Towards a Policy Language Enforcing Privacy by design in Distributed Services

Chinmayi Prabhu Baramashetru, S. Lizeth Tapia Tarifa, Olaf Owe, and Nils Gruschka

Department of Informatics, University of Oslo, Oslo, Norway
{cpbarama,sltarifa,olaf,nilsgrus}@ifi.uio.no

Abstract. From the very outset of the digital era, the protection of personal data against unauthorised usage and distribution has been one of the most significant challenges in distributed services. In this context, the EU enforced the general data protection regulation (GDPR), which imposes strict regulations to protect EU citizen's data in order to return the control back to data owners. In current GDPR implications, there is a distinct challenge in the adaptation of these requirements into technical solutions integrating data privacy in system design. In this paper, we identify issues in the existing service model architecture and motivate a formal policy language using language-based constructs to demonstrate built-in abilities for data protection by design.

1 Introduction

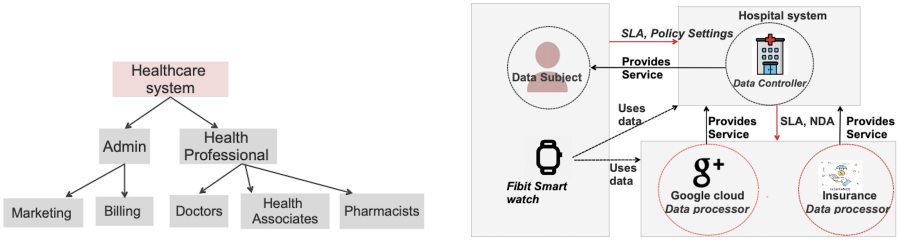
In today's digital world, every individual is part of an ecosystem that manipulates personal data to provide services to the customers. Users give information through a set of internet-connected devices (e.g., automated systems, wearables, voice assistant systems, etc.) in a distributed setup. Information is generally collected, stored, transferred, and used by service providers for purposes beyond the user's vision. There are instances where these service providers traded with personal data without explicit consent from the data subjects [1, 5]. The philosophy underpinning the establishment of GDPR [3] was to safeguard data subject's personal data. However, the GDPR document are mainly expressed in generic terms, and it does not provide clear evidence of how they should be systematically implemented in distributed environments. Therefore providing useful ways to be GDPR compliant is an open research challenge. Another issue in today's standard setup is power imbalance between the data subjects and data controllers. Often there is no room for negotiation for users to opt-in or opt-out from various add-ons that require additional data processing and affect their privacy preferences. Data subjects have to accept a broader consent without having means to express their preferences over the purpose of collection, location of data processing, transferring data to third parties, retention time concerning any personal data collected. It is hard to protect data subject rights with the existing power imbalance between users and stakeholders.

We envisage a policy language that upholds a *user-centric* approach of expressing privacy preferences, at different granularities, that can serve as the basis to guarantee compliance of policies across multiple stakeholders. We address this

issue from a programming language perspective: How to be proactive and design a language to stop data protection violations? Existing programming languages do not fully support GDPR requirements. We present ideas towards a privacy policy language with built in abilities to meet the GDPR requirements. We define policies as sets of tuples with five attributes imposing restrictions on entities that may access personal data for certain purposes, duration and location. We use static and run-time analysis to enforce such policies and associate them to a high-level modeling language oriented towards distributed systems. We also formalize the notion of policy compliance to define policy inheritance and policy inclusion.

2 Policy language conforming GDPR

In this section, we investigate gaps in the existing service model architecture and present how our language will facilitate data protection by design. We briefly mention the challenges associated with this research undertaking.



(a) Entity Hierarchy (b) Service model architecture

Fig. 1: Policy language Features

Flexible granularity of policies: While there are multiple languages designed to express user privacy preferences [6] [2] [4], there is still a lack of vital features to support GDPR requirements to protect data subjects rights. In our language, we envisage flexible granularity of policies allowing user-centric *filtering of information*. This means, in addition to expressing top-level policies for predefined purposes and entities, data subjects can express their obligations/exceptions using filtering options over policy elements. This offers flexibility over the granularity of policies. Consider various hierarchical structure, as the one given in Figure 1(a), a fine-grained policy could look like "I want to block ***Usage*** access on my ***Fitbit data*** for ***all purposes except special treatments*** from all health professional except for my ***cardiologist*** within ***Europe***". A policy language needs to consider crucial elements such as *purpose*, *entity*, *location*, *retention*, and *access* expressed as policy tuples to help users define who can collect, store, use, and transfer their data. The language provides a filtering option over the hierarchies used to express such privacy preferences, namely entity (e.g., doctor), purpose (e.g., special treatment), and location (e.g., Europe). We aim to formalize a language by defining taxonomies over these attributes using hierarchical structures as shown in Figure 1(a). We plan to carefully formalize

the language so that we don't allow or deny unjust provision of data.

Policy Compliance across multiple stakeholders: Consider the example of Fitbit service model architecture from Figure 1(b) where health wearables continuously collect sensitive information and constitute a digital link between the patients and doctors. For instance, a cardiologist might ask her heart patients to wear Fitbit for continuous monitoring, and if the sensors detect any problem, the doctor will be immediately notified. There is a trove of information accessible by various stakeholders with no adequate data protection tools. The main research problem associated with existing multi-stakeholder architectures is policy compliance. In the example model shown in the figure 1(b), DS and DC are bound by SLAs (service level agreements), data usage agreements, and data protection agreements. But the agreements are usually in natural language statements and stakeholders might tamper with the agreements to benefit their businesses. Hence, we formalize the notion of policy compliance which statically analyses the compliance across all the stakeholders.

3 Conclusion and Future Work

We have made a brief investigation of the current service model architecture and discussed gaps in existing policy languages and presented ideas towards a new user-centric policy language based on five vital elements embedded in policy tuples, featuring flexibility over the granularity of policies and achieving compliance between various stakeholders. We consider the policy language as work in progress. However, we do have ongoing implementations on formalizing the language and building the compliance checks based on the previous investigations with language-based constructs enforcing privacy by design to comply with GDPR requirements. We are planning to integrate the policy language with a modeling language for asynchronously communicating distributed systems.

References

1. As Facebook raised a privacy wall, it carved an opening for tech giants - The Netherlands New York Times. `https://www.nytimes.com/2018/12/18/technology/facebook-privacy.html`. (Accessed on 02/12/2021).
2. M. Azraoui, K. Elkhiyaoui, M. Önen, K. Bernsmed, A. S. De Oliveira, and J. Sendor. A-ppl: an accountability policy language. In *Data privacy management, autonomous spontaneous security, and security assurance*, pages 319–326. Springer, 2014.
3. European Parliament and Council. Regulation (EU) 2016/679 of the European parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/EC (general data protection regulation) with EEA relevance). `http://data.europa.eu/eli/reg/2016/679/oj/eng`, 2016.
4. M. Henze, J. Hiller, S. Schmerling, J. H. Ziegeldorf, and K. Wehrle. Cppl: Compact privacy policy language. In *Proceedings of the 2016 ACM on Workshop on Privacy in the Electronic Society*, pages 99–110, 2016.
5. D. Lyon. Surveillance, Snowden, and big data: Capacities, consequences, critique. *Big data & society*, 1(2):2053951714541861, 2014.
6. J. Yang, K. Yessenov, and A. Solar-Lezama. A language for automatically enforcing privacy policies. *ACM SIGPLAN Notices*, 47(1):85–96, 2012.