Evidence for Accountable Cloud Computing Services

Thomas Rübsamen[1], Christoph Reich[1], Aryan Taherimonfared[2], Tomasz Wlodarczyk[2], and Chunming Rong[2]

[1] Cloud Research Lab,
Furtwangen University of Applied Science,
D-78120 Furtwangen, Germany
{thomas.ruebsamen,christoph.reich}@hs-furtwangen.de
[2] Department of Electrical Engineering and Computer Science,
University of Stavanger,
N-4036 Stavanger, Norway
{aryan.taherimonfared,tomasz.w.wlodarczyk,chunming.rong}uis.no

Abstract. Evidence that allows assurance of accountability services, verification of compliance with the principles of accountability by service-providers and attribution of responsibility for breaches within the chain of accountability is essential. This paper defines how evidence may be required and proposes suitable ways of treating key accountability concepts. It shows the importance of verification and assurance, monitoring and auditing, and challanges of evidence in cloud computing. A discussion of logging and evicence gathering points complete the paper.

1 Introduction

Issues of transparency and control arise, when data moves from being stored locally to being stored remotely on the cloud. It becomes important to provision evidence for handling of confidential data in the Cloud by remote parties through whole lifecycle, also including deletion. However, this evidence is often not provided; transparency and verifiability are missing in the cloud context (especially at PaaS and IaaS levels). Moreover, there are additional related issues including cloud computing and globalization, increasing foreign government surveillance, the potential for light-touch self-regulation by the back door, weak certification for accountability, and weak links in terms of data protection along the service provision chain.

Currently, there is a lack of transparency and accountability from the provider side as for service provisioning/de-provisioning, tenant isolation, data processing and movement, privacy protection as well as many other aspects which used to be fully under the control and monitoring of the consumer. Even if key terms are being added into cloud contracts (Service Level Agreement), processes and techniques must be developed to continuously and automatically monitor and audit these terms and ensure adequate transparency. Cloud providers must be also prepared to provide adequate evidence about security and privacy provision.

A system for Evidence Collection that captures, integrates and processes the information including logs, policies and context in a way that preserves privacy and confidentiality and, supports audit and attribution is needed. An evidence framework for Cloud Computing does not exist yet. The main contribution of this paper is establishing necessary requirements for provisioning of evidence in a Cloud environment and how these requirements influence the tasks of monitoring and audit.

This paper is organized in the following way. In Section II we summarize existing related work. In this context, in Section III, we discuss general requirements necessary to provision evidence handling in a Cloud environment. In Section IV we discuss how these requirements influence the tasks of monitoring and audit. In Section V we summarize challenges of evidence provisioning in Cloud Computing. We conclude the paper in Section VI.

2 Related Work

One initiative towards evidence framework for Cloud Computing is an open architecture for digital evidence integration [1] by Schatz, B., and Clark, A. J. from the Common Digital Evidence Storage Format Working Group (CDESF). The architecture focused on digital evidence bags (DEB), a generalized method for collecting information about evidence and evidence metadata while keeping evidence integrity.

In Dykstra's paper [2] investigates how to obtain forensic evidence from cloud computing using the legal process by surveying the existing statues and recent cases applicable to cloud forensics. A sample search warrant is presented that could provide a sample language for agents and prosecutors who wish to obtain a warrant authorizing the search and seizure of data from cloud computing environments.

The paper from Haeberlen et al. [3], an accountable virtual machines (AVMs) has been introduced, which can execute binary software images in a virtualized copy of a computer system and can record non-repudiable information that allows auditors to subsequently check whether the software behaved as intended. Since this approach is basically VM logging and replaying, it is effectively the same as our full integrity checking, potentially with a lot of overhead.

In the paper of Poisel et al. [4] discuss digital forensics investigations at the hypervisor level of virtualized environments and introduce the topic of evidence correlation within cloud computing infrastructures.

The acquisition and analysis of digital evidence in cloud deployments is more complex, because data could be encrypted before being transferred to the cloud or it could be stored in different jurisdictions resulting in data being deleted before investigators have access to it [5].

Flaglien et al. [6] evaluated currently used storage and exchange formats for handling digital evidence against criteria identified in recent research literature. Formats intended for storing evidence from highly dynamic and complex sys-

tems are characterized by incorporating additional information, which can be processed by data mining tools.

Lu et al. [7] proposed to adopt the concept of provenance to the field of cloud computing by enabling a data object to report who created it and modified its contents, provenance could provide digital evidences for post investigations. Provenance information would have to be secured in cloud environments as leaking this information could breach information confidentiality and user privacy.

Marty's [8] approach utilize logging facilities to generate and collect relevant data to support the digital forensics investigation process.

The chain of custody documents how evidence was handled in the context of the digital investigations process [9]. The documentation describes how evidence was collected, analyzed, and preserved to be approved in court.

3 Accountability and Evidence

The A4Cloud FP7 research project [10] approach encompasses legal and regulatory mechanisms and a range of technological enhancements that can provide the necessary basis for trust. Customers, providers and regulators should be supported by preventive, detective, and corrective task (see [11]) and, for example, give cloud customers more control over their cloud services, ensure providers to meet their obligations, and enable cloud audits.

Technology can provide assistance in ensuring proper implementation of accountability. In particular, technology can be used to strengthen the enforcement and monitoring of policies and to help provide evidence, assurance and transparency. Hence, in accordance with Recommendation 5 from (Castelluccia et al, 2011 [12]), our approach is that privacy assessment, assurance, verification or enforcement should be evidence-based, and that these evidences might be derived from a number of sources, events and traces at different architectural layers.

The A4Cloud project identified a number of accountability attributes, like obligation, responsability, remediation, attributability, liability, sanctions, assurance, transparency, remediation, observability and responsiveness. These attributes have different importance from the perspective of a framework of evidence and identification of evidence types. We can divide these attributes into two general groups, those that reflect on accountability as a concept and those that reflect on how such concept should or could be implemented. Evidence of the following accountability attributes are of primary interest:

1. Attributability: Attributability describes a property of an observation that discloses or can be assigned to actions of a particular actor (or system element).
2. Observability: Observability is a property of an object, process or system that describes how well the internal actions of the system can be described by observing the external outputs of the system.
3. Assurance can take the form of evidence. An accountability system can produce evidence that can be used to convince a third party that a fault has

or has not occurred. In the context of accountability, assurance could refer to provision of ex ante evidence for compliance to governing rules, and possibly also to evidence that the governing rules and other factors provide appropriate grounds for trustworthiness .

4. Verifiability can be defined as the ability of an external party to observe a given aspect of a contractual relationship through the collected evidence. The quality or level of verifiability depends directly on the available evidence.

Remaining difficulties addressed by A4cloud is the development of mappings between the accountability contracts/SLAs and evidence available through logging. The framework should build an evidence base from which mappings of low level distributed remote IT logs can be mapped to high level policy requirements and service level agreements (SLAs). Evidence of accountability can therefore be provided and input to certification schemes or trustmarks. Figure 1 shows an overview of these relationships with log data being collected as evidence and evidence supporting auditing as well as assuring the previously mentioned accountability attributes addressed by the A4Cloud project.

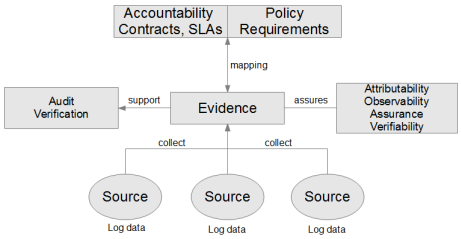

Fig. 1. Collecting Evidence and Mapping to Accountability

Environments in which there are diverse and heterogeneous service providers, make provision of protocols and models for trust verification and assurance difficult. The CloudTrust Protocol [13] defines some evidence categories, but has not covered other categories such as legal liability of the involved parties.

There are no efficient mechanisms available to gather convincing evidence from verified log data in distributed multi-tenancy environments, even if cloud providers would be willing to provide this. Although there are a number of existing logging approaches, they do not fit cloud computing very well. For example, EGEE LB log solution in grid computing is mostly used for debugging purposes only, as it keeps track of jobs. Even if verified log data is available, there are still challenges to make them compatible and interoperable. As different cloud providers implement and operate their systems differently, there is no guarantee that they all provide the same kinds of log information, which may expose weaknesses in their systems. There is currently no standard on log information to be delivered and there is no financial or regulatory incentive for the providers to provide such information. Furthermore, there is no accountability model for cloud, and therefore it is impossible to assign responsibilities even if the evidence exists. Neither are there any mechanisms for assigning responsibilities when the incident involves more than one provider based on gathered evidence in distributed systems.

4 Monitoring and Audit

Accountability mechanisms must be justified and Bennett [14] points out that a important process is independent testing of practices, provision of evidence that is taken into account, including auditing against the ISO 27001 series and associated cloud security standards. Evidence is provided by tools into trusted third party auditing processes against such standards.

ISO standards cover audit requirements at a high levelwhich is to maximize the effectiveness of and minimize interference to/from the information systems audit process. These solutions are not currently linked to formally defined accountability models, as accountability models only currently exist in terms of regulatory frameworks or point technical solutions. Accountability (for complying with measures that give effect to practices articulated in given guidelines) has been present in many core frameworks for privacy protection, like the Organization for Economic Cooperation and Development (OECD)'s privacy guidelines.

A4Cloud provide an approach based around a model of accountability that is interdisciplinary in approach, in which we build an evidence repository that provides evidence for preventive, detective and corrective accountability mechanisms by means of associated mechanisms for obtaining and negotiating obtaining these events from remote monitoring parties, and mechanisms for mapping the low level IT logs to what is in our repositories to policies and service level agreements (SLAs). In this way we bridge from distributed remote logs to high level policy requirements, and can detect policy violations. Audit capabilities in conjunction with external audit frameworks should be enhanced in order to strengthen the obligation for compliance and improve detection of violations.

5 Challenges of Evidence in Cloud Computing

Cloud forensics refers to digital forensics investigations performed in cloud computing environments. The process of a digital investigation can be separated into different phases as defined in the National Institute of Standards and Technology, "Electronic Crime Scene Investigation: An On-the-Scene Reference for First Responders" [15] each having its own specific purpose:

1. Securing Phase: The major intention is the preservation of evidence for analysis. The data has to be collected in a manner that maximizes its integrity. As can be imagined, this represents a huge problem in the field of cloud computing where you never know exactly where your data is and additionally do not have access to any physical hardware.
2. Analyzing Phase: Data from multiple systems or sources is pulled together to create as complete a picture and event reconstruction as possible.
3. Presentation Phase: Reporting all results in a clear and understandable way.

Current techniques in computer forensics can only analyze the evidence left behind by a careless intruder. We will use a combination of legal, technical and regulatory approaches to provide traceability, logging mechanisms and tools for

determining information provenance in distributed systems. This will underpin liability assignment and validation of insurance claims made in case of data breach or data loss. Evidence provided by our tools will enhance existing and developing certification schemes within the cloud.

With respect to the notion of evidence, it is important to differentiate between accountability and forensics. Digital forensics looks for unintended evidence, i.e. evidence that some party was not planning to leave and which collection was not planned ahead.

5.1 Sources of Evidence by Logging

The sources for logging can be manifold reaching from business relevant logging and operational logging. Operational logging could cover errors that concern a single cloud customer, critical conditions that impact all users, system related problems (e.g., failed resource access) and all activity that is executed by privileged accounts.

Sources of evidence to log, based on requirements and attributes, should be strengthened through the use of formal methods (e.g., formal logic). This is necessary to ensure the evidence quality in a situation where the amount of evidence-related data exceeds human reasoning capabilities.

Logging will need to be carried out at various stages of abstraction, i.e. at the system level, at the data level, at the service level, at the business level to determine when data is accessed, shared, moved, etc. The type of things that need to be logged at the data level are:

- *data creation:* the creation of a new data item, and the policies associated with this new item. The new item may be created by a user, or may arise from the automated copying or processing of data already in the system.
- *data access:* who accessed which data, for what purpose, the role of a person accessing the data, whether consent was obtained for usage from the data subject
- *data flow:* where the data is sent (including the jurisdiction), who shared data with whom
- *data type:* the type of data (e.g., is it personal, sensitive, etc.)
- *data deletion:* when was the data deleted, which erase method was used (unlink, delete data, delete backup, etc.)
- *data handling:* how data is handled to check conformance with some policies (e.g., data is stored password-protected or encrypted), data policy changes by the service provider, timing information (for example, for conformance to data retention policies)
- *data notification:* triggering and satisfaction of obligations

Subsequently, this information can be used in order to analyse whether organizational, regulatory and legal policies have been followed (this is a detective control, as opposed to checks made within the system associated with access control, etc. which are preventive). More specifically, we may want to focus on the following:

– segregation of duties; trans-border data flow; assurance that access control policies have been met
– assurance that obligations have been met
– records about how information was shared, with the context and associated obligations/sticky policies

5.2 Evidence Gathering Points

There are various locations to gather evidential data. As seen in Fig. 2 data (log data, memory, databases, etc.) can be collected at the network, hardware, host OS, hypervisor, the VMs, the CMS, the network and evidence data across other cloud platforms.

Network: In a complex computing model, such as Cloud, several stakeholders are involved. It should be possible to monitor networking resources which are utilized by a particular stakeholder. Networking resources can be either physical or virtual. Moreover, these resources can be shared among stakeholders. For instance in IaaS, a single network card in the host machine is utilized by several VMs and they may belong to different customers. Distinguishing between customer's traffic, which are hosted in a common set of substrates, is a key issue for accountability. This can also be applied to other service models of cloud, when traces of stakeholders' network activities must be available as an evidence type. However, existing networking devices and monitoring solutions are not compatible and efficient for such a multi-tenant environment.

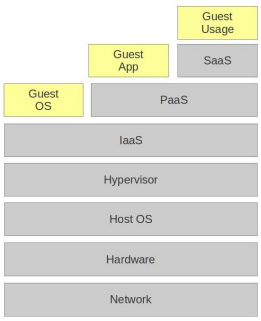

Fig. 2. Evidence Gathering Points

Hypervisor: The usage of data from hypervisors to prove various actual situations has been referred to as "virtual machine introspection" (VMI) and data gathered from this level of access supported the operation of Intrusion Detection Systems (IDS). It is suitable for investigating cloud infrastructures as long as there is access to the hypervisors.

VM: In order to obtain information from within VMs it could be helpful to install additional software inside the VMs. Carbone et al. [16] follow this approach by developing a secure and robust infrastructure called SYRINGE. The monitoring application is protected because it is put into a separate virtual machine as known from the out-of-guest approach. Nevertheless, it is possible to invoke guest functions by utilizing the function-call injection technique. The VM introspection make use of the guest OS knowledge of the deployed software architecture and can only be accessed with the customer's permission. A disadvantage arises from this component being susceptible to compromise from malicous entities.

CMS: The Cloud Management System (CMS) is a huge source for information gathering. It is the central controlling component of a cloud infrastructure and provides information about user logins, cloud service usage, access rights, configuration, resource provisioning, policies, etc.

IaaS: Except for traditional forensic acquisition at the virtual resources most interesting are VM snapshots which can accommodate preservation letters or serve as the acquisition image. Public clouds do not allow live forensics and access to volatile data. The storage is logical and focused on allocated space. Images can include data remnants or unallocated disk space. The logging may be co-located or spread across multiple and changing resources.

PaaS: In a web service PaaS the log data analysis can be carried out with the aforementioned methods, but relies on the cloud service provider. Multi-tanent log data must be separated or merged together from multiple resources.

SaaS: Access to application / authentication logs are possible to get and the SaaS application features may assist with network forensics. The logging information is located on the provider side and highly dependent of the application. The information may be inconsistent across API.

InterCloud: Cloud sources may be distributed over many providers and therefore collecting evidence over multiple sides is even more complex and difficult. There is a need of standardization of an evidence protocol, similar to the TrustCloud protocol.

6 Conclusion

The accountability approach taken in the EU FP7 A4Cloud project should help organisations meet their obligations and give cloud customers more control in cloud services. An evidence framework will be developed to assure accountability by building an evidence base gathering information. This information is collected at different level of the cloud stack and distributed in the infrastructure.

References

1. Schatz, B., Clark, A.J.: An open architecture for digital evidence integration. `http://eprints.qut.edu.au/21119/1/c21119.pdf`
2. Dykstra, J.: Seizing electronic evidence from cloud computing environments. In Ruan, K., ed.: Cybercrime and Cloud Forensics: Applications for Investigation Processes, Hershey, PA: Information Science (2013) 156–185
3. Haeberlen, A., Aditya, P., Rodrigues, R., Druschel, P.: Accountable virtual machines. In: Proceedings of the 9th USENIX conference on Operating systems design and implementation. OSDI'10, Berkeley, CA, USA, USENIX Association (2010) 1–16

4. Poisel, R., Malzer, E., Tjoa, S.: Evidence and cloud computing: The virtual machine introspection approach. Journal of Wireless Mobile Networks, Ubiquitous Computing, and Dependable Applications (JoWUA) **4**(1) (3 2013) 135–152
5. George, E., Mason, S.: Digital evidence and cloud computing. Computer Law & Security Review **27** (September 2011) 524–528
6. Flaglien, A., Mallasvik, A., Mustorp, M., Årnes, A.: Storage and exchange formats for digital evidence. Digital Investigation **8**(2) (2011) 122–128
7. Lu, R., Lin, X., Liang, X., Shen, X.S.: 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 (ASIACCS '10), New York, NY, USA, ACM (2010) 282–292
8. Marty, R.: Cloud application logging for forensics. In: Proceedings of the 2011 ACM Symposium on Applied Computing. SAC '11, New York, NY, USA, ACM (2011) 178–184
9. Casey, E.: Digital Evidence and Computer Crime - Forensic Science, Computers and the Internet, 3rd Edition. Academic Press (2011)
10. Pearson, S., Tountopoulos, V., Catteddu, D., Sudholt, M., Molva, R., Reich, C., Fischer-Hubner, S., Millard, C., Lotz, V., Jaatun, M., Leenes, R., Rong, C., Lopez, J.: Accountability for cloud and other future internet services. In: Cloud Computing Technology and Science (CloudCom), 2012 IEEE 4th International Conference on. (2012) 629–632
11. Pearson, S., Wainwright, N.: An interdisciplinary approach to accountability for future internet service provision. In Thampi, S.M., ed.: International Journal of Trust Management in Computing and Communications. Volume 1., INDERScience Publishers (2013) 52–72
12. Fischer-Hübner, S.: Transparency enhancing tools & hci for policy display and informed consent. In: Privacy, Accountability, Trust Challenges and Opportunities : ENISA Report. European Network and Information Security Agency, Technical Competence Department (2011)
13. Cloud Security Alliance (CSA): Cloud Trust Protocol. `https://cloudsecurityalliance.org/research/ctp`
14. Bennett, C.: 2. In: The Accountability Approach to Privacy and Data Protection: Assumptions and Caveats. Palgrave MacMillan (August 2012) 33–48
15. National Institute of Justice (U.S.): Electronic crime scene investigation: an on-the-scene reference for first responders. U.S. Dept. of Justice, Office of Justice Programs, National Institute of Justice (2009)
16. Carbone, M., Conover, M., Montague, B., Lee, W.: Secure and robust monitoring of virtual machines through guest-assisted introspection. In: RAID. (2012) 22–41