2022 IEEE/ACM 44th International Conference on Software Engineering: Companion Proceedings (ICSE-Companion) | 978-1-6654-9598-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICSE-Companion55297.2022.9793756

A Framework to Support Software Developers in Implementing Privacy Features

Anthony Mazeli
anthony.mazeli@bristol.ac.uk
University of Bristol
Bristol, United Kingdom

ABSTRACT

Software developers are inundated with responsibility to incorporate privacy artifacts into software design from the onset in line with best practices. However, little is understood about the struggles developers face implementing privacy into software design. This PhD will undertake: (1) a Systematic Literature Review (SLR) to understand developers interpretation or lack thereof of privacy regulations while incorporating privacy into software systems; (2) two task-based studies to analyze software developers' privacy compliance to ascertain whether or not they are able to comply with regulatory standards in implementing privacy into software design; (3) analyze mental models adopted by developers when trying to ameliorate their struggles, and (4) then design and evaluate a framework that helps developers make informed privacy decisions.

CCS CONCEPTS

• **Security and privacy** → **Usability in security and privacy**; **Privacy protections**; • **Software and its engineering** → **Software design techniques**; **Software design engineering**.

KEYWORDS

Developer Centered Privacy, Software Systems, Regulatory Standard, GDPR, CCPA, Regulatory Compliance, SLR, Mental Models.

ACM Reference Format:
Anthony Mazeli. 2022. A Framework to Support Software Developers in Implementing Privacy Features. In *44th International Conference on Software Engineering Companion (ICSE '22 Companion), May 21–29, 2022, Pittsburgh, PA, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3510454.3517054

1 INTRODUCTION

Software developers need to implement a range of privacy features in their applications to support users in protecting their personal information and to comply with a range of data protection regulations (e.g., GDPR [16], California Consumer Privacy Act (CCPA) [15]). Best practices prescribe Privacy by design (PbD) which incorporates privacy protection considerations into product applications during the initial design phase, rather than retroactively [5]. The aim is to allow a "bottom-up approach," where privacy is embedded responsibly from the outset during design processes using approved standards (e.g., Privacy Engineering ISO standards: ISO/IEC PCTR 27550, NIST: SP 800-53 Rev. 5) to implement privacy requirements [3, 8, 17]. However, little is understood about the struggles developers face in implementing privacy into software design.

Currently, whether these standards meet the intended compliance requirements for privacy remains a matter of the software developers' capacity to interpret privacy regulation which may be influenced by many factors—for instance, security vocabulary and compliance requirements [11, 21]. Moreover, Tahaei et al. [19] found that the claim of privacy consideration in initial design processes is inconsistently applied by developers. There is also an issue around tools, whether the perceived standard tools (e.g.,PETs, Crypto libraries) available sufficiently ensure accurate implementation in line with explicit compliance regulations. This remains debatable in the absence of standardized evaluation metrics or means of calibration to justify this assertion. Presently, there is no framework that succinctly communicates the balance between privacy implementation and risks associated with actions and outcomes in software development. This research seeks to extend the body of work around developers by analysing the struggles software developers face with implementing privacy features into software systems by developing a framework that supports developers when implementing privacy into software design.

2 PROBLEM STATEMENT

Implementing privacy features into software systems is becoming increasingly difficult for developers due to the absence of effective mechanisms for incorporating privacy into software design. There is an abundance of regulatory controls, but lack of clear communication conflating with developers' decision-making resulting in judgement errors [1, 6, 20].

A plethora of studies in privacy focus mainly on users' perspectives, but not on developers. However, software developers are the ones who directly impact privacy implementation in software design, which determines the amount of user data that is accessible or how it is used/processed. Some studies suggest developers' challenges often arise from mismatches between developers' attitudes and actual behaviour [7, 22]. In other cases, they are borne out of inadequate step-by-step guidance to accomplish the task [9, 19].Hence, the need for standard framework that sufficiently supports developers in addressing the challenges they face.

3 RELATED WORK

Recent studies identify the need for more guidance to ameliorate developers' struggles within the PbD ecosystem by suggesting

ICSE '22 Companion, May 21–29, 2022, Pittsburgh, PA, USA

ACM ISBN 978-1-6654-9598-1/22/05…$15.00
https://doi.org/10.1145/3510454.3517054

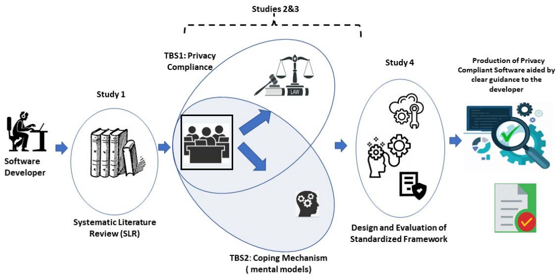
Studies 2&3
TBS1: Privacy Compliance
Study 1
Study 4
Production of Privacy Compliant Software aided by clear guidance to the developer
LAW
Software Developer
Systematic Literature Review (SLR)
Design and Evaluation of Standardized Framework
TBS2: Coping Mechanism (mental models)

Figure 1: Research methodology

consultation with Innovation and Security champions in software teams [19]. While this approach has worked in identifying the efficacy and pitfalls in security strategies, it is silent on the impact of regulatory control, cognitive biases, and corporate goals which also influence developers' best practices within privacy teams [2, 6, 9, 18]. Likewise, traditional interpretation difficulties of communicating and incorporating privacy considerations into software by developers persists [4, 10, 12–14]. Admittedly, efforts by privacy champions supported by management and a critical mass of software developers can enhance a good organizational privacy culture. However, this is more adept in facilitating security practices rather than guiding the implementation of privacy principles into software design [19]. This study complements the limited body of knowledge in the design of a framework that supports developers' decision-making process during privacy implementation in software.

4 RESEARCH HYPOTHESIS

This work hypothesizes that developers struggle because there is not enough support to help them understand the privacy requirements needed to comply with regulations. Moreover, if developers are adequately supported through innovative mechanisms or systematic frameworks, they would be able to implement privacy features into software design that will effectively meet regulatory standards and respect users' privacy.

5 EXPECTED CONTRIBUTIONS

This research will amongst other things: (1) empirically establish an understanding of the developers' struggles around implementing privacy into systems; (2) develop a comprehensive developer-centered privacy framework (i.e., guidelines, tools, methods); and (3) evaluate the derived framework in terms of its usability and how effectively it supports developers in privacy tasks.

6 METHODOLOGY

This research will employ mixed-methods; SLR, two task based studies and a design and evaluation of a systematic framework. Figure 1 shows a pictorial representation of the methodology.

Initial SLR: An initial SLR analysis of relevant papers using derived query strings from selected key words and a set of inclusion/exclusion criteria to review current literature, identify and aggregate gaps in this area will be conducted. This is designed to aid a clear understanding of current research on developers' interpretation or lack thereof of privacy implementation requirements in software design. The SLR would also seek to examine the impact/influence of pragmatic third-party requirements in software design regarding privacy and how software developers account for different regulatory standards (i.e., GDPR).

First Task-Based Study: Based on the initial SLR, a task-based study would be conducted to investigate software developers' privacy compliance or the lack thereof to understand whether or not software developers fully comply with the challenges they face when incorporating privacy into software design using available compliance regulatory standards(i.e., GDPR, CCPA). This would be conducted by closely observing and documenting the actions undertaken by software developers in incorporating privacy features into software design through a series of predefined and relevant tasks. The developers would be expected to use the think-aloud technique to enable documentation of their thought processes.

Second Task-Based Study: This will focus more on the coping mechanisms adopted by software developers. This aims to analyze developers' behaviour, practices, and cognitive approaches (i.e., mental models) they adopt to overcome the struggles they face while incorporating privacy into software artifacts. Like the previous study, developers would be subjected to simple software design tasks to observe and identify their points of difficulty and the mental models they adopt to mitigate these challenges. Again, developers will adopt the think-aloud technique.

Design and Evaluation of a Systematic Framework: An aggregate of empirical insights derived from key knowledge areas of the SLR and outputs from task-based studies would be incorporated into a framework (i.e., guidance, tools) that supports developers in implementing privacy artifacts into software design at scale. Based on this, proposed follow on field study(ies), interviews and surveys will evaluate, document, and review usability issues and limitations of the designed framework.

7 CONCLUSION AND EVALUATION

Software is at the heart of the digital economy. Though users have agency over their data in terms of how it is collected and used. Developers have a key role when it comes to features that support and empower users with respect to privacy (or otherwise). But developers are not privacy experts and need appropriate frameworks to help them with privacy implementation tasks. This research makes a contribution towards advancing state-of-the-art in this area and adopts appropriate mixed-methods in understanding the struggles software developers face in implementing privacy into software design using two studies to evaluate its claims. Further field study,interviews and surveys are proposed to complement the controlled experiments. The expected output is a systematic framework based on empirical insights that helps developers' implement privacy features in software systems. A review study is proposed to validate the derived framework, analyze its usability, efficiency and limitations in order to identify areas for future research.

ACKNOWLEDGMENTS

This work is supported by Engineering and Physical Sciences Research Council (EPSRC) - EP/S022465/1

REFERENCES

[1] Abdulrahman Alhazmi and Nalin AG Arachchilage. 2021. A Serious Game Design Framework for Software Developers to Put GDPR into Practice. In *The 16th International Conference on Availability, Reliability and Security*. 1–6.

[2] Abdulrahman Alhazmi and Nalin Asanka Gamagedara Arachchilage. 2021. I'm all ears! Listening to software developers on putting GDPR principles into software development practice. *Personal and Ubiquitous Computing* (2021), 1–14.

[3] Ingolf Becker, Simon Parkin, and M Angela Sasse. 2017. Finding security champions in blends of organisational culture. *Proc. USEC* 11 (2017).

[4] Andrew Begel and Beth Simon. 2008. Novice software developers, all over again. In *Proceedings of the fourth international workshop on computing education research*. 3–14.

[5] Ann Cavoukian et al. 2009. Privacy by design: The 7 foundational principles. *Information and privacy commissioner of Ontario, Canada* 5 (2009), 12.

[6] Souti Chattopadhyay, Nicholas Nelson, Audrey Au, Natalia Morales, Christopher Sanchez, Rahul Pandita, and Anita Sarma. 2020. A tale from the trenches: cognitive biases and software development. In *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. 654–665.

[7] Partha Das Chowdhury, Joseph Hallett, Nikhil Patnaik, Mohammad Tahaei, and Awais Rashid. 2021. Developers Are Neither Enemies Nor Users: They Are Collaborators. In *2021 IEEE Cybersecurity Development (SecDev)*. 22–26.

[8] Duy Dang-Pham, Siddhi Pittayachawan, and Vince Bruno. 2017. Applications of social network analysis in behavioural information security research: Concepts and empirical analysis. *Computers & Security* 68 (2017), 1–15.

[9] Yuanyuan Feng, Yaxing Yao, and Norman Sadeh. 2021. A Design Space for Privacy Choices: Towards Meaningful Privacy Control in the Internet of Things. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–16.

[10] Colin M. Gray, Yubo Kou, Bryan Battles, Joseph Hoggatt, and Austin L. Toombs. 2014. Using psycho-physiological measures to assess task difficulty in software development. In *Proceedings of the 36th ICSE. ACM*. 402–413.

[11] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. 2018. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering* 23, 1 (2018), 259–289.

[12] Andrew J Ko, Robert DeLine, and Gina Venolia. 2007. Information needs in collocated software development teams. In *29th International Conference on Software Engineering (ICSE'07)*. IEEE, 344–353.

[13] André N Meyer, Laura E Barton, Gail C Murphy, Thomas Zimmermann, and Thomas Fritz. 2017. The work life of developers: Activities, switches and perceived productivity. *IEEE Transactions on Software Engineering* 43, 12 (2017), 1178–1193.

[14] André N Meyer, Thomas Fritz, Gail C Murphy, and Thomas Zimmermann. 2014. Software developers' perceptions of productivity. In *Proceedings of the 22nd ACM SIGSOFT International Symposium on Foundations of Software Engineering*. 19–29.

[15] State of California Department of Justice. 2018. California Consumer Privacy Act (CCPA). https://oag.ca.gov/privacy/ccpa Last accessed November 2021.

[16] The European parliament and the council of the European union. 2018. General Data Protection Regulation (GDPR). https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679 Last accessed November 2021.

[17] Hiep Cong Pham, Linda Brennan, Lukas Parker, Nhat Tram Phan-Le, Irfan Ulhaq, Mathews Zanda Nkhoma, and Minh Nhat Nguyen. 2019. Enhancing cyber security behavior: an internal social marketing approach. *Information & Computer Security* (2019).

[18] Iskander Sanchez-Rola, Matteo Dell'Amico, Platon Kotzias, Davide Balzarotti, Leyla Bilge, Pierre-Antoine Vervier, and Igor Santos. 2019. Can i opt out yet? gdpr and the global illusion of cookie control. In *Proceedings of the 2019 ACM Asia conference on computer and communications security*. 340–351.

[19] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 1–15.

[20] Mohammad Tahaei and Kami Vaniea. 2019. A Survey on Developer-Centred Security. In *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. IEEE, 129–138.

[21] Ari Ezra Waldman. 2018. Designing without privacy. *Houston Law Review* 55, 659 (2018).

[22] Xueling Zhang, Xiaoyin Wang, Rocky Slavin, Travis Breaux, and Jianwei Niu. 2020. How does misconfiguration of analytic services compromise mobile privacy?. In *2020 IEEE/ACM 42nd International Conference on Software Engineering (ICSE)*. IEEE, 1572–1583.

Conceitos: Privacy by Design (PbD), Framework, Regulartory Standards (GDPR, CCPA, LGPD)