Towards Privacy-Preserving IoT Systems Using Model Driven Engineering (Extended Abstract)

Judith Michael[1], Lukas Netz[1], Bernhard Rumpe[1], Simon Varga[1]

Abstract: Collecting, storing and processing data from machines, processes and workers in production processes is increasing with technological possibilities and the availability of sensors. This raises the challenge of ensuring privacy for personal data within this context. For MDE approaches it is important to consider privacy already at the model level. This paper discusses a way to create privacy-preserving IoT systems using an MDE approach to support privacy and data transparency. We show the relevance and application on a use case from industrial production processes. Additionally, we discuss abilities for practical realization and its limitations. The work summarized in this extended abstract has been presented at the MDE4IoT workshop (MODELS 2019) and is published by CEUR-WS.org [Mi19b].

Keywords: Domain-Specific Languages, Generated Enterprise Information Systems, Information Portals, Internet of Things, Model-Based Software Engineering, Privacy-By-Design, Privacy Modeling

1 Main Findings

Problem Description. The rise of wearable technologies makes it possible to equip workers, products and machines in production processes with miniaturized sensors. This development goes hand in hand with questions about the informational self-determination, the security of data collected as well as data protection and transparency. This paper shows how it is possible to include privacy considerations in the MDE development process already on model level. The foundation for this work lies in the privacy-design strategies [CHH16] as well as an idea to include privacy checkpoints into system architectures. In preceding publications [Ma19; Mi19a], we have already taken these ideas into account. This paper goes a step further and discusses them in relation with model- driven engineering (MDE).

Use Case. The use case shows a part of a production process, so one station in a manufacturing area with several operators and robots collaborating with each other. Operators are wearing smart glasses and watches and are using smartphones and tablets which collect sensory information. Further sensory information is collected on products and related machines and robots. Assistive systems, which use such data ,e.g., to do ergonomic analyses to improve the ergonomic intervention processes have to follow security and privacy regulations and should provide informational self-determination facilities to inform operators what happens with

[1] Software Engineering, RWTH Aachen, Germany, www.se-rwth.de, {michael,netz,rumpe,varga}@se-rwth.de

the data. Thus, besides data collection, storage, removal as well as primary and secondary use, we have included an information portal into our approach which ensures privacy control and provides information. Operators can give their consent for data usage, withdraw it and get options to delete the data at any point. Privacy checkpoints within the components of the system architecture can help to ensure a privacy-preserving system design.

Results. The approach presented in this paper uses MDE tools and frameworks together with a set of domain-specific languages (DSLs) to create an Enterprise Information System (EIS). The EIS is considering privacy-preservation and makes the relevant information available for users and data providers to allow for informed decisions about their data use. Starting with structural information in the domain model, which is needed for MDE approaches, we add privacy concepts (privacy model) to support the execution of the privacy checkpoints. At the instance level, a purpose tree has to be defined which is used in concrete rules attached to a privacy policy instance. The purpose tree defines hierarchical relations between purposes as well as connected attributes and classes, which are needed for this purpose.

Employees define their privacy policy rules and can allow the data controller to give access to their data based on these rules. They can see decisions for data access based on the automatic comparison of privacy policies between data providers and data consumers. The data controller can add and change concrete purposes for data collection and storage. Employees are informed about changes and get possibilities to define new privacy policy rules or change existing ones. The traceability of data is ensured in the system architecture by considering all checkpoints. In our approach we demonstrate how to enable engineers to define software with model-driven approaches which meet the growing requirements for data protection and transparency.

References

[CHH16] Colesky, M.; Hoepman, J.; Hillen, C.: A Critical Analysis of Privacy Design Strategies. In: IEEE Security and Privacy Workshops (SPW). Pp. 33–40, 2016.

[Ma19] Mannhardt, F.; Koschmider, A.; Baracaldo, N.; Weidlich, M.; Michael, J.: Privacy-Preserving Process Mining: Differential Privacy for Event Logs. Business & Information Systems Engineering (BISE) 61/5, pp. 595–614, 2019, ISSN: 1867-0202.

[Mi19a] Michael, J.; Koschmider, A.; Mannhardt, F.; Baracaldo, N.; Rumpe, B.: User-Centered and Privacy-Driven Process Mining System Design for IoT. In: Information Systems Engineering in Responsible Information Systems. Vol. 350, LNBIP, Springer, pp. 194–206, 2019, ISBN: 978-3-030-21296-4.

[Mi19b] Michael, J.; Netz, L.; Rumpe, B.; Varga, S.: Towards Privacy-Preserving IoT Systems Using Model Driven Engineering. In: MDE4IoT & ModComp Workshops 2019. Vol. 2442, CEUR-WS.org, pp. 15–22, Sept. 2019.