Providing Continuous Assurance

Kocken, Jonne ; Hulstijn, Joris

Publication date:
2017

Document Version
Early version, also known as pre-print

Providing Continuous Assurance

Jonne Kocken[1] and Joris Hulstijn[2]

1. Deloitte Risk Advisory, Amsterdam, JKocken@deloitte.nl
2. Tilburg University, Tilburg, j.hulstijn@uvt.nl

Abstract. It has been claimed that continuous assurance can be attained by combining continuous monitoring by management, with continuous auditing of data streams and the effectiveness of internal controls by an external auditor. However, we find that in existing literature the final step to continuous assurance has been left unexplained. In this paper we propose an architecture and procedure for attaining continuous assurance. It consists of a dashboard architecture collecting results of continuous monitoring and continuous auditing, combined with a report generator and workflow system that is able to generate an official assurance report, when required. The proposal is motivated by a case study of a platform for monitoring the security of healthcare Apps. The case indicates that the proposal would be technically and legally feasible. Economic viability has not been investigated.

1 Introduction

The world is constantly changing. Facts and figures must be updated continuously. Decision makers need assurance, i.e. certainty or confidence, about the reliability of this ever changing data. This suggests a need for assurance over continuous streams of data, or in short, *continuous assurance*. Since the 1990's a family of computational audit approaches has been developed, known as online auditing [1, 2], continuous control monitoring [3] and continuous auditing [4, 5]. Continuous auditing has been defined as "a methodology for issuing audit reports simultaneously with, or a short period of time after, the occurrence of the relevant events" [6]. The purpose of these approaches is to provide continuous assurance [7]. See Chiu et al [8] for a recent survey.

The general idea is that continuous assurance, i.e. assurance over continuous streams of data, can by attained by combining internal continuous monitoring by management with continuous auditing of data streams and the effectiveness of internal controls by an external auditor. For example, IIA lists the following slogan [9]:

Continuous assurance = Continuous Monitoring + Continuous auditing

However, according to our literature review [10], the final step to continuous assurance has been largely left unexplained. It is not explained *how* the results of continuous monitoring and continuous auditing need to be combined to provide assurance, i.e. confidence or certainty. Consequently, to our knowledge, no audit firm currently offers a continuous assurance service to their clients, under that name.

In this paper we would like to investigate this mismatch. What does the term assurance really mean? How are continuous monitoring and continuous auditing operationalized in practice, and what are the tasks and roles associated with it? Based on this analysis, we will then propose an information architecture and a procedure, which will provide a service called 'assurance on demand'.

This research is limited to the IT audit domain. We envision a continuous auditing platform that provides assurance over the reliability of an IT environment. For other application domains, such as compliance or financial reporting, it could be different. As we argue in Section 2, the purpose of providing assurance is to inspire trust in the users of a system. That leads to the following research question:

How can continuous assurance be provided on the reliability of an IT environment, so that it will lead to increased trust among its users?

The research takes the form of design science [11, 12]: we investigate an artefact (architecture and procedure) in a specific context, in particular the IT audit department of an accounting firm, in this case Deloitte. Based on extensive interviews with auditing experts and potential clients, and desk research of the relevant standards and guidelines, we have identified objectives, and derived system requirements. We developed an architecture and a procedure that follows the currently applicable IT audit standards as much as possible. The proposal is motivated by a case study of a platform for automatically auditing the security of mobile healthcare (mHealth) applications. In the context of this case study, we have also evaluated technical and legal feasibility of the service. Economic viability has not been investigated.

The remainder of the paper is structured as follows. Section 2 details the literature review, explains the problem and analyses the notions of assurance and trust. That leads to a list of objectives and requirements. Section 3 specifies an architecture and a corresponding workflow procedure. The proposal is tested against the objectives. Section 4 discusses the case study. The paper ends with a discussion of future research.

2 Theoretical background

2.1 Literature review

We have performed a literature review about continuous auditing, continuous monitoring and continuous assurance. For more details we refer to [10]. First we searched databases like Scopus and Google Scholar with different combinations of keywords like 'continuous assurance', 'continuous audit', 'implementing', 'in practice', etc., and collecting all the papers that appeared relevant on the basis of the title. Next, existing literature review articles, like Brown et al. [13], Kuhn and Sutton [5] and Chiu et al. [8] were studied to find more articles that the web search might have missed. Here is a sample of the papers found: Alles et al. [14], Byrnes et al. [15, 16], Hardy [17, 18], Hardy and Laslett [19], Kogan et al. [20], Shin et al. [21]; Singh and Best [22], Singh et al. [23]; Vasarhelyi et al. [24]. All these papers do mention the concepts of continuous monitoring, continuous control monitoring, continuous

auditing, or continuous data assurance, as it is sometimes called. In addition, some papers also mention continuous risk monitoring and assessment, which is used to dynamically measure risk and provide input for audit planning. Verifying the adequacy of the risk management function can also be included under continuous auditing.

The typical exposition makes use of a diagram as in Figure 1, see e.g. [9]. Here, continuous assurance is shown to be composed of continuous verification of properties of data (on the right), by means of continuous application of computational auditing techniques, and by continuously monitoring the reliability of the data (on the left). Reliability of the data is ensured by a system of internal controls (implemented by management), that must have been shown to be adequately designed (audit testing of CM), and must be shown to be operationally effective. The continuous monitoring parts therefore ensures the continued operating effectiveness.

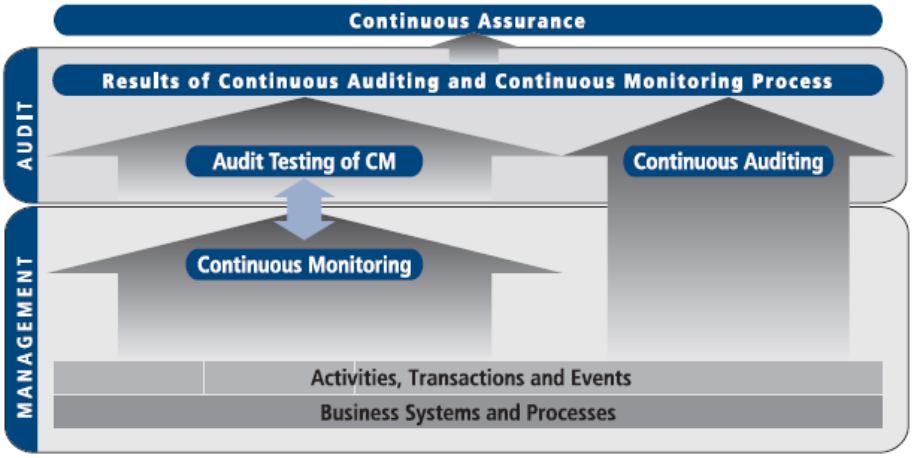

Figure 1. Typical conceptual model for continuous assurance [9], p 10.

However, our literature review indicates that beyond this conceptual model, the final step to continuous assurance has been largely left unexplained [10]. In particular, it is not explained *how* the results of the two 'pillars' need to be combined. Consequently, to our knowledge, no audit firm currently offers a continuous assurance service to their clients, under that name.

2.2 Assurance

One of the reasons that no continuous assurance service is offered under that name, could be that the term 'assurance' is loaded with connotations. The word assurance is often used in English to refer to the service provided by external auditors. It is what distinguishes accountants from consultants. Therefore, currently a service may only be called 'assurance' when it follows the standards and procedures for a full audit engagement, such as for instance those issued by the International Auditing and Assurance Standards Board [25]. For instance, they define an assurance engagement as

"... an engagement in which a practitioner expresses a conclusion designed to enhance the degree of confidence of the intended users other than the responsible party about the outcome of the evaluation or measurement of a subject matter against criteria" [25] p.6. An assurance engagement involves three separate parties: a practitioner (here: a qualified IT auditor), a responsible party (here: IT service provider) and intended users (here: clients of the IT service provider and indirectly also end-users). In IT auditing, these standards also mean that an 'assurance engagement' is typically distinguished from 'agreed upon procedures' or from a systems review. An assurance report aims for 'reasonable assurance', whereas a review may only provide 'limited assurance' [26], art. 14-16. An assurance report requires more effort and care in assessing the evidence and making the judgement, than a mere review. Typically, an assurance report is therefore also more expensive.

Moreover, we observe that in corporate practice, the term assurance is often used in a more limited sense, to refer to a written report signed by a qualified auditor. Such an attestation can be used to fulfill a role in a legal procedure, or in other circumstances that demand official proof. For example, under 'prudential supervision' a bank will need an assurance report of the reliability of IT services that have been outsourced, to demonstrate to the regulator that these outsourced services are 'in control'. For example, they could request an ISAE 3402 report, testifying reliability of the controls against the SOC2 criteria. Such a report is needed for compliance reasons, even when the outsourcing party is trusted.

Moreover, current standards prescribe that "The assurance report shall be in writing and shall contain a clear expression of the practitioner's conclusion about the subject matter information." [26], art 67. The reason is that a conclusion is usually qualified by certain assumptions and conditions. These may not be separated from the conclusion.

Clearly, demanding a written report is contrary to the vision of continuous assurance, which is automated. In that sense, continuous assurance can never be obtained, as it would require a constant stream of written reports, which is silly.

2.3 Trust

The word assurance is also used in relation to trust or confidence. In particular, Limperg famously formulated the theory of inspired confidence (in Dutch: leer van het gewekte vertrouwen), [27], see also [28] p 23. It is the function of the accountant to create trust among members of the general public. In publications about continuous assurance, the word assurance is often linked to the information needs of management, see e.g. Chiu et al [8]. After all, the fast pace of modern life demands information updates at or near real time. In particular, management needs to be able to rely on (trust) the continued accuracy and completeness of the financial results. Assurance by a trusted third party, an auditor, should create trust. Can trust also be inspired by an automated system?

There has been a lot of research on the notion of trust, in various disciplines [29]. In the economics literature trust is seen as "the willingness of a party to be vulnerable to the actions of another party based on the expectation that the other will perform a particular action important to the trustor, irrespective of the ability to control or monitor that party" [30] p 712. Compare also [31]. Typically, a person (the trustor) will trust

another person (trustee), to do an action. What about trust in systems? Here trustworthiness becomes more like reliability, dependability, or predictability. We trust a coffee machine to give us coffee when we insert a coin: we expect certain behavior, because we roughly know how a coffee machine works. In the same sense we can talk about trust in systems or software. Systems are considered reliable when they are predictable [32]. In this sense trust has been studied in the context of e-commerce [33]. Why would a buyer trust a seller mediated through a website?

We apply an e-commerce trust model from Tan and Thoen [34]. The model is originally about transactions: can a buyer or seller trust the other party to deliver or pay? In their model, the notion of transaction trust combines two notions, which are more fundamental: trust in the other party (party trust), and trust in control mechanisms that are built into the system or mechanism (control trust). Control trust is based on knowledge of how the controls work. Party trust involves personal trust, but it can be replaced by trust in an institution, compare [35]. Often, institutional trust is related to reputation. So someone will trade on eBay, when they trust eBay (party trust) and understand the guarantees built into the payment mechanism (control trust).

Here is the analogy of the e-commerce model to a continuous assurance service. According to the model, a user will trust the outcomes of a continuous auditing system, when (1) they trust the institution (accountants firm), and (2) they understand the tests and procedures that ensure reliability of those outcomes. The first part is common to all assurance reports: it must be written under responsibility of a qualified auditor, as we observed above. The audit profession is regulated. And although the reputation of the audit profession in general has been hit, individual auditors or firms can and do earn personal or institutional trust. The second part is about transparency and controls built into the audit process. Suppose that the system is set-up in such a way, that it is clear how a conclusion is derived, and that the effectiveness of this mechanism is also monitored on the dashboard itself. Then we expect that the behavior of the system will become predictable to its users, and therefore trustworthy. This expectation underlies the proposal in this paper.

3 Continuous auditing + assurance report on demand

In this section we specify an architecture and corresponding procedures to implement continuous auditing on the effectiveness of existing continuous monitoring procedures by management, and continuous auditing of the effectiveness of internal controls and of relevant properties of data streams, to attain continuous assurance. This entails a dashboard architecture that collects the results of the continuous auditing efforts, combined with the possibility to provide a written report on demand. The idea is that a user can trust (rely on) the effectiveness of the control settings and continuous data streams shown on the dashboard. That captures the trust-sense of assurance. However, when the user needs formal proof, a written assurance report can be generated automatically, based on the data collected in the dashboard. The report is generated according to all the standards and procedures of a traditional assurance engagement. In particular, the responsible engagement partner needs to sign. A workflow system

collects the data, summarizes it, stores it as evidence in a proper audit file, and requests the responsible people to sign off, using a qualified electronic signature. Estimates show that it is practically possible to generate such a report within 24 hours.

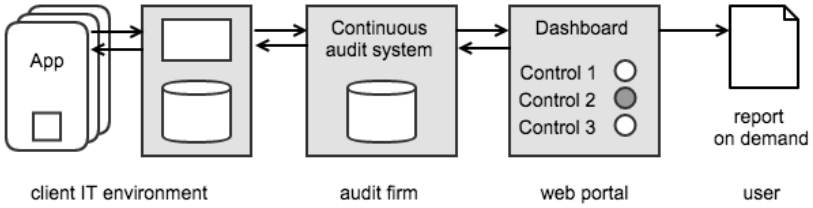

Figure 2. Dashboard showing the outcomes of continuously auditing a set of controls in the client's IT environment (left), by an audit firm, who can generate an assurance report when requested by the user (right).

3.1 Specifying a continuous assurance service

Based on this general set-up and the conceptual analysis of Section 2, we derive a list of requirements, for a service to be properly called continuous assurance. We state the following main objectives.

1. The assurance report on-demand system, combined with a continuous auditing system, should together deliver a form of continuous assurance over the reliability of an IT environment, in this case an mHealth application platform.
2. The assurance report on-demand service and IT architecture need to be adaptable to various application domains and IT environments.
3. The assurance service should have the potential to be economically viable. In other words, after a start-up period, clients are willing to pay for continuous assurance services, enough to sustain the investment and maintenance costs.

Based on the desk study and talks with experts, requirement 1 was further divided into three requirements called 'Continuous', 'Assurance' and 'On-demand'.

1.1 Continuous: monitoring updates on the dashboard are more frequent than the cycle time of the process that is being monitored.
1.2 Assurance: following international standards for assurance engagements [26], in particular:
1.2.1 - Conducted by a trustworthy and independent organization,
1.2.2 - Adequate and legally valid reasoning and reporting mechanism
1.2.3 - Based on sufficient appropriate evidence [36]
1.3 On demand: the workflow system should generate a report, providing assurance over the data of the continuous auditing dashboard, within 24 hours.

Requirement 1.1 is based on the definition of what it means to be continuous: "at or near real time" [7]. Note however that wat is considered 'real time' depends on the type of process being monitored. For example, an HR process, where employees typically enter or leave a firm by the 1st of the month, has a cycle time of one month. In that case

a monitoring frequency of once every two weeks is fast enough. By contrast, stock trading processes have cycle times of tenths of a second.

Requirement 1.2 is based in the IAASB standards [25, 26], but also reflects the Tan and Thoen theory of trust. In 1.2.1 we recognize the element of party trust or institutional trust; in 1.2.2 and 1.2.3 we see measures that should inspire control trust. Requirement 1.2.2 is a formal requirement that should cover the idea of an assurance report as a document, that fulfils a legal function. Requirement 1.2.3 is based on audit theory: any audit most be based on evidence that is appropriate (relevant to the decision to be made, legitimately obtained and reliable, i.e. accurate -- corresponds to reality -- and complete -- all relevant aspects of reality captured -- , and sufficient (enough, in the sense of a sample size) [36]. In automated auditing, tests usually cover the entire population, not samples, so the sufficiency requirement is achieved by definition.

Finally, requirement 1.3 is about the 'on demand' nature of the reporting service. It means in effect that, whenever a client demands a written assurance report, testifying the reliability of the data being shown on the dashboard and of the mechanism that produces that data, such a report could be produced within 24 hours. Obviously, this is only possible after a start-up period in which enough evidence is collected.

Based on these core requirements, a number of more detailed functional requirements, constraints, performance requirements and quality requirements were specified. We refer to [10] Ch 4 for more details. These requirements were iteratively evaluated and improved, with the aid of auditing experts from Deloitte, and an expert on the mHealth application. The final list of requirements was validated with these experts; they were considered relevant and appropriate.

Now we need to verify whether this set-up makes sense. Is such a design adequate to provide continuous assurance? In Section 2 we established that continuous assurance means assurance over the reliability of a continuous stream of data. Assurance by a third party is meant to inspire trust in its intended users. In addition, assurance can refer to a written report by a qualified auditor.

Proposition. A system that meets requirements 1.1 – 1.3 can in principle provide continuous assurance.

Here is a kind of proof. To provide continuous assurance the system must be able to (i) inspire trust in its users over the reliability of a continuous stream of data, and (ii) when demanded, provide a written report testifying reliability of that data.

Suppose the user has continued access to a dashboard with the outcomes of the continuous monitoring and the continuous auditing of the reliability of that data. The data shown on the dashboard is continuously being updated (1.1). The mechanism by which the data is produced is transparent, is verified to be reliable, and is itself being monitored. Usually, the mechanism remains stable. So barring external influences and changes, once trust has been established through party trust (1.2.1) and control trust (1.2.2, 1.2.3) it should continue. That covers condition (i).

Now suppose a client needs a written report, urgently. Through the workflow system the audit firm produces a written report within 24 hours (1.3). The time-scale at which the report is needed is based on external events such the news cycle, or regulatory

compliance, not on the cycle time of the process itself. We think that 24 hours is fast enough for such formal purposes. That covers condition (ii).

3.2 IT Architecture

In order for the continuous assurance scheme to work, it requires an IT architecture that can support the continuous auditing process, and automatically generate the assurance report. We propose an architecture consisting of two main components: the continuous audit system and a web portal that contains the dashboard and report generator.

All design choices are based on an extensive and iterative process of interviews with IT audit experts, security and privacy experts, and data analytics experts. In addition, all these systems need to go through quality review by the audit firm that needs to be properly documented, to ensure that they can be relied on for an assurance engagement.

3.3 Continuous audit system

The first component of the IT architecture we propose is the continuous audit system. This system needs to consist of several components, that together make it possible to automatically collect data from within a client's IT environment and store it within a secure environment. We distinguish a *continuous audit system server*, that runs the

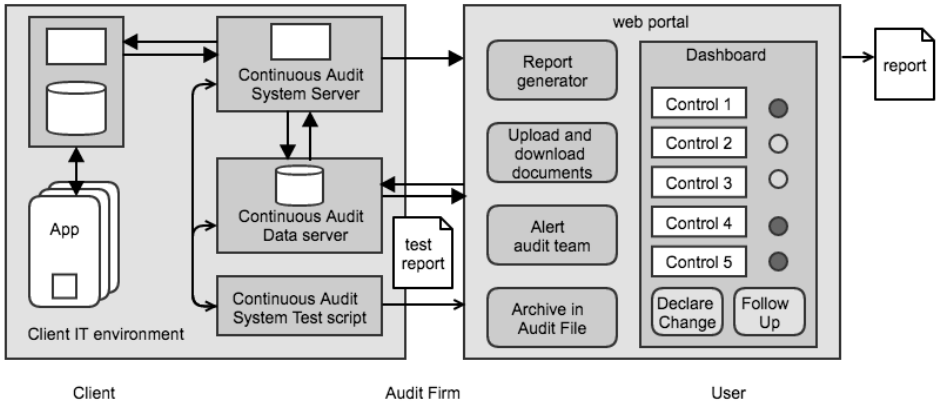

Figure 3 IT architecture

automated test protocols, and the *continuous audit data server*, that stores the resulting data. Furthermore, a *test script* is required that can automatically test whether or not the continuous audit system is functioning properly after it is set up within a client's IT environment. This test script is essential. Note that the use of test scripts is already common in regular IT audits. For example, scripts are routinely used to automatically collect log-in and authorization details of common operating systems.

Here are some essential characteristics of these components:

1. Continuous audit system server

- Secure server installed within the IT environment of the client.
- Has to be tailored to the client's system through APIs, and to the specific controls and data streams that need to be audited prior to installation.
- Connected to the network of the client with read-only access.
- Able to automatically collect audit data and send it to the audit firm through a secure VPN connection.

2. Continuous audit data server
   - Within a secure data center owned by the audit firm.
   - Stores client data in an environment separated from the audit firms own data and from the data of other continuous assurance clients.
3. Continuous audit system test script
   - Able to automatically test proper functioning of the continuous audit system server and its connection to the data server after it is installed at a client.
   - Able to generate a report that reports on the functioning of the system within a specific client's IT environment.
   - If the result is negative, the system needs to be changed and the script run again. This process needs to be repeated until the result is positive.

3.4 Web-portal and dashboard

The second part of the system needs to be able to store client specific information, host the dashboard functionality that is made available to the client and automatically import the continuous audit data into the audit file and generate an assurance report based on this data. We propose to host all these functionalities within a web-portal, that is operated by the audit firm. The web-portal needs to have the following functionality:

Web-portal
- Allows the audit team to create an engagement file for each assurance engagement.
- Allows the audit team to enter the required client information.
- Allows the audit team to view the status of continuous audit controls being tested for each client (i.e., whether the controls are operating effectively).
- Allows the audit team to download the documents required to generate an ISAE 3000 assurance report (e.g., Letter of Representation, fraud inquiry form, change inquiry form), which a client needs to upload after they request an assurance report.
- If documents are satisfactory, audit team members can send e-mail notification that an assurance report has been requested to the rest of the audit team.

The components that are hosted within this web-portal are as follows:

Report generator
- Able to automatically import the continuous audit data from the data server and process and archive it in a client-specific audit file.
- Able to automatically generate an assurance report based on continuous audit data.

Dashboard

- Provides client with an overview of the status of the controls that are being tested by the continuous audit system.
- Allows the client to declare a change to one of their controls, so that the continuous audit system can be adapted to accommodate the change.
- Allows the client to request an assurance report
- Allows the client to upload the documents required to generate an ISAE 3000 assurance report.
- When a control becomes not-effective, the dashboard allows the client to provide the audit firm with required information and follow-up procedures.

Change management is crucial in a set-up like this. A change inquiry form must be filled in by the client to record any changes in the design or implementation of controls. This form is based on a standard subsequent events form, subsequent events are about changes between the end of the testing period and the sending of the report. For continuous assurance the testing period goes right up to the moment the client requests a report, so there is no time in between testing and creating the report. However, time did pass between the testing of design and implementation and the creation of the report, so it needs to be verified that nothing was changed.

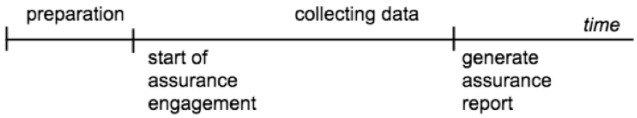

Figure 4. Time line of a continuous assurance engagement

3.5 Procedures

To attain continuous assurance, we propose a workflow process that utilizes the IT architecture described above to enable a client to request an assurance report at any time, which meets the standards of an ISAE 3000 assurance engagement. The process consists of three main phases: preparation, start of the assurance engagement and generating the assurance report (Figure 4). This last phase is re-run every time the client requests an assurance report. In the preparation phase, an initial Type I audit needs to be performed according to the ISAE 3000 process. This involves an audit on the adequacy of the design and implementation of the controls that are in scope. The continuous auditing provides the data to establish operating effectiveness of the controls, over a longer period of time. The phases contain the following steps:

1. Preparation
   - A control framework is created and Type I audit is performed
   - The continuous audit system is tailored to the client's systems and installed within their IT environment.
   - The test script is run and the test report is stored in the audit file. If necessary, the system must be changed and this step repeated until the result is positive.
2. Start of the assurance engagement

- Client information entered into the web-portal and dashboard functionality is made available to the client.
- A template for the assurance report is created for the client.
- A partner reviews the generic parts of the assurance report, i.e., those parts that will be the same every time the client requests an assurance report.

3. Generating the assurance report
   - If the client requests an assurance report, the system automatically sends them the documents necessary to be filled in for an ISAE 3000 assurance report.
   - Client fills out and signs the documents, and uploads them into the dashboard. The audit team is alerted that the documents have been uploaded.
   - The first audit team member who is able to, will review the documents. If they are satisfactory, the rest of the team is alerted that the process has been started.
   - The audit team uses the web-portal functionality to import the continuous audit data from the data server, store it in an audit file, and generates an assurance report based on this data.
   - A partner reviews those parts of the audit file and of the assurance report that have been automatically imported into the template.
   - The partner signs the assurance report and it is sent to the client.

4 Case: Monitoring security of Healthcare Apps

The proposal is motivated by a case study of a platform that allows continuous monitoring of the security of a set of mobile healthcare applications (mHealth Apps). Healthcare Apps provide a relevant case, as health-related data is considered sensitive, for instance according to EU Directive 95/46/EC, the General Data Protection Regulation [37], or the HL7 standard for Healthcare. In addition to legal constraints, it is likely that end-uses are themselves worried about integrity and confidentiality of their data, in particular, as changes take place to the data or to the App.

The consultancy branch of Deloitte in the Netherlands has developed a software platform for continuously monitoring the security settings of mobile applications, which are in turn developed by commercial software providers. For example, the platform monitors for each user the telephone's App Store for fake copies. It also makes sure that the unique hash code of a certified version of the App corresponds to the code of the version that is installed. Input and output streams are encrypted, and hashed to verify data integrity, etc. The platform has been running successfully in a laboratory setting. The challenge is to develop a viable assurance service on the basis of this continuous monitoring platform.

The architecture and procedures of Section 3 have been instantiated for the settings in the case study. The resulting design has been evaluated in interviews with audit experts, experts in information security, and experts in healthcare Apps. The design has also been discussed with representatives of hospitals, being the clients of the software provider building the mHealth Apps. In total 17 people were interviewed.

The interviews show that the proposed architecture is technically feasible. Many of the modules and software components are already available. A dashboard similar to the

one suggested here, is already in use in another application. A report generator and workflow system that collects evidence, can be built on short notice; most components are already available. For instance, the filing system that is used for storing and retrieving audit files, can also be used for this automated service. That suggests that building the system for a non-trivial case situated in healthcare, is technically feasible.

The proposed scheme is also legally or professionally feasible, in the sense that the resulting report would comply with relevant standards and procedures (ISAE 3000). The usage of an automated test script is common practice. Evidence about the adequacy of the design and implementation (Type I assessment) and evidence of operating effectiveness over a longer period of time (from the continuous auditing system) are stored in proper audit files. The report is reviewed and signed by a partner, based on the same requirements about evidence as in a regular IT audit. The general set-up was developed by the Deloitte Risk Advisory and validated – in the context of this pilot project – by experts in professional standards. Obviously this is only a pilot project and doesn't reflect the official policies of Deloitte. What the pilot shows is that a scheme like this could work, even in the context of a Big Four audit frim.

The economic feasibility, i.e. the viability of the business model of offering such a continuous assurance service, has not been evaluated properly. However, based on interviews with the software developer, who knows the market, and his clients, namely representatives of hospitals, we can make some observations. The interviews show that, first of all, the hospitals have a need for assurance over the security of healthcare Apps, both in the sense of reliability over streams of data and in the sense of a formal report. In particular, they recognize that security is an issue for their patients and that they have to be able to demonstrate that they are compliant. However, what has not been recognized is the need for assurance at or near real time. It is possible that security of healthcare Apps is not time-critical, in that sense.

5 Conclusions

In this paper we have investigated how to derive continuous assurance from continuous monitoring (by management) and continuous auditing (by an audit firm). We have analyzed the notion of assurance and related it to a model of trust for e-commerce [34]. It turns out the term assurance is typically used in two senses: as a way of inspiring trust or confidence, and as a formal report signed by a qualified auditor.

Based on this analysis we propose an architecture and a procedure, to provide a continuous assurance service. The user has continued access to a dashboard, which shows the results of the continuous auditing system. As the environment changes, the continuous audit results will change too, so that should inspire trust in the user: the system behaves in a predictable manner. These audit results are produced by a stable mechanism, that is verified automatically. This should create control trust. Moreover, the system has been set-up and verified by an audit firm. Audit firms are regulated by a professional body and a regulator. That should inspire party trust. Finally, whenever a user needs a formal report, the system starts a workflow procedure to generate an assurance report, and have it reviewed and signed by the responsible partner.

Based on a case study of a security platform for mobile Health Apps, we can say that the design is technically feasible. All components are available or can be built in short notice. The design is also legally or procedurally feasible, in the sense that it follows the same standards and procedures as other IT audits (ISAE 3000). That means, that the result may really be called assurance. We have not evaluated economic feasibility. However, interviews indicate that there is a need for assurance over security of mobile healthcare Apps. However, it is not clear whether customers would be willing to pay specifically for assurance, at or near real time.

This paper describes the outcomes of a pilot project conducted at Deloitte Netherlands. Results are hard to generalize. First, the pilot has not been finished. The software needs to be completed and the report generation process needs to be tested. Second, the research is set in the context of an IT audit assignment, which could be easier to automate than, say, a financial audit or compliance audit. Third, even when formal procedures are followed and the report may technically be called 'assurance', there could be other reasons for an audit firm not to develop this idea further. In particular, we can imagine that it is undesirable that an automated assurance service would compete with the usual human assurance services. This proposal should therefore be interpreted in the context of a debate about the revenue model of accounting firms. The current revenue model is based on billable hours. The continuous assurance model proposed here, would suggest a kind of subscription service, where the user would get assurance for a fixed fee per year.

Clearly, this topic calls for more research. In particular, we welcome research about future revenue models of audit firms. We also want to raise the fundamental question whether it is always possible or even desirable to automate the auditor's judgement and decision making, or to shift the main judgements to design time. In the past, there has been a lot of research about judgement and decision making in accounting [38], but not in the context of IT audits.

References

[1] H. S. Koch, "Online Computer Auditing through Continuous and Intermittent Simulation," *MIS Quarterly,* vol. 5, no. 1, pp. 29 - 41, 1981.

[2] M. A. Vasarhelyi, and F. B. Halper, "The Continuous Audit of Online Systems," *Auditing: A Journal of Practice and Theory,* vol. 10, no. 1, pp. 110-125, 1991.

[3] M. Alles, G. Brennan, A. Kogan, and M. A. Vasarhelyi, "Continuous monitoring of business process controls: A pilot implementation of a continuous auditing system at Siemens," *International Journal of Accounting Information Systems,* vol. 7, pp. 137-161, 2006.

[4] A. Kogan, E. F. Sudit, and M. Vasarhelyi, "Continuous online auditing: a program of research," *Journal of Information Systems,* vol. 13, no. 2, pp. 87–103, 1999.

[5] J. R. Kuhn, and S. G. Sutton, "Continuous Auditing in ERP System Environments: The Current State and Future Directions " *Journal of Information Systems* vol. 24, no. 1, pp. 91–111, 2010.

[6] CICA/AICPA, *Continuous auditing, Research report*, The Canadian Institute of Chartered Accountants (CICA), Toronto, Canada, 1999.

[7] M. A. Vasarhelyi, M. Alles, and A. Kogan, “Principles of analytic monitoring for continuous assurance,” *Journal of Emerging Technologies in Accounting,* vol. 1, no. 1, pp. 1-21, 2004.

[8] V. Chiu, Q. Liu, and M. A. Vasarhelyi, “The development and intellectual structure of continuous auditing research,” *Journal of Accounting Literature,* vol. 33, pp. 37–57, 2014.

[9] D. Coderre, *Continuous Auditing: Implications for Assurance, Monitoring and Risk Assessment, Guide 3*, Institute of Internal Auditors (IIA), 2005.

[10] J. Kocken, “Implementing Continuous Assurance within IT Service Organizations,” Technology, Policy and Management Delft University of Technology, Delft, 2016.

[11] A. R. Hevner, S. Ram, and S. T. March, “Design Science in Information Systems Research,” *Mangement Information Systems Quarterly,* vol. 28, no. 1, pp. 75-105, 2004.

[12] R. J. Wieringa, *Design science methodology for information systems and software engineering*, London: Springer Verlag 2014.

[13] C. E. Brown, J. A. Wong, and A. A. Baldwin, “A review and analysis of the existing research streams in continuous auditing,” *Journal of Emerging Technologies in Accounting,* vol. 4, no. 1, pp. 1-28, 2007.

[14] M. Alles, A. Kogan, and M. Vasarhelyi, “Putting Continuous Auditing Theory Into Practice: Lessons from Two Pilot Implementations,” *Journal of Information Systems,* vol. 22, no. 2, pp. 195-214, 2008.

[15] P. E. Byrnes, B. Ames, M. Vasarhelyi, and J. D. W. Jr., *The Current State of Continuous Auditing and Continuous Monitoring*, American Institute of Certified Public Accountants (AICPA), New York, 2012.

[16] P. Byrnes, T. Criste, T. Stewart, and M. Vasarhelyi, *Reimagining Auditing in a Wired World*, American Institute of Certified Public Accountants (AICPA). New York, 2014.

[17] C. A. Hardy, "Exploring Continuous Assurance In Practice: Preliminary Insights." p. paper 74.

[18] C. A. Hardy, “The Messy Matters of Continuous Assurance: Findings from Exploratory Research in Australia,” *Journal of Information Systems,* vol. 28, no. 2, pp. 357-377, 2014.

[19] C. A. Hardy, and G. Laslett, “Continuous auditing and monitoring in practice: Lessons from Metcash's Business Assurance Group,” *Journal of Information Systems,* vol. 29, no. 2, pp. 183-194., 2014.

[20] A. Kogan, M. G. Alles, and M. A. Vasarhelyi, “Design and evaluation of a continuous data level auditing system.,” *Auditing: A Journal of Practice and Theory,* vol. 33, no. 4, pp. 221-245, 2014.

[21] I. h. Shin, M. g. Lee, and W. Park, “Implementation of the continuous auditing system in the ERP-based environment,” *Managerial Auditing Journal,* vol. 28, no. 7, pp. 592-627, 2013.

[22] K. Singh, and P. J. Best, “Design and Implementation of Continuous Monitoring and Auditing in SAP Enterprise Resource Planning,” *International Journal of Auditing,* vol. 19, no. 3, pp. 307-317, 2015.

[23] K. Singh, P. J. Best, M. Bojilov, and C. Blunt, “Continuous Auditing and Continuous Monitoring in ERP Environments: Case Studies of Application Implementations,” *Journal of Information Systems,* vol. 28, no. 1, pp. 287–310, 2014.

[24] M. A. Vasarhelyi, M. Alles, and K. T. Williams, *Continuous Assurance for the Now Economy*, Institute of Chartered Accountants in Australia, Sydney, Australia, 2010.

[25] IAASB, *International Framework for Assurance Engagements*, International Auditing and Assurance Standards Board (IAASB), New York, 2005.

[26] IAASB, *ISAE 3000 (Revised): Assurance Engagements Other than Audits or Reviews of Historical Financial Information.* , IFAC, 2015.

[27] T. Limperg, “De functie van den accountant en de leer van het gewekte vertrouwen,” *Maandblad voor Accountancy en Bedrijfseconomie,* vol. 1932-2, 1932-9, 1933-9, 1933-10, 1932-1933.

[28] J. H. Blokdijk, F. Drieënhuizen, and P. H. Wallage, *Reflections on auditing theory, a contribution from the Netherlands*, Amsterdam: Limperg Instituut, 1995.

[29] R. Falcone, and C. Castelfranchi, “Trust and relational capital,” *Computational and Mathematical Organization Theory,* vol. 17, no. 2, pp. 179-195, 2011.

[30] R. C. Mayer, J. H. Davis, and F. D. Schoorman, “An integrative model of organizational trust,” *Academy of Management Review,* vol. 20, no. 3, pp. 709-734, 1995.

[31] D. G. Gambetta, "Can we trust trust?," *Trust*, D. G. Gambetta, ed., pp. 213-237, New York: Basil Blackwell, 1988.

[32] A. van Lamsweerde, "Engineering requirements for system reliability and security. ," *Software system reliability and security*, J. G. M. Broy and C. Hoare, eds., pp. 196–238: IOS Press, 2007.

[33] D. Gefen, “Reflections on the Dimensions of Trust and Trustworthiness among Online Consumers,” *ACM SIGMIS Database* vol. 33, no. 3, pp. 38-53, 2002.

[34] Y.-H. Tan, and W. Thoen, “Formal aspects of a generic model of trust for electronic commerce,” *Decision Support Systems,* vol. 33, no. 3, pp. 233-246, 2002.

[35] L. G. Zucker, "Production of trust: Institutional sources of economic structure, 1840-1920," *Research in Organizational Behavior*, B. M. Staw and L. Cummings, eds., pp. 53-111: JAI Press, Greenwich, CT, 1986.

[36] IFAC, "ISA 500: Audit Evidence," International Auditing and Assurance Standards Board (IAASB), International Federation of Accountants (IFAC), 2012.

[37] *Regulation (EU) 2016/679*, *General Data Protection Regulation* 2016.

[38] S. Bonner, “Judgment and Decision-Making Research in Accounting,” *Accounting Horizons,* vol. 13, no. 4, pp. 385-398., 1999.