Information Accountability supported by a Provenance-based Compliance Framework

Rocío Aldeco-Pérez and Luc Moreau
raap06r@ecs.soton.ac.uk, l.moreau@ecs.soton.ac.uk
School of Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ, UK

1 Introduction

Recent technology advances allow organisations to create a wide range of on-line facilities that offer personalised services to their customers. This personalisation is obtained by requesting personal information to the users, that later should be used under a set of rules that describes which processing should be performed over this data. However, if these usage rules are not followed, personal data could be exposed and used against the interest of its owner. For that reason, it is important that individuals or institutions can be held *accountable* for information misuse.

A lot of research is focused on developing better techniques to avoid information misuse by restricting access to information. However, access restriction alone can not properly solve this problem on the Internet, where information is widely available. In this case, when information becomes accessible, verifying the correct use of it after it was processed is also important. Evidence of the importance of this issue can be seen in legislative frameworks related to the use of private information, such as the Data Protection Act (DPA) [2], which establishes restrictions on the way that UK organisations may process private information. The notion of accountability is not restricted to individual's private data. It also applies to commercial data, to sensitive data, or even to data available under licenses that restrict the usage of data, or mandates actions to be performed.

An *accountable system* is defined [6] as a system that makes information usage transparent so that it can be determined later whether the use of such information is appropriate or not under a given set of rules. Weitzner *et al.* have argued that provenance could be used in the creation of accountable systems helping users to answer questions related to the processing of information [6]. Provenance consists of causal dependencies between data and events explaining what contributed to a result in a specific state [5]. If provenance of data is available, processing becomes transparent since the provenance of data can be analysed against usage rules to decide whether processing was performed in compliance with such rules [1].

A provenance-aware system [5] describes all the steps and data derivations involved in its execution, in the form of *process documentation*. Information related to a specific processing can be obtained from such process documentation by means of a provenance query [3], the result of which can be analysed to decide if the processing was performed in accordance with a set of usage rules. Examples of data processing requirements that can be checked include: *(i)* determining if the processing of some data is compatible with the purpose for which the data was captured and *(ii)* asserting that only information to be processed was captured.

In order to support this vision, we have created a provenance-based Compliance Framework. Such a framework consists of a view of past processing of information, a representation of the rules that processing should follow and a comparison stage in which the past processing is analysed against the processing rules. By using this framework, it is possible to decide if an application processed information in compliance to the predefined information usage rules. The framework components are platform-independent and reusable, as they can be applied to different systems to verify different rules. At the same time, our framework could be used to create automatic auditing tools for verifying diverse policies over data processing.

The aim of this paper is to present this Compliance Framework; specifically, the contributions of this paper are: *(i)* The Compliance Framework components, which comprises the Processing View and the Usage Rules Definition and *(ii)* The Compliance Framework verification process, in which its components are compared to check the correct processing of information.

2 Compliance Framework

The Compliance Framework consists of one view, one data representations and one comparison stage. Firstly, the Processing View, which is a view of a selection of process documentation related to the data in which we are interested. This view is represented as a provenance causal graph and can be seen as a specialisation of the Open Provenance Model [4]

Secondly, the Usage Rules Definition, which is a representation of a set of rules that should be followed while processing data. This definition contains the conditions under which processing is valid over a set of data, and it needs to be generated by any organisation or person that wants to monitor the use of a set of processing policies.

Both components are represented by casual directed acyclic graphs, as can be seen in Figure 1. Their vertices represent data and their edges represent casual relationships between them. Data includes *purposes* (p_i), which are the intentions for which a set of data is to be collected, *tasks* (t_i), which are the processes performed over data, *data* that is to be collected (D_{Ci}) and processed (D_{Ii}), and *results* (r_i), which are the output of a task.

The Processing View, which is presented in Figure 1(a), represents a general data processing life cycle. In this view, the relationships' names are presented in past tense expressing the fact that these actions happened in the past.

Accordingly, the life cycle began when an application requested a set of data from a user declaring the purpose for which such a set **was acquired** (*collection purpose*). After checking the application purpose, the user sent the requested set of data (*collected data*). The goal of the application was to achieve the collection purpose. For that reason, a *task* **was initiated by** the related purpose (now called, *processing purpose*). Such a task **used** a set of data that was a **subset of** the collected data (now called, *used data*). Note that the task could have used the previous collected set of data or a subset of it. Later, the task was executed with the used data as input and **generated** *results*. Note that, such results could have been used as collected data in the execution of a new task.

The Usage Rules Definition, which is presented in Figure 1(b), represents the processing rules that an application should follow while users' information is being processed. In this definition, the relationships' names are presented in present tense expressing the fact that this actions are expected to happen. Such a definition contains a set of purposes from which users' data is to be collected, the set of tasks that **are initiated by** a specific purpose and a set of data that a specific task **uses** in its execution. Note that this framework component contains more than one purpose and each purpose has more than one task. However, one task could be related to more than one purpose. For example, the task "create inventory" could be related to the purpose "on-line sales" and the purpose "create stock". At the same time, each task has one and only one set of data. However, one element of a set could be contained in more than one set. For example, the element "name" could be in the set related to the task "send medicine" and also in the set related to the task "create inventory".

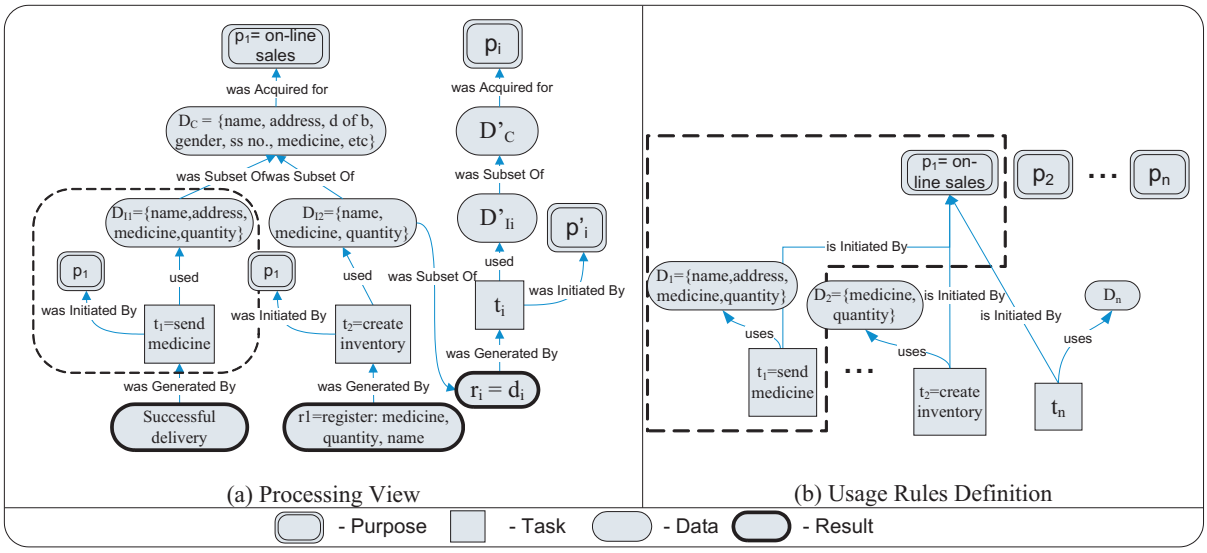

Figure 1: Compliance Framework Components

The last component is the comparison stage. In this stage, it is possible to verify several information usage requirements by using the already described components. Due to space restrictions we will focus on only one: Processing of data is compatible with the purpose for which it was captured. The verification of this requirement is performed by comparing the past processing, described in the Processing View, against the expected processing, represented by the Usage Rules Definition. To do this, information related to each requirement is extracted from the Processing View in form of a subgraph. Then, the data and the relationships of such a subgraph are compared with the content of the Usage Rules Definition. In the next section, we briefly explain how the presented requirement can be verified using the Compliance Framework in a practical example.

3 Application Example

Consider the following scenario: Alice is trying to get pregnant, and so has decided to take a fertility treatment. She decided to buy her treatment using the web page of a pharmacy. In order to get her treatment, she needs to provide her *name*, *address*, *date of birth*, *gender*, *social security number*, *the number of her clinic* and *her doctor's name*. At the same time, but unrelated to her attempt to get pregnant, she applies for a job in the same pharmacy - and she is rejected. She suspects that the pharmacy may have checked its records related to her name and realised that she has plans to have a family, and as a result marked her as a high risk employee for expensive maternity costs. If this is true, the company obviously misused Alice's personal information. When she sent her personal information to the pharmacy, she did that with the purpose of getting her treatment. From the point of view of the company, the purpose is "on-line sales". The on-line sales purpose could include verifying the existence of the medicine, charging the amount to her card, sending the medicine to her home and even creating a record of the product's sales. However, the types of data used in each of these tasks are different. For example, the company can create a record of the monthly sales, which includes the *medicine's name* and the *quantity sold*. Nevertheless, such a record cannot contain the name of the people that bought

that item, as the purpose of that record is not to identify each person. Next, we will use the framework components presented in Figure 1 to show how they can be effectively used to find misuse of information in this scenario.

We want to verify the requirement that data used in the performing of a task should only be data stated for that initial purpose, as defined in the Usage Rules Definition. To verify the compliance of this requirement it is necessary to extract some elements from the Processing View. In this case, given a result, we extract its corresponding task and the corresponding purpose and used data related to such a task. These can be seen in the dotted-line rectangle in Figure 1(a). These extracted elements can be seen as a subgraph of this component. In this case, this subgraph expresses that a "successful delivery" result was generated by the task "send medicine", which was initiated by the purpose "on-line sales" using the set of data D_{I1}. At this point, the processing of the data that was collected from Alice is transparent. We also need to check that if the processing was in accordance with the established rules. To check that, we only need to find the extracted subgraph in the Usage Rules Definition. This means, we need to establish that all the vertices and their contents, and all the edges with their relationship names, are contained in the Usage Rules Definition. In this case, as can be seen in the dotted-line rectangle in Figure 1(b), the extracted graph is contained in such a graphical definition, so that we can say that the processing of Alice's data was in compliance.

Now, we will analyse the next task. The task "create inventory" takes the set of data that contains the elements *name*, *medicine* and *quantity* from all the users that bought medicine on this pharmacy. The result of this task is a record that contains such elements. If we take the subgraph related to this task and compare it with the Usage Rules Definition, we see that there is a difference. The element *name* should not have been used. The reason is that an inventory of a pharmacy does not need the name of the costumers, just the item that was sold and the quantity of it. In that case, we can say that the processing of Alice's information was not in compliance, and therefore, such an inventory could be used against her interest.

4 Conclusions

In this paper, we have presented the provenance-based Compliance Framework by explaining its components. We have also explained how this framework can be applied to an application example over a very common scenario, on-line shopping. With this example, we explained how our framework helps us verify one requirement related to the processing of information. However, more requirements can be verified that are not presented for space restrictions.

Our work demonstrates that by using the Compliance Framework, individuals or institutions, which used information in a different manner from the stated, can be held *accountable* for misuse. Our framework comprises one view and one definition that are platform independent and reusable, and one comparison stage that is easy to implement in an automatic way. For these reasons, we have developed a prototype of it to evaluate its performance in different application areas and in the verification of different usage requirements.

References

[1] R. Aldeco-Pérez and L. Moreau. Provenance-based Auditing of Private Data Use. In *International Academic Research Conference, Visions of Computer Science*, September 2008.

[2] HomeOffice. Data Protection Act, 1998.

[3] S. Miles. Electronically querying for the provenance of entities. In *Proceedings of the International Provenance and Annotation Workshop IPAW*, pages 184–192. Springer, November 2006.

[4] L. Moreau, J. Freire, J. Futrelle, R. E. McGrath, J. Myers, and P. Paulson. The open provenance model: An overview. In *Provenance and Annotation of Data and Processes*, pages 323 – 326, 2008.

[5] L. Moreau, P. Groth, S. Miles, J. Vázquez, J. Ibbotson, S. Jiang, S. Munroe, O. Rana, A. Schreiber, V. Tan, and L. Varga. The provenance of electronic data. *Communications of the ACM*, 51(4):52–58, April 2008.

[6] D. J. Weitzner, H. Abelson, T. Berners-Lee, J. Feigenbaum, J. Hendler, and G. J. Sussman. Information accountability. *Commun. ACM*, 51(6):82–87, 2008.