<http://www.diva-portal.org>

Postprint

This is the accepted version of a paper presented at *IEEE International Conference on Software Architecture, ICSA 2019, Hamburg, Germany*.

Citation for the original published paper:

Colesky, M., Demetzou, K., Fritsch, L., Herold, S. (2019)
Helping Software Architects Familiarize with the General Data Protection Regulation
In: *2019 IEEE International Conference on Software Architecture Companion (ICSA-C)* (pp. 226-229). IEEE
<https://doi.org/10.1109/ICSA-C.2019.00046>

N.B. When citing this work, cite the original published paper.

© 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:kau:diva-71838>

Helping Software Architects Familiarize with the General Data Protection Regulation

Michael Colesky
Radboud University
Nijmegen, The Netherlands
mrc@cs.ru.nl

Katerina Demetrou
Radboud University
Nijmegen, The Netherlands
k.demetrou@cs.ru.nl

Lothar Fritsch
Karlstad University
Karlstad, Sweden
lothar.fritsch@kau.se

Sebastian Herold
Karlstad University
Karlstad, Sweden
sebastian.herold@kau.se

Abstract—The General Data Protection Regulation (GDPR) impacts any information systems that process personal data in or from the European Union. Yet its enforcement is still recent. Organizations under its effect are slow to adopt its principles. One particular difficulty is the low familiarity with the regulation among software architects and designers. The difficulty to interpret the content of the legal regulation at a technical level adds to that. This results in problems in understanding the impact and consequences that the regulation may have in detail for a particular system or project context.

In this paper we present some early work and emerging results related to supporting software architects in this situation. Specifically, we target those who need to understand how the GDPR might impact their design decisions. In the spirit of architectural tactics and patterns, we systematically identified and categorized 155 forces in the regulation. These results form the conceptual base for a first prototypical tool. It enables software architects to identify the relevant forces by guiding them through an online questionnaire. This leads them to relevant fragments of the GDPR and potentially relevant privacy patterns. We argue that this approach may help software professionals, in particular architects, familiarize with the GDPR and outline potential paths for evaluation.

Index Terms—software architecture; data privacy; decision support systems; design decisions

I. MOTIVATION

The introduction of new data protection laws, in particular the General Data Protection Regulation (GDPR) [1], provides more comprehensive protection than earlier, overridden legislation. This regulation is an important component of the EU's approach to protecting informational privacy, even for organizations outside the EU. This legislation, with its wide territorial scope (Art. 3), is a reason organizations around the world are giving privacy serious consideration. It affects any organization providing services in or from within the EU.

The strong binding force of the GDPR and the severe monetary penalties for not complying with it (up to 4% of the global turnover of a company breaking the law) make privacy and data protection important concerns also for software architects. Their design decisions for systems that use personal data may directly be affected by the GDPR. Decisions violating the regulation can lead to severe reputational and financial damage at the organizations developing or operating the system. In other words, the protection of privacy has become an important quality attribute of software systems, and design decisions

regarding privacy and data protection have, more than ever, become primary design decisions.

However, software architects, designers, and developers do not seem to be well prepared for this situation. Some of the factors contributing to this are:

- 1) The software architecture education, as reflected in textbooks, does not cover privacy preservation/data protection to the same degree as other typical quality attributes such as maintainability, performance, etc [2]–[4].
- 2) Even though the privacy community suggested 'privacy strategies' [5] that can be seen as architectural tactics [6] for privacy, they have not received much attention in neither software architecture literature nor practice.
- 3) Conceptual tools like privacy patterns [7] which could be used by practitioners as reusable solutions to reoccurring privacy problems are scarce and need to mature to be of significant help [8].
- 4) Practitioners are often not sufficiently familiar with the regulation and experience difficulties in interpreting the legal document as basis for technical decisions [9].

The research outlined in this paper contributes primarily to addressing the fourth aspect and is motivated by the following research question: How can we familiarize software architects (designers/developers) with the GDPR and help them make informed design decisions in line with the regulation?

The paper outlines emerging results from a study in which we extracted and categorized 'forces' in the GDPR that could potentially influence the architectural design of software systems. The term is used in the spirit of the patterns community in which a force denotes any aspect of a design problem that a pattern tries to balance and solve [10]. The formulation of such forces and their categorization form the basis for a first prototype of an online questionnaire. Software architects can use this prototype to identify relevant forces, parts of the GDPR relevant in their particular system or project context, and potential solutions in form of privacy design strategies and patterns.

The remaining paper is structured as follows. In the following section, some useful background and terminology is introduced. In section III, we explain the force extraction process and categorization, the tool prototype, and future evaluation plans. Section IV concludes the paper.

II. GDPR BACKGROUND AND TERMINOLOGY

The GDPR itself focuses on the ways in which organizations may process personal data. It aims for the ‘free [lawful] flow of information’ [1]. Article 4 (1) and (2) of the GDPR [1] provide definitions for ‘personal data’ and ‘processing’ respectively. Personal data in the GDPR refers to any information which relates to an identifiable natural person. Processing thereof is any usage of that personal data, from collection to erasure and anything in between.

An organization in terms of the GDPR is often the data controller (or just *controller*) [11]. The controller determines what is done with data, why, and how it is done. If an organization processes personal data on behalf a controller, they are the data processor (*processor*).

The user about whom the controller processes data is the data subject (which we still refer to as the *user*). All users are data subjects, regardless of identifiability. Yet, unidentifiable information is not protected by the GDPR (recital 26).

The regulation does not directly specify software design requirements. It rather prescribes certain data processing principles and obligations that have to be followed (Art. 5):

- Lawfulness, fairness and transparency (e.g. notice) of processing of private data;
- Compatible, specified, explicit, and legitimate purposes;
- Data accuracy and the limitation of its use and retention;
- Data integrity and confidentiality (security); and
- Accountability for compliance with the above.

Another obligation with direct potential impact on software architecting is data protection by design (Art. 25), also known as Privacy by Design (PbD) [1], [12]. It states that practitioners should consider the impacts on privacy in their decisions throughout the life of an information system. The secondary element of Article 25 is Data Protection by Default which can be seen as aspect of PbD [1]. A user who first starts using the product or service of the controller should initially receive the most privacy-friendly experience. This is limited by technical limitations and state of the art in terms of what is proportionate.

III. APPROACH

As outlined in Sec. I, the proposed approach is based on design forces extracted from the GDPR. These forces distill aspects of the regulation that might be architecturally relevant, formulated in simplified and condensed ways. This section describes the extraction process, the resulting categories of forces, the implemented prototype, and plans for future evaluations.

A. Extracting Design Forces from the GDPR

At first we sought to extract design forces from only a few of the GDPR recitals. We assigned first, second, and quality check reviewer roles for each recital among the authors.

We considered information in the recitals as being relevant as a force based on whether it applied to controllers and processors and whether it described an aspect potentially

Fig. 1. Categories of Design Forces

affecting the design of software systems. The recitals, or the relevant fragments of them, were then independently formulated as forces by two of the reviewers. The third reviewer then aggregated and finalized the formulations, or initiated discussions among the reviewers in case the extracted formulations diverged massively. We applied an iterative process starting with a few recitals in the beginning to establish and improve a common vocabulary and understanding of the task at hand.

Each resulting force was reviewed by at least two participating researchers, at least one being an expert on the legal aspects of privacy and one having a software engineering focus. This procedure ensured that the formulations as forces do not incorrectly distort, or overly simplify, the legal complexity of the regulation details and, on the other hand, that they are adequate for the software engineering domain.

In a second step we extended the extraction to the entire GDPR (not just the recitals). To do this we established which recitals related to each article, and then amended the resulting forces as necessary.

B. Categories of Extracted Design Forces

After having achieved consensus over the extracted forces among the authors, we explored how best to categorize the information. Figure 1 depicts the identified categories. The arrows indicate the order in which categories are processed when using the developed prototype.

The first of the categories of design forces is ‘Processing Grounds’. These forces result from the possible legal foundations defined in the GDPR that enable controllers to lawfully process personal data in the first place. These might be, for example legal (legally required to process) or contractual (fulfilling a valid contract with the user) grounds as well as grounds based on obtaining explicit and ‘Informed Consent’ [13]. Depending on the way that legal grounds are achieved, architects might want to think about specific design solutions—for example, how to achieve informed consent if other grounds do not apply.

Show **all/only** forces from these categories:

Processing Grounds Special Obligations General Obligations
User Rights Data Protection Principles Considerations

Does the controller, the one who determines the purposes and means for processing, have any special circumstances? There are additional concerns when dealing with multiple controllers or authorities, and some derogations for small to medium enterprises.

Select all that apply:

We are part of multiple controllers or undertakings
We have under 250 employees We fall in scope of multiple authorities

Previous Next Toggle this attribute Show All Filters

Fig. 2. Example of Filter Question

‘General Obligations’ gather forces that result from obligations that all controllers have to fulfil; in contrast, ‘Special Obligations’ contain forces that result from obligations specific to processing grounds, purposes or means, for example, when parts of the data processing is transferred to non-EU countries. As special rules apply in such scenarios, architects might want to make design decisions accordingly.

Forces in ‘User Rights’ refer mostly to functions that are needed to give users control over their data as granted by the GDPR. Examples are functions allowing users to execute their rights to be informed about processing of their personal data, or their right to restrict processing. Which functions need to be implemented and how to provide them in effective ways for a specific system are important architectural questions.

Regardless of which grounds justify the processing or which obligations apply, controllers must follow data protection principles; the corresponding category collects forces resulting from recitals dealing with these principles. For example, the principle of data minimization—personal data being adequate, relevant and limited to what is necessary in relation to the purposes for which it is processed—constitutes a force affecting the design of software.

Beyond the obligations controllers have to adhere to, there exist various further considerations. These present additional information which help to better understand the obligations and their enforcement.

C. Prototypical Tool Support

The tool prototype¹ offers both a wizard style questionnaire and a list of filtering options to review both based on and structured by the identified categories of forces. Both formats provide the same utility, though the wizard focuses on providing a single more detailed question at a time. We show an example of a wizard style filter question in Figure 2. In a separate figure we illustrate the summary shown upon clicking/tapping ‘Show All Filters’. Users of the tool can

return to where they left off in the wizard or select a specific question, as Figure 3 shows.

The questions presented by the wizard and their answers are the main function of the tool. By toggling an answer (all of which are in the form ‘select all which apply’) the user activates its related attributes. There can be more than one associated attribute, and these may overlap with other answers or answers of other questions. There is a hierarchy to these questions and answers similar to that of the forces. At the root are the categories, which receive a distinct style to help distinguish them from answers. This distinction is more important when viewing the summary of answers as they follow a similar structure.

Upon selecting any answer or category (whenever a force matches the attributes provided) the report is shown below the question/category. Figure 4 shows a filtered report following the running example. By default, it only displays the main heading with any relevant forces and minimized headings under it. Users can toggle these minimized headings individually or all at once by clicking/tapping ‘Show All Relevant Forces’. This button toggles to also hide all forces.

The report can present forces filtered in one of two ways. In the first of these, we show both forces which apply from the answers selected and also those from any selected categories. We refer to this as ‘inclusive’ or ‘all’ filter. The second filter, the ‘exclusive’ or ‘only’ filter, is used in the examples depicted in Figures 2–4. In this approach, we only show forces matching at least one selected category and at least one matching answer. This filtering approach allows users to focus on a subset of the forces. Users can toggle between these by clicking/tapping ‘all/only’ in the main category heading.

As depicted in Fig. 4, the description of forces also contain links back to the relevant GDPR recitals. Moreover, the user can choose to inspect privacy design strategies and privacy patterns that might be adequate to address the design force. This way, the user can easily trace back to the rationale of a force (the recitals) being selected as relevant for their context and continue to explore potential technical solutions.

D. Future Evaluation

The approach has not yet been evaluated by practitioners. The hypotheses to be tested in evaluations are, in line with the motivating research question, that the approach/tool improves familiarization with the GDPR and helps to make adequate design decisions. Software engineers could provide feedback on how easy the content is to understand and utilize. Legal practitioners may then evaluate how the tool influences design decisions by these engineers. They could also further evaluate legal completeness beyond our own assessment. Similarly, engineers could point out any incomplete information or missing functionality.

Given that these different aspects should be evaluated, a mixed methods approach to evaluation seems appropriate [14]. We intend to conduct case studies with industrial software engineers. In these, we would test the prototype to identify any shortcomings in the tool and to better understand its

¹ Accessible at <https://privacypatterns.cs.ru.nl/tool>

Fig. 3. Example of Filter Summary

+ Show All Relevant Forces (12)

- Forces for processing which can be used to personally identify any users
- + Other considerations (x)
- Principles of data protection (x)
 - + Consequences of not upholding data the protection principles (x)

Considerations: Accountability Derogation, SMEs

Controllers of medium or smaller enterprises below 250 employees may be exempt from some obligations such as with logging, but must meet the same level of protection.

[show privacy patterns](#) [Subject-matter and objectives; Recital 13;](#)

General Obligations: Cooperation

The protection of personal data should not hinder proportional and necessary measures taken by law enforcement or national security (governed by the Police Directive 2016/680/EC). This processing should be by a competent authority, and may receive additional regulation.

[show privacy patterns](#) [Material scope; Recital 19;](#)

Fig. 4. Example of Filtered Report

use in practice. After that, we intend to carry out a survey among practitioners to further assess the tool in order to achieve greater external validity. This will indicate whether a tool like the proposed can improve the familiarity with the GDPR among practitioners and help making design decisions in line with the GDPR and privacy concerns. Complementary experiments in which practitioners make design decisions with and without the tool will additionally be performed to study the effectivity and efficiency of the approach with greater internal validity.

IV. DISCUSSION AND CONCLUSION

Supporting software architects in making design decisions related to privacy protection has become increasingly important with the GDPR having entered into force. It is hence crucial to familiarize software architects with the regulation and to equip them with tools that help them interpret its legal formulations for technical decision making. We assume that an interactive tool like the suggested prototype could support such decision making processes as it links design forces relevant for a system or project of interest with their legal rationale in the GDPR and potential building blocks of a solution (privacy design strategies and patterns). This way software professionals, not just architects, can freely explore the associations between the legal regulations and implications in their technical solutions. We believe that this would not only support the process of making privacy design decisions, addressing the fourth factor as listed in Sec. I. It could also be used as an educational tool for those professionals that need

to extend their knowledge in the area of privacy strategies and privacy by design which tackles the first and second factor. Such tool support could even help raise awareness for the state of the art in privacy patterns and increase the maturing process (third factor). We aim at providing evidence for or against these hypotheses in our future evaluations.

ACKNOWLEDGMENTS

The authors would like to thank the Netherlands Organization for Scientific Research (NWO) under the ‘Patterns for Privacy’ project (CYBSEC.14.030) for their sponsorship.

REFERENCES

- [1] European Parliament and Council of the European Union, “General Data Protection Regulation,” *Official Journal of the European Union*, vol. 119, 2015.
- [2] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*, 3rd ed. Addison-Wesley Professional, 2012.
- [3] N. Rozanski and E. Woods, *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives*. Addison-Wesley Professional, 2005.
- [4] R. N. Taylor, N. Medvidovic, and E. M. Dashofy, *Software Architecture: Foundations, Theory, and Practice*. Wiley, 2009.
- [5] J.-H. Hoepman, “Privacy Design Strategies,” in *Proceedings of the IFIP International Conference on Information Security and Privacy Protection - IFIP SEC*, 2014, pp. 446–459.
- [6] L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*, 3rd ed. Addison-Wesley Professional, 2012.
- [7] G. Danezis, J. Domingo-Ferrer, M. Hansen, J.-H. Hoepman, D. Le Métayer, R. Tirta, and S. Schiffner, “Privacy and Data Protection by Design – from policy to engineering,” Tech. Rep., 2014. [Online]. Available: <https://publications.europa.eu/en/publication-detail/-/publication/6548a14b-9863-410d-a8a6-c15a0137d281/language-en>
- [8] J. Lenhard, L. Fritsch, and S. Herold, “A literature study on privacy patterns research,” in *2017 43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, 2017, pp. 194–201.
- [9] I. Hadar, T. Hasson, O. Ayalon, E. Toch, M. Birnhack, S. Sherman, and A. Balissa, “Privacy by designers: software developers’ privacy mindset,” *Emp. Softw. Eng.*, vol. 23, no. 1, pp. 259–289, 2018.
- [10] F. Buschmann, R. Maunier, H. Rohnert, P. Sommerlad, and M. Stal, *Pattern-Oriented Software Architecture, A System of Patterns*. Wiley, 1996.
- [11] Article 29 Data Protection Working Party, “Opinion 1/2010 on the Concepts of “Controller” and “Processor,”” Tech. Rep. October, 2010. [Online]. Available: http://ec.europa.eu/justice/article-29/documentation/opinion-recommendation/files/2010/wp169_en.pdf
- [12] A. Cavoukian, “Privacy by Design,” Ontario, Canada, Tech. Rep., 2009. [Online]. Available: <https://www.ipc.on.ca/wp-content/uploads/2013/09/pbd-primer.pdf>
- [13] C. Bier and E. Krempel, “Common Privacy Patterns in Video Surveillance and Smart Energy,” in *Proc. 7th International Conference on Computing and Convergence Technology*. IEEE, 2012, pp. 610–615.
- [14] C. Teddlie and A. Tashakkori, *Foundations of Mixed Methods Research: Integrating Quantitative and Qualitative Approaches in the Social and Behavioral Sciences*. SAGE Publications, 2009.