Privacy-Aware Cloud Deployment Scenario Selection[*]

Kristian Beckers[1], Stephan Faßbender[1], Stefanos Gritzalis[2], Maritta Heisel[1],
Christos Kalloniatis[3], and Rene Meis[1]

[1] paluno - The Ruhr Institute for Software Technology – University of Duisburg-Essen
{firstname.lastname}@paluno.uni-due.de
[2] Department of Information and Communications Systems Engineering,
University of the Aegean, Greece sgritz@aegean.gr
[3] Department of Cultural Technology and Communication, University of the Aegean, Greece
chkallon@aegean.gr

Abstract. Nowadays, IT-resources are often out-sourced to clouds to reduce administration and hardware costs of the own IT infrastructure. There are different deployment scenarios for clouds that heavily differ in the costs for deployment and maintenance, but also in the number of stakeholders involved in the cloud and the control over the data in the cloud. These additional stakeholders can introduce new privacy threats into a system. Hence, there is a trade-off between the reduction of costs and addressing privacy concerns introduced by clouds. Our contribution is a structured method that assists decision makers in selecting an appropriate cloud deployment scenario. Our method is based on the privacy requirements of the system-to-be. These are analyzed on basis of the functional requirements using the problem-based privacy threat analysis (ProPAn). The concept of clouds is integrated into the requirements model, which is used by ProPAn to automatically generate privacy threat graphs.

1 Introduction

Cloud computing is a relatively new technology that allows one to build scalable IT-infrastructures that multiple users can access over the network. There is an increasing trend to use clouds to outsource IT-infrastructures and services, but privacy concerns are a major show stopper for the usage of clouds[4,5,6]. The type and number of users that use a cloud and how they can access it heavily differs. The National Institute of Standards and Technology (NIST) defines four deployment models for the cloud: *private*, *community*, *public*, and *hybrid clouds* [1]. A private cloud is exclusively used by

[*] This work was supported in part by the EU project Network of Excellence on Engineering Secure Future Internet Software Services and Systems (NESSoS, ICT-2009.1.4 Trustworthy ICT, Grant No. 256980) and the Ministry of Innovation, Science, Research and Technology of the German State of North Rhine-Westphalia and EFRE (Grant No. 300266902 and Grant No. 300267002).

[4] http://download.microsoft.com/download/F/7/6/F76BCFD7-2E42-4BFB-BD20-A6A1F889435C/Microsoft_Ponemon_Cloud_Privacy_Study_US.pdf

[5] http://www.itu.int/dms_pub/itu-t/oth/23/01/T23010000160001PDFE.pdf

[6] http://www.virtustream.com/company/buzz/press-releases/neovise-research-report

a single organization, and therefore the costs for deployment are high, but the number of additional stakeholders is small, and they are most likely trustworthy, because they belong to the company or are bound to specific contracts. Community clouds are exclusively used "by a specific community of consumers from organizations that have shared concerns." [1]. The costs for this deployment scenario are lower than the costs for the private cloud, because a community of multiple companies shares the costs of the cloud infrastructure. The number of privacy-relevant stakeholders for a community cloud increases in comparison with the private cloud, because additionally there are stakeholders of the other companies of the community that also use the cloud. A public cloud "is provisioned for open use by the general public" [1]. Hence, the number of different stakeholders using a public cloud is larger than the number of those in the community cloud scenario, and, furthermore, it is harder to predict which stakeholders have access to the data in the cloud. But the deployment and maintenance costs for the public cloud are low, because cloud providers can sell their service to a larger number of customers. Hybrid clouds are "compositions of two or more distinct cloud infrastructures" [1].

For companies it is hard to choose the cloud deployment scenario that best fits their needs, when they want to outsource IT-resources. The motivation for outsourcing IT-resources into the cloud is surely the reduction of costs to build and maintain the IT-infrastructure. A barrier for the usage of cloud technology is the number of privacy threats inferred by the usage of cloud technology. As already sketched above, the different cloud scenarios have different properties concerning the costs for deployment and maintenance and the number of additional stakeholders.

In this paper, we present a method that guides requirements engineers and decision makers to decide which cloud deployment scenario best fits the needs of the customer concerning the privacy requirements that exist on the system-to-be. Our method is built upon the problem-based privacy threat analysis (ProPAn) [2] that visualizes possible privacy threats in the system-to-be based on the requirements that the system-to-be shall satisfy, and the facts and assumptions about the environment. The contribution of this paper is an extension of the ProPAn method that embeds the concept of clouds in an modular way into existing requirement models. The ProPAn-tool[7] was extended with wizards that guide the tool-user through the definition of the deployment scenarios and the resources that shall be outsourced into the cloud. From these definitions, diagrams are created, are stored in a UML model, and are used to visualize possible privacy threats that stem from the respective deployment scenario using ProPAn's privacy threat graphs. We applied the method proposed in this paper to a real case study. This case study is concerned with the Greek National Gazette (GNG) that wants to migrate some of its services into the cloud to reduce costs.

The remainder of this work is structured as follows. Section 2 introduces previous work, and Section 3 shows the contribution of this paper. Section 4 discusses related work, and Section 5 concludes.

[7] available at http://www.uni-due.de/swe/propan.shtml

2 Previous Work

Problem frames are a requirements engineering approach proposed by Jackson [3]. We developed the UML4PF-framework [4] to create problem frame models as UML class diagrams, using a UML profile. All diagrams are stored in *one* global UML model. Hence, we can perform analyses and consistency checks over multiple diagrams and artifacts of the software development process.

The first step of the problem frames approach is to create a *context diagram*. A context diagram represents the environment (e.g., stakeholders, other software) in which the machine (i.e., software) shall be built. The context diagram consists of domains and connections between them. Jackson distinguishes the domain types causal domains that comply with some physical laws, lexical domains that are data representations, and biddable domains that are usually people. Connections between domains describe the phenomena they share. Then the problem of building the machine is decomposed until subproblems are reached which fit to problem frames. Problem frames are patterns for frequently occurring problems. An instantiated problem frame is represented as a problem diagram, which, in addition to a context diagram, also contains a requirement. A requirement can refer to and constrain phenomena of domains. Both relations are expressed by dependencies from the requirement to the respective domain annotated with the referred to or constrained phenomena.

ProPAn extends the UML4PF-framework with a UML profile for privacy requirements and a reasoning technique. A privacy requirement in ProPAn consists of two domains of the system, namely a *stakeholder* and a *counterstakeholder*. It states that the counterstakeholder shall not be able to obtain personal information of the stakeholder using the system-to-be. The reasoning technique identifies to which domains personal information of the *stakeholder* can flow and which domains *counterstakeholders* can access. For each privacy requirement, we visualize the information flows starting from a stakeholder s and the access capabilities of the counterstakeholder c in the privacy threat graph $\mathcal{P}_{s,c}$. A privacy threat $\mathcal{P}_{s,c} \subseteq \mathsf{Domain} \times \mathsf{Statement} \times \mathsf{Domain}$ is a directed graph with domains as nodes and edges annotated with statements that refer to and constrain domains of the environment of the machine. In the UML4PF-framework, we distinguish the statement types requirements that are optative properties of the environment after the machine is integrated, facts that are indicative truths about the environment, and assumptions that are indicative properties of the environment that we rely on, but may not hold. As sketched above, we distinguish two kinds of edges in the privacy threat graph $\mathcal{P}_{s,c}$. Edges $(c, st, d) \in \mathcal{P}_{s,c}$ starting from the counterstakeholder c represent that the counterstakeholder has possibly access due to statement st to information about the stakeholder s available at domain d. All other edges $(d_1, st, d_2) \in \mathcal{P}_{s,c}$ have the semantics that due to statement st there is possibly an information flow from domain d_1 to d_2. We are able to derive both types of edges automatically from the UML model using the ProPAn-tool. An access edge (c, st, d) is generated if the statement st refers to or constrains the counterstakeholder c and the domain d. An information flow edge (d_1, st, d_2) is generated if the statement st refers to the domain d_1 and constrains the domain d_2. Details about the graph generation based on requirements can be found in [2] and an extension of ProPAn for the consideration of indirect stakeholders in [5].

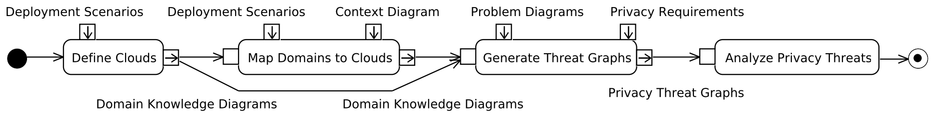

Fig. 1. Process for a privacy analysis of cloud deployment scenarios

3 Method

Our method is presented in Fig. 1 as a UML 2.0 activity diagram [6]. The starting point for our method is a requirements model of the software in problem frames notion (Context Diagram and Problem Diagrams) as a UML model. In the first step, we define the clouds, based on the given Deployment Scenario. The defined clouds are stored in Domain Knowledge Diagrams in the UML model. Based on the given context diagram and the defined clouds, we select the domains that are put into the cloud in the second step of our method. This information is again stored as domain knowledge diagrams in the UML model. To analyze the impact of the modeled cloud deployment scenario on the privacy of the system stakeholders, we apply ProPAn's graph generation algorithm on the given problem diagrams, the given Privacy Requirements, and the domain knowledge diagrams created in the previous steps. The result of this step is a set of Privacy Threat Graphs that visualize the possible privacy threats that exist in the system-to-be. Finally, these graphs are analyzed to decide whether the privacy threats that were identified for the defined cloud deployment scenario are reasonable or not in the last step of our method. The contribution of this paper is the modular integration of clouds into the requirements model in the first two steps of the method, so that these are considered by ProPAn's re-used graph generation algorithms. Additionally, we extended ProPAn's analysis step for the consideration of cloud-specific threats.

Running Example We illustrate our approach using a real-life scenario. In 2010, the Greek National Gazette (GNG) decided to provide a service for electronic submission of the manuscripts sent for publication. To reduce the costs for an own IT-infrastructure for the GNG system, it shall be investigated whether and which cloud infrastructures can be used for the system. The privacy requirement on the GNG system is that the anonymity of the employees involved in the GNG system shall be preserved against external entities. The system is concerned with the electronic submission of manuscripts and the digitalization of sent-in hard copies of organizations. Employees digitalize the hard copies using text scanners and format the documents to issues and paper volumes. Several integrity checks are performed before the documents are published on the online portal of the GNG with the consent of the government's and GNG's general secretary. Using the GNG portal, all Internet users are able to access the published manuscripts. For more details on the GNG system see, [7].

Step 1: Define Clouds In this step, we define the clouds of the deployment scenario we want to analyze. We distinguish three kinds of clouds: *private*, *community*, and *public* [1]. A *hybrid* cloud scenario can be analyzed by defining multiple clouds of different types. For the privacy analysis, we are interested in the number of stakeholders that are able to access the information provided to the cloud. These stakeholders vary for different cloud types. Beckers et al. [8] identified for their PACTS method eight stakeholders

Table 1. Overview of cloud stakeholders and their properties in the cloud deployment scenarios

Group	Stakeholder	Private		Community		Public	
		known	trusted	known	trusted	known	trusted
Provide and maintain cloud	Cloud Provider	yes	maybe	yes	maybe	yes	maybe
	Cloud Administrator	yes	maybe	yes	maybe	maybe	maybe
	Cloud Support	yes	maybe	yes	maybe	maybe	maybe
Use cloud to build services	Cloud Customer	yes	yes	yes	maybe	no	no
	Cloud Developer	yes	maybe	yes	maybe	no	no
Use Services	End Customer	yes	maybe	maybe	no	no	no
Indirect Environment	Legislator	yes	maybe	maybe	maybe	no	no

relevant for clouds and represent their relationship to the cloud using a cloud system analysis pattern. For the method presented in this paper, we derived Table 1 from the cloud system analysis pattern. Table 1 groups the eight stakeholders into four groups. The first group consists of the stakeholders that provide and maintain the cloud. These are the *Cloud Provider* that provides the cloud, and the *Cloud Support* and *Cloud Administrator* that both work for the cloud provider and have directly or indirectly access to the cloud. The second group summarizes the stakeholders that use the cloud to build services. These are the *Cloud Customer*, who deploys his/her infrastructure and services into the cloud of the cloud provider, and the *Cloud Developer*, who works for the cloud customer. The third group consists of the stakeholders that use the services that are run in the cloud. Only the *End Customer* of the cloud customer belongs to this group. The last group is the indirect environment of the cloud. We consider the *Legislator* as a relevant stakeholder, as they are may allowed to access the data of the cloud due to laws, regulations, or bills. The relevant legislators for a cloud are given by the locations of the cloud, cloud provider, cloud customer, and end customer.

Furthermore, Table 1 gives an overview whether these generic cloud stakeholders are known and trusted in the respective deployment scenario. We consider a stakeholder as *trusted* if we can neglect the assumption that the stakeholder introduces privacy issues. When we define a cloud, we first select the deployment scenario we want to consider. For the selected cloud deployment scenario, we have to check the respective entries of Table 1. For each maybe entry a stakeholder has in our selected deployment scenario, we have to decide whether we know/trust the stakeholder in the concrete cloud deployment scenario. Additionally, we have to consider if the other predefined entries are correct for our concrete scenario. For example, if we use a private cloud, we may want to consider cloud customers as possible malicious insiders or to be vulnerable to social engineering attacks. Then we change the trusted entry for the cloud developer of the private cloud scenario from yes to no. Another example is that we know all other cloud customers of a public cloud because the cloud provider makes the list of all its customers publicly available. In this case, we would change the known entry for the cloud customer of the public cloud scenario from no to yes. Note that a yes in the known/trusted column means that all possible instances of the generic stakeholder are known/trusted. Respectively, a no means that we may know/trust some instances of the generic stakeholder, but we do not know/trust all of them. Furthermore, we assume that an unknown stakeholder possibly acts maliciously and cannot be trusted. Hence, we do not allow that a stakeholder is unknown but trusted in a deployment scenario.

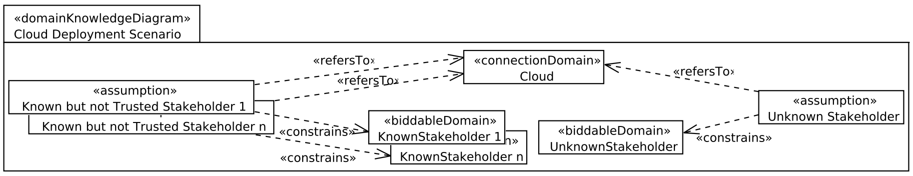

Fig. 2. Cloud definition patterns for the different deployment scenarios

Depending on the yes/no pair that we now have from the adjusted table for each stakeholder, we have to instantiate the stakeholders of the cloud. We distinguish three cases. First, a stakeholder can be known and trusted (yes-yes pair). Then we do not need to instantiate this stakeholder because we do not assume that any privacy issues are caused by him/her. Second, a stakeholder can be known but not trusted (yes-no pair). Then we create an instance of the stakeholder for each concrete stakeholder that we know but do not trust. Third, a stakeholder can be unknown (no-no pair). Then we create an unknown instance. For example, the cloud customer in a private cloud is only the organization for which the software is built, and is hence trusted. In this case, the instantiation is not needed. A community cloud has a set of organizations with shared concerns as cloud customers. These organizations are known, but we may decide that they are not trustworthy. In that case, we have to instantiate the cloud customer with the other organizations that use the cloud. In a public cloud scenario, the other cloud customers are not known in general and hence not trustworthy. In this case, we instantiate the cloud customer with the possibly malicious unknown cloud customer. The other cloud stakeholder are treated analogously.

The instantiated stakeholders and their relation to the cloud of the specific deployment scenario are represented in a domain knowledge diagram that is added to the global UML model. The general form of this domain knowledge diagram is shown in Fig. 2. The domain knowledge diagram represents the assumptions that the instantiated cloud stakeholders (known but not trusted or unknown) are possibly able to access all information that is accessable through the cloud. This is expressed by referring to the cloud and by constraining the cloud stakeholders to be able to access the information. The generation of the domain knowledge diagrams for the concrete deployment scenario can be performed in a computer-aided way on the basis of Table 1, using wizards.

Application to GNG Example The Greek National Gazette decided to evaluate a public and a private deployment scenario for the GNG system. To compare the two deployment scenarios, we created one model for the private and one for the public cloud scenario. In the public cloud scenario, we consider the fictive cloud provider Hulda. As Hulda is located in the USA, we have the USA as a legislator. All other cloud stakeholders are unknown and represented by possibly malicious instances. In the private cloud scenario, the GNG is itself the cloud provider, customer, and end customer. Greece as a legislator was not selected as possibly malicious legislator. Furthermore, we do not consider the cloud support for the private cloud scenario as the cloud administrators additionally shall provide the support. We only consider the cloud administrator and developer as relevant and possibly malicious cloud stakeholders.

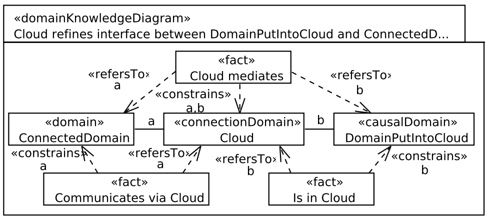

Fig. 3. Domain knowledge diagram introducing a cloud as a connection domain for a domain to be put in the cloud and a domain connected to it.

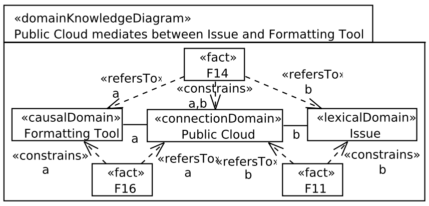

Fig. 4. Domain knowledge diagram introducing the public cloud as a connection domain for the Issue put into the cloud and the Formatting Tool connected to it.

Step 2: Map Domains to Clouds In this step, we have to decide which domains of our context diagram are put into which of the previously defined clouds. At this point, it is not necessary for our method to distinguish the different cloud service levels, such as *software as a service*, *platform as a service*, or *infrastructure as a service* [1]. This is because for the information flow analysis, it does not matter whether a domain is virtualized in the cloud or if the domain represents a cloud service. In any case, the incoming and outgoing information flows have to go through the cloud. If we decide that a domain shall be put into a specific cloud, then this cloud acts as a connection domain that refines all interfaces of the domain and acts as a mediator between the domain that is put into the cloud and the domains which are connected to it. The domain knowledge diagram in Fig. 3 illustrates what this means. The domain knowledge diagram contains three facts. The first fact constrains the Cloud to mediate between the DomainPutIntoCloud and its connected domain ConnectedDomain. The other two facts constrain the ConnectedDomain and the DomainPutIntoCloud, respectively, to use the Cloud as mediator. For each domain that shall be put into a specific cloud, we create a respective domain knowledge diagram on basis of the interfaces described in the context diagram. The creation of these domain knowledge diagrams can again be performed in a computer-aided way, using wizards.

Application to GNG Example The domains that shall be outsourced into a cloud are the lexical domains eDocument, Issue, and Paper Volume. As in both scenarios we only consider one cloud, the needed domain knowledge diagrams only vary in the name of the cloud. The domain knowledge diagram for the introduction of the Public Cloud as a connection domain between the Issue that is put into the cloud and the Formatting Tool that is connected to the Issue in the context diagram is shown in Fig. 4.

Step 3: Generate Threat Graphs The generation of the threat graphs is performed automatically by the ProPAn-tool. But before we can generate the graphs, we have to define a privacy requirement for each biddable domain whose privacy shall be protected. Note that a privacy requirement in the ProPAn method normally consists of a stakeholder whose privacy shall be preserved and a counterstakeholder from whom the stakeholder shall be protected. We extended the graph generation algorithms such that in the case that no counterstakeholder is specified in a privacy requirement, all biddable domains of the requirements model are considered as counterstakeholders. We create the domain knowledge diagrams in the previous steps in such a way that ProPAn's

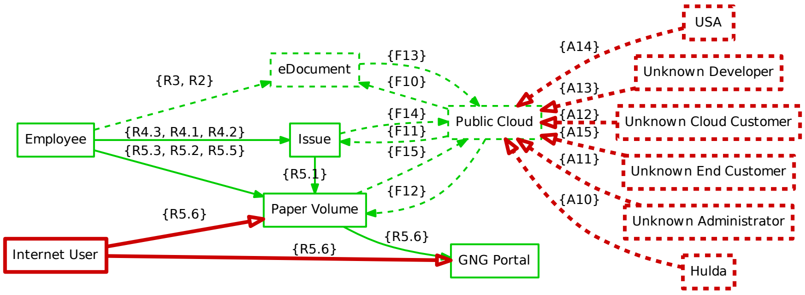

Fig. 5. Privacy threat graph for public cloud deployment

graph generation algorithms automatically consider these diagrams and add respective information flow and access edges to the privacy graphs.

Application to GNG Example The anonymity of the employees of the GNG shall be protected. Hence, we define a privacy requirement with the stakeholder Employee and leave the counterstakeholder undefined. The privacy threat graph for the GNG system with the public cloud deployment scenario is shown in Fig. 5. We draw the domains, information flows, and access edges as dashed lines, which are newly introduced by the cloud deployment scenario. The solidly drawn part of Fig. 5 is the privacy threat graph for the GNG system before the definition of a deployment scenario. The privacy threat graph for the private cloud deployment scenario looks similar to Fig. 5, but only contains the Cloud Developer and Cloud Administrator as cloud stakeholders.

Step 4: Analyze Privacy Threats To analyze the privacy threats that are introduced by the concrete deployment scenarios, we have to check the dashed edges of the respective privacy threat graph. These edges visualize the information flows and access relationships that are introduced by the deployment scenario and did not exist before. By comparing the different threat graphs of the deployment scenarios, we have to decide for the deployment scenario that fits best to the privacy needs of the system-to-be. We distinguish three kinds of edges in our analysis: the information flows directed to the cloud, the flows from the cloud back into the system, and the access edges pointing from the cloud stakeholders, who are considered as counterstakeholders, to the cloud.

First, we have to evaluate the information flows into the cloud with respect to our privacy requirements. We have to investigate which information relevant for the privacy requirements possibly flows into the cloud. If we can assume that there are no information flows relevant for the privacy requirements, then the cloud does not introduce additional privacy threats for the privacy requirements under consideration. Otherwise, we proceed with our method.

Second, we have to investigate whether the access capabilities of the cloud stakeholders of our concrete deployment scenario lead to a violation of our privacy requirements. For each access edge, we have to evaluate which information the stakeholder is able to access and whether this is a threat to our privacy requirements. To assist this evaluation process, we use five cloud-specific threats that are relevant for a privacy analysis. We selected these threats out of the ten that Beckers et al. [8] identified for their PACTS method. The threats and the stakeholders related to them are shown in

Table 2. Privacy relevant cloud threats

Cloud Threat	Provider	Admin.	Support	Customer	Developer	End Customer	Legislator
Insecure API	X	X		X	X	X	
Shared technology	X	X		X	X	X	
Malicious insider	X	X	X	X	X	X	X
Hijacking	X	X	X	X	X	X	X
Data location							X

Table 2. We use Table 2 to check for each cloud stakeholder in the privacy threat graph under consideration if the associated cloud threat has to be considered for our concrete deployment scenario. We structured the five threats into three groups. The first group represents threats that stem from the cloud technology. It consists of the threats *Insecure API* and *Shared technology*. The threat *Insecure API* refers to possibly insecure interfaces and APIs that are provided by the cloud provider to cloud administrators, cloud customers, cloud developers, and end customers. The provided interfaces have to ensure correct authentication, access control, encryption, and activity monitoring to protect against accidental and malicious attempts to access the cloud. The threat *Shared technology* arises because multiple services use the same hardware in a cloud infrastructure. As hardware is often not designed to offer strong isolation properties, there can be unintended information flows or possibilities to access information. Examples are shared CPU caches and hard disks. The second group represents malicious behavior of the cloud stakeholders. It consists of the threats *Malicious insider* and *Hijacking*. The threat *Malicious Insider* considers the misuse of a cloud stakeholder's capabilities to access information from the cloud or to get it from other cloud stakeholders for themselves or to provide the information to others. The threat *Hijacking* refers to attacks that try to steal or guess credentials and passwords of user accounts or cloud services. A hijacked account or service can be used to access the information provided by it and the information that it will provide in the future. We assume that each cloud stakeholder is able to perform an attack related to this threat group. The last group only consists of the threat *Data location*. The threat *Data location* refers to the location of the cloud servers. Depending on the location of the cloud servers, different legislators may have the right to access the information stored and processed on the servers.

Third, we have to consider if the introduced cloud adds an information flow feedback into the system. If multiple domains are put into a cloud, then it is possible that there are information flows between the domains in the clouds and those connected to them. These unintended information flows could stem from the *Shared technology* threat that we discussed above. From the information flow feedback, it is possible that counterstakeholders are able to access more information than they were able to access before the cloud was introduced.

On basis of the analysis of the generated privacy threat graphs, we have now to decide if the privacy threats introduced by the concrete deployment scenarios are acceptable or if respective countermeasures have to be implemented. The costs for the realization of respective countermeasures have to be compared with the cost reduction that is expected by the usage of the cloud infrastructure. This comparison can assist decision makers to select a concrete deployment scenario.

Application to GNG Example If we compare the privacy threat graph of the GNG system without a cloud with those graphs for the private and public cloud deployment scenario, then we observe that the complexity of these graphs is significantly increased. The graphs for the different deployment scenarios only differ in the number and kind of counterstakeholders that have access to the cloud. Hence, the analysis of the information flows going into the cloud and coming out of the cloud is the same for both scenarios, but the analysis of the access edges has to be done separately for both scenarios. The information flows into the cloud in both scenarios introduce a privacy threat to the anonymity of the employees that was previously not existent in the system-to-be. The new threat is that an employee's anonymity is possibly revealed by the collection of the information which and how documents are changed over the time. Using this meta information it is possible to reduce the set of employees who possibly performed the changes on a document. This threat stems from the logging mechanisms of clouds and the possibility to eavesdrop the connection to the cloud. For the GNG system, we do not expect relevant information flow feedback from the cloud to other domains. Such a flow would provide additional information to the Internet user which is able to access the published paper volumes using the GNG portal (see Fig. 5). We do not consider such a flow as relevant because the paper volumes are checked by an employee before they are uploaded to the GNG portal.

The public cloud deployment scenario has four unknown cloud stakeholders, namely the developer, the cloud customer, the end customer and the administrator (see Fig. 5). As all these stakeholders are potentially malicious, we have to consider all threats related to them in Table 2. These threats are *Insecure API*, *Shared technology*, *Malicious insider*, and *Hijacking*. We also assume that the fictive cloud provider Hulda possibly causes these threats. Furthermore, we know due to the *Data location* threat, that the USA is possibly able to access the data stored in the cloud. It is possible to implement countermeasures that mitigate these threats, but their implementation is expensive and the performance advantages of the cloud are reduced by their implementation. In the private cloud deployment scenario, we have only two cloud stakeholders. These are the developers and administrators of the private cloud. Due to Table 2, we have to consider whether these stakeholders will use insecure interfaces or APIs, or shared technology issues to access information they are not allowed to have. Furthermore, we have to investigate whether the administrators and developers have to be considered as malicious insiders providing sensitive information to others or use their privileges to hijack accounts or services. None of these threats can be neglected, but as the administrators and developers are employed by the GNG, it is easier to implement respective countermeasures as in the public cloud scenario.

To sum up, the privacy threats introduced by the public cloud scenario are the most critical ones. That is because it is not predictable who is able to access the information in the cloud, as there are multiple unknown and possibly malicious cloud stakeholders. For the private cloud scenario, we have only two newly introduced counterstakeholders, namely the cloud administrator and developer. As these two stakeholders are employed by the GNG, we are able to assume that these two cloud stakeholders are not malicious or we can easily implement countermeasures for the threats they introduce. Hence, the recommendation for the GNG system is to select the private cloud deployment scenario.

4 Related Work

An early analysis of privacy threats is necessary for all information systems. The introduction of clouds introduces further stakeholders and information flows that possibly lead to privacy threats depending on the selected cloud deployment scenario.

Kalloniatis et al. [7] propose a process for the evaluation of cloud deployment scenarios based on security and privacy requirements. The process identifies organizational entities and their needs and defines the security and privacy requirements for the system. Then cloud deployment scenarios are described and analyzed. On the basis of this analysis a deployment scenario is selected. The method is based on the PriS method [9] and Secure Tropos [10]. The processes described by Kalloniatis et al. is broader than the one described in this paper, as it analysis security and privacy in combination. But the process is at many points relatively abstract. We propose in this paper a more detailed method for the analysis of cloud-specific privacy threats.

The LINDDUN-framework proposed by Deng et al. [11] is an extension of Microsoft's security analysis framework STRIDE [12]. In contrast to ProPAn, the system to be analyzed is modeled as a data flow diagram (DFD), which has to be set up carefully. ProPAn is based on a problem frames model which is assumed to be already existing and which is systematically created using the problem frames approach [3].

The topic of cloud migration has already been discussed in various papers e.g., [13,14]. These works focus mainly on the financial costs of a cloud migration and identify privacy as a restricting factor for a migration. But in contrast to our work, they do not provide guidance for the identification of privacy issues that have to be considered when migrating to the could.

In contrast to the above methods, we integrate cloud deployment scenarios into a requirements model in a modular way to perform a privacy analysis and provide tool-support. The definition of deployment scenarios using separate diagrams allows us to evaluate different deployment scenario without effecting other artifacts.

5 Conclusion and Future Work

In this paper, we presented a privacy-aware decision method for cloud deployment scenarios. This method is built upon the ProPAn and PACTS method. The first step of the presented method is the definition of the clouds used in concrete deployment scenarios and their cloud stakeholders. Then we decide which domains shall be put into which defined cloud. We capture the defined clouds, cloud stakeholders, and the relation between existing domains and the defined clouds in domain knowledge diagrams. We can apply ProPAn's graph generation algorithms on these domain knowledge diagrams together with a given model of the functional requirements in problem frames notation. The resulting privacy threat graphs are then analyzed to decide which deployment scenario best fits the privacy needs in the last step of the method. To support our method, we extended the ProPAn-tool with wizards that guide the user through the definition of the deployment scenarios and that automatically generate the corresponding domain knowledge diagrams. The proposed method scales well due to the modular way in that the relevant knowledge for the cloud deployment scenarios are integrated into the requirements model and the provided tool-support. Our contributions are:

- A systematic method to analyze the privacy impact of cloud deployment scenarios on a concrete software that shall be built.
- An overview of the kinds of cloud stakeholders that have to be considered in the different deployment scenarios.
- A modular way to add the knowledge relevant for clouds into the problem frames requirements model using domain knowledge diagrams.
- A slight modification of ProPAn's graph generation that considers all biddable domains as possible counterstakeholders if no counterstakeholder is defined.
- A mapping of the cloud stakeholders to cloud-specific threats that they can cause.

The application of ProPAn and the extension presented in this paper to an industrial-size case study and an empirical evaluations are part of our future work.

References

1. National Institute of Standards and Technology: The NIST definition of cloud computing (2011)
2. Beckers, K., Faßbender, S., Heisel, M., Meis, R.: A problem-based approach for computer aided privacy threat identification. In: Annual Privacy Forum 2012. Volume 8319 of LNCS., Springer (2014) 1–16
3. Jackson, M.: Problem Frames. Analyzing and structuring software development problems. Addison-Wesley (2001)
4. Côté, I., Hatebur, D., Heisel, M., Schmidt, H.: UML4PF – a tool for problem-oriented requirements analysis. In: Proceedings of RE, IEEE Computer Society (2011) 349–350
5. Meis, R.: Problem-Based Consideration of Privacy-Relevant Domain Knowledge. In: Privacy and Identity Management for Emerging Services and Technologies. Volume 421 of IFIP Advances in Information and Communication Technology. Springer (2014)
6. UML Revision Task Force: OMG Unified Modeling Language: Superstructure. (May 2012)
7. Kalloniatis, C., Mouratidis, H., Islam, S.: Evaluating cloud deployment scenarios based on security and privacy requirements. Requir. Eng. **18**(4) (2013) 299–319
8. Beckers, K., Côté, I., Faßbender, S., Heisel, M., Hofbauer, S.: A pattern-based method for establishing a cloud-specific information security management system - establishing information security management systems for clouds considering security, privacy, and legal compliance. Requir. Eng. **18**(4) (2013) 343–395
9. Kalloniatis, C., Kavakli, E., Gritzalis, S.: Addressing privacy requirements in system design: the PriS method. Requir. Eng. **13** (August 2008) 241–255
10. Mouratidis, H., Giorgini, P.: Secure tropos: a security-oriented extension of the tropos methodology. International Journal of Software Engineering and Knowledge Engineering **17**(2) (2007) 285–309
11. Deng, M., Wuyts, K., Scandariato, R., Preneel, B., Joosen, W.: A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. RE (2011)
12. Howard, M., Lipner, S.: The Security Development Lifecycle. Microsoft Press, Redmond, WA, USA (2006)
13. Khajeh-Hosseini, A., Sommerville, I., Bogaerts, J., Teregowda, P.: Decision support tools for cloud migration in the enterprise. In: IEEE Int. Conf. on Cloud Computing (CLOUD). IEEE Computer Society (July 2011) 541–548
14. Hajjat, M., Sun, X., Sung, Y.E., Maltz, D., Rao, S., Sripanidkulchai, K., Tawarmalani, M.: Cloudward bound: Planning for beneficial migration of enterprise applications to the cloud. In: Proc. of the ACM SIGCOMM Conf. ACM, New York, NY, USA (2010) 243–254