What is Really Going on at Your Cloud Service Provider? Creating Trustworthy Certifications by Continuous Auditing

Sebastian Lins
University of Cologne
lins@wiso.uni-koeln.de

Scott Thiebes
University of Cologne
thiebes@wiso.uni-koeln.de

Stephan Schneider
University of Cologne
schneider@wiso.uni-koeln.de

Ali Sunyaev
University of Cologne
sunyaev@wiso.uni-koeln.de

Abstract

Cloud service certifications attempt to assure a high level of security and compliance. However, considering that cloud services are part of an ever-changing environment, multi-year validity periods may put in doubt the reliability of such certifications. We argue that continuous auditing of selected certification criteria is required to assure continuously reliable and secure cloud services and thereby increase the trustworthiness of certifications. Continuous auditing of cloud services is still in its infancy, thus, we performed a systematic literature review to identify automated auditing methods that are applicable in the context of cloud computing. Our study yields a set of automated methods for continuous auditing in six clusters. We discuss the identified methods in terms of their applicability to address major concerns about cloud computing and how the methods can aid to continuously audit cloud environments. We thereby provide paths for future research to implement continuous auditing in cloud service contexts.

1. Introduction

An increasing number of organizations outsource their data and applications to the cloud, empowering them to achieve financial and technical benefits. Cloud computing (CC) enables ubiquitous, on-demand, and up to date IT resources and services, like networks, applications, or storage on a pay-per-use basis [37]. However, some organizations are still hesitant to adopt cloud services because of security, privacy, and availability concerns as well as doubts about the trustworthiness of cloud providers [50]. Thus, providers have to address these concerns and prove their credibility to increase adoption of potential cloud customers.

Cloud service certifications (CSC) are good means to establish trust, increase transparency of the cloud market, and allow providers to improve their processes and systems [51]. Several CSC, such as "*CSA STAR*" [13] or *"EuroCloud Star Audit"* [19], have recently evolved. These CSC attempt to assure a high level of security, availability, and legal compliance, for a validity period of one to three years. However, cloud services are part of an ever-changing environment, resulting from fast technology life cycles and inherent CC characteristics, for instance, on-demand provisioning and entangled supply chains. Hence, such long validity periods may put in doubt the reliability of issued certificates. Conditions and requirements of CSC may no longer be met throughout these periods, for instance, due to configuration changes or major security incidents [18]. Thus, we argue that, after the initial certification process is accomplished, continuous auditing of selected certification criteria is required to assure continuously reliable and secure cloud services and to establish a trustworthy CSC. Nevertheless, continuous auditing cannot be realized solely manually due to high costs and considerable expenditures, hence, requiring (semi-) automated auditing processes [7, 31]. Automation enables efficient validation of individual customer requirements (e.g., data location), allows for automated management of certificates for cloud providers, and increases efficiency of audits [31, 57].

Continuous auditing of cloud services is still in its infancy and few studies concerning the usage of (automated) methods exist [1, 2, 5]. Our objective is to address this gap and provide an overview of methods that can be used for continuous and (semi-) automated auditing of cloud environments. We conducted a systematic literature review to answer the following research question: *Which (semi-) automated auditing methods exist and are applicable in the CC context?* By answering this research question, we contribute to the literature on continuous auditing and certification of cloud services. This paper may aid as starting point for future research to develop metrics and methods for continuous and (semi-) automated auditing of crucial requirements (e.g., security and availability) of distributed information systems and cloud services in particular. By contributing to the scarce literature on continuous auditing of cloud services [1, 2, 51], we provide a basis for future research to address the prevailing concerns of (potential) cloud adopters [40, 41]. By assuring continuously reliable and secure cloud services, providers can establish trust in their services and thereby drive adoption of (potential) customers [27, 51].

This paper proceeds with a theoretical background on CC and continuous auditing, followed by the applied research approach. We then present our results followed by a discussion and conclusions.

2. Background

CC enables ubiquitous, on-demand network access to a shared pool of managed IT resources. These resources refer to hardware (Infrastructure as a Service), development platforms (Platform as a Service), and applications (Software as a Service) and "can be rapidly provisioned and released with minimal management effort or service provider interaction" [37, p. 2]. CC entails five essential characteristics, that are, the provision of (i) on-demand self-service access to (ii) virtualized, shared, and managed IT resources that are (iii) scalable on-demand, (iv) available over a network, and (v) priced on a pay-per-use basis [37].

Extant research already proposes certifications as a means to assess quality and performance of IT services in procurement processes [44]. In the context of CC, recent research provides design recommendations for CSC and suggests continuous auditing to increase reliability in CSC and to continuously assure compliance of a service to the certification requirements [1, 33, 46, 47, 51, 56].

A continuous audit is defined as "a methodology that enables independent auditors to provide written assurance on a subject matter, for which an entity's management is responsible, using a series of auditors' reports issued virtually simultaneously with, or a short period of time after, the occurrence of events underlying the subject matter" [12, p. xiii]. Thus, continuous auditing enables auditors to react to changes or events concerning the subject matter and to adjust their auditing reports based on the assessment of these changes and events. The concept of continuous auditing in the IS context was first introduced by [24] and [53]. Until today, a variety of terms, like continuous assurance or continuous monitoring emerged, which are also used to describe work in this area [7]. Throughout this paper, we use these terms synonymously as continuous auditing.

The diffusion of continuous auditing is advanced by improvements in auditing technologies and increasing claims for up to date and accurate assurances [7, 57]. Through automated, continuous auditing, audit efficiency and relevancy of audit reports can be improved as well as auditing costs reduced [57]. Concerning CSC, continuous auditing can improve the trustworthiness of these certifications, by securing the compliance of selected certification criteria [27, 56].

3. Research Approach

We apply a two-step research approach. We first conduct a systematic literature review to identify relevant publications before extracting (semi-) automated auditing methods from identified publications.

3.1. Literature Review

To identify publications addressing (semi-) automated auditing methods, we applied a systematic scientific database search in databases we deemed to be representative as they cover a wide range of journals and conferences (i.e., they cover the top computer science and information systems journals and conferences). We searched AIS Electronic Library, ACM Digital Library, EBSCOhost, Emerald Insight, IEEE Xplore, ProQuest, and ScienceDirect. To cover a broad set of publications, we searched each database with the following string in title and keywords: *(certif* OR audit* OR monitor* OR assur*) AND (continuous OR permanent* OR dynamic OR auto* OR realtime OR computerized OR (machine AND readable))*.

To assure the transferability to the CC context, we limited our search to sources published after 1980, because in 1981 the concept of TCP/IP was introduced [43]. Furthermore, screening of randomly sampled articles that matched the keywords and were published before 1980 did not yield relevant articles. The search was limited to peer-reviewed articles, if possible. Because of this broad search string, we identified 10,831 articles in this initial search.

To identify possibly relevant publications, two researchers independently checked the relevancy of each article by analyzing title, abstract, and keywords. If any indication for relevancy appeared, the article was marked for further processing. After this analysis phase was completed, the researchers compared and grouped their results. A large number of publications from medical (e.g., glucose and heart rate monitoring), environmental (e.g., vegetation monitoring), automotive (e.g., safety assurance), sensor network (e.g., energy certification of wireless sensor networks), and power supply contexts (e.g., power monitoring) were identified through the broad initial search and were then excluded, leading to a remaining set of 262 possibly relevant articles. Afterwards, a detailed relevance validation was made on the remaining 262 articles. Inclusion and exclusion criteria are listed in Table 1.

Again, two researchers independently assessed each article and discussed the results. We excluded research that does not propose (semi-) automated methods (38), is not applicable to CC (27), or off-topic (156). Moreover, duplicates (8) and non-research articles were excluded (5), leading to a final set of 28 relevant articles.

Table 1. Inclusion and exclusion criteria

Inclusion Criteria
(semi-) automated auditing methods and models
Exclusion Criteria
Published before 1981
No (semi-) automated methods
Work-in-progress, editorials, forums

3.2. Data Analysis

We carefully read and analyzed all relevant publications to identify the considered methods. For each extracted method, a name and a description, as well as the original source were recorded. Each method was analyzed, whether it is required to be embedded into the system and if real-time information processing and a full automation are possible. In total, 46 (semi-) automated methods were extracted. Merging duplicates (e.g., embedded audit modules and audit data marts were mentioned multiple times across the articles) and similar methods, led to a final set of 22 (semi-) automated methods. Concerning similar methods, for example, the concepts of digital agents [57], intelligent agents [49], and agent-based continuous auditing models [11] were aggregated to digital agents.

Table 2. Clusters of identified methods

System Architecture
Basic architectural concepts as well as components to monitor the system and gather data.
Logging and Inspection
Methods to create and analyze logs which contain information about system operation.
Application, Virtualization and Network
Methods to audit applications, virtual environments, and network operations.
Data Integrity
Methods to ensure integrity within and across clouds.
Intrusion, Anomaly and Behavior of Malware Detection
Methods for monitoring infrastructure and networks to detect intrusions, anomalies, and behavior of malware.
Compliance
Methods to ensure contractual and regulatory compliance, especially to service level agreements.

Following our identification of methods, we applied an inductive clustering approach. For this clustering, we listed the methods with a name and short description. Two authors independently analyzed the methods, and grouped them regarding to the objectives and application contexts. Afterwards, the independent clusters were discussed, which lead to minor method shifting and cluster renaming. The inductive clustering resulted in six clusters, which are summarized in Table 2. Tables 3 to Table 8 summarize the identified auditing methods for each cluster.

4. Findings

4.1. System Architecture

Enabling continuous auditing requires providers and auditors to implement suitable mechanisms into their system architectures. A permanent communication connection to the auditing object is necessary (i.e., the cloud service). To enable a permanent communication, [15] developed a continuous auditing model, which effectively connects the auditor's systems to the auditee's systems, by using XML and CORBA. Similar auditing architectures include independent monitoring and control layers for continuously querying and analyzing auditing objects [3, 42, 52, 53]. Additionally, different kinds of middleware and standard interfaces [49] as well as independent auditing web services [39] are suggested to be integrated in continuous auditing systems, allowing cloud service customers to continuously review and validate the adherence with selected certification criteria. Moreover, several computer-assisted auditing technologies and tools are proposed to enable continuous auditing. For example, [30] analyzed the OVAL (Open Vulnerability and Assessment Language) and the XCCDF (Extensible Configuration Checklist Description Format) to enable (semi-) automated IT controls checks, patch and configuration management validation, and vulnerability assessments. The provided tools, processes, and capabilities may differ based upon the extent, kind, and objectives of the used auditing system [15, 42, 52, 53].

Aside from integrated or packaged auditing systems, researchers developed several individual components, which can be implemented into the auditors' and / or the auditees' systems. The most frequently mentioned component is an embedded audit module (EAM). EAMs are special purpose functions, programs, or other code objects that are embedded into the auditees' information systems and supervise all of the audit-related data in real-time [3, 9, 11, 24, 45, 48]. One of the advantages of EAMs is that they automatically act as triggers and inform the auditor when suspicious events appear, thus, eliminating the need for a high frequency of assurance queries [11, 24, 48]. However, EAMs are more vulnerable to manipulation, especially by the audtiees' employees who have the necessary privileges to interfere with the EAM [3, 24]. The use of EAMs for continuous auditing of cloud services may be limited, because the incorporation of EAMs into a cloud architecture that is distributed across different datacenters and locations requires a complicated development and customization process [3, 24].

Data captured by continuous audits can be stored in an audit data mart (ADM). ADMs are small, mostly auditee-independent data repositories in which relevant data from all application systems are automatically integrated [11, 14, 45]. Thus, ADMs enable a real-time data access on which continuous (semi-) automated analysis and auditing can be performed. ADMs may be used for continuous auditing of cloud services, if appropriate data formats are available and a secure access to the collected data is guaranteed.

Table 3. Cluster system architecture

Auditing Architectures
Auditing architectures that may consist of monitoring and control layers [3, 42, 52, 53] and a permanent communication connection to the auditing object [15] as well as integrating different kinds of middleware, standard interfaces [49], or auditing web services [39].
Computer-assisted Auditing Technologies & Tools
Computer-assisted technologies and tools for auditing, for instance, to enable (semi-) automated IT configuration management validation and vulnerability assessments [30].
Embedded Audit Module (EAM)
EAMs are special purpose code objects (e.g., programs) that are embedded into the auditees' information systems and supervise the auditee's system [3, 9, 11, 24, 45, 48].
Audit Data Mart (ADM)
ADMs are small data repositories in which relevant data from all application systems are automatically integrated and analyzed [11, 14, 45].
Digital Agents (DA)
DAs are intelligent and mobile software objects that achieve individual goals by autonomously performing actions that are traditionally undertaken by human auditors [11, 16, 21, 49, 57].

Furthermore, continuous auditing models that use multiple digital agents to support auditing processes are suggested [11, 16, 21, 49, 57]. Digital agents (DA) (also referred to software, autonomous, or intelligent agents) are software objects that achieve individual goals by autonomously performing actions and reacting to events in a dynamic environment. Furthermore, they are characterized by having different degrees of artificial intelligence and mobility (the ability to travel from one platform to another) [11, 49]. DAs are supposed to automatically perform activities that are traditionally undertaken by human auditors, for example, collecting and evaluating information and audit evidence as well as validating certification requirements [11, 49, 57]. Through their artificial intelligence, mobility and individual, autonomous acting, they seem to be very suitable for continuous auditing of cloud services, especially when comparing DAs to EAMs. However, high efforts and expense for DA development and implementation, and possible negative impacts on system performance have to be considered [11].

4.2. Logging and Inspection

Methods for logging events form a foundation for auditing processes. For many certification criteria, compliance can be automatically ensured based upon the computer-aided inspection of the resulting logs, without creating significant performance overhead.

A suggested solution to implement efficient logging structures in cloud environments is a layered logging framework to increase accountability of cloud services [29]. The framework consists of different logging layers: system layer, data layer, and workflow layer. First, the system layer creates logs, which contain information about the operating system, (file) system events, virtual and physical memory, and network traffic. Second, the data layer produces logs about the data storage system of a cloud service. Data layer logs can be subdivided into logs recording the provenance of data and logs documenting the consistency of stored data. Lastly, the workflow layer is concerned with how clouds can achieve high auditability. For example, logs are required for auditing the patch management process or to increase the accountability of cloud services by logging processes in detail. Additionally, policies, laws, and regulations require further information to be logged. This framework may serve as a foundation for future continuous auditing of cloud services regarding logging and log inspection. However, a precise implementation is still missing.

Table 4. Cluster logging and inspection

Logging Framework
Layered cloud logging framework to create logs, which contain information about the operating system, data, and workflows [29].
Abstract Execution Log Inspection
Using logs to monitor the execution of applications with limited log format requirements [26].

Aside from that, abstract execution logs to monitor the execution of applications are a suitable solution to enable continuous auditing of cloud service applications [26]. Such an approach enables heuristics-based log inspection techniques, which can inspect log lines with limited format requirements and can scale up to process log files which contain thousands or millions of log lines [26]. Supposedly, such a method can be used to automatically and continuously check whether different applications are actually running on a cloud infrastructure, for example, malware protection or antivirus software. Additionally, it may be used to automatically identify prohibited application execution (e.g., restricted access).

4.3. Application, Virtualization and Network

Especially when software is provisioned from multi-tenant and virtualized cloud environments, auditing applications is essential. Two types of methods can be distinguished: auditing of applications and auditing of interactions between applications, virtual machines, and virtual environments. To automatically determine the status of applications, a framework consisting of specific virtual machine (VM) and analyzer modules for virtualized cloud environments is proposed [35]. By using this framework, auditors can detect attacks on executables by noticing measurement changes, thus, increasing the security of VMs. A prototype of this framework was implemented and tested, and has demonstrated performance efficiency. However, the secure and flawless interaction of application instances running on different VMs in different virtualized cloud environments has to be validated as well. Therefore, an automated model is proposed that consists of three layers: local application surveillance (LAS), intra-platform surveillance (IPS), and global application surveillance (GAS) [23]. Each application instance is monitored by an LAS component, to examine if the instance violates any established monitoring rule and to detect malicious behavior or implementation flaws. Furthermore, to monitor interaction problems between different virtualized environments, IPS components are allocated to each virtual machine and are also interconnected with other IPS components of the same virtualized environment. IPS components evaluate the results of the LAS components from their allocated VM and check for security risks that might arise through interaction of different applications or VMs. Lastly, GAS components analyze data from different VMs referred to the same application. Therefore, GAS components receive and analyze information from several IPS components and have a global view of an application behavior in different virtualized environments. Both discussed examples were developed for cloud environments, hence, after integrating them into the cloud service architecture, they can be used to continuously and automatically inform an auditor about several certification violations regarding application or virtualization usage.

Further on, a dynamic network monitoring method to ensure network reliability and gather network information was identified through the literature review. This method is based on the incorporation of DA and was developed by the cross-platform language Python [58]. It enables automatic network monitoring as well as manual intervention (e.g., on emergency events). Additionally, experimental results show that the method can be used in various network environments and topologies. Thus, this method may be used to automatically monitor CC networks. Especially, by analyzing monitoring results, the auditor may continuously audit the reliability and availability of cloud services, which are still major concerns regarding CC [17, 50].

Table 5. Cluster application, virtualization and network

Framework for Increasing VM Security
Determines the status of applications and detects attacks on executables in cloud environments [35].
Application Monitoring Model
Monitors the secure and flawless interaction of application instances running on different VMs in different virtualized cloud environments [23].
Dynamic Network Monitoring
Ensures network reliability and gathers network information by incorporating DAs [58].

4.4. Data Integrity

Ensuring data integrity in cloud environments is a challenging task, because of multitenant architectures and distributed systems [50]. Thus, to create trustworthy cloud services, auditors should continuously validate that data integrity is maintained.

Hashing techniques have been identified as adequate methods for monitoring the integrity of large amounts of data and many files [34, 54, 55, 59, 61]. Five different methods could be identified that enable a third party to audit and validate the integrity of data stored in a cloud. Every method capacitates auditors to simultaneously verify the integrity of multiple users' data, which is important in multitenant cloud environments with many users operating at the same time. Moreover, simultaneous monitoring of multiple clouds and multiple owners is feasible [59] and auditors are able to detect anomalous behaviors of data operations [61]. Aside from that, these methods support dynamic data operations [54, 55, 59, 61], even on a fine-grained level, thus, facilitating the consideration of dynamic, minor data changes [34]. When auditing integrity, data security and privacy can be ensured by implementing cryptography [59], authentication [54], or authorization techniques [34, 61]. Furthermore, using periodic sampling audits [61] or moving computational operations onto the cloud server [59], auditors can reduce communication and computation cost, which leads to increased audit efficiency.

These methods form a comprehensive sample for enabling continuous, secure, and privacy-preserving auditing of cloud storage data integrity, with low computational overhead.

Table 6. Cluster data integrity

Auditing Scheme for Data Integrity
Enables auditors to verify data integrity of multiple users' data and incorporates authentication techniques for data protection [54, 55].
Multicloud Batch Auditing Protocol
Verifies data integrity of multiple clouds and multiple owners [59].
Periodic Sampling Audit
Using periodic sampling audits to verify data integrity of multiple users' data and to detect anomalous behaviors of data operations [61].
Authorized Auditing Scheme
Auditors can verify the data integrity of multiple users' data while to consider fine-grained data changes [34].

4.5. Intrusion, Anomaly and Behavior of Malware Detection

Security is one of the most important and most discussed topics concerning CC. Especially by establishing CSC, for example the '*CSA STAR Certification*' [13], auditors verify the implementation of proper security mechanisms for cloud services. Cloud services are particularly exposed by risks of malicious behavior from external attackers, cloud service users as well as malicious employees [17, 50]. Thus, auditors need to continuously verify, whether a cloud provider established and operates mechanisms to prevent intruders performing malicious operations.

In a survey on automated and real-time intrusion detection, several techniques, including the use of neural networks, expert systems, and model-based reasoning for intrusion detection are identified [36]. Such techniques monitor and analyze behavior and actions of users, compare them to established norms and past behavior, and check for suspicious events (e.g., sudden late hour accesses) to provide evidence by interpreting audit trails. The presented techniques mainly differ in matching user behavior as well as detection of suspicious events. More recently, machine-independent approaches for intrusion and anomaly detection using a knowledge-based system have been proposed [6]. Knowledge-based systems perform, for instance, intelligent analysis of operating system audit trails and assess unauthorized user activity in multi-user computer systems. When intrusions and anomalies are automatically and continuously detected, information overload is likely to appear, thus leading to limited decision making and action taking [42]. For that reason, an architecture was proposed that automatically and continuously detects anomalies and automatically aggregates and evaluates detected anomalies [42].

The presented methods show that development of continuous auditing mechanisms for intrusion and anomaly detection has already been well advanced. Furthermore, these methods detect intrusion and anomalies by analyzing audit trails independent from machines, thus, they can be used to enable continuous auditing of cloud services. Still, they need to be adjusted to deal with the challenges of cloud service characteristics. First of all, a high amount of customers using (different) cloud services will greatly increase the number of event records and the complexity of audit trails, compared to traditional in-house application usage. Secondly, due to virtualized and distributed cloud architectures, the detection of malicious behavior will be more difficult. Thirdly, cloud services are not only exposed by risks of external attacks, but also are confronted with malicious insider and malicious cloud customer behavior, thus, the detection mechanisms have to be adjusted and sensitized for this type of anomalies and intrusions.

When different users interact with system files simultaneously, malicious modification of such system files affects all users. For that reason, file system integrity must be continuously ensured, especially in the context of multitenant cloud services. Several file system integrity tools are already developed and allow administrators to automatically detect system changes and malicious file modifications [28]. However, the concept of file system integrity validation has to be adjusted for virtualized cloud environments. Implementing monitoring processes on VMs and modules into hypervisor levels is one proposed approach for matching requirements of virtualized environments [28]. Due to low performance overhead, this approach enables real-time file system monitoring, which is particularly suitable for virtualized cloud service contexts.

Regularly performing penetration tests is recommended to validate adequate security mechanisms [3]. By attempting to execute prohibited behavior or attacks on the cloud service, auditors can verify that such behavior is prevented or detected and compensated [3]. Such testing provides strong evidence for comprehensive and exhaustive protection mechanisms. However, it is highly unlikely that an auditor will be allowed to execute penetration tests on production systems [4], especially in a continuous frequency (e.g., weekly or monthly repetitions). Moreover, performing penetration tests in cloud environments will affect multiple customers (e.g., temporary performance losses or operational disturbances), even though some customers may not insist on continuous penetration tests.

Table 7. Cluster intrusion, anomaly and behavior of malware detection

Intrusion Detection Systems
Techniques to monitor and analyze behavior and actions of users to check for suspicious events and detect intrusions [6, 36, 42].
File System Integrity Tool for Virtual Machines
Administrators are able to automatically detect system changes and malicious file modifications [28].
Penetration Testing
Performing regularly penetration tests to validate adequate security mechanisms [3].

4.6. Compliance

Adherence to contractual (e.g., service level agreements) and regulatory agreements has to be continuously audited, especially in context of CC, in which customers have a lack of control and responsibilities may not be transparent [17].

To evaluate and validate adherence to contractual and regulatory agreements, measurable requirements or service level objectives have to be specified, for example, expected availability, throughput, or response time [22, 32]. Afterwards, requirements have to be transformed into a formal, machine-understandable representation, for instance, an ontology-based representation [32] or using a combination of XML, formulas, and logics [22]. To automatically monitor the adherence of specified criteria, mechanisms (e.g., digital agents) have to be incorporated into the cloud environment that can make assertions, ask queries, or gather necessary information [22, 32]. The identified approaches were developed in the context of web services and internet standards [22, 32], thus, they seem to be suitable to be used for semi automated cloud service auditing.

Another method validates the configuration of cloud services including security and service level agreement compliance, at the time services are created [10]. The proposed validation system consists of a variety of components, such as: an automation engine, DAs, an evidence database, and a script repository [10]. Several actions are performed after a service is created, to ensure configuration, security, and service level agreements compliance, for instance, anti-virus is running with the latest signature file and all available security patches are applied. Such an automation solution has already been implemented and deployed in a private enterprise cloud and in several customer-dedicated private clouds [10].

By applying any model of this cluster, a continuous auditing of contractual and regulatory compliance for cloud services can be performed. Therefore, auditors must be immediately informed after compliance violations appeared. In addition to that, (semi) automation of these validation processes can enable an efficient validation of individual customer requirements (e.g., a high availability rate) [31].

Table 8. Cluster compliance

Monitoring contractual and regulatory agreements
Methods for specifying agreements, transforming these agreements into a machine-understandable representation, and continuously monitoring the adherence to these agreements [22, 32].
Cloud Service Configuration Validation
Method to validate the configuration of cloud services including service level agreement compliance [10].

5. Discussion

Our systematic review of extant literature on methods for continuous auditing yielded in six clusters of 22 auditing methods. With this paper, we provide an overview on which methods may be used to continuously audit (distributed) information systems and evaluated their applicability for continuous auditing of cloud services.

Before implementing continuous auditing for cloud services, one first has to evaluate and specify which CSC criteria need a high frequency auditing after the initial certification process was completed. Thus, CSC criteria have to be evaluated, for example, regarding the following questions: Does the criterion imply actions, which have to be performed on a regular basis? Can the cloud provider easily realize benefits (e.g., cost reductions) by discontinuing adherence to the criterion? Consequently, not every CSC criterion needs to be continuously audited to ensure on-going adherence. Moreover, for each criterion one has to determine an auditing frequency (e.g., real-time, daily, monthly, or quarterly). After deciding which criteria may be suitable for continuous auditing, one has to select proper methods for every criterion or a bundle of criteria. For instance, certification criteria concerning regularly backup of data and software of a cloud service according to customer agreed backup policies may be continuously audited by applying the concept of digital agents (see section 4.1), who autonomously and automatically check for backup files, or by inspecting backup logs (see section 4.2). However, even if some CSC criteria are suitable to be continuously audited and proper methods can be implemented, economic feasibility and effectiveness has to be ensured. Hence, continuous auditing of criteria that only take in a minor role in CSC, but require high configuration and implementation expenditures on the other hand, may not be realized due to economic limitations.

When implementing continuous auditing, several challenges need to be considered. First of all, one has to decide between embedded and independent continuous auditing technologies. Embeddedness requires some degree of integration of an auditor's modules into auditees' systems (e.g., EAMs and VM modules). After the integration of monitoring and auditing techniques, detailed and real-time information can be continuously gathered and evaluated with low communications costs, thus, allowing auditors to validate the adherence to specified requirements. However, integration requires extensive modifications to the auditees' systems, which can be quite expensive to realize, especially post hoc [39]. Furthermore, integrated modules, developed for one cloud provider, may not be easily utilized for another provider [15]. Also, the scope of integration may be limited, because of regulatory or technical requirements as well as security concerns, for instance, regarding interference with critical cloud applications [15]. Moreover, auditees are not necessarily willing or obligated, and may be even resisting, to integrate auditors' techniques into their systems [3, 24, 42]. Besides, when embedding continuous auditing techniques, it is important to implement protection mechanisms to prevent any manipulation (e.g., withholding evidence) from cloud providers [24, 39].

Independent auditing techniques (e.g., DAs or externally analyzing audit trails) can overcome such problems of embeddedness. These techniques require a read-only access to the audit-related information [3, 11], thus, minimum interference with the auditees' cloud systems is required and conflicts are reduced [3, 15, 57]. However, having an independently developed auditing system that connects with each provider can be complex and expensive, especially concerning an efficient match of auditees' heterogonous data formats and legacy systems [15, 20]. Moreover, appropriate security mechanisms have to be established to prevent unauthorized access or any security risks [11, 15]. "Transmission of information between parties must be authorized and have confidentiality, integrity, and authentication" [57, p. 5]. Further problems may be a high performance impact on the auditees' systems [11, 24, 39], high performance requirements for auditors' systems, and high network communication costs [60].

In general, benefits related to continuous auditing will be difficult to specify and to quantify [7]. Cloud providers may realize several advantages, when participating in continuous auditing. First of all, internal processes and systems may be improved by implementing suitable monitoring techniques and evaluating continuous feedback about how they are performing [7]. Additionally, cloud providers can differentiate themselves on the cloud market by making their cloud services more transparent to customers, thus, they may realize competitive advantages. Further on, auditors' reports are more relevant to decision makers of potential cloud service adopters, because of timely information regarding the certified criteria [57].

Through automated continuous auditing, the audit efficiency can be improved, by reducing auditing time and errors in the auditing process [3, 24, 25, 57]. Furthermore, automated audit processes are often more cost-effective, by enabling auditors to test larger samples, and examine data faster and more efficiently, compared to their manual predecessors [7, 45, 57]. Continuous auditing allows the auditor to actively detect and investigate exceptions as they occur rather than to react after the exception has long occurred [8]. Hence, continuous auditing can be considered as proactive and enables corrective action to be taken as soon as a problem is detected [8, 20]. More importantly, through this timely detection and continuous assurance for certification adherence, continuous auditing can improve the trustworthiness of auditors' CSC [56]. Auditors can counteract the lack of cloud customers' control in CC environments [17] by increasing the transparency regarding operations of cloud providers.

Continuous auditing, however, has some limitations. Automation as well as continuous auditing require a strong formalization of auditing processes, which is not achievable for every process at the moment [3, 8]. Human auditors still need to manually validate specific requirements of cloud services, because some weaknesses might remain unrecognized on automated validation systems [38]. Aside from that, automation of industry specific auditing processes may be limited too, because of high expenditures for customization and individualization of auditing processes.

For continuous auditing of cloud services to become widely adopted, it must be technologically and economically feasible [53, 57]. Cloud providers, as well as auditors, must be motivated and have the expertise to participate in continuous auditing [57]. To motivate providers to participate in continuous auditing, perceived benefits must be higher than perceived expenditures. Especially if an increasing amount of customers demands trustworthy certified cloud services, providers may start to open up for continuous auditing. To enable and facilitate the diffusion process of trustworthy CSC and continuous auditing, future research concerning the development of continuous auditing frameworks and applicable methods in CC contexts is necessary.

6. Conclusion and Future Work

The ever-changing cloud environment, fast update cycles, and the increasing adoption of business-critical applications from cloud providers demand for highly reliable cloud services. Continuously auditing such cloud services can prove a high level of reliability to (potential) cloud service adopters. However, methods to efficiently assess cloud services continuously are still in their infancy. The existing work of academics and practitioners concerning continuous methods for auditing information systems provides a fruitful basis for future research to develop continuous auditing methods for cloud services. With our study, we provide a first step to increase trustworthiness of CSC, by evaluating the applicability of methods to continuously audit cloud services.

We contribute to practice by illustrating methods which can be used to enable continuous auditing of (distributed) information systems. Further on, we evaluate the applicability of these methods for service providers to continuously monitor their cloud services, and more importantly, for auditors to enable continuous auditing of cloud services. Additionally, some of these methods have already shown to be efficient in productive use. We want to encourage CSC auditors to implement continuous auditing techniques, to create trustworthy certifications. Furthermore, new business models, for instance, monitoring as a service, may emerge out of these contexts.

With this study, we provide a basic set of continuous auditing methods for further research. We also transferred the concept of continuous auditing in a new context, and provided a first evaluation about which methods may be suitable, but also demonstrated challenges, limitations and benefits of continuous auditing of cloud services. However, as the preceding discussion of findings reveals, there is still plenty of research to do. Further research should focus on the identification of additional methods, especially concerning security adherence and adherence to critical cloud service characteristics (e.g., availability and scalability of services). The identified methods need to be evaluated and implemented to proof their practical and economic applicability in cloud environments. Moreover, to build a precise framework for continuous auditing of cloud services, the identified methods for continuous auditing need to be mapped to certification criteria of existing CSC schemes and corresponding metrics to measure criteria adherence have to be developed. Furthermore, research should focus on evaluations regarding acceptance and benefits of cloud providers when participating in continuous auditing as well as drivers and inhibitors for cloud service customers' demand for continuous auditing. Besides, a framework and guidelines have to be specified, to handle violations of CSC criteria on a continuous basis.

7. Acknowledgement

This research was conducted in context of the Value4Cloud research project, funded by the German Federal Ministry for Economics and Technology (grant no. 01MD11043A).

8. References

[1] R. Accorsi, L. Lowis, and Y. Sato, "Automated Certification for Compliant Cloud-based Business Processes", BISE, 3(3), pp. 145–154, 2011.
[2] G. Aceto, A. Botta, W. de Donato, and A. Pescapè, "Cloud monitoring: A survey", Comput Netw, 57(9), pp. 2093–2115, 2013.
[3] M. Alles, G. Brennan, A. Kogan, and M.A. Vasarhelyi, "Continuous monitoring of business process controls", IJAIS, 7(2), pp. 137–161, 2006.
[4] M.G. Alles, A. Kogan, and M.A. Vasarhelyi, "Putting Continuous Auditing Theory into Practice", J Inform Syst, 22(2), pp. 195–214, 2008.
[5] http://www.bmwi.de/EN/Service/publications,did=476736.html, 2012.
[6] P.J. Best, G. Mohay, and A. Anderson, "Machine-independent audit trail analysis--a tool for continuous audit assurance", ISAFM, 12(2), pp. 85–102, 2004.
[7] C.E. Brown, J.A. Wong, and A.A. Baldwin, "A Review and Analysis of the Existing Research Streams in Continuous Auditing", JETA, 4(1), pp. 1–28, 2007.
[8] D.Y. Chan and M.A. Vasarhelyi, "Innovation and practice of continuous auditing", IJAIS, 12(2), pp. 152–160, 2011.
[9] Y. Chen, "CONTINUOUS AUDITING USING A STRATEGIC-SYSTEMS APPROACH", Internal Auditing, 19(3), pp. 31–36, 2004.
[10] T. Chieu, M. Singh, Chunqiang Tang, M. Viswanathan, and A. Gupta, "Automation System for Validation of Configuration and Security Compliance in Managed Cloud Services", ICEBE Proceedings, 2012.
[11] C.L.-y. Chou, T. Du, and V.S. Lai, "Continuous auditing with a multi-agent system", Decis Support Syst, 42(4), pp. 2274–2292, 2007.
[12] CICA/AICPA, "Continuous auditing. Research Report", The Canadian Institute of Chartered, Toronto, Canada, 1999.
[13] https://cloudsecurityalliance.org/star, 2014.
[14] J.S. David and P.J. Steinbart, "Drowning in Data", Strategic Finance, 81(6), pp. 30–36, 1999.
[15] H. Du and S. Roohani, "Meeting Challenges and Expectations of Continuous Auditing in the Context of Independent Audits of Financial Statements", International Journal of Auditing, 11(2), pp. 133–146, 2007.
[16] T.C. Du, E.Y. Li, and E. Wei, "Mobile agents for a brokering service in the electronic marketplace", Decis Support Syst, 39(3), pp. 371–383, 2005.
[17] https://www.enisa.europa.eu/activities/risk-management/files/deliverables/cloud-computing-risk-assessment/at_downl

oad/fullReport, 2009.

[18] http://enisa.europa.eu/activities/Resilience-and-CIIP/cloud-computing/incident-reporting-for-cloud-computing, 2013.

[19] http://eurocloud-staraudit.eu, 2014.

[20] S. Flowerday, A.W. Blundell, and R. von Solms, "Continuous auditing technologies and models", Computers & Security, 25(5), pp. 325–331, 2006.

[21] A. Fuggetta, G. Picco, and G. Vigna, "Understanding code mobility", IEEE Transactions on Software Engineering, 24(5), pp. 342–361, 1998.

[22] N. Goel, N. Kumar, and R.K. Shyamasundar, "SLA Monitor: A System for Dynamic Monitoring of Adaptive Web Services", ECOWS Proceedings, 2011.

[23] J. Gonzalez, A. Munoz, and A. Mana, "Multi-layer Monitoring for Cloud Computing", HASE Proceedings, 2011.

[24] S.M. Groomer and U.S. Murthy, "Continuous Auditing of Database Applications", J Inform Syst, 3(2), p. 53, 1989.

[25] S. Il-hang, L. Myung-gun, and W. Park, "Implementation of the continuous auditing system in the ERP-based environment", Managerial Auditing Journal, 28(7), pp. 592–627, 2013.

[26] Z.M. Jiang, A. Hassan, P. Flora, and G. Hamann, "Abstracting Execution Logs to Execution Events for Enterprise Applications", QSIC Proceedings, 2008.

[27] K.M. Khan and Q. Malluhi, "Trust in Cloud Services", Computer, 46(7), pp. 94–96, 2013.

[28] J. Kim, I. Kim, and Y.I. Eom, "NOPFIT: File System Integrity Tool for Virtual Machine Using Multi-byte NOP Injection", ICCSA Proceedings, 2010.

[29] R. Ko, P. Jagadpramana, M. Mowbray, S. Pearson, M. Kirchberg, Qianhui Liang, and Bu Sung Lee, "TrustCloud: A Framework for Accountability and Trust in Cloud Computing", SERVICES Proceedings, 2011.

[30] G. Koschorreck, "Automated Audit of Compliance and Security Controls", IMF Proceedings, 2011.

[31] T. Kunz, P. Niehues, and U. Waldmann, "Technische Unterstützung von Audits bei Cloud-Betreibern", Datenschutz und Datensicherheit, 37(8), pp. 521–525, 2013.

[32] S. Lamparter, S. Luckner, and S. Mutschler, "Formal Specification of Web Service Contracts for Automated Contracting and Monitoring", HICSS Proceedings, 2007.

[33] J. Lansing, S. Schneider, and A. Sunyaev, "Cloud Service Certifications: Measuring Consumers' Preferences for Assurances", ECIS Proceedings, 2013.

[34] C. Liu, J. Chen, L. Yang, X. Zhang, C. Yang, R. Ranjan, and K. Ramamohanarao, "Authorized Public Auditing of Dynamic Big Data Storage on Cloud with Efficient Verifiable Fine-grained Updates", IEEE Transactions on Parallel and Distributed Systems, 25(9), 2013.

[35] Q. Liu, C. Weng, M. Li, and Yuan Luo, "An In-VM Measuring Framework for Increasing Virtual Machine Security in Clouds", IEEE S&P, 8(6), pp. 56–62, 2010.

[36] T.F. Lunt, "A survey of intrusion detection techniques", Computers & Security, 12(4), pp. 405–418, 1993.

[37] http://csrc.nist.gov/publications/nistpubs/800-145/SP800-145.pdf, 2011.

[38] R.T. Mercuri, "On Auditing Audit Trails", Commun ACM, 46(1), pp. 17–20, 2003.

[39] U.S. Murthy and S.M. Groomer, "A continuous auditing web services model for XML-based accounting systems", IJAIS, 5(2), pp. 139–163, 2004.

[40] M. Nanavati, P. Colp, B. Aiello, and A. Warfield, "Cloud security", Commun ACM, 57(5), 2014.

[41] B. Narasimhan and R. Nichols, "State of Cloud Applications and Platforms: The Cloud Adopters' View", Computer, 44(3), pp. 24–28, 2011.

[42] J.L. Perols and U.S. Murthy, "Information Fusion in Continuous Assurance", J Inf Syst, 26(2), pp. 35–52, 2012.

[43] www.rfc-editor.org/info/rfc791, 1981.

[44] C.-P. Praeg and U. Schnabel, "IT-Service Cachet - Managing IT-Service Performance and IT-Service Quality", HICSS Proceedings, 2006.

[45] Z. Rezaee, A. Sharbatoghlie, R. Elam, and P.L. McMickle, "Continuous auditing: Building automated auditing capability", Auditing, 21(1), pp. 147–163, 2002.

[46] S. Schneider, J. Lansing, F. Gao, and A. Sunyaev, "A Taxonomic Perspective on Certification Schemes", HICSS Proceedings, 2014.

[47] S. Schneider, J. Lansing, and A. Sunyaev, "Empfehlungen zur Gestaltung von Cloud-Service-Zertifizierungen", Industrie Management, 29(4), pp. 13–17, 2013.

[48] B. Schroeder, "On-line monitoring: a tutorial", Computer, 28(6), pp. 72–78, 1995.

[49] J.M. Shaikh, "E-commerce impact: emerging technology - electronic auditing", Managerial Auditing Journal, 20(4), pp. 408–421, 2005.

[50] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing", J Netw Comput Appl, 34(1), pp. 1–11, 2011.

[51] A. Sunyaev and S. Schneider, "Cloud services certification", Commun ACM, 56(2), pp. 33–36, 2013.

[52] M.A. Vasarhelyi, M.G. Alles, A. Kogan, and D. O'Leary, "Principles of Analytic Monitoring for Continuous Assurance", JETA, 1(1), pp. 1–21, 2004.

[53] M.A. Vasarhelyi and F.B. Halper, "The Continuous Audit of Online Systems", Auditing, 10(1), pp. 110–125, 1991.

[54] C. Wang, S.S.M. Chow, Qian Wang, K. Ren, and W. Lou, "Privacy-Preserving Public Auditing for Secure Cloud Storage", IEEE Trans. Computers, 62(2), pp. 362–375, 2013.

[55] Q. Wang, C. Wang, K. Ren, W. Lou, and J. Li, "Enabling Public Auditability and Data Dynamics for Storage Security in Cloud Computing", IEEE Transactions on Parallel and Distributed Systems, 22(5), pp. 847–859, 2011.

[56] I. Windhorst and A. Sunyaev, "Dynamic Certification of Cloud Services", ARES Proceedings, 2013.

[57] J. Woodroof and D. Searcy, "Continuous audit implications of Internet technology", HICSS Proceedings, 2001.

[58] F. Wu, Z. Zhao, and X. Ye, "A New Dynamic Network Monitoring Based on IA", ISCSCT Proceedings, 2008.

[59] K. Yang and X. Jia, "An Efficient and Secure Dynamic Auditing Protocol for Data Storage in Cloud Computing", Transactions on Parallel and Distributed Systems, 24(9), pp. 1717–1726, 2013.

[60] H. Ye, J. Yang, and Y. Gan, "Research on Continuous Auditing Based on Multi-agent and Web Services", ICMeCG Proceedings, 2012.

[61] Y. Zhu, G.-J. Ahn, H. Hu, S. Yau, H. An, and C.-J. Hu, "Dynamic Audit Services for Outsourced Storages in Clouds", IEEE Transactions on Services Computing, 6(2), pp. 227–238, 2013.