Trustworthy Selection of Cloud Providers Based on Security and Privacy Requirements: Justifying Trust Assumptions

Michalis Pavlidis[1], Haralambos Mouratidis[1], Christos Kalloniatis[2], Shareeful Islam[1], and Stefanos Gritzalis[3]

[1] School of Architecture, Computing and Engineering, University of East London, U.K.
m.pavlidis@ieee.org, {haris,shareeful}@uel.ac.uk
[2] Cultural Informatics Laboratory, Dept. Of Cultural Technology and Communication, University of the Aegean, Greece
chkallon@aegean.gr
[3] Laboratory of Information and Communication Systems Security, Dept. of Information and Communications Systems Engineering, University of the Aegean, Greece
sgritz@aegean.gr

Abstract. Cloud computing is a new paradigm with a promising potential. However, issues of security, privacy, and trust raise concerns and discourage its adoption. In previous work we presented a framework for the selection of appropriate cloud provider based on security and privacy requirements criteria. However, the adoption of cloud includes release of control over valuable assets, which constitutes trust in the cloud provider of paramount importance. In this paper we extend the framework by incorporating trust and control concepts in its language and adding a new activity to properly identify and reason about trust assumptions during the selection of appropriate cloud provider. Also, the CASE tool was extended to support the new activity. A case study is used to illustrate the usefulness of our approach.

Keywords: Cloud Computing, Security, Privacy, Requirements, Trust, Control.

1 Introduction

Cloud computing is an evolving paradigm that is radically changing the way humans store, share and access their digital files. Its promise is the introduction of a rapid elastic and unlimited computation, storage, and bandwidth with a significant lower cost. However, to fully realize the potential of the cloud, appropriate security and privacy solutions must be adopted. Many organisations and individuals are still avoiding cloud services mostly because they are not sure if the services provided, by various providers, are suitable for their security and privacy requirements [1]. This is especially true since organisations and individuals would have to hand in their personal and organizational data into service providers over which they have no control.

S. Furnell, C. Lambrinoudakis, and J. Lopez (Eds.): TrustBus 2013, LNCS 8058, pp. 185–198, 2013.

It is therefore important, that appropriate software engineering techniques must be developed to support the structured and systematic identification of security and privacy requirements that an organization might have for their systems and based on those requirements to support selection of appropriate cloud services. However, and despite the recent research interest in developing software engineering techniques to support systems based on the cloud, the literature fails to provide a systematic and structured approach that enables software engineers to identify security and privacy requirements and select a suitable service provider based on such requirements.

To this end, in previous work [2] we proposed a novel framework to support elicitation of security and privacy requirements and selection of a service provider based on those requirements. The framework consists of a modeling language, a process and a tool. The analysis performed by that framework, trusts that the cloud provider will deliver the required security and privacy mechanisms needed for the identified security and privacy requirements. However, blind trust is not ideal, but trust should be supported by appropriate justification. We want to be able to feel confident, in as higher degree as possible, that the cloud provider will deliver as promised and reasonably rely on them to care for our valuable assets. In order to be able to understand that, we need to clearly understand the relevant underlying trust assumptions, make them explicit and justify them.

The work presented in this paper, extends our previous work to address the above challenges and to support justified trust assumptions through a systematic trust based process. In other words, we want to support the decision making process by identifying underlying trust assumptions and justifying the trust that we place on cloud providers. The language is extended with trust and control concepts, new activities are added into the process to identify direct and indirect trust relationships, and also the tool is extended to support the activities. The rest of the paper is organized as follows. Section 2 presents an overview of our previous work. In section 3 we present the extended framework that incorporates the trust process. An illustration of the framework is presented in section 4 using a case study while section 5 presents the related work and section 6 concludes the paper.

2 Background Information on the Framework

The framework we already presented in [1] consists of a language and a process that is focused on the requirements engineering stage. The language employs concepts from the requirements, security and privacy engineering domains, and it is based on our previous work on security requirements engineering, and in particular Secure Tropos [3] and privacy requirements engineering, and in particular PRiS [4]. However, the language is enriched with new concepts, such as cloud actor, measure, and mechanisms, which are necessary to support the selection of cloud providers. The process supported by the framework is iterative and it is based on the development of a set of models that are incrementally refined to include further details. It provides a structured way of eliciting and analysing security and privacy requirements, identifying relevant security and privacy mechanisms and of selecting an appropriate cloud service provider based on these mechanisms. It comprises of three main activities: *the Security*

and Privacy Cataloguing, the *Security and Privacy Analysis,* and the *Selection of Cloud Service Provider*. Each one of these activities has specific inputs and it results in specific outputs. The first two activities enable developers to understand the security and privacy requirements of the system and identify relevant security and privacy mechanisms that the cloud providers should deploy to support the identified security and privacy requirements. Once all the security and privacy mechanisms have been identified, the third activity supports the selection of an appropriate service provider based on the degree of satisfaction of these mechanisms by potential cloud providers. Our framework makes use of an analysis technique based on an independent probabilistic model, which uses the measure of *satisfiability* [5]. In our work, satisfiability represents the probability that the security and/or privacy mechanism will be satisfied. Thus, the evaluation results in contribution relationships from the cloud provider to the probability of satisfying the security and/or privacy mechanisms of the system identified in the previous activity of our process.

To express the contribution of each provider to the satisfiability of each security/privacy requirement of the system, a weight is assigned. Weights take a value between 0 and 1. The allocation of such weights is performed by the security, privacy and cloud experts after studying the required security and privacy mechanisms and the various characteristics and provisions that a potential cloud provider has in place to support these mechanisms. The overall satisfiability level is calculated by summing up all the satisfiability values of an individual cloud provider and dividing that sum by the number of security and privacy mechanisms required by the system. The cloud provider with the highest satisfaction level is the preferred provider.

The framework is supported by a tool that has been developed based on the Open Models Initiative ADOxx Platform (www.openmodels.at). The tool provides an environment for developers to create a number of diagrams that support the described process. In particular, the process described in the previous section results in the development of four artefacts represented in terms of four diagrams. These are the *Security and Privacy Reference Catalogue Diagram*, the *Security and Privacy Organisational Diagram*, *the Security and Privacy System Requirements Diagram* and the *Cloud Provider Selection Diagram* respectively.

3 Framework Extension

The above described framework, helps to select among potential cloud providers based on the probabilities. However, trust is more than subjective probabilities [6], and the selection of a cloud provider should not only be based on calculation of probabilities. Even, if there is a probability that the cloud provider has the capability to support the required security and privacy mechanisms it does not mean that this will happen. What is required is a structured process that can reveal underlying trust relationships, reason about them and enable their justification.

In previous work [7] we have presented a process for trustworthy information systems development that uses a language [8] based on trust and control concepts. We incorporate this work into the framework described in the previous section, to enhance its cloud provider selection activity by considering trust relationships. In

particular, the extension of the framework is threefold. The language is extended with trust and control concepts, new activities are added into the process to identify direct and indirect trust relationships, and also the tool is extended to support the new activities.

3.1 Language Extension

The language has been enhanced with the following trust-related concepts that allow a better understanding of the factors that affect confidence: **Resolution.** Resolution of a dependency is the indication of how the uncertainty in the fulfillment of a dependency is removed in order to build confidence in the dependency. It is necessary to be identified as a dependency implies a vulnerability for the depender because the dependee might not fulfill the dependency. There are two types of resolution, i.e., trust and control, that can be identified to feel confident in the fulfilment of a dependency. Also, there can be more than one resolution. **Trust.** Trust is the positive expectation of one actor about the behaviour of another actor by whom she/he might be positively or negatively affected [9]. In the context of a dependency, the depender is the trustor and the dependee is the trustee. There are four types of trust resolution:

- **Experiential Trust.** Experiential trust is trust that originates from previous direct experience with the trustee. The depender then is actually depending on himself and there can be only one instance of experiential trust, as there is only one instance of someone's self.
- **Reported Trust.** Reported trust is trust that originates from a third party (the reporter) who reports that the trustee is trustworthy. Therefore the depender depends on the reporter to trust the dependee. There can be more than one of reporters who are reporting whether the dependee is trusted. Apart from human the third party can also be a system, such as a reputation system.
- **Normative Trust.** Normative trust is trust that originates from the system environment norm. The depender is then depending on the environment norm. There can be only one environment norm.
- **External Trust.** External trust is trust that originates from sources outside of the system environment. These for example can be government bodies. The depender is then depending on an external source of trust. There can be more than one external sources of trust.

Trust Relationship. Trust relationship is defined as a relationship that exists between the trustor and the trustee and resolves a dependency based on trust. There are two types of trust relationship, i.e., direct and indirect. Direct trust relationship is the trust relationship that exists between the two actors of a dependency and it is not implied by any other trust relationship. Indirect trust relationships are trust relationships that are implied by direct trust relationships or control relationships and need to exist in order to support them. **Control.** Control is the power that one actor has over another actor. It helps to build confidence in another actor. Control specifies the ability of an actor to gather information about another actor in order to decide whether to execute an action. In addition, control specifies the action that is required for the dependee to

behave in an expected way. So, to achieve control, an actor needs to ensure observation and deterrence capabilities. **Entailment.** Entailment is a condition of trust that is required to be valid for having confidence in the dependency from which it is required. For example, if there is a reported trust resolution then it requires the entailment "the reporter is trusted" to be valid. Also, if there is a control resolution then it requires the entailment "the controller is trusted" to be valid. Such assumptions of conditions of trust require evidence in order to be justified.

3.2 Process Extension

An extension has been applied also on the framework's process. In particular, during activity 3 "Selection of cloud service providers", new steps have been added to identify resolutions and entailments, and examine the validity of the entailments. Figure 1 shows the updated activity, using the Software & Systems Process Engineering Metamodel Specification (SPEM). In particular, for each candidate cloud provider a resolution and entailment diagram needs to be constructed. The diagrams enable the identification of indirect trust relationships and the reasoning of them. A resolution can be trust or control and if it is trust resolution, then it can have single or multiple types of trust. Depending on the type of trust, new dependencies may be introduced. So, a reported-based trust resolution creates a new dependency that needs to be resolved. The new dependency is on the reporter. While the other three types of trust resolution, i.e., experiential, normative and external, do not introduce new dependency. If the resolution is control-based then it introduces a new dependency in a similar way as the reported trust resolution. But, this time the dependency is on the controller. Therefore, whenever there are new dependencies created by a reported trust resolution or a control resolution the activity has to be applied again in order to identify resolutions for the new dependencies. At the end there is a list of resolutions that show why the cloud adopter is confident in the fulfilment of the security and privacy mechanisms and a resolution diagram that graphically shows the resolutions in order to allow better understanding and analysis.

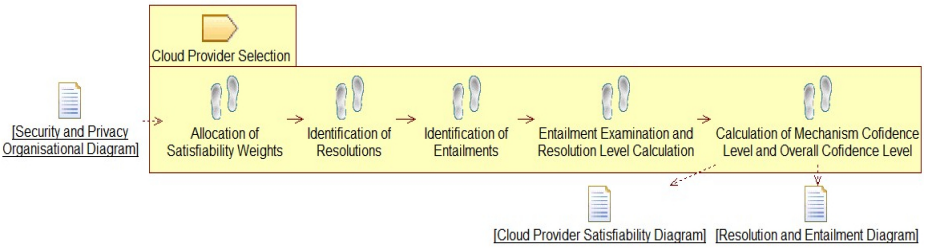

Fig. 1. Extended process definition in SPEM

The next step identifies and analyses entailments, which are the trust conditions that need to be in place to justify trust relationships. This step starts by identifying entailments based on the resolutions identified from the previous step. Therefore resolution diagrams are necessary to identify the entailments. Entailments can be

identified based on the following five cases and graphically represented in an entailments diagram that shows from which resolution they originated: i) Control based resolution requires an entailment that the controller is trusted; ii) Experiential trust requires an entailment that the trustor can trust himself; iii) Reported trust requires an entailment that the reporter is trusted; iv) Normative trust requires an entailment that the environment norm is trusted; v) External trust requires an entailment that the external source of trust is trusted.

At this stage, evidence is collected in order to validate the entailments. However, not all entailments may be valid due to lack of evidence or conflicting evidence. Then the resolution level of a dependency on a cloud provider is calculated by dividing the number of valid dependency entailments with the number of all identified dependency entailments.

$$Dependency\ Resolution\ Level = \frac{Number\ of\ Valid\ Entailments}{Total\ Number\ of\ Entailements}$$

Then summing up all Resolution levels RL multiplied with the Satisfiability levels SL and dividing that sum with the overall number of security and privacy mechanisms m calculate the overall score of a single cloud provider. At the end the provider with the highest overall score level is selected.

$$Overall\ Score = \frac{\sum_{x=1}^{m} RL_x \times SL_x}{m}$$

3.3 Tool Extension

The tool was extended to support the creation and analysis of diagrams related to the trust analysis (Figure 2). In particular, the following diagrams are now supported by the tool: **Resolution diagram**. This diagram graphically shows the resolutions of the dependencies on the cloud providers for the provision of the identified mechanisms. Also, it shows the indirect trust relationships that are implied from the existence of direct trust relationships. **Entailment diagram**. This diagram graphically shows the entailments and from which resolutions originate. Also, it contains a list of valid entailments, which contains the conditions of trust that are true and a list of invalid entailments, which contains the conditions of trust that are not true and as a result further actions are required if the particular cloud provider is selected.

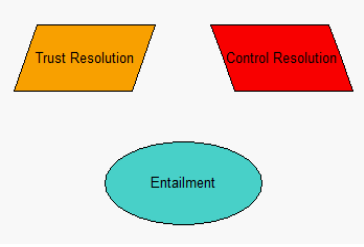

Fig. 2. Trust and control graphical notation

4 Case Study

In our previous work [1] the framework was applied on a real-world case study based on the development of a cloud based solution for the domain of Electronic-Point-Of-Sale (EPOS). The case study reported on a project that took place between the School of Architecture, Computing and Engineering at the University of East London and a company specialising at the provision of EPOS solutions[1]. EPOS Ltd depends on the Cloud Provider to *Provide EPOS Software as Service*, *Manage EPOS Software Licencing* and *Provide Cloud Services*. Figure 3 illustrates the partial result of the analysis that took place as part of that project. In particular, Figure 3 focuses on one of the EPOS Solutions goals, i.e. *Provide EPOS Software as Service* and on two security constraints (Ensure Availability of Software, Ensure Data Confidentiality) and one privacy constraint (Ensure Data Residency) related to that goal. For each one of these constraints, relevant security and privacy measures and mechanisms were identified as shown in the diagram.

Based on the set of security and privacy mechanisms identified, the next activity aims to evaluate how specific service providers satisfy the security and privacy mechanisms identified in the previous step. In the rest of the case study the focus is on five of the security and privacy mechanisms. During the project discussed above, our analysis consisted of the evaluation of three cloud providers[2]. The outcome was the Satisfiability diagram shown in Figure 4.

Following the new activities and language extensions described in the previous section, we have enhanced the analysis of the case study to consider trust relationships during the selection of the cloud provider. For each of the three cloud providers a resolution and entailment diagram is constructed. To keep the length of the paper to a minimum, we have combined in our illustration the resolution and entailment diagrams for each of the cloud providers.

The combined diagram for Cloud provider 1 (CP1) is shown in Figure 5. There are a number of dependencies on the cloud provider to provide the five mechanisms. In particular, the dependencies for *Log Data*, for *Pseudonymisation*, and for *ACID* are resolved by *Reported Trust*. The reporter is the *University Partner 1*, who reports that CP1 is trusted for the provision of *Log Data*, *Pseudonymisation*, and *ACID* respectively. Nevertheless, as stated in the previous section, reported trust resolutions create new dependencies. The new three dependencies are on the *University Partner 1* who is reporting that CP1 can be trusted for the provision of the three mechanisms respectively. Therefore, new resolutions need to be identified for the new dependencies. The resolutions of the new dependencies are *Experiential Trust* as there is previous direct experience with the *University Partner 1*. However, the remaining two dependencies on CP1 for the provision of *VM Isolation* and *Data Tokenization* could not be resolved.

---

[1] For confidentiality reasons we are not allowed to disclose the name of the company so we use the name "EPOS Ltd" to refer to it throughout the paper.

[2] For confidentiality reasons, we are not able to reveal the true identities of the analysed cloud providers. We report however, the real satisfability scores.

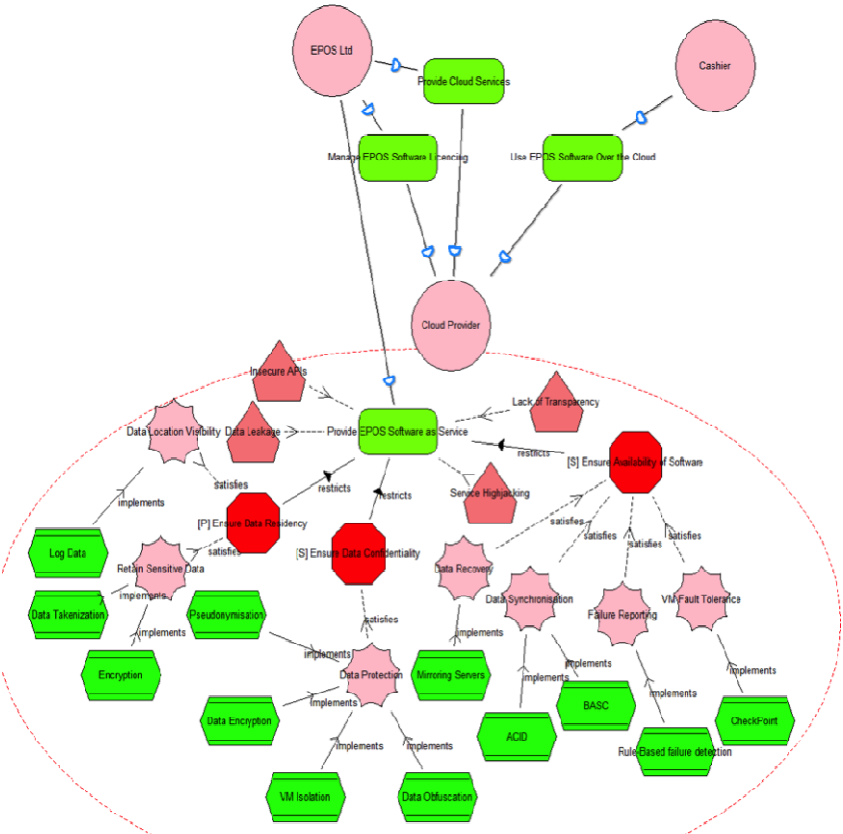

Fig. 3. Security and privacy analysis diagram

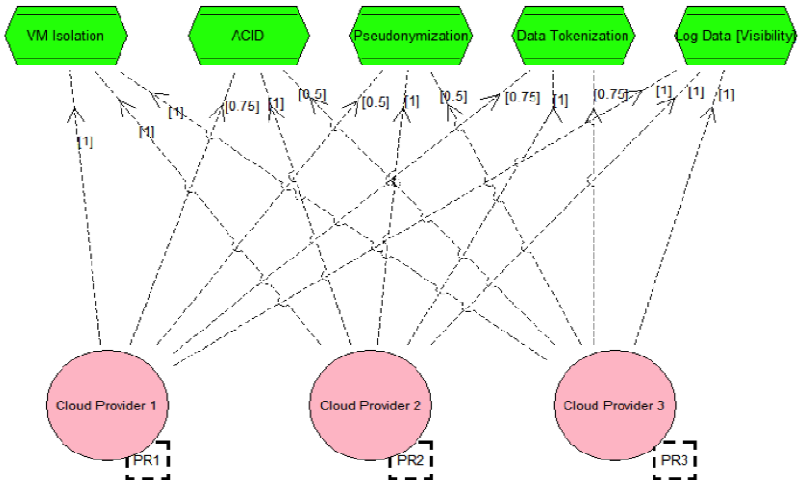

Fig. 4. Satisfiability diagram

Each identified resolution requires an entailment. Therefore, the entailments are identified based on the rules described in the previous section. The entailments that we need to trust ourselves for what the *University Partner 1* is reporting are valid, as there is a long history of collaboration, which make the specific dependencies on the CP1 resolved. Since, there is only one resolution for each of these dependencies with valid entailments then their resolution level is 1. On the other hand, the resolution level of the dependencies without resolutions, and therefore without valid entailments, is 0.

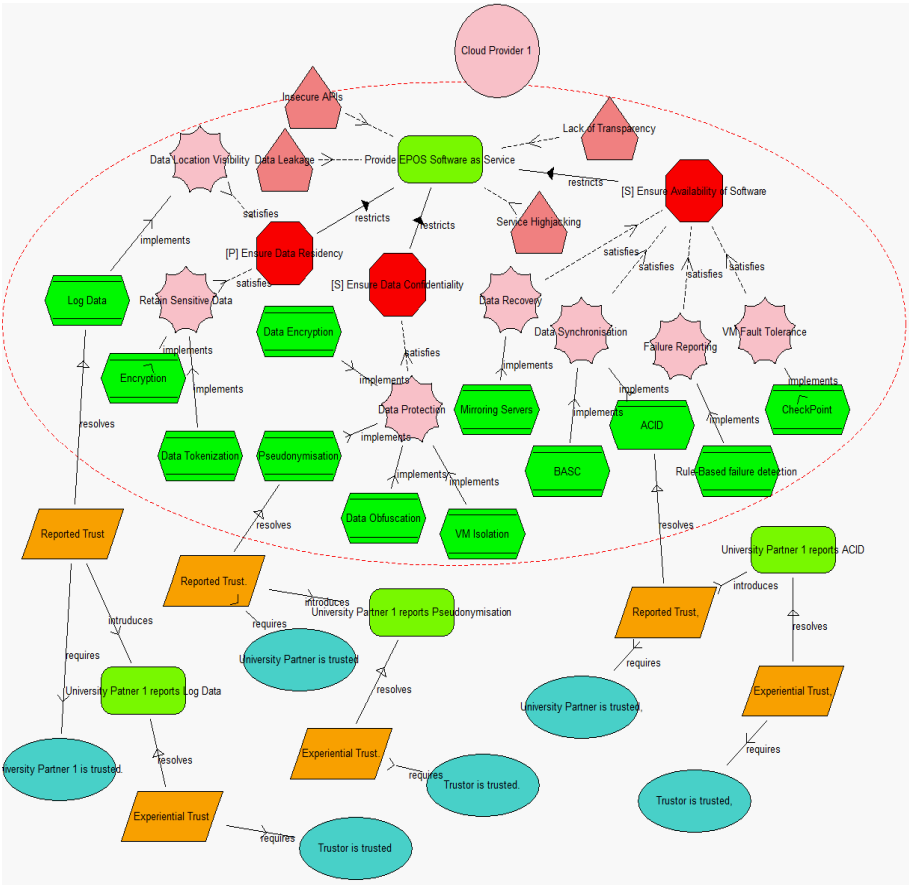

Fig. 5. Resolution and entailment diagram for cloud provider 1

Next, the analysis of the resolutions and entailments of the second cloud provider is carried out and it is shown in figure 6. CP2 is being audited regularly and audits are a form of control on cloud providers as they monitor their performances and services provision. Therefore, all five dependencies on CP2 are resolved through *Audit Control*. Control resolutions though, introduce new dependencies on the controller to successfully audit the cloud provider. The controller in this case is a third party who is performing the audits, and the dependencies on it introduce new uncertainty. These

new dependencies though could not be resolved. However, there is another source of trust to resolve the initial dependencies, which is *Normative Trust*.

Normative trust requires an entailment that the environment norm is trusted. In fact, CP2 is a company of high reputation with a large list of clients. Therefore, the entailments that the norms are trusted are valid. Since, there is one valid and one invalid entailment for the resolutions of each of the dependencies on CP2, the resolution level of those dependencies is 0.5 for each one of them.

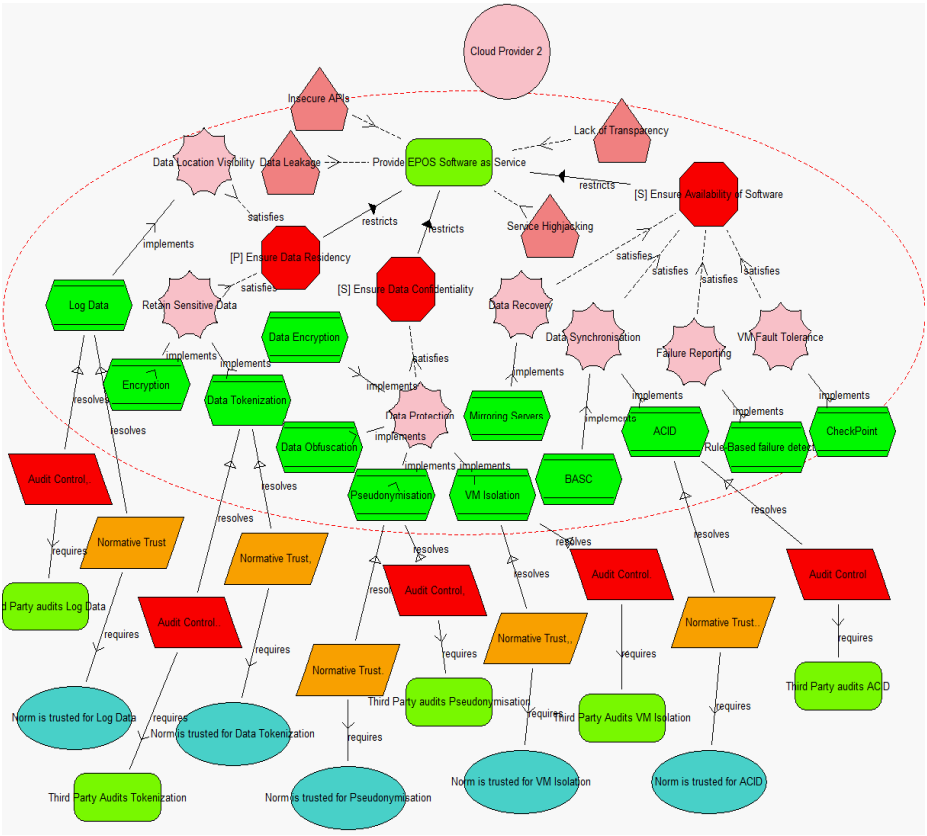

Fig. 6. Resolution and entailment diagram for cloud provider 2

Similarly the resolution and entailment diagram for Cloud Provider 3 (CP3) is constructed and shown in figure 7. CP3 is also under audit checks, but these checks are limited to the provision of *Data Tokenization*, *Pseudonymisation*, and *VM Isolation*. The existence of control resolutions has as result the introduction of new dependencies. We depend on a third party, as the controller, to control the cloud provider, i.e. perform audits checks. Again, though, no resolutions of the dependencies on the third party could be found. However, reported trust resolutions of the dependencies on CP3 for the provision of *Pseudonymisaton* and *VM Isolation* were identified. The reporter is another university partner, named for the purpose of

our explanation as *University Partner 2*. As said before, reported trust resolutions create new dependencies on the *University Partner 2*, who is reporting that the CP3 can be trusted for the provision of *Pseudonymisation* and *VM Isolation*. *Experiential Trust* is then identified to resolve these new dependencies.

The reported trust resolutions require an entailment that the *University Partner 2* is trusted for what is reporting, while the experiential trust resolutions require entailments that we can trust ourselves. Again, there is a long history of collaboration with the *University Partner 2* to trust our judgment, which makes the corresponding entailments valid. As a result, the entailments the University Partner is trusted for reporting that CP3 provides *Pseudonymisation* and *VM Isolation* are valid. The resolution level of the dependencies without resolutions or with resolutions but with out valid entailments is 0, while the resolution level of the dependencies for the provision of *Pseudonymisation* and *VM Isolation* is 0.5 as each one of them has one resolution with valid entailment and one with invalid entailment.

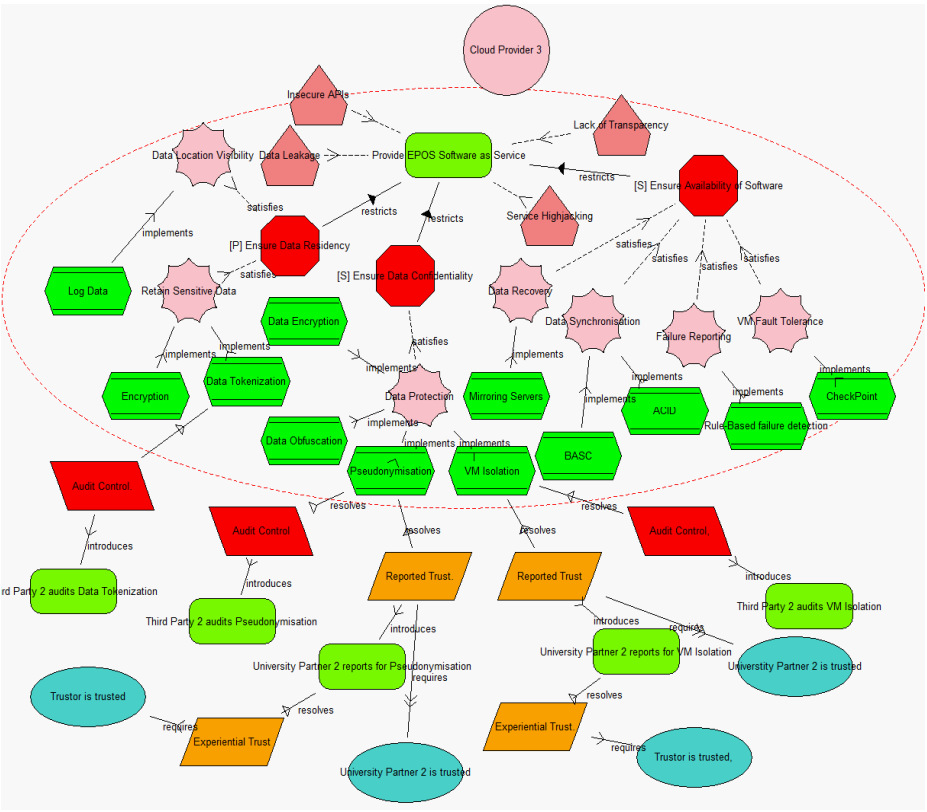

Fig. 7. Resolution and entailment diagram for cloud provider 3

Once the resolution levels of the dependencies on the cloud providers and their Satisfiability levels have been calculated then we can follow the steps of the selection activity, as described in the previous section. For each of the providers we take each

dependency for the provision of a mechanism and multiply its Resolution Level with its Satisfiability Level. Then their sum is divided with the number of mechanisms in order to produce an overall score for each cloud provider as shown below:

$$CP1 = (0*1+1*0.75+1*0.5+0*0.75+1*1) / 5 = 0.45$$
$$CP2 = (0.5*1+0.5*1+0.5*1+0.5*1+0.5*1+0.5*1) / 5 = 0.5$$
$$CP3 = (0.5*1+0*0.5+0.5*0.5+0*0.75+0*1) / 5 = 0.15$$

The provider with the highest score, and therefore preferred, is provider 2.

5 Related Work

The literature has examples of works that focus on security requirements analysis and/or privacy requirements analysis. For example, methods, such as Secure Tropos [3], SQUARE [11], and SecReq [10,12], focus explicitly on security issues, while others, such as PriS [4] and LINDDUN [13], focus on privacy issues. Most of the works related to security focus on the requirements stage. However, none of these works considers their analysis within the context of cloud computing. On the other hand, there are works [14-17] that have been developed based on the idea of cloud computing, but these mostly focus on implementation concerns related to security and privacy in the cloud, and they do not provide a methodology to support the elicitation and analysis of security and privacy requirements and the selection of an appropriate cloud provider based on such requirements. Also, the literature provides examples of works [18-20] in trust analysis on the cloud but, again, these works focus on implementation concerns and trust is considered as a narrow concept that is limited only to security or accountability among others, excluding issues such as shared interests and goodwill.

The work presented here differs from these approaches in that the proposed framework provides explicit support for elicitation and analysis of security and privacy requirements within the context of cloud computing and a systematic process to analyse trust as part of the cloud provider selection process. Assumptions about trust relationships are explicitly identified along with their underlying trust relationships. There is a systematic approach towards better understanding of why there is trust, or there is no trust, in a specific cloud provider.

6 Conclusion

The adoption of cloud computing imposes an unavoidable release of control over valuable assets. As a result trust in the cloud provider is required for a confident adoption of cloud computing and full utilization of its benefits. In this paper we extended previous work to incorporate a new activity that enables to identify direct and indirect trust relationships and to analyse the respective trust assumptions during the selection of a cloud provider.

By applying the extended framework on the case study we have illustrated the applicability and the benefits of our approach. In particular, we identified trust

assumptions that are underlying the successful provision of five specific security and privacy requirements by three potential cloud providers and reasoned about them. However, it does not guarantee that the requirements will be met but that there is confidence in their fulfillment and that the selection of the cloud provider has been justified. If these had been left unexamined then the selection of the cloud provider could have been wrong, as the cloud provider would not have met the security and privacy requirements that we focused on in the case study.

Future work will focus on methods that will further support the process of the validation of entailments. For instance what kind and how much evidence is required for entailments to be valid. We also plan to formalise the work and to enhance the tool to better support our framework.

References

1. Kalloniatis, C., Mouratidis, H., Islam, S.: Evaluating Cloud Deployment Scenarios Based on Security and Privacy Requirements. Requirements Engineering Journal, REJ (2013), http://dx.doi.org/10.1007/s00766-013-0166-7
2. Mouratidis, H., Islam, S., Kalloniatis, C., Gritzalis, S.: A framework to support selection of cloud providers based on security and privacy requirements. To appear in Journal of Systems and Software (2013)
3. Mouratidis, H., Giorgini, P.: Secure Tropos: A Security-Oriented Extension of the Tropos Methodology. International Journal of Software Engineering and Knowledge Engineering 17(2), 285–309 (2007)
4. Kalloniatis, C., Kavakli, E., Gritzalis, S.: Addressing privacy requirements in system design: The PriS method. Requirements Engineering Journal 13(3), 241–255 (2008)
5. Giorgini, P., Mylopoulos, J., Nicchiarelli, E., Sebastiani, R.: Reasoning with Goal Models. In: Spaccapietra, S., March, S.T., Kambayashi, Y. (eds.) ER 2002. LNCS, vol. 2503, pp. 167–181. Springer, Heidelberg (2002)
6. Castelfranchi, C., Falcone, R.: Trust Is Much More than Subjective Probability: Mental Components and Sources of Trust. In: 33rd International Conference on System Sciences, Hawaii (2000)
7. Pavlidis, M., Islam, S., Mouratidis, H., Kearney, P.: Modeling Trust Relationships for Developing Trustworthy Information Systems. International Journal of Information Systems Modelling and Design 5(1) (2014)
8. Pavlidis, M., Mouratidis, H., Islam, S.: Dealing with Trust and Control: A Meta-Model for Trustworthy Information Systems Development. In: Sixth IEEE International Conference on Research Challenges in Information Science, Valencia, Spain (2012)
9. Mollering, G.: The Trust/Control Duality: An Integrative Perspective on Positive Expectations of Others. International Sociology 20(3), 283–305 (2005)
10. Schneider, K., Knauss, E., Houmb, S.H., Islam, S., Jürjens, J.: Enhancing Security Requirements Engineering by Organisational Learning. Requirements Engineering Journal (REJ) 17(1), 35–36 (2012)
11. Mead, N.R., Steheny, T.: Security Quality Requirements Engineering (SQUARE) methodology. SIGSOFT Software Engineering Notes 30(4), 1–7 (2005)
12. Houmb, S.H., Islam, S., Knauss, E., Jürjens, J., Schneider, K.: Eliciting Security Requirements and Tracing them to Design: An Integration of Common Criteria, Heuristics, and UMLsec. Requirements Engineering Journal 15(1), 63–93 (2010)

13. Deng, M., Wuyts, K., Scandariato, R., Preneel, B., Joosen, W.: A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. Requirements Engineering Journal 16(1), 3–32 (2011)
14. Smith Gillam, L., Li, B., O'Loughlin, J.: Adding Cloud Performance To Service Level Agreements. In: 2nd International Conference on Cloud Computing and Services Science (CLOSER), Portugal (2012)
15. Islam, S., Mouratidis, H., Weippl, E.: A Goal-driven Risk Management Approach to Support Security and Privacy Analysis of Cloud-based System. In: Security Engineering for Cloud Computing: Approaches and Tools. IGI Global Publication (2012)
16. Wenzel, S., Wessel, C., Humberg, T., Jürjens, J.: Securing Processes for Outsourcing into the Cloud. In: 2nd International Conference on Cloud Computing and Services Science. SciTe Press (2012)
17. Khajeh-Hosseini, A., Sommerville, I., Bogaerts, J., Teregowda, P.: Decision Support Tools for Cloud Migration in the Enterprise. In: 4th International Conference on Cloud Computing. IEEE Computer Society (2011)
18. Ko, R., Jagadprama, P.: TrustCloud: A Framework for Accountability and Trust in Cloud Computing. In: World Congress on Services (2011)
19. Peterson, G.: Don't Trust. And Verify: Security Architecture Stack for the Cloud. IEEE Security and Privacy (September/October 2010)
20. Pearson, S., Benameur, A.: Privacy, Security and Trust Issues Arising from Cloud Computing. In: 2nd IEEE International Conference on Cloud Computing Technology and Science (2010)