cloud computing

The National Institute of Standards and Technology (NIST) has a standard definition of cloud computing which is defined as 'a model for enabling ubiquitous, convenient and on-demand network access to a pool of shared configurable resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction' (Mell and Grance 2011, 2–3). Additionally, cloud computing operates under three service models: Software as-a-Service (SaaS), Platform as-a-Service (PaaS) and Infrastructure as a Service (IaaS) (Mell and Grance 2011). Important aspects according to European Network and Information Security Agency (ENISA, 2009) that makes the cloud ready for DF investigations is to prioritize digital evidence gathering mechanism while the enablers of this mechanism in the cloud are virtualization and how the cloud is distributed. The distribution and open nature of the cloud make it harder to conduct digital forensic investigations unlike in the traditional forensic processes.

Related and previous work

This section presents related work used and previous work that has been presented at research conferences (Kebande and Venter, 2016a; 2016b). To begin, Spyridopoulos and Katos (2011) identified the following requirements that a digital forensic tool should meet in the cloud environment: the tool should identify a digital source using a master server, activate the cloud, be able to create an image or a clone of the digital source and should be executed in the cloud environment. Among other requirements, the authors have recommended additional features based on the NIST specification. The authors' work was generic and based on any reliable digital forensic acquisition tool and not specific to a NMB. In spite of that, Dlamini et al. (2014) identified four basic requirements for the cloud as follows: minimizing the cost and time that is needed when acquiring digital evidence in the cloud environment, ensuring minimal disruption of business process, a demonstration of due diligence and being able to comply with corporate governance, legal and regulatory mandates and, lastly, being able to securely and selectively gather and preserve admissible digital evidence that can be used in legal proceedings in a court of law.

The studies by the abovementioned researchers have highlighted a set of requirements that can be required during acquisition of digital forensic information; however, an architectural design of a system was hardly explored in their study. Additionally, Mouton and Venter (2001) proposed a list of requirements for achieving digital forensic readiness. The target in this case was DFR for IEEE 802.15.4 wireless sensor networks. Figure 2 shows a high-level view of the cloud forensic readiness (CFRaaS) model that was previously proposed by Kebande and Venter (2014a), Kebande and Venter (2016a). The CFRaaS model in Figure 2 allowed proactive planning and preparation of the cloud where a NMB that was used as a forensic agent was able to capture digital information from the cloud environment, which could thereafter be used by an organization to achieve incident preparedness. The captured information is used for DFR purposes, which Rowlingson (2004) describes as a business requirement to be able to employ PDE when needed during litigation.

Proposed proactive forensic monitoring approach in the cloud

In this section, the authors present the approaches that have been used to gather digital forensic information from the

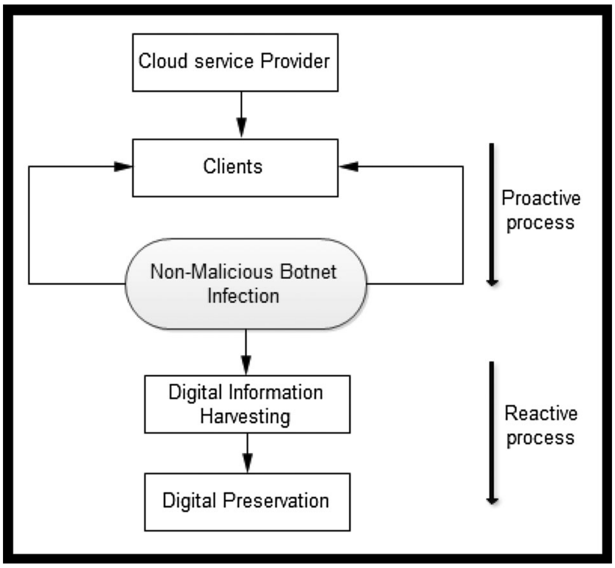


Figure 2: Overview of a high-level view of the CFRaaS model.
Source: Kebande and Venter, 2014b

cloud environment. Proactive monitoring entails gathering information and waiting for potential incidents and then having to respond to the resultant emergencies. This approach enables one to forensically log useful information that can be used to respond to incidents like: event logs, system processes, processor utilization, keystrokes, spam attempts and application services (Kebande and Venter 2017).

High-level view of the approach

Figure 3 illustrates a high-level overview of the proactive approach based on the CFRaaS model that can easily be leveraged by developers. It consists of the following components: *Cloud environment* (labelled 1), *forensic monitoring using a non-malicious botnet* (labelled 2), *digital preservation process* (labelled 3) and an *evaluation of the requirements* (labelled 4). An explanation of the components of the Figure 3 is given further on.

Figure 3 shows an approach for collecting digital forensic information from the *cloud environment* that can be used as potential evidence. This evidence can be used to create a hypothesis that can be used to prove or disprove the occurrence of an incident in a court of law. The rectangle labelled 1 represents the cloud, *monitoring is done by an NMB* that 'infects' virtual instances in the cloud environment in the rectangle labelled 2. It is important to note that infection in this context represents a positive connotation. The forensically captured information is digitally preserved in a DFR approach through the creation of cryptographic hashes as shown in rectangle labelled 3 in a *digital preservation process*. On the same note, the arrow pointing downwards shows the *proactive approach* of planning and preparing for potential security incidents. The circle labelled 4 represents the requirements that have to be met in order for DFR to be achieved in the cloud using a NMB (Kebande and Venter 2016a).

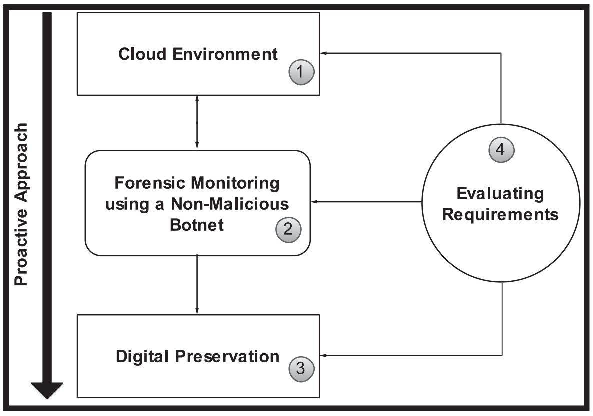


Figure 3: Diagram showing the proposed proactive monitoring approach with requirements.
Source: Kebande and Venter, 2016

Formalism of the Cloud Model (CM)

The authors use the formalism that is based on the actions that are provided between the Cloud Service Providers (*CSPs*) and the cloud clients (*Cl*) to logically model a formal cloud where the CFRaaS model is based (Kebande and Venter 2017). In the cloud environment, there exists interactions between the *Cl*, and the *CSPs*. Logically, the underlying infrastructure between these cloud-based technologies is able to be separated through the concept of virtualization which, according to Gong et al. (2010), is represented as loose coupling. Loose coupling allows different components in a system to interconnect for purposes of interdependence (Kebande and Venter 2017). In this context, the services and applications are offered by the *CSPs* and the *Cl* is able to interact with the cloud servers and the datacentres, keeping in mind that the cloud operates on the client-server architecture. Consequently, the *Cl* does not have any control of the data in the cloud. To present a formal logic model of the interactions and the actions between the *CSPs* and the *Cl* that support the CFRaaS model, the authors describes a Cloud Model (*CM*) that is represented by a *Cl* and *CSPs* as shown in Equation (1).

$$CM = \{CSP_1, CSP_2 \dots .CSP_m\}, [m \geq 1] \tag{1}$$

And

$$CSP = \{Cl_1, Cl_2 \dots .Cl_p\}, [p > 0] \tag{2}$$

Based on Equation (1) and (2) of the *CM*, Gong et al. (2010) highlighted that coupling between entities can be represented as a set. From equation (1) and (2) the *Cl* and *CSPs* are represented as a set as shown.

$$Set(Cl_i, CSP_n)$$

This is then followed by showing the independence of the *Cl* and the *CSPs* which is as shown in Equation (3) and (4) respectively.

$$Cl_i \cap Cl_n = \phi, (0 \leq i, n \leq p, i \neq n) \tag{3}$$

$$CSP_i \cap CSP_n = \phi, (0 \leq i, n \leq p, i \neq n) \tag{4}$$

Therefore, the interconnection between the *Cl* and the *CSPs* that shows how independent each entity is, is shown in

Equation (5).

$$Set(Cl_{i1}, CSP_{n1}) \cap Set(Cl_{i2}, CSP_{n2}) = \phi, [0 \leq i, n \leq p, i \neq n] \quad (5)$$

This shows that the users of the cloud are able to get access to the multiple provisioned services in the cloud at the same time; however, the datacentres remain independent irrespective of the cloud deployable model. A formalization of the cloud architecture is discussed in the next section.

Formalization of the cloud architecture (CA)

In this section, the authors provide formalism based on the entities of the Cloud Architecture *(CA).* This formalism gives a description of the entities that allow the normal operation of the *CA*. Mathematical formulations have also been employed coupled with set theory. Based on the formalism that has been mentioned, a number of definitions are given too.

The *CA* according to Varia (2008) constitutes different designs that are aimed at allowing applications to be built on the underlying infrastructure. The main role of the *CA* in this context is to set a platform through which cloud-based activities can be monitored effectively. Consequently, the *CA* comprises services that are deployed to deliver the roles of a datacentre where the main goals remain reliability, scalability, availability and effectiveness. Additionally, *CA* consists of the following components: Physical server (*Ps*), a Virtual server (*Vs*), Operating System (OS), applications and services (*appns*). The *Ps* comprises of a Datacentre (Dc), while the *Vs* allows the deployment of VMs. Next, the OS allows one to add different *appns* over the internet.

In order to implement DFR in the cloud environment, it is necessary for the cloud to adapt to DF processes; however, the *appns* should also be able to be accessed at the user levels. Based on the *CM*, presented in the previous section, in Equation 1 and 2, the *CSP* comprise a set of clients, *Cl*, which can be represented as shown in Equation (6).

$$CSP = \{Cl_1, Cl_2 \ldots . Cl_i\}, i\varepsilon N \quad (6)$$

From Equation (6), the cloud is normally distributed across datacentres (*Dc*) which can be a limited number that is > 0. This is represented as shown in Equation (7).

$$CSP = \{\{Cl_i\{Dc = \{Dc_1, Dc_2 \ldots . Dc_j\}, j\varepsilon N, Dc > 0\}\}\} \quad (7)$$

In the context of Equation (7), the *CSPs* extend their services to the deployable models in the cloud that are represented as shown in Equation (8).

$$CSP = \left\{ \begin{matrix} Cl_1^{pr}, & Cl_2^{pr}, & \ldots & Cl_n^{pr}, \\ Cl_1^{PB}, & Cl_2^{PB}, & \cdots & Cl_n^{PB}, \\ Cl_1^{HB}, & Cl_2^{HB}, & \ldots & Cl_n^{HB}, \end{matrix} \right\} Cl \geq 1 \quad (8)$$

where *Cl* is the client that represents the users of the cloud, *Pr* represents the private cloud deployable model, *PB* represents the public cloud deployable model and *HB* represents the hybrid deployable model. Services are deployed to one or more clients as shown in Equation (8).

A *Dc* constitutes a network that has a *Vs,* which allows the deployment of the VM. Apart from that, there exists the *Ps* that is able to give support to the OS. Therefore, *Dc* is composed of entities *Ps, Vs* and OS respectively. Together with the *CSP*, this is represented as shown in Equation (9) and (10) respectively.

$$Dc = \{Ps, Vs, OS\} \quad (9)$$

$$CSPs = \{Cl_i\{Dc_j\} = \{Ps, Vs, OS\}, j\varepsilon N, Dc > 0\}\}\} \quad (10)$$

Ps and *Vs* are able to support a number of cloud resources (R_n) and VM_n while the OS is able to support $appns_n$ as shown in Equation (11), (12) and (13) respectively.

$$Ps = \{R_1, R_2 \ldots . R_n\} \quad (11)$$

$$Vs = \{VM_1, VM_2 \ldots . VM_n\} \quad (12)$$

$$OS = \{appns_1, appns_2 \ldots . appns_n\} \quad (13)$$

Therefore, the overall logic formulation for the entities of the cloud is given as shown in Equation (14).

$$\mathrm{CSP} = \{Cl_i\{Dc_j\} = \left\{ \begin{matrix} Ps = \{R_1, R_2, \ldots R_n\} \\ Vs = \{VM_1, VM_2, \ldots VM_n\} \\ OS = \{appns_1, appns_2 \ldots appns_n \end{matrix} \right\}, \; j \in N, Dc > 0\}\} \quad (14)$$

where $Ps = \{R_1, R_2 \ldots . R_n\}$represents a number of cloud resources, $Vs = \{VM_1, VM_2 \ldots . VM_n\}$ represents the virtual machines that are able to support virtualization. Finally, $OS = \{appns_1, appns_2 \ldots . appns_n\}$ represents the application and services in the cloud environment. Basically, the formalism of the *CA* provides a theoretical approach that is aimed at making the cloud forensically ready for DFIs. The entities *Ps, Vs* and *OS* have been employed to help in the execution of cloud services. In the next section, the reader is introduced to the requirements and capabilities of the CFRaaS.

CFRaaS model formulation

In this section, a method for formulating the CFRaaS model is given. Based on the CFRaaS requirements, the systematic formulation of the CFRaaS model is formalized in terms of the specifications that follow: the CFRaaS model is composed of *CSPs*, clients and PDE that is harvested in a DFR approach. Firstly, we define a domain that represents a *CSP.* The *CSP* provides services to clients that are represented as a set of activities *Ac*

(Kebande and Venter 2016a, 2016b, 2017). This can be represented by the following equation:

$$CSP = \{Ac_1, Ac_2, \ldots\ldots Ac_n\} \tag{15}$$

Forensic logs that exist as PDE are captured based on the timeline of activities. It is essential to perform analysis on forensic logs that exist as digital evidence when they are captured from different cloud sources before a DFI is conducted. Therefore, forensic log files can be represented with their respective tag names and the time the activity occurs. This can be represented using the following equation:

$$< (Lt_1, To_1), (Lt_2, To_2) \ldots\ldots\ldots\ldots\ldots (Lt_n, To_n) > \tag{16}$$

where Lt_i is used to denote the name of the log or the tag name and To_i denotes the number of times that Lt_i occurs. This is the occurrence of a particular log file in the cloud. This implies that in each particular activity that Ac generates, one or more forensic log files can be used as PDE. This is represented by the following equation:

$$Ac_i = \{Lt_{i1}, Lt_{i2} \ldots\ldots\ldots\ldots\ldots\ldots\ldots Lt_{ip}\}, i^{\varepsilon}[1, n] \tag{17}$$

When PDE is captured from a client c in a given environment, then i represent the origin or the source that the forensic log is extracted from. Additionally, a series of Potential Security Events (PSEs) with different attributes at a given time may be registered within the forensic logs. The existence of these events may be represented by the following equation:

$$Lt_{ij} = \{e_{ij1}, e_{ij2,} \ldots e_{ijm}\}, i^{\varepsilon}[1, n], j^{\varepsilon}[1, p] \tag{18}$$

where, e_{ij} is a PSE that may be listed under an extracted forensic log. The attribute at for an event ei may be represented by varying records that may include: timestamp (t), occurrence (X) and size (s).This is represented by the following:

$$E_{ijq} = \{at_{ijq1}, at_{ijq2}, \ldots at_{ijqk}\}, i^{\varepsilon}[1, n], j^{\varepsilon}[1, p), k^{\varepsilon}[1, m] \tag{19}$$

Based on formulations that are represented in Equations 1, 2, 3, 4 and 5, the components of the CFRaaS model have been formalized. Therefore, the equation representing the CFRaaS model is represented as:

$$CSP = \{Ac\{Lt\{e_{ij}\{at_{ij}\}\}\}\} \Leftrightarrow dp \tag{20}$$

where, dp has been used to represent the process of preserving PDE digitally. In the next section the CFRaaS model requirements are discussed.

CFRaaS model requirements

The requirements proposed in this paper are aimed at facilitating a DFR process, which involves collection of relevant PDE from the cloud environment that is related to digital crimes. Nevertheless, it is the authors' opinion that the requirements play a significant role in enhancing effective DFR process in the cloud environment without having to modify the functionalities of the existing cloud architecture. On the same note, the authors have concentrated on introducing the unique requirements that can support the CFRaaS that was proposed by Kebande and Venter (2014a) in previous research.

Proposed CFRaaS model requirements

A description of the requirements and the specifications of the CFRaaS are highlighted in this section that shows the capabilities of CFRaaS. The authors will employ use-cases and a process view diagram to bring out the requirements needed, in order for the forensic tool to be able collect evidence that can be presented in a court of law. From a cyber-criminal perspective, using the cloud to conduct digital crimes is more advantageous because a cyber-criminal can go unnoticed easily. This is because of challenging cyber-investigation processes brought about by the inadaptability of DF techniques in the cloud. Apart from that, cross-cutting jurisdictions, multi-jurisdiction and lack of a common international law for cross-border cyber-investigation are a challenge too. Furthermore, locating data provenance poses a challenge too, owing to the fact that a server may be located in a completely different territory or jurisdiction. More so, the distribution of data centres may also act in favour of a cyber-criminal in some cases. In order for the Law Enforcements Agencies (LEAs) and the DFIs to launch a DF investigation in the cloud, the CSP should employ the CFRaaS. This will only make the cloud forensically ready by collecting useful information for digital investigation purposes. The requirements discussed in the context of this research paper are aimed at creating a relationship between the CSPs, DFIs and the LEA. The requirements proposed in this research paper are categorized into two: architectural requirements and general requirements; an explanation is given further on in this paper (Kebande and Venter 2016a).

Architectural requirements

The purpose of this section is to propose architectural requirements for the CFRaaS model. This allows DFR processes to be achieved effectively in the cloud environment without having to modify the functionality of the existing cloud architecture. The CFRaaS's architectural requirements are divided into two: five functional requirements (FR) and five non-functional requirements (NFR) as shown in Table 1. FR are shown on the left side of Table 1 while NFR are shown on the right side of Table 1. Additionally, FRs were discussed in the section 'Functional requirements' and NFR in the section 'Non functional requirements', respectively (Kebande and Venter 2016a).

Functional requirements

In this section, the authors present a description the system architecture's FRs. Mainly, this represents the primary system requirements. According to Pohl (2010), FRs are statements of services that the system is supposed to provide. Furthermore, Pohl (2010) highlights that FRs

Table 1: Functional and non-functional requirements.

	Functional requirements		Non-functional requirements
1	Standard implementation of DFR in the cloud	1	Scalability
2	Collaborative with legally competent bodies	2	Security
3	Event reconstruction	3	Usability
4	Incident response procedure	4	Flexibility
5	Efficacy and ease of use	5	Auditability

can also be seen as a way that a system is able to react when it receives particular inputs in different scenarios. Table 1 shows the list of functional requirements that the CFRaaS model deals with.

Standard implementation of DFR process in the cloud: CFRaaS allow standardized DFR processes to be implemented in a cloud environment. Furthermore, standard implementation of DFR in the cloud allows captured evidence to be accepted as admissible evidence. The proposed DFR process complies with the standard of ISO/IEC 27043. It allows the use of a software application as a forensic agent in a proactive approach to collect PDE, which can be used to prepare and plan for potential security incidents in the cloud environment. The benefit of using this process, is that there is no modification, alteration or tampering of the functionalities of the existing cloud architecture. This is simply because all DFR activities are conducted outside the cloud environment. Nonetheless, the process also adheres to a standard process that is highlighted in ISO/IEC 27017. ISO/IEC 27017 is a standard that provides guidance in information security aspects that are based on cloud computing. It deals with information technology and security techniques. In this case, the CSP and the cloud client should come to terms on the different roles and responsibilities that are played by each party.

Collaborative with competent legal bodies: CFRaaS allows interaction, collaboration with Law Enforcement Agencies (LEA), DFIs and other competent investigation bodies within a given jurisdiction. One is able to uphold competent investigations by maintaining the chain of custody through the collaboration of distributed DF investigators. The DF legal framework should provide rules that allow admissibility of digital evidence, if it is lawfully admitted in a court of law during trial. Although there is no full control of the processes and the evidence in the cloud, the architecture should be relevant and have scope that enables collaboration with regard to identification of digital artifacts.

Event reconstruction: CFRaaS has to allow various sequences of events to be examined and analyzed before a hypothesis that may be used in a court of law is created. The characteristics that PDE portray and the manner of occurrence of potential security events should explicitly be presented through a reconstruction approach. By supporting event reconstruction, efficiency of analysis will be improved thereby increasing the chances of admissibility.

Incident response procedures: CFRaaS should allow Incident Response Procedures (IRP) through collecting and preserving digital evidence. IRP ensures that PDE interpretation is ideal during the investigative process before admissibility in a court of law. IRP are digital forensic tasks that are associated with competent bodies. One should be able to conduct IRP when there is availability of digital evidence that is captured through DFR process. The IRP contains instructions for detection and response to potential security incidents.

Effectiveness and ease of use: CFRaaS allows effective communication between the tasks which are provided through an interface when preparing the cloud for digital investigation. This helps to prepare for security incidents in a shorter time. It also must be very user friendlier and simple whenever users interact with it.

Non-functional requirements

According to Malan and Bredemeyer (2001), NFR is used to describe various constraints and qualities that stakeholders are interested in on a system. This means that NFR has the capability of affecting stakeholders' degree of satisfaction, which eventually implies that the NFR should be prioritized in any system. As a result, investigation by the authors identified the following as the NFR that the CFRaaS architecture has to meet. A summary of the NFR is also given in Table 1.

Scalability: The process of conducting DFR in the cloud should be able to accommodate the demand of users and the forensic processes. The CFRaaS architecture should be sound enough to meet the needs of emerging processes with the least time possible. Additionally, the CFRaaS architecture should be able to withstand overstraining and tolerate errors. If the system cannot scale to business volume, then obstacles may arise that may hinder the forensic readiness process.

Security: Security as a requirement ensures that the forensic services are prevented from potential attacks. This requirement ensures that the NMB solution is protected from other malicious attacks that might want to infiltrate or defeat the purpose of the NMB. Tampering of digital evidence is one aspect through which the security of the system might be compromised. If security is not enforced at this level, forensic evidence contamination, tampering or theft might be experienced. On the same note, Richter, Kuntze, and Rudolph (2010) highlighted that for digital evidence to be admissible in a court of law, the system must be secure and the data that is within must be authentic and possess integrity.

Usability: For any system to be effective it must be tangible and relatively easy to use. The forensic processes and tasks that are based on this architecture should be easy for novices to learn. This should also apply to other forensic experts, where the CFRaaS should allow users to gain an understanding of exactly what the intent of the system is. This allows different collaborating legal bodies in different jurisdictions to interact well, as well as any other casual user. If usability is not enforced properly the performance objectives and forensic tasks that a user may want to perform might be hindered.

Flexibility: Carrier and Spafford (2004) highlighted that for a framework to support future technologies it should be sufficiently flexible. Based on this study, a forensic process should incorporate flexibility. This enables proper execution of digital forensic tasks and the ability to incorporate other investigative technologies at the same time. Apart from that, flexibility should show the ability to support system processes by reacting quickly to internal and external changes.

Auditability: One of the CFRaaS model requirements that was discussed earlier in this paper is the ability of the system to perform forensic logging. The captured forensic logs are then isolated and used as PDE. Once the investigation has been closed, one should be able to perform an audit of the processes conducted by the system – either during or before PDE presentation. Irrespective of that, auditing may also be conducted by using the reconstructed events as mentioned in the CFRaaS model. Therefore, it is a requirement that an audit must be conducted after forensic processes have been completed.

General requirements

This section provides a discussion of the proposed requirements as a contribution to how an agent-based solution can be used to conduct digital forensic readiness in the cloud environment. The primary objective of the requirements is to portray a relationship between the NMB solution and the different jurisdictions that govern different cloud models. The requirements provided in this section have been used to analyze and specify the cloud forensic readiness (CFRaaS) model that was proposed initially by Kebande and Venter (2015a). The requirements are aimed at allowing the NMB to be deployed in a scalable environment to collect PDE. Furthermore, they will ensure that there is a functional relationship between the structural components of the model. A summary of the requirements is given in Table 2 and explanations are given thereafter.

Table 2 shows a summary of the proposed general requirements that should be taken into consideration in order to achieve DFR in the cloud when an NMB is used. Forensic logging allows the CFRaaS model to collect and manage potential evidence, hashing is done to captured digital evidence to maintain integrity, all events and forensic logs should have a timestamp to prevent tampering with evidence. Characterization as described by Kebande and Venter (2015d) allows one to isolate PDE based on the causality and characteristics during DFR approach. After characterization, forensic activities like manipulation and computations of PDE happen outside the cloud which renders the functionality of existing architecture hard to modify thus saving cost and time. The NMB solution is protected from other malicious activities through encryption and deployment in a trusted cloud. Deterrence of the NMB is done through obfuscation; this was highlighted by Kebande and Venter (2015b) as changing the NMB's pattern to a nonsensical pattern. Event reconstruction has been presented in CFRaaS model as a way of reconstructing events based on similarity measure (Kebande and Venter 2015d, 2016b). This approach should be sensitive to the local laws of a given jurisdiction and law enforcement requirements. Finally, reporting provides examination

Table 2: Table showing the proposed requirements for achieving DFR in the cloud environment (Kebande & Venter, 2016).

	Requirement	Summary
1	**Forensic logging capability and management**	Forensic logs to be used as digital evidence should be captured in a virtualized environment. It is important to know how logging is done, what is logged and when to log.
2	**Integrity and authenticity**	The retained digital evidence should be digitally preserved. Verification authenticity should be possibe if there is a need for a digital forensic investigation.
3	**Timestamping**	Each log should have a timestamp to in order to prove its integrity. All events and activities should have time stamp representation.
4	**Digital evidence characterization**	Digital evidence should be grouped in respective file formats for possible incident identification. Activity analysis should be conducted to isolate potential security incidents.
5	**Non-modification of existing cloud architecture**	Functionalties of existing cloud architecture are not modified or tampered with because evidential activities are conducted outside the cloud environment. Activities like computation of evidence and analysis are conducted outside the cloud environment.
6	**Security implementation**	The software application solution should be protected from other malicious activities. The software application should be deployed in a trusted cloud environment.
7	**Obfuscation**	Software application's patterns are changed in a nonsensical manner to deter surveillance to avoid defeating its purpose. Obfuscation is enforced for privacy reasons.
8	**Event reconstruction**	A hypothesis that should prove a fact in a court of law should be developed based on events. Structure and occurrence of events should be easily distingushed.
9	**Legal requiements**	The legal perspective and provisons across diverse jurisdictions should be known prior to a digital forensic investigation.
10	**Forensic reporting**	A readiness report that shows the interpretation process as a result of digital evidence retention should be generated.

notes through interpretation of what the requirements have achieved in the model.

CFRaaS architectural design

Overview

This section proposes a novel contribution towards the architectural design of the CFRaaS. It is organized into five structures: the high-level structure in Figure 4 and more detailed functional structures in Figures 5–8 respectively. At a later stage, the CFRaaS architecture is presented using use-cases and process views.

High-level overview of CFRaaS architecture

An organization of four distinguishable layers is shown by the high-level structure in Figure 4. The layers (labelled 1–4) include: Provider layer, Cloud Layer, Digital Forensic Readiness Layer (DFRL) and Incident Response Procedures (IRP) layer. Each of these layers is described below in detail.

Figure 4 shows the high-level architectural diagram of CFRaaS. Sensitive and critical information that is related to digital crimes is captured using an NMB in a proactive process from the cloud environment (labelled 2) as PDE by the CSPs in the provider layer (labelled 1). This was described in a previous (see the section 'Related and previous work'). The captured PDE is digitally preserved in a forensic database, then pre-analyzed for possible incident detection purposes in a DFR approach layer in process (labelled 4). Finally, the incident response layer (labelled 4) is a reactive process that allows proper examination and analysis on PDE by DF investigators and LEAs. More details on the high-level CFRaaS architecture are explained in later sections.

Provider layer

The provider layer (PL) is the business layer that comprises the services that are provisioned by the cloud service providers (CSPs) over the internet. In this layer, convenient, secure and reliable services are provisioned to different cloud clients under properly agreed-upon service level agreements (SLAs). An SLA is a contract that explicitly states the services that a provider will give, and the standard of the services being provided. Implementing an SLA ensures that forensic monitoring process is effected while the client's data is protected at the same time. In spite of that, potential digital information that is related to crimes is captured through monitoring. This is achieved through the deployment of an NMB as a forensic agent that is able to infect Virtual Machines (VMs) in the cloud environment. This happens through a well-governed, secure virtual administration process. The processes in PL is shown in Figure 4. An explanation on the relationship of each component is provided in the next paragraph.

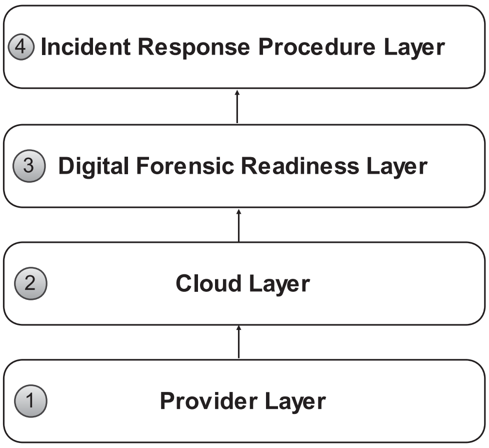

4 Incident Response Procedure Layer
3 Digital Forensic Readiness Layer
2 Cloud Layer
1 Provider Layer

Figure 4: High-level overview of the CFRaaS.

Considering the PL in Figure 4, individual cloud roles and services (labelled 1) are able to be deployed to different consumers depending on the cloud model and the business requirements that suit a given company. Nevertheless, depending on the type of workload, the services may be deployed by creating, handling and managing the number of role instances. Apart from that, service orchestration allows a well-planned provisioning and automation of different DFR tasks in the cloud environment that solely relies on the rules for collecting PDE. Through automation, a secure forensic monitoring is enhanced by managing the configuration of VMs and forensic databases.

Virtualization (labelled 2) in Figure 5 in the CFRaaS is an enabler that ensures that resources are scaled within the cloud environment. It gives room for separation of VMs from the physical infrastructure which allows PDE to be captured. The VM is isolated in a runtime environment which may be the Operating System (OS) or applications. Delport, Köhn, and Olivier (2011) highlighted that isolating an instance for forensic investigation helps in preserving integrity of forensic evidence. An advantage of using virtualization is that it enhances the forensic monitoring process in multiple sources. Moreover, in this context, the virtualized resources are provided as services within the cloud environs.

Access control (labelled 2) governs the control of DFR tasks and processes in the CFRaaS architecture. Through access-control, the CSP is also able to meet the corporate security and privacy requirements by providing secure access to all the PDE that is captured from the cloud environment. This can only be achieved through enforcing comprehensible and enforceable SLAs that provide security assurance during the forensic monitoring process. CSPs should provide administration of the process of deploying an NMB through determining the users that are allowed to execute tasks and commands within the CFRaaS architecture. Access control maintains the DFR User Functions (DUF) by maintaining a number of CFRaaS login procedures for managing evidence accessibility.

In the next step (labelled 3) an NMB that is used as a forensic agent is deployed to collect PDE as Software as a Service (SaaS) in the cloud layer. In this process, a proactive activity of gathering critical, sensitive and unstructured data related to crimes from heterogeneous sources happens in the cloud environment in a forensic logging process (labelled 4). It should be noted that the functionality of the existing cloud architecture is not modified because all the captured data is manipulated outside the

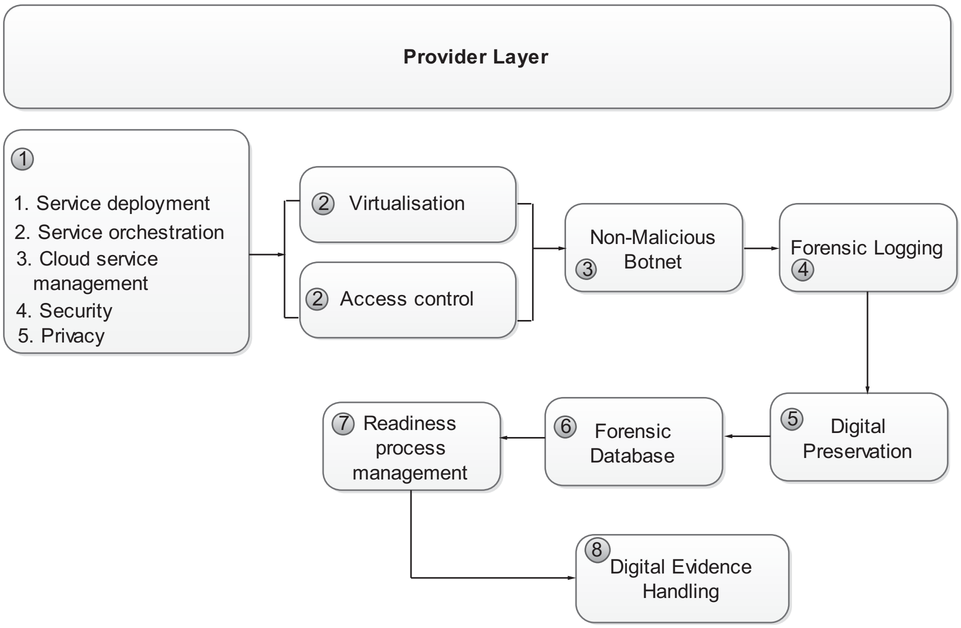


Figure 5: Illustrating components of the provider layer.

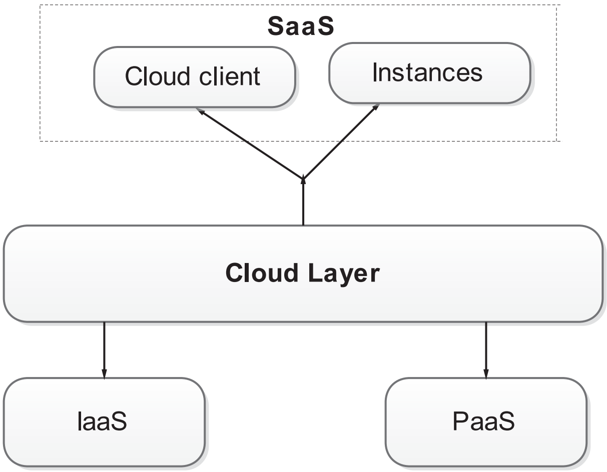


Figure 6: Cloud layer of CFRaaS.

cloud. Examples of digital data captured from the cloud include: CPU usage, RAM usage, keystrokes, database activities, virtual images, system logs, audit logs, hypervisor error logs, network logs, VM images, and so forth.

Lastly, in the PL, DFR processes (labelled 5–8) are handled at this level. This includes digital preservation, forensic database, readiness process management and digital evidence handling readiness process. These processes are aimed at coordinating PDE handling through managing the operations to ensure effective and stable operation.

Cloud layer

The cloud layer provides elasticity and flexibility by allowing virtual forensic administration in the IaaS of the NMB solution that is implemented as the SaaS cloud model. The cloud layer consists of the following deployable models: SaaS, IaaS and PaaS, which are provisioned by the CSP. Forensic logging that is managed by the provider layer happens in the cloud environment. The NMB is executed in the VM, which enables collection of digital evidence. Collection of digital evidence occurs based on existing laws and privacy policies and procedures within a given jurisdiction. The cloud layer is shown in Figure 6.

Following forensic monitoring discussed in the preceding section, Figure 6 shows the cloud layer clients. Potential evidence is captured from the clients' instances. This is achieved through the deployment of an NMB solution in the SaaS computing model. Forensic administration of what is logged is administered through the IaaS. The PaaS provides a ready environment for the NMB development. In the next section the reader is introduced to the DFR layer.

Digital forensic readiness layer

Digital Forensic Readiness Layer (DFRL) is presented as a proactive process that happens before incident detection (see Figure 7). In the readiness process groups of the ISO/IEC 27043, DFR is tasked with planning, implementation and assessment readiness activities. Planning process group as a readiness process is used to oversee for the following readiness activities: defining the scenario, *identifying PDE sources*, planning of *pre-incident collection* processes, *data handling and storage*, *planning pre-incident analysis, incident detection planning* and a definition of the CFRaaS system architecture (ISO/IEC 27043:2015).

Implementation Process Group (IPG) implements the processes of CFRaaS system architecture, *implements PDE gathering, PDE storage* and how data that represents evidence is handled. IPG also deals with *pre-incident analysis* on PDE and the implementation of the process of *incident detection.* Assessment Process Group (APG) of DFRL is concerned with the assessment of the IPG. The main tasks of AGP are to assess the IPG and to

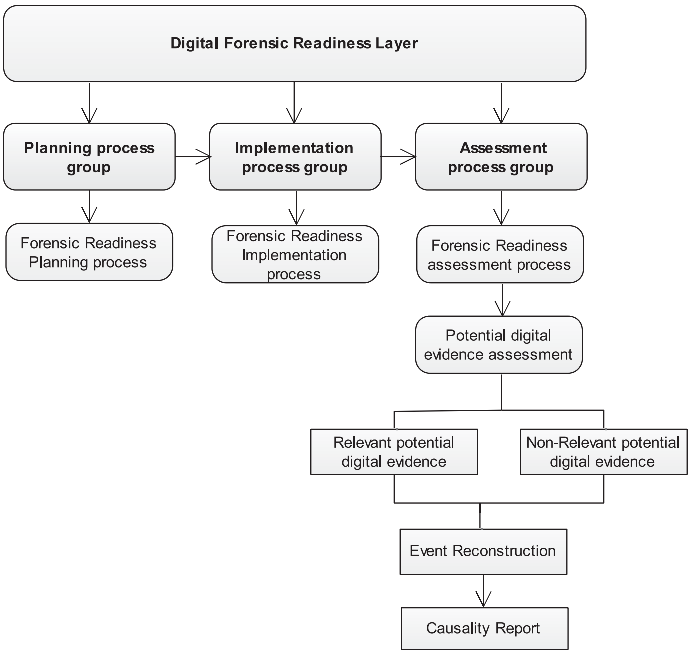

Digital Forensic Readiness Layer
Planning process group
Implementation process group
Assessment process group
Forensic Readiness Planning process
Forensic Readiness Implementation process
Forensic Readiness assessment process
Potential digital evidence assessment
Relevant potential digital evidence
Non-Relevant potential digital evidence
Event Reconstruction
Causality Report

Figure 7: Digital forensic readiness layer.

implement the results of the AGP. PDE assessment is aimed at performing evaluation of the captured evidence to increase the chances of detection. Reconstruction is done to examine the characteristics of relevant potential evidence that may help to generate causality. Causality acts as a step towards creating a hypothesis that can be used to prove or disprove a fact in a court of law.

Incident response layer

Casey (2011) highlighted that whenever there is suspicious behaviour that culminates in a potential security incident, then Incident Response Procedure (IRP) becomes necessary. IRP presents the steps that are followed to tackle incidents by Law Enforcement Agencies (LEAs) and how to collaborate with competent bodies. IRP is a reactive process, which is not part of the DFR functions, but it subsequently occurs after incident detection. It's the actual process of DFI.

In this context, IRP consists of initialization, acquisitive and investigative processes as shown in Figure 8. These processes were highlighted in the comprehensive and harmonized digital forensic investigation process model by Valjarevic and Venter (2015). Initialization deals with the inception of the digital investigation process. Initialization represents *incident detection, first response, planning* and preparing for *post-incident response*. Note that the concurrent processes that were discussed previously are implemented alongside the IRP layer.

On the other hand, the acquisitive process *identifies PDE* and *evidence collection*, and then it follows the process of *acquisition*, *storage*, *transportation and preservation of PDE*. Finally, the investigative process performs the following: PDE examination and analysis, interpretation, reporting, presentation of digital evidence and investigation closure. IRP end-users in this context are the DF investigators and Law Enforcement Agencies

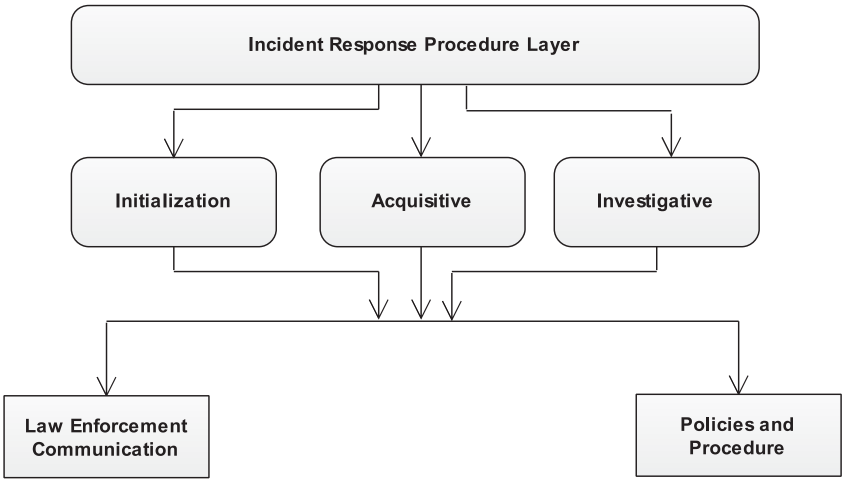

Incident Response Procedure Layer
Initialization
Acquisitive
Investigative
Law Enforcement Communication
Policies and Procedure

Figure 8: Incident response procedure layer.

(LEAs) as shown in Figure 8. IRP relies on DFRL to conduct a DF investigation where the policies and procedures should be followed.

Process view

This section presents the events view that used to show the capabilities of CFRaaS. This is presented using a use-case that shows different scenarios as requirements that a *forensic readiness administrator* and *forensic investigator* meet while implementing DFR and IRP activities. The use-cases that are shown in Figure 9 are processes that have been standardized as highlighted in previous literature. They allow the CFRaaS to be implemented in a standardized approach for effective planning before potential security events can occur. The use-cases illustrate two actors that include a *forensic readiness administrator* and a *forensic investigator.*

The processes involved (*Readiness Planning, Implement Readiness* and *Deploy NMB* and *Assess Readiness*) have also been mapped to the standard of ISO/IEC 27043: 2015. Apart from those, the six use-cases in the *Readiness Planning* process comply with the ISO/IEC 27043: 2015 standard. These processes are directly used by the *forensic readiness administrator* while planning for forensic readiness activities. The use-cases include: *Observe Concurrent Processes, Define the Scenario, Identify Evidence Sources, Plan Pre-incident Gathering, Store Data as Evidence and Plan Incident Detection*. Based on these processes, a *forensic investigator* is able to interact with the processes in the system.

The *Readiness Process Management* process that is shown in Figure 8 is extended by seven use-cases that are used as techniques for achieving forensic readiness. The process begins by deploying a forensic agent that has the capability of conducting forensic logging. The use cases include: *Observing Concurrent Processes, Forensic Logging, Performing Digital Preservation, Store Evidence, Conduct Evidence Assessment, Reconstruct Events* and Identification of *Causality.*

An assessment of readiness processes is a process that is used to check if the readiness processes have been implemented. The process consists of four different use-cases namely: Verify implemented readiness process, Observe concurrent processes, Generate examination notes and Presentation of a readiness report (Figure 10).

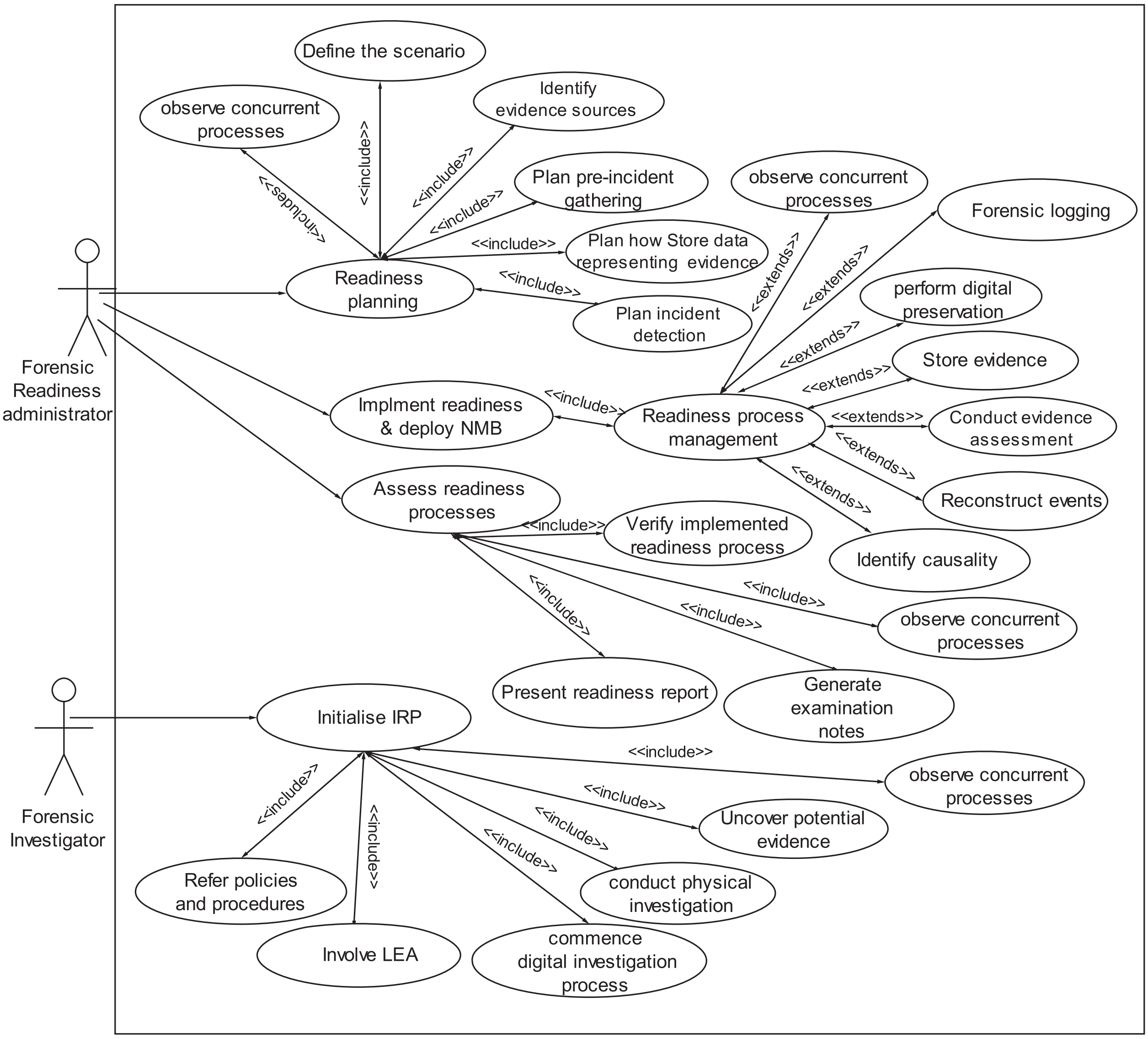

Define the scenario
observe concurrent processes
Identify evidence sources
<<include>>
<<include>>
<<includes>>
Plan pre-incident gathering
<<include>>
observe concurrent processes
Forensic logging
<<include>>
Plan how Store data representing evidence
<<include>>
Readiness planning
Plan incident detection
<<extends>>
<<extends>>
perform digital preservation
<<extends>>
Forensic Readiness administrator
Store evidence
<<extends>>
Implment readiness & deploy NMB
<<include>>
Readiness process management
<<extends>>
Conduct evidence assessment
<<extends>>
<<extends>>
Reconstruct events
Assess readiness processes
<<include>>
Verify implemented readiness process
Identify causality
<<include>>
<<include>>
<<include>>
observe concurrent processes
Present readiness report
Generate examination notes
Initialise IRP
<<include>>
observe concurrent processes
Forensic Investigator
<<include>>
<<include>>
<<include>>
Uncover potential evidence
<<include>>
<<include>>
conduct physical investigation
Refer policies and procedures
Involve LEA
commence digital investigation process

Figure 9: Event view of CFRaaS use cases.

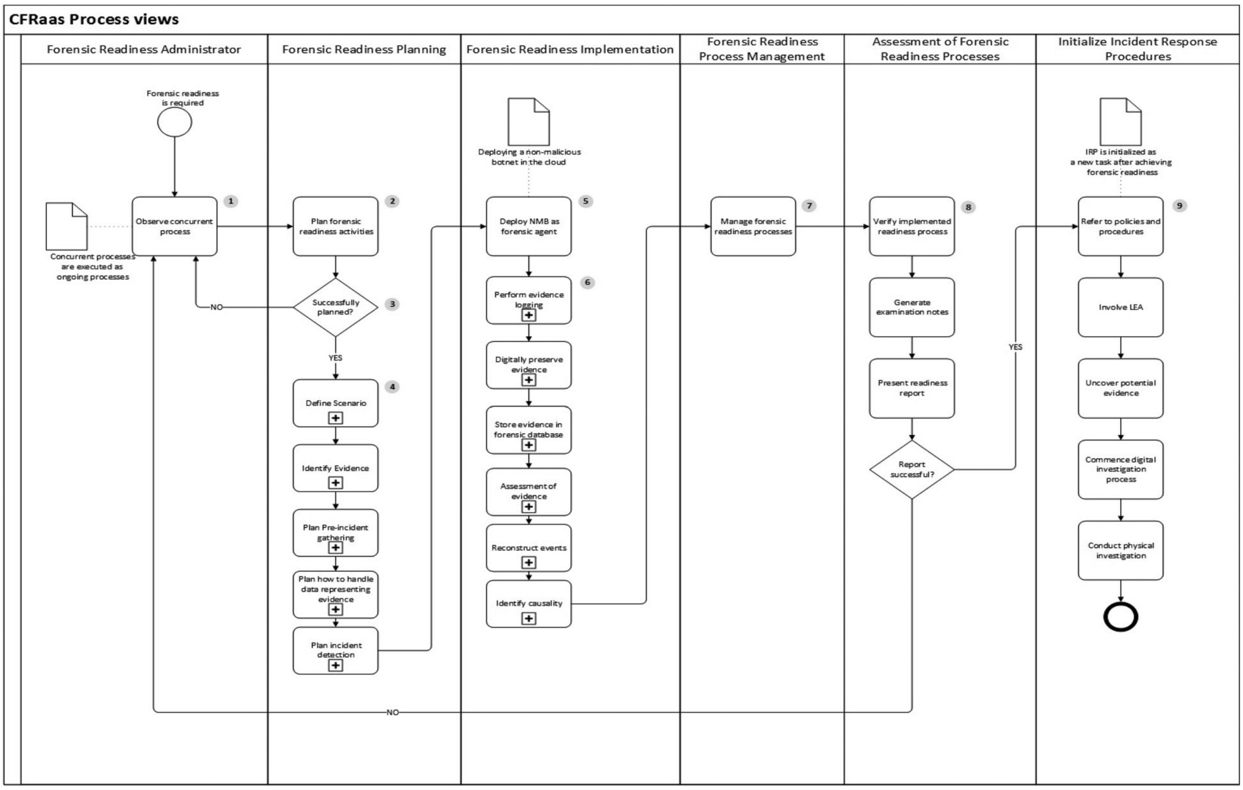

CFRaas Process views
Forensic Readiness Administrator
Forensic Readiness Planning
Forensic Readiness Implementation
Forensic Readiness Process Management
Assessment of Forensic Readiness Processes
Initialize Incident Response Procedures
Forensic readiness is required
Concurrent processes are executed as ongoing processes
Observe concurrent process
1
Plan forensic readiness activities
2
Successfully planned?
3
NO
YES
Define Scenario
4
Identify Evidence
Plan Pre-incident gathering
Plan how to handle data representing evidence
Plan incident detection
Deploying a non-malicious botnet in the cloud
Deploy NMB as forensic agent
5
Perform evidence logging
6
Digitally preserve evidence
Store evidence in forensic database
Assessment of evidence
Reconstruct events
Identify causality
Manage forensic readiness processes
7
Verify implemented readiness process
8
Generate examination notes
Present readiness report
Report successful?
YES
NO
IRP is initialized as a new task after achieving forensic readiness
Refer to policies and procedures
9
Involve LEA
Uncover potential evidence
Commence digital investigation process
Conduct physical investigation

Figure 10. CFRaaS process view.

A forensic investigator interacts with the system by executing the IRPs. The main task of a forensic investigator is to initiate a DF investigation process. CFRaaS allows the LEAs, policies and procedures including concurrent processes to be adhered to while initializing IRPs. Some of the use-cases involved in the IRP process include: observing the concurrent processes, uncovering potential evidence, conducting a physical investigation, commencing the digital investigation process, involving the LEAs and reference to policies and procedures. Nevertheless, CFRaaS allows proactive preparation and preparing for security incidents and it also shows the approach of the reactive IRP process. In the next section, a sequence of executions of events and interaction is presented.

Procedural flows for the CFRaaS architecture

Figure 11 shows procedural flows of the CFRaaS architectural diagram. Its aim is to assist DFIs and LEAs, reducing the need to conduct forensic investigations in a forensically ready cloud environment. It is based on the CFRaaS model that has previously been presented in this research. Like the CFRaaS model, the procedural flows have four different classes of entities (actors): Provider Layer (PL (also known as CSPs)), cloud layer, DFR layer (DFRL) and IRP layer.

A CSP is an entity that is used to provide services to the users of the cloud through management, service and dynamic provision of virtualized resources. The cloud is used as a platform, which consists of clients who wants to access the services provided by CSPs in the PL. DFRL is a proactive layer that allows forensic planning and preparation before the occurrence of security incidents. Essentially, the IRP layer is represented as a reactive process that involves the LEA and DFIs as shown in Figure 9. According to ISO/IEC 27043, IRP can be mapped to comprise the following: the initialization process that handles the first response when an incident occurs, the acquisitive process that ensures PDE is acquired and the investigative process that investigates the cause of incident.

To understand the procedural flows of the high-level CFRaaS, the authors consider a situation where a cloud client wants to access a service or the resources that are offered by the CSP. Before the client can access the services, the first step is to get familiarized with the Service Level Agreements (SLAs) on forensic monitoring. An SLA is a contract that explicitly states the services that a provider will deliver and the standard of the services being delivered. After agreeing to the SLAs, forensic readiness is achieved through deployment of an NMB that acts as a forensic agent to collect and preserve digital information. After this, if an incident is detected, then the IRP process may commence. The procedural flow in CFRaaS that is illustrated in Figure 9 is discussed below:

(1) A cloud user in the cloud layer is trying to access resources offered by the CSP in step 1.
(2) The CSP makes sure the user is well acquainted with monitoring SLAs. This allows the cloud user to proceed to access resources or not in step 2.
(3) The CSP then initiates a monitoring process by deploying an NMB as a forensic agent that performs forensic logging. In this way, cloud forensic readiness is achieved in step 3.

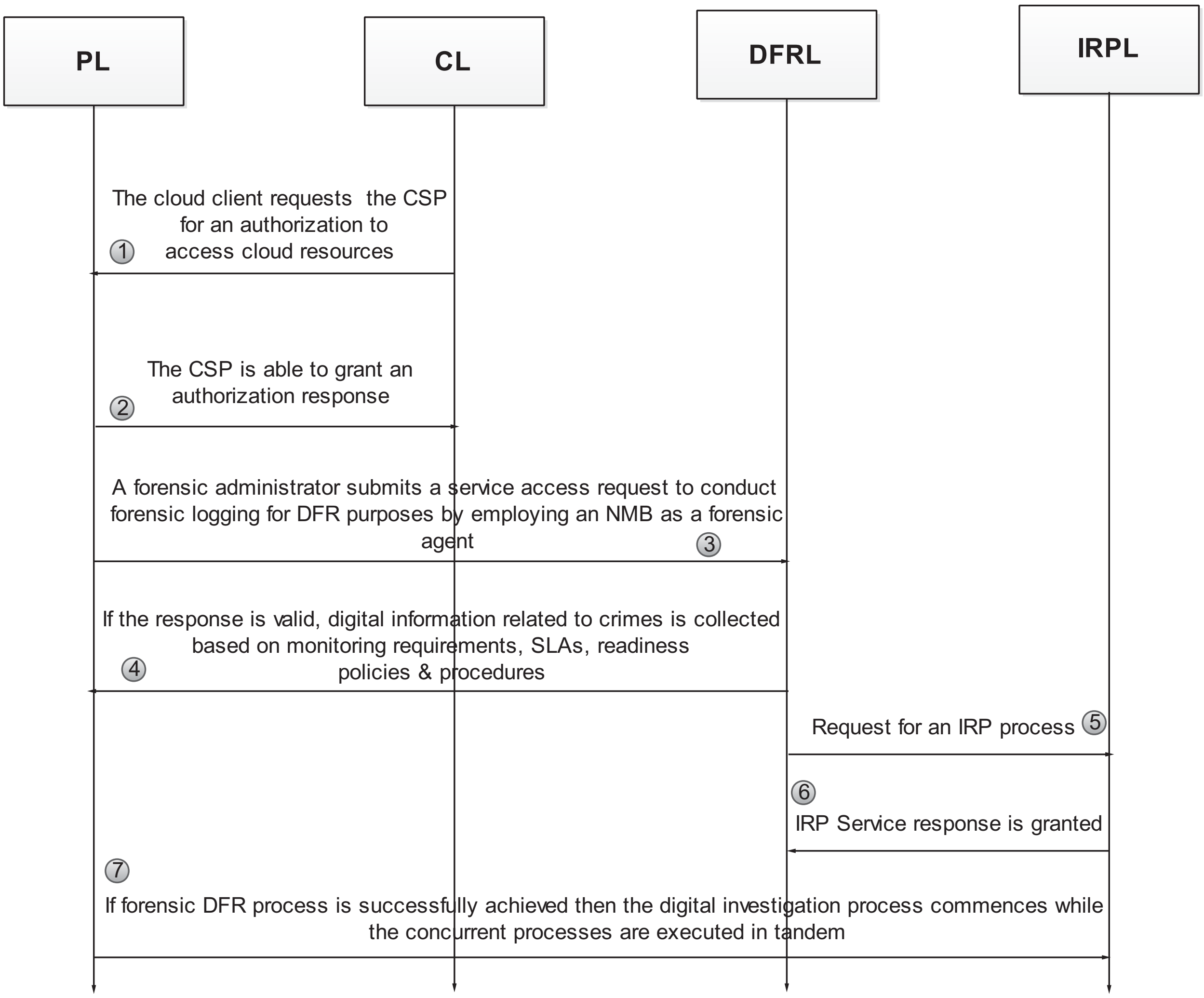

PL
CL
DFRL
IRPL
1 The cloud client requests the CSP for an authorization to access cloud resources
2 The CSP is able to grant an authorization response
A forensic administrator submits a service access request to conduct forensic logging for DFR purposes by employing an NMB as a forensic agent 3
4 If the response is valid, digital information related to crimes is collected based on monitoring requirements, SLAs, readiness policies & procedures
Request for an IRP process 5
6 IRP Service response is granted
7 If forensic DFR process is successfully achieved then the digital investigation process commences while the concurrent processes are executed in tandem

Figure 11: Procedural flow of the high-level CFRaaS.

(4) The CSP ensures that there is incidental planning and preparedness through the proactive DFRL in steps 4 and 5.

(5) The IRP finds the root cause of the security incidents through examination and analysis in steps 6 and 7.

Prototype implementation

The authors present a discussion on how the implementation of CFRaaS was achieved based on the propositions that described in the previous sections of this research paper. Firstly, the authors discuss how the NMB is able to be executed in the cloud environment to conduct Digital Forensic Readiness (DFR). Next, an explanation on how this was achieved is given. Figure 12 shows an experimental set-up that allows the NMB to collect digital information.

Figure 12 shows an experimental set-up of the CFRaaS prototype, which is organized into a number of steps. An NMB is deployed to infect the virtual instances of computers in the first step labelled 1 by either a botmaster/adversary. In this context, NMB is deployed by an administrator for forensic readiness purposes. This is done via the command and control (C&C) center. After this, the NMB is then executed to Virtual Machines (VMs) clients in an 'infection' process in step 3. Note that infection in the context of this research is represented as having a positive connotation. The NMB is able to perform forensic logging in step 4 after which the captured forensic logs are pushed into a forensic database for storage. The CFRaaS prototype is organized into five distinct processes that comply with the readiness processes that are mentioned in the standard of ISO/IEC 27043. Each of the prototype processes is explained in the sections that follow.

Planning and preparation phase

The essence of this phase is to allow the execution of the NMB in the virtual environment. In this phase, the NMB is designated to handle specific functions during execution. For example, the NMB is a software application with modified functionalities to depict a botnet that is executed to perform forensic processes. Additionally, the NMB deployment phase allows the CFRaaS prototype to be able to log events and processes successfully.

NMB deployment phase

The essence of this phase is to allow the execution of the NMB in the virtual environment. In this phase, the NMB is designated to handle specific functions during execution. For example, the NMB is a software application with

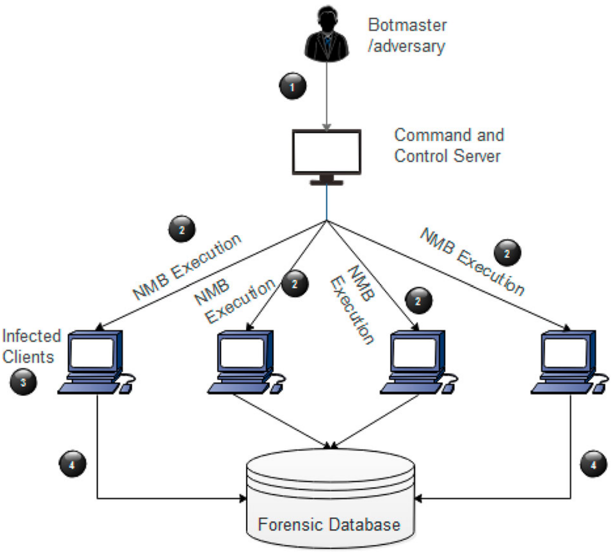


Figure 12: Experimental set-up of the prototype.

modified functionalities to depict a botnet that is executed to perform forensic processes. Additionally, the NMB deployment phase allows the CFRaaS prototype to be able to log events and processes successfully.

Monitoring phase

The monitoring phase enables the setting of the CFRaaS prototype through configuration of all the required tools. This phase allows the command and control (C&C) center to be able to connect to the internet and be able to initiate and halt the deployment of the NMB. In this phase, one is able to configure the CFRaaS prototype to be able to control the NMB in order for it to be directed to specific target. Figure 13 shows the C&C server control panel with different IP addresses, machine IDs, the timestamp the agents are dispatched and the time that the logs are received. Action represents control, which is used to start and to stop the 'infection' process.

Forensic logging phase

The forensic logging phase allows traffic (forensic logs) to be captured in the cloud environment by the deployed NMB. Capturing of traffic by the CFRaaS prototype is initiated from the C&C as is shown in Figure 13. There are two options in Figure 13 namely 'start' and 'stop'. By clicking 'start' the CFRaaS invokes the NMB to designated IP 196.249.12.226 and then traffic starts to be captured and once 'stop' is clicked the process of capturing forensic logs is halted.

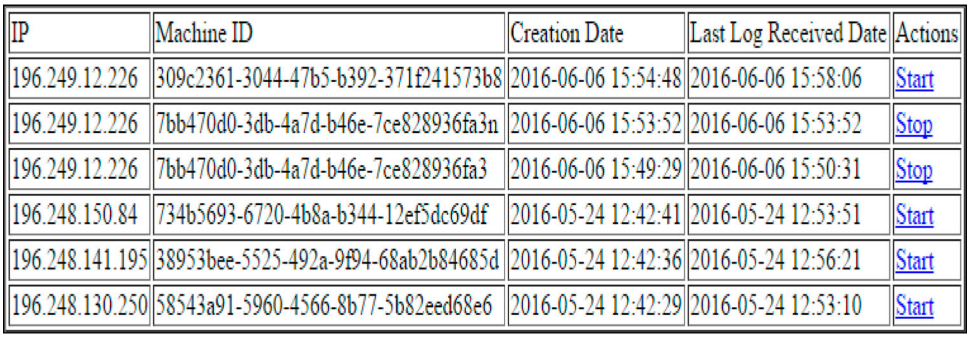
Figure 14 shows a block of PDE that is captured when the NMB is executed in the host that is shown as: *Logger.xp3.biz.* It is worth noting again that the notion behind this experiment is to collect digital forensic information that may be used as potential evidence.

The block represents the IP address, timestamps, CPU, RAM and keystrokes usage that have the probability of being used as PDE. To give an example why it is important to capture this kind of information, the authors provide an instance where malware might be consuming or diverting CPU processing power for unwanted tasks. By capturing this kind of information, a digital forensic analyst may be able to detect if a given malware is consuming the processing power at any given time which might end up interfering with the overall performance of the system.

After the potential evidence is captured, it is posted to the database using the POST/sendata.php HTTP/1.1 request that is shown in Figure 14 for possible anomaly detection if there is a potential incident. Figure 15, on the other hand, shows a block of forensic log data represented as (rawData) that is stored in MySQL database. Also shown is the hash created as a mode of digitally preserving the log. The timestamp that shows the time the forensic log is received is also shown, as well as the IP address of the forensic client and machine ID.

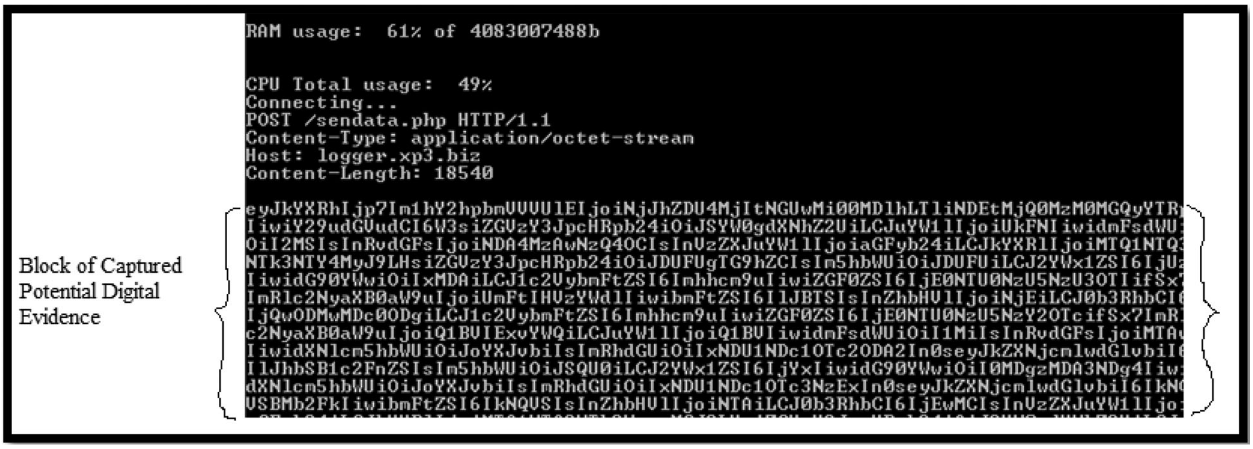
The *rawData* that is shown in Figure 16 represents the block of captured PDE that has been posted to the forensic database. This can fully be seen inside a circle of Figure 16 that shows the entire log that can be used as PDE in a DFR approach. Notwithstanding that, Figure 14 shows captured PDE after running the NMB on a set of clients in the cloud environment. The timestamp recorded from the report in Figure 15 shows *2016-03-13 13.12.21*, with IP address *196.248.99.47* is simply a representation of the IP address that is stored in the forensic database is shown in Figure 15. Other system IP addresses that are captured include: *196.248.159.209*, *196.248.96.30*, *196.248.99.38* and *196.248.117.128* respectively (see Figure 15).

Figure 17 highlights the system username and the key values that are entered every time that the keyboard is pressed. The timestamp that is associated with every time the key is pressed has also been captured. Also shown, in the last column of Figure 15, is the log entry ID of the forensic logs that were posted to the database.



IP	Machine ID	Creation Date	Last Log Received Date	Actions
196.249.12.226	309c2361-3044-47b5-b392-371f241573b8	2016-06-06 15:54:48	2016-06-06 15:58:06	Start
196.249.12.226	7bb470d0-3db-4a7d-b46e-7ce828936fa3n	2016-06-06 15:53:52	2016-06-06 15:53:52	Stop
196.249.12.226	7bb470d0-3db-4a7d-b46e-7ce828936fa3	2016-06-06 15:49:29	2016-06-06 15:50:31	Stop
196.248.150.84	734b5693-6720-4b8a-b344-12ef5dc69df	2016-05-24 12:42:41	2016-05-24 12:53:51	Start
196.248.141.195	38953bee-5525-492a-9f94-68ab2b84685d	2016-05-24 12:42:36	2016-05-24 12:56:21	Start
196.248.130.250	58543a91-5960-4566-8b77-5b82eed68e6	2016-05-24 12:42:29	2016-05-24 12:53:10	Start

Figure 13: Command and control server control panel.



```
RAM usage:  61% of 4083007488b


CPU Total usage:  49%
Connecting...
POST /sendata.php HTTP/1.1
Content-Type: application/octet-stream
Host: logger.xp3.biz
Content-Length: 18540

eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNjJhZDU4MjItNGUwMi00MDlhLTliNDEtMjQ0MzM0MGQyYTR
IiwiY29udGVudCI6W3siZGVzY3JpcHRpb24iOiJSYW0gdXNhZ2UiLCJuYW1lIjoiUkFNIiwidmFsdWU
OiI2MSIsInRvdGFsIjoiNDA4MzAwNzQ4OCIsInVzZXJuYW1lIjoiaGFyb24iLCJkYXRlIjoiMTQ1NTQ
NTk3NTY4MyJ9LHsiZGVzY3JpcHRpb24iOiJDUFUgTG9hZCIsIm5hbWUiOiJDUFUiLCJ2YWx1ZSI6IjU
IiwidG90YWwiOiIxMDAiLCJ1c2VybmFtZSI6Imhhcm9uIiwiZGF0ZSI6IjE0NTU0NzU5NzU3OTIifSx
ImRlc2NyaXB0aW9uIjoiUmFtIHVzYWdlIiwibmFtZSI6IlJBTSIsInZhbHVlIjoiNjEiLCJ0b3RhbCI
IjQwODMwMDc0ODgiLCJ1c2VybmFtZSI6Imhhcm9uIiwiZGF0ZSI6IjE0NTU0NzU5NzY2OTcifSx7ImR
c2NyaXB0aW9uIjoiQ1BVIExvYWQiLCJuYW1lIjoiQ1BVIiwidmFsdWUiOiI1MiIsInRvdGFsIjoiMTA
IiwidXNlcm5hbWUiOiJoYXJvbiIsImRhdGUiOiIxNDU1NDc1OTc2ODA2In0seyJkZXNjcmlwdGlvbiI
IlJhbSB1c2FnZSIsIm5hbWUiOiJSQU0iLCJ2YWx1ZSI6IjYxIiwidG90YWwiOiI0MDgzMDA3NDg4Iiw
dXNlcm5hbWUiOiJoYXJvbiIsImRhdGUiOiIxNDU1NDc1OTc3NzExIn0seyJkZXNjcmlwdGlvbiI6IkNQ
USBMb2FkIiwibmFtZSI6IkNQVSIsInZhbHVlIjoiNTAiLCJ0b3RhbCI6IjEwMCIsInVzZXJuYW1lIjo
```

Figure 14: A block of captured potential digital evidence.

Digital preservation phase

This phase highlights how the CFRaaS prototype is able to preserve the forensically captured logs. In order to maintain the integrity of the forensically logged data, the CFRaaS prototype is able to digitally preserve the forensic logs. The objective of employing digital preservation in the CFRaaS prototype is to make sure that no changes are experienced to the forensically logged potential evidence. The forensic logs that are shown in Figure 18 are encoded using MD5 and the resulting hash value is stored in a forensic database as shown in the third column of Figure 18, with its corresponding log file. The generated hash can be seen in the third row of Figure 18. The importance of storing the log files in the forensic data is to allow verification for purposes of checking the integrity of the forensic logs. In order to stop the logging process, the user may click 'stop functionality' at the C&C center (see Figure 13).

Once this functionality is clicked, the CFRaaS prototype is able to create an MD5 hash value for each log file that is captured which is then stored in the MYSQL database as shown in Figure 18. A unique identity that is presented as a primary key shown on the left side of the figure is used to identify each forensic log (tuple) inside the table of the forensic database. It is worth noting that the authors' focus is on forensically capturing the logs and digitally preserving them and not on what is contained in the forensic logs.

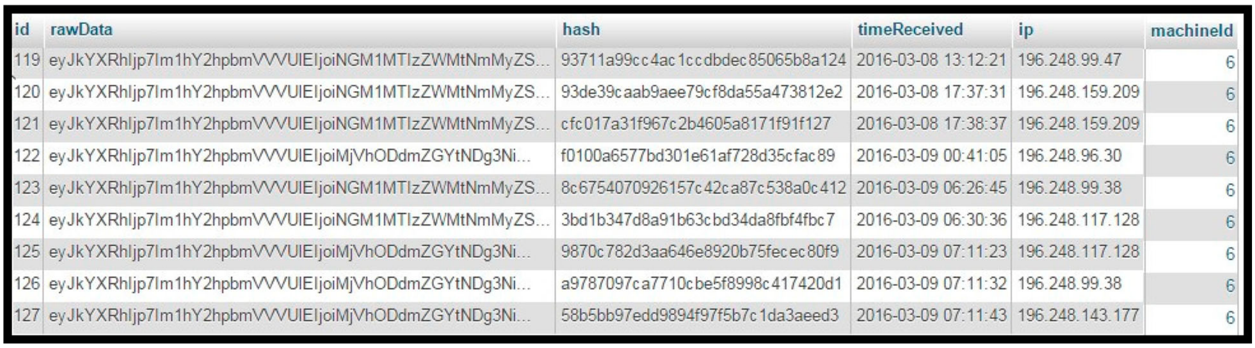
Pre-incident analysis phase

In this phase, an analysis is done on the captured forensic logs that are stored as PDE prior to incident detection. In the CFRaaS prototype, a pre-incident analysis was performed by checking the RAM and CPU usage as traffic was captured. RAM and CPU usage that are monitored as a block of digital data are also posted to the forensic database. The RAM graph in Figure 19 that shows if the usage is normal or not was generated based on the digital data that was previously pushed to the forensic database as shown previously in Figure 18.

The importance of the CPU and RAM graphs presented in Figures 18–20 is that they monitor whether there is any unusual activity that might consume memory or CPU processing speed. Monitoring these processes might help to detect if there is any unusual activity or a potential intrusion. A CPU utilization graph is shown in Figure 19 with the respective timestamps. Different points of Figure 20 are labelled as *x, y, z* and *v*. The labelled points help to monitor the usage and the performance of the CPU. For example, the following parameters according to Shropshire (2015) might create anomalies in the CPU energy consumption rate: CPU load, memory consumed, network packets received, network packets transmitted, disk reads and disk writes.

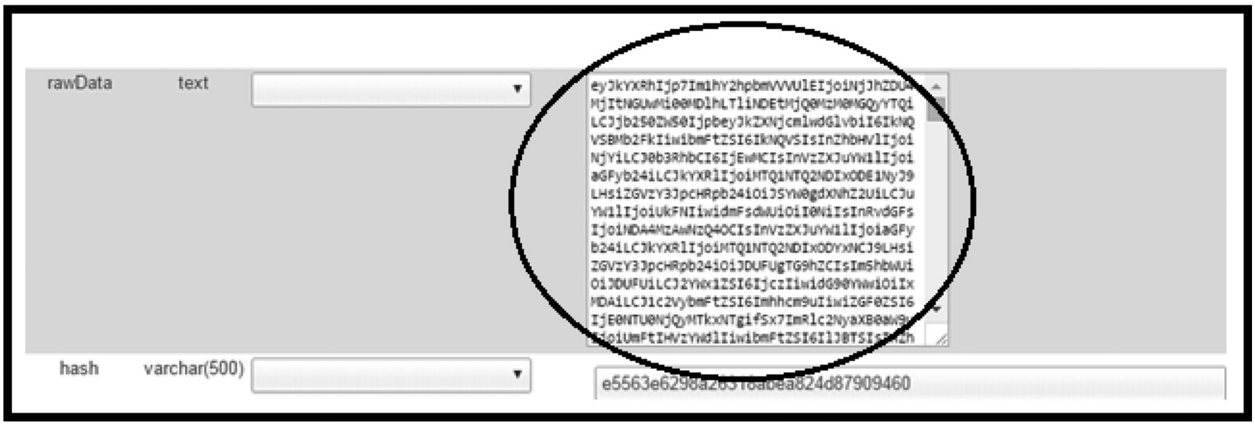
Forensic readiness report phase

The outcome of a digital investigation process or the steps taken to collect potential evidence is presented using a report. It is an integral part of any digital forensic investigation process. The authors have filtered the report using the computer IP address *196.248.115.230* with a start date of *2016-02-14* and timestamp *09:36:00* and end date *2016-02-14* and timestamp *12:54:00.* Figure 20



id	rawData	hash	timeReceived	ip	machineId
119	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	93711a99cc4ac1ccdbdec85065b8a124	2016-03-08 13:12:21	196.248.99.47	6
120	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	93de39caab9aee79cf8da55a473812e2	2016-03-08 17:37:31	196.248.159.209	6
121	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	cfc017a31f967c2b4605a8171f91f127	2016-03-08 17:38:37	196.248.159.209	6
122	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	f0100a6577bd301e61af728d35cfac89	2016-03-09 00:41:05	196.248.96.30	6
123	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	8c6754070926157c42ca87c538a0c412	2016-03-09 06:26:45	196.248.99.38	6
124	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	3bd1b347d8a91b63cbd34da8fbf4fbc7	2016-03-09 06:30:36	196.248.117.128	6
125	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	9870c782d3aa646e8920b75fecec80f9	2016-03-09 07:11:23	196.248.117.128	6
126	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	a9787097ca7710cbe5f8998c417420d1	2016-03-09 07:11:32	196.248.99.38	6
127	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	58b5bb97edd9894f97f5b7c1da3aeed3	2016-03-09 07:11:43	196.248.143.177	6

Figure 15: Forensic log, hash, timestamp, machine IP address, machine ID.

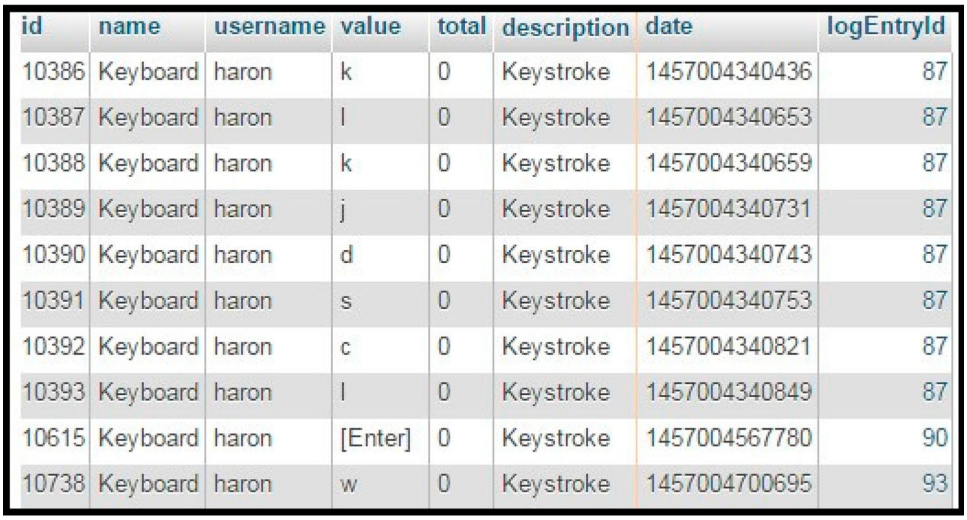


rawData text

```
eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNjJhZDU4
MjItNGUwMi00MDlhLTliNDEtMjQ0MzM0MGQyYTQi
LCJjb250ZW50IjpbeyJkZXNjcmlwdGlvbiI6IkNQ
VSBMb2FkIiwibmFtZSI6IkNQVSIsInZhbHVlIjoi
NjYiLCJ0b3RhbCI6IjEwMCIsInVzZXJuYW1lIjoi
aGFyb24iLCJkYXRlIjoiMTQ1NTQ2NDIxODE1NyJ9
LHsiZGVzY3JpcHRpb24iOiJSYW0gdXNhZ2UiLCJu
YW1lIjoiUkFNIiwidmFsdWUiOiI0NiIsInRvdGFs
IjoiNDA4MzAwNzQ4OCIsInVzZXJuYW1lIjoiaGFy
b24iLCJkYXRlIjoiMTQ1NTQ2NDIxODYxNCJ9LHsi
ZGVzY3JpcHRpb24iOiJDUFUgTG9hZCIsIm5hbWUi
OiJDUFUiLCJ2YWx1ZSI6IjczIiwidG90YWwiOiIx
MDAiLCJ1c2VybmFtZSI6Imhhcm9uIiwiZGF0ZSI6
IjE0NTU0NjQyMTkxNTgifSx7ImRlc2NyaXB0aW9u
IjoiUmFtIHVzYWdlIiwibmFtZSI6IlJBTSIsInZh
```

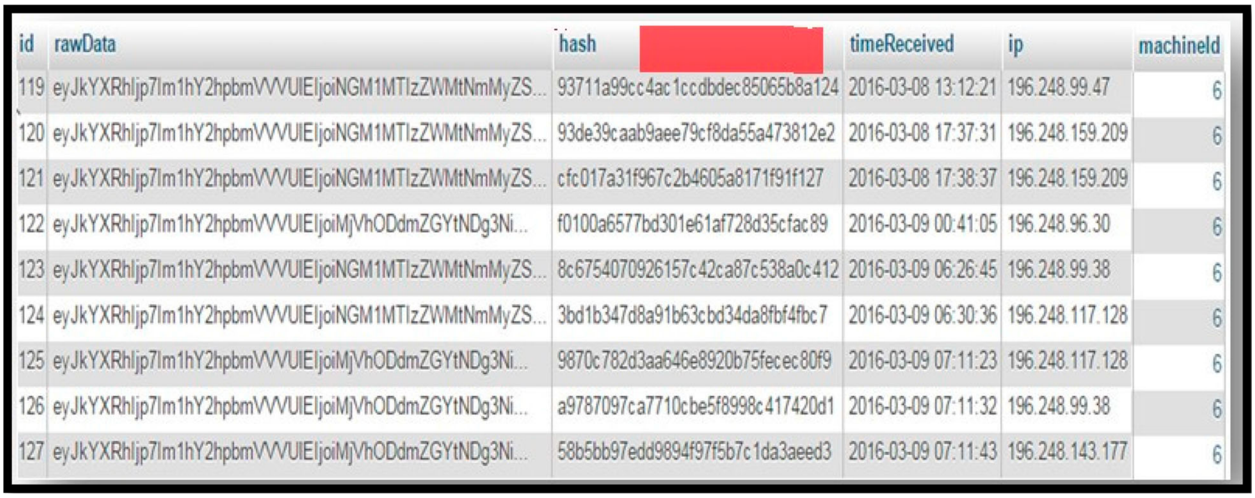
hash varchar(500) e5563e6298a26316abea824d87909460

Figure 16: Sample potential evidence represented as rawData in the database.



id	name	username	value	total	description	date	logEntryId
10386	Keyboard	haron	k	0	Keystroke	1457004340436	87
10387	Keyboard	haron	l	0	Keystroke	1457004340653	87
10388	Keyboard	haron	k	0	Keystroke	1457004340659	87
10389	Keyboard	haron	j	0	Keystroke	1457004340731	87
10390	Keyboard	haron	d	0	Keystroke	1457004340743	87
10391	Keyboard	haron	s	0	Keystroke	1457004340753	87
10392	Keyboard	haron	c	0	Keystroke	1457004340821	87
10393	Keyboard	haron	l	0	Keystroke	1457004340849	87
10615	Keyboard	haron	[Enter]	0	Keystroke	1457004567780	90
10738	Keyboard	haron	w	0	Keystroke	1457004700695	93

Figure 17: Forensically captured keystrokes in a readiness approach.



id	rawData	hash [redacted]	timeReceived	ip	machineId
119	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	93711a99cc4ac1ccdbdec85065b8a124	2016-03-08 13:12:21	196.248.99.47	6
120	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	93de39caab9aee79cf8da55a473812e2	2016-03-08 17:37:31	196.248.159.209	6
121	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	cfc017a31f967c2b4605a8171f91f127	2016-03-08 17:38:37	196.248.159.209	6
122	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	f0100a6577bd301e61af728d35cfac89	2016-03-09 00:41:05	196.248.96.30	6
123	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	8c6754070926157c42ca87c538a0c412	2016-03-09 06:26:45	196.248.99.38	6
124	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiNGM1MTIzZWMtNmMyZS...	3bd1b347d8a91b63cbd34da8fbf4fbc7	2016-03-09 06:30:36	196.248.117.128	6
125	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	9870c782d3aa646e8920b75fecec80f9	2016-03-09 07:11:23	196.248.117.128	6
126	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	a9787097ca7710cbe5f8998c417420d1	2016-03-09 07:11:32	196.248.99.38	6
127	eyJkYXRhIjp7Im1hY2hpbmVVVUlEIjoiMjVhODdmZGYtNDg3Ni...	58b5bb97edd9894f97f5b7c1da3aeed3	2016-03-09 07:11:43	196.248.143.177	6

Figure 18: MD5 hash for each block of forensically captured log file.

shows how the CPU monitoring graphs report can be generated.

A report can either be generated using the computer username or IP address as shown in Figure 21. For the sake of this research, the authors generated a report using the computer name, which displays the IP address. In addition, PDE is able to be extracted based on the date. The authors used a start date of *2016-02-14* and a time *of 09:36:00* and an end date as *2016-02-14* and timestamp *12:54:00* and the result are shown using the CPU usage (See Figure 20).

Legal considerations

According to the Federal Rules of Evidence of the USA, Rule 901(a) stipulates that in order to satisfy the requirements of identifying items that represent evidence, it is important that the proponent must produce evidence that supports the findings of the claim. Additionally, the process or evidence system mentioned in 901(b) notes that evidence should describe a process that is able to show that the deduced results are accurate (Hannon 2014). Similarly, the CFRaaS presented in this paper satisfies the requirements mentioned in ISO/IEC 27043; therefore, digital evidence that is produced by the NMB can be relied upon.

Importantly, data that exist as PDE in the cloud are bound to move and direct control of the data may reside in a totally different jurisdiction. Based on this fact, the legal protection of this data should be provided depending on the type of cloud service being offered by a CSP and the jurisdiction under which the data resides at the time of investigation. Therefore, the laws to be followed should be dependent on the cloud environment and the jurisdiction that is under investigation. Since there were no acceptable standards on how to govern the cloud at the time of writing this paper, the understanding of the investigation techniques of a public cloud is totally different compared with that of a private cloud in a given country or jurisdiction. Therefore, in order for the aforementioned requirements to be acceptable in the CFRaaS model, they should be sensitive to the stipulated local laws and regulations that govern a given jurisdiction. Furthermore, according to Brownlee and Guttman (1998), these laws may at times include specific requirements like privacy, data protection, confidentiality, how retained data that is to be used for forensic purposes is handled and the requirements of the law enforcement agencies. Moreover, some of these laws can be statutory or case laws (Killcrece

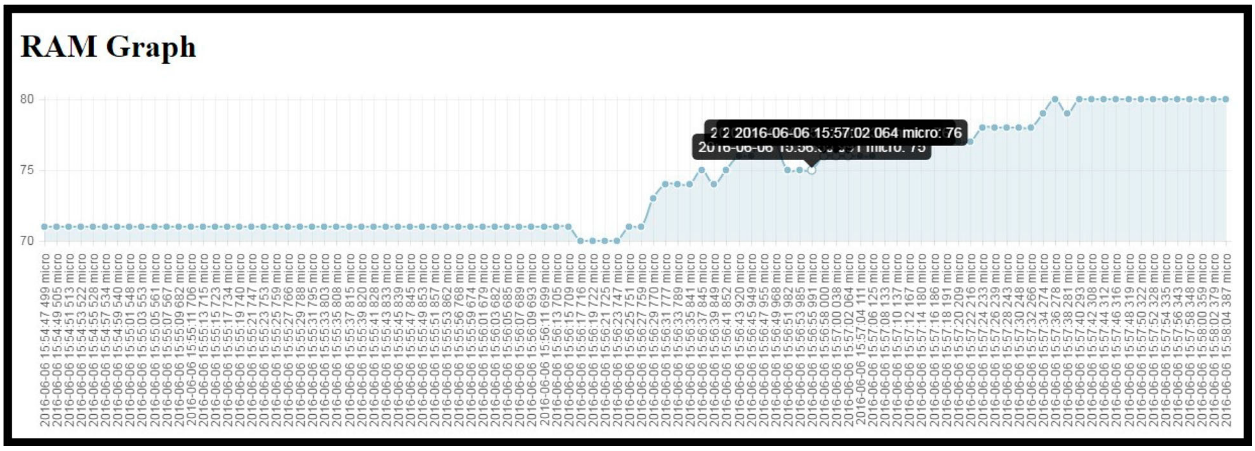
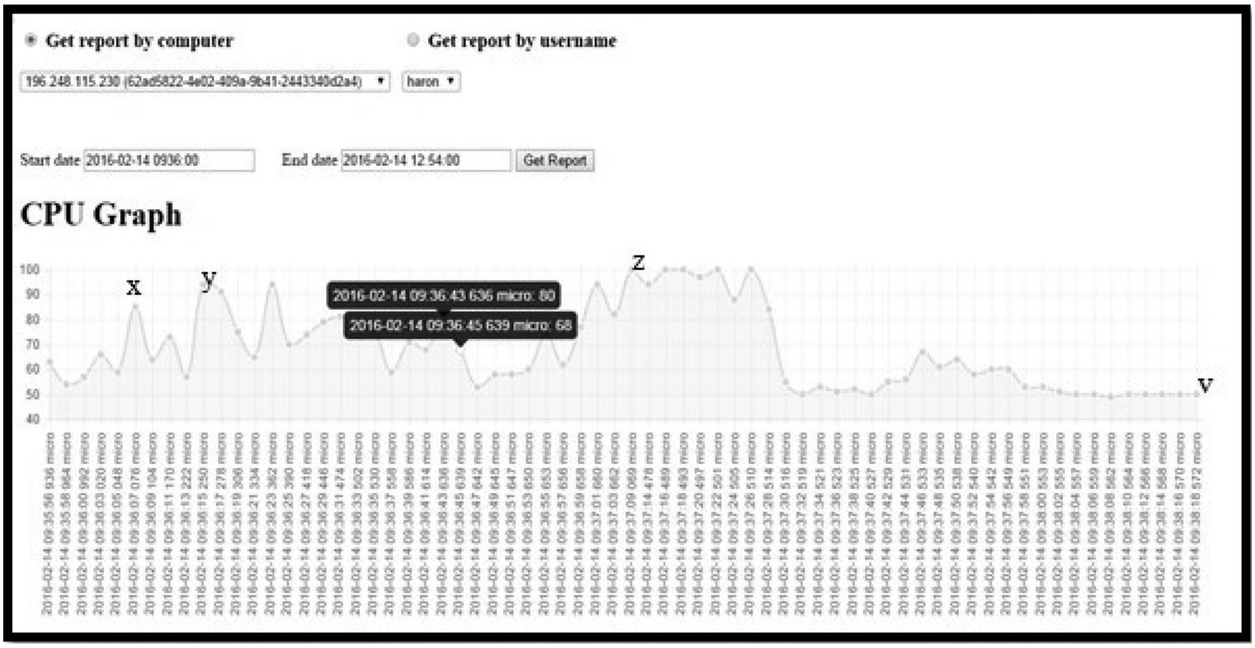


Figure 19: RAM utilization graph.



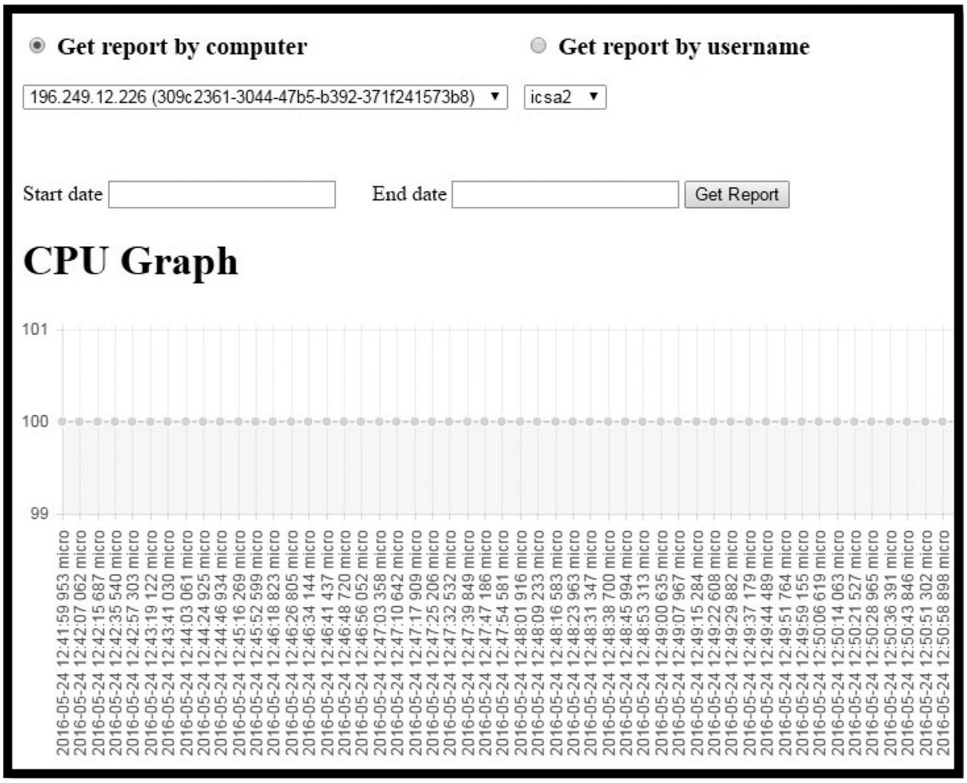
● Get report by computer ○ Get report by username

196.248.115.230 (62ad5822-4e02-409a-9b41-2443340d2a4) ▾ haron ▾

Start date 2016-02-14 09:36:00 End date 2016-02-14 12:54:00 Get Report

CPU Graph
100
90
80
70
60
50
40
x
y
z
v
2016-02-14 09:36:43 636 micro: 80
2016-02-14 09:36:45 639 micro: 68
2016-02-14 09:35:56 936 micro
2016-02-14 09:38:18 572 micro

Figure 20: CPU utilization with timestamps graph captured by an NMB solution.



● Get report by computer ○ Get report by username

196.249.12.226 (309c2361-3044-47b5-b392-371f241573b8) ▼ icsa2 ▼

Start date End date Get Report

CPU Graph

101
100
99
2016-05-24 12:41:59 953 micro
2016-05-24 12:42:07 062 micro
2016-05-24 12:42:15 687 micro
2016-05-24 12:42:35 540 micro
2016-05-24 12:42:57 303 micro
2016-05-24 12:43:19 122 micro
2016-05-24 12:43:41 030 micro
2016-05-24 12:44:03 061 micro
2016-05-24 12:44:24 925 micro
2016-05-24 12:44:46 934 micro
2016-05-24 12:45:16 269 micro
2016-05-24 12:45:52 599 micro
2016-05-24 12:46:18 823 micro
2016-05-24 12:46:26 805 micro
2016-05-24 12:46:34 144 micro
2016-05-24 12:46:41 437 micro
2016-05-24 12:46:48 720 micro
2016-05-24 12:46:56 052 micro
2016-05-24 12:47:03 358 micro
2016-05-24 12:47:10 642 micro
2016-05-24 12:47:17 909 micro
2016-05-24 12:47:25 206 micro
2016-05-24 12:47:32 532 micro
2016-05-24 12:47:39 849 micro
2016-05-24 12:47:47 186 micro
2016-05-24 12:47:54 581 micro
2016-05-24 12:48:01 916 micro
2016-05-24 12:48:09 233 micro
2016-05-24 12:48:16 583 micro
2016-05-24 12:48:23 963 micro
2016-05-24 12:48:31 347 micro
2016-05-24 12:48:38 700 micro
2016-05-24 12:48:45 994 micro
2016-05-24 12:48:53 313 micro
2016-05-24 12:49:00 635 micro
2016-05-24 12:49:07 967 micro
2016-05-24 12:49:15 284 micro
2016-05-24 12:49:22 608 micro
2016-05-24 12:49:29 882 micro
2016-05-24 12:49:37 179 micro
2016-05-24 12:49:44 489 micro
2016-05-24 12:49:51 764 micro
2016-05-24 12:49:59 155 micro
2016-05-24 12:50:06 619 micro
2016-05-24 12:50:14 063 micro
2016-05-24 12:50:21 527 micro
2016-05-24 12:50:28 965 micro
2016-05-24 12:50:36 391 micro
2016-05-24 12:50:43 846 micro
2016-05-24 12:50:51 302 micro
2016-05-24 12:50:58 898 micro

Figure 21: Selecting CPU usage report.

et al. 2003). The relevant laws that have been considered for this research paper include: Rule 901 and Rule 701 of The Federal Rules of Evidence of USA, Case laws for USA, Association of Chief Police Officers (ACPO)-UK, Electronic Communications and Transactions (ECT) Act 25 of 2002 of South Africa, Regulation of Interception of Communications and Provision of Communication-related Information Act (RICA) 70 of 2002 of South Africa, and the Protection of Personal Information (POPI) Act 4 of 2013 of South Africa. The laws may give a provision, or a direct or indirect authority to relevant digital evidence that may be needed to conduct digital forensic investigations. Having looked at the legal considerations, the reader should have insight into the legal ramifications of the proposed concept. In the next section, the reader is introduced to the concept evaluation of the study.

Critical evaluation of the proposed concept and scenario

The possible applicability of the proposed CFRaaS can help digital forensic investigators, practitioners and law enforcement agencies while building a digital forensic tool that may address digital forensic readiness based on the guidelines that are stipulated in the ISO/IEC 27043. Nevertheless, based on the hypothetical scenario in the introductory section of this paper, the availability of digitally preserved PDE in the cloud environment may aid in the creation of a hypothesis. This hypothesis in the long run will be used in a court of law to prove whether Anthony in the hypothetical scenario is guilty or will be exonerated.

Additionally, Alice as depicted in the hypothetical scenario was able to cover her malicious tracks simply because the organization was not readily prepared for any forensic investigation. On the same note, the presence of CFRaaS enables the CSP to have a mechanism for preserving proof of any security incident and providing it to digital forensic investigators. This allows digital investigators to check and prove the provenance regarding the incident. Thus, in the case of Anthony, an investigator would have been able to show how Alice was able to provide false evidence and falsify the intrusion incident, thus causing Anthony to be arrested.

Due to the rise of the significant amount of PDE in the cloud, the primary decisions that need to be made when considering the aforementioned requirements should be the legal considerations of jurisdictions, cost and time saved when DFR process is achieved. On the one hand, jurisdictions represent the legislation that may allow digital investigations in the cloud based on the law. On the other hand, cost and time saving are represented as the availability of PDE when needed for a digital forensic investigation (DFI). Moreover, based on the legal perspective, the requirements are tailored in such a way that privacy and confidentiality implications are taken into stringent consideration. Nevertheless, the CFRaaS model requirements allow a digital forensic readiness process to be achieved where a government's LEAs can use the accumulated PDE to reconstruct a particular incident or event for purposes of post-event response. In a previous research paper by Kebande and Venter (2015d), the authors presented an event reconstruction process on a CFRaaS model as a process that is able to distinguish events, discover the structure of events, distinguish one event from another by focusing on the relationship that exists between the events and also predict the behaviour of events based on a similarity measure. On the same note, the authors highlighted that based on the requirements illustrated, the CFRaaS model can help characterize PDE based on the causality and the characteristics during a digital forensic readiness (DFR) approach. Apart from that, characterizing PDE in the cloud environment during DFR makes detection more effective through providing information needed to conduct a DFI which is the ultimate goal DFR in the cloud environment (Kebande and Venter 2015c).

Consequently, it is imperative to note that the CFRaaS model increases effectiveness and flexibility for any organization that has enforced DFR as a corporate goal. Even though the cloud has a number of significant forensic challenges, the CFRaaS model can be tailored to be evidence specific. As a result of this, digital forensic investigators are able to profile possible attack scenarios.

If we review the hypothetical case scenario in the introduction, it is evident that if a proactive DFR mechanism had been put in place then Anthony would not have been wrongly charged. Apart from that, the cost of conducting digital forensic investigations would also have been significantly reduced.

Conclusion and future work

The authors of this paper have presented a discussion on the CFRaaS architecture. It is imperative to note that the CFRaaS architecture is aimed at organizations that need to enforce DFR in the cloud environment in order to allow digital forensic monitoring. Notably, it is critical for any organization to have a clear understanding of the requirements that may be used to achieve incident preparedness. Even though a number of research studies that have collected forensically sound evidence have been conducted or proposed, the cloud and/or the novel concept of using a NMB for digital forensic readiness purposes have hardly been the target of research studies.

Functional and quality requirements that are needed by the CFRaaS system are highlighted in this paper. The proposed requirements have addressed the techniques that can be used to achieve DFR in the cloud. The authors believe that this is a significant step towards implementing a successful cloud model that may enable DFR to be achieved in the cloud environment in accordance with ISO/IEC 27043. Furthermore, the authors explored different layers of the CFRaaS as follows: provider layer, cloud layer, DFRL and IRPL.

The authors aim in their future work to develop and implement a CFRaaS prototype as proof of a concept and to test the CFRaaS model for effectiveness. Furthermore, the authors hope that the model coupled with the proposed requirements can be standardized so that it can support future investigative technologies.

Acknowledgements

This work is based on research supported by the National Research Foundation of South Africa (Grant-specific unique reference number UID85794). The grant holder acknowledges that opinions, findings and conclusions or recommendations expressed in any publication generated by the NRF-supported research are those of the author(s) and the NRF accepts no liability whatsoever in this regard. The authors wish to thank the ICSA Research Group of the Department of Computer Science at the University of Pretoria for the support afforded in the conducting of this research.

Funding

The work was supported by National Research Foundation (Grant No. UID85794).

ORCID

Victor Rigworo Kebande http://orcid.org/0000-0003-4071-4596

H. S. Venter http://orcid.org/0000-0002-3607-8630

References

Brownlee, Nevil, and Erik Guttman. 1998. “Expectations for Computer Security Incident Response.” [Online]. http://www:ietf:org/rfc/rfc2350:txt.

Carrier, Brian D., and Eugene H. Spafford. 2004. “Defining Event Reconstruction of Digital Crime Scenes.” *Journal of Forensic Science* 49 (6): JFS2004127-8.

Casey, Eoghan. 2011. *Digital Evidence and Computer Crime: Forensic Science, Computers, and the Internet*. Baltimore: Academic Press.

Delport, Waldo, Michael Köhn, and Martin S. Olivier. 2011. “Isolating a Cloud Instance for a Digital Forensic Investigation.” In *ISSA*.

Dlamini, Moses, Hein Venter, Jan Eloff, and Mariki Eloff. 2014. “Requirements for Preparing the Cloud to Become Ready for Digital Forensic Investigation.” In *13th European Conference on Cyber Warfare and Security ECCWS-2014 the University of Piraeus Piraeus, Greece*, p. 242.

ENISA. 2009. *Cloud Computing: Benefits, Risks and Recommendations for Information Security*. Heraklion: European Network and Information Security Agency.

Gong, C., Liu, J., Zhang, Q., Chen, H., and Gong, Z. 2010. “The characteristics of cloud computing.” In *Parallel Processing Workshops (ICPPW)* (275–279), 39th International Conference. New York City: IEEE.

Hannon, Michael J. 2014. “An Increasingly Important Requirement Authentication of Digital Evidence”.

IDC (International Data Corporation). 2015. IDC Reveals Cloud Predictions for 2015. http://patch.com/massachusetts/framingham/idc-reveals-cloud-predictions-2015-0.

Innovative integration Inc. 2015–2017. Forecast: Cloud Computing to Skyrocket, Rule IT Delivery [Online]. http://www.innovativeii.com/2015-2017-forecast-cloud-computing-skyrocket-rule-delivery/.

ISO/IEC 27043: 2015. Information Technology-Security Techniques-Incident Investigation Principles and Processes. http://www.iso.org/iso/iso_catalogue/catalogue_tc/catalogue_detail.htm?csnumber=44407.

Kebande, Victor R., and Hein S. Venter. 2014a. “A Cognitive Approach for Botnet Detection Using Artificial Immune System in the Cloud.” In *Cyber Security, Cyber Warfare and Digital Forensic (CyberSec), 2014 Third International Conference* (52–57). New York City: IEEE.

Kebande, Victor R., and Hein S. Venter. 2014b. “A Cloud Forensic Readiness Model Using a Botnet as a Service.” In *The International Conference on Digital Security and Forensics (DigitalSec2014)*, 23–32. Ostrava: The Society of Digital Information and Wireless Communication.

Kebande, Victor, and H. S. Venter. 2015a. “A Functional Architecture for Cloud Forensic Readiness Large-Scale Potential Digital Evidence Analysis.” In *Proceedings of the 14th European Conference on Cyber Warfare and Security 2015: ECCWS 2015*, p. 373.

Kebande, Victor R., and Hein S. Venter. 2015b. “Adding Event Reconstruction to a Cloud Forensic Readiness Model.” In *Information Security for South Africa (ISSA), 2015*, pp. 1–9. IEEE.

Kebande, Victor R., and H. S. Venter. 2015c. “Obfuscating a Cloud-Based Botnet Towards Digital Forensic Readiness.” In *ICCWS 2015-The Proceedings of the 10th International Conference on Cyber Warfare and Security*, p. 434.

Kebande, Victor, and Hein Venter. 2015d. “Towards a Model for Characterizing Potential Digital Evidence in the Cloud Environment During Digital Forensic Readiness Process.” In *ICCSM2015-3rd International Conference on Cloud Security and Management: ICCSM2015*, p. 151. Academic Conferences and Publishing Limited.

Kebande, Victor R., and Hein S. Venter. 2016a. “On Digital Forensic Readiness in the Cloud Using a Distributed Agent-Based Solution: Issues and Challenges.” *Australian Journal of Forensic Sciences*, 50 (2): 1–30.

Kebande, Victor, and Hein Venter. 2016b. “Requirements for Achieving Digital Forensic Readiness in the Cloud Environment Using an NMB Solution.” In *11th International Conference on Cyber Warfare and Security: ICCWS2016*, p. 399. Academic Conferences and Publishing Limited.

Kebande, Victor, and Hein Venter. 2017. “Novel Digital Forensic Readiness Technique in the Cloud Environment.” *Australian Journal of Forensic Sciences*. doi:10.1080/00450618.2016.1267797.

Killcrece, Georgia, Klaus-Peter Kossakowski, Robin Ruefle, and Mark Zajicek. 2003. *State of the Practice of Computer Security Incident Response Teams (CSIRTs)*. (CMU/SEI Report Number: CMU/SEI-2003-TR-001). Pittsburgh, Pennsylvania: Software Engineering Institute.

Malan, Ruth, and Dana Bredemeyer. 2001. “Functional Requirements and Use Cases.” *Bredemeyer Consulting*.

Mell, P., and T. Grance. 2011. “The NIST Definition of Cloud Computing (Draft).” NIST Special Publication, vol. 800, p. 7.

Mouton, Francois, and Hein S. Venter. 2001. “Requirements for Wireless Sensor Networks in Order to Achieve Digital Forensic Readiness.” In *WDFIA*, pp. 108–121.

Palmer, Gary. 2001. “A Road Map for Digital Forensic Research.” In First Digital Forensic Research Workshop, Utica, New York (pp. 27–30).

Pohl, Klaus. 2010. *Requirements Engineering: Fundamentals, Principles, and Techniques*. Berlin: Springer Publishing Company, Incorporated.

Richter, Jennifer, Nicolai Kuntze, and Carsten Rudolph. 2010. “Security Digital Evidence.” In *Systematic Approaches to Digital Forensic Engineering (SADFE), 2010 Fifth IEEE International Workshop on*, pp. 119–130.

Rowlingson, Robert. 2004. “A Ten Step Process for Forensic Readiness.” *International Journal of Digital Evidence* 2 (3): 1–28.

Shropshire, J. 2015. “Securing Cloud Infrastructure: Unobtrusive Techniques for Detecting Hypervisor Compromise.” In *ICCSM2015-3rd International Conference on Cloud Security and Management: ICCSM2015*. Sonning Common: Academic Conferences and Publishing International.

Spyridopoulos, Theodoros, and Vasilios Katos. 2011. “Requirements for a Forensically Ready Cloud Storage Service.” *International Journal of Digital Crime and Forensics* 3 (3): 19–36.

Tan, John. 2001. “Forensic Readiness.” Cambridge, MA:@ Stake: 1–23.

Valjarevic, Aleksander, and Hein Venter. 2015. *A Comprehensive and Harmonised Digital Forensic Investigation Process Model*. Pretoria: University of Pretoria.

Varia, Jinesh. 2008. “Cloud Architectures.” *White Paper of Amazon, jineshvaria.s3.amazonaws.com/public/cloudarchitectures-varia.pdf*: 16.

Yasinsac, Alec, and Yanet Manzano. 2001. “Policies to Enhance Computer and Network Forensics.” In *Proceedings of the 2001 IEEE Workshop on Information Assurance and Security*, pp. 289–295.