THE JOURNAL OF
DIGITAL FORENSICS,
SECURITY AND LAW

Journal of Digital Forensics,
Security and Law

Volume 13 | Number 3 Article 7

9-30-2018

A Forensic Enabled Data Provenance Model for Public Cloud

Shariful Haque
University of Alabama, mshaque@crimson.ua.edu

Travis Atkison
University of Alabama, atkison@cs.ua.edu

Recommended Citation

Haque, Shariful and Atkison, Travis (2018) "A Forensic Enabled Data Provenance Model for Public Cloud," *Journal of Digital Forensics, Security and Law*: Vol. 13 : No. 3 , Article 7.
DOI: https://doi.org/10.15394/jdfsl.2018.1570
Available at: https://commons.erau.edu/jdfsl/vol13/iss3/7

A FORENSIC ENABLED DATA PROVENANCE MODEL FOR PUBLIC CLOUD

Md. Shariful Haque, Travis Atkison*
University of Alabama
Tuscaloosa, AL, USA
mshaque@crimson.ua.edu, atkison@cs.ua.edu

ABSTRACT

Cloud computing is a newly emerging technology where storage, computation and services are extensively shared among a large number of users through virtualization and distributed computing. This technology makes the process of detecting the physical location or ownership of a particular piece of data even more complicated. As a result, improvements in data provenance techniques became necessary. Provenance refers to the record describing the origin and other historical information about a piece of data. An advanced data provenance system will give forensic investigators a transparent idea about the data's lineage, and help to resolve disputes over controversial pieces of data by providing digital evidence. In this paper, the challenges of cloud architecture are identified, how this affects the existing forensic analysis and provenance techniques is discussed, and a model for efficient provenance collection and forensic analysis is proposed.

Keywords: data provenance, conceptual model, cloud architecture, digital forensic, digital evidence, data confidentiality, forensic requirements, provenance challenges

1. INTRODUCTION

In recent years, advancement in data processing and communication technology and theabundance of digital storage capacity enable coupling multiple computing resources tomanage large amount of data. The concept of cloud computing was developed to offerthis computing power and storage as service virtually. In such a utility-based businessmodel, a consumer can utilize the offered services on-demand following the "pay-as-you-go"approach (Voorsluys, Broberg, & Buyya, 2011). However, efficient data management must include some other parameters to maintain the quality of the data and enable reusing it. Data provenance or lineage is a form of metadata that stores the origin of a piece of data, keeps track of its ownership, and manages the history of the computational processing the data goes through in its lifetime. This data management technique is useful not only as a source of data regeneration or as a component for identifying errors through backward tracking, it also helps in regulatory purposes and resolving disputes by facilitating a proper investigation in digital forensics. These as-

*Corresponding Author

pects of provenance records make it an essential topic to be discussed in cloud architecture.

Clouds basically use the concept of multitenancy and virtualization to ensure efficient utilization of available resources. It identifies and solves some challenges of large scale data processing, such as on-demand resource allocation based on computational requirements, distribution and coordination of jobs among different machines, automatic recovery management, dynamic scaling of operations based on workloads, and releasing the machines when all the jobs are complete (Amazon Web Services, 2008). These features make cloud computing a popular choice for small and medium scale industries, and research says this market will expand with a 30% CAGR (Compound Annual Growth Rate) to reach 270 billion by 2020 (Market Research Media, 2016). As cloud computing grows in popularity, so do concerns about security, compliance, privacy and legal matters (Chung & Hermans, 2010). Storing data in a location with an unknown owner's record with thinner boundaries, and backing up data by an untrusted third party are a couple of the notable security concerns with cloud architecture (Hashizume, Rosado, Fernández-Medina, & Fernandez, 2013). These issues can eventually affect the trustworthiness of the data, as well as the metadata associated with it, making it difficult to find accurate evidence to apply in forensics. Though there are several digital forensic tools to apply in general IT scenarios, most of them have failed to prove their success over cloud architecture. Thus, new digital forensic methods must be developed to meet the challenges of a cloud architecture.

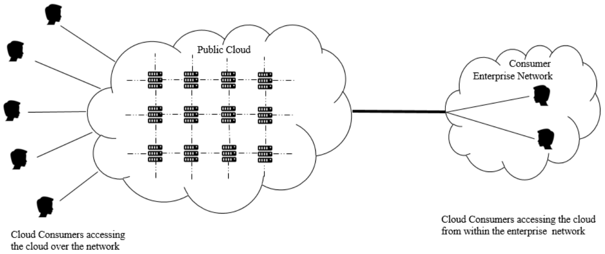
In this paper, a provenance model to increase the capability of digital forensics in a cloud environment will be examined. While this model can be experimented on other cloud deployment models, the public cloud architecture will be the focus as it introduces more challenges than the other models. The rest of the paper is organized as follows: In Section 2, basic concepts of public cloud, digital forensic and data provenance along with their challenges and requirements will be discussed. In Section 3, previous work on related areas will be explored. In Section 4, the proposed data provenance model will be discussed along with additional design considerations. Finally, conclusions and future work will be discussed in Section 5.

2. BACKGROUND

2.1 Public Cloud

(Mell, Grance, et al., 2011) defined Cloud computing as a "model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction." With lots of services offered through advanced technologies, cloud computing is redefining computing technology. Its "pay-as-you-go" and "provision-as-you-go" make it the most suitable choice for storing personal information, maintaining shared documents with a large group of users, and hosting large applications that require higher computational power. As cloud computing offers wide ranges of services based on demand from a single user to a large organization, different service and deployment models have been designed. One of the widely used deployment models of cloud service is public cloud, which (Jansen & Grance, 2011) describe as an "infrastructure and computational resource" that is "owned and operated by an outside party that delivers to the general public via multi-tenant platform". Figure-1 depicts a general architecture of the public cloud model where cloud consumers are accessing the infrastructure of a cloud provider.

Figure 1. Public Cloud Model (Bohn et al., 2011)


Public Cloud
Consumer Enterprise Network
Cloud Consumers accessing the cloud over the network
Cloud Consumers accessing the cloud from within the enterprise network

2.1.1 Challenges

As its definition suggests, the public cloud model is actually designed to offer individuals and enterprises the opportunity to minimize cost with on-demand infrastructure and computation support in a shared environment. However, the shift of control over the data and application to a different administrative entity raises some security and privacy concerns. A cloud provider's reliable and powerful infrastructure is often vulnerable to threats like media failures, malicious attacks or software bugs. Their ability to access a user's data with malicious intention is another type of privacy issue found in the public cloud model. (Ren, Wang, & Wang, 2012) identified data owners' inability to monitor and define the access control policy and lack of control over the record relevant to cloud resource consumption as possible security challenges.

(Bohn et al., 2011) and (Ren et al., 2012) considered hardware virtualization as another critical privacy and security issues. Hardware virtualization leads to a multitenant architecture where the physical infrastructures are shared among multiple consumers with logically separated control over the resources. An attacker might be able to overcome this logical separation using configuration errors or software bugs to gain illegitimate access. Accessing services over the internet also increases vulnerabilities through various network threats (Bohn et al., 2011).

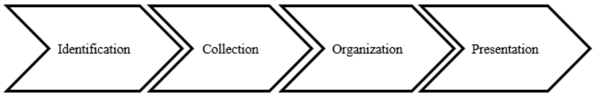
When an organization manages its own resources and records in its secured computing environment, it can clearly arrange the necessary protective measures and have a detailed idea about the location and structure of the data. On the contrary, cloud computing services are dispersed and data is duplicated over multiple locations to ensure availability of the records. Cloud providers maintain this information solely to eliminate possible security breaches. However, the inability to locate the actual position of the data is a major compliancy concern for an organization while exporting its business control over to the cloud (Kandukuri, Rakshit, et al., 2009).

2.2 Digital Forensics

With the expanding use of technological devices and widespread adoption of communication networks to share data, it has become necessary to develop technologies that can store and examine records from different sources. Digital forensics is a logical approach to identifying, collecting, and examining data while ensuring the integrity and chain of custody is properly preserved and maintained. (Kent, Chevalier, Grance, & Dang, 2006) describe several applications of digital forensics based on a variety of data sources, such as investigating cybercrime and violations of internal policy, reconstructing security incidents, troubleshooting operational problems and recovering from system damage.

(Zawoad, Hasan, & Skjellum, 2015) provide an elaborate description of the steps of the digital forensic process based on the formal definition. Figure-2 demonstrates the digital forensic process flow. The identification process consists of two steps where the incident and most relevant evidence are identified along with their possible correlation with other incidents in the system. In the collection step, all digital evidence related to an incident is accumulated from different sources, preserving their integrity. Later, the records are properly organized by extracting and investigating the data characteristics. In the final stage, a well formulated document is prepared by the investigator to present to the court.

Figure 2. Process Flow of Digital Forensic (Zawoad et al., 2015)



2.2.1 Challenges

Applying the digital forensic process in a simple computing architecture is complex by nature in and of itself; however, when a cloud architecture is taken into consideration, the process becomes even more challenging. The distributed nature of cloud infrastructure and some of its features affect the credibility of the collected artifacts and eventually increase the complexity of the forensic process in each of its steps.

(O'shaughnessy & Keane, 2013) illustrated some of the issues relevant to the cloud forensic process while describing the difference from the general approach of digital forensics based on the "Integrated Digital Investigation Process Model" of (Carrier, Spafford, et al., 2003). Carrier illustrated the relevance between digital and physical investigation and described the digital investigation model with five phases – preservation, survey, search and collection, reconstruction and presentation. However, (Alqahtany, Clarke, Furnell, & Reich, 2015) identified the challenges based on the general process model of forensics – identification, collection, organization and presentation. Forensic challenges in cloud computing in this paper will be described referencing both of these models.

Any digital investigation must begin with the collection of the digital device to preserve the artifact related to an occurrence. In cloud computing, volatile memory of a virtual machine instance and the unavailability of the virtual images demands creation of advanced technologies to preserve evidence. Logs, the valuable container of evidence, are maintained by the cloud providers, but integrity of this data might be affected by unlawful actions. A cloud environment also imposes restrictions over the control of data and relational information (e.g., location of data) which reduces the opportunity of effective data preservation.

Once the necessary evidence is preserved and ready for initial analysis to build a satisfactory theory, the incident identification process can begin. (O'shaughnessy & Keane, 2013) described the effect of cloud service models in this phase. In both Software as a Service (SaaS) and Platform as a Service (PaaS), the consumer has limited access to the cloud infrastructure, which leaves the investigator with a limited option to identify any occurrence directly in the server instance. Infrastructure as a Service (IaaS) provides better access than the other two models as it allows the client to configure their server with necessary logs. The different level of access in the different service models eventually hinders the process of building a unified model for incident identification and documentation.

Data collection involves accumulating data from different sources like storage devices, client's browser history, client's communication with the service provider and access information of the cloud and other network level data. Distributed cloud architectures introduce various challenges for the investigator in this phase. They require extensive knowledge of the cloud architecture, data storage techniques, extraction mechanisms for preserving data integrity and maintaining privacy of cloud consumers. Also, shared storage for managing the logging information for multiple enterprises raise concern for privacy concern if any incident needs thorough investigation and require collecting the complete logging information. Data collection also might be affected by integration of inefficient provenance techniques, lack of time synchronization and inappropriate maintenance of the chain of custody.

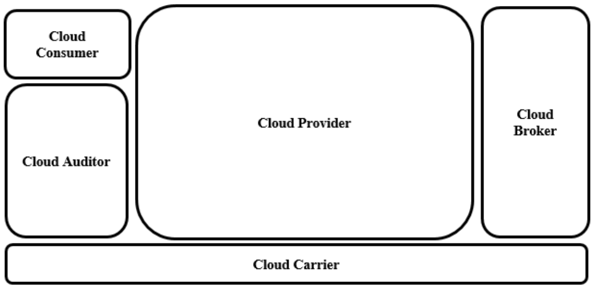
Collected records are further used in a controlled environment to reconstruct an incident to solidify theory building in the reconstruction phase. This requires arranging records from different sources in a unified structure, identifying correlations between different events to successfully recreate the incident. Unavailability of proper forensic tools and utility applications in a cloud environment introduces critical challenges in this phase.

A successful forensic investigation ends with the submission of a well-formulated documentation of an incident in the court with proper digital and physical artifacts to defend the report. The distributed nature of the cloud might require a detailed explanation for proper understanding by the jury, location of data might introduce difficulty to determine the judicial boundary, and sources and the data collection process might be questioned for the lack of credibility; hence, all these issues need to be overcome during this phase.

2.2.2 Requirements

(Bohn et al., 2011), in their conceptual reference model of cloud architecture, mentioned about five major cloud actors with their defined responsibilities and offered services. Figure 3 illustrates a cloud architecture focusing on the cloud actors, based on the high-level architecture shown in (Bohn et al., 2011). Each actor has sole or shared responsibilities to execute different business transactions or operations in a cloud environment. In this scenario, cloud providers offer services which are negotiated by the brokers, connected and transported by the carrier, used by the cloud consumers and evaluated independently for performance and security by auditors.

Figure 3. Conceptual Architecture of Cloud



The contribution of these actors is later used by (Ruan & Carthy, 2012) when discussing the different types of digital investigations. While an investigator is responsible for the external investigations, cloud actors play a vital role in the internal investigations by taking various measures for security into consideration. Based on the (Cloud Security Alliance, 2011) defined by Cloud Security Alliance, (Ruan & Carthy, 2012) presented a list of responsibilities of different cloud actors in digital forensics. The responsibilities are identified as required criteria of cloud forensics.

Ensuring data ownership with stewardship, data retention and disposal, data retention and disposal, facility security, clock synchronization, and integrating audit log and intrusion detection are identified as the sole responsibilities of the cloud provider [23]. Data ownership with stewardship helps in determining ownership of the records and maintaining a chain in custody in the forensic investigation. With a proper backup facility and redundant data storage policy, the provider can ensure successful data retention. Strict data disposal operations also need to be integrated to ensure complete removal of data from all the sources without leaving any option to recover them. Backups are the only possible means of reconstructing any event or incident.

Cloud providers also need to ensure there are controlled access and authorization checks for any entry to the facility and during data relocation to a different facility. In digital forensics, identifying events in proper order helps in incident reconstruction. Cloud providers need to ensure that all the processes are running under a synchronized environment. Integrating different logging functionality to record every single operation is another important responsibility of cloud providers, which undoubtedly plays the most vital role in digital forensic.

(Ruan & Carthy, 2012) also listed some shared responsibilities of providers and consumers based on the (Cloud Security Alliance, 2011). Defining meaningful taxonomy, depending on data type, origin, legal or contractual constraints and sensitivity, helps in forensic investigations. All the important assets need to be recorded with ownership information, and different data-related operations need to be logged with common forensic related terms. For the purpose of facilitating regulatory, statutory and contractual requirements for compliance mapping, elements might be assigned with legislative domains and jurisdiction. Other shared responsibilities of the provider and consumer include authorization, multi-factor authentication, establishment of policy and procedure as a part of incident management, and preservation of data and evidence integrity.

2.3 Data Provenance

Data provenance has proved its applicability and necessity in different application domains, data processing systems and representation models. For e-science, it can be used for transformation and easy derivation of data. In warehousing, it can be used to analyze and represent data. For Service Oriented Architectures (SOA), it can be used to execute complex computations. In document management and software development tools, it can be used to identify the lifecycle of a document. As the nature of the large range of applications differs, so does their technique of managing the provenance records.

Data provenance was classified in several categories based on the techniques used to trace the records and the way these are preserved. The techniques for data tracing are classified under two approaches: “lazy” and “eager” (W. C. Tan, 2004). In the “lazy” approach, provenance records are computed only when they are required, while provenance records are carried with the data in the “eager” approach. Sequence-of-Delta and Timestamping are two alternative approaches of archiving provenance records (W. C. Tan, 2004). The Sequence-of-Delta approach stores the reference version and a sequence of forward or backward deltas between versions. On the other hand, timestamping uses versions and times to identify data at various steps of its lifetime.

Provenance can also be discussed from the points of the affect of the source data on the existence of data records, or the locations from which those records are fetched (Buneman, Khanna, & Wang-Chiew, 2001). These concepts are termed as why-provenance and where-provenance, respectively. From the aspect of data processing architecture, another classification was proposed by (Simmhan, Plale, & Gannon, 2005). This approach was later adopted by (Muniswamy-Reddy, Holland, Braun, & Seltzer, 2006) and elaborated with more categories. They extend the database-oriented approach of (Simmhan et al., 2005) to include a file and file-system oriented approach called Provenance Aware Storage System (PASS). A grid-based solution comprises the service-oriented architecture in which software-development tools are part of a scripting-architecture. Environment architecture was the fourth category included by (Muniswamy-Reddy et al., 2006).

In a domain-specific approach of storing provenance, data and provenance are loosely coupled, as data is managed by the file system and provenance is stored in database systems (Muniswamy-Reddy et al., 2006). Lineage File System (LinFS) used a third party database to store provenance records while able to track it at file system level automatically (Sar & Cao, 2005). The obvious issue with file and provenance separation was addressed in Flexible Image Transport System by (Wells, Greisen, & Harten, 1981) where the file header contains the additional attributes as a provenance record. Another approach for tight coupling between data and provenance was later introduced in PASS. PASS tracks the provenance automatically like LinFS and manages its database directly into the kernel by its file system PASTA (Muniswamy-Reddy et al., 2006). PASS proved its superiority by ensuring better synchronization between data and provenance; it also provides security over the provenance and features to query those records. File Provenance System (FiPS) is another provenance system that collects provenance records automatically operating at the system level and below the Virtual File System (VFS) layer (Sultana & Bertino, 2013).

2.3.1 Challenges

Like digital forensics, the dynamic nature of cloud computing and its architecture introduces several challenges to establish a successful provenance technique. Researchers have highlighted heterogeneous architecture of cloud infrastructure, granularity , virtualization, availability and data integrity as the prime challenges while identifying a better provenance approach in the cloud (Muniswamy-Reddy, Macko, & Seltzer, 2010; Sakka, Defude, & Tellez, 2010; Zhang, Kirchberg, Ko, & Lee, 2011; Katilu, Franqueira, & Angelopoulou, 2015; Abbadi, Lyle, et al., 2011).

(Muniswamy-Reddy et al., 2010) discovered the lack of extensibility of existing provenance systems in the area of cloud computing and also their inability to support availability and scalability as some of the critical areas. Incompatible availability features between the provenance system and the cloud architecture may affect the principle goal of provenance architecture. In addition, lack of scalability becomes clearly visible when a database is used to store provenance in the cloud architecture. This feature makes provenance records query-able, but in a distributed architecture, it introduces a deadlock or scalability bottleneck. Introduction of a parallel distributed database, a possible solution to this problem, is often criticized because of its lack of maintainability and cost.

(Sakka et al., 2010) highlighted challenging areas arise due to the variety in cloud architecture. Difference in policies applied by these architectures to identify objects weaken the ability to establish a unique object identification technique in cloud. Also, there is no efficient and unified architecture for an inter-operable provenance system which could manage the heterogeneous policies followed by different entity and multiple level of granularity.

The virtualization feature of cloud architecture and its nature of fault tolerance also imposes some technical issues in provenance collection. (Zhang et al., 2011) discussed the necessity of maintaining a record of operating systems, server locations and memory management of virtual and physical environments to provide more accurate information to the consumers. They also prioritized the maintenance of provenance information about data migration which takes place in case of hardware failure.

(Abbadi et al., 2011) presented taxonomy of cloud infrastructures with a description of the dynamic nature of this structure and identified the challenges in the area of logging and auditing. It is imperative for a system administrator to identify the location and reason of an error through auditing and logging. But cloud's multilayer architecture makes it difficult to identify the origin of data from humongous resources collected from a large number of diversified sources. (Abbadi et al., 2011) defined the role of the administrator in this scenario as identifying the time intervals when physical layers and virtual layers are used by the virtual resources and applications respectively, and combining these records with other relevant log files. They also spotted some shortcomings like lack of protection in maintaining logging and auditing records and lack of trustworthiness of the provider maintaining the cloud infrastructure.

2.3.2 Requirements

In this section, requirements of data provenance will be discussed from two different aspects: the mandatory properties to fulfill the general criteria of a provenance record and the required features for a successful provenance technique in the cloud.

(Zhao, Bizer, Gil, Missier, & Sahoo, 2010) categorized the requirement of provenance based on concerning areas such as content, management and usage which are essential for an effective resource description framework (RDF) model. They presented an elaborated discussion over each criterion. These information include identity of the content, evaluation of the content over the time, records of processes or entities contributed in its evaluation including justification and design choices behind the evaluation process. From the management perspective of the provenance records, a well-formatted representation language and the accessibility of the provenance records are also considered as important attributes.

(Moreau et al., 2011) also identified a similar set of integral components in their Open Provenance Model (OPM). Basic information about an object, processes and agents involved in its evaluation, interdependencies between different entities, and actual role of the agents are the most important tokens of a provenance record. This information later needs to be recorded in an easily representable language supported by heterogeneous systems and should allow customization to include additional information. Additionally, the recorded information should be easily accessible with a simple query.

With the necessary attributes mentioned above, provenance system also needs to follow some standards in terms of data maintenance, dependency on the associated application, level of granularity and data integrity and presentation (Muniswamy-Reddy et al., 2010; Katilu et al., 2015; Taha, Chaisiri, & Ko, 2015; Y. S. Tan, Ko, & Holmes, 2013).

The data-coupling feature confirms the association between the data and provenance records. Appropriate data coupling cancels the chances of misleading data that might contain new provenance information for old data or vice-versa. This coupling must also include the causal ordering among different entities while recording the provenance information (Muniswamy-Reddy et al., 2010).

A provenance tracking system should enable tracking a particular object from every host in a distributed environment, which is an essential property to detect an incident and ensure accountability across a system. That way, if multiple host contribute in any operation on an object, provenance records should maintain record for each responsible hosts. A provenance system should also be independent of the application and platform so that no additional configuration is required to track if a new application is introduced in a system or if the kernel of a system is modified (W. C. Tan, 2004).

Provenance system should strictly maintain the integrity of the records by making them temper-evident and ensure persistence of the record. Data integrity assure the accuracy or correctness of the records over time while temper-evidence confirms reliability, authenticity, admissibility, believeability and completeness (Taha et al., 2015). On the other hand, persistence confirms the existence of provenance information even when the actual object is erased from the system as long as the object has a descendant. The nature of provenance information also demands confidentiality in every level of data management e.g., data access, data storage and data collection.

Capturing provenance records from multiple levels is also considered to be an essential property of a provenance system. Multilayer granularity provides complete scenario of any incident occurred with an object in different level of a system (Katilu et al., 2015). An advanced interface to represent the provenance record along with an efficient data retrieval technique is another measure that defines the success of a provenance system. This sort of interface with efficient query mechanism helps identifying any occurrence quickly and understand patterns and characteristics of provenance record (Katilu et al., 2015).

3. RELATED WORKS

Digital provenance, in the cloud environment, has been studied for a long time in order to meet the critical requirements imposed by the architecture and to mitigate the challenges revolved around digital forensics. Some researchers proposed different provenance techniques, while others have identified efficient approaches for effective forensic solutions (Muniswamy-Reddy et al., 2010; Zhang et al., 2011). Several techniques were also proposed where provenance architecture is specially discussed considering the requirements of digital forensic (Li, Chen, Huang, & Wong, 2014; P. M. Trenwith & Venter, 2014; Lu, Lin, Liang, & Shen, 2010).

(Muniswamy-Reddy et al., 2010) extended PASS for cloud architecture which was initially used in local file systems and network attached storage. They have implemented three provenance recording protocols using Amazon Web Service (AWS) - single cloud storage, cloud storage with cloud database, and cloud storage with cloud database and messaging service based on PASS. Cloud messaging service in the third architecture ensured its superiority over the other two by confirming provenance datacoupling through transaction and providing the fastest service.

Cloud providers usually track every operation executed inside a cloud environment to ensure immediate response to any incident. Flogger is such a logging mechanism which can track file provenance both in virtual and physical machines (Ko, Jagadpramana, & Lee, 2011). DataPROVE, a data provenance technique was proposed based on records generated by Flogger in collaboration with other system monitoring tools like User, Process, Event Tracker, Change Tracker and Network Traffic Tracker (Zhang et al., 2011). It collects provenance information in the application layer service like PaaS, SaaS and IaaS, rather than just virtual and physical machine information. It also captures other cloud related essential provenance information like client identification record and transfer of information between different virtual and physical machines placed in different locations. Additionally, DataPROVE also provides efficient data API for indexing and query operations.

S2Logger is another approach of maintaining secured data provenance through logging at the block-level and file-level. It is implemented based on Flogger (Suen, Ko, Tan, Jagadpramana, & Lee, 2013). In this mechanism, (Suen et al., 2013) addressed features like capture mapping between virtual and physical resources, proper maintenance of the provenance chain and data integrity and confidentiality. S2Logger also introduced an end-to-end data tracing mechanism that is capable of capturing detail provenance information from all nodes connected in the cloud environment.

(Lu et al., 2010) have designed a provenance mechanism enabling the forensic features in the cloud environment. Provenance was designed based on a bilinear pairing technique considering some critical aspects of cloud like data confidentiality, anonymous access of the user, and dispute resolution. This mechanism introduced computation and communication overhead at the data owner's end. This issue was later identified by (Li et al., 2014), and they have introduced a broadcast encryption method to overcome this. They have also ensured anonymity of the user by implementing an attribute-based signature (ABS) scheme for access control and privacy by designing an anonymous key-issuing protocol.

(P. M. Trenwith & Venter, 2014) discussed another source of log information which may allow the investigator to answer to the physical location of the data. They adopted a provenance technique in their proposed system where transport layer protocol information was recorded to identify the location of data. They also considered the application layer data to answer the additional queries like who, when, what and how information which are required for a successful forensic investigation. Provenance record is proposed to be stored in a centralized logging server which eventually can overcome the problem of strong coupling between data and its meta data and can also ensure data integrity. They proposed another model for provenance with features like digital object tracking and location identification at any point of the lifetime of that particular digital object (P. Trenwith & Venter, 2015). In this model, they introduced the concept of a wrapper object which wrapped a file along with its provenance records embedded together and the location of the wrapper object would be maintained in a separate centralized server.

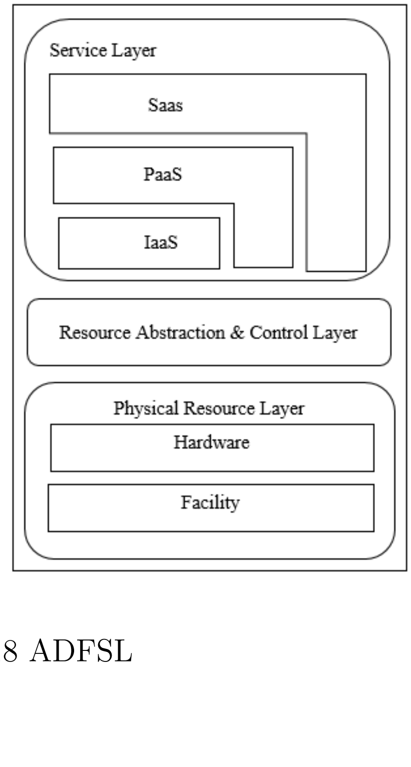
4. FORENSIC ENABLED PROVENANCE MODEL

4.1 Additional Design Consideration

Designing an efficient and forensic enabled provenance mechanism in a cloud environment requires that some other areas be considered which vastly contribute to the success of the process. Identifying the possible sources of provenance records, deciding on a unified model to store those records, data storage with required features for efficient data management, and ensuring security and integrity of the data for forensic operations are important factors that must be considered for a provenance mechanism.

General cloud architecture can be divided into several layers – physical resource layer, resource abstraction layer and service layer as shown in Figure 4. (Ruan & Carthy, 2012) identified the sources of forensic artifacts from each of the layers. Among these, it is necessary to identify the actual sources which provide necessary provenance related information and help in forensic analysis. Hard disks, network logs and access records are important sources from the physical layer. Event logs and virtual resources are necessary from the abstraction layer, and access logs, events logs, rigid information about the virtual operating system, and application logs are vital sources from the service layer. (Imran & Hlavacs, 2013) identified provenance information in different sublayers of the service layer and developed an integrated provenance data collection process. In our conceptual model approach of collecting provenance data, we will also consider the featured criteria used in these proposed systems which can be helpful for the forensic process.

Figure 4. Cloud System Architecture ([(Bohn et al., 2011)])



The format and properties for logging provenance data collected from different sources is important in a provenance collection system. (Marty, 2011) has identified some important properties of provenance data like timestamps, application, user, session id, category, and reason and severity of the event. Applications running in the service layer mostly contribute to this type of information which can be collected from different application and event logs. A distributed file system, like Google File System (GFS), maintains a separate set of information vital for provenance and forensics. The GFS master usually keeps information file and chunk (part of a file) namespace, mapping from file to chunks, and location of the replica of the chunks (Ghemawat, Gobioff, & Leung, 2003). The Hadoop Distributed File System (HDFS) also maintains a transaction log, called EditLog, in its NameNode which contains every modification recorded in file system metadata (Borthakur et al., 2008). Google cloud maintains region and zone related information while creating a virtual machine instance recording which has provenance data along with other information that may guide an investigator in forensic analysis. The heterogeneous characteristics of these records make the process of choosing a particular format very hard. Marty has suggested a syntax of logging the information as "key-value" pair, which is considered as an ideal syntax of logging information in our design approach. We also include a unified naming approach of the "key" to ensure synchronicity of the same properties coming from different from sources.

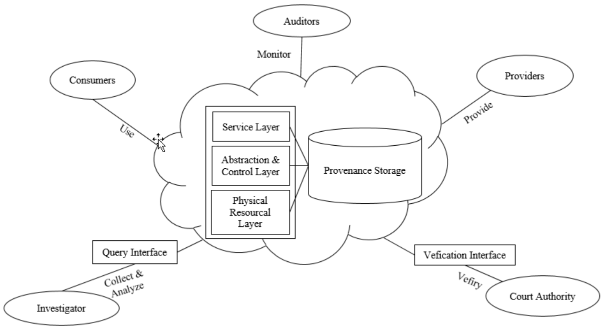
Provenance data records must be easily searchable; therefore, a data repository that can provide the best performance and an efficient query interface must be chosen. (Muniswamy-Reddy, 2006) explored different database architectures to choose the best data repository. They considered aspects like the format of maintaining dependency of the files and different approaches of storing the annotation metadata. They compared Berkley DB as a schemaless database, MySQL as a RDBMS database, XMLDB as XML database and OpenLDAP as a representative of an LDAP architecture and found that Berkley DB outperformed other database architectures in terms of standard database related operations in this scenario. While implementing PASS in cloud, (Muniswamy-Reddy et al., 2010) also used SimpleDB and Amazon S3 to store provenance objects.

Provenance records logged in different layers are basically maintained by different actors of the cloud. Cloud consumers are usually responsible for the records maintenance service layer while providers manage the logging activities of the abstraction and physical layers. Provenance records in these layers may often be affected by the collusion between the consumer and provider which eventually can mislead the forensic investigator. The investigator can also play the role of an accomplice with either of these parties, which ultimately diminishes the whole approach of data provenance and forensic analysis. To ensure integrity of the provenance data and trustworthiness of the collection system, it is necessary to include a verification process which can be used to validate the provenance information. (Zawoad et al., 2015) proposed a Proof Publisher Method (PPM) in their Open Cloud Forensic (OCF) which can be used by the court authority to verify the forensic reports provided by the investigators. They also proposed a reliable way of collecting the evidence through secured read-only APIs accessible only by the investigator and court authority.

4.2 Proposed Model

Figure 5 shows the proposed Forensic Enabled Cloud Provenance Model. This section will provide a description of the proposed model and surveys its requirements of provenance, forensic analysis, and necessary design considerations. The concept of the general actors of cloud operations and forensic analysis have been utilized in this model; however, the contribution of each of the cloud actors and access mechanisms to the provenance records for the investigator and court authority have been redefined. It is also necessary to introduce a unified logging mechanism in different layers of a cloud environment to leverage data manipulation while recording provenance information in a persistent database.

Figure 5. Forensic Enabled Cloud Provenance Model



In a cloud model, consumers and providers are the two main contributors to the cloud operation, while a cloud auditor basically performs analysis on the cloud services and provides feedback. In the proposed model, an additional responsibility for the cloud auditors is included. Usually cloud services like SaaS and PaaS are designed to provide the consumer with less control over the system while IaaS allows configuring the necessary applications. In these scenarios, a cloud auditor's additional responsibility will be monitoring whether the services used by the consumers are capable of logging every operation or whether a provider enables the logging features for each application where the consumer has limited access. The providers additional responsibility will be introducing policies in SLA which will clearly instruct the clients about installing necessary auditing features. The provider should also redesign the access policies so that the client can have enough options for customization. The provider should also be capable of monitoring every operation performed by the client. In an ideal scenario, the provider should install firewalls and other applications to detect intrusions or any form of malicious activity.

Database architecture is an important aspect of designing an ideal provenance model. It is required not only to accommodate a variety of information or manage large volumes of data, but also to ensure availability and scalability. Considering these facts, the proposed system is a schema-less distributed database system to store the provenance record coming in heterogeneous format from different sources. The distributed nature of the database architecture will ensure the availability and scalability features.

Data will be stored in a key-value format where the key name will maintain a consistent naming convention to facilitate the query operation maintained through a unified data modeling policy. Each transaction will be logged separately with the reference to the dependent object. Keeping a single instance of the provenance record requires updating the instance very frequently and hence reduces the expected performance of the system. In the case of recording provenance for a file, it is recommended that a copy of the content that will be logged in the repository be maintained. Provenance record of a data should also maintain relationship information with the process or environment variables that effected it at any point of its lifetime.

A timestamp with every transaction is a vital field. Loggers in different layers will require maintaining strict synchronization of time to capture the sequence of events properly. To maintain proper chronology of the data from different systems, each transaction will be saved with timestamp information for both the source and data storage system. This will eventually help in identifying the unsynchronized nodes or sources.

Trustworthy data storage policy is another important concern. It is necessary to ensure integrity of the data and employ techniques to easily identify any forged data and possible miscreants. To ensure this, the database should be designed for capturing necessary audit information. Audit information will include the application or user identity which eventually stores or update any information. For all of the data that will be saved an additional attribute will be added as provenance. This attribute will contain the hash value of the data item. This hashed data will be used later for verification of the information.

If any issue is reported by any of the other three actors of cloud services, the process of forensic analysis starts. In this case, an investigator will conduct the preliminary investigation by collecting provenance records from the provenance storage and finally prepare a report eligible for submission to the court authority. The court authority will verify the report and provide the verdict. To facilitate this process, query and verification interfaces for the investigator and court authority have been introduced in the proposed system. These services will be provided by the service providers.

4.3 Discussion

This section briefly demonstrates how the proposed model of provenance collection overcomes the challenges already discussed in Section 2 and how efficiently this model can facilitate the process of forensic analysis. In table-1, we also present a comparative analysis with the existing approaches in terms of the challenges of provenance management for forensic analysis in public cloud addressed by those models.

One of the primary concerns about cloud computing is the shift of control of data or files to a different authority. Data might be affected by malicious attacks, software bugs or system failure. The distributed nature of cloud architecture can ensure availability of data in these scenarios. Any unlawful access from a provider's end can be easily identified through the provenance information, as every access to any file or data is recorded by the provenance system.

Allowing multi-tenant facility through virtualization and identifying the location is the most notable challenge in a public cloud environment. Collecting snapshots of virtual machine logs with additional information to identify the virtual machine instance can solve this problem. File system metadata contains details about the file status and location. Integrating necessary measures already discussed in 4.1 can lead to an efficient solution to these issues. Maintaining provenance records with location information will eventually help investigators identifying the location of an object and collect necessary artifacts accordingly.

Forensic investigation is often hampered because of the lack of synchronization between stored records and variations of records coming from different sources. In the proposed model, a unified model to identify attributes of the entity and relevant operations is introduced. This will facilitate the process of analysis as investigators can easily identify and differentiate between entities of a particular piece of information and their effect on the provenance record or forensic analysis.

Managing the large collection of provenance data is a difficult job. The large volume of information, variety in data formats and increase in volume in high velocity make the provenance collection process a highly challenging task. To ensure efficient forensic analysis, a provenance collection technique can compromise none of these concerns. Hence, it is recommended to ensure a better data management policy. The proposed model suggests a unified naming convention for "keys" coming from different sources. A well-chosen naming conversion helps to identify the exact information in search and can also identify relations in log information coming from different sources.

Regeneration of data or an incident from the lineage information is another focus of a data provenance system. Provenance records maintained with the actual content can ease this process as an investigator can easily extract any verified content from the provenance storage and apply the operations which occur in the other version of the data to check if these operations generate the same result. This forensic enabled data provenance model also considers the storing of the data along with its metadata in the provenance log.

This proposed provenance model collects information from different layers of a cloud architecture. It also considers necessary measures to include location information and other attributes required to identify the related entities or actions of data. This large volume of information opens enough opportunity for proper analysis.

A loose coupling between data and provenance records is maintained as provenance information is stored separately in a database. Separation between these two records ensure the existence of provenance information even if the actual data is missing. By nature, provenance information is immutable and persistent. If any provenance is deleted from the database intentionally, a database audit can provide necessary hints to identify the miscreants.

The proposed model also prioritizes the

		Proposed Approaches					
		(Muniswamy-Reddy et al., 2010)	(Zhang et al., 2011)	(Li et al., 2014)	(P. M. Trenwith & Venter, 2014)	(Lu et al., 2010)	Our Approach
Public Cloud	Virtualization		✓				✓
	Security & Privacy	✓	✓	✓		✓	✓
	Data Location				✓		✓
Digital Forensic	Data Preservation	✓		✓	✓		✓
	Identification	✓		✓	✓	✓	✓
	Collection		✓		✓	✓	✓
	Scenario Reconstruction	✓		✓	✓	✓	✓
	Presentation						✓
Provenance	Data Coupling	✓				✓	✓
	Cross-Host Tracking					✓	
	Application Independence			✓	✓		
	Multilayer Granularity		✓		✓		✓
	Data Integrity	✓	✓	✓	✓		✓
	Confidentiality		✓		✓	✓	✓
	Query-able	✓	✓				✓
	Persistence	✓	✓				
	Causal Ordering	✓	✓				✓

Table 1. Comparative analysis of the proposed approaches

necessity of a query and verification interface for the investigator and court authority. A well-designed interface with features of executing customize requests to extract required information from the provenance database can prove the efficiency of a provenance model. Effective data visualization is another important aspect of these interfaces. A detailed overview of an event in chronological order, providing information about a file with location and list of processes that contributed in evolution of the file can help an investigator define the legal boundary while requesting for actual artifacts from the providers without intervening other users' information and identifying the actual lineage of the file. A step by step verification can also help a jury or court authority to detect the possible symptoms of collusion between the consumer and cloud service provider, consumer and investigator, or provider and investigator.

5. CONCLUSION

Maintaining the provenance information of data is as important as the data itself. In modern day computing technology, it is an integral part of data to ensure its authenticity and integrity. In this paper, we have identified the challenges of cloud architecture, discussed how this affects the existing forensic analysis and provenance techniques, and discussed a model for efficient provenance collection and forensic analysis. We have also briefly highlighted the necessity of provenance data visualization and explained why or how it can be helpful for modern day forensic analysis.

REFERENCES

Abbadi, I. M., Lyle, J., et al. (2011). Challenges for provenance in cloud computing. In *Tapp.*

Alqahtany, S., Clarke, N., Furnell, S., & Reich, C. (2015). Cloud forensics: a review of challenges, solutions and open problems. In *Cloud computing (iccc), 2015 international conference on* (pp. 1–9).

Amazon Web Services. (2008, September). *Building greptheweb in the cloud, part 1: Cloud architectures.* (http://developer.amazonwebservices.com/ connect/ entry.jspa?externalID=1632 [September 30 2016])

Bohn, R. B., Messina, J., Liu, F., Tong, J., & Mao, J. (2011). Nist cloud computing reference architecture. In *Services (services), 2011 ieee world congress on* (pp. 594–596).

Borthakur, D., et al. (2008). Hdfs architecture guide. *Hadoop Apache Project*, *53*.

Buneman, P., Khanna, S., & Wang-Chiew, T. (2001). Why and where: A characterization of data provenance. In *International conference on database theory* (pp. 316–330).

Carrier, B., Spafford, E. H., et al. (2003). Getting physical with the digital investigation process. *International Journal of digital evidence*, *2*(2), 1–20.

Chung, M., & Hermans, J. (2010). From hype to future: Kpmg's 2010 cloud computing survey. *KPMG*, 8–28.

Cloud Security Alliance. (2011, September). *Cloud controls matrix v1.2 - cloud controls matrix working group.* (https://cloudsecurityalliance.org/ download/ cloud-controls-matrix-v1-2/)

Ghemawat, S., Gobioff, H., & Leung, S.-T. (2003). *The google file system* (Vol. 37) (No. 5). ACM.

Hashizume, K., Rosado, D. G., Fernández-Medina, E., & Fernandez, E. B. (2013). An analysis of security issues for cloud computing. *Journal of internet services and applications*, *4*(1), 5.

Imran, M., & Hlavacs, H. (2013). Layering of the provenance data for cloud computing. In *International conference on grid and pervasive computing* (pp. 48–58).

Jansen, W., & Grance, T. (2011). Sp 800-144. guidelines on security and privacy in public cloud computing.

Kandukuri, B. R., Rakshit, A., et al. (2009). Cloud security issues. In *Services computing, 2009. scc'09. ieee international conference on* (pp. 517–520).

Katilu, V. M., Franqueira, V. N., &

Angelopoulou, O. (2015). Challenges of data provenance for cloud forensic investigations. In *Availability, reliability and security (ares), 2015 10th international conference on* (pp. 312–317).

Kent, K., Chevalier, S., Grance, T., & Dang, H. (2006). Guide to integrating forensic techniques into incident response. *NIST Special Publication*, *10*, 800–86.

Ko, R. K., Jagadpramana, P., & Lee, B. S. (2011). Flogger: A file-centric logger for monitoring file access and transfers within cloud computing environments. In *Trust, security and privacy in computing and communications (trustcom), 2011 ieee 10th international conference on* (pp. 765–771).

Li, J., Chen, X., Huang, Q., & Wong, D. S. (2014). Digital provenance: Enabling secure data forensics in cloud computing. *Future Generation Computer Systems*, *37*, 259–266.

Lu, R., Lin, X., Liang, X., & Shen, X. S. (2010). Secure provenance: the essential of bread and butter of data forensics in cloud computing. In *Proceedings of the 5th acm symposium on information, computer and communications security* (pp. 282–292).

Market Research Media. (2016, September). *Global cloud computing market forecast 2015-2020.* (https://www.marketresearchmedia.com/?p=839)

Marty, R. (2011). Cloud application logging for forensics. In *Proceedings of the 2011 acm symposium on applied computing* (pp. 178–184).

Mell, P., Grance, T., et al. (2011). The nist definition of cloud computing.

Moreau, L., Clifford, B., Freire, J., Futrelle, J., Gil, Y., Groth, P., … others (2011). The open provenance model core specification (v1. 1). *Future generation computer systems*, *27*(6), 743–756.

Muniswamy-Reddy, K.-K. (2006). Deciding how to store provenance.

Muniswamy-Reddy, K.-K., Holland, D. A., Braun, U., & Seltzer, M. I. (2006). Provenance-aware storage systems. In *Usenix annual technical conference, general track* (pp. 43–56).

Muniswamy-Reddy, K.-K., Macko, P., & Seltzer, M. I. (2010). Provenance for the cloud. In *Fast* (Vol. 10, pp. 15–14).

O'shaughnessy, S., & Keane, A. (2013). Impact of cloud computing on digital forensic investigations. In *Ifip international conference on digital forensics* (pp. 291–303).

Ren, K., Wang, C., & Wang, Q. (2012). Security challenges for the public cloud. *IEEE Internet Computing*, *16*(1), 69–73.

Ruan, K., & Carthy, J. (2012). Cloud computing reference architecture and its forensic implications: A preliminary analysis. In *International conference on digital forensics and cyber crime* (pp. 1–21).

Sakka, M. A., Defude, B., & Tellez, J. (2010). Document provenance in the cloud: constraints and challenges. In *Meeting of the european network of universities and companies in information and communication engineering* (pp. 107–117).

Sar, C., & Cao, P. (2005). Lineage file system. *Online at http://crypto.stanford.edu/ cao/ lineage. html*, 411–414.

Simmhan, Y. L., Plale, B., & Gannon, D. (2005). A survey of data provenance in e-science. *ACM Sigmod Record*, *34* (3), 31–36.

Suen, C. H., Ko, R. K., Tan, Y. S., Jagadpramana, P., & Lee, B. S. (2013). S2logger: End-to-end data tracking mechanism for cloud data provenance. In *Trust, security and privacy in computing and communications (trustcom), 2013 12th ieee international conference on* (pp. 594–602).

Sultana, S., & Bertino, E. (2013). A file provenance system. In *Proceedings of the third acm conference on data and application security and privacy* (pp. 153–156).

Taha, M. M. B., Chaisiri, S., & Ko, R. K. (2015). Trusted tamper-evident data provenance. In *Trustcom/bigdatase/ispa, 2015 ieee* (Vol. 1, pp. 646–653).

Tan, W. C. (2004). Research problems in data provenance. *IEEE Data Eng. Bull.*, *27*(4), 45–52.

Tan, Y. S., Ko, R. K., & Holmes, G. (2013). Security and data accountability in distributed systems: A provenance survey. In *High performance computing and communications & 2013 ieee international conference on embedded and ubiquitous computing (hpcc_euc), 2013 ieee 10th international conference on* (pp. 1571–1578).

Trenwith, P., & Venter, H. (2015). Locating and tracking digital objects in the cloud. In *Ifip international conference on digital forensics* (pp. 287–301).

Trenwith, P. M., & Venter, H. S. (2014). A digital forensic model for providing better data provenance in the cloud. In *Information security for south africa (issa), 2014* (pp. 1–6).

Voorsluys, W., Broberg, J., & Buyya, R. (2011). Introduction to cloud computing. *Cloud computing: Principles and paradigms*, 1–41.

Wells, D., Greisen, E., & Harten, R. (1981). Fits-a flexible image transport system. *Astronomy and Astrophysics Supplement Series*, *44*, 363.

Zawoad, S., Hasan, R., & Skjellum, A. (2015). Ocf: an open cloud forensics model for reliable digital forensics. In *Cloud computing (cloud), 2015 ieee 8th international conference on* (pp. 437–444).

Zhang, O. Q., Kirchberg, M., Ko, R. K., & Lee, B. S. (2011). How to track your data: The case for cloud computing provenance. In *Cloud computing technology and science (cloudcom), 2011 ieee third international conference on* (pp. 446–453).

Zhao, J., Bizer, C., Gil, Y., Missier, P., & Sahoo, S. (2010). Provenance requirements for the next version of rdf. In *W3c workshop rdf next steps.*

Publication Information

The *Journal of Digital Forensics, Security and Law* (*JDFSL*) is a publication of the Association of Digital Forensics, Security and Law (ADFSL). The Journal is published on a non-profit basis. In the spirit of the *JDFSL* mission, individual subscriptions are discounted. However, we do encourage you to recommend the journal to your library for wider dissemination.

The Journal is published in electronic form under the following ISSN:

ISSN: 1558-7223

The Journal was previously published in print form under the following ISSN:

ISSN: 1558-7215

The office of the Association of Digital Forensics, Security and Law (ADFSL) is located at the following address:

Association of Digital Forensics, Security and Law
4350 Candlewood Lane
Ponce Inlet, Florida 32127
Tel: 804-402-9239
Fax: 804-680-3038
E-mail: office@adfsl.org
Website: http://www.adfsl.org