CLOUD
ACCOUNTABILITY
PROJECT

D: C-3.2 Whitepaper: Interoperability for accountability

Deliverable Number	D33.2
Work Package	WP 33
Version	Final
Deliverable Lead Organisation	CSA
Dissemination Level	PU
Contractual Date of Delivery (release)	30/09/2014
Date of Delivery	31/10/2014

Editor

Alain Pannetrat (CSA)

Contributors

Remi Douence (EMN), Åsmund Ahlmann Nyre (SINTEF), Martin Gilje Jaatun (SINTEF), Vasilis Tountopoulos (ATC), Theofrastos Koulouris (HP)

Reviewers

Michela D'Errico (HP), Tobias Pulls (KAU)

A4Cloud
www.a4cloud.eu
Accountability For Cloud and Other Future Internet Services
FP7-ICT-2011-8-317550-A4CLOUD

SEVENTH FRAMEWORK PROGRAMME

Executive Summary

Interoperability has been instrumental in the creation of cloud services, allowing Cloud Providers to leverage standardized interfaces and formats to deliver services to an unlimited set of customers. This first level of interoperability is not enough for many customers, who are also demanding application data level interoperability to enable portability across service providers in order to avoid vendor lock-in. In this whitepaper, we examine a third level of interoperability: enabling and automating accountability.

In this work, which is conducted as part of the Cloud Accountability Project[1] (A4Cloud), we analyse the interoperability features needed to support accountability in interactions between three core actors of the accountability supply chain: the Cloud Subjects, the Cloud Customers and the Cloud Providers. In particular, if we restrict ourselves to the data protection domain, this means that we will focus our attention on interoperability between data subjects, data controllers and data processors.

Our analysis is conducted on three complementary levels.

First, by identifying cloud actors and interactions, we propose a set of 17 abstract logical interoperability requirements for accountability. Among these requirements, we highlight that a common data handling policy language with standardized semantics is a pre-condition for the creation of interoperability for the purpose of accountability.

Next we look at the concrete technical interoperability feature of the toolset that is created as part of the A4Cloud project, showing how some of our abstract logical requirements are instantiated. While our toolset does not cover all 17 logical requirements, it offers some promising features: a powerful policy language called A-PPL, and a proposal for an API that would allow Cloud Subjects to query Cloud Providers about the handling of their personal data.

Finally, we take a step back and look at interoperability from a broader non-technical perspective, highlighting terminology, organisational and cross-border interoperability issues for accountability.

[1] http://www.a4cloud.eu/

Table of Contents

1 Seeking interoperability for accountability

Cloud computing offers many appealing advantages in terms of cost and flexibility, but many individuals and businesses do not fully trust cloud service providers (CSPs) to handle their data correctly. This lack of trust results from a loss of control by Cloud Customers combined with a general lack of transparency of the data handling practices of cloud service providers. These concerns are starting to be addressed by Cloud Providers and by legislators. Cloud providers are increasingly submitting their service to independent third party audits, obtaining certifications in order to provide assurance to their customers about the security and governance of their platforms. In the EU, the data protection legal framework is undergoing a major revision, which is reinforcing transparency and compliance requirements for IT services, in order to better protect individuals and enhance trust in the digital world [38]. These initiatives and others are part of what could be described as a general trend towards "accountability".

Yet, the notion of accountability may seem vague or lacking a clear path towards implementation, both for Cloud Customers and providers alike. The A4Cloud project (The Cloud Accountability project) was launched precisely to study how to define and operationalize accountability in the cloud, where accountability is defined as follows:

> *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. [4]*

As part of the contributions of the A4Cloud project, this whitepaper analyses the interoperability features that are needed to support accountability in the cloud. To understand the scope of this work, we may consider a cloud supply chain, where one entity A is accountable to another entity B, which itself may be accountable to an entity C, and so and so forth. This process is implemented through specific practices and mechanisms and requires participating entities or supporting tools to exchange information with one another. In this work we look at these exchanges and examine the interoperability features that entities A, B and C need to share in order to enable and automate accountability in the supply chain.

This whitepaper is organized as follows:

- Section 2 provides the analytical foundation that is used to build this work, by:
    - Detailing the notions of interoperability for accountability, cloud actors and interactions.
    - Defining the scope of our analysis, which focuses on Cloud Subjects, Customers and Providers.
- Section 3 presents abstract logical interoperability requirements that we can infer by analysing the accountability interactions between actors.
- Section 4 moves down to examine concrete technical interoperability requirements for accountability, by looking at the toolset that has been built in the A4Cloud project.
- Section 5 concludes our analysis by taking a broader look at the notion of interoperability, discussing terminology, organisational and cross-border aspects of interoperability.

2 Introduction

The scope of this whitepaper is to analyse the interoperability features needed to support accountability in interactions between 3 core actors of the accountability supply chain: the Cloud Subjects, the Cloud Providers and the Cloud Customers. In particular, if we restrict ourselves to the data protection domain, this means that we will focus our attention on interoperability between data subjects, controllers and processors.

This introductory section is designed to justify the scope of our work and to lay the foundation of our analysis. The following sections largely build upon the concepts that we introduce here. As such, we will define more formally what we mean by "interoperability for the purpose of accountability". We will recall the accountability actor taxonomy that is used in the A4Cloud project, and highlight the 4 categories of interactions that exist between accountability actors.

2.1 Defining interoperability for the purpose of accountability

The IEEE standards glossary defines[2] interoperability as the *"ability of a system or a product to work with other systems or products without special effort on the part of the customer"*, which is "*made possible by the adoption of standards*".

In practice, even if we limit ourselves to the cloud, interoperability can serve different purposes.

The first purpose of interoperability in the cloud is service delivery to the customer. Indeed, one of the attractive aspects of the cloud ecosystem is the ability to build new cloud services and applications from other pre-existing cloud services and applications. This is typically exemplified by cloud services like Dropbox [10], which builds upon Amazon storage, or more complex services like Netflix[3], which combine IaaS, PaaS, and content distribution networks across the globe. The ability to make services work together seamlessly across supply chains is made possible by *interoperability.* This interoperability comes through building block standards such as:

1) Web services foundations: XML, HTTP, REST, XML-RPC or SOAP, JSON, etc.
2) Remote IaaS access: SSH and Remote Frame Buffer (VNC).
3) Security: XACML, OpenID, SSL/TLS, IPsec, etc.

In turn, interoperability drives the automation of the processes involved in the provisioning of cloud services, unleashing the efficiencies that make the cloud successful.

Another desirable purpose of interoperability in the cloud is portability, which enables users to move data and applications from one Cloud Provider to another. While portability is still limited in the cloud today, there are some significant standardization initiatives that are moving the cloud towards greater portability, such as for example:

1) OVF [12]: a standard for packaging virtual "appliances" in portable containers.
2) OCCI[4]: an API standard for the management of IaaS and PaaS.
3) CDMI [19]: an API specification standard for managing virtual storage.

In addition, some IaaS Cloud Providers and open-source cloud platforms have adopted Amazon Web Services' APIs as *de facto* interoperability standards in order to convince customers to adopt their platform as a viable alternative.

In this whitepaper, we discuss a third and distinct purpose for interoperability: supporting accountability in the cloud. In short, we ask: what types of data exchanges, APIs and data formats are needed to support accountability across the cloud supply chain.

2.2 Cloud accountability actors

Accountability involves at least two entities A and B, where one entity is responsible to the other for the definition, implementation and/or demonstration of data stewardship practices. The roles that A and B will take influence the interoperability requirements needed to support accountability. Indeed we do not expect to see the same types of data exchanges between an individual user and a Cloud Provider, and between one Cloud Provider and another provider in the supply chain. As a consequence, it is important to identify accountability actors in order to better qualify their interoperability requirements.

In the A4Cloud project, we chose to extend the NIST cloud supply chain taxonomy [16] to create the following cloud accountability taxonomy composed of seven main roles [4]:

- **Cloud Subject**: An entity (individual or organisation) whose data are processed by a Cloud Provider, either directly or indirectly.

[2] From the IEEE Standards Glossary. https://www.ieee.org/education_careers/education/standards/standards_glossary.html
[3] http://techblog.netflix.com/search/label/cloud%20architecture
[4] http://occi-wg.org/

- **Cloud Customer**: An entity (individual or organisation) that (1) maintains a business relationship with, and (2) uses services from a Cloud Provider.
- **Cloud Provider**: An entity responsible for making a (cloud) service available to Cloud Customers
- **Cloud Carrier**: The intermediary entity that provides connectivity and transport of cloud services between Cloud Providers and Cloud Customers.
- **Cloud Broker**: An entity that manages the use, performance and delivery of cloud services, and negotiates relationships between Cloud Providers and Cloud Customers.
- **Cloud Auditor**: An entity that can conduct independent assessment of cloud services, information system operations, performance and security of the cloud implementation, with regards to a set of requirements, which may cover information security, data protection, information system management, laws or regulations and ethics.
- **Cloud Supervisory Authority**: An entity that oversees and enforces the application of a set of rules.

The NIST role taxonomy was chosen as a foundation because of its universal adoption. However, it has some shortcomings when used to describe accountability scenarios. For example, if we look at the data protection domain, which is central in the A4Cloud project, we can observe that the "data subject" is invisible in the NIST taxonomy, except when she/he is also a Cloud Customer. Similarly, Data protection authorities or telecom regulators may be seen as auditors but also have the distinct characteristic of holding enforcement powers, which auditors lack. This conducted us to add the role of Cloud Supervisory Authority as well. For more detailed analysis of the reasons that led us to extend and modify the NIST taxonomy we refer the reader to [4].

We note that the role of Cloud Carrier defined by NIST is unlikely to be considered in the context of accountability, since a Cloud Carrier does not normally take any responsibility for data stewardship but merely acts as a neutral transporter[5] (much like an ISP). In the case where a Cloud Carrier takes a stronger role in terms of data stewardship we may consider it as a Cloud Provider instead without loss of generality.

We also note that our definition of Cloud Subject accounts both for individuals and organisations. An organisation can be a Cloud Subject when it provides business confidential data to a Cloud Provider (or to a Cloud Customer which itself uses a Cloud Provider). Often, organisational Cloud Subject will also have the role of Cloud Customer. Since the main area of application of accountability in A4Cloud is data protection rather than business confidentiality, we will generally focus on the case of individual Cloud Subjects. As such, the terms Cloud Subject can generally be used interchangeably with Data Subject throughout this work.

2.3 Accountability interactions

We identify 4 types of accountability interactions between accountability actors:

1) **Agreement** covers all interactions that lead one actor to taking responsibility for the handling of certain data provided by another party according to a policy. These interactions may include a negotiation phase.
2) **Reporting** covers all interactions related to the reporting by an actor about current data handling practices (e.g. reporting incidents on customer data) and policies.
3) **Demonstration** covers all interactions that lead to one actor demonstrating the correct implementation of some data handling policies. This includes external verifications by auditors or cryptographic proofs of protocol executions for example. We emphasize that *Demonstration* is different from *Reporting* in that it implies some form of proof or provision of evidence.
4) **Remediation** covers all interactions that lead one actor to seek and receive remediation for failures to follow data handling policies.

[5] The issue of *net neutrality* is out of scope of this work.

In the A4Cloud project, we used a taxonomic analysis based on the “attributes of accountability” as detailed in [3] and [4]. For the sake of brevity, we will not reproduce this analysis here, but we can nevertheless highlight that these interactions can also quite naturally be derived from the definition of accountability we presented in section 1. **Agreement** encompasses interactions that contribute to the establishment of “*internal and external criteria*” that one party has to “*comply* [with] *in a responsible manner*”. **Reporting** describes interactions that relate to “*explaining and justifying*” while **Demonstration** describes interactions that relate to “*ensuring implementation*”. Finally, **Remediation** covers interactions between actors that follow from “*remedying any failure to act properly*”.

2.4 Focussing on the core supply chain

Excluding cloud carriers, we have six actors and four types of interactions. Though not all interactions apply to all pair of actors, this combination leads to 31 logical interoperability requirements as described in [3]. However, in practice, it becomes apparent that some requirements are of greater interest or priority in order to automate accountability in the cloud ecosystem. For example, creating a technical standard that allows customers to query Cloud Providers about data stewardship status would have an immediate and strong impact on accountability (requirement R6 in [3]). On the other hand, an *automated* notification mechanism that would allow regulators to request remediation from Cloud Providers is likely to have a low impact (requirement R15 in [3]), since it is a rare event that can be satisfactorily done in a non-automated way. As already described by [20] we believe that the implementation of interoperability features for accountability should prioritize requirements that target actors by frequency of accountability interaction (from high to low):

- Cloud providers and customers (high priority)
- Cloud subjects,
- Brokers,
- Auditors and regulators (low priority)

In this whitepaper, we will focus our analysis on the three first actors: Cloud subjects, Cloud Customers and Cloud Providers. We believe that achieving interoperability between these three actors first will yield the strongest impact on automating accountability in the supply chain.

2.5 Use case

In this section, we introduce a use case to better illustrate the interoperability aspects of accountability, during interactions between cloud accountability actors (as presented in Section 2.3).

For the selected use case, we assume a workflow enabling the provision of a “Wearables” service (see Figure 1 below) by gathering, managing and storing user’s personal data, which are used to keep track of user’s health status for wellbeing over time. The users (Cloud Subjects) are equipped with wearable devices provided by the “Wearable Co.” (Cloud Customer), which are used to record the relevant information and transmit it to the cloud platform CardioMon (Cloud Provider), which shares such data with the Map-On-Web SaaS provider (Cloud Provider) to offer visualisations over the collected and processed data of the customers. Both CloudMon and Map-On-Web also use

In more details, the “Wearables” service aims to enable customers to maintain their profile by dynamically updating their personal data (such as everyday activities, heartbeat rate and blood pressure etc.) produced in a specific location. The service collects this information on a daily basis to automatically build wellbeing training programs, offering additional aggregated statistics over the customers’ data on a monthly or annual basis. This is facilitated by the CardioMon service. Through this cloud SaaS, the service feeds the customers with notifications on physiological health thresholds per age group and location, taking climate and altitude factors into account, and alerts the customers if his or her monitored health indicators exceed these thresholds. Using the Map-on-Web service, the wearable customers are provided with map visualisations of the geographical distribution of the aggregated statistics.

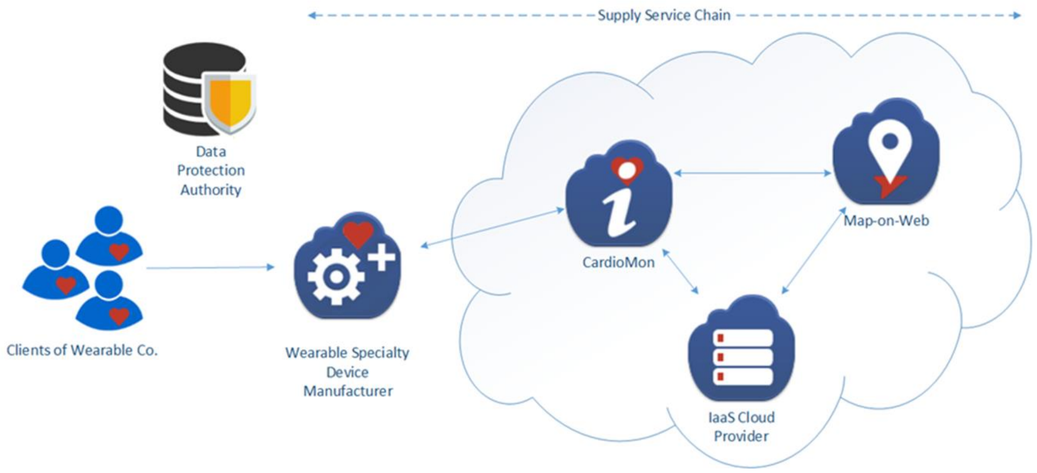

Figure 1: The "Wearables" use case

3 Logical interoperability requirements for accountability

With three actors and four possible types of interactions, we elaborated a set of 17 logical interoperability requirements for the purpose of accountability, which we detail in this section.

A logical interoperability requirement that emerges naturally by listing all interactions in an accountability framework is the ability of parties to share a common understanding of security and privacy policy **semantics** and their associated **metrics**, be it for the purpose of agreement, reporting, demonstration or remediation. Unfortunately, this common ground for semantics and metrics hardly exists today [24]. For example, all major Cloud Providers use different semantics and metrics for availability [23], which suggests that building interoperable policy negotiation protocols even for such a common attribute as "availability" would prove challenging. The same can be said if two interoperating actors have different interpretations of properties and concepts behind keywords such as "consent", "confidentiality level" or "user information" for example.

With the above in mind, from the point of view of interoperability, we are not suggesting that all actors must use a common unique definition of "availability, "confidentiality" or "consent" (though this may be desirable form a legal point of view). We merely underline that the policy language semantics must be sufficiently precise to define attributes in a way that they are interpreted unambiguously and uniformly by all entities in the supply chain. In practice, this means for example that we may have several flavours of "availability" or "consent" each different but defined precisely in a way that all actors are able to understand which flavour is "selected". For example, in [23] the authors show that there are two common models of availability in the cloud, one based on the percentage of successful requests served and the other based on uptime failure rates in time slots. Each one of these models requires a set of parameters to be correctly defined such as sample sizes, failure thresholds and failure events. Being able to specify these parameters in an SLA or policy language is therefore necessary to express the semantics of the property called "availability". Similar caution is required for many other policy attributes.

From these observations we provide two foundational requirements that are common to all interactions between actors:

R01. Shared and well-defined semantics for data handling, security and privacy attributes.

R02. Shared and well-defined metrics for data handling, security and privacy attributes.

To establish the other logical interoperability requirements we need for the purpose of accountability, we examine the four interaction paths that exist between the three actors we focus on, as shown on Figure 2.

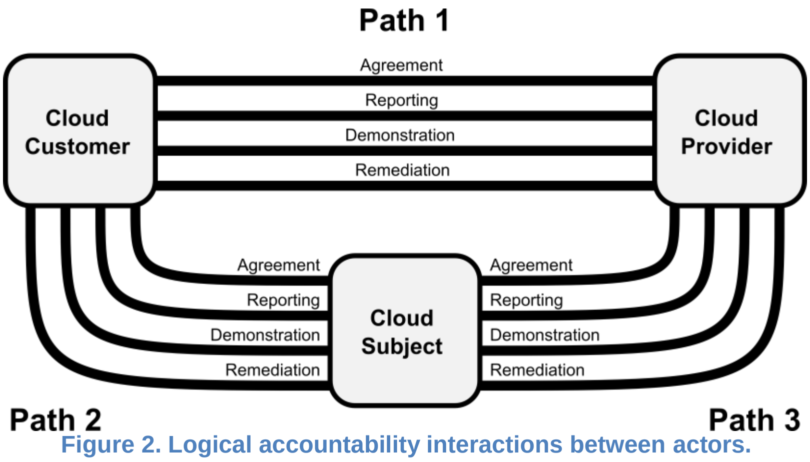

Figure 2. Logical accountability interactions between actors.

We note that paths 2 and 3 will largely produce equivalent requirements: in some cases the Cloud Subject will interface with a Cloud Customer while in other he will exchange data directly with a Cloud Provider. From an interoperability perspective, the Cloud Subject won't exchange data with a Cloud Provider much differently than with a Cloud Customer. We have therefore chosen to group these paths and their related interoperability requirements together in our analysis.

In the following paragraphs we derive requirements from each applicable family of interactions between actors. When possible, we have attempted to illustrate the derived requirements with examples of existing protocols and languages that provide at least part of the required features listed in the requirements.

3.1 Path 1: Cloud Providers and Cloud Customers

We examine Path 1: the accountability interactions between Cloud Providers and Cloud Customers.

3.1.1 Agreement: Cloud customer expresses requirements to Cloud Provider OR Cloud Provider details offering to customer.

3.1.1.1 Description

Cloud customers must find a Cloud Provider who's service offering matches their data handling requirements, which may come from:

- Internal governance, risk and compliance frameworks (e.g. ISO 27001 [25], CSA Cloud Control Matrix[6].)
- Regulations (e.g Directive 95/46/EC [22]) and ethics.
- Relaying requirements from their own customers in the supply chain, when the customer also acts as a Cloud Provider or relays the data subject's requirements.

To achieve this, either:

- The Cloud Customer expresses requirements and the provider indicates its ability to implement them, potentially after a negotiation.
- The Cloud Customer chooses a provider that offers data handling practices that are compatible with its requirements.

3.1.1.2 Interoperability requirements

R03. A language which is able to express:

[6] https://cloudsecurityalliance.org/research/ccm/

1. Obligations and rules applicable to security and privacy attributes (attributes defined by R01) including where applicable:
    a. Purpose definition and limitation;
    b. Security measures (confidentiality, integrity, availability, key management, purpose limitation measures, etc.);
    c. Retention and deletion quality;
    d. Access control to data by privileged and non-privileged staff;
    e. Mechanisms for the exercise of user rights (information, modification and deletion)
    f. The location of data in relation to applicable law;
    g. Transfer of data to third parties;
    h. Mechanisms for the implementation of consent and withdrawal of consent, where applicable.
2. Scope of responsibility for data handling obligations, with the identification of the corresponding parties taking responsibilities for data stewardship.
3. Obligations regarding:
    a. The process of reporting
    b. The process of demonstration
    c. The process of remediation.

> Example(s): The PrimeLife Policy Language [9]. Additionally, some machine-readable SLA languages [27][29] already specify a small subset of the above elements.

R04. A protocol between the two parties, formalizing the acceptance of the terms defined by R03, potentially after negotiation.

> Example(s): SLA negotiation protocols, such as WS-Agreement [28], with some enhancements needed to cover a wider set of policies.

3.1.2 Reporting: Cloud provider informs Cloud Customer about policy implementation

3.1.2.1 Description

In an accountable cloud, the provider should give the customer feedback on the state of data handling, such as:

- Current data handling policy description.
- Indicators for performance, compliance and incident management,
- Data handling reports (who, what and when).
- Location of data
- Effective deletion of data (the gap between deletion requests and their execution),
- Configuration and supply chain changes
- Incident reports, policy and data breach reports.

We distinguish between:

1. Reporting the policy agreed terms (e.g. data A can be transferred to third parties).
2. Reporting the actual data handling practices (e.g. data A was transferred to X).
3. Alerting on deviation / breaches of the policy (e.g. alert: data B was lost).

This distinction will be used to form three types of requirements.

3.1.2.2 Interoperability requirements

R05. A protocol to report the general data handling policy applied by the Cloud Provider to data provided by any Customer, and/or, where applicable, specifics terms that apply to the Customer as negotiated through R04.

Example(s): CSA's CTP [30], or a machine readable version of the CSA PLA[7].

R06. A protocol to report the actual data handling practices performed by the Cloud Provider and, where applicable, compliance indicators relative to the terms of the agreement reached through R04.

Example(s): CSA CTP [30].

R07. A protocol to report data breaches from Cloud Provider to Cloud Customer.

3.1.3 Demonstration: Cloud provider demonstrates data handling practices to Cloud Customer

3.1.3.1 Description

Demonstrating policy application can be performed by several means, such as:

- Technical tools, including cryptography and monitoring.
- Audits by the customer or by a trusted third party.

We distinguish two modes of verification:

1. Evidence based: The Cloud Provider presents evidence artefacts for verification by the customer or allows the customer to directly gather evidence from the provider by performing some interactive tests, in order to verify claims made in the policy from R04.
2. Trust based: The Cloud Provider presents a certification or a trust-mark that supports claims made about the policy from R04, without providing directly the underlying evidence.

3.1.3.2 Interoperability requirements

R08. One or more of:

a. A language to describe evidence that supports claims related to the terms of the agreement reached through R04, along with a supporting protocol to either:
    i. Query evidence gathered by the Cloud Provider for verification by the Cloud Customer.
    ii. Query evidence gathered by a trusted third party (Auditor), and provided to the Cloud Customer by the Cloud Provider.

b. A protocol that enables the cloud Customer to directly test claims made by the Cloud Provider.

c. A machine-readable language that describes the certification by a trusted party of claims made in R04, along with a mechanism for the Cloud Customer to verify the authenticity of the certificate.

Example(s):
(a) ISO 27001 reports, CSA CloudAudit details [33], if they map to policy requirements,
(b) Data retrievability and redundancy testing [34].
(c) ISO 27001 certificates, CSA Star Certification[8].

3.1.4 Remediation (1d): Cloud Customer seeks remediation from Cloud Provider.

3.1.4.1 Description

Cloud customers deal with remediation in two ways:

[7] https://cloudsecurityalliance.org/research/pla/

[8] https://cloudsecurityalliance.org/star/

1. Without interactions, when the Cloud Provider automatically provides compensation to the Cloud Customer based on pre-agreed terms negotiated during a previous Agreement interaction (e.g. a refund for a missed availability performance target).
2. With interactions, when a Cloud Customer makes a request for remediation to the Cloud Provider after detecting a failure (or momentary disruption) in the implementation of the data handling policy defined through R04 and that the Cloud Provider did not implicitly address in a satisfactory way.

By definition, the first case does not impose any interoperability requirements, due to the absence of interaction. We will therefore focus on the second case.

3.1.4.2 Interoperability requirements

R09. A protocol to submit requests for remediation and receive information on the outcome of the request.

Example(s): Service support desks.

3.2 Path 2 and 3: Cloud Subject and Cloud Customer/Provider

We examine Path 2 and 3: the accountability interactions between Cloud Providers and Cloud Customers.

3.2.1 Agreement (4a): Cloud subject and Cloud Provider negotiate data stewardship.

3.2.1.1 Description

To discuss “agreement” accountability interactions, we focus exclusively on cases where the Cloud Subject provides data to a Cloud Provider/Customer for the execution of a contract to which he is a party, or when his/her data is collected by the Cloud Provider/Customer on the basis of the legitimate interest of the Provider/Customer, in the field of data protection. This is the case for the majority of online businesses. By contrast we exclude cases where the Cloud Subject does not have control or choice on stewardship of data handed to a Cloud Provider/Customer. This is often the case when data processing is based on a legal obligation, general public interest or the vital interest of the Cloud Subject (data subject), where there is generally no process of agreement.

There are many modalities that lead Cloud Subject and Cloud Provider to negotiate data stewardship practices:

- Explicitly, when the Cloud Subject globally accepts data handling policy of a service, potentially selecting optional sub-conditions (checkboxes).
- Implicitly, when acceptance of data handling is assumed by default, leaving the Cloud Subject the option to refuse the conditions at a later stage (opt-out).

It is debatable whether an accountable organization would use the second approach, as it poses an implicit contradiction to the transparency attribute of accountability: the Cloud Subject is only made aware after the fact that his data is being processed, and then given an opportunity to opt-out. There is therefore a time-window during which data is being processed non-transparently, that is without the knowledge of the Cloud Subject. This discussion is prominent in the field of online behavioural advertising [39]. On the other hand, in situations where a data processing presents a low privacy risk, some service providers argue that using an explicit agreement protocol for data handling policies may lead a cumbersome online experience for data subjects without a real benefit in terms of privacy. Looking a this question from an interoperability perspective, we can argue for a technical solution that could enable Cloud Providers to always use the “explicit” approach, without its drawbacks: the use of machine-to-machine data policy agreement mechanism. Previous adoption failures concerning P3P [40] and DNT [41] show however that a consensus on this approach is difficult to achieve.

Interoperability requirements

R10. A language equivalent to R03, for the purpose of describing data handling policies from the point of view of Cloud Subjects.

R11. A protocol to negotiate elements of R10 with the Cloud Provider/Customer tailored to the interests and needs of Cloud Subjects.

R12. A “Cloud Subject friendly” control interface for the policy language described in R10.

Example(s): Browser control panel, the PrimeLife Privacy Preference Editor [44].

3.2.2 Reporting: Cloud Provider/Customer informs Cloud Subject about data handling practices.

3.2.2.1 Description

In Europe, data controllers have a legal obligation to provide information to data subject (i.e. individual Cloud Subjects) about personal data processing. As a consequence, the reporting interaction between Cloud Providers/Customers and Cloud Subjects is supported by a legal requirement.

The recently introduced data breach notification framework in the telecom sector [36] also offers regulators the possibility to oblige Cloud Customers/Providers to inform data subjects of a breach, if they have not done so already. According to the regulation, the notification to the regulator is an obligation unless data was encrypted, but the notification to the data subject is subject to an evaluation of the existence of the adverse effects of the breach.

Just as we did for Cloud Customers and Cloud Providers in 3.1.2, we distinguish (1) reporting the data handling policy, (2) reporting the actual data handling and compliance level to the policy and (3) reporting “data breach alerts”.

3.2.2.2 Interoperability requirement

R13. A protocol to query information in a “Cloud Subject friendly” format, presenting the general data policy and/or specific terms agreed in R11.

R14. A protocol to report the actual data handling practices performed by the Cloud Provider and, where applicable, compliance indicators relative to the terms of the agreement reached through R11, (with results presented in a “Cloud Subject friendly” format).

R15. An alert protocol that allows Cloud Subject to be informed about a breach should one occur. Such an interface should at least provide information about the nature of the breach and actions that the Cloud Subject can take to mitigate effects of the breach.

3.2.3 Demonstration: Cloud Customer demonstrates data handling principles to Cloud Subject.

3.2.3.1 Description

Cloud Subject should be able to verify the claims of a Cloud Provider/Cloud Customer with the same tools that a Cloud Customer uses to verify the claims of a Cloud Provider (see 3.1.3). In practice, most users would not have the expertise to evaluate evidence provided by a Cloud Provider/Customer, and rely on trust, be that in a third party (certification and trust-mark), which is embodied in option (c) of requirement R16 below.

Interoperability requirements

R16. One or more of:

a. A language to describe evidence that supports claims related to the terms of the agreement reached through R11, along with a supporting protocol to either:
    i. Query and verify evidence gathered by the Cloud Provider for verification by the Cloud Customer.

   ii. Query and verify evidence gathered by a trusted third party (Auditor), and provided to the Cloud Customer by the Cloud Provider.

b. A protocol that enables the Cloud Subject to directly test claims made by the Cloud Provider.
c. A language that describes the certification (or trust-mark) by a trusted party of claims made in R04, along with a protocol for the Cloud Subject to verify the authenticity of the certificate.

3.2.4 Remediation: Cloud Subject seeks remediation from Cloud Customer/Provider.

Description

Much like Cloud Customers, Cloud Subjects deal with remediation in two ways:

- Without interactions, when the Cloud Provider automatically provides compensation to the Cloud Subject based on pre-agreed terms negotiated during a previous Agreement interaction (e.g. a refund for a missed availability performance target).
- With interactions, when a Cloud Subject makes a request for remediation to the Cloud Provider after detecting a failure (or momentary disruption) in the implementation of the data handling policy defined through R10 and that the Cloud Provider did not implicitly address in a satisfactory way.

By definition, the first case does not impose any interoperability requirements, due to the absence of interaction. We will therefore focus on the second case.

3.2.4.1 Interoperability requirements

R17. A protocol for Cloud Subjects to submit requests for remediation to Cloud Providers/Customers and receive information on the outcome of the request.

3.3 Summary of logical interoperability requirements

The following table summarizes the 17 logical interoperability requirements we have detailed previously.

Table 1. Summary of logical interoperability requirements for accountability.

Path	Actors involved	Summarized requirements
All	All	R01. Standardized semantics for data handling, security and privacy. R02. Metrics for data handling, security and privacy.
1	Customer ⬄ Provider	R03. Data handling policy and obligation language R04. A negotiation protocol for the terms of R03. R05. A policy reporting protocol. R06. A compliance/practice reporting protocol. R07. A data breach notification protocol. R08. An evidence reporting protocol. R09. A remediation request protocol.
2 or 3	Cloud Subject ⬄ Customer or Cloud Subject ⬄ Provider	R10. Data handling policy and obligation language, equivalent to R03 but scoped for Cloud Subjects. R11. A negotiation protocol for R10. R12. A user-friendly control panel for R10. R13. A user-friendly policy reporting protocol. R14. A user-friendly compliance/practice reporting protocol. R15. A data breach notification protocol. R16. A trust-mark or evidence verification protocol. R17. A remediation request protocol.

4 Technical interoperability requirements in A4Cloud

A set of accountability tools is being developed as part of the A4Cloud Project (see Annex A), which broadly speaking cover five functional areas:

1) Contract and Risk management
2) Cloud subject control
3) Policy definition and enforcement
4) Evidence and validation
5) Incident response and remediation

As these tools come to life, the tool designers in the A4Cloud project begin to be faced with the choice of translating some of the logical interoperability requirements we highlighted in Section 3 into concrete technical interoperability requirements, defining Application Programming Interfaces (APIs) and data formats. In this section we examine some of the key interoperability features and gaps of this toolset, from the perspective of accountability, as we reach the end of the second year of the project.

4.1 APP-L: a policy language for accountability

Accountability is established and verified against as set of "*internal and external criteria*" (see section 1). For the purpose of interoperability, actors therefore need a machine-readable language to express some of these "*internal or external criteria*", which typically take the form of a data handling policy language. In A4Cloud, we use an XML-based language for this purpose, which is called the "Accountable Primelife Policy Language" or "A-PPL". A-PPL is a Domain Specific Language (DSL) that enables to define policies with accountability obligations.

A-PPL is an extension of the PrimeLife Policy Language (PPL) [9], which itself is an extension of XACML [17]. Both PPL and A-PPL express an obligation as a set of triggers and actions, where triggers are events filtered by conditions, and where actions (notify or log) are executed by the data controller. These languages includes XACML rules for access control and a data authorization language.

A-PPL extends PPL in the following way:

- First, some roles are added (for instance Auditor) to the four roles in PPL (Data Subject, Data Controller, Downstream Data Controllers, Data Processor).
- Second, the PPL "notify" action is extended with different kinds of notification and recipient, and the PPL "log" action is extended to make explicit the information to be logged.
- Third, new triggers (event conditions) are introduced to deal with policy updates and data subject complaints.
- Finally, new actions are introduced for evidence request, collection and analysis.

A-PPL can be used on the data controller's side to specify data handling policies. A policy specifies statements on access control, authorizations and obligations. In A4Cloud policies include three classes of obligations (data handling obligations, logging/monitoring obligations and incident management obligation). More details can be found in [5], [6] and [13].

To use A-PPL policies, Cloud Providers need to install the A-PPL engine (see Annex A). This engine:

- Acts as a protected database that stores the data that is subject to specific obligations,
- Has access to the policies that apply to this data, expressed in A-PPL, and
- Acts as a policy enforcement point.

Each time an entity wishes to access data and use it for a certain purpose, it makes a request to the A-PPL engine, the request is matched against the policy and the request is either granted or rejected, optionally triggering additional actions such as logging or notifying someone.

4.1.1 Use of A-PPL by other tools

The XML based A-PPL language is powerful but technically complex and requires a high level of expertise to create rules and obligations. To enable its use by a greater number of actors, the A4Cloud project has also defined a higher-level language called Abstract Accountability Language (AAL). End users can write rules in AAL with more ease than A-PPL and the AccLab (Accountability Lab) tool can then perform guided translation of ALL to A-PPL.

The A-PPL language is also used by two other tools in the A4Cloud toolset: the Audit Agent System (AAS) and the Data Transfer Monitor Tool (DTMT):

- The AAS verifies collected evidence provided by software agents against policies, which can be expressed in A-PPL.
- The DTMT logs data transfer API calls at the IaaS level (for example, a request to migrate a virtual machine to another location). The collected logs are matched against a policy language to verify compliance. Though the native policy language of DTMT is not A-PPL, DTMT has the ability to translate a subset of A-PPL into its own native policy language, and is therefore interoperable with A-PPL.

With additional work, the A-PPL language has therefore the potential to become a general data handling policy language that could be used throughout the supply chain to represent data handling policies.

4.1.2 Interoperability perspective

From an interoperability perspective, A-PPL allows us to partially fulfil requirement R03 and R01.

We highlight that the goal of A-PPL is to handle obligations related to data handling, as well as logging and incident management. Therefore it does not cover all the possible features of a policy language as defined in requirement R03 in section 3.1. While A-PPL is a powerful language for the description of personal data handling and related obligations, it was not designed to specify broader policy requirements, such as the security and privacy policy that applies to the cloud service that uses that personal data. For example, A-PPL is not designed to specify security requirements such as “availability”, “use of TLS/SSL”, “Recovery point objectives”, “encryption of data at rest”, etc. A complete policy framework would therefore require either to extend A-PPL with additional features or to use it alongside other domain specific languages, which cover security policy elements that are not expressed in A-PPL currently.

A-PPL is based on XACML and inherits the semantics that are specified in that standard. The language extends XACM with a large set of XML elements to describe accountability obligations. However, many of these elements are not formally specified yet, as illustrated by these few examples:

- A-PPL allows specifying that data must be deleted after a certain period of time, but does not specify what “deletion” means, though it can have several meanings in a cloud service (e.g. remove link to data or fully overwrite data, include data on backups or not).
- A-PPL allows to specify a retention period, triggered from the point data is collected, but it is unclear what happens when data is later modified: is the retention period reset or does it continue to run from the initial point of collection in time.
- A-PPL recently introduced the possibility to specify obligations related to consent or the location of data processing, but does not define the concepts of “location” or “consent”.

As a consequence, A-PPL in its current form does not fully satisfy requirement R01 by providing well-defined semantics. This could lead to different interpretations of accountability policies by actors in the supply chain. These ambiguities can be expected in an on-going research project, and would require a real standardization initiative to be fully solved.

4.1.3 The Data Track and the Transparency Log

The Data Track tool (DT) and the Transparency Log (TL) form a pair of tools that allow Cloud Subjects (data subjects) to perform *reporting* interactions with a Cloud Provider. More precisely:

- The Data Track tool allows users:
  - To query a provider about what personal data they hold about the user;
  - To query a provider about the disclosure of this data to other entities;
  - To request a provider to delete or modify personal data.
- Once coupled with the Transparency Log, the Data Track tools allows users to:
  - Receive notifications about policy violations and view up-to-date information about how data is processed.

The detailed functioning of these tools is presented in [45] and [8]. From a point of view of interoperability, these tools present some attractive features. In particular:

- The Data Track tool defines an API that Cloud Providers should implement in order to allow data subjects (Cloud Subjects) to get information about the handling of their data, and also to potentially request deletion or change of the data.
- The Transparency Log defines a secure channel for Cloud Providers to send notifications to end users, notably notification of violations.

4.1.4 Interoperability perspective

The TL partially fulfils requirement R15, while DT fulfils requirement R14.

We note that the TL only provides a secure channel and does not specify the format of the notifications, which alone only entails partial interoperability. On the other hand, the query API defined in the context of the DT tools is a promising approach to fulfil one of the requirements we defined in requirements R14, namely the ability for users to get reports on the handling of their data. Making such an interface a standard across all Cloud Providers (and customers) would hugely benefit Cloud Subject and transparency in the cloud in general.

4.2 Incident notification in the supply chain

The two main tools that deal with incident response are the Incident Response Tool (IRT) and the Remediation and Redress Tool (RRT). At this stage of the A4Cloud project, these tools have not reached a sufficient level of maturity to evaluate their interoperability features. Nevertheless, we can highlight one key interoperability issue closely related to incident *response*: incident *notification* in the supply chain.

Taking as an example the use-case described in Section 2.5, consider the case where the IaaS provider experiences a partial loss of the data stored by CardioMon, for instance due to a software misconfiguration. The IaaS provider should notify CardioMon of the problem and CardioMon will notify Wearable Co., which in turn will notify the affected wearable users and take corrective measures. It is important to highlight a few important facts from our example:

- The IaaS provider cannot directly notify wearable users of the incident, since the IaaS provider normally has no way of knowing which wearable users are affected. What the IaaS provider will know is which resources were affected by the failure: a set of virtual containers, for example.
- CardioMon cannot directly notify wearable users of the incident, since it is providing a service to Wearable Co. Instead it will have to translate and add contextual information to the notification received by the IaaS provider and pass this to Wearable Co.

Wearable Co. will have to translate the notification provided by CardioMon into several notifications that are directed to wearable users (i.e. its customers). The notifications provided by the IaaS and the one provided to the wearable user are therefore likely to be very different in content. The TL provides us a tool to send notifications to the Cloud Subject and could be used as well to send data between a Cloud Provider (the IaaS provider in our example) and a Cloud Customer (Wearable Co in our example). As noted previously however, the TL does not define a format for the notification data that is exchanged between entities. We could easily extend our example to more complex supply chains but the interoperability requirements remain the same. From an interoperability point of view, in order to fulfil requirements R07 and R15, we would need to define a machine readable incident notification format that is flexible enough to cover all interactions in the supply chain. Addressing this issue is on-going work in this project.

4.3 Summarizing the interoperability of the A4Cloud platform

The following tables takes each one of the 17 logical interoperability requirements we detailed in section 3 and summarizes their level of fulfilment by the A4Cloud toolset.

Table 2. Summarizing interoperability requirement fulfilment.

Interactions	Interoperability requirement for accountability	Solution in A4Cloud
All	R01. Standardized semantics for data handling, security and privacy.	Very partially provided by A-PPL.
	R02. Metrics for security and privacy attributes.	*Missing*
Customer ⇔ Provider	R03. Data handling policy and obligation language	Partially provided with A-PPL
	R04. A negotiation protocol for the terms of R03.	Manual
	R05. A policy statement reporting protocol.	*Missing*
	R06. A compliance/practice reporting protocol.	*Missing*
	R07. A data breach notification protocol, to customers.	Work in progress.
	R08. An evidence reporting protocol.	Manual
	R09. A remediation request protocol.	*Missing*
Cloud Subject ⇔ Customer or Cloud Subject ⇔ Provider	R10. Data handling policy and obligation language, equivalent to R03 but scoped for Cloud Subjects.	Partially provided by AAL / A-PPL
	R11. A negotiation protocol for R10.	Manual
	R12. A user-friendly control panel for R10.	*Missing*
	R13. A user-friendly policy statement reporting protocol.	*Missing*
	R14. A user-friendly compliance/practice reporting protocol.	Provided through the combination of DT and TL tools.
	R15. A data breach notification protocol.	In development
	R16. A trust-mark or evidence verification protocol.	*Missing*
	R17. A remediation request protocol.	In development

The following figure presents the same information from an accountability interaction perspective, providing a snapshot of our progress at end of year 2 of the project. Each one of the 4 accountability interactions is represented with a colour, with the following meaning:

- Light grey: An interoperable interface will unlikely be provided by the A4Cloud toolset for this accountability interaction.
- Dark grey: An interoperable interface is not provided in the A4Cloud toolset, but interactions are conducted manually.
- Orange: An interoperable interface will likely to be provided by the A4Cloud toolset, covering at least some requirements related to this accountability interaction, ort t.
- Green: An interoperable interface that covers all or most of the requirements related to this accountability interaction is already provided by the A4Cloud toolset.

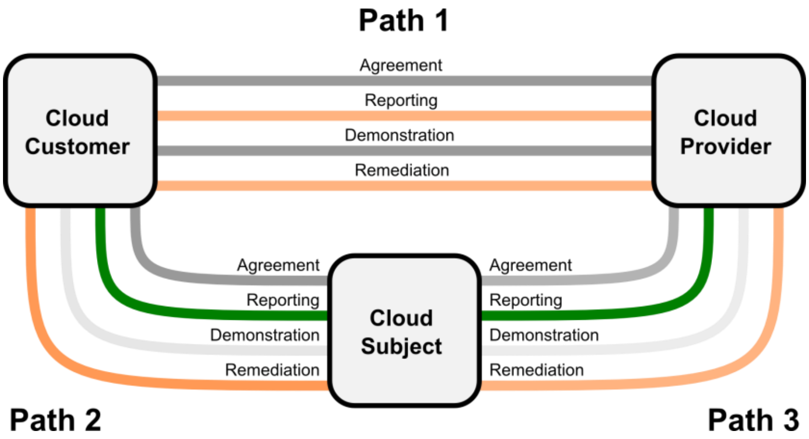

Figure 3. Accountability interactions coverage by the A4Cloud toolset

5 A broader view of interoperability

In this section we extend the definition of interoperability beyond the commonly accepted technical definition we described in the introduction. Instead, we take a broader view and examine interoperability from a terminology, governance, organizational and cross-border perspective.

5.1 Terminology standards in policies and SLA

Data protection is one of the main sources of requirements considered in the A4Cloud project, and it comes with its own terminology and concepts, which vary from one country to another. Yet, accountable organizations and customers will need to interpret requirements that relate to data protection in policies and SLAs, with a common understanding. As developed in D:C-3.1, a key interoperability requirement for accountability is the use of standardized security and privacy semantics. We therefore consider the question of whether differences in terminology standards affect interoperability. While we apply this example to data protection here, it could be generalized to other application domains of accountability.

In the data protection domain, interpretations of some key notions change depending on the applicable legal framework. While this deliverable does not aim to conduct a comparative analysis of the EU and US data protection regulations, we can highlight two notable examples of diverging interpretations that exist between those two regions:

- **Personal data**. EU regulators take a broad view on what constitutes personal data. It typically includes any information that allows singling out an individual or household (WP29 2007), even if his/her “real” name is not directly known by the data controller (e.g. IP addresses, email address, cookies UIDs, etc.). In the US, there is no federal level general data protection law and various sectorial or state laws have divergent definition of “personal data”. Nevertheless, in their privacy policies, US IT companies typically limit the scope of “personal data” to data that are directly linked to a “real” identity (e.g. name and address, social security number, credit card number, etc.).

- **Consent and choice**. In the EU, the concept of consent normally requires a positive and informed action from the data subject[9] (Cloud Subject), whereas in the US, there is a tendency to rely more on implied consent. Though reality is more nuanced than that, we can simplify things by saying that, the EU tends to lean toward an “opt-in” approach, whereas the US goes towards an “op-out” approach.

In the situation where a cloud supply chain crosses two regions, for example with an EU customer and a US provider, these differences need to be taken into account. From an interoperability point of view, the only requirement is that all partners in the supply chain need to have a common interpretation of these concepts, or there should be the possibility to distinguish different versions of these concepts to adapt to different constraints. As such, a US Cloud Provider can technically choose to adopt (or not) the EU definition of “personal data” and consent for its EU customers. We highlight that interoperability

[9] See Article 2 of Directive 95/46/EC

is therefore different from compliance: a policy language may represent policies that are not compliant with a specific regulation.

To summarize, at the technical level, the specification of the policy language should [2]:

- Allow to represent and distinguish different interpretations of what seems on a higher level as a singular concept (e.g. the policy language should define several nuances of consent).
- Be specified in an unambiguous way to allow a similar interpretation form all actors, which process the policy.

There are cases however where a policy is possible in a region, and not in another. If a customer wants a specific policy that the provider cannot implement, this should be detected during negotiation phase. Again, this is only possible if the underlying technical specification of the policy language is clear and unambiguous.

5.2 The organizational dimension of interoperability

Organizations tend to standardize their management process in order to adopt a systematic approach and increase the quality of their processes. This is notably the case for Quality Management Systems and Information Security Management Systems, where organizations follow organizational standards such as ISO/IEC 9001 or ISO/IEC 27001 [25] respectively. When organizations decide to also apply a systematic approach to provide accountability, they must identify the accountability features that already exist through the implementation of related organizational standards, and then implement additional measure that target accountability gaps in their processes. How can an organization do this while maintaining a certain level of interoperability with their already adopted process management standards?

To examine this question, we will take the example of an organization that has implemented an ISMS, and look at possible approaches for adding an accountability management process.

The most famous ISMS is probably ISO/IEC 27001, an organizational standard that aims to bring a formalized and managed approach to information security. As such, this standard directs organizations to: (i) establish an information security governance process, (ii) conduct a risk analysis, (iii) apply adequate risk treatment notably by selecting relevant controls, and (iv) monitor and update the information security process through time.

ISO/IEC 27001 is not the only standard for ISMS. In the area of cloud computing, the control framework usually associated with ISO/IEC 27001 (i.e. ISO/IEC 27002) does not cover cloud specifics very well. This situation has lead the CSA to propose an alternative framework called Cloud Control Matrix (CCM[10]) and is motivating the current development of ISO/IEC 27017, which aims to provide a set of cloud specific control objectives. Many providers also chose to apply PCI-DSS [18] in order to process online payments, FedRamp[11] to work with US government contracts, and various other information security frameworks such as NIST SP 800-53 [15], or COBIT [11]. This multiplicity of approaches can create important complexity and it is natural to evaluate whether some frameworks are “interoperable” with others in order to reduce work. In fact this is largely the case: the CSA CCM reference documentation contains a table of correspondence between all major control frameworks cited above and more.

Though our exploration of the standard landscape in the A4Cloud project, both through a work package dedicated to standards [1] and the Conceptual Framework [4] we have determined that there is very limited work which could be considered as the equivalent to ISMS for accountability. To the best of our knowledge, the only existing framework that approaches this idea is the new “Privacy Office Guide to Demonstrating Accountability” by NYMITY [43]. Yet, there are naturally some elements of accountability in ISMS approaches, because they typically include “governance”, “risk” and “compliance”, all of which concur to promote adequate data stewardship principles.

[10] https://cloudsecurityalliance.org/research/ccm/

[11] http://cloud.cio.gov/fedramp

In order to promote the creation of standardized and interoperable accountability management, there are two possible approaches:

1) Create a new independent "accountability management system" standard, mirroring what exists for ISMSs today.
2) Create an "interoperable" extension to an existing ISMS control framework, which adds accountability specific control.

In A4Cloud, we have decided to examine both approaches because they are complementary in furthering our understanding of organizational accountability practices:

- The first approach is emerging as a result of the work done to create the A4Cloud reference architecture (see in particular "Accountability Governance" in [7]), which proposes a set of controls and best practices for the management of accountability in an organization.
- The second approach has largely been the focus of the work conducted in work packages related to standardization (A-5) and conceptual models (C-2) (see in particular the work on accountability maturity models).

5.3 The cross-border dimension of interoperability

In the data protection domain, the topic that often comes first when discussing cloud services in a globalized world is international data transfers. The EU data protection rules state that data may not be transferred outside the European Economic Area (EEA) to a country that does not ensure an adequate level of protection [22]. The European Directive provides derogation to this principle in limited cases (e.g. unambiguous consent of the data subject, performance of a contract.) and offers various tools to enable cross-border transfer of personal data.
We briefly recall these tools[12]:

- First and foremost, personal data can be transferred without restriction outside the EEA to a set of countries, which are considered as providing "adequate" data protection. These countries include Andorra, Argentina, Australia, Canada (commercial organisations), Switzerland, Faeroe Islands, Guernsey, Israel, Isle of Man, Jersey, New Zealand and Uruguay.

- The US is not considered globally as a country providing "adequate" safeguards, however transfers are permitted to companies that comply individually with US Department of Commerce's Safe Harbour Privacy Principles.

- Multinational corporations that have a presence in the EEA and wish to transfer personal data within their organisations to locations outside the EEA can adopt Binding Corporate Rules (BCR). These corporations need to get their BCRs approved by a national data protection authority.

- A controller within the EEA can transfer data to another controller or processor outside the EEA, provided that these transfers are described in a contract containing a set of "standard contractual clauses" that have been formalized by the European Commission[13].

These rules create the conditions for an organisation to be accountable to another (and to data subjects) across jurisdictions. While such conditions are not sufficient for accountability, they are necessary.

As a consequence, we can view these cross-border rules as legal interoperability "enablers", which organisations must be aware of.

6 Conclusion

Interoperability can serve many different purposes in the Cloud: service delivery, portability and accountability. By creating interoperable interfaces and data formats to exchange accountability

[12] For more details see for example http://ec.europa.eu/justice/policies/privacy/docs/international_transfers_faq/international_transfers_faq.pdf

[13] See Decision 2001/487/EC, 2004/915/EC and 2010/87/EU.

information between cloud actors, we can automate accountability, making it cheaper and more universal through the adoption of relevant technical standards.

In this work, we highlighted a set of 17 core logical interoperability requirements for accountability. These requirements were contrasted with the actual concrete technical interoperability requirements that stem from the construction of the A4Cloud accountability toolset. We highlighted the potential central role that A-PPL can play as a core accountability policy language across accountability tools and actors. We also underlined the promising idea of a standardized universal interface that could be offered by all Cloud Providers or customers in order to allow Cloud Subjects to obtain information about the handling of their data. Despite these interesting approaches, there are still interoperability gaps in the A4Cloud project. We acknowledge that there is still a long way to go in order to provide an interoperable accountability framework.

One particular area that we believe is worth investing into is the definition of APIs and data models that allows Cloud Providers to report data handling back to Cloud Customers in an interoperable way. Since a Cloud Customer can also have the role of a Cloud Provider to yet another Cloud Customer, such API and data models, must aim to allow standardized data handling reporting in the cloud supply chain. These API and data models would ideally cover both the reporting of incidents (Requirement R07) and the reporting of the current state of data handling practices (Requirements R06). The latter is necessary foundation for the creation of a cloud incident-reporting framework.

7 References

[1] A4Cloud. “D:A-5.1 Report on A4Cloud contribution to standards”. Public deliverable, September, 2014.

[2] A4Cloud. “D:C-3.1 Requirements for cloud interoperability.” Public deliverable, 2013.

[3] A4Cloud. “MS:C-3.1 Logical interroperability requirements for accountability”, Internal milestone report, 2014.

[4] A4Cloud. “D:C-2.1 Report on the A4Cloud Conceptual Framework”. Public deliverable, September 2014.

[5] A4Cloud. “D:C-4.1 Policy representation framework”. Public deliverable, December 2013.

[6] A4Cloud. “D:C-4.2 Policy representation and enforcement techniques”. Public deliverable, September 2014.

[7] A4Cloud. “D:D-2.2 High-level architecture”, Public deliverable, September 2014.

[8] A4Cloud. “D:D-5.2 User centric transparency tools”, Internal deliverable, September 2014.

[9] Claudio A. Ardagna, Laurent Bussard, Sabrina De Capitani Di Vimercati, Gregory Neven, Stefano Paraboschi, Eros Pedrini, Stefan Preiss, Dave Raggett, Pierangela Samarati, Slim Trabelsi, and Mario Verdicchio. “Primelife policy language.” 2009.

[10] Idilio Drago, Marco Mellia, Maurizio M. Munafo, Anna Sperotto, Ramin Sadre, and Aiko Pras. “Inside dropbox: understanding personal cloud storage services.” *2012 ACM conference on Internet measurement conference (IMC '12).* New York: ACM, 2012. 481-494.

[11] ISACA. “COBIT 5: A Business Framework for the Governance and Management of Enterprise IT.” 2012.

[12] ISO/IEC 17203:2011 “Open Virtualization Format.” International Organization for Standardization., 2011.

[13] Monir Azraoui, Kaoutar Elkhiyaoui, Melek Onen, Karin Bernsmed, Anderson Santana De Oliveira, and Jakub Sendor. “A-PPL: an accountability policy language for cloud computing.” Research Report RR-14-294, 2014.

[14] NIST. “NIST Special Publication 800-146: Cloud Computing Synopsis and Recommendations.” 2012.

[15] NIST. “Security and Privacy Controls for Federal Information Systems and Organizations.” *SP 800-53.* National Institute of Standards and Technology (NIST), April 2013.

[16] Fang Liu, Jin Tong, Jian Mao, Robert Bohn, John Messina, Lee Badger and Dawn Leaf, “NIST Cloud ComputingReference Architecture”, Special Publication 500-292, National Institute of Standards and Technology (NIST), September 2011.

[17] OASIS. “eXtensible Access Control Markup Language (XACML) Version 3.0.” 22 January 2013.

[18] PCI SSC LLC. “Payment Card Industry (PCI) Data Security Standard.” no. 2. PCI Security Standards Council LLC, October 2010.

[19] SNIA. “Cloud Data Management Interface.” Storage Networking Industry Association, 12 April 2012.

[20] Vasilios Tountopoulos, Massimo Felici, Alain Pannetrat, Daniele Catteddu, Siani Pearson. “Interoperability Analysis of Accountable Data Governance in the Cloud.” *Communications in Computer and Information Science.* Springer, 2014.

[21] WP29. “Opinion 4/2007 on the concept of personal data.” *WP 136.* Article 29 Data Protection Working Party, 20 June 2007.

[22] European Commission, "Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data." *Official Journal L* 281.23/11 (1995): 0031-0050.

[23] Giles Hogben, Alain Pannetrat, “Mutant Apples: A critical examination of cloud SLA availability definitions", accepted for publication at IEEE 5th international conference Cloud Computing Technology and Science (CloudCom). December 2013.

[24] G. Hogben, M.Dekker (Eds.) Procure Secure, A guide to monitoring of security service levels in cloud contracts, ENISA, 2012

[25] ISO/IEC 27001:2005, “Information technology -- Security techniques -- Information security management systems – Requirements”, International Organization for Standardization.

[26] Cloud Security Alliance: “Cloud Control Matrix”, https://cloudsecurityalliance.org/research/ccm/.

[27] Kearney, K.T.; Torelli, F.; Kotsokalis, C., "SLA★: An abstract syntax for Service Level Agreements," *11th IEEE/ACM International Conference on Grid Computing (GRID), 2010,* vol., no., pp.217,224, 25--28 Oct. 2010.

[28] Alain Andrieux, Karl Czajkowski, Asit Dan, Kate Keahey, Heiko Ludwig, Toshiyuki Nakata, Jim Pruyne, John Rofrano, Steve Tuecke, Ming Xu: “Web Services Agreement Specification” (WS--Agreement), September 7, 2006.

[29] Ludwig, H., Keller, A., Dan, A., King, R., Franck, R.: Web service level agreement (WSLA) language specication. IBM Corporation (2003)

[30] Cloud Security Alliance. CTP Data Model and API – Version 2.5, August 2013.

[31] Cloud Security Alliance. CTP Reference Specification of Service Security Attributes – version 0.2, July 2013.

[32] Cloud Security Alliance. Privacy Level Agreements for CSPs serving the EU. https://cloudsecurityalliance.org/research/pla/

[33] Cloud Security Alliance. CloudAudit. https://cloudsecurityalliance.org/research/cloudaudit/

[34] Juels, Ari, and Alina Oprea. "New approaches to security and availability for cloud data." *Communications of the ACM* 56.2 (2013): 64-73.

[35] Cloud Security Alliance. “STAR Certification / Attestation”, https://cloudsecurityalliance.org/star/

[36] European Commission. "Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector." *Official Journal L* 201.31 (2002): 07.

[37] Commission Regulation (EU) No 611/2013 of 24 June 2013 on the measures applicable to the notification of personal data breaches under Directive 2002/58/EC of the European Parliament and of the Council on privacy and electronic communications

[38] Proposal for a regulation of the European Parliament and of the Council on the protection of individuals with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation). 2012/0011 (COD), Brussels, 25.1.2012.

[39] Article 29 Data Protection Working Party, Opinion 15/2011 on the definition of consent, Adopted on 13 July 2011.

[40] Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., & Reagle, J. (2002). The platform for privacy preferences 1.0 (P3P1. 0) specification. *W3C recommendation*, *16*.

[41] W3C Tracking Protection Group, “Tracking Preference Expression (DNT)”, W3C Editor's Draft 02 October 2013, http://www.w3.org/2011/tracking-protection/drafts/tracking-dnt.html

[42] Article Data Protection Working Party, Opinion 1/2010 on the concepts of "controller" and "processor", Adopted on 16 February 2010.

[43] Terry McQuay, Lauren Reid, “A Privacy Office Guide to Demonstrating Accountability”, NIMITY Research report, 2014.

[44] Simone Fischer-Hübner, Harald Zwingelberg (Eds.) “UI Prototypes: Policy Administration and Presentation – Version 2”, The PrimeLife Project, Public Deliberable, June 2010.

[45] Tobias Pulls, Roel Peeters, Karel Wouters, “Distributed privacy-preserving transparency logging”, Proceedings of the 12th annual ACM Workshop on Privacy in the Electronic Society, WPES 2013, Berlin, Germany, November 4, 2013.

8 Index of figures

9 Index of tables

Annex A – The A4Cloud Toolset

The A4Cloud toolset is composed of the following tools:

- **Contract and risk management:**
  - **The Data Protection Impact Assessment Tool (DPIAT):** This tool is used by Small-Medium Enterprises (SMEs) to identify the risks in a given configuration and environment of carrying out a certain business transaction, such as buying a new cloud service.
  - **The Cloud Offerings Advisory Tool (COAT):** This tool is designed to assist potential Cloud Customers (SME organizations and individuals) in assessing and selecting cloud offerings, with respect to certain security and privacy requirements.
- **Cloud subject control:**
  - **Data Track (DT):** This tool is used by data subjects to get a user-friendly visualization of all personal data they have disclosed to cloud service, with the additional capability to rectify data if necessary.
  - **The Transparency Log (TL):** This cryptographic tool provides a secure and privacy-preserving unidirectional asynchronous communication channel, typically between a Cloud Subject and a Cloud Provider. Messages can be stored on untrusted system and can be still be securely retrieved asynchronously by recipients.
  - **The Plug-in for Assessment of Policy Violation (PAPV):** This is a plug-in component to the Data Track tool that provides an assessment on the criticality of previously detected policy violations. By using it, data subjects can check which policy violations are the most relevant ones.
- **Policy definition and enforcement:**
  - **The Accountability Lab (AccLab):** This tool translates human readable accountability obligations expressed in the Abstract Accountability Language (AAL) into our lower level machine-readable accountability policy language called Accountable Primelife Policy (A-PPL) language.
  - **The Accountable Primelife Policy Engine (A-PPL Engine):** This tool enforces data handling policies and actions (e.g. log or notify), described in A-PPL.
- **Evidence and validation:**
  - **The Audit Agent System (AAS):** This tool enables the automated audit of multi-tenant and multi-layer cloud applications and cloud infrastructures for compliance with custom-defined policies, using software agents.
  - **The Data Transfer Monitoring Tool (DTMT):** This tool automates the collection of evidence describing how data transfers within a cloud infrastructure comply with data handling policies.

- **Incident response and remediation:**
    - **The Remediation and Redress Tool (RRT):** This tool assists Cloud Customers (individuals or SMEs) in responding to real or perceived data handling incidents.
    - **The Incident Response Tool (IRT):** This tool is the entry point for handling anomalies and violations in cloud services, such as privacy violations or security breaches. The tool receives incident notifications and takes the initial steps to respond to these incidents, by sending alerts to the user and gathering comprehensive information related to the incident.

These tools are detailed extensively in [7].