AUTOMATING COMPLIANCE FOR CLOUD COMPUTING SERVICES

Nick Papanikolaou[1], Siani Pearson[1], Marco Casassa Mont[1] and Ryan Ko[2]
[1]*Cloud and Security Lab, HP Labs, Bristol, United Kingdom*
[2]*Cloud and Security Lab, HP Labs, Singapore*

Keywords: Cloud Computing, Compliance, Accountability, Natural Language Processing, Policy Enforcement.

Abstract: We present an integrated approach for automating service providers' compliance with data protection laws and regulations, business and technical requirements in cloud computing. The techniques we propose in particular include: natural-language analysis (of legislative and regulatory texts, and corporate security rulebooks) and extraction of enforceable rules, use of sticky policies, automated policy enforcement and active monitoring of data, particularly in cloud environments. We discuss ongoing work on developing a software tool for natural-language processing of cloud terms of service and other related policy texts. We also identify opportunities for future software development in the area of cloud computing compliance.

1 INTRODUCTION

This paper presents tools and techniques for automating compliance with law, regulations, and other requirements, particularly in the context of cloud computing. The most widely used definition of cloud computing is by NIST (refer to (Mell and Grance, 2011) for details on the service and deployment models mentioned):

"Cloud computing is a model for enabling convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. This cloud model promotes availability and is composed of five essential characteristics, three service models, and four deployment models."

What makes compliance difficult for providers of cloud computing services (referred to heretofore as *cloud service providers*) is the sheer number and complexity of laws and regulations that need to be understood and enforced in their systems. Cloud service providers tend to host their customers' data and the computing infrastructure they use in several, disparate data centers, which are physically located in several different jurisdictions. If a customer's data is stored in a data center located in Germany, for example, it will be subject to German data protection law, which is much more restrictive than data protection law in many other countries. In addition to national laws and regulations, there are international agreements and treaties regarding the transfer of data between different jurisdictions (aka. **transborder data flows**), and the US-EU Safe Harbor agreements are a well-known example. Cloud service providers are expected to take all the relevant rules into account and take appropriate measures.

The way a cloud service provider handles its customers' data is usually specified in a written contract or agreement which comprises the ToS (Terms of Service) and SLA (Service Level Agreement). No commonly accepted standard exists for the format or content of cloud ToS and SLAs, nor any consensus about the expected security and privacy practices of service providers.

This poses difficulties for customers and providers alike, who have expectations (and duties) with regards to a given service offering. End-users require clarity and understanding on issues such as:

- how long a provider keeps data which has been stored or exchanged through its cloud services;
- how and when such data is destroyed;
- what remediation procedure exists in case of data loss and in case of data breach,
- to what extent data will be shared with parties external to the service provider and for what

Papanikolaou N., Pearson S., Mont M. and Ko R..
AUTOMATING COMPLIANCE FOR CLOUD COMPUTING SERVICES.
DOI: 10.5220/0003976906310637
In *Proceedings of the 2nd International Conference on Cloud Computing and Services Science* (CloudSecGov-2012), pages 631-637
ISBN: 978-989-8565-05-1

purpose (e.g. targeted advertising).

Enterprise customers typically require assurances regarding:

- service availability (e.g. estimated downtime per calendar month);
- cost of basic services versus added-value offerings;
- how data stored by a provider is kept isolated from other customers' data (particularly for multi-tenancy arrangements);
- encryption methods used, if any, and authentication technologies;
- backup methods and regularity of backup;
- remediation procedures and compensation offered in cases of data loss and data breach.

Although the field of cloud computing still lacks well-defined standards and best practices, they are actively being developed, and it is likely that cloud service providers will have a business need to adopt them in the future. This introduces another level of compliance and, unless cloud service providers are equipped with appropriate controls and tools, much manual effort may be required to achieve it.

There is also a need for tools that ensure what we might call **self-compliance**, namely compliance of a cloud service provider with its own stated policies. To date there is no obvious way of ensuring that the Terms of Service stated by cloud service providers are actually adhered to fully in practice.

We are interested in developing software tools to enable cloud service providers to be accountable with regards to their data governance practices. In the context of this paper **accountability** refers to the goal of preventing harm to a cloud provider's customers by enforcing adequate protections on these customers' data, and having available effective reporting and auditing mechanisms. See (Pearson, 2011) for a discussion of definitions of accountability.

While accountability in the broadest sense can be guaranteed only through a combination of law, regulation and technical enforcement mechanisms (e.g. in the context of privacy, such mechanisms are Privacy Enhancing Technologies), our focus is on the technical aspects. What is practically required for a cloud provider to be accountable is a set of tools to track the location, flows, and accesses of its customers' data. As we shall see, this capability allows a provider to readily demonstrate compliance to the law and adherence to all relevant regulations and other restrictions. More importantly, this capability allows any instances of non-compliance to be detected easily, so that suitable corrective action can be taken.

There is currently no widely accepted methodology or toolset for technically achieving accountability in cloud computing, with potential solutions being heavily dependent on the particular platform and virtualization technology used by a vendor. What is clear is that a variety of mechanisms need to be put into place to protect against data leakage and to enforce legislation and other related restrictions on the storage and transfer of data, especially across national borders.

This paper presents ongoing work on developing software tools to automate compliance in the cloud, particularly natural-language processing of cloud terms of service; we show how such tools fit within a framework enabling cloud service providers to achieve accountability. Finally the paper identifies several classes of software tools to develop in the future, in order to further automate accountability in the cloud.

1.1 Previous and Related Work

In previous work the authors have developed technical mechanisms for controlling the flow of data in an IT infrastructure, notably through the use of privacy controls (Casassa Mont et al., 2010) , sticky policies (Pearson et al., 2011) , and policy enforcement (Papanikolaou et al., 2011). Although the cited works do not specifically focus on cloud computing scenarios, we expect these techniques to be readily extendable and adaptable to suit the needs of a cloud service provider.

- Comparison of policies and decision support
- Automated enforcement of security and privacy rules

Related work in the context of website privacy policies includes May and Gunter's formalism of policy relations, which are formal relationships defined over the intended semantics (or the authors' interpretation thereof) of P3P (May et al., 2009) . In a previous paper (Papanikolaou et al., 2011) we developed a mapping from P3P to CSP, enabling direct comparison of privacy policies using the model-checker FDR.

The EnCoRe research project is developing a platform for expressing and enforcing privacy preferences for personal data; recent case studies include a system for managing data held within an enterprise's HR systems, and health data stored about individuals and tissue samples in a biobank. Through the use of a suitable policy enforcement architecture, legal and regulatory privacy rules, along with individuals' privacy preferences, can be automatically enforced so that unauthorized and/or unsuitable access to data is prevented. In (Casassa Mont et al., 2010) we proposed a simple conceptual model for representing privacy rules, which can be directly mapped to technically enforceable access control policies (expressed e.g. using XACML).

2 TECHNIQUES FOR EXTRACTING AND ENFORCING SECURITY AND PRIVACY RULES IN CLOUD COMPUTING INFRASTRUCTURE

We are working on tools to automate many of the processes required to ensure that a provider is accountable, although we recognise the difficulty of mapping and linking legal and regulatory requirements - which are high-level and expressed in natural language - to technically enforceable policies on particular data items.

Key techniques that should be used to achieve a significant degree of automation include:

Natural-language processing, in particular, extraction of policy rules from legislative and regulatory texts and corporate rulebooks; these rules should be represented in a form that can be interpreted by a technical enforcement mechanism (esp. a Policy Enforcement Point or PEP), but possibly also so that they can be incorporated into a compliance checker of information governance software (cf. Governance / Risk Management Compliance (GRC) Platforms, widely used in industry). It should be noted here that no natural-language processing system can operate with 100% accuracy, but use of such systems can help to reduce significantly the overall amount of human intervention in the process of policy creation and management. In this paper we present two techniques involving natural-language processing, that we are currently investigating:

- automated information extraction
- segmentation and tagging of terms of service for decision support

Use of sticky policies: by strongly binding policies to the data they are associated with, it is easier for providers to control accesses to data within their cloud infrastructure and there is no need for a central policy repository. From the point of view of automating accountability, the use of sticky policies is a very useful technique. Sticky policies provide a means of data encryption, since the data which a policy is bound to cannot be accessed unless that policy is complied with.

Automated policy enforcement: the deployment of control points throughout a cloud provider's infrastructure where policy rules can automatically be enforced and human users only notified in case of failure or error is essential. We refer to the following current and future HP Labs European and TSB research projects for more related work on policy enforcement: EnCoRe, Information Stewardship in the Cloud, and TrustDomains.

Active monitoring for compliance: we believe that it is fundamental for cloud providers to have in their infrastructure mechanisms for automatically detecting compliance problems and potential sources of such problems. It is possible to formulate and regularly check system invariants corresponding to conditions that should never occur at certain end points, such as links between a provider's data centres, and particularly cross-border links.

There is scope for integration of several of the different approaches described so far into a natural-language processing pipeline, which can be integrated with technical enforcement mechanisms to achieve compliance for privacy: this starts with the initial task of analysing natural-language privacy texts, to the extraction of formalized rules and their automatic enforcement. We are working on developing tools for automating privacy in cloud computing and, for this, natural-language analysis of provider ToS, international laws and regulations will need to be combined with suitable enforcement methods such as distributed access control, sticky policies and policy-based obfuscation.

Figure 1 depicts the integration of the techniques mentioned as a pipeline; each arrow shows a flow between processes. The processing of raw text describing laws, regulations, business rules and terms of service as well as the generation of machine-readable rules are to be typically performed outside the cloud service provider’s infrastructure, while the resulting machine-readable rules are fed into the infrastructure to enforce appropriate control on the customers’ data. The mechanisms of policy enforcement, and particularly the use of sticky policies which are attached to data, are to be implemented within the cloud service provider’s infrastructure.

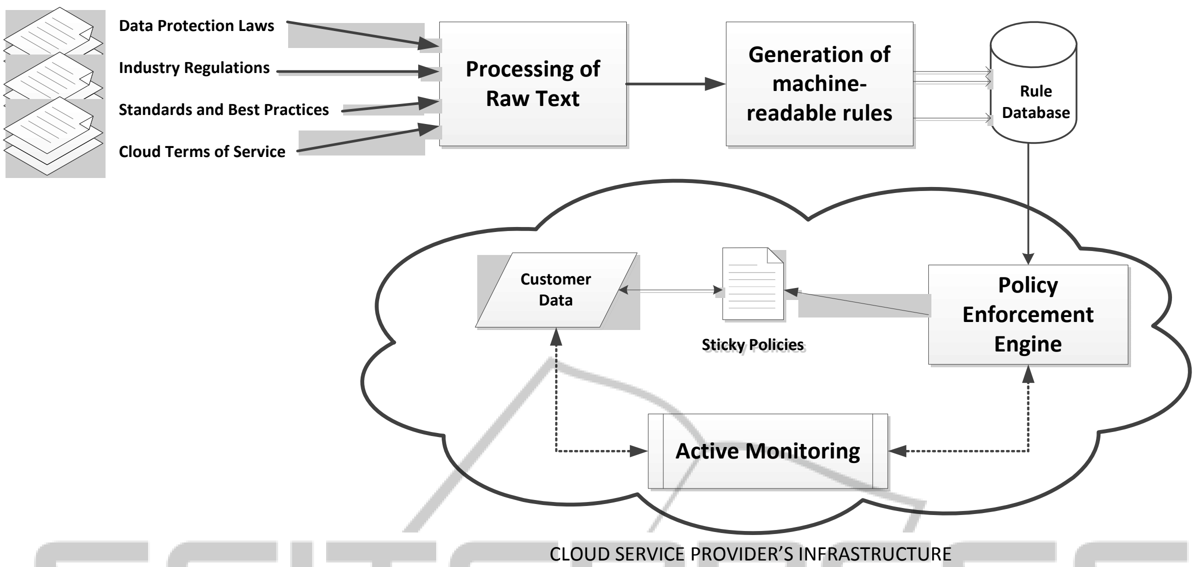

Figure 1: Extracting and enforcing cloud terms of service using a semi-automated tool.

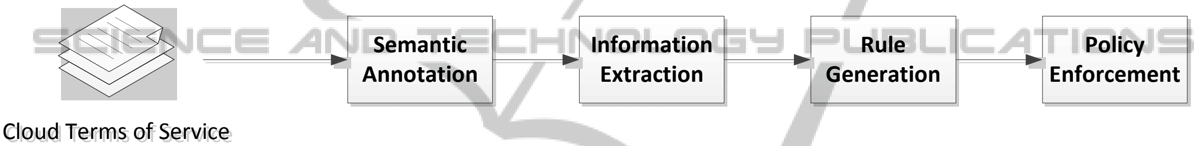

Figure 2: Extracting and enforcing cloud terms of service using a semi-automated tool.

3 A SOFTWARE TOOL FOR ANALYSING CLOUD TERMS OF SERVICE

Figure 2 presents our current model for analysing cloud terms of service.

We are developing a tool for marking up and extracting information from cloud terms of service, namely, the contract documents that describe a customer's relationship with a cloud service provider. Our tool is not fully automated as it requires, as a first step, a human user to indicate which sections of such documents describe which types of rules; this process is referred to as semantic annotation. Our tool provides a text editor with functions to highlight portions of text that describe restrictions, obligations, and other types of constraint with a particular colour. Output from the tool includes a marked-up version of the original contract, with semantic tags. This output can then be fed into a separate processor, which is work in progress, whose functions include information extraction and rule generation. These functions are described in more detail next.

3.1 Automated Information Extraction

Having analysed several real-world cloud ToS, we have observed that:

- Cloud ToS are almost always formatted as rich-text web documents with headings and numbered paragraphs ("clauses" – in the legal sense, not the grammatical sense of the word).
- Significant portions of these texts contain disclaimers, enabling the service provider to refuse being held accountable in certain cases (these parts of the ToS actually state what the provider will not be expected to do, rather than what the provider's actual practices are).
- If a service provider has several similar offerings (e.g. in the case of AWS) there will typically be two documents of interest – (i) a core agreement which sets out the main terms of service, and (ii) separate ToS for each of the different offerings (e.g. in the case of AWS offerings include: EC2, S3, EBS, SQS, SNS, SES, VPC, FWS, SimpleDB, GovCloud). See http://aws.amazon.com for more details on these services.

A recent legal research paper (Bradshaw et al., 2010) documented the features and caveats of different cloud service level agreements, including discussions of both the general service descriptions and the terms and conditions available online.

While a cloud service provider may employ legal experts to draw up their terms and conditions in writing, it is the developers and system administrators that are responsible for making sure these terms are indeed enforced in the infrastructure used for a particular cloud offering. It is in the interest of the latter to have machine readable rules that are in 1:1 correspondence with the statements made in the written ToS.

Natural-language analysis of the written ToS can certainly assist in the creation of such rules; if the written style of an ToS is very prescriptive, enforceable rules are easier to generate automatically. Otherwise human intervention will be required to ensure that generated rules are:

- **correct:** namely, that they express what actions a system needs to implement to make sure the requirements of the ToS are fulfilled on a constant basis;
- **as complete as possible:** namely, that the machine readable rules capture all those aspects of the ToS that can be enforced automatically.

We are not aware of any previous work that addresses the whole lifecycle of natural-language analysis of privacy texts with the goal of enforcing suitable rules, e.g. in an enterprise setting (although the EU CONSEQUENCE project mentioned before does take an holistic approach it does not involve natural-language analysis). As stated in the Introduction, achieving compliance with privacy legislation and regulations is a central concern in enterprises, and means of automating compliance are highly desirable. Since new privacy rules are almost exclusively expressed using natural-language, means of automatically analysing the appropriate texts and extracting rules from them necessary – the resulting rules can then be incorporated into existing enterprise rule-bases, such as those used in compliance checkers or information governance (GRC) platforms.

The most naïve analysis seeks to find in the text of an ToS occurrences of particular verbs, namely verbs which are prescriptive by nature; examples include:

“The Provider will provide a backup of data […]”;

“The User will not upload pornographic images to the service”

since these typically arise in statements expressing duties and obligations (see also (Bradshaw et al., 2010; Breaux et al. 2011; Breaux et al., 2006)). Certain verb groups appear in phrases expressing rights, typically rights of the customer but not necessarily:

“The Customer may request in writing a full copy of data held [...]”

“The Provider can refuse to provide access to the service at any time [...]”

In the case of simple prescriptive sentences it is possible to represent the information given by a triple

(verb, subject, object)

In a Prolog program this would be declared as a Horn clause of the form

```
verb(subject, object).
```

Such a representation says nothing of the nature of the rule or (legal) clause appearing in the ToS, but may assist a service provider in automatically generating a set of access control rules for enforcement within its infrastructure. Our tool uses a form of markup referred to as a formal requirements specification language (RSL); the RSL we are using is due to Breaux and Gordon (Breaux and Gordon, 2011).

3.2 Semantic Annotation and Tagging of Cloud Terms of Service

Extending the method of simple analysis presented in the previous section, our tool is designed to detect delimiters and punctuation, so that long-winded sentences of legalese may be separated into their constituent parts. In a given sentence, those secondary clauses, which serve only to explicate and/or amplify the main thrust of the sentence, may be ignored (subject to interpretation and the judgment of a human user, of course; this suggests the process cannot be completely automated), and a semantic representation can be built of the remaining constituents of the sentence.

An interesting toolkit that we are considering to use to automate part of this task is GATE (“General Architecture for Text Engineering”) (Cunningham et al., 2011) , whose user interface provides a helpful facility for tagging and colour-coding portions of text of particular semantic relevance. The technique that applies here is known as semantic annotation. We believe that such an approach is highly beneficial for the visual representation of the terms and conditions contained in a given cloud ToS.

4 APPLICATIONS AND FUTURE WORK

Here we discuss applications of the above techniques and particularly, what other types of tools need to be developed to improve compliance in cloud computing.

4.1 Decision Support Tools

We believe that being able to efficiently (and automatically) extract security and privacy stipulations from cloud ToS is also a key business advantage, enabling decision support in enterprises for the selection of cloud services and providers as necessary during the course of their daily operations.

4.2 Software Tools for Visualising and Understanding Policies

It has often been noted that presenting privacy policies and similar documents describing terms and conditions directly to end-users rarely draws their attention, and often users tend to click through any agreements of this sort if they require access to a service, thus ignoring details which could have significant consequences to them and their data. Since cloud services are almost exclusively purchased online, and terms and conditions are always presented on-screen to users, it is unlikely that customers of these services will pay due attention to the fine print; we believe that security and privacy policies should be presented in a more visually appealing fashion, which aids comprehension and allows users to compare competitors' data handling practices. This idea is certainly not new, and several previous authors have developed and demonstrated ways to help users visualise and understand terms and conditions; the P3P policy language (Cranor et al., 2002) was designed in part to allow the development of visual tools to understand privacy policies. Research projects such as PRIME, PrimeLife, and EnCoRe have developed user interfaces and dashboards for privacy settings and preferences. Clearly these efforts need to continue and be extended to applications specific to cloud computing.

Through analysis of cloud ToS, it should certainly be possible to generate comprehensible visual representations of a service providers' security and privacy practices. Of course, unless such representations are standardised, this task will be non-trivial.

4.3 Software Tools for Checking Compliance of Cloud Terms of Service with Prevailing Laws, Regulations and Standards

Cloud service providers are likely to audit their systems on a regular basis to ensure that their policies are valid and conform to current law, standards and best practices, adapting ToS and actual practices as necessary.

From this perspective, natural-language analysis can be used to extract rules from legislation and standards; these rules can then be compared and contrasted to ToS rules, triggering changes and extensions as required.

The extraction and representation of policy rules can then be seen as but the first part in a larger process or lifecycle. ToS have to be maintained, adapted, enforced, and audited. One can envisage how metrics for similarity of ToS can be defined or other measures for determining the degree of compliance to a particular industry best practice. This is clearly a very promising direction for investigation, with important implications for enterprises.

4.4 Software Tools for Generating Model or Template Cloud Terms of Service

Natural-language analysis of cloud ToS can help to detect language patterns that are common to such texts. This could be extremely useful in designing templates or 'model ToS'. To have industry agreement on what constitutes a model ToS would be an important step for cloud computing, and hopefully pave the way for the establishment of standard policies and commonly agreed security levels.

Taking this further, it is possible to develop natural-language generation tools which mechanically produce the text of cloud ToS for particular applications. If standards were to be established for the security levels specified by cloud ToS, the format and content of these documents would be well-established, making document generation significantly automatable.

5 CONCLUSIONS

We believe that it is beneficial and possible for cloud service providers to automate a number of tasks related to the requirement of accountability. We have identified some specific techniques, namely: natural-language analysis of law, regulation and corporate guidelines on security and privacy of customer data in order to generate technically enforceable policies; use of sticky policies to achieve a strong binding between data and the stipulations that apply to the use and dissemination of that data; and active monitoring of a cloud provider's infrastructure to detect potential compliance problems. More in-depth analyses of ways to achieve accountability in the cloud are available in some of our previous work (see also (Casassa Mont et al., 2010); (Pearson, 2011); (Pearson et al., 2011) ; (Mowbray et al., 2010) ; (Ko et al., 2011a) ;(Ko et al., 2011b)).

Our main contribution in this paper has been to describe ongoing work on developing software tools for automated information extraction of cloud terms of service, and to identify classes of related software tools needed to achieve full accountability in cloud computing. There is clearly much work to be done to achieve this important goal for the sake of future cloud service users.

REFERENCES

Mell, P., Grance, T. The NIST Definition of Cloud Computing: Recommendations of the National Institute of Standards and Technology. NIST Special Publication, 2011, 800-145.

Bradshaw, S., Millard, C., Walden, I. 2010. Contracts for Clouds: Comparison and Analysis of the Terms and Conditions of Cloud Computing Services. Queen Mary University of London, School of Law Legal Studies Research Paper No. 63/2010.

Breaux, T. D., Gordon, D. G. 2011 Regulatory Requirements as Open Systems: Structures, Patterns and Metrics for the Design of Formal Requirements Specifications. Technical Report CMU-ISR-11-100, Institute for Software Research, Carnegie-Mellon University.

Breaux, T. D., Vail, M.W., and Antón, A.I. 2006. Towards Regulatory Compliance: Extracting Rights and Obligations to Align Requirements with Regulations. In *Proceedings of 14th IEEE International Requirements Engineering Conference (RE'06)*, 2006.

Cunningham, H., Maynard, D., Bontcheva, K., Tablan, V., Aswani, N., Roberts, I., Gorrell, G., Funk, A., Roberts, A., Damljanovic, D., Heitz, T., Greenwood, M.A., Saggion, H., Petrak, J., Li, Y., Peters, W. 2011. Text Processing with GATE (Version 6). Department of Computer Science, University of Sheffield.

Cranor, L., Langheinrich, M., Marchiori, M., Presler-Marshall, M., Reagle, J. 2002. The Platform for Privacy Preferences 1.0 (P3P1.0) Specification. W3C Recommendation.

May, M., Gunter, C., Lee, I., Zdancewic, S. 2009. Strong and Weak Policy Relations. In *Proceedings of the 2009 IEEE International Symposium on Policies for Distributed Systems and Networks (POLICY '09).* IEEE Computer Society, Washington, DC, USA, pp. 33-36, 2009.

Papanikolaou, N., Creese, S., Goldsmith, M. Refinement checking for privacy policies. Science of Computer Programming. Article in Press, DOI:10.1016/ j.scico.2011.07.009.

Casassa Mont, M., Pearson, S., Creese, S., Goldsmith, M., Papanikolaou, N. A Conceptual Model for Privacy Policies with Consent and Revocation Requirements. In Proceedings of PrimeLife/IFIP Summer School 2010: Privacy and Identity Management for Life, Lecture Notes in Computer Science, Springer (2010).

Pearson, S. Toward Accountability in the Cloud. View from the Cloud, IEEE Internet Computing, IEEE Computer Society, July/August issue, vol. 15, no. 4, 2011.

Pearson, S., Casassa Mont, M., Kounga, G. 2011. Enhancing Accountability in the Cloud via Sticky Policies. Secure and Trust Computing, Data Management and Applications, Communications in Computer and Information Science, vol. 187, Springer Verlag, Heidelberg, pp. 146-155.

Mowbray, M., Pearson, S. and Shen, Y. 2010. Enhancing privacy in cloud computing via policy-based obfuscation. Journal of Supercomputing. DOI: 10.1007/s11227-010-0425-z.

Ko, R. K. L, Jagadpramana, P., Mowbray, M., Pearson, S., Kirchberg, M., Liang, Q., Lee, B.S. 2011a. TrustCloud: A Framework for Accountability and Trust in Cloud Computing, 2nd IEEE Cloud Forum for Practitioners (ICFP), IEEE Computer Society, Washington DC, USA.

Ko, R.K.L., Lee, B. S., Pearson, S. 2011b. Towards achieving accountability, auditability and trust in cloud computing. A. Abraham et al. (Eds.), ACC 2011, Part IV, CCIS 193, pp. 432–444, Springer-Verlag, Heidelberg.