Towards a Forensic-based Service Oriented Architecture Framework for Auditing of Cloud Logs

Sean Thorpe
Computational Science Research Group
University of Technology
Kingston, Jamaica
sthorpe@utech.edu.jm
Membership Number: 92312894

Tyrone Grandison
Proficiency Labs
Ashland, Oregon
tgrandison@proficiencylabs.com

Indrajit Ray
Department of Computer Science
Colorado State University
Fort Collins, USA
indrajit@cs.colostate.edu

Arnett Campbell, Janet Williams, Khalilah Burrell
Computational Science Research Group
University of Technology
Kingston, Jamaica
arcampbell,jwalters,kburrell{@utech.edu.jm}

***Abstract*— Cloud computing log digital investigations relate to the investigation of a potential crime using the digital forensic evidence from a virtual machine (VM) host operating system using the hypervisor event logs. In cloud digital log forensics, work on the forensic reconstruction of evidence on VM hosts system is required, but with the heterogeneous complexity involved with an enterprise's private cloud not to mention public cloud distributed environments, a possible Web Services-centric approach may be required for such log supported investigations. A data cloud log forensics service oriented architecture (SOA) audit framework for this type of forensic examination needs to allow for the reconstruction of transactions spanning multiple VM hosts, platforms and applications. This paper explores the requirements of a cloud log forensics SOA framework for performing effective digital investigation examinations in these abstract web services environments. This framework will be necessary in order to develop investigative and forensic auditing tools and techniques for use in cloud based log-centric SOAs.**

Keywords-SOA;cloud;web; forensic

I. INTRODUCTION

Cloud Computing is the elastic on-demand provision of scalable virtualization technology services to end users of web-enabled logical domains [1].

Cloud forensics is an amalgamation of cloud computing and traditional digital forensics. Against this background, data cloud provisions are based on the use of service oriented architectures. By definition, service oriented architectures and web services facilitate the integration of enterprise applications between businesses and government organizations both within the physical, as well as the logical, domains of a data cloud. The cost of integration and enhanced flexibility is increased heterogonous complexity. As more organizations adopt cloud-enabled web services for increasingly sensitive, mission-critical data, the potential impact of breaches of Web services increases both for individuals and organizations.

Increasing impacts can result in a worsening of the risk environment for all parties. Cloud-based web services security and auditing is therefore an important concern. The services oriented architecture paradigm presents a number of significant challenges with respect to the auditing and monitoring of cloud-based transactions. The need to provide forensic auditing tools that can aid in the investigation of breaches of security, deliberate or accidental, in such abstract environments is obvious. Such techniques increase the possibility of detection and apprehension of criminal actors and aid in the assurance of the transaction process for all involved. An increased level of assurance of such logical systems should ease concerns with the utilization of web services technologies, thus opening opportunities for government, business and individuals in the near future.

This paper explains how cloud forensics can contribute to the security and assurance of cloud-enabled service oriented architectures, improving the confidence of vested stakeholders using these domains, and reducing the confidence of potential attackers that they may be anonymous and may go undetected. We discuss the challenges in cloud forensic investigations involving Web services, and suggests ways in which they may be overcome. Additionally, this work identifies the need for a cloud SOA framework for developing Web services that record enough potential evidence to support and complement a manual data centre investigation.

II. RELATED WORK – SERVICED ORIENTED ARCHITECTURE

Service oriented architecture (SOA) describes a paradigm for the development, deployment and use of online software systems working on the basis of a service provider publishing a description of the services it can provide, in a form of registry, which is queried by clients in order to discover and then dynamically invoke the desired services [4]. In this paper, we will use the abbreviation SOA both to refer to the paradigm and to specific systems implementing

it. This paper focuses on Web services, the best known examples of SOAs, in which the mechanism of publication, discovery and invocation is facilitated through the use of standard Web formats and protocols [15]. There are at least two participants in any SOA transaction – the cloud service provider and the cloud service requester. Both are software agents, representing different individuals or cloud organizations (or perhaps different sections of the same organization). A given cloud service requester may not know which cloud service provider has the desired service; it simply knows which service it requires, and interrogates the registries of known cloud service providers to find the service. The cloud service requester can then select its desired service and invoke it. Web services use standardized Internet technologies, such as XML, to implement a platform independent and interoperable SOA.

A Web service has an interface described in a machine-process format called the Web Service Description Language (WSDL). This WSDL interface defines the message formats, data types, transport protocols, and serialization formats that a Web service requester should use when it interacts with the Web service. It is, in essence an agreement not dissimilar to the contract programming model of agreed specifications of APIs, except it is machine processed and thus machine-enforceable [5]. In practice, many Web service clients are configured with pointers to the WSDL describing the services a company wishes to provide. The initial vision of the SOA community was that Web service requesters would obtain the WSDL for a Web service through querying the Web service provider's Universal Description, Discovery and Integration (UDDI) registry [8]. Simple Object Access Protocol (SOAP) is used as the message format for messages between the Web service requester and the provider, consisting of formatted XML requests and XML responses. Through the use of these standard formats for the registry, the interface, and messages, any conforming software agent, no matter the language in which it was written or the platform for which it was written, can take the place of the provider or the requester.

III. Computer Forensics

The term computer forensics describes the discovery, examination and analysis of digital evidence typically stored on or generated by a computer or computer system. Computer forensics is the investigation of situations where there is computer-based (digital) or electronic evidence of a crime or suspicious behavior [19]. Investigations of breaches of security or suspicious events in, or transaction auditing of SOAs, would employ digital evidence in an effort to reconstruct the events under investigation. The distributed nature of SOAs poses particular challenges to cloud forensic investigations, but the standards-driven nature of SOAs also provides an opportunity to address those challenges.

The challenges faced in forensic investigations of cloud based SOAs include challenges faced in the traditional digital forensic investigations of any distributed physical network system. In conventional computer forensic investigations of stand-alone computer systems, there is one primary source of digital evidence – the computer's hard disk. In network forensics, there are a number of different potential sources of digital evidence. However, the technical difficulty and expense involved in recording large volumes of network data, coupled with the lack of economic incentive to collect such information, means that the wider variety of potential sources does not translate into a larger volume of digital evidence. In fact, most network forensic systems are highly ad hoc in nature, depending on network eavesdropping tools such as packet capture software to monitor key points in the network [13]. There are significant technical challenges in accurately reconstructing network traffic through analyzing the recordings of such eavesdropping tools even in ideal circumstances. Eavesdropping tools are also vulnerable to simple confusion techniques making it easy for an attacker to deliberately obfuscate their actions [3]. Regardless of the difficulty of undertaking forensic investigations in a distributed network environment, it is nevertheless desirable to have the capability.

IV. The Need for Cloud Forensics

The relationship between forensics and overall system security is harder to see than the direct relationship seen between, for example, a firewall and network security. No security system is ideal and presents the suitable and important roles of forensics. Robust and accurate forensic techniques increase the likelihood both of detection of malfeasance and final attribution of the illicit actions to the perpetrator. There is no suggestion at present that the use of cloud-enabled web services provides a new set of actual criminal aims. It may, however, provide a new set of *ways* that criminal acts may be committed. The set of influences that may contribute to an adversary's decision to act is complex. Once a target is defined for any attack, an adversary will require some set of capabilities and resources to undertake the attack [10]. The nature of the system itself and the security measures in place will, to a large extent, determine these requirements.

Simply having the capability and resources to act does not make the action inevitable. A combination of factors such as perceived benefit, level of potential punishment, and so on will come into play before any actor will take action. It may not necessarily follow that an adversary will perceive a system with a high degree of security measures in place as a higher risk target. Ideally, however, the aim is to make the system both difficult to attack and to increase the attacker's perception of risk in attacking the system. Various studies of risk perceptions have identified the *affect heuristic* as a factor in determining the level of perceived risk for some action or event and this is no different for the cloud. If, the benefit is seen as low then the risk is perceived as higher and if risk is perceived as higher then benefit is perceived as lower [11]. Perceptions of likelihood of detection and consequent identification have also been identified as possible modifiers on the behavior of potential adversaries [10]. Therefore the role of digital forensics is to increase the perceived risk for an actor.

The ability to reconstruct some set of transactions allows for an increase in trust for all parties. Primarily, it allows for some reasonable expectation that disputes over transactions may be solved in something other than an arbitrary manner. Digital forensics provides a set of tools to produce information, which can, to some degree of accuracy, reconstruct the sequence of events involved in a transaction. This level of surety is obviously useful for civil dispute resolution. While the behavior of actors is complex and factors other than those discussed here will clearly come into play. The following section discusses the challenges faced in undertaking such investigations.

V. Investigative Challenges in Cloud-Enabled Service Oriented Architectures

Given the documented difficulties facing forensic investigations of traditional physical networks, it seems desirable to avoid similar difficulties in forensic investigations of cloud-based network SOAs. A greater commonality of interest exists between a cloud service requester and a cloud service provider in an SOA transaction than exists in a generic network transaction. This commonality of interest should make it easier for both parties to work together to introduce forensic systems which will allow them to improve security in service oriented architectures. The major issue facing forensic investigations of network systems is the lack of relevant evidence collected for the specific purpose of such an investigation. The collection of forensic data in cloud-based networks is, for the most part, an ad hoc process, dependent on the likes of firewall logs, intrusion detection system logs, network eavesdropper logs, and so on. The scope of data collected from such sources is too narrow for many purposes [13].

However, in SOAs, all major stakeholders have an interest in facilitating the post hoc forensic investigation and audit of SOA transactions. There are, nevertheless, a number of challenges which confront both the collection of adequate digital evidence to facilitate post hoc investigation and the actual post hoc forensic investigation of SOAs. These challenges originate from either social or technical considerations. Challenges to the actual forensic investigation are mostly technical in nature. Challenges to developing the ability to collect adequate data to conduct such an investigation can be both technical and social.

VI. Technical Challenges

Web services are platform independent, which is to say that they are completely interoperable irrespective of the network configuration, hardware and software employed by the cloud provider and cloud requester. It is this platform independence which poses the most obvious technical challenge to a forensic investigation. Each platform involved will require a particular set of tools and techniques to be used in evidence recovery. This will be especially true of any data collected by a VM host operating system or runtime environment specific tool, such as hypervisor system logs, or by a network monitoring tool specific to a certain network configuration, such as firewall logs. Each platform has its own inherent issues, which can further complicate matters for a forensic investigator. For example, the amount of detail in hypervisor system event logs on say a Windows Azure system is highly dependent on the auditing configuration of the Windows VM host involved, and the procedure is different altogether on Unix-like VM host systems like say Xen Citrix. Difficulties for forensic investigations dependent on the general logging and audit tools provided for particular operating systems or platform are likely to persist while Web services transactions take place between disparate VM hosts, VM configurations and platforms.

Likewise, the disparity between the information found from the firewall logs, IDS logs, and other sensor logs of two different cloud networks is unlikely to be resolved while forensic investigations of Web services are dependent on this sort of generic network sensor information. Cloud forensic systems dependent on eavesdropping tools face a number of difficulties, and are vulnerable to deliberate confusion techniques by attackers with obvious interest in obscuring their actions. A forensic data collection system dependent on traffic interception must be sufficiently "sensitive", which is to say that it receives all messages exchanged between the service provider and requester. It must also be "selective", meaning it rejects spurious data which can make it difficult for investigators to recognize data relevant to the cloud investigation. Whilst sensitivity is a well-understood requirement, selectivity of traffic sensors in network forensics is often misunderstood and thought to be easily achieved through only a token evaluation of traffic metadata [3]. The sheer quantity of data collected in traditional forensic investigations has been recognized as a challenge which confronts researchers and investigators alike, and becomes exacerbated when migrated to the cloud domain. Excessive volumes of data can make the search for relevant digital evidence somewhat like searching for a needle in a haystack. Solutions to massive datasets in stand-alone computer forensics include data mining [2] and our own work in VM automated profiling to help narrow the field of search [6, 7], but it seems that by focusing on traffic sensor selectivity, this problem could be largely avoided and or significantly reduced in a cloud SOA forensic investigations. Web services are generally stateless from one invocation to the next [17], meaning that it may not always be necessary for SOAP traffic monitoring systems to attempt to keep a record of the state of a Web service provider. In the case of certain complex Web services, however, it may still be necessary to track the state of each invocation.

The technical standards which specify the Web services architecture themselves pose a challenge to the introduction of forensic data collection into a Web services environment. Within the standards, which apply to Web services, there is a lack of consideration for the collection and storage of digital evidence for the purposes of post hoc investigation or auditing. The cloud infrastructure storage of raw network data is impractical due to the high volume of data which would be recorded. The high storage capacity requirements of raw network data would introduce excessive expense, or

longevity concerns, due to the need to overwrite old data to conserve space [13]. Storing higher-level data would reduce the required storage capacity, thereby allowing the record of a longer time period to be maintained. Given the standardized nature of the technologies employed in Web services, it should be possible to collect a narrower set of data rather than simply collecting all network data. As an example, SOAP requests for service invocation and the Web service's SOAP response could be stored, providing investigators with the cloud requester's input and the service's output. This would allow investigators to reconstruct the SOA transaction. The solutions to the range of technical challenges described in this section are tractable. The adopted cloud SOA is not a purely technical system however; social influences will play their part. In the next section, we briefly discuss possible social challenges that may need to be considered.

VII. Social Challenges

For the purposes of this paper, we consider social challenges to be problems that arise not from a direct technical difficulty but from the parties involved in the use or development of the cloud SOA. One considers social problems to be those where a technical solution may exist or be capable of development but there is resistance to such development or deployment. The possible ways in which social challenges may present themselves are many and is the subject of an independent paper. Some possible considerations are outlined. One difficulty in deciding on how to approach the problem of making provision for cloud forensic investigation in SOAs is the difficulty in defining exactly what it is they will do. By their very nature SOAs form webs of cloud applications which can be put together in an ad hoc, as needed basis. Therefore deciding in advance exactly what information is involved in an exchange is almost impossible. The unknown nature of the exact cloud VM transactions makes it difficult to decide what may be safe to store or not store or indeed what may be required to reconstruct the transaction. It is therefore difficult to answer privacy and confidentiality concerns before the actual transaction takes place. A company, for example, may wish their use of a certain service to remain confidential. The requirement of a cloud forensic investigation that actions may be able to be assigned to a fixed party then may become problematic. A challenge then is to balance the competing needs of cloud forensic investigations and concerns of the parties involved in the cloud transactions. For the developers themselves social problems exist. Security measures in general have been considered to be an impediment to development of product. Add to this, the possibility that extra resources may be required to fulfill the requirement, and cloud forensics may then become a source of weakening the business case for the use of SOAs in the first place. In concert with these factors, developers may be over-confident in their ability to produce a completely secure cloud system, reducing impetus to facilitate investigation. For example, Oracle's declaration of an "unbreakable" system that have proven to be optimistic [12]. It is difficult to conceive of a system so secure that there exists no possibility of a breach, and where that possibility exists then so does the requirement for investigation.

Finally, there is a potential for conflict between cloud requesters and cloud providers of Web services. The use to which information can be put is a topic of growing concern. The need that different parties see for forensic information may differ and could require different kinds of information. This has the potential to cause conflict between what the parties are *willing* to provide and what is *necessary* to provide. Therefore any definition of information requirements must have a means for conflict resolution. This section has outlined potential social impediments to implementing a comprehensive system for carrying out cloud forensic examinations in SOAs. Next, we define what the likely requirements for a cloud SOA framework.

VIII. Requirements for a Cloud Based SOA Framework

The standards-driven nature of Web services in particular, and service oriented architectures generally, provides an opportunity to incorporate forensic data collection standards as an intrinsic part of cloud SOAs. Providing for forensic data collection in a standardized form has benefits for a cloud investigation. These benefits include greater efficiency through similarity in the process of discovering digital evidence, and greater confidence in the quality of that digital evidence. One therefore proposes that a framework to support forensic investigations be incorporated into the standards that govern cloud SOAs. Standardizing forensic data collection in cloud SOAs would address the challenges discussed earlier through the process of establishing an industry standard, and by providing a mechanism for the parties in a cloud SOA transaction to negotiate what information would be recorded. The process of expanding existing SOA standards or writing new ones to accommodate cloud forensic data collection would necessitate enumerating the concerns of legitimate cloud SOA participants. These concerns could be evaluated from the perspectives of all involved parties, and a middle ground could be determined which was in the common interest.

Such a standard could also incorporate a "handshaking" stage, at which both the cloud service requester and the cloud service provider would agree to the amount of information stored about a service invocation. If the parties could not agree on a required level of identifying information the invocation request could be rejected. The introduction of such a stage would mean that both parties had to find a mutually agreeable level of information which could be recorded about the transaction before that transaction took place. If cloud forensic data collection were to become part of the standards-mandated framework for SOAs, any potential competitive disadvantage to conducting such data collection would be eliminated. While building the capacity to collect cloud forensic data about SOA transactions remains optional, providers who choose not to implement such measures may enjoy a competitive advantage, especially in time-to-market terms. Such an advantage is gained at the expense of the overall security

posture of the system, which could have repercussions for the service provider's clients. Without the capacity for forensic data collection being included in an accepted standard, a potential customer has no capacity to build confidence in the security of a given service.

The incorporation of forensic data collection systems into SOA standards would level the playing field between all service providers. A forensic data collection system for SOAs must include a sensor and a log for the monitoring and storage of messages. As investigations into SOAs will primarily concern higher-level application logic (e.g. the details of a service invocation) rather than lower-level network traffic, every piece of network traffic need not be monitored and recorded. It may be desirable to allow configuration as to which messages are logged, in-line with privacy or other concerns. The sensor must be placed logically within the SOA to intercept incoming SOAP messages prior to their processing, as well as outgoing SOAP messages. The sensor should not process message payloads; it should merely record them in the log. Many attacks on SOAs consist of messages with payloads containing attack code, or which are over-sized and take an excessively long period of time to parse. For example, Web services are vulnerable to attacks which cause a denial of service by providing well-formed XML documents to a service which are oversized or contain excessive nesting of elements [14]. Yu categorizes just fewer than 60% of attacks against Web services and applications as "input manipulation" attacks, which prey on processing an attacker's input [16].

In order to avoid the forensic data collection system failing to record or even being brought down by the very sorts of attacks it is supposed to help investigate; its sensor must record messages in its hypervisor logs prior to any processing of the message's content. A framework to support the post incident forensic investigation of service oriented architectures can be established through the incorporation of cloud forensic log data collection into SOA standards. Such a framework needs to ensure that such a forensic data collection system provides continuous service even during attacks on the SOA. It can provide a mechanism for the parties involved in an SOA cloud transaction to negotiate about the level of information to be stored about the transaction. A standardized framework would make forensic investigations more efficient, and raise consumer confidence in SOA security. We propose the adoption of a standards-driven framework for data collection to facilitate cloud forensic investigations of SOAs. In the next section, we propose a strawman cloud audit approach for such a framework.

IX. CLOUD FORENSIC AUDIT FRAMEWORK

We adopt from [1] that a cloud forensic auditor is a party that can perform an independent forensic examination of cloud service controls with the intent to express a legal opinion. Forensic audits are performed to verify conformance to standards through review of objective evidence. A cloud forensic auditor can evaluate the services provided by a cloud service provider in terms of security controls, privacy impact, performance, etc. The audit will include interactions between the cloud customer and the cloud service .

Forensic capabilities and segregation of duties between cloud actors in delivering these capabilities, to facilitate both internal and external cloud investigations, need to be reflected into auditable regulatory or contractual language. Currently, this is still missing from the literature. A key set of terms for service level agreements have been identified and recommended by Ruan et.al. [19].

As a basis of understanding for the SLAs that the auditor could provide, lets evaluate the relationship between the cloud consumer, the cloud service provider (sometimes seen as the cloud carrier based on the jurisdiction). As matter of distinction, however, the cloud carrier acts as that intermediary that provides connectivity and transport of cloud services between the cloud customers and providers, where such services are enabled through network, telecommunication, and other access devices. Typically, the cloud provider arranges for two unique SLAs, one with a cloud carrier (e.g. SLA2) and one with a cloud requester/consumer (e.g. SLA1). A cloud provider may request dedicated and encrypted connections to ensure the cloud services are consumed at a consistent level according to the contractual obligations with the cloud consumers. In this case, the provider may specify its requirements on capability, flexibility and functionality in SLA2 in order to provide essential requirements in SLA1.

In the ideal case, carriers are not likely to be involved with the cloud forensic investigation. However, they can play a useful role in providing pre-investigative and supportive capabilities, such as evidence transport, claim of custody, and inter-cloud forensic capabilities.

As basis of understanding the system components for delivering the cloud services mentioned, lets take a look at the NIST cloud stack architecture [1] (figure 1). The generic stack diagram (figure 1) shows a grouping of three types of system components for delivering cloud, i.e. Physical Resource Layer, a Resource abstraction layer, and a Service Layer. Similar to the traditional computer systems stack, a list of forensic artifacts and its order of volatility need to be identified and specified for the cloud system stack. The following few paragraphs describe these stack layers.

The physical resource layer includes hardware computing resources such as computers (CPU and memory), networks (routers, firewalls, switches, network links and interfaces, such as heating, ventilation, and air conditioning (HVAC), power, communications, and other aspects of the physical data centre environment. A forensic artifact for the hardware layer includes hard disk, network logs, router logs, etc. This layer also includes data centre artifacts such as access records, facility logs, activity logs, interior and exterior camera footage, biometric records, visitor records, organization charts, contact information etc.

The resource abstraction and control layer contains the system components that cloud providers use to provide and manage access to the physical computing resources through the software abstraction. Resource abstraction components typically include software elements, such as virtual

machines, hypervisors, virtual storage data, and other resource usage abstractions. Forensic artifacts in this layer include hypervisor event logs, virtual images, etc.

The service layer is the layer where the cloud provider defines interfaces for cloud consumers to access computing services. Access interfaces for each of the three service models are provided at this layer. The service layer is where the segregation of duties between the provider and the consumer comes in, and the segregation is where the interface is. Forensic artifacts that reside from the service interface above can be collected by the consumer. Forensic artifacts that reside from the service interface below (including the Resource Abstraction and Control layer and Physical Resource layer) need to be collected by the provider. Ideally a set of standardized forensic interfaces need to be defined and integrated into different service layers corresponding to forensic capabilities required by both consumer and provider.

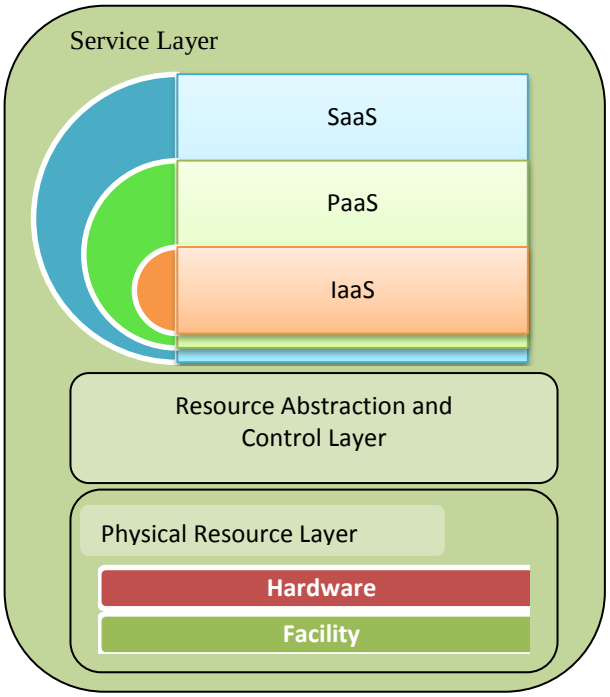

Figure 1. Cloud System Environment.

The IAAS interface layer can also be described as the OS (Operating System) as this layer accesses operating systems and drivers, and is hidden from SAAS and PAAS consumers. An IAAS cloud allows one or more OS's to run virtualized on a single physical host. Generally consumer, have broad freedom to choose which OS to be hosted among all the OS's that could be supported by the cloud provider. The IAAS consumers should assume full responsibility for the guest OS, while the IAAS provider issues responsibility for the host OS [1]. Forensic artifacts on this layer are similar to artifacts in virtual OSes.

The PAAS can also be called the Middleware Layer, as this layer provides software building blocks (e.g. libraries, databases, Java Virtual Machine) for developing application software within the cloud. The middleware is used by PAAS consumers, and is installed, managed and maintained by IAAS consumers, or PAAS providers, and hidden from SAAS consumers. Forensic artifacts on this layer are similar to artifacts in traditional (integrated) development environments, which include source code, performance logs, debugging logs, access logs, account information etc.

The SAAS layer can also be called the Application layer, as this layer includes software applications targeted at end users or programs. The programs are used by SAAS consumers, or they installed, managed and maintained by PAAS consumers, IAAS consumers or PAAS providers. Forensic artifacts on this layer are similar to artifacts in traditional software applications, e.g. application logs, authentication and authorization logs, account information, etc. The only difference is that the software is hosted remotely from the consumer via the browser (or via other thin or thick clients) thus thin/thick client forensic data collection will play a major role in forensic data collection on this layer from the consumer side.

Based on the discussion above, researchers argue that forensic acquisition within the cloud has to resort to a hybrid approach remote, live, virtual, network, thin client, thick client, and large scale acquisition due to the nature of the artifacts in the cloud environments. A list of proactive forensic artifacts needs to be identified across the cloud system stack to ensure forensic readiness. The identification of pro-active artifacts must evolve closely with the development of cloud solutions. Equally, a list of reactive artifacts also needs to be identified with the order of volatility for post incident forensic evidence collection. On the latter point, the authors have contributed to development of artifact repositories that profile the abstraction layer activities [6,7] using the hypervisor event logs. However, a lot more still needs to be done across all the layers. Arguably some of the eDiscovery methodologies can be borrowed in identifying and collecting reactive forensic artifacts, such as creating data maps [18] for these artifacts.

Formally as a part of collecting these forensic artifacts, an understanding of the type of cloud interactions that exist becomes relevant. There are various ways for cloud actors to interact in cloud investigations. There are three major organizational scenarios for a cloud investigation based on the analysis of the forensic implications of the three main usage scenarios described Liu et.al. [1]. The interaction scenarios are detailed views of the organizational dimension described by Ruan et.al. [19] and are analyzed under the aspects of SLAs, internal and external investigations, and forensic artifacts. In scenario 1 below depicted by Figure 2, the simplest scenario for cloud actors' interaction is provided.

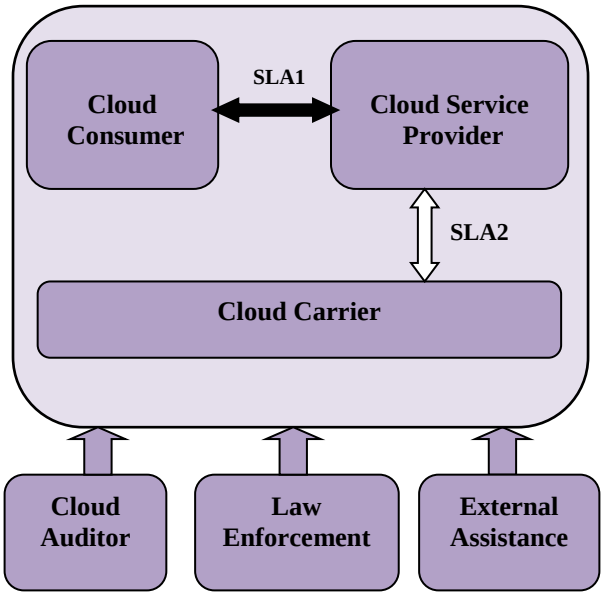

Figure 2. Cloud Actors interaction scenario 1

In a service offering there is a single relation between the cloud consumer and the cloud provider, where the cloud provider may or may not provide services via a cloud carrier. The consumer signs an SLA (SLA1) with the provider. The provider signs a separate SLA (SLA2) with the carrier when the relation between carrier and the provider exist. A cloud auditor may be involved to audit SLA(s). Forensic segregation of duties, requirements and implementations need to be defined and audited through the SLA(s). An internal investigation exists when the consumer and the provider shared systems. An external investigation is initiated by law enforcement towards the consumer, provider or shared system used by both parties. Provider or consumer may resort to external assistance in enhancing forensic capabilities in facing in internal or external investigations. Forensic artifacts are scattered between the consumer and producer systems.

In scenario 2 (figure 3 below), the cloud broker is acting as a cloud provider to the cloud consumer. The consumer signs an SLA A with the broker. The broker signs a range of SLAs (SLA B1, SLA B2, SLA B3 and so on) with multiple providers, and may sign a separate SLA, C, with a cloud carrier when services are delivered through the carrier.

The actual provider(s) is invisible to the cloud consumer. A cloud auditor may be involved to audit SLAs. Forensic segregation of duties, requirements and implementations need to be defined and audited through the SLA(s). An internal investigation happens within the shared cloud environment among cloud consumer, broker and provider(s). Forensic artifacts are scattered across consumer, provider and broker systems.

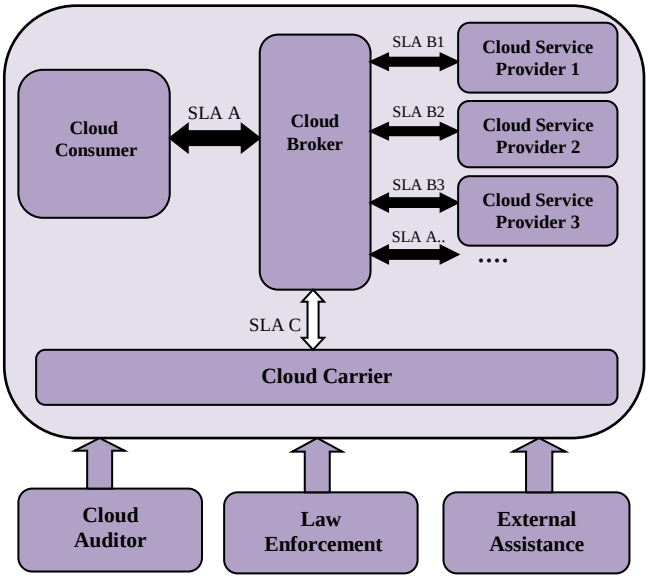

Figure 3. Cloud Actors interaction in scenario 2

In a third scenario (figure 4), there is a linear chain of dependencies between cloud entities. One cloud consumer uses service(s) from a cloud provider, which uses services from a third party cloud provider. This is similar to scenario 1. Each pair of service relation between the two cloud entities is defined via a SLA (e.g. SLA A1, SLA A2,….). In cases where the services are delivered through a cloud carrier, separate SLAs (e.g. SLA B1, SLA B2, SLA B3) are specified between the cloud entity and the cloud carrier. A cloud auditor might be involved to audit the SLAs among entities, in which case forensic requirements and performances should be audited and evaluated. An internal investigation happens within the cloud system shared among the chain of cloud entities that may affect the whole chain of cloud entities later on in the investigative process. Any pair of cloud entities on the two sides of the SLA may resort to external assistance in enhancing forensic capabilities in both the internal and external investigations, which should be specified within the SLA. Forensic artifacts are scattered throughout the chain of the cloud entities in the shared environments. Segregation of duties between each pair of cloud entities is similar to scenario 1 explained earlier.

In general the cloud audit interaction scenarios described under this proposed framework arguably suggest that there are clearly different forensic outcomes that are possible. We also believe that this view is compounded given the different cloud deployment models now used by cloud providers and consumers, namely: public clouds, private clouds, community clouds, and hybrid clouds. Given the relevance to the proposed SOA framework, we take a brief look at there technical, organizational and legal dimensional relevance for the cloud auditor. In the case of the public cloud, the infrastructure and the computing resources are made available to the general public over a public network. In this case, cloud consumers are often small enterprises or personal users who have minimum or no forensic capabilities, or large enterprise or government agencies

seeking cheap deployment of non-mission critical services. Technically, this deployment allows for easy registration and anonymous usage that could be exploited by malicious insiders. As such providers need to provide service provisions that deliver strong capabilities in evidence segregation in elastic multi-tenant environment and evidence acquisition with the proliferation multiple client end points. The provider must manage control of organizational service policy, and, legally, multiple jurisdictional SLAs are a standard with little or any room for any customization.

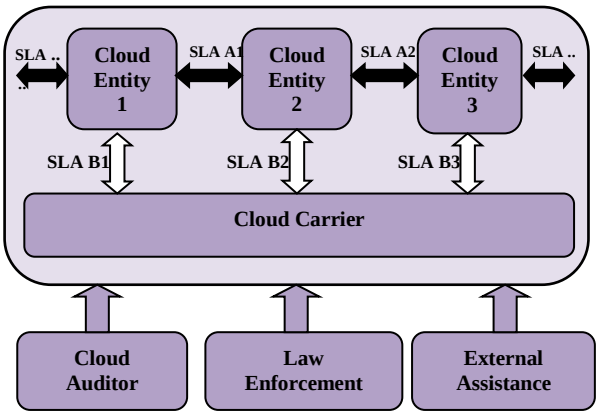

Figure 4. Cloud Actors Interaction Scenario 3.

In the event where the public cloud deployment is internal to the organization, the cloud consumers will have a certain level of internal security/forensic implementations, thus migrating to the cloud can result in an upgrade in security/forensic implementations from the consumer side. An extra layer of authorization/access control can be added through the enterprise network. Organizationally, the consumer will have some responsibility on policy and procedures on forensic implementations. Legally the consumer in this case can decide where his/her data resides as form of jurisdictional control via the SLA.

A private cloud setup, on the other hand, gives a single cloud consumer's organization the exclusive access to and usage of infrastructure and computational resources. It is likely to be managed by either the cloud consumer organization or by a third party. It may be hosted on site on the organization's premises or it may be hosted off site to the outsourced third party provider. In the case of the internal setup, the consumers will encourage a high level of internal security and forensics implementation before cloud migration is done. From an organizational standpoint, collaborative efforts need to be made by forensic teams from both the consumer and provider side to deliver strong forensic capabilities. Legally, data reside on premise, therefore evidence will be in the same jurisdiction as the customers.

Outsourced private clouds are normally cheaper than onsite private clouds as maintenance is off premises. The legal implications in a private cloud are that data can be in multiple jurisdictions, which makes SLA for case evidence difficult.

Community clouds serve as a group of consumers who have shared concerns such as mission objectives, security, privacy, and compliance policy, rather serving a single organization as a private cloud does, e.g. IBM's Federal Community Cloud (FCC) serves federal organizations. Similar to private clouds, community clouds can be managed and maintained by third parties and be deployed on site or off site. For an onsite community cloud, resources are managed by a single host or by multiple host organizations with a joint effort. Evidence segregation is needed among multiple tenant organizations using the same community cloud. Legally, evidence can be situated in multiple jurisdictions, when hosting and tenant organizations are geographically remote. In the case of the outsourced community cloud, multiple organizations share a private cloud hosted a cloud provider and consumers access the host remotely. Technically, support is provided by the private cloud host and the tenant organizations. Organizationally, policies are shared among provider and consumer organizations.

A hybrid cloud is a composition of two (2) or more cloud deployment models (on site/off site, private, community or public clouds) that remain distinct entities but are bound together by standardized or proprietary technology that enables data and application portability [18]. Security and forensic SLAs are extremely complex, and are subject of independent review.

Over time, the thoughts are to have a unified cloud audit framework model of multiple cloud platforms. In essence, the views presented for the cloud SOA are just initial musings. However, the guiding principles for building such a service would fall in line with this strawman approach for a cloud SOA audit framework.

X. CONCLUSION

This paper discussed the value of facilitating post incident cloud forensic investigations of service oriented architectures. The challenges are technical, organizational, legal and social – all of which hold back the integration of cloud data collection mechanisms to facilitate such investigations. Based on a preliminary analysis of the cloud reference architecture, the considerations presented are important for better integration of the missing considerations of forensic capabilities within a cloud forensic service oriented audit framework standardization process.

REFERENCES

[1] F.Liu ,J.Tong,J.Mao,J.Bohn,R.Messina,J.Badger,D.Leaf. NIST Cloud Computing Reference Architecture.National Institute of Standards and Technology,Special Publications 500-291.

[2] N.L. Beebe, and J. G. Clark (2005). Dealing with Terabyte Datasets in Digital Investigations.Research Advances in Digital Forensics. M. Pollitt and S. Shenoi. Norwell, Springer: 3-16. ComputerWire. (2002). "Oracle posts fix - servers 'unbreakable' again?"

[3] E. Cronin, M. Sherr, et al. (2006). On the Reliability of Network Eavesdropping Tools. Advances in Digital Forensics II. M. Olivier and S. Shenoi. Orlando, Springer: 199-213.

[4] S. Hashimi, (2003, 18 August 2003). "Service-Oriented Architecture Explained."Retrieved 28 July, 2007, http://www.ondotnet.com/pub/a/dotnet/2003/08/18/soa_explained.html.

[5] S. Jones, (2005). "Toward an acceptable definition of service [service-oriented architecture]." Software, IEEE 22(3): 87-93.

[6] S.Thorpe,I.Ray,T.Grandison,A.Barbir et.al (2013). Virtual Machine Profiling Tool Software Implementation and Evaluation within the Cloud Forensics data centre environment(under review).

[7] S.Thorpe, I.Ray,T.Grandison,A.Barbir, et al. (2013). Towards a Theoretical Hypervisor Log Centric Virtual Machine Profiling Object Model within the data cloud Forensics environment(under review).

[8] OASIS. (2004, 19 October 2004). "UDDI Version 3.0.2: UDDI Spec Technical Committee Draft." Retrieved 28 July, 2007, from http://www.oasis-open.org/committees/uddispec/doc/spec/v3/uddi-v3.0.2-20041019.htm.

[9] Office of the Privacy Commissioner. "Office of the Privacy Commissioner." Retrieved 31 July 2007, from http://www.privacy.gov.au/.

[10] T. Parker, E. Shaw, et al. (2004). Cyber Adversary Characterization Auditing the Hacker Mind. Rockland, Syngress Publishing Inc.

[11] E.M. Peters, B. Burraston, et al. (2004). "An Emotion-Based Model of Risk Perception and Stigma Susceptibility: Cognitive Appraisals of Emotion, Affective Reactivity, Worldviews, and Risk Perceptions in the Generation of Technological Stigma." Risk Analysis 24(5): 1349-1367.

[12] Reuters. (2001, 12/10/2001). "Hackers take up Larry Ellison's challenge." Retrieved 31 July, 2007, from http://www.usatoday.com/tech/news/2001/12/10/oracle-hackerschallenge.htm.

[13] K. Shanmugasundaram, H. Brönnimann, et al. (2005). Integrating Digital Forensics in Network Infrastructure. Research Advances in Digital Forensics. M. Pollitt and S. Shenoi. Norwell, Springer: 127-140.

[14] A.Singhal, and T. Winograd (2006). Guide to Secure Web Services (DRAFT). Recommendations of the National Institute of Standards and Technology, National Institute of Standards and Technology: 140.

[15] World Wide Web Consortium. (2004, 11 February 2004). "Web Services Architecture - W3C Working Group Note 11 February 2004." Retrieved 27 July, 2007, fromhttp://www.w3.org/TR/ws-arch/.

[16] W. D.Yu, D. Aravind, et al. (2006). Software Vulnerability Analysis for Web Services Software Systems. Computers and Communications, 2006. ISCC '06. Proceedings. 11th IEEE Symposium on.

[17] O. Zimmerman, M. Tomlinson, et al. (2003). Perspectives on Web Services: Applying SOAP, WSDL and UDDI to Real-World Projects. Berlin, Springer-Verlag.

[18] Australian Government (2007). “Do Not Call Register”. Retrieved July 31 2007,from https://www.dontocall.gov.au/. .

[19] K.Ruan,J.Carthy,T.Kechadi. Cloud Forensics-: Key terms for Service Level Agreements .Advances in Digital Forensics VIII, Springer.