Tools for Cloud Accountability: A4Cloud Tutorial

Carmen Fernandez-Gago, Vasilis Tountopoulos, Simone Fischer-Hübner, Rehab Alnemr, David Nuñez, Julio Angulo, Tobias Pulls, Theo Koulouris

Tools for Cloud Accountability: A4Cloud Tutorial*

Carmen Fernandez-Gago[1], Vasilis Tountopoulos[2], Simone Fischer-Hübner[3], Rehab Alnemr[4], David Nuñez[1], Julio Angulo[3], Tobias Pulls[3], and Theo Koulouris[4]

[1] Network, Information and Computer Security Lab
University of Malaga, 29071 Malaga, Spain
{mcgago,dnunez}@lcc.uma.es
[2] Athens Technology Center S.A., Athens, Greece
v.tountopoulos@atc.gr
[3] Karlstad University, Sweden
{simone.fischer-huebner,julio.angulo,tobias.pulls}@kau.se
[4] HP Labs, Bristol, UK
{rehab.alnemr,theofrastos.koulouris}@hp.com

Abstract. Cloud computing is becoming a key IT infrastructure technology being adopted progressively by companies and users. Still, there are issues and uncertainties surrounding its adoption, such as security and how users' data is dealt with that require attention from developers, researchers, providers and users. The A4Cloud project tries to help solving the problem of accountability in the cloud by providing tools that support the process of achieving accountability. This paper presents the contents of the first A4Cloud tutorial. These contents include basic concepts and tools developed within the project. In particular, we will review how metrics can aid the accountability process and some of the tools that the A4Cloud project will produce such as the Data Track Tool (DTT) and the Cloud Offering Advisory Tool (COAT).

1 Introduction

Cloud computing is an evolving technology that is adopted progressively by companies and users creating a vast market. Still, there are issues and uncertainties surrounding its adoption, such as security and how users data is dealt with that require attention from developers, researchers, providers and users. It is essential that there are tools and mechanisms available that can help providing trust in the cloud. According to the definition provided by the A4Cloud project [3], *Accountability* consists of defining governance to comply in a responsible manner with internal and external criteria, ensuring implementation of appropriate actions, explaining and justifying those actions and remedying any failure to act properly. The A4Cloud project will then provide the tools and mechanisms needed in order to achieve accountability for cloud providers and users. The development of these tools come first from a conceptual level to go then into the development level. In this paper we will describe the problem of accountability and how the A4Cloud

* This work has been partially funded by the European Commission through the FP7/2007-2013 project A4Cloud under grant agreement number 317550. The fifth author is funded by a FPI fellowship from the Junta de Andalucía through the project PISCIS (P10-TIC-06334).

project is addressing it. In particular, it will be very useful to have mechanisms that determine in a quantitative or qualitative way how transparent a service provider is. Thus, defining metrics can be useful for determining accountability. In order to elicit these metrics, we introduce the process that we follow. This process consists of a top-down approach for the identification of concepts to measure and a bottom-up approach that serves as a way to provide evidence, based on existing controls [1, 2, 5].

The mechanisms that the A4Cloud project introduces are implemented through a set of tools that are introduced in this paper. These tools cover different aspects that include regulatory aspects, socio-economical or legal aspects. In this paper we will concentrate on two specific tools within the toolset of A4Cloud: the Data Track Tool (DTT) and the Cloud Offering Advisory Tool (COAT). The DTT aims to provide information to the users about how their personal data is dealt with. The COAT tool helps users deciding about the best cloud service provider to use by reconciling the users' requirements on transparency, legal terms, privacy or security with those offered by the providers.

The structure of the paper is as follows. In Section 2 we introduce the problem of accountability and how the project A4Cloud can help solving it. Thus, Section 3 describes how defining metrics can be useful for aiding achieving accountability. Section 4 gives a general overview on the A4Cloud tools and the following sections describe two of them. In particular, Section 5 describes the Data Track Tool (DTT) and Section 6 the Cloud Offering Advisory Tool (COAT). Finally, Section 7 concludes the paper and outlines the future research within A4Cloud.

2 The Objectives of the A4Cloud Project

A4Cloud's goal, among others, is to understand what users need to trust a cloud provider with their personal data. A4Cloud focuses on the *accountability for cloud and other future internet services* as the most critical prerequisite for effective governance and control of corporate and private data processed by cloud-based IT services. The project goal is to increase trust in cloud computing by devising methods and tools, through which cloud stakeholders can be made accountable for the privacy and confidentiality of information held in the cloud. These methods and tools will combine risk analysis, policy enforcement, monitoring and compliance auditing. They will contribute to the governance of cloud activities, providing transparency and assisting legal, regulatory and socio-economic policy enforcement. The A4Cloud project has four interlocking objectives to bring users, providers, and regulators together in chains of accountability for data in the cloud, clarifying liability and providing greater transparency overall to [5]:

1. Enable cloud service providers to give their users appropriate control and transparency over how their data is used.
2. Enable users to make choices about how cloud service providers may use and will protect data in the cloud.
3. Monitor and check compliance with users' expectations enforce business policies and regulations.
4. Implement accountability ethically and effectively.

[5] The description is taken from the official documentation of the project

3 Accountability Metrics

One of the important aspects behind the accountability concept is the ability of an organization to demonstrate their conformity with required obligations [4]. The concept of Accountability goes beyond behaving in a responsible manner, and deals also with showing compliance and providing transparency to the internal process of accountability provision. One of the goals of the A4Cloud project is the demonstration of this through the measurement of the degree of such conformity and the provision of meaningful evidence. Thus, measurement becomes an important tool for assessing the accountability of an organization by external authorities (and organizations themselves, in the case of self-assessment).

If we are interested in assessing how accountable an organisation is we should provide techniques for measuring the attributes that influence accountability. How much or to what extent they should be measured is a key issue. One of the goals of A4Cloud is to develop a collection of metrics for performing meaningful measures on the attributes that influence accountability.

3.1 The Role of Metrics in Accountability

Measured serviceT is include in the definition of cloud computing given by NIST [12] as one of its main characteristics. This characteristic is defined as the capacity of cloud systems for measuring aspects related to the utilization of services, in order to provide automatic control and optimization of the usage of cloud resources, and ultimately, to support transparency and enhance trust of cloud users with regard to cloud providers. Metrics in cloud computing environments are also of paramount importance for other reasons. For instance, metrics can also be derived on the consumer side, enabling cloud users to monitor the quality of service of the cloud provider and to verify the compliance of agreed terms. Metrics are also a tool that facilitate the decision making process of cloud consumer organizations, as they can be used for making informed decisions with regard to the election and evaluation of cloud providers.

As for cloud service governance, metrics are very useful means for assessing performance of operational processes and for demonstrating the implementation of appropriate practices through the provision of quantifiable evidence of the application of such practices. Metrics also support accountability governance and can be used as an instrument for identifying strengths and weaknesses in the security and privacy mechanisms in place. From the perspective of the accountability framework, metrics are a means for demonstrating accountability, through the provision of quantifiable evidence of the application of proper practices and the performance of operational processes. This way, progress in the implementation of accountability practices can be justified in a quantitative way.

3.2 Eliciting Metrics for Accountability

In order to measure the accountability attributes we need to have a clear target of the aspects of the attributes that are to be measured. The definitions of the attributes are in some cases vague, subjective or ambiguous, thus it is difficult to measure specific

aspects. We need a suitable model that allows us to identify measurable factors from the definitions of the attributes. Once these specific factors are identified we need to derive metrics for them based on the analysis of existing control frameworks. Thus, the process of eliciting accountability metrics consists of two complementary approaches:

- A top-down approach. This approach is based on the definition of a Metamodel for Accountability Metrics, to aid during the initial phases of the elicitation of metrics.
- A bottom-up approach. It is used for complementing the previous one, based on the analysis of relevant control frameworks.

Metrics Metamodel The goal of the metamodel for eliciting accountability metrics [13] (see Figure 1) is to serve as a language for describing accountability attributes (property in the figure) in terms of entities, evidence and actions, and metrics for measuring them. In this metamodel, metrics are defined in two kinds of central inputs: evidence and criteria. We claim that any assessment or evaluation (i.e, a metric) can only be made using as input some tangible and empirical evidence, such as an observation, a system log, a certification asserted by a trusted party, a textual description of a procedure, etc. That is, a metric does not directly measure a property of a process, behaviour, or a system, but uses the evidence associated with them in order to derive a meaningful measure. On the other hand, criteria are all the elements that convey contextual input that may constrain what should be measured, such as stakeholder's preferences, regulations and policies.

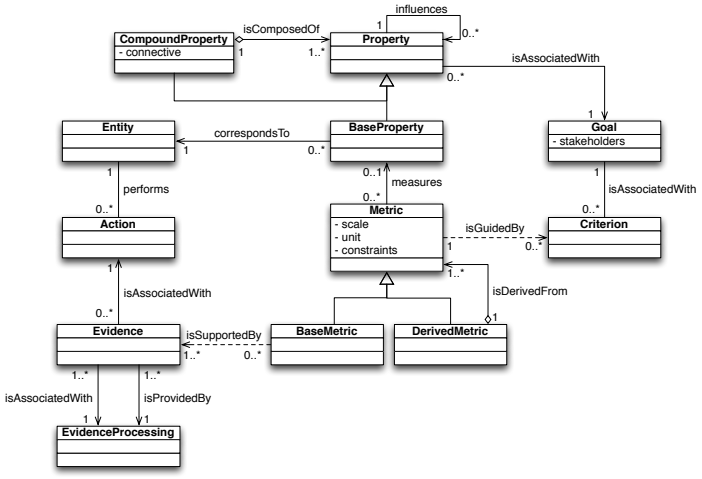

Fig. 1. Metrics Metamodel

This top-down approach is useful for reasoning about high-level concepts such as Accountability, however, it does not guarantee to reach measurable concepts. Actually, the value of the proposed metrics metamodel lays principally in aiding to correctly

identify and specify the subconcepts that are relevant or influence the Accountability Attributes, rather than being a method for extracting relevant metrics. For this reason, we need a complementary strategy.

Besides evidence and criteria the metamodel includes other elements such as *property*, which refers to the accountability attributes; *goal* that refers to a high-level description of the property that is modelled; *entity*, which is a physical or conceptual objects that performs an *action*; and *metric*, which is an evaluation method for assessing the level of satisfaction of a non-functional property in a quantitative or qualitative way, on the basis of evidence and contextual criteria.

Bottom-Up Approach Control frameworks that are relevant for accountability, such as the Cloud Control Matrix [2], the Generally Accepted Privacy Principles [1], and NIST SP 500-83 [5], are specifically designed for covering the categories of mechanisms that implement security, privacy and information governance. For this reason, it is fair to assume that they can be used as sources of evidence from where metrics can be derived. Thus, we can use the application of these frameworks for audit records as evidence for deriving metrics. The steps of the bottom-up approach are as follows:

1. To analyse relevant control frameworks in the light of Accountability Attributes. The goal of this step is to select those controls that influence Accountability.
2. To study the nature of the control, in order to identify whether there is any quantifiable element in the description of the control that is susceptible to being measured. Qualitative elements may be identified too, if they have at least an ordinal nature.
3. To define a metric that measures the identified elements, using the qualitative or quantitative elements identified in the previous step.
4. To check that the metric supports the concept of Accountability and, in particular, the Accountability Attribute to which is related to.

4 An Overview of the A4Cloud Tools

The A4Cloud project has developed a conceptual model for accountability in [9], which defines accountability attributes, practices and mechanisms and how they relate to each other. The accountability mechanisms incorporate legal, regulatory, socio-economic and technical approaches, which are integrated into a framework to support an accountability -based cloud approach to cloud data governance and are functionally classified into preventive, detective and corrective.

In this paper, we focus on the A4Cloud toolset, which provides implementations for these mechanisms. The tools comprising this toolset are designed considering the existing gaps in accountability practices, thus, they aim to implement those functions of the accountability mechanisms, for which little or no support was found to exist out there to complement current privacy and security mechanisms.

4.1 The Architecture of the A4Cloud Tools

The definition and the design principles of the toolset are based on the fact that each A4Cloud tool addresses different elements of accountability, and may operate over different time scales, while interacting with data at different stages of its life cycle. In that

respect, the tools implementing preventive mechanisms investigate the potential risks in cloud data governance in order to form policies and decide on relevant mechanisms that should be enacted. The tools implementing detective mechanisms put in place detection and traceability measures to monitor misbehaviours, such as policy violations, in the normal operation of cloud processes. Finally, the tools implementing corrective mechanisms provide notification and remediation, as a response to detected abnormalities of the cloud service chains.

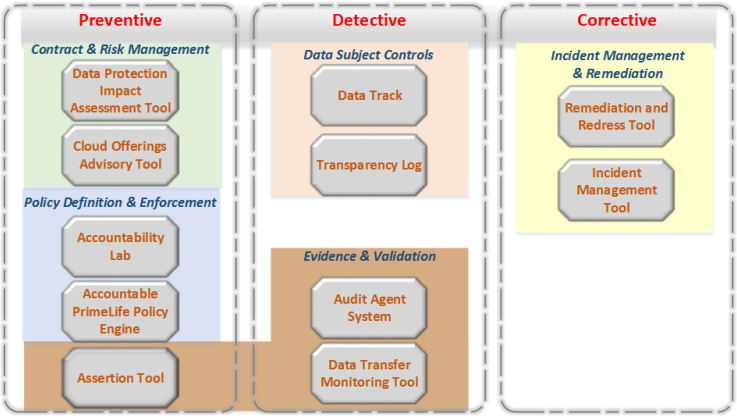

Fig. 2. The high level view of the A4Cloud Toolset Architecture

The A4Cloud toolset is composed of eleven tools, as shown in Figure 2. The tools can be further classified into five functional areas, according to the scope of each tool and the functions provided in the three phases of the accountability framework. These areas are analysed in the following lines.

The Contract and Risk Management area addresses the need for support in managing risks and cloud service contract selection in the context of accountability for classified data in the cloud. The respective tools serve a preventive role, which is realised through two complementary mechanisms. The first one has to do with the assessment of the risks associated with various facets of the cloud service consumption process, involving personal and/or confidential data and elicitation of actionable information and guidance on how to mitigate them, which is implemented through the Data Protection Impact Assessment Tool (DPIAT). The evaluation of cloud offerings and contract terms complements this mechanism, which is performed through the Cloud Offerings Advisory Tool (COAT), with the goal of enabling a more educated decision making on which service to select.

The Policy Definition and Enforcement area hosts two tools that supplement the tools in the previous area as preventive mechanisms to support accountability. In this category, we introduce the Accountability Lab (AccLab), as a tool, which translates human readable accountability obligations expressed in Abstract Accountability Language (AAL) [8] into an A4Cloud specific lower level machine-readable accountability

policy language, called Accountable PrimeLife Policy Language (A-PPL) language. On top of it, we provide the Accountable PrimeLife Policy Engine (A-PPL Engine), which enforces data handling policies and actions, as they are specified in A-PPL, which takes the form of a sticky policy that travels with the data downstream [8].

Moving to the implementation of the detective mechanisms, the Evidence and Validation category of tools offers accountability by implementing mechanisms for the monitoring of the appropriate software resources to control and verify the accountability policy-based operations occurred in complex cloud service provision chains. This is enabled through the Audit Agent System (AAS), which enables the automated audit of multi-tenant and multi-layer cloud applications and respective infrastructures for compliance with accountability policies, using software agents. Furthermore, we automate the collection of evidence, describing how data transfers comply with data handling policies within a cloud infrastructure through the Data Transfer Monitoring Tool (DTMT). In this category, we, also, include the Assertion Tool (AT) that ensures the validation of the A4Cloud tools through a test case-based methodology, during the development and deployment of accountability mechanisms.

In A4Cloud, we put particular emphasis on enabling individuals, whose personal data are collected and/or processed by cloud service providers, to take control over how these data are exploited along cloud service chains. To this direction, we introduce Data Track (DT), which is used by data subjects to get a user-friendly visualisation of all personal data they have disclosed to cloud service providers, with the additional capability to rectify data if necessary. DT embeds a Plug-in for Assessment of Policy Violation (PAPV) that provides an assessment on the criticality of previously detected policy violations. In order to secure the communication between these subjects and the cloud providers, the A4Cloud toolset offer the Transparency Log (TL), as a privacy-preserving channel to facilitate offline data exchange as well.

With respect to the implementation of corrective mechanisms, the architecture of the A4Cloud toolset introduces the Incident Management and Remediation functional area, which supports accountability through the Incident Management Tool (IMT) and the Remediation and Redress Tool (RRT). IMT generates notifications on detected anomalies and violations in cloud services, while RRT assists cloud customers in requesting appropriate remediation and implementing respective redress actions.

4.2 Tools Collaboration for Accountability Support

In this section, we describe the accountability information flow, depicting the tools dependencies and their interaction for implementing accountability along the three phases. Thus, the tools in the A4Cloud toolset generate accountability specific data objects, which are shared among them to accomplish the respective functions laid on the preventive, detective and corrective mechanisms. The flow of the accountability information among the tools is depicted in Figure 3, which is achieved in a semi-automatic way along the implementation of the accountability lifecycle processes.

As shown in Figure 3, the type of data that are collected from the data subjects drives the definition of specific accountability obligations identified for the respective cloud providers processing such data, which are analysed along with the privacy and security requirements of the end users and the organisational level policies for providing security

in their services, such as access control and encryption. This information is exploited by the tools of the Contract and Risk Management category to reduce the risks of the loss of data governance in complex cloud service provision chains. The outcome of this tool category is the impact assessment report, which elaborates on the privacy risks and the proposed mitigation for a cloud service process chain, based on risk and trust models, and the cloud offering report, analysing the privacy and security guarantees for given functional features offered by cloud providers.

Given the outcome of the previous category, AccLab is used to compile the obligations into A-PPL policies, setting the legal and technical conditions, under which a cloud service that involves the processing of personal and/or business confidential data is operating. The enforcement of these policies is handled by the A-PPL Engine, which generates logs with respect to performed data handling actions against the rules of the A-PPL policies.

These policies are used in the Evidence and Validation functional area to configure the detection mechanisms applied in a cloud service chain. The functions in this tool category exploit the logs produced by the external cloud resources, which are aggregated in the form of evidence records, and produce an incident referring to an abnormal behaviour of the cloud service chain with respect to the A-PPL policies. This tool category, also, enables the cloud providers demonstrating their compliance to the policies by generating audit reports, based on a collection of evidence.

The incidents are utilised by IMT to alert the cloud stakeholders about detected violations and formulate a set of corrective actions that could be undertaken in response to the occurred incidents, through RRT, which could take the form of simple remediation reports, as shown in Figure 3 or actually redress. In parallel, the A4Cloud toolset enables verification of the followed data handling processes by the cloud providers, through a set of tools used to control the cloud subjects data disclosure in the cloud.

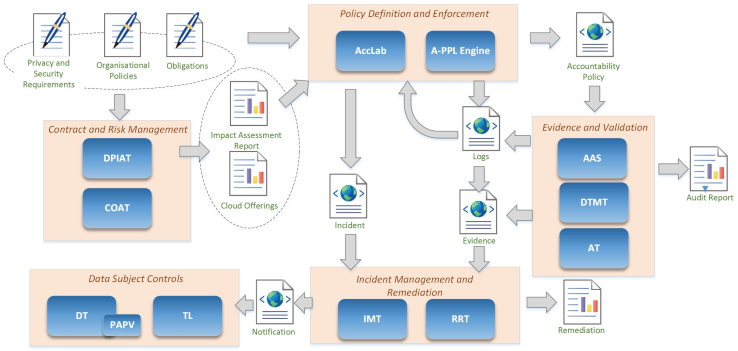

Fig. 3. The flow of the accountability information in the A4Cloud Toolset

5 Data Track Tool

As part of the European FP6 and FP7 research projects PRIME[6] and PrimeLife[7], the Data Track tool was developed [11, 14]. The PRIME Data Track tool was initially a history function for keeping log records for each transaction in which a user discloses personal data. Each log record included information for the user about which personal data were disclosed to whom, for which purposes, which credentials and/or pseudonyms have been used in the context of the disclosure as well as the details of the agreed-upon privacy policy. These transaction records were stored at the user side in a safe manner (protected by the PRIME core). The follow-up PrimeLife and A4Cloud projects have extended the Data Tack to allow users to exercise their data subjects' rights pursuant to Art. 12 EU Data Protection Directive 95/46/EC to access their data at the remote (cloud) services sides online and to correct or delete their data online if the service provider allows it.

In its backend, the architecture of the Data Track consists of four high-level components. First, the *user interface* component, which displays different visualizations of the data provided by the Data Track's *core*. Second, the *core* component is a backend to the UI with local encrypted storage. Through a RESTful API, the core is able to provide a uniform view to the UI of all users' data obtained from a service provider via *plugins*. Third, the *plugin* component provides the means for acquiring data disclosures from a source and parsing them into the internal format readable by the core. Fourth, the Data Track specifies a generic *API* component that enables a service provider to support the Data Track by providing remote access, correction, and deletion of personal data. Based on the solution proposed by Pulls *et al.* [15], the transfer of data through a service's API can be done in a secure and privacy-friendly manner. By retrieving data from different services through their provided APIs, users would be able to import their data immediately into the Data Track and visualize it in different ways. The possibility to immediately import data into the Data Track and visualize it is an important feature that can add instant value to the tool and provide users with immediate gratification.

Usability tests of early design iterations of the PrimeLife's Data Track already revealed that many of the test users had problems to understand whether data records were stored in the Data Track client on the users' side (i.e., under the users' control) or on the remote service provider's side (i.e., outside the user's control). Therefore, in the A4Cloud project, we have developed and tested an alternative Human Computer Interaction (HCI) concept consisting of graphical user interface (UI) illustrations of where data is stored and to which entities data has been distributed (see [10], [7]). One main motivation for this new UI concept of so-called trace view illustrations is that graphical illustrations of data storage and of data flows have the potential to display data traces more naturally, like in real world networks.

[6] EU FP6 project PRIME, `https://www.prime-project.eu/`
[7] EU FP7 project PrimeLife `http://primelife.ercim.eu/`

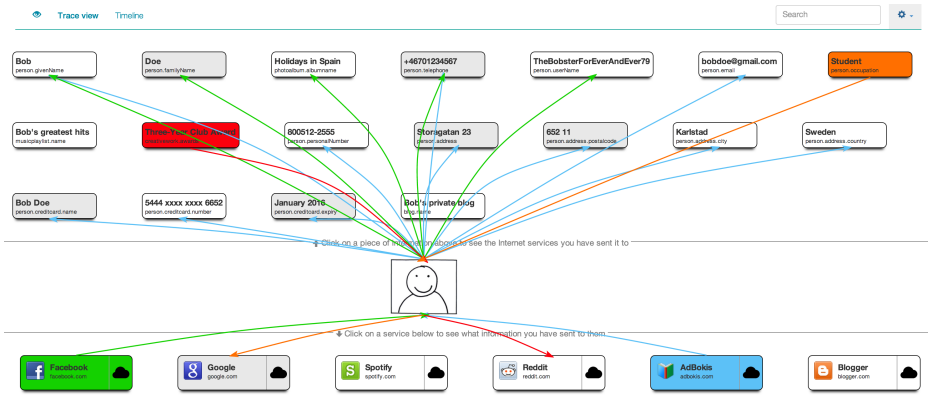

Fig. 4. The trace view interface of the Data Track tool

The trace view visualization After several rounds of paper sketches and lo-fi mockups, which were discussed and refined with the help of domain and HCI experts[8], an interactive prototype of the Data Track's graphical user interface, the trace view, was implemented using HTML5 and jQuery libraries (shown in Figure 4). In the trace view the user is represented by a profile picture in the middle of the screen, motivated by design experts suggesting that users focus most of their attention in the middle of the screen after gazing at the top left corner. In particular, we wanted to give users the perception that this user interface is a *place* that relates to them (i.e. data about them and about the services that they have contacted).

The user interface is then separated into two main panels, following the design guidelines which advice that clearly separating different regions in the screen diminishes the users' cognitive demands. The services to which the user has (explicitly or implicitly) disclosed data appear in the bottom panel and the data attributes that have been disclosed by the user to these services appear in the top panel. By clicking on one (or many) of the services at the bottom of the interface, the interface shows a *trace* from the service to the user, and then from the user to the data items that she has disclosed to that specific service. If the user clicks instead on a data item at the top panel, the trace shows which online services have that particular data items. The traces are coloured allowing the users to easily differentiate between them.

Each service in the bottom panel contains a button with a cloud icon from which users can also access the data about them stored on the services' sides (as seen in Figure 5). Clicking on this cloud icon opens a modal dialog where users can review their personal data that the selected service has stored in their databases (Figure 6). Contrasting colours, an explicit headline and adequate spacing are used to differentiate between the personal data that was explicitly disclosed by the user from the personal data that has been implicitly collected or derived by the service provider. In this view, users also

[8] Early versions of lo-fi mockups with a trace view visualization were developed within the scope of a Google Research Award project in discussion with technical and HCI specialists from Google

Fig. 5. A node representing a service provider, from where users can also access their data located at the services' side by clicking on the cloud icon.

have the possibility to exercise their rights to correct or remove their personal data if the respective service provider allows it.

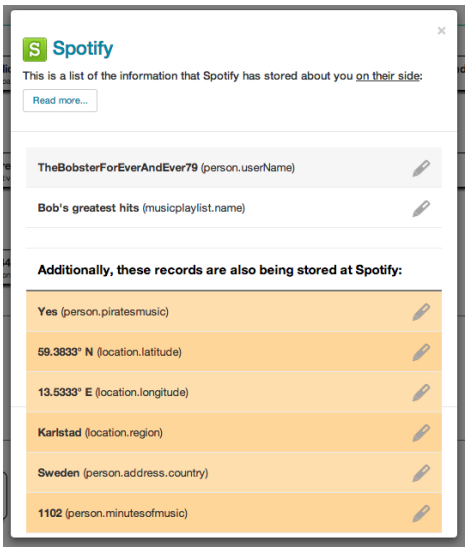

Fig. 6. Information about a user stored at the services' side.

Evaluation The Data Track trace view was evaluated in two iteration cycles with 14 and 17 volunteering test participants between 19 and 40 years, which were recruited from the region of Karlstad in Sweden. 16 of them were students and 15 had other professional background. For the evaluations, the test participants were first introduced to an eShopping scenario, where they had to conduct a purchase transaction for an eBook with fictitious personal data that was claimed to be send to an online bookshop. Then, they were asked to used the Data Track tool to complete different tasks in regard to tracking the data that they previously released.

Both test rounds confirmed that participants easily understood and appreciated having an overview of the data that they have sent to different service providers using coloured tracing lines. However, the test of the first design iteration showed that the controls to access their personal data remotely on the services side did not provide

enough *affordance*. Besides, it was still hard for them to grasp the distinction between data logged locally by the Data Track program and data about them stored remotely in the services' databases. Therefore we included an introduction tour in the second design iteration that illustrated the different aspects of the user interface and explained the distinction of the views showing personal data stored by the Data Track under the user's control and the dialog that showed personal data stored remotely at the service provider. The tour does not only explain the difference between these views, but also how to access them. We also included timely tooltips to explain interface elements that were deemed important when users moved the mouse over them. The tests of the second design iteration showed that in general the user interface evoked the right mental model in 13 out 17 participants, who understood that the Data Track records shown in the trace view were under their control. When asked to identify where would they click to access their personal data that the bookstore had stored about them in their servers, only 4 participants did not complete the task successfully, but they understood the idea after getting assistance from the test moderator. Once the modal dialog opened, all participants correctly identified that more data than they have explicitly disclosed was collected and stored on the service's servers. Eye-tracking analysis of the results revealed that participants paid a lot of attention to the section of the dialog on the bottom displaying the implicitly collected data, which allows us to assume that the test users found especially the functionality of the Data Track allowing users to access also implicitly disclosed data as valuable.

In A4Cloud, the Data Track is combined with the transparency logging tool by Pulls *et al.* [15], from which the Data Track receives information about the flow of the user's personal data along chains of cloud providers. These data flows along cloud chains can also be visualised by the Data Track trace view user interface that we are currently implementing within the A4Cloud project (see also [7] for further discussion and illustrations).

6 Cloud Offerings Advisory Tool

Finding a trustworthy cloud provider among the abundance of available offerings is not an easy task particularly for individuals or small and medium enterprises (SMEs) who do not have the professional advisors available to large enterprsies. Cloud brokers aim to match users' requirements with the offerings but only with a focus on functional requirements and rarely on non-functional ones. In A4Cloud we have developed a brokerage tool, Cloud Offerings Advisor Tool (COAT), that matches the users' non-functional requirements - such as transparency, legal terms, court of choice, privacy and security, etc.- with the contract terms in cloud providers' service offerings. The tool, has several benefits for both cloud customers and providers. For the customers, it will provide an easy comparison for alternative cloud offerings based on customers' requirements, hence increasing transparency and in the process easing the public concern about the security and privacy risks of moving to the cloud. The tool will help customers in understanding the risks involved and help them make appropriate decisions. If a cloud provider is offering unique terms in their offers, COAT can highlight these unique terms in the offer giving the provider a competitive advantage in such a

vast market. COAT can then increase market exposure for some cloud providers. The tool is unique in giving the users the option to state their security and privacy requirements so they get matches based on them. It is also unique in the categorization and structuring of the contractual terms to make it easy for users to understand these terms and the security and privacy requirements they are choosing. In the next subsection we elaborate on the tool design and development and how we analysed the requirements to be included in the tool. More details on the tool and the analysed requirements can be found in [6].

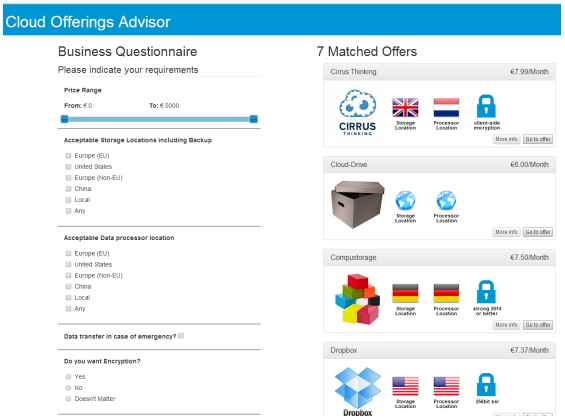

Fig. 7. COAT Interface

6.1 COAT Design and Architecture

COAT filters the variety of offers being presented to customers based largely on the security and privacy attributes of the cloud service. It is aimed primarily at individuals and SMEs. The tool acts as an independent web-based broker that: checks user requirements; matches offers by cloud service providers; compares these offers; explains the terms of offerings; suggests best offerings that match the user requirement; gives general guidance to customers on service offerings. The tool also educates the user on the meaning of the requirements being selected via an explanation text associated with each requirement as shown in Figure 8.

The web-based interface lands on a page which asks the user about their: *Location* (anticipates it first based on the IP address) and their *Role* (whether they are a business SME or an end-user). The tool proceeds by asking the user about the type of service they are searching for, shown in Figure 9 (for SME and non-expert end-users). The tool then uses *dynamic filtering*:

– after selecting the service type. It shows the users the initial list of service offerings filtered only by the type of service they offer.

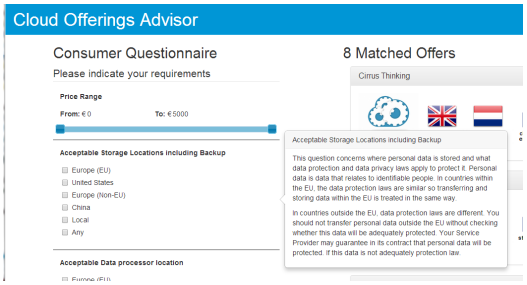

Fig. 8. COAT: Information and Guidance Text for the User-requirements

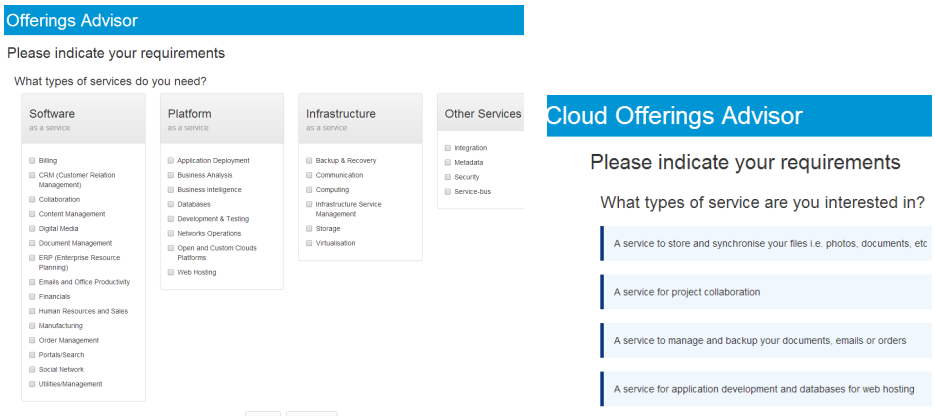

Fig. 9. COAT: Service-types question shown to expert (left) and non-expert (right) end-users

– during filling/answering the requirements questionnaire the list is updated after answering each requirement, filtering the service offerings based on the values of these answers.

Figures 7, 8 and 9 are some snapshots of the tool. The **inputs** to the tool are: User information (location and role), user needs and requirements (answers to the requirement questionnaire), structured service offerings (contract details), and a model of cloud contracts and points of attention. The **outputs** are: matching results of service offerings, guidance on things to pay attention to when exploring and comparing the terms of service offerings, overview of comparable service offerings along with links to their contract details (organized by attributes to facilitate easy understanding of contract terms), a requirement list to give to the Cloud Service Provider (CSP), and SME guidance. The main internal processes are: Matching offers to requirements, Assessment of a cloud service provider offering from a privacy and security perspective, Comparison of offerings (from a data protection compliance and provider accountability point of view), Guidance on the meaning of the comparison attributes and education of users on security, and Logging of the offered advice and the user's decision.

The tool connects to a database of predefined questions regarding the user's requirements and a database of service offerings (MySQL).The *server-side application* and the *webservice layer* that provide access to the questionnaire management and matchmaker are written in Java. The *Matchmaker component* along with the *Questionnaire management* (logic) are implemented in Java as well. The *client-side application* is implemented using HTML5 and JavaScript and is backed by Backbone[9] for a client-side MVC structure. The offers management and associated webservices are written in Python[10]. The Search Index used to find the matched service offerings is done by SOLR. We use RESTful API as a transport layer and JSON[11] as the data-interchange format.

We evaluated the tool by testing it in two workshops: one for cloud service providers and one for cloud customers. The overall feedback was positive. One of the feedback resulted in creating a new service-types page for the customer (figure 9, right handside) to make it simpler for the non-expert users to select the type of services they want. Another feedback was a concern that some cloud service providers will not cooperate in entering their contract details in the tool service-offerings side (populating the tool with offers). However, our argument is that the tool provides good exposure for them and more specifically an exposure to the unique terms that they can offer to their users; this would give small(er) businesses a competitive advantage over large cloud providers. The participants in the cloud customers' workshop evaluated the tool as easy to use and that it has useful functionalities.

7 Conclusion

In this paper, we have provided a general description of the A4Cloud project, which aims to address the problem of accountability in the cloud. This project tackles the problem of accountability from different perspectives: technical, legal, regulatory or socio-economic. The project has provided mechanisms for accountability that are introduced in the conceptual way and later on implemented through a toolset that can be used for the different cloud actors.

We have concentrated here in accountability metrics that are developed from the conceptual point of view. We have also given an overview on the accountability tools and have emphasized in two of them in particular: the Data Track Tool (DTT) and Cloud Offering Advisory Tool (COAT).

In the future we will continue working on the development of the A4Cloud tools and will start the validation of the DTT, COAT and the other tools. As for the work on metrics we will apply it on the development of an Accountability Maturity Model (AMM) and contribute to some consolidated standards on metrics (NIST, ISO), including the ones that we have defined for accountability.

[9] Backbone: `http://backbonejs.org/`
[10] Python Programming Language: `https://www.python.org`
[11] JSON: `http://json.org/`

References

1. AICPA/CICA. Generally accepted privacy principles. http://www.aicpa.org/INTERESTAREAS/INFORMATIONTECHNOLOGY/RESOURCES/PRIVACY/GENERALLYACCEPTEDPRIVACYPRINCIPLES/Pages/default.aspx.
2. Cloud security alliance. cloud control matrix (ccm) v3. https://cloudsecurityalliance.org/research/ccm/.
3. The Cloud Accountability Project. http://www.a4cloud.eu/.
4. Implementing accountability in the marketplace – a discussion document. accountability phase iii. Centre for Information Policy Leadership (CIPL), November 2011.
5. National institute of standards and technology. nist sp 800-53 – security and privacy controls for federal information systems and organizations. revision 4., 2013.
6. Rehab Alnemr, Siani Pearson, Ronald Leenes, and Rodney Mhungu. Coat: Cloud offerings advisory tool. In *IEEE6th International Conference on Cloud Computing Technology and Science, CloudCom 2014, Singapore, 15-18 December*, 2014.
7. Julio Angulo, Simone Fischer-Hübner, Tobias Pulls, and Erik Wästlund. Usable transparency with the "Data Track" - A tool for visualising data disclosures. In *SIGCHI Conference on Human Factors in Computing Systems (CHI '15). Work-in-progress track. Seoul, South Korea.*, April 2015.
8. Walid Benghabrit, Hervé Grall, Jean-Claude Royer, Mohamed Sellami, Monir Azraoui, Kaoutar Elkhiyaoui, Melek Önen, Anderson Santana De Oliveira, and Karin Bernsmed. A cloud accountability policy representation framework. In *CLOSER 2014, 4th International Conference on Cloud Computing and Services Science, 3-5 April 2014, Barcelona, Spain*, Barcelona, SPAIN, 04 2014.
9. Massimo Felici, Theofrastos Koulouris, and Siani Pearson. Accountability for data governance in cloud ecosystems. In *IEEE 5th International Conference on Cloud Computing Technology and Science, CloudCom 2013, Bristol, United Kingdom, December 2-5, 2013, Volume 2*, pages 327–332, 2013.
10. Simone Fischer-Hübner, Julio Angulo, and Tobias Pulls. How can cloud users be supported in deciding on, tracking and controlling how their data are used? In Marit Hansen, Jaap-Henk Hoepman, Ronald E. Leenes, and Diane Whitehouse, editors, *Privacy and Identity Management for Emerging Services and Technologies - 8th IFIP WG 9.2, 9.5, 9.6/11.7, 11.4, 11.6 International Summer School, Nijmegen, The Netherlands, June 17-21, 2013, Revised Selected Papers*, volume 421 of *IFIP Advances in Information and Communication Technology*, pages 77–92. Springer, 2013.
11. Simone Fischer-Hübner, Hans Hedbom, and Erik Wästlund. Trust and assurance HCI. In Jan Camenisch, Simone Fischer-Hübner, and Kai Rannenberg, editors, *PrimeLife - Privacy and Identity Management for Life in Europe*, chapter 13, page 261. Springer, June 2011.
12. Peter Mell and Timothy Grance. The NIST definition of cloud computing. Technical Report SP 800-145, 2011.
13. David Nuñez, Carmen Fernandez-Gago, Siani Pearson, and Massimo Felici. A metamodel for measuring accountability attributes in the cloud. In *Cloud Computing Technology and Science (CloudCom), 2013 IEEE 5th International Conference on*, volume 1, pages 355–362. IEEE, 2013.
14. John Sören Pettersson, Simone Fischer-Hübner, and Mike Bergmann. Outlining "Data Track": Privacy-friendly data maintenance for end-users. In *Advances in Information Systems Development*, pages 215–226. Springer, 2007.
15. Tobias Pulls, Roel Peeters, and Karel Wouters. Distributed privacy-preserving transparency logging. In *Workshop on Privacy in the Electronic Society (WPES)*, pages 83–94, Berlin, Heidelberg, Germany, November 2013.