OCF: An Open Cloud Forensics Model for Reliable Digital Forensics

Shams Zawoad, Ragib Hasan, and Anthony Skjellum*
{zawoad, ragib}@cis.uab.edu, skjellum@auburn.edu
Department of Computer and Information Sciences
University of Alabama at Birmingham, AL 35294, USA
*Department of Computer Science and Software Engineering
Auburn University, AL 36849, USA

***Abstract*—The rise of cloud computing has changed the way computing services and resources are used. However, existing digital forensics science cannot cope with the black-box nature of clouds nor with multi-tenant cloud models. Because of the fundamental characteristics of clouds, many assumptions of digital forensics are invalidated in clouds. In the digital forensics process involving clouds, the role of cloud service providers (CSP) is utterly important, a role which needs to be considered in the science of cloud forensics. In this paper, we define *cloud forensics* considering the role of the CSP and propose the Open Cloud Forensics (OCF) model. Based on this OCF model, we propose a cloud computing architecture and validate our proposed model using a case study, which is inspired from an actual civil lawsuit.**

***Index Terms*—Cloud Forensics, Forensics Science, Digital Investigation**

I. Introduction

Cloud computing is rapidly being adopted by business and IT organizations since it offers a high degree of scalability, convenient pay-as-you-go services, and low cost computing. The rapid adoption of cloud computing has effectively increased the market value of clouds, which crossed the $100 billion milestone in 2013 [1], which will continue to grow in the future [2], [3], [4]. According to a report from Market Research Media, the cloud computing market is expected to grow at a compound annual growth rate (CAGR) of 30% and will reach $270 billion in 2020 [4].

On the other hand, since in today's world most business records (92-99%) are stored electronically [5], the Federal Rules of Civil Procedure (FRCP) have broadened the scope of evidence in the 2006 amendment to include Electronically Stored Information (**ESI**) to be used in civil litigation [6]. Because of the rapid adoption of clouds, it is clear that a significant portion of the ESI will be stored in clouds. There have already been incidents where the availability of the massive computation power and storage facility of clouds are used for malicious purposes [7], [8], [9], [10]. It was reported recently that in order to launch Distributed Denial of Service (DDoS) attacks, adversaries are now placing a new Linux DDoS Trojan – *Backdoor.Linux.Mayday.g* in compromised Amazon EC2 virtual machines (VM) and launching attacks from those VMs [7]. For this types of attacks, we need to execute digital forensics procedures in the cloud to determine facts concerning a given incident. This type of forensic investigations are known as *cloud forensics*.

However, many of the assumptions of traditional digital forensics are invalidated in the cloud computing model. One of the major assumptions of digital forensics procedures and tools is that investigators (or users) have certain physical access to the evidence, which is an invalid assumption in clouds; sometimes it is even impossible to identify the physical location of the data stored in clouds. In clouds, each server contains files from many users. Hence, it is infeasible to seize servers from a data center without potentially violating the privacy of many other users. The trustworthiness of such evidence would also be questionable, because other than the Cloud Service Provider's (CSP) word/warranty, there is no routine way to determine the integrity of the evidence so obtained. To provide on-demand services, cloud providers do not typically support persistent storage for terminated VMs. Hence, data residing in cloud VMs will become unavailable after terminating such VMs. This in turn makes it almost impossible to do forensics investigation if some illegal activities have occurred using VMs that have subsequently been terminated. Finally, cloud providers and investigators can collude with a malicious user to hide traces of an illegal activity or to frame an innocent user. For these reasons, we need to take special care to provide support for reliable forensics in current cloud infrastructures.

While there are several research works that addressed the challenges of cloud forensics [11], [12], [13], [14] and proposed solutions to overcome some of the problems [11], [15], [16], [17], a formal model of reliable cloud forensics does not yet exist. To address this gap, we offer a redefinition of the cloud forensics process and propose in particular the Open Cloud Forensics (**OCF**) model. This model considers the new role of the CSP to support reliable digital forensics in the cloud. We argue that to support such reliable digital forensics, a continuous process flow should be executed by the CSP, which is a part of the cloud forensics process and is referred to as *continuous forensics*. Based on the OCF model, we then propose a forensics-aware cloud computing system and validate the proposed system using a case study.

Contributions. The contributions of this work are as follows:

1) We extend the existing definition of digital forensics and

redefine it in the context of clouds to support reliable digital forensics in the cloud. The new definition of cloud forensics will ideally guide future research in this area.

2) We propose the open cloud forensics (OCF) model, which includes *continuous forensics* support by the CSP – an integrated part of our cloud forensics definition. The proposed OCF model will ideally inspire future researchers to design forensics-aware cloud computing architectures.
3) While many architectures can be spawned from the OCF model, we present a forensics-aware cloud architecture, which supports the new cloud forensics definition and the OCF model. The design is validated by a case study, which is inspired from an actual civil lawsuit.

Organization The rest of the paper is organized as follows: Section II provides the background knowledge of digital forensics and cloud forensics and presents the motivation behind our work. In Section III, we discuss some contemporary research works on cloud forensics. Section IV presents our proposed cloud forensics process, the OCF model, and an OCF supported cloud architecture. In Section V, we discuss how the proposed model can work in a real-life scenario of cloud forensics and finally, we conclude in Section VI.

II. Background and Motivation

In this section, we present a brief overview of digital forensics and cloud forensics, and explain why we need a new forensics model for the cloud.

A. Digital Forensics

The National Institute of Standards and Technology (NIST) defines digital forensics as *“an applied science to identify an incident, collection, examination, and analysis of evidence data”* [18]. Maintaining the integrity of the information and a strict chain of custody for the data are mandatory. Several other researchers define digital forensics as the procedure of examining a computer system to determine potential legal evidence [19], [20]. From the above working definitions, we can state that digital forensics comprises four main processes:

- *Identification:* There are two main steps in identification: identification of an incident, and identification of the evidence, which will be required for successful investigation of that incident, with potential correlation to other incident(s).
- *Collection:* In the collection process, an investigator extracts the digital evidence from different types of media (*e.g.*, hard disk, cell phone, e-mail, and many other types of data). Additionally, the investigator preserves the integrity of the evidence.
- *Organization:* There are two main steps in the organization process: examination and analysis of the digital evidence. In the examination phase, an investigator extracts and inspects the data and its characteristics. In the analysis phase, he or she interprets and correlates the available data to come to a conclusion, which can serve to prove or disprove civil, administrative, or criminal allegations (when interpreted legally).
- *Presentation:* In this process, an investigator makes an organized report to state his or her findings about the case. This report should be appropriate for presentation to the judge and jury.

Figure 1 illustrates the flow of the processes of digital forensics.

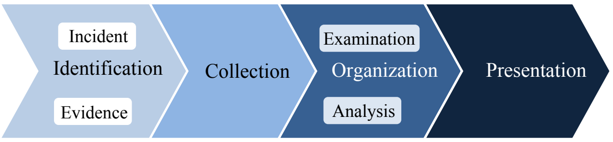

Fig. 1: Digital Forensics Process Flow

B. Cloud Forensics

NIST recently established the NIST Cloud Computing Forensic Science Working Group (NCC-FSWG) to research cloud forensic science challenges and to develop solutions, standards, and technology that will mitigate the challenges that cannot be handled with current technology and methods [21], [22]. NIST defines cloud forensics as *“the application of scientific principles, technological practices and derived and proven methods to reconstruct past cloud computing events through identification, collection, preservation, examination, interpretation and reporting of digital evidence [22].”*

Different steps of digital forensics, as shown in Figure 1 and the control over evidence vary in the cloud according to the service and deployment models of cloud computing. For example, the evidence collection procedures in Software-as-a-Service (SaaS) and Infrastructure-as-a-Service (IaaS) are different. In the private cloud deployment model, we can have physical access to the digital evidence, but we rarely can get physical access to the public deployment model.

C. Motivation

Some fundamental characteristics of clouds make digital forensics more challenging as compared to dedicated server and system environments. Hence, we need to model the digital forensics in such a way that we can overcome the challenges imposed. Below, we present an hypothetical case to illustrate the challenges of digital forensics imposed by clouds.

Motivating Case Study. The motivating (hypothetical) case study is inspired by the Quantlab Technologies Ltd. v. Godlevsky case [9]. In this case, plaintiffs brought suit against defendants for copyright infringement, breach of contract, misappropriation of trade secrets, and fraud.

Mallory worked in a software development company BISoft and there she developed a business analysis algorithm and a business intelligence system for high volume business data. The software proved popular among in industry and BISoft reaped large profits from the system. Though the company maintains strict rules to protect their intellectual property, Mallory managed to export the code of the developed system *to CloudCo's storage, an arms-length 3rd party CSP. Later, Mallory formed her own company and used substantially the same designs and code to develop a business intelligence system. BISoft filed a case against Mallory accusing her and her company of stealing intellectual property and Bob, a digital forensics investigator, was assigned to determine the facts.*

However, the following characteristics of clouds can serve to hinder Bob's investigation:

Physical inaccessibility. Evidence collection procedure is harder in the cloud due the physical inaccessibility of digital evidence. The existing stat-of-the-art digital forensic procedures and tools that Bob could use in traditional computing systems was a poor fit with the cloud because the assumption of having physical access to the computing resources (*e.g.,* hard disk, network router, etc) is invalid in clouds. Sometimes, we do not even know where the data is located in a large, distributed cloud infrastructure. Location of data is important for many reasons: for example, a warrant must specify a location, but in a cloud, data may not be located at a precise location or a particular storage server and may transition over time as well. A number of researchers address this issue in their work [23], [12], [24], [25], [26], [27]. Because of the physical inaccessibility, Bob will have less control over the evidence and needs to rely on CloudCo to collect the digital evidence from the cloud computing environment. This is a serious bottleneck in the collection phase (and could arguably raise doubts about chain of custody and integrity).

Volatile Data. Data that resides in a virtual machine (VM) is volatile since these data cannot be sustained without power. After terminating a VM, no data will be preserved, except that which was written to disk. The volatile data can be documents, network logs, operating system logs, and registry logs stored on volatile volumes or in memory. In order to provide on-demand computational and storage services, CSPs do not provide persistent storage to a VM instance. Hence, if Mallory chose VMs running on *CloudCo* to store the stolen code and then terminated the VMs after she developed her own software, that will lead to a complete loss of the crucial evidence, such as logs, information about data possession, and/or provenance. Although there is a way to preserve VM data by storing an image of the VM instance, Mallory would not definitely use it in order to reduce her digital footprint, assuming "knowledge of guilt" or simply avoidance of potential exposure to future forensics drove her behavior while using the CSPs resources.

Multi-tenancy. Cloud computing is a single-owner, multi-tenant system, while traditional computing is single owner system (potentially multi-tenant or otherwise single-tenant). To offer an analogy, a cloud can be compared to a motel, while the other can be compared to a person's home, or to an apartment complex. In clouds, multiple Virtual Machines (VM) routinely share the same physical infrastructure. Hence, in our hypothetical scenario, Mallory's allegedly stolen code and other documents of legitimate users can be stored in and on the same storage device(s). Given this property of clouds, it is difficult if not impossible for Bob to confiscate such a shared storage device without violating the privacy and usage rights of other *CloudCo* users. Mallory may also repudiate data contained on the storage device as evidence that contains information of other users, not hers. In such a case, if Bob finds any trail of the stolen intellectual property from the cloud, he also needs to prove it to the court that the evidence presented actually owns to Mallory. Conversely, if Mallory would have stored the documents in her personal computer, there would be prima facie evidence that she would be responsible for all the evidence found in her computing system (which Mallory would have to argue against).

Collusion Between Different Entities. In traditional digital forensics, investigators have full control over the evidence (*e.g.*, router logs, process logs, and hard disks). Whereas users or investigators have limited control over the evidence stored in clouds. Hence, one of the major challenges of establishing trustworthy forensics support in cloud infrastructures is dependency on the cloud providers who are not necessarily completely honest (or may employ dishonest actors). With the state-of-the-art frameworks for collecting evidence from a cloud, Bob needs blindly to take *CloudCo*'s assertations as valid, since he cannot verify whether *CloudCo* is providing valid evidence or not. Such gaps provide opportunities for a defense to raise objections of reasonable doubt in criminal investigations and to impair any "more likely than not" standard in civil litigation.

The *CloudCo's* employee, who will collect data on behalf of Bob is most likely not a licensed forensics investigator and it is impossible in any event to guarantee his or her integrity in a court of law. Mallory could potentially collude with that employee of *CloudCo* to hide important evidence or to inject invalid evidence to mitigate her guilt or help establish her innocence. Such a malicious employee could provide incomplete logs, remove documents without keeping any trace, could maintain false timestamps, and could tamper with various provenance data or meta-data. Conversely, Bob could also be malicious and could alter any kind of evidence before presenting to court. In a traditional system, only the suspect and the investigator can collude. The potential for three-way collusion in clouds certainly increases the attack surface and makes cloud forensics more challenging.

III. Related Work

Cloud forensics is a relatively new area of discourse. Since cloud computing is based on extensive network access, and since network forensics handles forensic investigation in private and public network, Ruan *et al.* defined cloud forensics as a subset of network forensics [28]. They also identified three dimensions in cloud forensics – technical, organizational, and legal. However, this definition is not complete since analyzing a hard drive stored in Amazon's infrastructure is not an instance of network forensics.

Logs are in heterogeneous formats in clouds and hence, it is difficult to examine and analyze log evidence. Marty proposed guidelines to overcome this problem [29]. The proposed guidelines instructs us to focus on three things: when to log, what to log, and how to log. At minimum, he suggests logging the timestamps record, application, user, session ID, severity, reason, and categorization, so that we can get the answer of *what*, *when*, *who*, and *why* (the "4 Ws"). He also recommended syntax for logging, which was represented as a key-value pair and used three fields to establish a categorization schema – object, action, and status [29].

Dykstra *et al.* illustrated the difficulty of data acquisition by using a hypothetical case study of child pornography [12]. To investigate this case, the forensics investigator needed bit-for-bit duplication of the data to prove the existence of contraband images and/or video. In a cloud, the investigator could not collect data himself/herself. Furthermore, the data cannot be seized by confiscating the storage server in a cloud, as the same disk can contain data from many honest users.

Zafarullah *et al.* were able to monitor the Eucalyptus behavior and log all internal and external interaction of Eucalyptus components [30]. From the logs, the were able to track a DDoS attack launched from their Eucalyptus cloud. To make the network, process, and access logs available to customers, Bark *et al.* proposed to expose read-only APIs by CSPs [11]. By using these APIs, customers can gather valuable information and can provide this to investigators. Zawoad *et al.* proposed a Secure Logging-as-a-Service (LaaS) securely to store VM activities, a procedure that ensures integrity and confidentiality of logs from a malicious CSP and investigators [31]. To detect temporal inconsistencies in a VM's timeline, Thorpe *et al.* developed a log auditor by using the 'happened before' relation [32] in the cloud environment [17].

Delport *et al.* focused on isolating an instance to mitigate the multi-tenancy issue [33]. Isolation is necessary because it helps to protect evidence from contamination. Virtual Machine Introspection (VMI) can also be helpful in forensic investigation. In [34], Hay *et al.* showed that if a VM instance is compromised by installing some rootkit to hide the malicious events, it is still possible to identify those malicious events by performing VMI.

Patrascu *et al.* proposed a cloud architecture to monitor the activities in a cloud environment [35]. Using the proposed framework, they collected logs from different layers of the cloud. They also presented a data center topology to deploy the proposed architecture. Recently, Dykstra *et al.* implemented FROST, a forensic data collection tool for OpenStack [36]. Using FROST, cloud users/investigators can acquire an image of the virtual disks associated with any of a user's virtual machines, and validate the integrity of those images with cryptographic checksums. It is also possible to collect logs of all API requests made to CSP and OpenStack firewall logs for VMs. While these two efforts are big steps towards providing forensics support in the cloud, these works treated cloud service providers as honest and reliable. However, in an adversarial situation, CSPs as well as investigators can be malicious, non-compliant, and/or could tamper with the logs. Hence, trustworthiness of the data collected through the proposed architectures could be questionable. Moreover, these solutions only focused on logs, which is narrow and not all the information one might wish to collect in a forensic investigation.

IV. Open Cloud Forensics

State-of-the-art digital forensics models do not presently consider third-party CSPs in the investigation process. However, we argue that without defining the role of CSPs in forensics investigations, cloud forensics cannot be defined properly and it may not be possible to execute digital forensics procedures in a trustworthy manner. We introduce the notion of *Continuous Forensics* in the cloud forensics model to facilitate the digital forensics procedures. In this section, we first amend cloud forensics by considering the important role of CSPs. We present the threats that exist in the cloud forensics process. Based on our definition of cloud forensics and the threat model, we then propose the Open Cloud Forensics (OCF) model. Following that, a cloud architecture is proposed, which supports this OCF model in order to ensure reliable forensics in the cloud.

A. Cloud Forensics Process

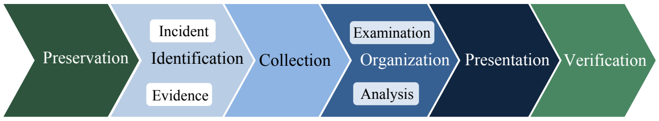

Fig. 2: Cloud Forensics Process Flow

We define cloud forensics as the science of preserving all evidence possible while ensuring the privacy and integrity of the information, identification, collection, organization, presentation, and verification of evidence data to determine the facts about an incident involving clouds. Figure 2 illustrates the proposed cloud forensics process flow.

As we note from Figure 2, the preservation stage of all possible evidence and the verification of evidence are introduced with the state-of-the-art digital forensics process flow presented in Figure 1. Because of the volatile nature of cloud data and possible manipulation of evidence by malicious cloud providers, we need to include these two steps. The preservation stage should always be online/running and, hence, we refer to this as a continuous forensics process.

In the verification stage, the court authority will needfully verify the cloud-based evidence provided by an investigator. The verifier will use the information stored in preservation stage to decide on the integrity of the evidence. Trustworthiness of evidence and availability of the volatile data depends on how efficiently and securely we preserve such data.

B. Threat Model

A cloud provider may be honest but it can employ disgruntled or malicious personnel with superuser access or its machines can be compromised by malicious users, yielding them superuser access. Hence, unlike the existing body of work on cloud forensics [15], [17], [36], *a priori* we do not consider the CSP as honest. In our threat model, users, investigators, and CSPs—all three entities—can be malicious and hence can collude to provide fake or falsified evidence to the auditor.

Collusion between different entities increases the capability of the attackers and hence, expands the attack surface. Ensuring reliability of the evidence becomes more challenging when different malicious entities collude rather than acting singly. For example, a malicious user acting alone cannot modify the evidence stored under the control of CSP unless he colludes with the CSP or compromises the CSP in order to alter the evidence. A user can delete evidence that is under his control (intentionally or unintentionally) or can provide false evidence to an investigator. However, when the CSP is honest, the investigator can detect at least some of all such alterations of evidence made by a malicious user. On the other hand, if the CSP and the user both provide the same falsified evidence to the investigator, it will be difficult to verify its integrity…it may simply be accepted as valid.

Likewise, an investigator can present false evidence to the court to frame an honest user, or to save malicious user from conviction. However, if the dishonest investigator acted alone, this malicious behavior could be detected, when the auditor verify the evidence provided by the investigator with the evidence stored in clouds. However, when the investigator colludes with the CSP, it will be difficult for an auditor to determine the trustworthiness of the evidence. This gap itself provides a basis for potential defense claims as well.

Moreover, after providing evidence to an investigator, a CSP can potentially repudiate any evidence. As data are comingled in the cloud, a malicious user could claim that a particular evidence does not belong to him. An intruder as well as a malicious cloud employee could acquire the evidence of a user to learn the user's activity or confidential information.

C. The OCF Model

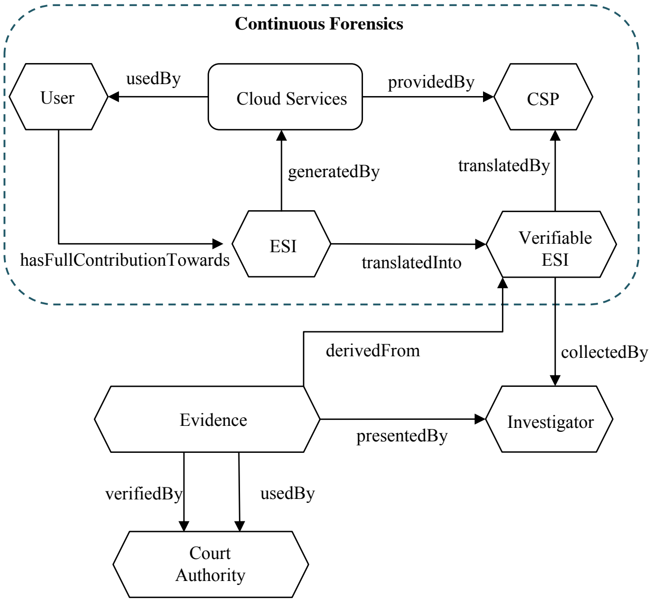

Fig. 3: Open Cloud Forensics Model

Based on our definition of cloud forensics and the threat model, we propose the Open Cloud Forensics (OCF) model, which is depicted in Figure 3. In the OCF model, four entities are involved: user, CSP, investigator, and court authority.

Let U be the set of all users who are the customers of a given CSP. If there are n such users: $U = \{u_1, u_2, ..., u_n\}$. The CSP provides m number of services to its customers. Let S be the set of all services. $S = s_1, s_2, ..., s_m$. In the OCF model, the set of services S includes but is not limited to software, computing and storage resources, platforms, etc. Any of these services (when used by a user) creates ESI such as documents, activity logs, file system provenance, and many others. An ESI generated for accessing service s_i by user u_j at time t is described as $E^{s_i}_{tu_j}$. Hence, if the user u_j has access to q number of services, where $q \leq m$, then all the ESI of user u_j between time ts and te can be defined as

$$E_{u_j} = \bigcup_{1 \leq i \leq q} \bigcup_{ts \leq t \leq te} E^{s_i}_{tu_j} \quad (1)$$

The complete set of ESI between time ts and te for n number of users, E^{ts}_{te} can defined as

$$E^{ts}_{te} = \bigcup_{1 \leq j \leq n} E_{u_j} \quad (2)$$

Now, it is the role of the CSP to translate all the ESI to verifiable ESI, which preserves their integrity, as well as the privacy of the ESI. Hence, for every ESI E_i, there will be a corresponding verifiable ESI denoted as VE_i.

The above sequence of actions—where a user accesses a cloud service, which in turn generates ESI, and is finally translated into verifiable ESI—are referred to as a continuous forensics process. The continuous forensics process is marked with blue dotted line in Figure 3. Without the presence of this continuous forensics support in the cloud, the next steps of the forensics process may not be trustworthy.

Let user u_j be a malicious user, who executed an illegal activity using service s_i provided by the CSP. An investigator subsequently gathers relevant verifiable ESI from the set of verifiable ESI of user u_j for service s_i, $VE^{s_i}_{u_j}$, analyze the ESI and present evidence to the court. The set of verifiable evidence presented to the court is denoted as $PE^{s_i}_{u_j}$ and $PE^{s_i}_{u_j} \subset VE^{s_i}_{u_j}$. The court authority later verifies the integrity of the evidence $PE^{s_i}_{u_j}$ and rules based on the evidence.

D. OCF-Supported Clouds

Based on the OCF model, we design a cloud computing system, which is illustrated in Figure 4. We introduce the following features to support the OCF model.

- First, to prevent the loss of volatile ESI, we propose a continuous synchronization feature, which will store sufficient volatile ESI efficiently in a persistent storage without hampering the CSP's business model.
- Second, to translate the ESI to verifiable ESI, we propose a *proof publisher module (PPM)* that will create cryptographic proof of all the ESI and publish to the Internet[1], so

[1] Another trusted repository could be used as well, this is just one example, analogous to how Bitcoin registers transactions.

that neither a dishonest cloud provider nor an investigator can alter evidence after-the-fact.

- Third, all the ESI will be made available to the investigators through APIs so that investigators do not need physical access to the cloud infrastructure to acquire possible evidence.
- Finally, the court authority can use the published proofs of the ESI to verify the integrity of the evidence.

In the following sections, we briefly describe each of the features.

1) Continuous Synchronization: Persistent ESI will be directly stored within the persistent storage. We need the continuous synchronization scheme for volatile ESI. Since CSPs do not provide persistent storage to VMs, turning off or rebooting a VM will ultimately lose all the data residing in that VM. Data that are volatile in nature must be stored in a persistent database so that we can gather the evidence even from a terminated VM.

One possible solution to this problem is that CSPs will provide a continuous synchronization API to customers. Using this API, customers can preserve the synchronized data to any cloud storage (*e.g.,* Amazon S3), or to their local storage. However, if the adversary is the owner of a VM, this mechanism will not work. Trivially, he or she will not be interested to synchronize his/her malicious VM. Hence, we propose that the CSP will constantly monitor all the VMs running in the cloud host machines and store the volatile data in a persistent store.

Volatile data can be network logs, OS logs, and registry logs, etc. However, we need to carefully select which of the volatile data will be preserved to what extent. Storing all the volatile data for a long period of time may not be economical for the CSP. Hence, based on the business model of the CSP and government regulations, we can select the crucial pieces of volatile data and define a retention period for those data.

When a VM is in its active state, the host machine can track which data belongs to which VM. Hence, while preserving the data, the CSP can take care of segregating the data according to VM owners. Thus, multiple VM owners' data will not be commingled. The CSP can preserve the confidentiality of the data by using public-private key based encryption, where private keys are only accessible to users and law enforcement agencies. This will ensure the confidentiality of data from malicious cloud employees.

2) Cryptographic Proof: To prevent collusion between the CSP, investigators, and cloud users, we propose a proof publisher module (PPM), as mentioned above. This module will be responsible for generating and preserving the proofs of the ESI stored in the persistent storage. The proof will not be the data itself, rather we propose to use a cryptographic accumulator such as a One-Way accumulator in order to preserve the proof. Using a cryptographic accumulator has some benefits. *First*, for file system data, the preservation of the plain files as the proof will increase the storage volume significantly as compared to preserve the cryptographic accumulator-based proofs of files. *Second*, a cryptographic proof does not disclose original data. At the end of each day (or other audit period), the PPM will publish the generated proof publicly on the Internet so that the cloud provider cannot modify any generated proof after-the-fact.

3) Availability of Evidence: We propose to provide secure read-only APIs to law enforcement agencies and other narrowly authorized parties. Only the investigators and the court will have access to these APIs. They can collect the preserved evidence through these APIs. To implement this feature, the CSP needs to accommodate an additional web server, which will communicate with the previously described persistent proof storage to collect the requested ESI by an API call. The web server can provide Representational State Transfer (REST) based API, where the requested ESI will be the resource. To retrieve these evidence, GET operations can be used on the resources. Caller of a REST service can pass different parameters to retrieve his desired ESI.

4) Integrity Verification by the Court: Since we cannot guarantee the integrity of the evidence provided by an investigator, the court authority needs to verify the integrity of the evidence. In this regard, the court will collect the cryptographic proofs available on the Internet. If a piece of evidence is valid, it should be present in the cryptographic proofs. If a piece of evidence is removed, that should also be detected from the proofs. Proofs of any faked instances of evidence will not be present in the published proofs.

Once, a proof is published, none of the entity can modify or deny the proof. Therefore, when the proof is made publicly available, neither the CSP nor investigator can alter any evidence or provide falsified evidence. Thus, using the periodically published cryptographic proofs, a verifier can determine the integrity of the evidence even when three entities (user, CSP, and investigator) collude with each other.

V. Discussion

In this section, we show how existing forensics procedures fail for the hypothetical scenario presented in Section II-C and how our proposed model can ensure reliable forensics for the same scenario.

A. Unreliable Forensics Process in Current Clouds

Using traditional digital forensics methods, we cannot execute a reliable forensic investigation in the current clouds. This situation is illustrated in Figure 5. In the case study, Mallory removed the stolen codes after using or further transferring it. When Bob requested Mallory's ESI from *CloudCo*, the firm will fail to provide such removed documents. In current clouds, there is no way for Bob to recover those deleted files. If Mallory used a VM running on clouds to store the documents, the situation will be more complicated for Bob. Since the storage of a VM is volatile, there will be no guaranteed trail of such files in the cloud. In this situation, the evidence presented to the court will be incomplete and forensics process will not be reliable. Moreover, the court authority trusts Bob's and *CloudCo's* honesty to validate the evidence. However, such honesty is not guaranteed since either or both *CloudCo* and

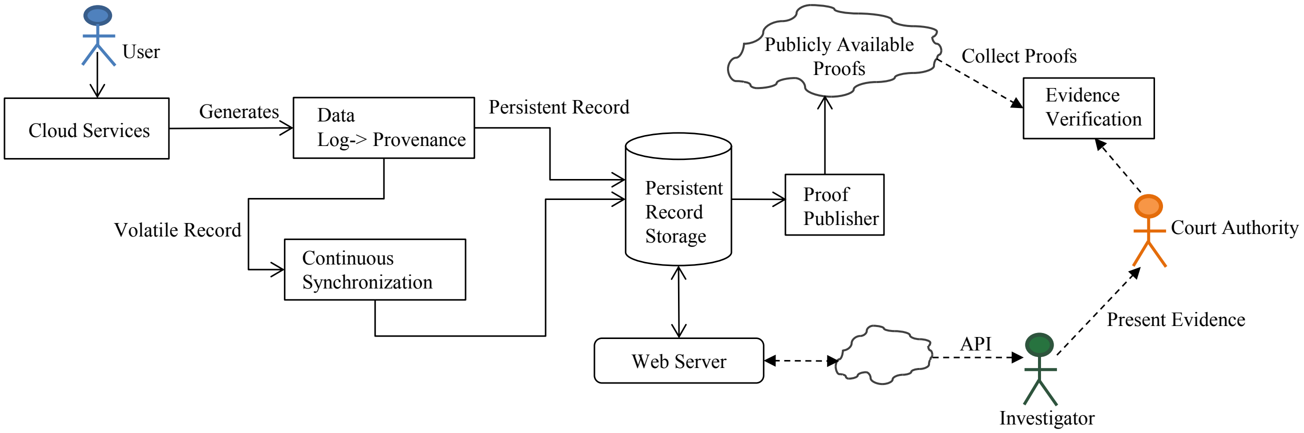

Fig. 4: An OCF-Supported Cloud

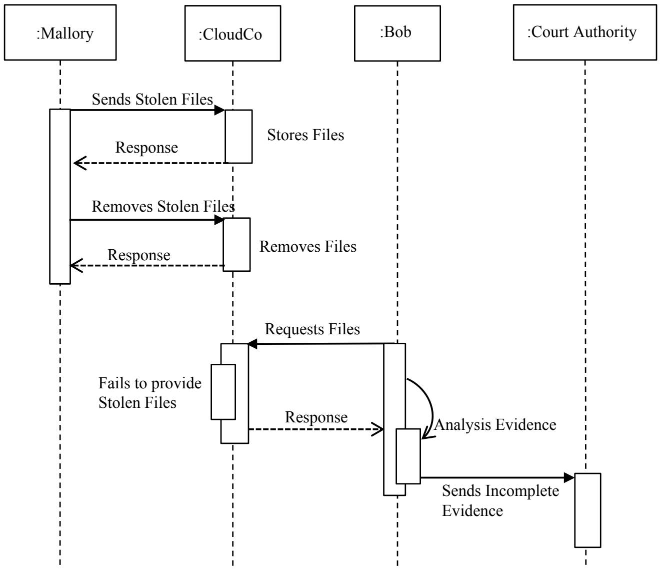

Fig. 5: Unreliable Digital Forensics on Current Clouds

Bob could collude with Mallory to remove evidence or provide false evidence to exonerate her. There is no reliable way for the court to verify the evidence in the current situation.

B. Reliable Digital Forensics by OCF Supported Cloud

First, let's assume that *CloudCo*, the cloud service provider that Mallory used to store the stolen intellectual properties, deployed a OCF supported cloud infrastructure. Then, the sequence of action presented in Figure 6 will be executed.

Mallory uses the storage as a service provided by *CloudCo*. Mallory sent a stolen code file to *CloudCo* at time t. According to the OCF model, this action will create an ESI $E^{S_{storage}}_{tMallory}$. *CloudCo* stores this ESI and creates a proof of the ESI $VE^{S_{storage}}_{tMallory}$ and publishes it to the Internet, which will make the evidence verifiable. Even if Mallory stores the stolen codes in a VM and terminates the VM after using the code, the continuous synchronization scheme will store the proof $VE^{S_{storage}}_{tMallory}$, which cannot be altered by any of the entity.

Later, Bob collects files from *CloudCo* using the secure API, analyzes and presents the evidence to the court. The court authority collects proof of evidence from the Internet and verifies the integrity of the evidence provided by Bob. Mallory can remove the incriminating files and collude with *CloudCo*. However, neither of them can alter the proofs available on the Internet. Similarly, Bob cannot modify the proofs either. Hence, even if Mallory were to collude with Bob, then the court authority could detect any alteration of the evidence using the proofs. Hence, the proposed cloud forensics model can support reliable forensics in a strong adversarial scenario.

VI. Conclusion

Because of the widespread adoption of clouds, it is becoming increasingly important to ensure that clouds provide support for reliable digital forensics investigations. Making the cloud forensics-aware also has the broader impact of bringing regulatory compliance to the realm of clouds. In this paper, we first identified the limitations of digital forensics in current cloud infrastructures. By examining cloud architectures and various entities involved in a cloud, we defined the cloud forensics process flow and proposed the Open Cloud Forensics (OCF) model. This model can be used by cloud architects to design clouds that support trustworthy cloud forensics investigations. We proposed an OCF-supported cloud architecture and showed how it can support reliable digital forensics in a realistic scenario. In the future, we plan to implement the proposed OCF supported, forensics-aware cloud infrastructure.

References

[1] B. Deeter and K. Shen, "BVP Cloud Computing Index Crosses the $100 Billion Market Milestone," http://goo.gl/mEuEi4, 2013.

[2] IDC, "U.S. Public IT Cloud Services Revenue Projected to Reach $43.2 Billion in 2016," http://goo.gl/nXL4t3, 2012.

[3] INPUT, "Evolution of the cloud: The future of cloud computing in government," http://goo.gl/Ksuc5i, 2009, [Accessed May 5, 2014].

[4] Market Research Media, "Global cloud computing market forecast 2015-2020," http://goo.gl/AR3FBD, [Accessed May 5, 2014].

[5] J. Ruhnka and J. W. Bagby, "Litigation support and risk management for pretrial discovery of electronically stored information," *CPA JOURNAL*, vol. 77, no. 5, p. 50, 2007.

[6] Federal Rules of Civil Procedure, "Rule 34," http://goo.gl/NfL61.

[7] Infosecurity-magazine, "Ddos-ers launch attacks from amazon ec2," http://goo.gl/vrXrHE, July 2014, [Accessed September 25, 2014].

[8] The Register, "Amazon cloud hosts nasty banking trojan," http://goo.gl/xGNkNO, 2011, [Accessed July 9th, 2014].

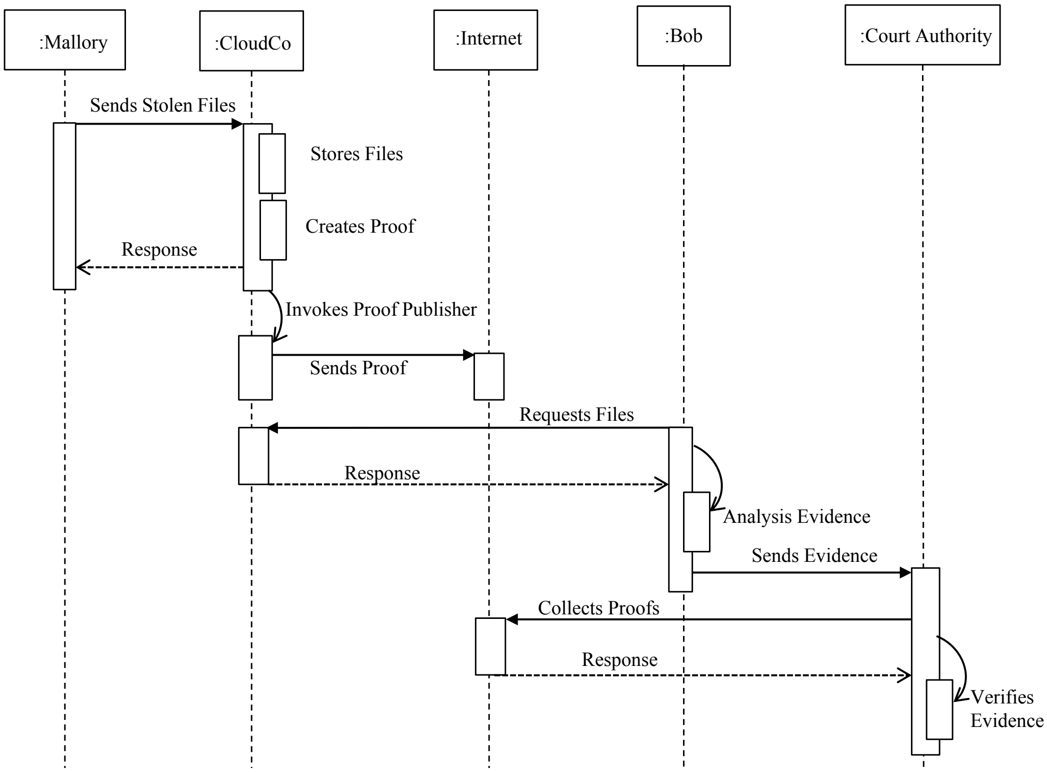

Fig. 6: Cloud Forensics Process Flow on an OCF Supported Cloud

[9] Dist. Court, SD Texas, "Quantlab technologies ltd. v. godlevsky," Civil Action No. 4: 09-cv-4039, 2014.

[10] www.bbc.com, "Lostprophets' Ian Watkins: 'Tech savvy' web haul," http://goo.gl/C8FVnC, December 2013.

[11] D. Birk and C. Wegener, "Technical issues of forensic investigatinos in cloud computing environments," *Systematic Approaches to Digital Forensic Engineering*, 2011.

[12] J. Dykstra and A. Sherman, "Understanding issues in cloud forensics: Two hypothetical case studies," *Journal of Network Forensics*, vol. b, no. 3, pp. 19–31, 2011.

[13] G. Grispos, T. Storer, and W. Glisson, "Calm before the storm: The challenges of cloud computing in digital forensics," *International Journal of Digital Crime and Forensics (IJDCF)*, 2012.

[14] D. Reilly, C. Wren, and T. Berry, "Cloud computing: Forensic challenges for law enforcement," in *proceedings of the Internet Technology and Secured Transactions (ICITST) Conference*. IEEE, 2010, pp. 1–7.

[15] J. Dykstra and A. Sherman, "Acquiring forensic evidence from infrastructure-as-a-service cloud computing: Exploring and evaluating tools, trust, and techniques," *DoD Cyber Crime Conference*, January 2012.

[16] J. Dykstra and A. T. Sherman, "Design and implementation of frost: Digital forensic tools for the OpenStack cloud computing platform," *Digital Investigation*, vol. 10, pp. S87–S95, 2013.

[17] S. Thorpe and I. Ray, "Detecting temporal inconsistency in virtual machine activity timelines." *Journal of Information Assurance & Security*, vol. 7, no. 1, 2012.

[18] K. Kent, S. Chevalier, T. Grance, and H. Dang, "Guide to integrating forensic techniques into incident response," *NIST Special Publication*, pp. 800–86, 2006.

[19] D. Lunn, "Computer forensics–an overview," *SANS Institute*, vol. 2002, 2000.

[20] J. Robbins, "An explanation of computer forensics," *National Forensics Center*, vol. 774, pp. 10–143, 2008.

[21] NIST, "Cloud forensic science," http://collaborate.nist.gov/twiki-cloud-computing/bin/view/CloudComputing/CloudForensics, [Accessed July 5th, 2014].

[22] P. Mell and T. Grance, "Nist cloud computing forensic science challenges," *Draft NISTIR 8006*, June 2014.

[23] D. Birk, "Technical challenges of forensic investigations in cloud computing environments," in *Workshop on Cryptography and Security in Clouds*, January 2011.

[24] H. Guo, B. Jin, and T. Shang, "Forensic investigations in cloud environments," in *Computer Science and Information Processing (CSIP), 2012 International Conference on*, Aug 2012, pp. 248–251.

[25] M. D. Ludwig Slusky, Parviz Partow-Navid, "Cloud computing and computer forensics for business applications," *Journal of Technology Research*, vol. 3, July 2012.

[26] D. Reilly, C. Wren, and T. Berry, "Cloud computing: Pros and cons for computer forensic investigations," *International Journal Multimedia and Image Processing (IJMIP)*, vol. 1, no. 1, pp. 26–34, 2011.

[27] S. Wolthusen, "Overcast: Forensic discovery in cloud environments," in *In proceedings of the Fifth International Conference on IT Security Incident Management and IT Forensics (IMF)*. Ieee, 2009, pp. 3–9.

[28] K. Ruan, J. Carthy, T. Kechadi, and M. Crosbie, "Cloud forensics: An overview," in *proceedings of the 7th IFIP International Conference on Digital Forensics*, 2011.

[29] R. Marty, "Cloud application logging for forensics," in *Proceedings of the ACM Symposium on Applied Computing*, 2011, pp. 178–184.

[30] Z. Zafarullah, F. Anwar, and Z. Anwar, "Digital forensics for Eucalyptus," in *Proceedings of Frontiers of Information Technology (FIT)*. IEEE, 2011, pp. 110–116.

[31] S. Zawoad, A. K. Dutta, and R. Hasan, "SecLaaS: Secure logging-as-a-service for cloud forensics," in *Proceedings of the 8th ACM Symposium on Information, Computer and Communications Security (ASIACCS)*. ACM, 2013.

[32] L. Lamport, "Time, clocks, and the ordering of events in a distributed system," *Communications of the ACM*, vol. 21, no. 7, pp. 558–565, 1978.

[33] M. K. Waldo Delport, Martin S. Olivier, "Isolating a cloud instance for a digital forensic investigation," in *Information and Computer Security Architecture (ICSA)*, 2011.

[34] B. Hay and K. Nance, "Forensics examination of volatile system data using virtual introspection," *ACM SIGOPS Operating Systems Review*, vol. 42, no. 3, pp. 74–82, 2008.

[35] A. Patrascu and V.-V. Patriciu, "Logging system for cloud computing forensic environments," *Journal of Control Engineering and Applied Informatics*, vol. 16, no. 1, pp. 80–88, 2014.

[36] J. Dykstra and A. T. Sherman, "Design and implementation of frost: Digital forensic tools for the OpenStack cloud computing platform," *Digital Investigation*, vol. 10, pp. S87–S95, 2013.