

The Developer Factor in Software Privacy

Mohammad Tahaei

Doctor of Philosophy
School of Informatics
University of Edinburgh
2021

Abstract

Computer programming operates and controls our personal devices, cars, and infrastructures. These programs are written by software developers who use tools, software development platforms, and online resources to build systems used by billions of people. As we move towards societies that rely on computer programs, the need for private and secure systems increases. Developers, the workforce behind the data economy, impact these systems' privacy, and consequently, the users and society. Therefore, understanding the *developer factor* in software privacy provides invaluable inputs to software companies, regulators, and tool builders.

This thesis includes six research papers that look at the developer factor in software privacy. We find that developers impact software privacy and are also influenced by external entities such as tools, platforms, academia, and regulators. For example, changes in regulations create challenges and hurdles for developers, such as creating privacy policies, managing permissions, and keeping user data private and secure. Developers interactions with tools and software development platforms, shape their understanding of what privacy means, such as consent and access control. Presentation of privacy information and options on platforms also heavily impact developers' decisions for their users' privacy, and platforms may sometimes nudge developers into sharing more of their users' data by using design (dark) patterns.

Other places developers learn about privacy include universities, though they may not learn how to include privacy in software. Some organisations are making efforts to champion privacy as a concept inside development teams, and we find that this direction shows promise as it gives developers direct access to a champion who cares about privacy. However, we also find that their organisation or the wider community may not always support these privacy champions. Privacy champions face an uphill battle to counter many of the same privacy misconceptions seen in the general population, such as the 'I've got nothing to hide' attitude.

Overall, I find that research in *developer-centred privacy* is improving and that many of the approaches tried show promise. However, future work is still needed to understand how to best present privacy concepts to developers in ways that support their existing workflows.

To my parents

Acknowledgements

I would like to express by gratitude to my supervisor Dr Kami Vaniea for giving me the freedom to explore my ideas while mentoring me; my collaborators Dr Alisa Frik, Dr Naomi Saphra, and Dr Maria Wolters, for their contributions; my host supervisor at the University of British Columbia, Professor Konstantin Beznosov, for giving an opportunity to experience a different working environment and lab, and his contributions to my research; the committee members of my annual reviews Dr Markulf Kohlweiss and Dr Chris Lucas for their feedback; and the thesis committee Professor Sascha Fahl and Professor John Vines, for reading my thesis and their feedback.

I would like to also thank my labmates and friends, Sara Albakry, Kholoud Althobaiti, Lucy Havens, Adam Jenkins, Dilara Kekulluoglu, Florian Mathis, and Nicole Meng, for their feedback, support, and fruitful discussions; Microsoft Research and the School of Informatics at the University of Edinburgh for funding my PhD; and my parents and sisters whose support and encouragement have always driven me forward.

Thank you all.

Declaration of Authorship and Publications

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

All chapters except for Chapter 1: Introduction and Chapter 8: Final Thoughts, are published in peer-reviewed venues. I declare that I substantially contributed to all papers and was involved in all phases of the research process, including study conceptualisation, data collection, data analysis and interpretation, and writing of the papers. All included contributions here are confirmed by the co-authors.

- ▶ Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Deciding on Personalized Ads: Nudging Developers About User Privacy. Symposium on Usable Privacy and Security (SOUPS) 2021.
Contributions: MT proposed the initial idea, designed the study, conducted the quantitative analysis, and wrote the first draft of the paper. AF assisted in study design and qualitative analysis. KV provided feedback throughout the project. All authors contributed in the writing.
- ▶ Mohammad Tahaei and Kami Vaniea. 2021. “Developers Are Responsible”: What Ad Networks Tell Developers About Privacy. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems Extended Abstracts. ACM, New York, NY, USA, 1–12. DOI: [10.1145/3411763.3451805](https://doi.org/10.1145/3411763.3451805)
Contributions: MT proposed the initial idea, conducted the experiment, and wrote the first draft of the paper. KV provided feedback throughout the project. Both authors contributed in the writing. A position paper was published in ‘What Can CHI Do About Dark Patterns?’ CHI ’21 workshop [328] based on this paper.
- ▶ Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, 1–15. DOI: [10.1145/3411764.3445768](https://doi.org/10.1145/3411764.3445768)
Contributions: MT proposed the initial idea, designed the study, conducted the interviews, and wrote the first draft of the paper. AF was the second coder for the qualitative analysis, and assisted in study design and recruitment. KV provided feedback throughout the project. All authors contributed in the writing.

- ▶ Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 2020. Understanding Privacy-Related Questions on Stack Overflow. Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, 1–14. DOI: [10.1145/3313831.3376768](https://doi.org/10.1145/3313831.3376768)

Contributions: MT proposed the initial idea, collected the data, and wrote the first draft of the paper. KV was the second coder for the qualitative analysis and provided feedback throughout the project, and NS did the programming for topic modelling. All authors contributed in the writing.

- ▶ Mohammad Tahaei, Adam Jenkins, Kami Vaniea, and Maria K. Wolters. “I Don’t Know Too Much About It”: On the Security Mindsets of Computer Science Students. 9th International Workshop, STAST 2019, Luxembourg City, Luxembourg, September 26, 2019, Revised Selected Papers. In: Socio-Technical Aspects in Security and Trust. Ed. by Thomas Groß and Tryfonas Theo. First Edition. Springer International Publishing, June 2021. DOI: [10.1007/978-3-030-55958-8](https://doi.org/10.1007/978-3-030-55958-8)

Contributions: MT proposed the initial idea, designed and conducted the interviews, and wrote the first draft of the paper. AJ was the second coder for the qualitative analysis. KV closely supervised the project, and MW provided additional feedback throughout the project. All authors contributed in the writing. An early version of this paper was published as a Doctoral Consortium poster in ITiCSE ’19 [322].

- ▶ Mohammad Tahaei and Kami Vaniea. A Survey on Developer-Centred Security. 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 2019, 129–138, DOI: [10.1109/EuroSPW.2019.00021](https://doi.org/10.1109/EuroSPW.2019.00021)

Contributions: MT proposed the initial idea, collected and read the papers, and wrote the first draft of the paper. KV was the second coder for the qualitative analysis and provided feedback throughout the project. Both authors contributed in the writing.

Mohammad Tahaei

Contents

PROLOGUE	1
1. Introduction	3
1.1. A Survey On Developer-Centred Privacy and Security	5
1.2. Understanding Privacy-Related Questions on Stack Overflow . . .	6
1.3. Privacy Champions in Software Teams	6
1.4. On the Privacy and Security Mindsets of Computer Science Students	7
1.5. What Ad Networks Tell Developers About Privacy	8
1.6. Nudging Developers About User Privacy	9
1.7. Final Thoughts	9
1.8. Contributions	10
RESEARCH PAPERS	11
2. A Survey On Developer-Centred Privacy and Security	13
2.1. Introduction	14
2.2. Systematisation Approach	15
2.3. Methodology Results	19
2.4. Research Theme Results	23
2.5. Discussion	31
2.6. Conclusion	34
3. Understanding Privacy-Related Questions on Stack Overflow	37
3.1. Introduction	38
3.2. Related Work	39
3.3. Method	42
3.4. LDA Findings	47
3.5. Coding Findings	48
3.6. Privacy Aspect Thematic Analysis Findings	51
3.7. Discussion	58
3.8. Limitations	62
3.9. Future Work	62
3.10. Conclusion	63
4. Privacy Champions in Software Teams	67
4.1. Introduction	68
4.2. Related Work	69

4.3. Method	71
4.4. Results	74
4.5. Discussion	90
4.6. Conclusion and Future Work	95
5. On the Privacy and Security Mindsets of Computer Science Students	99
5.1. Introduction	100
5.2. Related Work	101
5.3. Methodology	103
5.4. Results	107
5.5. Discussion	114
5.6. Limitations	116
5.7. Future Work	116
5.8. Conclusion	117
6. What Ad Networks Tell Developers About Privacy	121
6.1. Introduction	122
6.2. Background	122
6.3. Method	123
6.4. Findings	125
6.5. Discussion and Future Work	130
7. Nudging Developers About User Privacy	137
7.1. Introduction	138
7.2. Related Work	140
7.3. Method	141
7.4. Results	146
7.5. Discussion and Future Work	157
7.6. Conclusion	161
EPILOGUE	165
8. Final Thoughts	167
8.1. Discussion	167
8.2. Future Directions	169
8.3. Conclusion	171
REFERENCES	173

APPENDIX	219
A. Appendices for Privacy Champions Study	220
A.1. Screening Survey	220
A.2. Interview Script	221
A.3. Codebook	222
B. Appendices for Computer Science Students Study	223
B.1. Interview Script	223
C. Appendices for What Ad Networks Tell Developers About Privacy Study	225
C.1. Screenshots & Summary of Presented Privacy-Related Information	225
D. Appendices for Nudging Developers About User Privacy Study	235
D.1. Survey Instruments	235
D.2. Ads Options on Google AdMob Developer Dashboard	240
D.3. Participants' Demographics and Opinions About Ad Networks . .	241

List of Figures

3.1.	A sample privacy-related question with an accepted answer.	43
3.2.	Count of questions mentioning privacy per year (SO-privacy).	43
3.3.	Top 50 most commonly used tags by users (SO-privacy).	45
7.1.	Participants' choices between personalised and non-personalised ads across the six conditions.	147
C.1.	GAM's warning about obtaining consent from users in the European Economic Area in the <i>Get Started</i> page.	225
C.2.	<i>Obtaining Consent with the User Messaging Platform</i> page in GAM provided a sample code for obtaining consent from users that constantly shows the popup to the user until they consent. Developers who use this sample code spread a 'nagging' dark pattern in their apps.	225
C.3.	<i>Precise Location Data Policy</i> page in GAM provided a sample code for obtaining location consent from users without providing a 'I do not consent' or 'No' button. Developers who use this sample code spread a 'forced action' dark pattern in their applications.	226
C.4.	Google Analytics had the default option on to share our data with Google (GAM). Turning off the default option would result in seeing a list of sub by default on permissions for all the grey items (e.g. Google products, Benchmarking). 'Preselection' dark patterns happens here as the default setting to share information between multiple services is not in the best interest of user privacy.	226
C.5.	Google Analytics is turned on by default when creating an account on Firebase (GAM). 'Preselection' dark patterns happens here as the default setting to share information between multiple services is not in the best interest of user privacy.	227
C.6.	When creating an app on GAM, we were asked to enable users metrics for powerful reports. The box was pre-ticked. 'Preselection' happens here as sharing user data with multiple services is not in favour of user privacy.	227
C.7.	In the GAM account page under <i>Blocking controls</i> we could change the ad content rating. The default value was on the MA. 'Toying with emotion' has been applied to encourage developers stay with the MA ratings.	227

C.8. In the GAM account page under <i>Blocking controls</i> we could ‘allow’ or ‘block’ certain categories. The use of grey and blue colour to use ‘aesthetic manipulation’ dark pattern is easily visible. Grey commonly has a passive and negative tone whereas blue is known to have a positive tone [371].	228
C.9. CCPA and GDPR sections of GAM have pre-selected items for information processing and personalised ads. ‘Preselection’ dark pattern occurs here because GAM by default collect information and also shows personalised ads (hence collects more information as well).	229
C.10. In the <i>funding choices</i> service we could ‘allow’ and ‘block’ certain ad vendors. All vendors were ‘Allowed’ by default. GAM pre-ticked the box for automatically adding new vendors to list. ‘Preselection’ dark pattern spreads via this interface to end-users if developers do not make an effort to change the defaults.	229
C.11. <i>Funding choices</i> is a service from GAM to create consent popups. It provides two ready-to-use consent popups to developers, the first option does not have a ‘Do not consent’ button.	230
C.12. <i>Funding choices</i> is a service from GAM to create consent popups. Several items could be customised for users. These are some of the default values. ‘Preselection’ dark pattern spreads via this interface to end-users if developers do not make an effort to change the defaults.	230
C.13. AMN in the <i>Quick Start Guide</i> page asks developers to add internet, network, wifi access, coarse and fine location for higher revenues for developers. While location permissions are called as ‘optional’ the sample code includes both fine and coarse location permissions. ‘Sneak into basket’ dark pattern is present here because developers may copy paste this code without fully being informed about what the sample code does.	231
C.14. IAB’s sample values for CCPA’s <i>US Privacy String</i>	231
C.15. AMN’s sample code in their <i>FAQ</i> page. <code>enableGeoLocation</code> is switched on in the sample code (‘sneaking’ [134]). ‘1---’ is provided as an example privacy string value. ‘1’ means version 1 and ‘-’ means ‘Not Applicable’. Two dark patterns are visible here, if developers copy paste this code ‘sneak into basket’ occurs and ‘preselection’ also happens because the defaults are not in favour of user privacy (see Figure C.14 for IAB code samples.)	232
C.16. AMN lets developers to block ad categories. By default no categories are blocked. ‘Preselection’ dark pattern happens here as developers may never visit this page and the defaults are not in the best interest of end-users.	232

C.17. FAN's sensitive categories are all active by default. The two blocked options are blocked by us. 'Preselection' dark pattern happens here as developers may never visit this page and the defaults are not in the best interest of end-users. The green bar on the top right corner will change as developers pick several items, hinting a loss of revenue as they block more categories.	233
C.18. FAN users' data policy is under developer's setting page, next to roles and permissions and notification. It is disabled by default to no limit for data use. 'Preselection' dark pattern happens here as data collection which is not in favour of user privacy is turned on by default.	233
C.19. TMP's permissions for older versions of Android in the <i>Integrate the MoPub SDK for Android</i> page.	233
C.20. TMP's content categories. Default blocked categories cannot be changed and are greyed out. Blocked items by the developer are highlighted by blue.	234
D.1. Screenshot from Google AdMob developer dashboard: Blocking controls -> Manage EU user consent (as of Jan'21).	240
D.2. Screenshot from Google AdMob developer dashboard: Blocking controls -> Manage CCPA settings (as of Jan'21).	240
D.3. Responses about who decides what revenue model and ad network to use in the apps participants develop.	241
D.4. Involvement in in-app advertising activities.	241
D.5. Perceived control over ad networks' data collection.	242
D.6. Expected change in app's revenue and number of users if personalised ads are chosen over non-personalised ads.	242

List of Tables

- 2.1. Publication venues reviewed papers drawn from. 18
- 2.2. List of all reviewed publications grouped by theme. 20

- 3.1. Stats for SO users and users in our subsets. 44
- 3.2. Stats for questions. 44
- 3.3. LDA topics and the top five words in the topic (SO-privacy). 48
- 3.4. Number of questions, total views, and sub-themes for each theme for the 294 qualitatively analysed questions. 51

- 4.1. Summary of participants’ demographics. 75

- 5.1. Interview study demographics. P = participant without computer security background; PS = participant who self-describes as having taken a computer security course in the past. 106
- 5.2. Topics mentioned during free-listing, number of words participants listed associated with that topic, number of unique participants listing at least one word associated with the topic, and a set of sample words representing the range. 108

- 7.1. Generalised linear mixed model regression. Outcome variable is the binary choice between personalised (coded as 0) and non-personalised ads (coded as 1). OR: odds ratios, CI: confidence intervals, conditional R²: .614 (represents how much of the variance is explained by the model [195]), No. observations: 800. 148
- 7.2. Constructed themes from participants’ answers about the primary reason for choosing the ad type. 150
- 7.3. Constructed themes around participants’ reasons for not including ad networks in their apps (*N* = 123). 155
- 7.4. Constructed themes around participant’s information sources for building their consent forms (*N* = 71). 157

- C.1. Presented privacy-related information on the ad networks’ pages. 225

- D.1. Summary of participants’ demographics and prior experience with ads (*N* = 400, unless otherwise specified). 243

PROLOGUE



1. Introduction

Computers store photos, personal messages, business communications, and financial records, enabling people to access them on-demand, on any device, in any location. One of the challenges that this digitisation introduces to people is keeping their digital belongings private¹ and secure.² To help people keep their digital belongings private and secure, computer scientists have been researching cryptography, secure programming practices, and privacy engineering to keep people safe from malicious parties. In the late 1990s, researchers found that a software system not only needs to provide a secure back-end but also should consider the *human factor* in its design [14, 370]; otherwise, a so-called secure system would likely fail to provide its promised level of security.

Since realising that the human factor is one of the critical elements in building private and secure systems, researchers have been looking at various technologies built on top of computer programs (e.g., operating systems, email clients, and web browsers [126]) to make them usable for users. The movement has provided insights into the human factor. It has shed light on the diversity of users such as designers and managers who formulate the requirements and shape of the system, software developers³ who write the code and get the system running, system administrators who keep the systems operating and alive, and end-users who use the software. This thesis focuses on the *developer factor* because decisions developers make, tools they use, and code they write and include in their programs impact end-users' ability to keep their digital belongings private and secure.

Like other engineering disciplines, developers rely on tools—any instrument that assists developers in programming and gives feedback about computer programs, such as application programming interfaces, libraries, integrated development environments, and graphical or command-line interfaces—built by others to accomplish their tasks. Notably, developers may not be experienced in privacy and security tasks such as cryptography, secure coding practices, and leaks of personal information to third-parties, and therefore rely on tools to accomplish such tasks. However, these tools may not be usable, leading developers to make mistakes and, consequently, leaving privacy and security vulnerabilities in their programs. In 2013, the widespread tool OpenSSL, which is used for establishing a secure connection between a client and a server, used an unintuitive argument

¹ '... the right to life has come to mean the right to enjoy life,—the right to be let alone; the right to liberty secures the exercise of extensive civil privileges; and the term "property" has grown to comprise every form of possession—intangible, as well as tangible'. (Warren and Brandeis, 1890)

² 'Protected from or not exposed to danger; certain to remain safe and unthreatened'. (Oxford English Dictionary, 2021)

³ Developers from now on, anyone who writes computer program.

1. Introduction

list that caused some developers to disable the security validations, leaving their programs open to a man-in-the-middle attack [128].

When looking at software's privacy and security aspects, the first component, *privacy*, is an ambiguous term in the software ecosystem and a challenging task to translate to technical requirements. Europe and the United States (California, particularly) recently introduced regulations to give users control over their data [72, 127]. As a result, many large tech companies started to adjust their data handling policies, giving users the option to opt-out from behavioural marketing or selling of their data to third-party companies. These changes also influence how developers work. They now have to work with privacy policies, consider personal information as sensitive content and know (or learn) how to handle such data, and think about what permissions they need to ask the user for when building apps.

When it comes to *security* tasks, developers use tools to accomplish these tasks, such as encrypting data and establishing a secure connection between a client and a server. However, security tools have usability shortcomings, limiting their ability to assist developers [135, 329], and sometimes even leading developers to misunderstand the vulnerabilities of their software. Due to an interface's deficiencies, developers may make mistakes or make assumptions that can have catastrophic consequences, such as thinking that the network connection cannot be exploited because they trust the tool, when in reality the transferred data could be stolen by malicious parties [113].

Motivated by developers' impact on software privacy and security, this thesis aims to look at the privacy component in software teams, focusing on the developer factor. Its main research objective is to understand what privacy means for developers and how much support they receive from tools, team members, and platforms provided by large tech companies to accomplish privacy tasks. I focus on the privacy aspects because a major body of research looks at the security aspects [326] and the research community is still in its early stages of scoping privacy aspects in software systems. Based on the findings, I provide recommendations to support developers in performing privacy tasks.

This thesis has been organised in the following way. Chapter 2 provides a systematic literature review of developer-centred privacy and security, synthesising the common themes in this area. It also identifies research gaps, some of which I focus on in the later chapters. In Chapter 3, I analyse privacy-related questions on Stack Overflow to understand what topics developers associate with privacy and what challenges developers face while interacting with these topics, providing a starting point for understanding privacy from a developer perspective. Chapter 4 is an interview study with developers who care about and advocate for privacy in the workplace. This chapter also provides insights about privacy practices and challenges in software teams.

Chapter 5 gives insights into computer science students' (as the future workforce of software development teams) considerations of privacy and security features in software design, and their mindsets about privacy and security. In Chapter 6 and Chapter 7, I focus on developer-facing interfaces in ad networks and look at how choices and options given to developers by software development platforms can impact their users' privacy. I also discuss the use of dark patterns (either intentional or unintentional) in these platforms as a way to interfere with developers' decision making. I end the thesis with my final thoughts about developer-centred privacy which includes a discussion, a future direction, and a conclusion section (Chapter 8).

1.1. A Survey On Developer-Centred Privacy and Security⁴

Taking a classic research approach, I first conducted a systematic literature review of developer-centred privacy and security. I collected papers that included privacy, security, software development, and the human factor, which resulted in 49 papers (as of October 2018). Chapter 2 presents the research themes, which include structuring software development, privacy and data, and tool adoption; and then identifies the research gaps, which include comparisons of students and professional developers, tools usability, and privacy support. I find that developer-centred privacy and security as a newly established field consists of several studies that look at the developer's privacy and security needs and requirements for building usable tools, however, primarily focusing on security aspects. Therefore, I decided to put more emphasis on the privacy aspects as it is currently under-investigated. In the following chapters, I focus on privacy challenges for developers, how privacy is championed in software teams, the privacy and security mindset of computer science students, and the usability of privacy interfaces for developers, which I identified as research gaps in this research review.

⁴This chapter was published as: Mohammad Tahaei and Kami Vaniea. A Survey on Developer-Centred Security. 2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW), Stockholm, Sweden, 2019, 129–138, DOI: [10.1109/EuroSPW.2019.00021](https://doi.org/10.1109/EuroSPW.2019.00021)

1.2. Understanding Privacy-Related Questions on Stack Overflow⁵

After the literature review, I realised that privacy had created new challenges for developers, and only a handful of papers look at the challenges of privacy for developers [326]. Therefore, I decided to look closer at this area by first identifying privacy-related issues that developers may face (Chapter 3). I chose Stack Overflow as a source for a qualitative study to determine problems developers associate with privacy. Stack Overflow is a resource where developers seek advice and solutions for software development tasks. I built my exploratory research questions around identifying privacy hurdles a developer may have:

- ▶ What topics do Stack Overflow users associate with the word ‘privacy’?
- ▶ What or who is pushing Stack Overflow users to engage with privacy-related topics?

I collected all Stack Overflow questions with privacy in their titles or tags ($N = 1,733$) and then picked a random sample for thematic analysis ($N = 294$). We found that developers have problems with privacy policies, setting permissions, and some of them do have concerns for user privacy and look for privacy-preserving solutions. Additionally, our results suggested that developers’ privacy concerns and questions were heavily driven by large tech companies such as Google, Apple, and Facebook. These companies are privacy influencers who define what content is considered sensitive and are major drivers that bring developers to Stack Overflow to ask privacy-related questions. Changes to privacy terms and introductions of new privacy terms from these companies can impact how developers think about and interact with privacy and, therefore, impact software’s privacy ecosystem.

1.3. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges⁶

To understand how privacy is done in software teams, how it is championed, and gather data about privacy from the people who do privacy in the field and

⁵ This chapter was published as: Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 2020. Understanding Privacy-Related Questions on Stack Overflow. Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, 1–14. DOI: [10.1145/3313831.3376768](https://doi.org/10.1145/3313831.3376768)

⁶ This chapter was published as: Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. ACM, New York, NY, USA, 1–15. DOI: [10.1145/3411764.3445768](https://doi.org/10.1145/3411764.3445768)

complement my artefact study with Stack Overflow, I designed an interview to understand additional resource developers depend on, such as peer support (Chapter 4). Motivated by champions’—who are a source of motivation and act as facilitators for new ideas and innovations [277]—contributions in teams, I studied a specific type of people in software teams who actively and passionately promote user privacy: privacy champions. I built my research questions to do an exploratory study for understanding this valuable group:

- ▶ What privacy champions find motivating, rewarding, challenging and frustrating in promoting user privacy in their organisations?
- ▶ What strategies and channels do they find the least and most effective in achieving that goal?
- ▶ What resources do they use to keep up with the latest in privacy?

The interviewed privacy champions ($N = 12$) actively engage with teammates about privacy topics, encourage others, and promote user privacy by having water-cooler conversations. They find it challenging to evaluate privacy, and the research community’s metrics are not yet utilised, suggesting a potential track for improvement. They perceive code reviews and practical training as more instructive than general privacy awareness and on-boarding training.

Privacy champions’ experience demonstrates that incorporating privacy considerations into design reviews has a bigger impact on the end-user privacy in the final decisions and products and yields better educational effects on developers, than company-wide awareness programs or on-boarding privacy training for new hires. When supported by management and a critical mass of other developers, privacy champions’ efforts may be effective in promoting organisational privacy culture and implementing *Privacy by Design* principles.

1.4. ‘I Don’t Know Too Much About It’: On the Privacy and Security Mindsets of Computer Science Students⁷

The privacy champions study’s findings showed that some of the privacy champions (3/12) were motivated by what they studied at university and their educational background in privacy and security. Therefore, I built an interview study with computer science students to understand their privacy and security mindsets and

⁷ This chapter was published as: Mohammad Tahaei, Adam Jenkins, Kami Vaniea, and Maria K. Wolters. “I Don’t Know Too Much About It”: On the Security Mindsets of Computer Science Students. 9th International Workshop, STAST 2019, Luxembourg City, Luxembourg, September 26, 2019, Revised Selected Papers. In: Socio-Technical Aspects in Security and Trust. Ed. by Thomas Groß and Tryfonas Theo. First Edition. Springer International Publishing, June 2021. DOI: [10.1007/978-3-030-55958-8](https://doi.org/10.1007/978-3-030-55958-8)

experiences with these aspects of software development (Chapter 5). My research questions were:

- ▶ What are computer science students' comprehension of privacy and security related concepts?
- ▶ To what extent do computer science students consider privacy and security while coding applications, and how do they implement it?

The interviewed computer science students ($N = 20$) had a range of hacker and attack mindsets, lack of experience with security application programming interfaces, a mixed view of who is in charge of privacy and security in the software life cycle, and a tendency to trust other peoples' code as a convenient approach to build software rapidly. They rarely brought up a privacy and security consequence when asked to design a hypothetical app suggesting that expecting developers to consider privacy and security in their design is not realistic as academia may not prepare, teach, and equip them with the required skills.

1.5. 'Developers Are Responsible': What Ad Networks Tell Developers About Privacy⁸

The Stack Overflow study's findings suggested that developers tend to follow the privacy requirements imposed by large tech companies such as Apple and Google. To understand the privacy interfaces of these platforms, I studied a controversial type of platform, advertisement networks, provided by large corporations for monetising software (Chapter 6) with one research question:

- ▶ What do ad networks tell developers about privacy?

I conducted a walkthrough of four popular ad networks with a senior Android developer. We find that the ad networks' documentation does not provide privacy details, and several of them do not provide information about how to build a consent form for users. Besides, the documentation interferes with developer decision-making by using anti-design patterns (dark patterns) that may nudge developers to make a decision that is not in the best interest of users but favouring the ad network. Our results extend the literature in the dark patterns community [204] to the developer-facing interfaces. We show that dark patterns may appear in the form of graphical interfaces and programming code snippets provided by software development platforms, where they may impact millions of people.

⁸ This chapter was published as: Mohammad Tahaei and Kami Vaniea. 2021. "Developers Are Responsible": What Ad Networks Tell Developers About Privacy. Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems Extended Abstracts. ACM, New York, NY, USA, 1–12. DOI: [10.1145/3411763.3451805](https://doi.org/10.1145/3411763.3451805)

1.6. Deciding on Personalised Ads: Nudging Developers About User Privacy⁹

The end-users literature in privacy and security nudges show that users' decisions are impacted by the information presentation such as the order of choices, colour of buttons, and chosen defaults [10]. The ad networks study (Chapter C) showed that the privacy-related information on ad networks' pages is inconsistent and uses dark patterns, and therefore, may impact developers' decisions about user privacy. The results informed an online experiment ($N = 400$) to understand the impact of choice framing and nudging on developers' decisions (Chapter 7). The research questions were:

- ▶ How does choice framing in ad networks impact developers' decisions about ad personalisation?
- ▶ What are the reasons behind developers' choices of personalised or non-personalised ads?

The findings suggest that framing and wording of choices impact developers' decisions. Participants who saw choices framed in terms of privacy consequences of personalised and non-personalised ads were 11.05 times more likely to choose a non-personalised ads compared to a condition where participants did not see privacy-related framing and wordings.

Our results suggest that developer-facing interfaces impact developers' decisions and developers deserve to use usable interfaces with transparent options. The use of nudges, either intentional or unintentional, by large software development platforms such as app stores and ad networks, may drive thousands of developers' decisions and consequently impact their users' privacy. We suggest policy makers to recognise implications of design and interfaces on developers' decisions and enforce policies that minimise the use of nudges and provide transparent options to all types of users, including developers.

1.7. Final Thoughts

Chapter 8 wraps up the thesis into a discussion section by putting all the findings into one concrete section, followed by several future work directions. I end this thesis with a final note as a conclusion section.

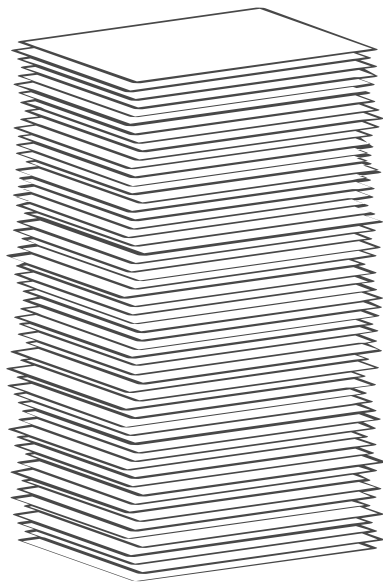
⁹ This chapter was published as: Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 2021. Deciding on Personalised Ads: Nudging Developers About User Privacy. The Symposium on Usable Privacy and Security (SOUPS) 2021.

1.8. Contributions

This thesis provides insights into *privacy*, a controversial topic that is being discussed in society, industry, and the research community, from a developer-centred privacy viewpoint. While research in developer-centred security provides invaluable inputs for assisting developers in doing security tasks, it may not cover areas that are privacy-related. My work aims to disentangle privacy from security and better understand privacy on its own terms. Privacy and security are two connected topics, but one can be achieved without the other. One may have a secure system, but if the builders of the system do not respect the privacy of its users, then information that those users wish to keep to themselves may be sold to third-parties. Therefore, I believe such separation between privacy and security supports arguments for dedicating time and resources to privacy alone.

My contributions in this thesis cover gaps that I found after my literature review [326]: most of the research papers look at the security domain or combine privacy with security, and privacy is an under-investigated area in the developer-centred security research area. I find that developers need support in accomplishing privacy tasks as much as they need support in security tasks. This support can come in various forms including online software development communities and forums, academia, and software development interfaces and programming tools provided by tech companies. Each chapter of this thesis gives details about the impact of these factors and how they can better support developers in performing privacy tasks.

RESEARCH PAPERS



2. A Survey On Developer-Centred Privacy and Security

Front Cover

To understand the scope of research in developer-centred privacy and security, I took a classic approach of doing a systematic literature review to collect related work, synthesise them into themes, and identify research gaps. This chapter provides the basis for my thesis. It provides a systematised overview of the relatively new field of developer-centred privacy and security which aims to understand the context in which developers produce privacy- and security-relevant code as well as provide tools and processes that better support both developers, and private and secure code production. I report here on a systematic literature review of 49 publications on privacy and security studies with software developer participants.

2.1. Introduction

Software is increasingly being integrated into all aspects of society, it controls everything from small home appliances such as kettles [224], to large systems like power plants [188], as well as data management infrastructures such as health records [202]. Each of these systems has a specific non-security goal (tea, power, health) but they also have an expectation from the public that they will provide other non-functional requirements such as safety, reliability, and privacy.

While it would be wonderful if developers could do all of that, instead we see that the introduction of security vulnerabilities into code is becoming a very large problem. The most common types of coding errors have remained relatively stable over time. Code Injection, for example, has topped the OWASP top ten vulnerabilities list for the last eight years [252] and 78.5% of recently scanned applications still suffer from it [352]. In 2013 alone, 88% of apps (out of 11,748) on Google Play had at least one cryptographic Application Programming Interfaces (APIs) mistake [106].

Developers, much like end-users [14, 287], need support to create applications that are functional, usable, efficient, maintainable, and secure. The emerging and rapidly expanding area of Developer-Centred Security (DCS) aims to address some of these needs by applying existing methodologies from Human Computer Interaction (HCI) to the area of software development and security [3, 135, 265, 376]. The application of HCI to software development has seen great success in the fields of API usability [226] and end-user software engineering [171].

To help developers make better use of security technologies many different approaches have been suggested such as security APIs [24], static [339] and dynamic [262] code analysis tools, and code creation processes such as pair programming [362]. However, many of these proposed solutions have had little success, potentially due to usability and workflow related issues. To systematically address these issues researchers need to understand the landscape of software development as it relates to both software developers and to security.

We present a formal structured literature review which identifies works that feature the trio of *security, software development*, and at least one study involving *users*. We identified 49 relevant research papers which we then sorted into eight themes: organisations and context, structuring software development, privacy and data, third party updates, security tool adoption, application programming interfaces, programming languages, and testing assumptions.

Our review highlights a lack of research in several aspects of DCS including, how to make security a business value and security often being ignored because it is a secondary requirement. To our surprise, even though programming languages are a fundamental tool in software engineering, only one paper discussed issues

around security evaluation of programming languages with user studies. Software updates, which are critical for software security and a point of discussion in end-user usable security [349], are rarely discussed in DCS literature; with only one reviewed paper considering the challenges of using packages and libraries.

2.2. Systematisation Approach

We used a Systematic Literature Review [244] approach in order to identify all relevant literature. Two authors were engaged in every step, and decisions were agreed by both researchers to minimise the effects of bias and priming.

Selecting Literature

DCS is a relatively new area which crosses several fields resulting in papers in a range of publication venues. We decided to cover the top five publications in three fields: HCI, Software Systems, and Computer Security & Cryptography. To select specific high-quality publication venues in those areas, we used a Google Scholar feature that ranks scientific publications based on their h5-index and h5-median. We chose the top five listed venues (Table 2.1). We also explicitly added the Privacy Enhancing Technologies Symposium, Symposium on Usable Privacy and Security (SOUPS), and IEEE Secure Development Conference. The first covers privacy areas, the second specifically targets usable security research, and the third is a newly established conference on secure development.

For the search itself, we used the The ACM Digital Library for ACM publications (SOUPS 2005–2013), IEEE Xplore digital library for IEEE publications, Engineering Village (EV) for PETS, International Cryptology Conference (CRYPTO), USENIX Security, Journal of Systems and Software, and SOUPS (2014 onward). We used EV because USENIX Security, CRYPTO, and ScienceDirect indexing websites do not support complex search queries. We limited our search to title, abstract and keywords. We also limited results to those published before October 13, 2018.

Practical Screen

The query in Listing 2.1 was executed on the 18 venues (Table 2.1) which generated 1922 results. The resulting publications were then loaded into a reference management software (*Zotero*) which was used to remove duplicate papers as well as store notes taken during screening.

2. A Survey On Developer-Centred Privacy and Security

Listing 2.1: Executed query.

```
("security" OR "privacy" OR "cryptography")
AND
("human" OR "empirical" OR "user" OR "users" OR "interview" OR "
  interviews" OR "survey" OR "surveys" OR "lab study" OR "laboratory
  study" OR "think aloud" OR "cognitive walkthrough" OR "questionnaire"
  OR "questionnaires" OR "usability" OR "usable")
AND
("developer" OR "developers" OR "development" OR "software" OR "app" OR "
  application" OR "programmer" OR "programmers" OR "software engineer"
  OR "software engineers" OR "system administrator" OR "system
  administrators")
```

One researcher then went through and applied the following screening criteria by looking at the title and abstract of each publication. The researcher intentionally took a slightly broad view of the criteria in the first pass erring on the side of inclusion rather than exclusion.

Security - The paper had to directly involve cyber security, though it did not have to be the primary topic.

Software Development - The paper had to involve the process of software development or at least code creation. Management practices that directly impacted developers were also included.

User Study - The paper had to include a user study involving research subjects. Studies that only had artefact analysis, such as only looking at code samples, or tools were excluded.

Full Papers - Posters and extended abstracts were excluded.

The first pass resulted in 46 publications marked for potential inclusion. A second researcher then went through and reviewed all the included publications also looking through the publication content, they identified publications which did not meet the criteria, these were then discussed and 18 were excluded. The final set included 28 papers.

Snowball

To further improve our coverage, we also use a snowball approach to find additional literature. For each of the identified papers above, a researcher read through the titles of all references to find any relevant-sounding papers not already identified. For relevant titles, they also read through the abstracts and full papers. Snowballing resulted in 21 new items. Three of these papers were earlier versions of one of the publications we had initially identified. However, they had not appeared in our

search results. The final set, including snowballing, included 49 papers. The first paper appeared in 2008 and the last paper was published in 2018.

Synthesise Methods

We identified a set of assessment criteria based on our own experience as well as criteria used in previous literature to evaluate papers [85, 104, 172, 175, 207, 281, 292]. One researcher then went through all the papers and extracted the methodology information. When uncertain, they discussed the outcome with the other researcher.

Synthesise Themes

Both authors reviewed the papers' main contributions in the final set and constructed an affinity diagram [86] to highlight the main contribution themes. Affinity diagrams are a grounded approach for sorting qualitative data into themes. In this case, we used the papers themselves as the unit of analysis and sorted them based on their primary topic. Both authors then completed a more in-depth reading of papers theme-by-theme resulting in several iterations to the themes and construction of sub-themes. While each paper is grouped under the theme associated with its primary contribution in Table 2.2, many papers touch on multiple themes.

Limitations

We limited our initial query to 18 venues which we selected from Google Scholar. While we believe that this approach created a strong starting point, one could argue that a different selection approach might be more relevant and results in a more appropriate sample. To catch studies that we could not find in our initial search, we carried out a snowballing method to include more papers. We chose to limit our review to papers with user studies in them to both scale the review to a reasonable size and to focus on papers which take a deep look at the human factors issues developers face.

Table 2.1.: Publication venues reviewed papers drawn from.

Publication	h5-index	h5-median	Total	Selected
Computer Security & Cryptography				
Computer and Communications Security	77	128	395	4
Security and Privacy	74	129	69	2
Information Forensics and Security	73	103	261	0
USENIX Security Symposium	70	106	66	1
International Cryptology Conference	62	84	65	0
Human Computer Interaction				
Computer Human Interaction	86	117	150	2
Computer-Supported Cooperative Work	56	79	34	2
Pervasive and Ubiquitous Computing	52	76	65	0
User Interface Software and Technology	45	72	17	0
Affective Computing	39	54	2	0
Software Systems				
International Conference on Software Engineering	74	102	154	6
Transactions on Software Engineering	56	83	79	1
Journal of Systems and Software	51	71	92	0
Foundations of Software Engineering	50	82	44	3
Programming Language Design and Implementation	50	78	30	0
Others				
Symposium on Usable Privacy and Security	31	60	265 ¹	10
Secure Development Conference	NA	NA	15	0
Privacy Enhancing Technologies Symposium	NA	NA	119	0
Snowballing	NA	NA	21	18 ²
Final set			1943	49

¹Has results from multiple sources.²All items are selected, three items fit into one of the lists above.

2.3. Methodology Results

We begin by discussing the methodology information presented in the papers in terms of three main criteria: research design, data collection, and data analysis. This document comes with following supplementary materials: A BibTeX file, an Excel spreadsheet, and database queries. The files are available on the [workshop's website](#) and <https://doi.org/10.7488/ds/2535>.

2.3.1. Research Design

Research Questions

Research questions show the purpose and outcome of the research [244] and are a vital component of an empirical research [104]. 'Why' and 'how' questions are preferred research questions in case studies [281, 382], such as 'How do users respond to and perceive the code generation and explanation approaches?' [379, p. 3] and 'Why do developers use CI [Continuous Integration]?' [155, p. 1]. Of our reviewed papers 26 explicitly state their research questions.

Pilots

When dealing with human subjects, it is prudent to conduct pilot studies before the main experiment [104, 207]. In software engineering studies that include human participants, it is advised to do pilot testing to ensure that software functions as expected during the experiment and that the tasks are clear [172]. Of our reviewed papers, 12 explicitly stated that they conducted a pilot study.

Context of the Study

Participants are effected by study context such as the lab setup, outside events, or their own expectations. Events occurring at the time of research can also impact participants' behaviour [175]. Of our reviewed papers 15 discuss the context of their study such as the time period and contextual information of the study. Only Sheth et al. gave a precise time period and elaborated on the events related to the study that might have influenced the results, i.e. NSA PRISM scandal in 2013 which could have influenced privacy concerns of participants during the study [299].

2. A Survey On Developer-Centred Privacy and Security

Table 2.2.: List of all reviewed publications grouped by theme.

Publication	Study Method	Participants	N	Research Question	Pilot Case	Context	Recruitment	Demographics	Ethics	Mixed Methods	Data Analysis	Quotes	Compare with Literature	Limitations	Study Materials
Organisations and Context															
Xie et al. [380]	Semi structured interviews	Developers	15	○	○	○	●	●	○	○	○	●	○	○	○
Xiao et al. [377]	Semi structured interviews	Software professionals	42	○	○	○	●	●	○	○	○	○	○	○	○
Witschey et al. [372]	Interviews	Developers	42	○	○	○	○	○	○	○	○	○	○	○	○
Witschey et al. [373]	Survey, survey	Developers	14, 61	○	○	○	○	○	○	○	○	○	○	○	○
Türpe et al. [342]	Survey, survey, observations, interviews	Mix	15, 12, 23, 15	○	○	○	○	○	○	○	○	○	○	○	○
Weir et al. [366]	Interviews	App security experts	12	○	○	○	○	○	○	○	○	○	○	○	○
Poller et al. [267]	Survey, survey, observations, interviews	Mix	15, 12, 23, 15	○	○	○	○	○	○	○	○	○	○	○	○
Haney et al. [148]	Semi structured interviews	Mix (with crypto background)	21	○	○	○	○	○	○	○	○	○	○	○	○
Assal et al. [35]	Semi structured interviews	Developers	13	○	○	○	○	○	○	○	○	○	○	○	○
Thomas et al. [335]	Semi structured interviews	Security experts	32	○	○	○	○	○	○	○	○	○	○	○	○
Structuring Software Development															
Bartsch [52]	Semi structured interviews	Mix (70% developers)	10	○	○	○	○	○	○	○	○	○	○	○	○
Yskout et al. [383]	Lab experiment (architectural design task)	Students	90	○	○	○	○	○	○	○	○	○	○	○	○
Edmundson et al. [105]	Online tasks	Developers	30	○	○	○	○	○	○	○	○	○	○	○	○
Yskout et al. [384]	Lab experiment (architectural design task)	Students	64	○	○	○	○	○	○	○	○	○	○	○	○
Acar et al. [3]	Survey, lab experiment	Developers	295, 54	○	○	○	○	○	○	○	○	○	○	○	○
Ur Rahman et al. [345]	Survey	Software practitioners	9	○	○	○	○	○	○	○	○	○	○	○	○
Hilton et al. [155]	Semi structured interviews, survey, survey	Developers	16, 51, 532	○	○	○	○	○	○	○	○	○	○	○	○
Privacy and Data															
Balebako et al. [45]	Semi structured interviews, online survey	Developers, mix (58% developers)	13, 228	○	○	○	○	○	○	○	○	○	○	○	○
Sheth et al. [299]	Survey	Developers, users	408 (267, 141)	○	○	○	○	○	○	○	○	○	○	○	○
Third Party Updates															
Derr et al. [94]	Survey	Developers	203	○	○	○	○	○	○	○	○	○	○	○	○
Security Tools Adoption															
Ayewah et al. [40]	Survey, interviews, lab study	FindBug users, FindBug users, students	400, 12, 12	○	○	○	○	○	○	○	○	○	○	○	○
Ayewah et al. [39]	Survey	Developers	252	○	○	○	○	○	○	○	○	○	○	○	○
Xie et al. [378]	Lab experiment	Students	9	○	○	○	○	○	○	○	○	○	○	○	○
Xie et al. [379]	Lab experiment	Students, developers	18, 9	○	○	○	○	○	○	○	○	○	○	○	○
Johnson et al. [165]	Lab study	Developers	20	○	○	○	○	○	○	○	○	○	○	○	○
Zhu et al. [386]	Lab study (programming)	Students	20	○	○	○	○	○	○	○	○	○	○	○	○
Zhu et al. [387]	Think aloud	Students	8	○	○	○	○	○	○	○	○	○	○	○	○
Thomas et al. [333]	Lab experiment	Students	28	○	○	○	○	○	○	○	○	○	○	○	○
Smith et al. [305]	Lab study	Developers, students	10 (5, 5)	○	○	○	○	○	○	○	○	○	○	○	○
Whitney et al. [369]	Field studies	Students	72	○	○	○	○	○	○	○	○	○	○	○	○
Christakis et al. [83]	Interviews, survey	Developers	5, 375	○	○	○	○	○	○	○	○	○	○	○	○
Assal et al. [37]	Cognitive walkthrough	Security experts, developers	4, 4	○	○	○	○	○	○	○	○	○	○	○	○
Thomas et al. [334]	Observations	Developers	13	○	○	○	○	○	○	○	○	○	○	○	○
Do et al. [101]	Task based	Developers (academics, professionals)	18 (9, 9)	○	○	○	○	○	○	○	○	○	○	○	○
Nguyen et al. [233]	Online programming task	Developers, Students	40 (16, 24)	○	○	○	○	○	○	○	○	○	○	○	○
Tabassum et al. [321]	Lab experiment (programming)	Students	23	○	○	○	○	○	○	○	○	○	○	○	○
Gorski et al. [133]	Online between subject	Developers	53	○	○	○	○	○	○	○	○	○	○	○	○
Application Programming Interfaces															
Fahl et al. [113]	Interviews	Developers	14	○	○	○	○	○	○	○	○	○	○	○	○
Jain et al. [163]	Lab experiment, programming tasks	Students	25	○	○	○	○	○	○	○	○	○	○	○	○
Oliveira et al. [245]	Survey	Developers	47	○	○	○	○	○	○	○	○	○	○	○	○
Oltrogge et al. [247]	Survey	Developers	45	○	○	○	○	○	○	○	○	○	○	○	○
Nadi et al. [227]	Survey, survey	Developers	11, 37	○	○	○	○	○	○	○	○	○	○	○	○
Lo Iacono et al. [192]	Survey	Developers	55	○	○	○	○	○	○	○	○	○	○	○	○
Acar et al. [2]	Online between subject	Developers	256	○	○	○	○	○	○	○	○	○	○	○	○
Naiakshina et al. [229]	Lab experiment (programming)	Students	20	○	○	○	○	○	○	○	○	○	○	○	○
Oliveira et al. [246]	Online programming task	Developers (professionals, students)	109 (70, 39)	○	○	○	○	○	○	○	○	○	○	○	○
Programming Languages															
Prechelt [268]	Lab experiment	Developers	27	○	○	○	○	○	○	○	○	○	○	○	○
Testing Assumptions															
Acar et al. [6]	Online between subject	Developers	307	○	○	○	○	○	○	○	○	○	○	○	○
Naiakshina et al. [230]	Lab study (programming)	Students	40	○	○	○	○	○	○	○	○	○	○	○	○

A full circle means that the paper has an explicit and clear statement for that criteria.

A half circle means that there are some efforts to cover the metric (implied or partially covered).

An empty circle means the paper does not contain information on the specified criteria.

2.3.2. Data Collection

Sample and Population

All reviewed papers reported their sample size. On average, the number of participants in quantitative studies (surveys) was 195 (range: 9–532, median: 55, SD: 165.4), and in qualitative studies (interviews, lab experiments, observations, and online programming tasks), it was 38.8 (range: 4–307, median: 20, SD: 58.4).

Providing a detailed demographics of the participants assists the reader in understanding the context and also gives a better sense of the results. Of our reviewed papers, 12 provide no demographic information about the participants. The remaining 37 provide a mix of different demographic information ranging from basic age/gender, to more complex measurements of experience. In software engineering experiments it is recommend to report experience with programming languages, technologies related to the tool, industry, and natural language [172]. For instance, Acar et al. presented a detailed table for participants' demographics which gave information about invited/valid participants and some extra information from developers' Github profiles such as public repositories/gists and followers/followings [6].

Recruitment

Participant recruiting and sampling can impact generalisability and therefore the types of conclusions that can be drawn from the research. 29 papers reported their recruitment strategy clearly. For instance, Acar et al. state how they recruited and compensated developers. They ask Github developers to donate their time for research which proved to be a suitable method of recruiting developers [6].

Ethics

When dealing with human participants in research, it is necessary to treat them ethically. One method of doing so is to have an ethics review board that reviews and approves research plans in advance of research [288]. In our paper set, we notice that several studies do not explicitly report ethics and only 13 of papers explicitly mention that they have an ethics approval, e.g., institutional review board approval. Additionally, two papers mention collecting informed consent from participants but do not mention ethics approval, we recognised this as partial fulfilment of ethics criteria.

Mixed Methods

Gathering data from multiple sources, e.g., interviews and surveys, provides a richer view of the topic being studied [104]. 13 studies use a mixed method approach. For instance, Nadi et al.'s results take advantage of two artefact analyses and two surveys [227].

2.3.3. Data Analysis

Data Analysis Process

A detailed report of all steps researchers take increases the validity, reproducibility, and gives a chance for other researchers to learn. For example, when the research involves qualitative analysis, it is often recommended to have at least two researchers work to analyse the data because one researcher could bring bias to the results [180]. However, in our sample, in several studies, data analysis is not described thoroughly, and in a few cases, the authors do not mention it explicitly at all. For example, Hilton et al. gave a link to the online codebook [155]. Additionally, 15 studies publicly shared all their study material. For instance, [155, 227, 299] provided a link to all their artefacts.

Quotes

Quotes bring evidence to the report [104]. Therefore if a study contains video, audio or written material from participants, it is beneficial to add quotes to the report. In our sample, 34 papers include quotes from participants. For example, Jain and Lindqvist incorporate relevant quotes in the text which fosters the results validity and also work as examples of how authors interpret the interviews [163].

Situate

Proper research highlights its position among other works, highlighting similarities and differences [104]. 31 of the reviewed papers discuss similar works in background; however, comparison of the results with other papers is sometimes overlooked. 14 papers compare their work with previous literature both in the background and discussion section.

Limitations

Study design decisions typically introduce limitations which need to be reported carefully so that other researchers can accurately draw conclusions from the research [292]. Bringing such issues to reader's attention increases the validity of the study as it gives more context and shows that the authors are aware of potential threats to their results [85]. Some common limitations in our sample are recruitment, small sample size, and generalisation of the results. 34 papers have an explicit section to elaborate on their limitations.

2.4. Research Theme Results

2.4.1. Organisations and Context

Developers work within the larger context of organisations, teams, communities, and cultures. These social structures impact many elements of development from the types of tools a developer may be allowed to use by their organisation [372, 373, 377] to the assumptions of how to best handle non-functional requirements (NFRs) [267].

Non-Functional Requirements

Security is often referred to as a NFR in that it is expected to be included as part of high quality code development, but is rarely listed as an explicit requirement [267]. As a result, developers prioritise security below more-visible functional requirements or even easy-to-measure activities such as closing bug tracking tickets [35, 267, 377]. Pressure-related issues like budget and deadlines can also cause security to be prioritised lower [380]. To quote a participant from Poller et al. 'If security is not on the list [of features], then is it really worth the time and extra energy to do it?' [267, p. 11]. The non-functional nature also made security challenging to assign as a task or decide who's job it is. This issue extended to tools such as libraries where developers assumed that issues like security were already correctly handled [35]. Managers generally view security as an important NFR, but expressing security downward is challenging to do without compromising team autonomy or creating more bureaucracy. A manager described it as one of the code quality 'ilities' along with other NFRs such as usability, scalability, and maintainability [267].

Some organisations attempt to use external pressures such as penetration testing to motivate developers and help them understand the value of security, but without internal sustained support the motivation tends to lose priority compared to the

functional requirement deadlines imposed on teams [267, 342, 377]. Similar to developers, managers are asked to make risky decisions and may choose to release code with known problems if doing so is aligned with business needs [335].

Clients do not necessarily prioritise security when providing feature lists unless the software they want has an obvious security focus, such as financial software, or the contracting organisation initiates discussions on the topic. This lack of guidance from clients forced developers to derive security guidance from the other functional requirements, or initiate a discussion about security needs with clients themselves [52].

Dedicated Security Teams

One issue is where to locate security within an organisation. Obviously it would be best if developers built security in from the start, but doing so requires a large amount of knowledge which takes time to learn and it can be difficult to self-motivate, especially when security is not seen as a measured functional requirement [267, 342]. Developers are also much more likely to learn about security because they are enthusiastic about it and want to know more, than if they are task driven [366].

Alternatively, security knowledge can be concentrated in a testing team or a set of security experts which act as a kind of roving source of security knowledge. While good at their jobs, these teams must convince others of the importance of security to get their changes made, they are also limited in their throughput and unable to minutely examine all generated code [335]. External penetration testing organisations can also be contracted to find security vulnerabilities. While effective at their primary task of finding problems, these penetration tests do not necessarily motivate developers to make long-term changes to their practices [267, 342]. Developers perceive security as the security team's problem and do not attempt to gain knowledge or install support tools on their own [35, 377].

Communication Around Fixing

Communication between testers, auditors, and developers is also a challenge. Security auditors consider a large part of their role to be motivating and convincing developers of the importance of identified vulnerabilities, especially when the vulnerability produced no visible problem [335]. Developers have difficulty seeing how a vulnerability could be practically exploited and find specific examples of actual attacks motivating [377]. Correcting vulnerabilities also required understanding it, which takes communication effort and occasionally results in dedicated training sessions run by the security team for often-observed problems [267, 335].

Security experts also struggled with having to communicate with many different teams, all of which have their own priorities and communication cultures [335].

Champions

Security champions are an often unofficial role held by someone on the development team who has limited security knowledge but considers security to be important and is willing to champion it. Champions are useful in getting security into products in a variety of ways. By putting champions in teams, managers can positively impact the security of a product without having to provide top-down pressure [267]. Security testers valued champions highly enough to find budget to send them to places like security conferences despite them being located on different teams [335].

Security Oriented Organisations

Structure and practices become different when companies focus heavily on security. These companies have a culture of security and they apply it in every step of development. They do not ignore security, or sacrifice it for the sake of releasing the product earlier. The customers in this market also demand security, and security is part of the culture and mindset of every entity involved in the process. Third party libraries are a point of discussion, some companies have to trust them because they are either certified by standard organisations or have been used by the community for a while, therefore they can trust them [148].

2.4.2. Structuring Software Development

Security Design Patterns

Security design patterns provide solutions for common problems faced during code design, such as how to best provide feedback to a user about password strength. Such patterns are widely used in HCI design, but currently are not as successful in security. Yskout et al. conducted a lab study with 32 teams of students, some with and some without access to the security pattern catalogue. Surprisingly, participants ended up with similar results in terms of productivity and security [384]. When the security pattern catalogue was annotated (security objective, applicability, trade-off labels, and relationships) participants found it easier to locate a suitable pattern [383].

Software Development Methodologies

There are many methodologies for how to ‘best’ develop and deploy code as a holistic process, each of which have positive and negative impacts on security. Software development methodologies (SDMs) provide guidance on how to structure code development. Some SDMs, such as Agile, are feature focused where developers prioritise features each week and then work to get them implemented. The short iterations make it challenging to do full-stack testing on every iteration [52].

Continuous integration (CI) systems ‘automate the compilation, building, and testing of software’ [155, p. 1] such that developers are encouraged to integrate their work frequently allowing for fast testing. CI is good in that it enables frequent automatic testing of code. However, it also faces practical problems such as needing to give the automated systems access to protected machines, the potential that the tested code is malicious, and the difficulty of allowing developers to run the same tests on their own computers which have less access. Hence, CI can introduce complexity and further security challenges to a project [155].

The fast nature of testing can also lead to prioritisation of automated tests over more manual tests such as penetration testing [345]. Manual code reviews can also be expensive and impractical as it needs several reviewers to look at a piece of code to find vulnerabilities [105]. While effective, automated tests can easily overlook unexpected issues, leading security experts to use both methods when reviewing code [335].

Information Sources

Information sources, such as documentation, are important to software developers especially when interacting with topics like APIs [2, 227]. To determine the effects of various information sources on code security, Acar et al. conducted a lab study comparing four information sources: official documentation only, StackOverflow only, book only, and free choice [3]. They found that participants that used StackOverflow were more likely to create functional code, but less likely to produce secure code.

2.4.3. Privacy and Data

Privacy is a complex topic and how people generally manage it is a well researched topic outside the scope of our research. However, privacy as a software requirement has only been marginally studied. Sheth et al. conducted a survey of Europeans and North Americans and found that Europeans were significantly less willing to give up privacy for functionality [299]. They also find that ‘data privacy is often an implicit requirement: everyone talks about it, but no one specifies what it

means and how it should be implemented . . . While almost all respondents agree about the importance of privacy, the understanding of the privacy issues and the measures to reduce privacy concerns are divergent' [299, p. 9].

Developers are also not always aware of software privacy issues, nor do they endanger users' privacy deliberately. Balebako et al. showed that only 1/3 of App developers ($N = 228$) knew what data was being collected by third-party tools [45]. Revenue models are a determinant of how much data developers collect which means if an app's revenue model is ad-based, it is likely to collect more data than a paid app. Company size is also a player in privacy; smaller companies tend to be less privacy-conscious [45, 377].

2.4.4. Third Party Updates

When vulnerabilities are discovered in software, developers (hopefully) fix them and release an update to their code. Like any other software, libraries can have vulnerabilities and updates to address those vulnerabilities. One potential source of insecure software happens when a library author updates their code, but developers who use that library do not switch to the updated library. An analysis of over a million apps showed that 85.6% of the libraries could be updated by simply changing the version number of the library [94]. Developers avoid updating libraries primarily because the update may break their app or require them to spend time adjusting to new library structures. When they want to understand potential library changes, they use change logs [94].

2.4.5. Security Tool Adoption

Software is commonly written with the assistance of tools such as Integrated Development Environments (IDEs). Several research papers have endeavoured to understand the tool needs of developers, improve existing tools to better support security, or build new security-focused tools.

Security tools generally see poor adoption by developers. To understand why, researchers have surveyed [39, 40, 83, 373] and interviewed [40, 83, 165, 334, 377] developers and auditors. They find that organisation and team policies are a driving factor to tool adoption [83], though many organisations do not encourage their use [39, 40], larger organisations make more use of security tools than small ones [377], and peer tool use positively impacts use [373, 377]. Though, surprisingly, having more concern about security did not lead to greater security tool usage, but having an academic background or training in security did [373]. Existing tools also exhibit pain points by checking for the wrong types of problems by default, having poor warning messages [40, 83], interrupting work flow [83, 165, 334], having too many false positives [83, 165], not providing enough support for

2. A Survey On Developer-Centred Privacy and Security

team work, and integrating poorly with IDEs [165]. When using tools, developers want to know about the type of attacks, available solutions, vulnerabilities, and flow of data in their programs [305]. Visualising tool output also helps [37].

Several works focus on the creation and evaluation of specific tools which are often built as plugins for popular IDEs. FindBugs is a static analysis tool that looks for coding defects by analysing software in the abstract, allowing it to identify issues at the logical level. It has been the subject of multiple research projects which considered both the context of use and usability needs in its design and evaluation [39, 40, 305]. New tools which focus on usability, also use it as a starting point and a basis for comparison [37].

The Application Security IDE (ASIDE) is a static analysis Eclipse plugin helps developers find potentially vulnerable web application code ‘in situ’ as they code, similar to the underline in a word processing spell checker [378]. The initial user study with students had mixed results [378], but a larger follow-on study found students using ASIDE improved in their post-usage code security knowledge [386]. Another ASIDE study compared giving graduate students auto generated code fixes to giving them explanations of identified vulnerabilities [379]. They found that students given code solutions were more likely to implement them. ASIDE was then further improved to allow developers to annotate security-critical code so that the tool could provide better analysis [333, 387]. However, they found that student participants did not have enough security knowledge to accurately provide the annotations.

Two other studies also proposed and tested fast-feedback tools. CHEETAH provides feedback-on-compile warnings to developers using an IDE static taint analysis plugin [101]. Their study compared their feedback-on-compile approach to feedback-on-request and found that developers both preferred the feedback-on-compile approach and that they fix bugs faster. However, the feedback-on-request also introduced a time delay, possibly confounding results. FixDroid is an IDE plugin for Android developers that highlights insecure code, and on mouse over provides a short explanation and recommendation to fix the problem [233]. They find that developers using FixDroid’s feedback produced more secure code.

Gorski et al. designed an online programming experiment (study design modelled on [2]) with Python developers to find out whether showing developers API-integrated security warnings would help them with security APIs [133]. These security warnings include a warning and an example of a secure code snippet. Developers who had access to warnings and code examples created more secure code compared to who did not have access to warnings.

Education

We did not explicitly search for educational approaches in security; however, three papers showed up in our results. Tabassum et al. investigated two approaches to teaching security and secure coding, teaching assistant vs. tools [321]. Educational Security in the IDE (ESIDE) is an Eclipse IDE plugin, which provide fast feedback warnings, detailed explanations, and suggested remediation code. The user study had some serious limitations, but generally found that students found ESIDE interesting, used the suggested code samples, but did not engage with the educational information and did not try and fix errors on their own [321]. Two other papers look at the potential benefits of a similar tool in teaching secure coding [369, 386].

2.4.6. Application Programming Interfaces

Considering Options

While designing and coding, developers must make decisions about many features, including privacy and security. Many external pressures, like deadlines often overload developers, quality and functionality and security is often not a part of developers' decision-making process. Security is often overlooked because it needs extra effort and it is 'blind spot' in developers mindset [245]. API blind spots are the points where developers incorrectly use an API, often because they just trust the API to do the right thing without doing additional checks [246]. However, if developers are nudged with the security, they change their programming approach and consider security and secure programming [229, 245]. The involvement of customers in security also impacts decision making, and while customers are often unclear about their security requirements, developers felt that customer engagement on security topics was helpful [52, 380].

The features provided by readily available APIs can also impact privacy and security choices of developers. In a case of geo-location libraries, developers who have access to a library with privacy-preserving options are more willing to use coarse location information over those with only access to the standard library [163].

Testing the Usability of Security APIs

Security APIs (SecAPIs), as a subset of APIs, are built to help developers with security concepts such as encryption and authentication. They offer a level of abstraction and work as a layer between developers and the low-level details that developers may get wrong on their own. SecAPIs broadly fall into two categories,

security primitives which required security knowledge to understand and *security controls* which were found to be a more appropriate abstraction level for developers. Unlike other APIs, SecAPIs do not well support learning-by-doing [192].

Acar et al. compared the usability of five Python cryptographic APIs and found that good documentation with examples is more helpful to developers than just having a simple API [2]. They also found that when developers do not find a solution easily, they tend to look online, and typically find an insecure solution on StackOverflow. Resulting in the concerning situation where developers think they have a secure solution, but actually have an insecure one. Another study similarly found that developers struggle to use SecAPIs finding them overly complex for basic tasks and they want better documentation and higher abstraction levels [227].

Transport Layer Security (TLS) and its predecessor Secure Sockets Layer (SSL) are common protocols to encrypt data in transit. Unfortunately, developers have difficulty using these protocols correctly [113]. Fahl et al. tried notifying app developers of TLS-related vulnerabilities in their apps [113]. However, after three months 34.6% had not made corrections. Similarly, Oltrogge et al. found that developers found TLS pinning too complex of a topic to use [247].

2.4.7. Programming Languages

Only one paper conducted a laboratory study of the relative usability of different programming languages (PLs). Undoubtedly more such studies exist based in artefact analysis, but Prechelt's study is notable in that: 1) all the development teams had the same brief, 2) PLs were assigned (Java, PHP, Pearl) and 3) developers had two days to complete the brief [268]. Theoretically, this design allows for a more causation-style analysis than artefacts provide. However, they find no definitive links between PLs and security, suggesting that security issues may be more develop-based than language-based.

2.4.8. Testing Assumptions

While several developer studies use students as their subjects, it is important to know if this group is representative of professional developers. In a Github sample, status (being a student or a professional) and security background were not a significant factor in terms of the security and functionality of the final code, and the only distinguishing feature was years of experience [6]. In contrast, a different study observed no impact of years of experience on security [105, 230]. These studies provide an initial suggestion that students might be representative of developers with a similar level of experience.

Naiakshina et al. ran a lab study (similar to their 2017 study [229]) with forty CS students to find out if priming effected the production of secure code [230]. In their 2017 study they used an interview to gather data from participants, and in their 2018 study they used a survey instead of interviews. Results show that interviews generated valuable results, particularly in developer studies because this field is still at its early stages, so interviews allow for more flexibility in data collection. An interesting side takeaway is that PL experience is not a deciding factor in developers security which contradicts with findings in [6]. Another contradictory result is around copy/pasting behaviour. Acar et al. show that copy/pasting results in insecure code [3], but [230] observed that using copy/paste tended to result in more secure code.

2.5. Discussion

2.5.1. Methodology and Ability to Generalise

Developers are a challenging group to study because their work is undertaken over an extended time frame, collaborative, involves multiple stake holders, and requires decision making based on a combination of prior experience and online research. These features make it challenging to create a lab or online study that properly replicates the experience of software development without making it long and expensive. To handle these problems we see a heavy reliance on retrospective and opinion studies like surveys and interviews where developers can reflect on past work. For lab studies involving tools, we see efforts to use code that the developer is already familiar with possibly as a way to better evaluate the tool [233, 386]. However, in our opinion, some of these efforts worked poorly, such as asking professional developers to write a class example as a way of getting them to write something small, but as a side effect they also explicitly left out some checks because adding them would be confusing to students [379].

Length of the study was also a tricky subject. Longer lab studies took as much as 3 days [268] and shorter ones took as little as 15–20 minutes [321]. Most of the ASIDE studies lasted for about 3 hours and had developers use their coursework as the code base to make starting easier. Our impression from reading many such papers, is that studies that allowed less than 3 hours tended to suffer from the developers not having enough time to really get involved with the programming or interact with the tool in a realistic way.

As a field, DCS is relatively young and many of the methodologies have not caught up to the standards of HCI research. For example, several methodologies mentioned conducting a think aloud study where they interrupted the participant whenever they did something interesting. While that is an acceptable approach

in industrial research where the goal is to find big problems, in academic HCI research interrupting the participant mid-task is known to impact their behaviour and disrupt their work flow which invalidates later results. It was also surprising how few studies attempted to compare their results to a control or similar tool on the same tasks. Acar et al. did this by making the control be a book [3]. But many other studies only tested their new tool and made no attempt to even compare against other tools on the same task. Much less base their tasks to other study setups.

2.5.2. Research Gaps

We identified several research gaps while reviewing that we feel are important to fill, though this list is hardly exhaustive.

When to Interrupt the User

The majority of the tools in our reviewed papers took the view that providing fast feedback *in-situ* was a good thing because it would give developers a chance to make changes right away. However, this assumption is only weakly tested in [101] which looked at feedback-on-compile vs. feedback-on-request. But given their confounds, the question is still open. The assumption is also at odds with how humans write text and when it is best to interrupt them doing so [119]. Best practice in that area suggests waiting till a user has reached the end of a text passage before interrupting with feedback. But the equivalent of finishing a text passage in code is not necessarily clear at this stage.

Are Students Similar to Professional Developers?

Many papers use students as participants under the assumption that they are similar to real software developers (and easier to find). However, there were only two studies [6, 230] in our set that compared students to developers and their findings disagree on several points. Shortly after we completed our review, Naiakshina et al. released a new paper [228] which roughly duplicated their earlier work on password storage conducted with student participants [229]. They find that students and developers have difficulty securing passwords and that paying more for developers does not improve the situation. While this lends some support to the view that students are similar to professional developers, the issue is still open.

Tools

While there is abundance of work in security tool adoption (Section 2.4.5), the work focuses on only a few IDE plugins, yet several other security plugins are available [54] which could benefit from usability analysis. For example, SpotBugs is a successor of FindBugs, and comparing it with FindBugs [306] could be quite interesting, especially given that most papers focus on only a single tool. In relation to Section 2.4.1, there is also a question of how these tools, e.g., static analysis tools, integrate with developers' workflows, and how these tools can be used in organisations.

Testing Support for Team Development

No study in our review tried simulating team or collaborative aspects of programming. Several studies commented on the frustrations developers have with tools that do not well support team work. While this concern is minimally elaborated on, issues like having to share the particulars of a code problem with the security team or needing to add output to a bug tracker come up elsewhere. It would be interesting to see studies of how developers communicate security issues in teams and the types of details that need to be shared and tracked.

Learning Support

Due to online resources having a negative impact on code security [3], offering reliable documentation to developers in real-time as an IDE plugin could be beneficial. Tools such as ESIDE [321] and FixDroid [233] do provide educational feedback which tends to be ignored in studies by task-driven developers, and example solutions which developers do appear to use. However, both of these works are initial case studies and do not look at different warning texts, styles or even the appropriate length of education to provide to developers. Links to 'good' external resources are also not tried.

Privacy Support

Privacy features can be complicated to define and include in software requirements. Efforts to bring privacy into design [141, 177] are currently under research though more research is required as it is a hot-button issue, especially with the recent passing of GDPR in the EU [127]. Privacy perceptions and security mindsets effect decisions (see [361] for average users' mental models in security). This correlation becomes more relevant when developers are responsible for building tools that can impact people's day to day lives. More research is needed to find out privacy

mindsets of software developers and how to support them in developing privacy aware tools for the general public [295, 297]. Cultural differences are another major theme here. International companies treat the data and privacy of users around the globe based on their home country. However, a single geographical area is not a representative sample of all countries [289]. Hence, more research is needed to learn about how software developers and companies can adapt their technologies to various cultures.

2.6. Conclusion

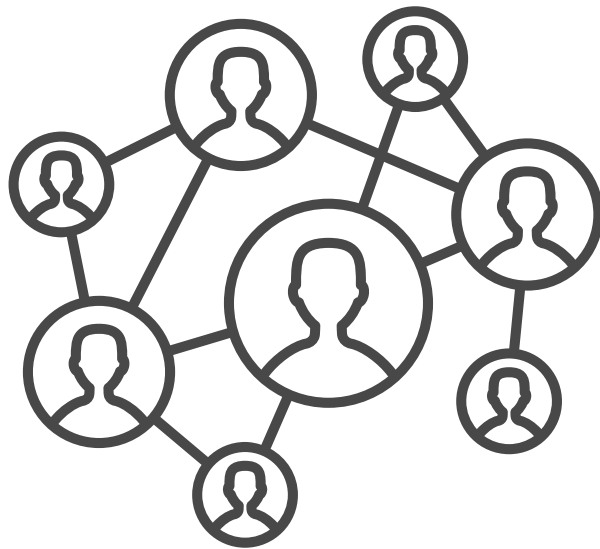
This paper reports a systematic literature review of 49 research papers that are DCS related. Every study has at least one user study of software developers. We discuss our sample set from two viewpoints, methodology and findings. In the former, we look at research design, data collection, and data analysis of papers as a sample set. We observed similar issues in DCS research addressed in the literature [85, 172]. In the latter, we synthesis outcomes of all studies. Eight themes emerge from our dataset: organisations and context, structuring software development, privacy and data, third party updates, security tool adoption, application programming interfaces, programming languages, and testing assumptions. Our results facilitate entry of early researchers to the field of DCS and assist research veterans in discovering areas that are not yet researched thoroughly and need more investment.

Acknowledgements

We thank Kholoud Althobaiti, Tariq Elahi, Adam Jenkins, Maria Wolters, and everyone associated with the TULiPS-Lab at the University of Edinburgh for helpful discussions and feedback. We also thank the anonymous reviewers and our shepherd whose comments helped improve the paper greatly. This work was sponsored in part by Microsoft Research through its PhD Scholarship Programme.

Back Cover

After the literature review, I find that privacy aspects are under-investigated in the developer-centred privacy and security and this area can be further explored. Therefore, in the next chapters after the literature review, I study four areas related to developer-centred privacy: privacy-related questions on Stack Overflow, privacy champions, privacy and security mindset of computer science students, and privacy interfaces in ad networks.



3. Understanding Privacy-Related Questions on Stack Overflow

Front Cover

In the literature review, I found that privacy is yet hard to define in software teams, and developers find it difficult to contextualise and conceptualise what privacy means and what tasks are related to privacy. Therefore, I decided to analyse one of the largest question and answer websites directed at software developers to understand their challenges when interacting with privacy interfaces and in general understand what topics they associate with privacy.

My co-authors and I apply topic modelling techniques to 1,733 privacy-related questions to identify topics and then qualitatively analyse a random sample of 315 privacy-related questions. Identified topics include privacy policies, privacy concerns, access control, and version changes. Results show that developers do ask Stack Overflow for support on privacy-related issues. We also find that platforms such as Apple and Google are defining privacy requirements for developers by specifying what 'sensitive' information is and what types of information developers need to communicate to users (e.g., privacy policies). We also examine the accepted answers in our sample and find that 28% of them link to official documentation and more than half are answered by Stack Overflow users without references to any external resources. While Stack Overflow is only one of the platforms that developers use, studying it enabled us to understand what challenges developers face while interacting with privacy-related technologies. This study contributes to the area of developer-centred privacy by showing that developers need further assistance in privacy tasks that may not appear in lists of classic functional software development tasks.

3.1. Introduction

When designing software, developers have to make a range of decisions that impact many aspects of the software such as efficiency, maintainability, and privacy. Developers located in large organisations may have access to dedicated staff with training in such topics to assist them, but for many developers, they are expected to incorporate these features into their code on their own. This observation begs the question of how developers manage privacy in software as well as how they interpret and think about privacy-related coding issues.

Privacy and security can be challenging for developers to get right, even with the support of tools [37, 101, 233]. Developer errors are a common source of vulnerabilities [106] with many causes ranging from APIs with poor developer support [2, 227] to static analysis tools that produce too many false positives [165]. Privacy can also be challenging for small organisations where their income depends privacy-unfriendly monetisation methods such as ad networks [45, 210].

Privacy, as a social norm, can define how security is being implemented as a technological requirement [47, 102]. While prior research has found several reasons for developers' poor security practices [135, 265], we know comparatively little about the privacy challenges and concerns they face. Efforts to introduce privacy into technical levels such as privacy by design [157] are still nascent, and there is a gap between these frameworks and how software developers approach privacy [141].

Stack Overflow [368] (SO) is one of the largest developer Q&A platforms and defines itself as 'an open community for anyone that codes'. It attracts a wide range of developers who ask questions about programming, security, and data management [62, 280, 381]. SO's dataset has been heavily used for research on such topics as: what factors makes it a successful Q&A platform [198], security issues developers face and how they interact and build knowledge around it [193, 381], and the negative impact of SO code snippets in software security [3].

Our research combines techniques from the literature on SO analysis with questions about the privacy-related tasks of developers. Our research questions are:

RQ1: What topics do SO users associate with the word 'privacy'?

RQ2: What or who is pushing SO users to engage with privacy related topics?

To answer our research questions, we collect SO questions that mention 'privacy' in the title or tags and then apply topic modelling and manual qualitative analysis methods. We find that developers ask questions when dealing with permissions, access control, encryption, and privacy policies. Similar to other works [62, 280, 338], we look at question types such as 'how' questions that ask for instructions and help, 'conceptual' when they look for advice and suggestion in early stages of development, 'errors' which includes crashes, and 'unexpected' which

includes surprises from updates or features being added or removed. We further analysed the accepted answers, which shows that 28% of those link to official documentation.

3.2. Related Work

3.2.1. Stack Overflow

SO is aimed at software developers, covering various topics such as website development, databases, version control, and security [53]. It has an Alexa rank of 43 [20] and more than 50 million unique visitors per month (as of September 2019) [250].

SO Users - SO surveys developers every year and publishes the results. The 2019 survey includes responses from 88,883 software developers from 179 countries in which 85.6% of respondents are SO users. Most respondents 'said they are professional developers or who code sometimes as part of their work, or are students preparing for such a career' [95]. Over 85% visit SO at least a few times per week, with over 60% visiting every day and 96.9% using it to find answers to specific questions. 73.9% are employed full-time at companies whose size ranges from 'just me' to '10,000 or more employees'.

Impact of SO on software security - Developers utilise SO knowledge and code snippets to build their projects [25, 273, 375]. A study of 289 open-sources projects showed that 30.5% of projects contained code matching code found on SO with some modification [375]. However, code reuse from SO can also introduce vulnerabilities [3, 4, 115]. For example, Fischer et al. found that snippets from SO questions that contained security-related code were observed in 15.4% of applications on Google Play (1.3m apps), and 97.9% of those apps had at least one snippet with insecure code [115].

Researchers have also studied the topics developers talk about; including analysis with natural language processing techniques (NLP) [21, 53, 62, 280, 338, 381] and manual qualitative techniques [62, 161, 193, 194, 227, 232, 259, 274, 338]. For example, an analysis of questions about Puppet, a configuration language tool, shows a need to support Puppet syntax error finding [274].

Topics - Prior topic modelling of security SO questions found five main categories: web security (51%), system security (19%), cryptography (17%), software security (9%), and mobile security (4%); with popular subjects including: password, hash, signature and SQL injection (out of 30,054 posts) [381]. Such outcomes can help both industry and researchers to understand better the challenges developers are facing. For example, injection (such as SQL, NoSQL, LDAP) and broken

authentication such as passwords, keys, and session tokens are the two top risks in OWASP's ten most critical web application security risks [253], which are similar to the findings of Yang et al. who also studied SO questions [381].

Question types - Questions posed on SO can be a good indicator of the areas of development SO users require guidance on. For example, they ask questions around library features, then clarify optimal implementations once they are confident with basic functionality. They will ask for solutions, workarounds and explanations when their code has errors or unintended features. Finally, they may ask for improved solutions with best practices [21, 62, 232, 280, 338].

3.2.2. Privacy and Developers

There is no unified cross-discipline definition of privacy [304]. Daniel J. Solove describes privacy as 'too complicated a concept to be boiled down to a single essence' [308, p.485], so he instead made a taxonomy of activities that potentially can be harmful to privacy: *information collection* (e.g., surveillance), *information processing* (e.g., identification), *information dissemination* (e.g., disclosure), and *invasion* (e.g., decisional interference) [308]. In the engineering realm, privacy is defined as a set of requirements collected from stakeholders. For instance, software developers are expected to pay attention to activities that can threaten privacy in information systems such as data transfer, storage, and processing [314]. *Notice and choice*, *privacy-by-policy*, *privacy-by-architecture* [314], and *Privacy by Design* (PbD) [78, 79, 157, 174, 374] are some examples among many other frameworks which include practices and guidelines to bring privacy into the design space.

Prior research uses PbD to understand the privacy practices of software developers and development [57, 136, 141, 157, 295]. Ann Cavoukian, who coined the term, describes PbD as 'assures an end-to-end chain of custody and responsibility right from the very start' [79, p. 406]. PbD thus aims to bring privacy into the system development process [138].

PbD for developers - Semi-structured interviews with 27 developers showed that they interpreted the concept of privacy as a set of smaller concepts, such as security, confidentiality, purpose specification, and consent. In contrast, concepts such as notice, minimisation, and rectification were not mentioned by many participants. Participants reported that they were familiar with other privacy concepts, such as user transparency and automatic expiration date, yet admitted they used these technologies infrequently [141].

Interviews with senior engineers show that privacy is seen as a burden which no one views as their own responsibility as well as a concept that is hard to define because it is wrapped up in legal jargon [57]. These results are similar to a study that was carried out 15 years previous, which indicates stagnation in

engineer's mindsets [57]. Beyond interviews, a discourse analysis of two mobile developer forums for privacy relevant conversations found that developers of these forums were concerned about how third-parties are collecting data, the privacy implications of features requested by end-users, and the legal consequences of their actions [300].

Developers are one of several privacy decision-makers - The costs and effects of developers' choices and mistakes in software systems can be enormous [113, 128]. These decisions, however, are influenced by the choices made in designing the systems they are dependent on, including platforms, APIs, and human organisations. For example, mobile platforms shape the privacy mindsets of their developers; iOS developers are more concerned about 'notice and consent' as Apple promotes it, while Android developers advertise privacy as an extra feature to stand out in the market [136].

API design influence developer choices. A lab study with developers given the choice between coarse and precise location APIs found that they chose the coarse location option [163], providing more privacy. Nudges and help from documentation [42], models [187], and IDE plugins [185] can also assist developers in privacy-friendly software development.

Organisation internals are another key factor in the privacy and security practices of developers [35, 36, 45, 141, 380]. For example, the size of the company influences the privacy behaviour of developers; larger companies are more concerned about having a privacy policy (PP) [45]. Moreover, some developers follow practices suggested by their employers, such as programming languages and tools [36]. They also benefit from the advice of security experts in their organisation [36].

3.2.3. Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) [64] is a common method of topic modelling. It is an unsupervised method, meaning that the topics are not labelled by humans, but are discovered naturally through patterns of clustering in the data. For example, LDA might discover that documents fall into two topics, one in which typical words include (baseball, bat, pitcher), and another in which these common words are (neural, Gaussian, marginalised). A human annotator is needed to label these topics as 'baseball' and 'machine learning', as the model does not assign labels. Note that the word 'statistics' could easily signify either topic; vocabulary is not exclusive to a single topic, but has different distributions according to topic. LDA models text generation as a two-step process: first, a mixture of topics is sampled through the Dirichlet distribution, then a mixture of vocabulary items is sampled from the Dirichlet distribution associated with each topic. The model assumes that the words in a document are sampled by selecting a topic from the mixture of topics and a word from the mixture of words associated with that topic. We

interpret these topics by inspecting the words most indicative of each topic. We take advantage of this automation to analyse a larger dataset than is feasible with human annotation. The approaches in Section 3.2.1 use LDA to find topics in SO questions [21, 53, 62, 280, 338, 381].

3.2.4. Our Contribution

A systematic literature review of developer-centred security shows that few papers study the intersection of developers and privacy, and further research is needed in this area [326]. Our work contributes to this research area by studying SO privacy-related questions using both automatic (LDA), and manual (qualitative coding) approaches. Our approach is a bottom-up analysis which builds upon questions developers asked when they faced a privacy-related problem or felt the need to dispel confusion on a related topic. This study complements existing interview work in the privacy space.

3.3. Method

3.3.1. Data Collection

We collected three data sets from SO; each composed of question and answer text as well as metadata such as the number of views and votes. *SO-all* is the set of all SO questions. We use this set to provide comparison statistics. *SO-privacy* is the set of all SO questions where the word ‘privacy’ appeared in either the question title or tags ($N = 1,733$). The term ‘privacy’ was selected after iterating on several alternatives and finding minimal improvement of quality. We use this set for most of the quantitative analysis, including the LDA topic model. Finally, *SO-privacy-rand* is a set of 315 questions randomly selected from *SO-privacy* and is used in the manual qualitative coding. Figure 3.1 shows a sample privacy-related question. All data was collected using the Stack Exchange Data Explorer [110]. The research was conducted in accordance with our institute’s ethics procedures.

Looking at *SO-privacy*, the first question was created on 02 August 2008 (for *SO-all* it was on 31 July 2008), and the most recent was created 17 August 2019. 1,428 questions have at least one answer, and 790 have an accepted answer. Tables 3.1 and 3.2 provide a comparison between the data sets in terms of users and questions. Figure 3.2 shows the number of questions asked by year and Figure 3.3 shows the top 50 tags assigned by askers in *SO-privacy*.

How to disable Google asking permission to regularly check installed apps on my phone?

Asked 5 years, 11 months ago Active 1 year, 5 months ago Viewed 103k times

▲ I'm developing an Android app, which I therefore endlessly build and install on my test device. Since a couple days I get with every build/install a question asking

83

▼ Google may regularly check installed apps for potentially harmful behaviour. Learn more in Google Settings > Verify apps.

★ 13 I get the option to Accept or Decline. I've declined about a hundred times now, but it seems to be Google's policy to keep on asking until I get sick of the message and finally click Accept. But I don't want that!

So my question: how do I let Google know once and for all that I do not want them regularly checking installed apps on my phone?

android permissions privacy policy

edited May 9 '14 at 6:54 asked Oct 9 '13 at 7:35

kramer65 11.4k 68 205 352

Particularly need a solution for this to support automated UI testing, e.g. with Espresso, because the APK can't even be installed on a new emulator instance unless the Accept/Decline button is clicked. Is there a `GrantPermissionRule` (developer.android.com/reference/android/support/test/rule/...) for this? – Michael Osofsky Apr 4 '18 at 19:23

10 Answers

▲ On Android prior to 4.2, go to **Google Settings**, tap **Verify apps** and uncheck the option **Verify apps**.

98

▼ On Android 4.2+, uncheck the option **Settings > Security > Verify apps** and/or **Settings > Developer options > Verify apps over USB**.

✓

edited Apr 3 '14 at 11:14 answered Oct 9 '13 at 7:42

Helen 39k 5 95 150 Sunny 9,397 7 47 80

23 Ah! I just now see it under Settings > Developer Options > Verify apps over USB. Sorry, I just got so sick of this message and the fact that I couldn't find the setting.. – kramer65 Oct 9 '13 at 7:45

8 Not in Settings app find the Google Settings app on your phone. – Sunny Oct 9 '13 at 7:46

Ah, and I had never heard of the Google settings app either.. Cheers! – kramer65 Oct 9 '13 at 7:47

It's the default settings app ;) – CommonGuy Oct 9 '13 at 8:33

2 On Android 5 I had to use the Google Settings app. Verify apps over USB was grayed out in the Developer options. – Rolf Nov 16 '15 at 15:57

Figure 3.1.: A sample privacy-related question with an accepted answer.

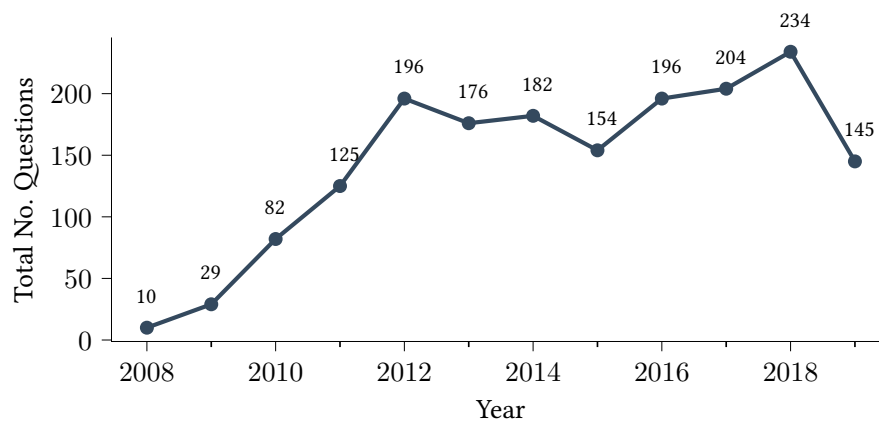


Figure 3.2.: Count of questions mentioning privacy per year (SO-privacy).

Table 3.1.: Stats for SO users and users in our subsets.

	Reputation ¹	Views ²	Up Votes ³	Down Votes ³
All users (10,901,490)				
Avg	106	14	11	1
SD	2,312	708	180	361
SO-all question askers (3,501,541)				
Avg	2,631	389	270	29
SD	12,929	3,780	903	428
SO-privacy question askers (1,684)				
Avg	3,430	448	268	49
SD	18,453	1,974	706	648
SO-privacy-rand question askers (312)				
Avg	4,889	602	312	110
SD	25,413	2,927	785	1,135

¹Can be gained by posting good questions and answers.

²Number of times the profile is viewed.

³How many up/down votes the user has cast.

Table 3.2.: Stats for questions.

	Score ¹	Views	Answers	Comments	Favourites ²
SO-all (18,123,431)					
Avg	2	2,279	2	2	3
SD	23	18,419	1	3	20
SO-privacy (1,733) - Used for LDA and qualitative analysis section					
Avg	3	1,416	1	2	3
SD	16	7,338	2	2	11
SO-privacy-rand (315) - Used for coding findings section					
Avg	4	1,378	1	2	3
SD	25	5,281	1	2	8

¹The difference between up votes and down votes.

²Similar to bookmarking a question.

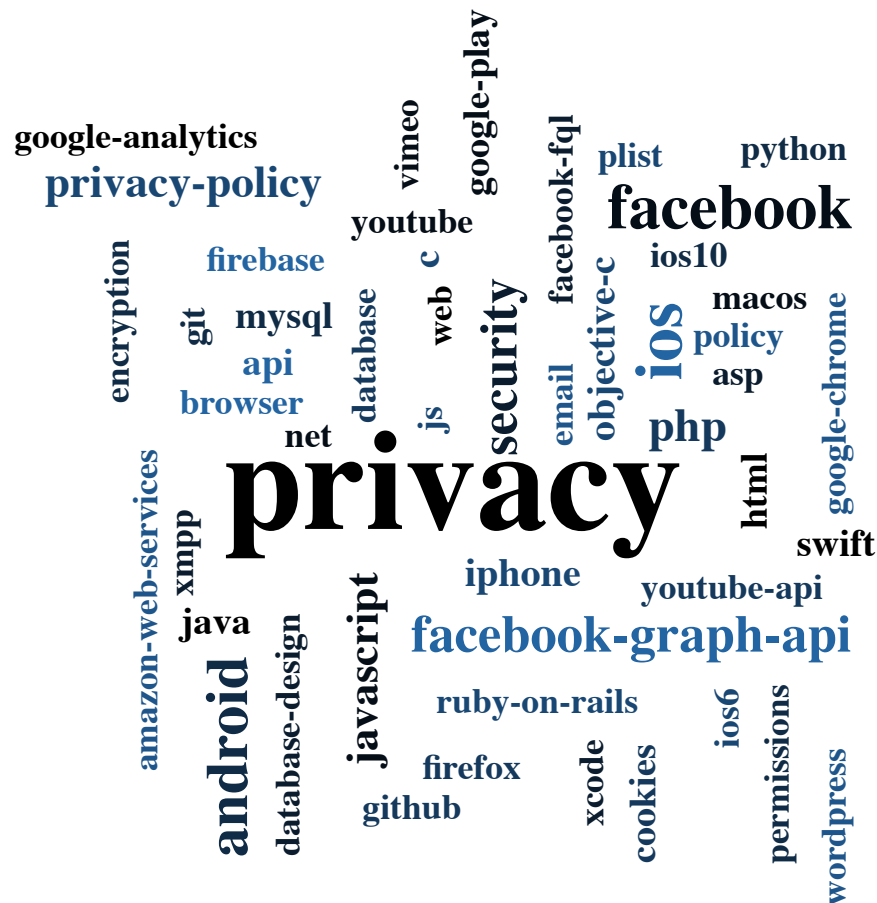


Figure 3.3.: Top 50 most commonly used tags by users (SO-privacy).

3.3.2. Topic Modelling

Documents were formed from SO-privacy by concatenating the question title and body, lemmatised with stop words removed using spaCy [311]. All code samples and URL details were removed so the topics would be based only on natural language data. We ran a bigram LDA at 2,000 iterations, with a variety of topic counts, from 5 to 60. After discussions among researchers, we selected 15 topics as the best setting.

3.3.3. Qualitative Analysis

Two researchers first independently read through 40 questions drawn at random from SO-privacy, and also reviewed the output of the LDA topics. Then during multiple discussion sessions and meetings they shared their observations and identified four interesting elements of the questions deserving of further analysis: 1) the question type, based on existing taxonomies [21, 53, 62, 338]; 2) the driver that makes the user need to ask the question (e.g., compiler error, client requirement,

or Facebook warning); 3) the aspect of privacy that the question relates to (e.g., setting app permissions); 4) accepted answers.

Question Type

In prior work, question type focuses on the shape of the question, such as ‘how do I . . . ?’ questions. After reviewing both the question types found in prior work [21, 53, 62, 338] and the shapes of questions found in the SO-privacy set, we narrowed the question types to: 1) conceptual questions that ask for higher level explanation, as well as moral, legal, and regulatory advice; 2) unexpected behaviour the asker wants to be explained; 3) error questions where the asker provides an error and asks how to fix it or why it is happening; and 4) questions looking for instructions, solutions, and best practice.

Coding procedure - After the question type codebook was solidified, both researchers coded 10% of the data. The question types inter-rater reliability kappa was 70%. One researcher coded the rest of the data for question types, and the other researcher coded another 10% to make sure they did not drift apart and have a similar understanding of the data. Their final kappa was 77% which is considered as a good agreement [285].

Drivers

A driver is the event, technology, or motivation that caused the asker to post a question on SO. Some drivers are expected, such as getting a compiler error, while others are more unique to our data, such as concern over how to comply with the General Data Protection Regulation (GDPR). Our practice questions cited many reasons for interacting with privacy, such as requirements from clients, concern about laws, and the development platform (e.g., Facebook) giving privacy warnings that prevented code deployment. Unlike question type, drivers were quite varied and not easy to classify a-priori. Therefore, we decided to use open coding. One researcher went through all the questions and provided one or more open codes. A second researcher did the same for 10% of the dataset. We do not report the kappa values as they were open coded.

The two researchers then completed a thematic analysis [180] of the driver codes, resulting in four themes: 1) feedback from platforms such as operating systems (OS) or companies (e.g., Facebook, Google Play, Apple Store), 2) personal concerns and business reasons (e.g., company or client requirements), 3) laws and regulations such as GDPR, and 4) too vague or unclear to code properly.

Privacy Aspect

The privacy aspect of a question describes how it relates to the concept of privacy. SO questions can be complex and contain multiple parts, not all of which involve privacy. For example, an asker wants to make sure users scroll to the bottom of the PP page before the 'accept' button activates, but is having trouble with the way the fonts are showing on the page. In this case, the privacy aspect is ensuring users read the PP. Similar to the drivers, privacy aspects appeared to have a wide range which was hard to categorise a-priori. Therefore, we decided to open code the privacy aspect. Because aspects seemed to involve both a subject (PP, camera) as well as an action (create, change, use), coders were encouraged to create open-codes that contained both subjects and actions, where appropriate. For example, 'create a PP' or 'read camera permission state'. As with the drivers, one researcher open coded the whole dataset, and the other coder did the same for 10%. Two researchers then grouped the codes into themes using thematic analysis [180]. We do not report the kappa values as they were open coded.

Accepted Answers

One researcher analysed the accepted answers, that is 'When a user receives a good answer to their question, that user has the option to "accept" an answer' [251], and coded them into these categories: 1) provides a solution, explanation, advice, opinion, sample code by an SO user, 2) links to another SO question, 3) when there is a link to an official documentation with or without any further explanation, and 4) links to an unofficial resource with or without any further explanation.

3.4. LDA Findings

Table 3.3 shows the 15 LDA topic clusters generated from SO-privacy and their researcher-generated labels. The topics include a wide set of common privacy and security concepts such as access control, secure storage, data management, confidentiality, user consent, human factors, and tracking.

Apps are a large issue for developers, with terms like 'app' occurring in multiple topics as well as the names of platforms that host apps such as Google and Facebook. App-related concepts such as permission settings are also a clear cross-cutting topic ranging from photo to location permissions.

Server-side issues also appear across several themes such as database design, handling of sensitive data, encryption, blockchain, handling account access, and storing passwords. The topics suggest that developers are encountering privacy

3. Understanding Privacy-Related Questions on Stack Overflow

Table 3.3.: LDA topics and the top five words in the topic (SO-privacy).

Topic Label	Top Five Words
1 Access to and read contents	app, application, use, android, privacy
2 Set the privacy field	user, privacy, like, page, facebook
3 App purchase and user registration	device, ios, cloud, feature, access
4 Privacy and permission settings and dialogues	app, user, privacy, access, ios
5 Crash reporting, analytics tools, and trackers	crashlytics, tracker, integrate, news, advertiser
6 PPs in Google Play and Android	app, policy, privacy policy, privacy, store
7 Concerns about using Google services	google, button, use google, ad, click
8 Publicity of sensitive data in code repositories	analytics, firebase, repository, google analytics, git
9 Design a db schema with privacy settings	table, privacy, column, transaction, mysql
10 Privacy values in Facebook, YouTube, and plists	privacy, post, set, facebook, api
11 Image privacy statements in Instagram and Windows	image, windows, statement, instagram, privacy statement
12 Store users' sensitive data securely	datum, user, use, address, information
13 Access to, create, and upload photos and albums	photo, album, picture, save, access photo
14 Private and public variables	file, private, privacy, use, code
15 Browsers errors (cookies and security settings)	use, privacy, website, browser, site

not just as part of user-facing elements such as dialogues and alerts, but also in the design of their back-end infrastructure.

We also see a topic on public/private variable scopes (topic 14). Examination of questions associated with this topic show typographical errors where the user wrote 'privacy' when they meant 'private'. While this topic is outside our scope, it is nice to see it neatly forms a distinct topic.

We find that 'want' and 'need', indicating that the asker is attempting a specific task as in 'I need to access a file', are highly-ranked in topics 1, 2, 6, 11, 12, and 14. This behaviour can be connected to the qualitative question type *How*. 'Thanks', a marker of politeness and possibly of discomfort with the SO community, is in the top 20 words indicating topics 1 (content access) and 10 (Facebook/YouTube/plists). This politeness divide may indicate differences in the background and persona of users interested in those topics.

3.5. Coding Findings

Of the 315 randomly selected in SO-privacy-rand, 21 were excluded due to either being about private variables (scoping) or being too vague to understand. This section focuses exclusively on the remaining 294 questions. Because the research is bottom-up, we decided to use SO users' definition of 'privacy' to understand their usage of the word rather than force our understanding of it. Consequently, the only posts we excluded were clear misspellings, most commonly those confusing 'privacy' with the scoping word 'private' as in public/private classes. This confusion was common enough to appear in the LDA results (topic 14 in Table 3.3). One

interesting result of this user-lead definition is that some clusters are technically more security-focused or more UI-focused. But in all cases, the asker explicitly used the term ‘privacy’ in the title or tags indicating that they thought the question was privacy related in some way. The quotes included in this section are referred to by the question identifier and date.

3.5.1. Question Types

How (186, 63%) - These questions include instructions, solutions, best practices, and possibilities: ‘I’ve used my personal email address for [Git repository] commits and I’m trying to set it to another one, before I make the repository public. . . . Is there a way to remove it from there, too, without losing my history?’ (13323759–2012).

Abstract or Conceptual (50, 17%) - These questions ask for explanations, legal/policy/requirements advice, background information on a component or process, or further conceptual understanding. The asker’s goal was to get advice about legal, policy, regulation, moral, or ethical implications:

What is the hidden cost using these CDN services? If the script is not cached by the browser and it loads the script from google what could google potentially do with the information? Could it be usefully extrapolated in conjunction with other services such as search, analytics or adsense? Nothing is free, what’s the catch? (10133816–2012)

Error (46, 16%) - These questions quote a specific *error message* to understand the provenance of errors, exceptions, crashes, or even compiler errors. Includes warnings that are blocking progress to working project state (compilation, upload to store, etc.), including emailed ‘fix this’ warnings from platforms. Questions containing compiler or similar errors are regularly observed on SO [62, 232, 280, 338]. Notably, the privacy questions quote warning messages from platforms: ‘I still get privacy error with “NET::ERR_CERT_AUTHORITY_INVALID” in the browser when I hit the ELB url using https . . . ’ (45295709–2017).

Unexpected (12, 7%) - The asker wants some observed unexpected behaviour explained. Includes surprise due to features having been added or removed with a new version as well as unexpected behaviours that arise from OS or device revisions. A common example was the sudden addition or removal of permission dialogues when the developer switches to a new API version or different behaviour on different OS versions: ‘I set microphone permission in info.plist file so record audion permission alert displaying in iOS 10.3.2 but its not appearing in iOS 10.3.3 devices’ (46297966–2019).

3.5.2. Drivers

The largest driver was *personal concerns, client or company requirements* (144, 49%). This finding is unsurprising, as this group includes cases where no driver is explicitly cited. Common external drivers in this group included a client requesting a feature, or commentary on what an app's end-users wanted. The second most common driver was feedback from a *platform* (136, 46%). This finding also makes sense since many third-party platforms, such as Facebook, have requirements that developers must follow. A common issue was that Google requires a URL to a PP if sensitive permissions are being used, resulting in several askers turning to SO to understand either why Google thought they were using sensitive permissions, or how to create a PP that met Google's requirements.

Drivers coming from *laws and regulations* (5, 2%) were least common. These included concerns around topics like GDPR or speculation about if an action was or was not legal. In SO-privacy-rand, we only observed question about EU regulations; however, in the broader SO-privacy sample, we observed mentions of regulations from other countries, such as the USA's Health Insurance Portability and Accountability Act (HIPAA).

3.5.3. Accepted Answers

Answers contain sample codes, explanations, links to and quotes from other resources, opinions, hints, and screenshots. Out of 130 questions with an accepted answer: (76, 58%) were answered by *SO users*; (36, 28%) had a link to *official documentations*, websites, blogs; (17, 13%) had a link to *unofficial resources* such as websites, blogs, Wikipedia, an app, or a GitHub project; (4, 3%) were pointed to *another SO question*. Dual coding occurred in links to another SO questions in which two had a link to an official doc (included in the official group as well), one had a link to an unofficial doc (included in the unofficial category too), and one provided a link to another SO question.

For links to unofficial sources, Wikipedia and GitHub were most common. GitHub occurred eight times as a source for referring to issues and bugs, projects and pages that could be helpful to the asker. Wikipedia was used as a source for further details and explanation of concepts in five answers (with the concepts: AOL search data leak, flag fields, segmentation fault, ePrivacy Regulation (European Union), and P3P).

Table 3.4.: Number of questions, total views, and sub-themes for each theme for the 294 qualitatively analysed questions.

Theme	Sub-themes (separated by ';')	No. Questions	Total Views
Access control	Dealing with privacy settings; I'd like to do it, but how?; UI elements; Browsers.	119 (40%)	103,654
Privacy policies	How to do it?; I got an error while trying to implement it; I have got an error in usage descriptions; Do I need a privacy policy? Why?	39 (13%)	127,225
Encryption	How do I achieve it?; Tell me more about it.	10 (3%)	11,100
Privacy and code issues	-	5 (2%)	2,523
Versions and updates	Device and OS versions cause unexpected results; Updates cause unexpected results.	11 (4%)	22,269
Developers with privacy concerns	How to implement privacy?; Can I trust this service or company?; Tell me about it.	71 (24%)	57,136
Developers ignoring PbD principles	-	18 (6%)	9,681
Developers as end-users	How do I protect my data?; Privacy in version control systems (Git); I have privacy concerns, thoughts?	21 (7%)	89,279

3.6. Privacy Aspect Thematic Analysis Findings

Each subsection describes the (sub)themes, number of questions, percentage, and the number of question views associated with the theme. Table 3.4 gives an overview of the themes.

3.6.1. Access Control

SO users often struggle to find information about updating and changing the privacy status of posts, images, and videos on social networks. They also ask about how to implement systems that have different levels of access control.

Dealing With Privacy Settings

When SO users want to set the privacy field of a post, image, or video on social networks (Facebook, Youtube, Vimeo, Google Calendar) they may not be able to find the right values, keys, and features needed to do so. For example, finding how to set the privacy settings of videos on Vimeo via API: 'How to change a Vimeo's video privacy via API (PHP)? . . . I've followed every step specified by the Vimeo's API Documentation but I can't get it to work. What am I doing wrong?' (52080930–2018). Another user is looking for which privacy setting are available through the API: 'Which Facebook Privacy settings can be accessed through API? I'm about to start an ASP.NET project which uses Facebook API to get/set Facebook Privacy settings . . . or is there any other way to access other privacy settings, too?' (9093704–2012).

3. *Understanding Privacy-Related Questions on Stack Overflow*

Errors messages can happen when doing things like: accessing restricted resources on an OS, setting the status of posts on social networks, or defining custom access control. The user below is trying to develop a messaging app for iOS with private and public lists, but received an error: 'error:Error Domain=com.quickblox.chat Code=503 "Service not available." So if all privacy list works perfectly then how can my blocked users could send me messages?' (27665795–2016).

I'd Like to Do It, but How?

Other entities such as OS or personal drivers lead SO users to ask questions about how to handle access control or provide it to users. For example, 'We develop a rails-based healthcare application. What is the best way to configure our s3 implementation so that only the authenticated user has access to the image?' (30602560–2018).

SO users also look for practices to design databases which provide levels of access control to users:

Database Design - Users and their privacy . . . It's a good choose? I'm not quite sure if i should create a new table to handle the privacy settings. I must admit that database design isn't my specialty so really need some feedback about this. (5211799–2011)

Askers in this code tend to express personal or 'right thing' motivations for adding access control to their databases.

UI Elements

When a privacy dialogue pops up, developers want to get notified about the user's decision so they can react by making changes to the interface or logic of the program: 'It would be a simple thing to reload the view content once the user grants permission, but I'm having a surprisingly hard time finding a method that is called when that happens' (29338752–2015). Drivers for these questions come from platforms forcing access controls and permission requests.

Browsers

Browsers have several features, such as cookie blocking and certificate checking, that are intended to protect users' privacy and security. However, these features can cause issues for developers, such as getting certificate errors when they are using 'localhost' during testing, managing cookies, and generating certificates. 'My question is how to setup valid SSL certificate on localhost? or do I need to edit my configuration?' (35565278–2016). Similarly with cookie blocking: 'If we set on

IE11 privacy settings to medium, we successfully get our value from session, but if we set to “Block All Cookies” - we get null. What can cause it? How to avoid?’ (24059471–2014). The driver for these questions generally comes from browser behaviour and errors.

3.6.2. Privacy Policies

Developers are often compelled by law, forced by platforms or are personally motivated to provide privacy policies (PPs) to users. SO users ask conceptual questions around PPs as well as more specific questions about how to write them.

How to Do It?

When writing a PP for their apps, SO users have to deal with multiple aspects of composition: wording, technical changes to make their code compliant, effects of third-parties such as analytics libraries, platforms’ PP interfaces, and reusing PPs on multiple platforms. For example, complying with GDPR: ‘Due to GDPR I am requiring to check the users location whether the user is from European Union’ (50253418–2018). Another user reacted to a news article about Apple’s policy against analytics tools and is concerned that their app might be rejected by Apple because of third-party libraries: ‘I’ve just integrated Crashlytics into my code, app is still waiting for review . . . My question is should we be worried by using Crashlytics & Firebase’s screen tracking (analytics). Will Apple object it?’ (54658427–2019).

I Got an Error While Trying to Implement It.

Questions in this theme deal with errors users got from the platform while trying to publish their app:

i can’t publish my facebook application, when i click “yes” on Status and Reviews of developers platform i see this message “You must provide a valid Privacy Policy URL in order take your app Live. Go to App Details and make sure it is valid.” in privacy field i have a right url and i tried also to change it with others but continue to see the message. this happens not just for one application but also for others. (26944634–2018)

I Have an Error in the Usage Description.

Apple, in particular, forces ‘usage description’ for accessing restricted resources such as contacts and location. SO users ask questions about errors and crashes they get during development because they do not know how to set these values. They are also confused about messages they receive from Apple after submitting apps without the correct usage descriptions: ‘iOS 10 GM release error when submitting apps “app attempts to access privacy-sensitive data without a usage description” due to GoogleSignIn, AdMob’ (39383289–2018).

Do I Need a Privacy Policy? Why?

SO users are confused about why or if a PP is necessary. For simple apps, it can be unclear if a PP is even necessary, or even what the definition of ‘sensitive data’ is.

My app’s operates on a simple couple of button clicks. However, as I am gearing up to release it, I couldn’t help but notice nearly all the apps have at least a privacy policy and terms/conditions on there page. Is it legally necessary to have both? Or is it just good practice? (56606092–2019)

3.6.3. Encryption

A fairly small set of questions fall into the encryption theme. Most have a personal motivation or a client requirement.

How Do I Achieve It?

Users asked questions about how to implement encryption solutions.

What could be the best solution to store this data encrypted in a remote database and that only the data’s owner could decrypt it? How to make this process transparent to the user? (You can’t use the user’s password as the key to encrypt his data, because you shouldn’t know his password). (39772–2013)

Questions about encryption errors are also asked: ‘I’m using the GnuPG class from PHP. I’m not having any problem importing valid public key but if I try to import something random like “test” which obviously isn’t a public key, I’m getting error 502 bad gateway.’ (34557651–2016)

Tell Me More About It.

These questions ask for further information about encryption solutions.

Since the salt is used to add a huge range of password possibilities . . . what is the purpose of letting the salt insecure? . . . Is there something that I dont understand? I know that knowing the salt dont break the security but, saying that it “need not be kept secret” sounds strange to me. (6176848–2011)

3.6.4. Privacy and Code Issues

This theme includes errors that are specifically code level and raised due to a function call, security flag, and static analysis tools:

We use HPE to check the code potential risks, i got one critical issue below in Log util class “The method d() in LogUtil.java mishandles confidential information, which can compromise user privacy and is often illegal”. how can i do to fix this? (44410004–2017)

3.6.5. Versions and Updates

SO users ask questions when they observe OS and platform behaviours that violate their expectations or desires.

Device and OS Versions Cause Unexpected Results.

Multiple versions for OS and devices can cause frustration for SO users. They test their code on one OS or device, and expect the same behaviour on others. But, this is not always a valid assumption: ‘But sometimes iPhone 5s running iOS 8.4 and always iPhone 6 Plus running iOS 9 does not show my app under the privacy photos list’ (32646366–2015).

Updates Cause Unexpected Results.

Updates to OS, platforms, and PPs can be a pain point for SO users: ‘I want to give the users of my App the option to control which lists their actions show to by default. The new API seems to have taken a feature away because I can’t see where that control is!’ (7523282–2012).

3.6.6. Developers With Privacy Concerns

This theme includes questions which are generally in alignment with PbD principles such as minimise, hide, abstract, control, enforce, and inform [157]. Askers in this theme looked for solutions to collect less data, mask personal information, remove unnecessary data, minimise tracking, and other approaches to protect privacy.

These users ask questions about protecting resources such as cookies, location, handwritten documents, browsing habits, IP address, data to build charts and graphs, messages, email address, contacts, Apple ID, phone number, card number, names, health data, country, phone calls, patient health information, personal documents, Facebook activity, images, driver licenses, device IDs, browser history, birth dates, social security numbers, passwords, videos, and the phone's screen.

How to Implement Privacy?

Questions in this theme ask about developing privacy-preserving solutions. The motivations come from either personal concerns or requirement from clients. 'My add displays private data, so I don't want it to be possible to see the app contents in the task switcher' (13260462–2012). Similarly:

I want to mask PII (personal Identification Information) like Name. Birth Date, SSN, Credit card Number, Phone Number, etc. It should remain same formate , means it looks like real data. And shouldn't be reversible. And it should take less time to mask. (22387577–2016)

Can I Trust This Service or Company?

Specific questions around trusting services are gathered in this theme. The motivations for these questions are either personal or business reasons. For example, when users want to decide to use services (an API or a product) in their projects, they have questions about how much they can trust it with their data and intellectual property:

Can I trust react-devtools not to breach my privacy? . . . The tool (and react) is made by Facebook, a company infamously known for their complete lack of moral when it comes to data gathering and creepy surveillance of us all. And it requires the ability to access everything you are browsing (which is probably needed to work it's magic), in order to be installed. (54549807–2019)

Tell Me About It.

Conceptual questions around minimising lifetime of data, privacy implications of services (e.g., Google visualisation tools, Google Drive, tracking and cookies, anonymisation). 'Linking to Google PlusOne, without embedding the button (for privacy reasons) It seems that Google only offers code to embed the +1 button. However, there are heavy privacy concerns (plus quite some load time) associated with it' (9248204–2013).

3.6.7. Developers Ignoring PbD Principles

SO users ask questions about workarounds to gain access to data protected by permissions or platform protections. They also have fundamental questions about the reasons for implementing privacy-preserving solutions. They look for access to resources such as: data belonging to other apps, WiFi, Bluetooth, device settings, unique device ID, scores in games, internet and camera permissions, make/model/serial number of computers, screenshots and videos, locations, IP addresses, names, and email address.

SO users ask the community about whether there is a need to do a task with privacy in mind or they can do it without needing privacy permissions. 'How should an app communicate with a server operated by its developer without android.permission.INTERNET? Or is there a reliable source stating that this is impossible in Android?' (29545251–2015).

Some questions looked for instructions on how to collect data, access restricted resources without following proper steps, store sensitive data, combine data from multiple sources, enable cookies, bypass permissions, and identify users.

How can I read the users computer make, model and serial number from inside MS Edge browser? Using Microsoft Edge web browser, under windows 10, how can I access the make/model and serial number of the computer that the browser is running on? (43492726–2017)

3.6.8. Developers as End-Users

Users also ask questions about how to protect the privacy of their own data, software, and identity.

How Do I Protect My Data?

This theme includes questions around implementing a solution or finding a better approach to protect their own data or intellectual property. ‘Whenever I start the program a little eye icon appears in the upper right corner above the scroll bar. It can’t be clicked. I assume it’s Google uploading my usage data. How can I disable that?’ (19327361–2013).

Privacy in Version Control Systems (Git)

SO users want to protect their source code and identity in version control systems. They also look for suggestions about how to provide access control to projects in these systems. ‘Is it possible to completely remove an issue from the GitHub issue tracker?’ (3081521–2019).

What files/folders should I ignore in a git repository of an iOS app? ... Do the files generated by cocoapods contain some of my private information? Does info.plist file contain my private stuff as well? Also, when I was putting Firebase into my app, I downloaded a GoogleService-Info.plist. Should I ignore it as well? What things should I ignore? (37479924–2016)

I Have Privacy Concerns, Thoughts?

Questions around personal privacy concerns are grouped here. For personal reasons, SO users look for suggestions to protect their own data in the workspace or from other software companies.

I recently purchased an advanced chat script which includes free installation on my server. I don’t know how to install it but the company says they provide installation if I provide them with the following information: [list of resources to provide access] I don’t feel comfortable giving all that info out to them but I know it’s required for them to integrate the script to work with my online forum. (4973811–2011)

3.7. Discussion

We are not the first to explore how developers think about and interact with privacy concepts. In particular, Hadar et al. conducted a set of interviews with developers with the aim of understanding their thinking and attitudes around privacy [141]. Similar to our findings, they find that developers often conflate the word ‘privacy’ with security concepts. For example, equating permissions

with privacy even though they are technically an access control topic, and have applications beyond privacy. From our own work, we see the conflation of privacy with security potentially coming from the phrasing on platform websites, such as calling permissions ‘privacy permissions’. Developers then learn to equate permissions with the term ‘privacy’. Also, when speaking of privacy, SO users employ language similar to the language of developers in other contexts: encryption, access control, data collection, data removal, data lifetime, and anonymisation are all recurring themes both in our data and in findings from Hadal et al.’s interviews with developers [141].

Of our random question sample (SO-privacy-rand), 17% were conceptual, indicating that developers are looking for advice around privacy-related tasks in the early stages of software development. Such decision-making questions can impact the privacy as well as the security of software: ‘Security defines which privacy choices can be implemented’ [47, p. 669].

3.7.1. Supporting Privacy Policy Creation Tasks

While there is research on making PPs understandable for end-users [150, 291, 319], there is minimal research on helping developers craft PPs. The lack of support can be seen in the wild, where there are still numerous apps without PPs [388] as well PPs that contain misleading and contradictory statements [27]. In our data, many questions ask for help creating privacy policies. Based on our observations, we hypothesise that some of the problems observed in the wild might be coming from developers who: 1) do not know that they need a PP, 2) do not see a reason for adding a PP, 3) do not know what language needs to be in a PP for their app’s unique profile, 4) are trying to add a PP but cannot do it because of complicated procedures as well as unhelpful user interfaces, and 5) see PPs as a wall that is blocking their app being published, with the resulting frustration leading to reluctance to prepare a well written PP.

Developers are sometimes confused about why a PP is needed because they honestly believe that they are not collecting any sensitive data. The developer’s understanding of ‘sensitive’ sometimes differed from the platform’s definition. Advertising and tracking libraries were another common cause of confusion. Developers were not using sensitive permissions directly, but had included an advertising library which was using some. When they tried publishing their app on an app store, they got a warning that a PP was needed due to sensitive permission usage. They then turned to SO to understand the cause of the issue. Similarly, third-party APIs have privacy implications that need to be reflected in the developer’s own PP, with some users turning to SO to figure out exactly what they needed to add to their PP because they used, for example, Firebase which is an app-building infrastructure. The above scenarios exhibit three important

themes: 1) the role of platforms in defining what ‘sensitive’ is, 2) the awareness developers have around the types of data their apps collect, and 3) the implications of third-party code and services on PPs.

Writing a PP is a challenging task for developers, especially if they are freelancers or part of small companies with limited legal resources. There is much potential for providing more support to them in this space, particularly automated support which can identify third-party libraries and services in their code and walk them through setting up a PP that correctly describes how data will be collected, stored, and used. This support is particularly needed when the privacy implications of using a service are not immediately obvious. For example, uploading images to Google’s image search to find similar images may cause Google to retain and index the uploaded image as Google puts in its PP: ‘When you upload, submit, store, send or receive content to or through our Services, you give Google (and those we work with) a worldwide license to use, host, store, reproduce . . .’ [132]. Another advantage of automated support is the capability to automatically detect and adapt to changes that occur when third-party PPs change, such as library version updates.

Another possible solution is to better integrate privacy checking into the code development process so that developers can address issues early instead of being rejected when publishing their app or receiving legal complaints after it has been published. Both situations frustrate developers, who feel that they are ‘done’ only to find that they have not yet fulfilled legal obligation. For example, one common cause of app rejection on the Apple Store is ‘Requesting Permission’ without suitable disclosure to the user about permission usage [30].

There are some tools to help with early identification of potential privacy issues due to permission usage. For example, Coconut is an IDE plugin that warns developers during coding when they are dealing with privacy-related tasks such as dealing with user location [185], allowing developers to make any necessary changes earlier in the development process. Such tools could be improved by supporting changes that occur between versions of third-party code. Otherwise, if Apple decides that a new permission is needed to access a specific resource, the developer might only discover the change through experimentation or user complaints.

3.7.2. Platforms as Privacy Drivers

Platforms such as Apple and Android exert powerful influence on privacy ecosystem [136]. They define the meaning of sensitive content, data, and resource such as camera, contacts, and location [8, 98]. One of the main reasons to ask questions about privacy on SO is rooted in platform requirements. SO users see platforms as gatekeepers for publishing their apps, and perhaps their income source. This

gatekeeper role gives platforms the power to enforce privacy behaviour in the applications they host. While some percentage will always try and circumvent, we found that the majority of SO developers were honestly trying to follow the requirements set by platforms.

Platforms also operate as an intermediary between the developer and user on privacy issues. For example, iOS decides when to ask the user about a permission usage and also controls the design of the permission UI the user sees. On one hand, this intermediary role removes a great deal of responsibility from the developer and gives users a more consistent experience. On the other hand, developers lose the ability to control the full experience of their apps. They also have hand-off related problems when the user is taken to a privacy-related platform screen and then somehow must seamlessly return to the app, even if they have just denied vital permissions.

3.7.3. Shadow Documentation

While platforms often provide guidance and documentation around their APIs, libraries, and services, developers still need more specific or targeted guidance. One role SO fills is to provide this documentation through community sourcing answers to specific questions. It is effectively producing a shadow version of official documentation, in a form similar to a Q&A. Parnin et al. studied the Android API documentation and found that 87% of class documentation is also covered on SO [258], making SO a near-complete replacement for consulting official Android documentation. We see similar behaviour with privacy posts: the answers not only include official documentation, but also provide documentation-like information that does not appear elsewhere. Examples include guidance around how to write a PP or how to interpret permissions in relation to existing company guidelines. In effect, SO also hosts community-generated developer-friendly shadow documentation of company policies, PPs, laws, and regulations. On SO, legal-jargon heavy 'documentation' is translated into case-specific guidance phrased in a developer-friendly way.

3.7.4. Topic Modelling

Topic modelling, which formed the core of our automated analysis, proved an effective way to analyse the entirety of our corpus without the expense or time investment of human annotation. It confirmed impressions of categories from the qualitative annotations, while also pointing to more granular categories and related patterns around language use.

Our manual qualitative analysis and our LDA produced similar high-level results. For example, privacy policies, permission settings, browser errors, and privacy

in code repositories come up in both methods. While the obvious difference is scale and time required, there were less-obvious interesting differences such as LDA's natural focus on company names (i.e. Google, Facebook) where the manual coding abstracted these to 'platforms'. Overall, LDA found topics that are relevant and interesting, with more granularity than the qualitative topics we identified. However, many details of interest were not evident from LDA topics. LDA is a bag-of-words model, meaning that it lacks syntax and semantics in its resulting topics. Consequently the model cannot differentiate issues like if the asker was trying to preserve privacy or intentionally circumvent protections to collect protected data.

Though most of our focus was on qualitative findings, LDA suggested potential avenues of future research. We observed differences across topics in the use of nontechnical vocabulary, like 'want', which suggested that some topics are goal-oriented while others are more curiosity-driven and abstract. We also observed differences in the use of polite words like 'thanks', which relate to linguistic register, or formality. Differences in register highlight how different topics draw users of disparate technical backgrounds or who project different personas through their language use.

3.8. Limitations

Not all SO users are native English speakers; therefore, we may have misinterpreted some questions because of language issues. Furthermore, we collected questions from SO's full history, hence, some questions may be outdated, though we generally found that while technology aged, the high-level problems remained relevant. SO askers are only occasionally explicit about their driver for posting. Drivers such as compiler errors, or platform requirements are clear from text, but motivation-style drivers like personal or client are very challenging to differentiate cleanly. While the difference would be interesting, we cannot provide it with high confidence.

When starting our qualitative analysis process, we reviewed three privacy frameworks [157, 308, 314] to create a group grounding for the term 'privacy'. While we ultimately decided to use SO users' own definition of the word, this early review may have impacted our analysis.

3.9. Future Work

Prior work has shown that traces of SO code snippets are visible in the apps that people use. Potential future work might look at traces of SO's answers in app PPs. A further step in understanding privacy on SO is to explore answers

and questions together to understand the dynamics between users and how they build knowledge around privacy-related topics. Developers in small companies who integrate ad networks for monetisation view advertisement companies as being responsible for user privacy [210]; our work points to similar questions about how developers view app stores and themselves in relation to users' privacy. Experiments with LDA point to distinct nontechnical language use in different topics; future work could look at politeness, formality, and other aspects of persona associated with different privacy topics, possibly investigating what questions are asked by different communities or skill levels of programmers.

3.10. Conclusion

We analysed privacy-related questions on SO with LDA and qualitative analysis. Our results show that SO users face challenges while writing and modifying privacy policies; working with or designing systems with access control; dealing with updates to platforms and APIs; and deciding on privacy aspects of their projects. Platforms can use these results to improve the privacy-related workflows to create an experience that is efficient and convenient. Google, Apple, and Facebook are privacy influencers who define what content is considered sensitive, and are major drivers that bring developers to SO to ask privacy questions. Any of these entities have the ability to impact how developers think about and interact with privacy and impact the privacy ecosystem of software.

Acknowledgements

We thank everyone associated with the TULiPS Lab at the University of Edinburgh for helpful discussions and feedback. We also thank the anonymous reviewers whose comments helped improve the paper greatly. This work was sponsored in part by Microsoft Research through its PhD Scholarship Program and a Google Research Award.

Back Cover

Looking at the privacy-related questions, I find that platforms are a major driver of privacy ecosystem in software systems. If Apple and Google introduce a new privacy feature or requirement, developers start to ask questions about it on Stack Overflow and try to satisfy the requirement. In Chapter 6 and Chapter 7, I follow this direction and study how ad networks, as an example of a software development platform with privacy consequences for users, present privacy-related to developers and how choice framing and wording of the options given to developers can impact their decisions about users' privacy.



4. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges

Front Cover

Prior work identified in the literature review shows security champions influence and help developers in doing security tasks. Borrowing this concept from the security domain, this chapter looks at how developers get help and assistance in doing privacy tasks from their colleagues who are passionate about privacy: privacy champions. This chapter also shows that bottom up approaches to integrating privacy into software teams can be an effective way of privacy promotion.

To understand privacy champions' motivations, challenges, and strategies for protecting end-user privacy, my co-authors and I conducted twelve interviews with privacy champions in software development teams. We find that common barriers to implementing privacy in software design include: negative privacy culture, internal prioritisation tensions, limited tool support, unclear evaluation metrics, and technical complexity. To promote privacy, privacy champions regularly use informal discussions, management support, communication among stakeholders, and documentation and guidelines. They perceive code reviews and practical training as more instructive than general privacy awareness and on-boarding training. We observed that privacy champions do exist in software teams, and while security champions are more acknowledged in the literature, our work contributes to the field of developer-centred privacy by bringing this valuable type of developers into the surface and acknowledging their efforts as well.

4.1. Introduction

With the rise of technologies that collect data about every moment of peoples' lives, user data has become the economy's new oil [290] making it valuable for businesses but potentially privacy-harmful for consumers. Regulations, consumer education, and privacy-preserving technologies are often seen as the main strategies for addressing online privacy issues. However, regulations are by nature less agile than businesses, especially in a highly innovative field like technology. Regional differences in laws also make it hard to reconcile the privacy protection questions that spill across the borders of a single state or country. The effectiveness of consumer education is limited by users' bounded rationality and other human factors, such as memory, attention, and beliefs [11]. The lack of transparency about data flows in user interfaces further diminishes users' ability to make informed privacy choices [99, 134]. Oftentimes the only choice available to the users is to avoid or limit using the technologies altogether, as many systems do not offer usable and effective privacy-preserving options, resulting in 'learned helplessness' among the users [103, 122]. Therefore, privacy-preserving technologies and product features are one of the most immediate and effective solutions for supporting consumer online privacy.

Software developers play a central role in the data economy. Software development teams can decide which libraries, tools, and platforms to use, what data to collect, and how to present information to users, which means that their choices directly impact user privacy [163]. Prior work has suggested that the success of implementing privacy engineering in organisations predominantly depends on the organisational culture around user privacy in software development and product design teams [38, 141, 357]. Therefore, it is important to promote privacy-preserving principles, such as 'Privacy by Design' [78], which aim at including privacy considerations into design and development processes from the early stages [157]. Yet, shifting organisational culture is not a trivial task. While organisations increasingly recognise *security* values and try to improve security posture [89, 336], there are still few examples and little guidance on how to build *privacy* culture in the organisations. However, lessons from prior successes of building organisational culture around security might be useful.

One promising approach for inducing organisational change is to promote ideas through enthusiastic early adopters willing to put ideas into practice. Such enthusiasts who have a special interest, and often expertise, in a subject are called 'Innovation Champions' (or 'advocates'). They encourage others and aid with overcoming barriers that a new idea could face [277]. This approach has been explored in software teams with Security Champions [146, 326, 337]. Security Champions play an intermediary role to facilitate conversations between security and development teams [335].

Our study leverages the literature on Innovation Champions and Security Champions, to study the role and experiences of *Privacy Champions* in software teams. We believe that we can learn from these people about effective and ineffective strategies and communication channels they use to promote privacy values on the ground. This information and empirical evidence contributes to understanding best practices and forming recommendations for promoting privacy values in software teams and product design.

We conducted 12 semi-structured interviews with Privacy Champions who are part of software teams to understand their motivations, challenges, strategies, and communication channels for promoting user privacy within their teams and organisations. Our results suggest that negative privacy culture and attitudes, tensions between privacy and business priorities, lack of standardisation, evaluation metrics and automated privacy tools, and technical complexity are common barriers for implementing privacy in software design. Most Privacy Champions agree that regular privacy-focused meetings, informal discussions, management support, facilitation of communication among stakeholders (e.g., between legal and product teams), appropriate privacy documentation and guidelines are particularly useful in promoting user privacy, while shaming or punishing the developers for not implementing privacy features are ineffective. Privacy Champions' experience demonstrates that incorporating privacy considerations into design reviews has a bigger impact on the end-user privacy in the final decisions and products and yields better educational effects on developers, than company-wide awareness programs or on-boarding privacy training for new hires. We conclude that similar to Security Champions' programs aimed at facilitating security practices [146, 326, 337], Privacy Champions' efforts, when supported by management and a critical mass of other developers, may be effective in promoting organisational privacy culture, and implementing Privacy by Design principles.

4.2. Related Work

Privacy regulations, such as the General Data Protection Regulation (GDPR) [127] in the EU and California Consumer Privacy Act (CCPA) [72] in the US, have forced companies to modify their services and products to comply with them [51, 84, 114, 184]. Those privacy regulations introduce such concepts as the 'right to be forgotten', the 'right to data portability', and the 'right to restriction of processing', with the implementation of them left to developers. Such frameworks as 'Privacy by Design' [78] are intended to bridge regulations with technical implementations. Yet, there are still gaps in developers' understanding of privacy and privacy frameworks [141, 357]. For example, developers' opinions about privacy are limited by security vocabulary and compliance requirements, and privacy is rarely considered in the design process [357]. In addition to regulations,

developers are also having to contend with requirements set by software platforms like the Android App store, these platforms require even independent developers to engage in privacy-related activities like writing privacy policies, declaring permission usage, and getting consent from users [330].

4.2.1. Security Champions

One way to support company innovations in general, and privacy innovations specifically, is to have a ‘champion’ who advocates for these innovations and is willing to promote it actively [277]. ‘Where radical innovation is concerned, the emergence of a champion is required . . . the new idea either finds a champion or dies’ [294, p. 8]. Prior research acknowledges the role of champions in software teams for promoting the use of software technologies such as Java generics [256, 257], usability [223], and security practices [144–146].

Security Champions in development teams have an interest in security but they are not necessarily security experts or have a formal Security Champion title [146, 282, 326, 335, 337, 365]. They can positively influence the security practices of others [90–92] often with a bottom-up approach instead of a top-down approach [56, 90, 264]. Such behaviours and attitudes are valuable in organisations that prioritise security [149]. Peer developers view Security Champions as essential players in software security [335, 363, 364]. They can be an experienced hacker who helps testers in finding vulnerabilities [353], an intermediary between the security and development teams [121, 335], or the leader in threat modelling activities [60, 307]. They are involved in several security-related activities such as educating other developers [144, 146, 147, 159, 351], increasing awareness [90, 146, 147], and promoting the adoption of technologies [144, 145, 147].

Security Champions’ motivations are primarily internal (e.g., sense of duty and evidence of impact), but also external (rewards and punishments) [125, 147]. Broadly, Security Champions are hierarchists who follows the security policies [56, 59], have personality traits such as good imagination, altruism, morality and openness to experience [125] with good communication and soft skills [144], understand the balance between security and business processes [59], and have a thorough understanding of risks associated with actions and outcomes [59].

Our study builds on the importance of having a champion for new ideas and innovations in companies to make them successful. We explore how *Privacy Champions* in software teams promote privacy, what motivates them, what strategies they use, and what feedback they receive while playing this role.

4.3. Method

We conducted 12 semi-structured interviews with members of software development teams who actively promote user privacy in their teams and organisations, who we refer to as ‘Privacy Champions’. Our interview script was designed to address the following research questions:

RQ1: What Privacy Champions find motivating, rewarding, challenging, and frustrating in promoting user privacy in their organisations

RQ2: What strategies and channels do they find least and most effective in achieving their goals

RQ3: What resources do they use to keep up with the latest in privacy

The study received approvals from the ethics boards of the authors’ respective institutions. All participants provided informed consent to participate in the study and be audio recorded.

4.3.1. Recruitment

Prior research identified innovation and Security Champions using such methods as screening surveys [277], and nomination by peers [153], executives [168], and self-nomination [146, 147]. We believe that the role of successful Privacy Champions need to be recognised by their community, not only by themselves. Hence, we used the referral and snowballing techniques for recruiting participants. In our recruitment messages, we asked the recipients to nominate someone in their organisation or network, who can be described in at least one of the following ways: (1) they formally or informally promote best practices for users’ privacy, educate others, persuade, and advocate for privacy adoption throughout the software development process, and (2) they have an official or unofficial role within their team acting as the ‘voice’ of users’ privacy for the product or team, for example by giving privacy-related advice that can influence decisions and privacy practices.

We posted the recruitment messages on Twitter, and in security, privacy, and software development-related LinkedIn, Reddit, and Slack groups, mailing lists, and public fora. We also sent direct messages to LinkedIn users with privacy and security-related titles, and reached out to the employees of software companies in our personal networks. The interviewer did not personally know any of the participants, and the resulting sample is diverse in terms of participants’ characteristics and background.

We included in the message a link to a short screening survey and our contact details for questions. Based on the screening survey results, we sent the eligible candidates a link to the interview booking system, where they could select the

date and time for a 30-60 minute interview. We thanked survey respondents who did not meet our selection criteria for their interest in our research and asked them, and selected interview respondents, to share information about our study with other potential candidates.

4.3.2. Procedures

Screening Survey

After reading the consent form and providing consent to participate, respondents answered questions about demographics, employment status, job title and role, industry sector they work in, and language proficiency (see Appendix A.1). We excluded participants who were students or were not working in software teams, and invited the eligible participants for an interview.

Interview

Before starting the interview, we first read aloud the consent form's key information, as a reminder. We started the audio recording and the interview upon receiving participants' verbal consent. Due to the similarities in research goals, our questions were partially inspired by an interview study with Security Champions [146, 147]. We asked participants about definition of privacy in their work context, motivations, frustrating and rewarding aspects, strategies and communication channels and their (in)effectiveness, feedback they receive from others, and resources to keep up with the latest in privacy (Appendix A.2 includes the interview script). After conducting two pilot interviews with Privacy Champions from our personal networks to validate the interview script and timing (not included in our analysis), we slightly modified the script to improve clarity. All the interviews were conducted virtually using participant's preferred online calling service, audio-recorded, and transcribed by professional GDPR-compliant services.

4.3.3. Analysis

Two of the authors independently built initial codebooks based on two interviews, while continuing the recruitment. Then they merged the initial codebooks, discussing and resolving disagreements and differences. After applying the merged codebook to two additional interviews, they added and merged some of the codes to reach a comprehensive structure. After applying the modified (final) codebook to the rest of the available interviews, they found that all raised themes fit within the codebook structure, suggesting that saturation was reached. Thus, they stopped the recruitment, and, using the final codebook, re-coded the interviews used for

the initial codebook development and validation (see Appendix A.3 for the final codebook). All themes were mentioned by multiple participants, signalling that they are recurring. All interviews were coded by both researchers resulting in an inter-rater agreement rate of 55% (calculated as Brennan and Prediger Kappa [69]). Most disagreements were related to minor differences in coding policy (e.g., applying high-level codes to an excerpt that contains multiple lower-level codes) and due to similarities between the related groups of codes (e.g., ‘conversations and discussions’ can be a strategy and a communication channel, but the coders might have applied just one of the two codes). The researchers together discussed, resolved disagreements, and re-coded the excerpts for the groups of codes with the Kappa below 60%. They achieved the final agreement rate of 75% (with the agreement rate on individual groups of codes between 60% and 100%), which is considered satisfactory [176, 179]. The quantified insights in the results section are based on this final analysis.¹ These numbers are reported to show the frequency of occurrences and should not be interpreted for generalisation purposes. We used MaxQDA software for qualitative analysis and calculation of the agreement rates.

4.3.4. Limitations

While the variety of channels we used for recruitment resulted in sufficiently diverse sample, it does not represent the software industry and cannot be generalised to all companies and software teams. Our study was limited by the availability of participants, which are hard to recruit and incentivise, given their busy schedules and high incomes, in comparison to previously-used broader populations that included people advocating to managers and end-users [146, 147]. We suspect that finding a Privacy Champion is particularly challenging because it is not a well-defined role, usually informal, and many developers blend the concepts of privacy and security [141, 261, 357]. Moreover, our study was conducted during the COVID-19 pandemic when most businesses were closed or working remotely; hence, minimising the chances of in-person networking and recruitment in workshops, meetings, and conferences.

We made a particular effort in increasing gender diversity by posting in 18 LinkedIn groups, Slack channels, and forums specifically directed at women in tech, and encouraged participation of women, and representatives of gender and ethnic minorities in other channels. Despite our efforts, the sample is still male-dominated, which is in line with Stack Overflow’s 2020 Developer Survey [96].

Although prospective participants working in the big tech companies acknowledged that non-disclosure agreements prohibited them from discussing the details

¹In evaluating the inter-rater agreement, we did not consider the codes on which we did not intend to report quantified results (i.e., warm-up questions Q1 and Q2 about participants’ jobs and beliefs about why they were nominated for the interview).

of their work, our research does not rely on obtaining such details, as our analysis focuses on higher-level patterns. Moreover, by nature, Privacy Champions may be privacy protective, concerned about sharing contact details, using the interview booking system, online calling services, and limit active participation in the social media and online forums. Our transparency in the recruitment materials, consent mechanism, use of various recruitment strategies, and offering alternative choices was focused on mitigating those concerns.

4.4. Results

In this section, we describe our participants, their conceptualisations of privacy, motivations to be Privacy Champions, challenges, and the strategies and resources that they use.

4.4.1. Participants

Recruitment was done during July and August 2020. We received 29 complete responses to the screening survey, which on average took 5 minutes ($SD = 3$ minutes) to complete, excluding one participant who completed the survey in over 24 hours. We screened out 7 respondents because they were students or were not working in software teams. We reached out to 22 eligible candidates, of which 14 participants signed up for an interview, one later cancelled, and one did not show up and did not reschedule. In total, we conducted 12 interviews, which on average took 36 minutes each ($SD = 10$ minutes). Participants received a 20 USD (or equivalent in their local currency) gift card for their time.

While participants hold different job titles, they all work full-time in software teams and interact with other developers, and are proficient or fluent in English. They are employed in the business sector except for one from the non-profit sector (P9). Six are employed in North America, five in Europe, and one in Asia. On average, they have 10 years of experience ($SD = 6$ years), and work in a team size of 10 ($SD = 13$ members). Nine participants identify as male, two as female, and one preferred not to identify their gender. The average age is 33 years old ($SD = 7$ years). Eight participants hold an official title or a role related to privacy or security, and one (P8) holds an informal Security Champion role. P11 previously worked as a privacy architect working with developers, and most of our conversation with him was about his previous role. Table 4.1 shows a summary of participants' demographics.

During the recruitment we received a number of interesting informal comments from the people who saw the recruitment message. First, they acknowledged that it would be easier for them to nominate a Security Champion than a Privacy

Table 4.1.: Summary of participants' demographics.

ID	Role	Job Title	Sector	Current Continent	Gender	Number of Employees	Team Members	Years of Experience	Age
P1	Privacy and/or security eng.	Sr. Security Engineer	Business	North America	M	1,000-9,999	13	9	30-34
P2	Privacy and/or security eng.	Contractor Cryptographer	Business	Europe	M	100-999	4	8	25-29
P3	Software development	R&D Software Engineer	Business	Europe	M	+10,000	6	15	35-39
P4	Privacy and/or security eng.	Privacy Officer	Business	Asia	M	+10,000	50	2	18-24
P5	Privacy and/or security eng.	Head of R&D	Business	Europe	M	1-9	5	10	25-29
P6	Privacy and/or security eng.	Sr. Product Security Engineer	Business	North America	M	10-99	4	15	35-39
P7	Managing software develop.	Sr. Manager Research Engineer	Business	North America	NA	1,000-9,999	6	10	35-39
P8	Software development	Software Engineer	Business	Europe	F	1,000-9,999	6	2.5	25-29
P9	Privacy and/or security eng.	Research Engineer	Non-profit	North America	F	1,000-9,999	3	5	25-29
P10	Research: new features	Sr. Privacy Researcher	Business	North America	M	100-999	10	14	35-39
P11	Research: telecom security	Technical Staff	Business	Europe	M	+10,000	8	25	45-49
P12	Software development	Software Engineer	Business	North America	M	100-999	4	7	30-34

Champion, suggesting that the latter role is not yet as well defined or common as the former one. Second, they often asked if a privacy officer or another privacy expert from a legal department would qualify for the study, as those are the only people who directly address privacy issues in their company, to the best of their knowledge. Moreover, the official titles of most of our participants are primarily related to security, while their actual formal responsibilities and informal activities often include privacy as well. These observations align with the insights from the interviews regarding the overlap of privacy considerations with security engineering and legal perspectives on data protection (see more details in Section 4.4.2).

When we asked participants why they believe their colleagues nominated them for the interview, they attributed it to either formal responsibilities (such as being a member of a special interest group focused on privacy, or being a point of contact for user data protection), or informal aspects of their advocacy (e.g., being vocal about privacy, and having a reputation of privacy enthusiast).

4.4.2. Privacy Conceptualisations

We asked participants to define the term 'privacy' as they normally use it in their work context, and describe what are the differences between privacy and security. In line with privacy literature [236, 255, 308], the majority of Privacy Champions (7/12) acknowledged that privacy is a broad, complex, and contextual term: 'Privacy is really hard to define, because it's so contextual' (P9).

Privacy as Data Protection

Almost all (11/12) Privacy Champions, in the context of their work, refer to privacy as protection of personal data from unauthorised access: 'Privacy really means . . . that we're going to do our utmost not to leak their [users'] data, that we're going to protect their data and that we're going to do our best to secure it' (P9). Among data

protection techniques and approaches, participants mentioned: anonymisation (6/12), data minimisation (5/12), encryption (4/12), differential privacy (3/12), and Privacy by Design (1/12). We discuss participants' opinions about the relative effectiveness of these approaches in Section 4.4.5.

Privacy as Transparency and Trust

Privacy Champions (9/12) also referred to privacy as ensuring transparency about data practices and respecting users' trust, by meeting their expectations, and respecting their preferences. Less often, they referred to privacy policies as an instrument for ensuring transparency. Some even openly criticised using legal documents for communicating privacy information to the users: 'These ridiculous legal terms, terms of service pages that continue to get more lengthily and more complex and smaller font and basically aren't able to provide humans with an intuition of what's exactly happening' (P7).

Privacy as Data Management and Control

Many participants (8/12) conceptualise privacy as users' ability to manage and control their personal data, for example, through consent mechanisms:

Privacy . . . means that I as a user can give my consent to someone to process my data in a controlled manner . . . and if at any point I wish to be forgotten, I should have this right preserved, and that should be mandatory. (P5)

Privacy as Legal Compliance

Some participants (7/12) mentioned legal compliance, but few rely on it as the primary working concept: 'You need to make sure that the data you store complies with regulations and the intent that the user supplies the data with' (P2).

Privacy as Human Right and Ethical Value

Several participants (5/12) acknowledge a broader, non-technical, view on privacy as a fundamental human right and ethical value, enabling personal freedom: 'I think privacy is important for freedom, democracy' (P6). While not necessarily used as a working concept in their daily job, as we discuss in Section 4.4.3, this conceptualisation is a common driver for Privacy Champions to advocate for privacy in their work.

Comparisons Between Privacy and Security

Many participants recognised the close relationship between privacy and security, to the point where a few mixed the two terms or found the boundaries between them blended or 'blurry' (P4). Many participants (8/12) saw the reinforcing relationship between these concepts, whereas security enables privacy: 'I think privacy is a subset of security' (P10). Others (7/12) viewed privacy as a broader concept where 'privacy goes further than security' (P6).

However, two participants acknowledged potential tensions and contradictions between privacy and security: 'Even though security and privacy often get lumped together in terms of the technical underpinnings of what is required to achieve these systems they can often be at odds in terms of how to accomplish them' (P7).

Some participants (5/12) mentioned that security values are more widely recognised than privacy values, and that security is a more mature field with more defined terms, taxonomies, metrics, and established best practices, which may create a useful benchmark for privacy: 'With privacy, it feels a lot more abstract, when you're trying to argue for it' (P2).

P1 emphasised the value of differentiating between the user-focused privacy roles (e.g., usable privacy researchers or ethics experts) and technical security roles and having 'someone whose job is explicitly to be the privacy advocate for the users, whose job is not to know what cryptography is . . . who has a little bit more of that social scientist in them' (P1).

While the official job titles of the majority of our participants are shaped around security, their conceptualisations of privacy are not limited to security concepts, as it is typical among software developers [141]. Broad understanding of privacy reassures the Privacy Champions' potential in promoting privacy values in their organisations beyond the common security and legal frameworks.

Socio-Cultural Differences in Approaches to Privacy

Three Privacy Champions acknowledged country-level differences in privacy cultures. P12 believes that people in Europe are more concerned about privacy than people in the US and 'that privacy is much more of a first-class concern there than here' (P12).

Moreover, P1 highlights the socio-political differences between the US and Europe, which lead to diversity in their approaches to addressing privacy issues, and recommend a more unified approach that brings together the perspectives of different stakeholders:

4. *Privacy Champions in Software Teams*

America has been very American about it and said, . . . ‘Let’s let the corporations solve the problem for us’. Europe is very European about it and says, . . . ‘Let’s have the government just solve the problem for us’. Frankly what we need is a much more multi-stakeholder conversation. (P1)

Similarly, findings from Bamberger and Mulligan show that US privacy is based on ‘risk management to avoid harm to consumer expectations’ and the European privacy culture is formed ‘as an individual human right and eschewed the language of risk and consumer’ [49, p.12].

4.4.3. Motivations

We asked Privacy Champions about what motivates them to promote privacy, what they find rewarding in that process, and what positive feedback they receive from their colleagues. We found that participants are driven by both personal and organisational motivators. Prior work has seen similar trends that these two factors are complimentary and affect individual performance at work [23]. Motivation is important for Privacy Champions because one of their main tasks is motivating others [160, 277].

Personal Motivations

Most participants (10/12) mentioned personal motivations for promoting privacy in the organisation, such as strong personal privacy attitudes, human rights and societal benefits, and empathy towards users: ‘I always put myself in the other person’s shoes. I would not like my data to be tampered with’ (P4). Thus, Privacy Champions (6/12) find it rewarding to see the impact of their efforts on end-users and society.

Interestingly, a few people admitted that personal experience with privacy violation, or big media stories (e.g., Snowden revelations) inspired them to become Privacy Champions in their organisations: ‘The Snowden revelations came out and I felt extremely strongly that what he did was heroic and that I should figure out a way to support that kind of effort’ (P10).

Experiences and expertise gained during school and work projects also inspired some of our participants (3/12), and contributed to the perceived sense of personal responsibility (6/12) for building products and services that protect user privacy:

It’s not like one day I woke up and said, ‘I want to be a champion of privacy’. It’s just that my project required me to use this data . . . I saw how important it is to keep this data safe and so I tell everyone else . . . how they should also handle this type of data. (P3)

Finally, some Privacy Champions enjoy solving technically challenging tasks, and find it rewarding when they discover privacy-preserving solutions for real-world problems (3/12):

It can be a bit of a fight sometimes to get people to . . . go through the pain of adding this extra . . . [privacy-preserving] feature . . . but it's very satisfying to come out of this with something that is much better than the way that the average company does it. (P6)

Organisational Motivations

Organisational motivations (8/12) also drive Privacy Champions' work in promoting user privacy. Participants see the respect of user privacy as a competitive advantage or even existential requirement for a company that wants to have a successful software product on the market: 'If we are perceived as an organisation that doesn't care about user privacy, then that will harm us. If we are perceived as an organisation that does care, that will benefit us' (P1). It highlights the value of privacy as a central attribute of the company brand and corporate identity. Addressing privacy issues is especially important to the success of the companies working on emerging technologies, due to potential lack of users' familiarity with and trust in such technologies and their data practices: 'We are in emerging technology . . . so there's this business understanding that we will freak people out, and we will ruin our business, if we don't respect people's privacy' (P9).

Strong corporate privacy culture attracts people with positive privacy attitudes, and offers an opportunity to align the professional goals with personal values. 'I have developed my professional trajectory in order to create opportunities to work on things that matter . . . the promotion and development of privacy-enhancing technologies . . . is very much aligned with the goals of the organisation' (P7).

Privacy Champions (5/12) find it encouraging and rewarding also when they notice an improvement in company' privacy culture and values: 'The awareness I create through this process, that's the most rewarding thing' (P5).

Recognition by peers and managers, their requests for advice, further encourage Privacy Champions: 'The most implicit form of a reward system is from leadership, who aren't usually bothered by these small things, when they come down to your level and are like, 'We have a problem, and we need help with so-and-so problem'' (P4).

In contrast, weak privacy culture not only inhibits their enthusiasm but may also turn Privacy Champions away from the company entirely: 'I actually left a previous job because I disagreed with the privacy aspect of the project I was asked to work on' (P12).

Only one participant mentioned tangible incentives contributing to their motivation to promote privacy. Most of the participants are not advocating for privacy in exchange for rewards. However, while Privacy Champions find positive feedback, and recognition of the value of their work intrinsically rewarding, they also appreciate more formal rewards, such as career promotions or additional compensation (2/12): ‘It’s not part of my job, so when it comes to career advancement, getting recognition, getting compensation, there are some shortcomings’ (P12).

4.4.4. Challenges

We asked Privacy Champions about challenges and frustrations in promoting privacy, instances when they felt their efforts were not appreciated, and negative feedback received from colleagues.

Indifferent or Negative Attitudes

Privacy Champions perceive mixed privacy attitudes from their teams and organization. In Section 4.4.3 we discussed how positive culture, attitudes, and feedback encourage Privacy Champions. Conversely, indifference, ‘I’ve got nothing to hide’ mentality [309], or even openly negative privacy attitudes, such as annoyance and push back from the team members, make it challenging for Privacy Champion (11/12) to advocate for privacy values: ‘‘I have nothing to hide,’ people are really difficult to deal with. When you run into people with that mindset, it can be very difficult to engage with them’ (P9). The indifference and unawareness of the privacy benefits among clients and users circles back and also negatively affects the attitudes of engineering teams: ‘When I would argue for privacy, I would get push back from people that was, ‘Users don’t care, nobody cares, why are you bothering me about this? I have a job to do, just let me get my job done’’ (P1).

However, some participants noted that engineers’ attitudes have been shifting to the positive direction over time, thanks to the changes in social norms, emergence of privacy regulations and requirements, and efforts of the Privacy Champions, which we discuss in more details in Section 4.4.5:

Right now, privacy’s become . . . the priority, before you move on to anything else. People have started to act upon it faster . . . because they understand the impact of not handling data privacy in the right way.
(P4)

Tensions Between Priorities

Engineers' push back is related to the tension between privacy features and other, technical or business, priorities (9/12), such as primary technical features and performance, or additional time, efforts, and financial resources it takes to address privacy, postponing deadlines, and increasing the costs:

'If you want to . . . have these techniques that retain privacy, usually this translates into a cost. That could be performance. That could be money. That could be user experience' (P5).

Lack of Standardisation and Evaluation Metrics

Privacy Champions agreed (8/12) that 'privacy is hard to measure', for two main reasons. First, privacy lacks standardised definitions and taxonomies: 'There's no national law or agreement on what privacy standards should be. There are things like the NIST [National Institute of Standards and Technology] privacy framework, but there's no consensus, it's not widely known, widely shared' (P12).

Second, there is a lack for metrics for evaluating privacy risk, harm, and penalties for violating privacy and metrics for evaluating the effectiveness of privacy protection approaches. The ambiguity of the existing frameworks leave engineering teams in uncertainty about the privacy status of their products and whether the deployed protective measures are adequate and sufficient:

What is, for example, the minimum anonymity set that we can have in our products? . . . Is it enough to put people in buckets of 3 people, or should we be looking at 100 people? . . . can we do it even if there's only 100 people in that particular country? Those are numbers that we've been asked to formalise . . . We haven't been able to do that yet. (P6)

Without being able to quantify the benefits and extent of improved privacy and costs of its violation, it is hard for Privacy Champions and engineering teams to advocate the business impact of privacy, or argue for the project timeline extension or budget increase necessary for addressing privacy concerns.

Additionally, there are practical challenges with standardisation of privacy due to high context-dependency, and heterogeneity of users' preferences and needs: 'GDPR, it was definitely trying to answer the question of what I hear is the right answer for all EU citizens, as if all EU citizens were exactly the same with the exact same desires for privacy' (P1).

Technical Complexity

Privacy Champions (6/12) mentioned that building privacy features is technically difficult: 'How can we enable applications like procreated rendering and other really important product directives . . . while still protecting privacy? That's been really difficult' (P9). Sometimes the technical complexity relates to the lack of knowledge in the development team: 'Typically we can identify a risk, but the developer may not be aware of privacy preserving techniques that might be used to mitigate that risk' (P6). However, more often it just translates into extra effort and time, creating the tensions described in Section 4.4.4.

The complexity can also arise from the fact that broad privacy-related goals and vague guidelines are difficult to translate into specific technical requirements and then practices, especially when they are: 'generated from the legal documents . . . They were all very, very fuzzy . . . There's very, very little of the how we should do things, how we should integrate this for the engineering processes' (P11).

Communication Issues Between Stakeholders

Ensuring privacy in a product requires involvement of various stakeholders, to consider the multitude of conflicting interests.

Given that developer, manager, and lawyer stakeholders come from different backgrounds, are members of separate teams, and hold various places in the corporate hierarchy, the communication between them can be challenging (5/12), due to discrepancies in terminology and conceptualisations. Similar to the difficulty of translating privacy goals into technical requirements (see Section 4.4.4), the conversation between developers and legal departments demanding compliance without taking into consideration technical limitations may be frustrating for both parties: 'Having this engineering background is very, very different to how the lawyers perceive the system . . . there was no understanding of the engineering process' (P11).

A female Privacy Champion, brought up a communication issues specifically associated with *gender biases*. She had to seek her manager's support to convey her ideas and prove herself as a female Privacy Champion and engineer to teammates: 'I can be overlooked in meetings sometimes. I think it is more because of my gender than anything else . . . I've had to Slack my opinions through my director, who has then raised them in meetings for me ' (P9). She emphasised the positive impact of gender diversity on the breadth of ideas and considerations of privacy implications: 'Sometimes men are like, 'Why would you need to protect a phone number more?' Women are like, 'Because sharing your phone number gets you harassed'. It does give you a different perspective on privacy' (P9).

This is in line with the literature suggesting that cybersecurity needs to be more inclusive and diverse [144, 222]. These observations highlight the importance of increasing gender diversity in privacy community specifically and tech companies in general, and the importance of management support in overcoming gender bias. However, delivering all the female employees' opinions through a team manager is not the most effective way of communication, and also not the most fair to the women who do not get credit for their ideas. Therefore, it is important that management encourages women to speak up and independently express their opinions in meetings and company's communication channels. This will increase the diversity of perspectives, and breadth of ideas, eventually leading to better privacy solutions.

4.4.5. Strategies

Privacy Champions mentioned a variety of strategies and techniques that help promote privacy in teams and organisations; these range from formal documentation and policies, and specific libraries and tools to informal 'water-cooler conversations' (P12).

In general, our participants emphasised the effectiveness of a 'collaborative tone' (P7) when promoting privacy values. On the other hand, participants' opinions about the effectiveness of enforcement of the policies regarding privacy are mixed. For instance, some Privacy Champions think that enforcing policies signals management's serious intentions about it, and makes developers recognise the importance of addressing privacy issues and put extra effort in it: 'These kinds of decisions need to be enforced by upper management . . . Developers always go for the easy solution, and having privacy in mind when dealing with users' data, unless it's enforced, it's just extra work' (P5). Others believe that without explanations of reasoning behind mandatory processes, those mandates do not reach the full potential and developers may treat the processes as a 'box-ticking exercise' (P7) and hence ineffective.

Improving Company Culture

All participants (12/12) acknowledged that improving company culture regarding privacy is essential in promoting end-user privacy values in software development teams. Privacy Champions suggest to encourage regular formal and informal discussions about privacy to not only shape individuals' mindsets or educate about certain practices but also to build the collective organisational privacy culture: 'It's less about individual features, but more about bending the arc of the organisation over time, to value privacy more highly, by simply engaging with it publicly a lot' (P12). Privacy Champions suggest encouraging in product teams

4. *Privacy Champions in Software Teams*

general empathy towards users' needs and expectations and sense of personal responsibility to make them 'feel that they both can be and should be thinking about the implications for the users' (P7) and reflect on 'What are the kinds of user harms that are occurring because we did privacy wrong in our product, and how can we design our product to be more privacy friendly' (P1).

In those discussions, to help justify additional costs, time or work load required by privacy engineering, Privacy Champions find it especially effective to emphasise risks and potential costs associated with not addressing privacy issues and also pointing out the benefits and competitive advantages of privacy-friendly products: 'Acknowledge and accept that it is extra work to do things in a privacy-conscious manner but it's worthwhile work. It is to the benefit of the company, . . . of the user, . . . of the society' (P12).

Privacy Champions also find management support important in promoting privacy culture, by talking about it in company wide speeches 'to inspire people' (P1), and explaining the value of privacy:

The CEO, chief legal officer and head of product all stand up and say 'Look, from a product perspective . . . from a legal perspective . . . from the perspective of doing the right thing for our users, this is super important'. (P1)

Facilitating communication between teams improves the overall culture of privacy in the company as well. Our participants recognised the benefits of forming special interest groups focused on privacy and integrating Privacy Champions into various teams, to have at least one or two privacy expert in every team and to help different stakeholders and teams understand each other's perspectives, terminology, requirements, and needs.

Integrating Privacy Champions into engineering teams helps to make the process of addressing privacy considerations and implementing Privacy by Design principles more straightforward, fast, and less bureaucratic, reducing the tensions between privacy and time (see Section 4.4.4): 'We'll work with your design, we'll point out places where it could be tightened up and so on, and we will reduce the amount of documentary evidence required in order to pass a privacy audit' (P11).

Communication Channels. Privacy Champions use various channels for promoting privacy values and organisational culture, including verbal (10/12) and written (4/12) communications, productivity and communication platforms (4/12), and special events (3/12).

One-on-one discussions and group meetings are the main verbal channels that Privacy Champions deploy for promoting privacy. Among written materials, while Privacy Champions find guidelines and documentation generally useful (see Section 4.4.5), P8 brought up an issue with keeping them updated and navigating

through them: ‘Searching content on Confluence [wiki] is quite hard, and most of the documentation is quite old . . . Or there’s a lot of archives documentation that when you search you can’t really find it’ (P8). P1 further warns about the trade-off between the informativeness of detailed documentation and educational materials and its poor fit for lifting motivation to implement privacy in product design: ‘Those detail-heavy classes and detail-heavy instruction material are very bad at inspiration but very good at education’ (P1). Personal or company blogs, and books were mentioned by a few participants as resources that can be shared with colleagues as a point of reference.

Among productivity platforms, Slack is commonly used by Privacy Champions to answer specific questions about privacy or communicate with peers about privacy less formally: ‘I keep an eye out for when people are talking about security and privacy things and will try to tactfully insert my opinions without steamrolling everything’ (P12). GitHub is used not only to discuss, but even to document identified privacy issues: ‘A GitHub issue. That’s where we do our security reviews. If you want to do security reviews, you raise that as a GitHub issue, and then we ask questions’ (P6).

Finally, special events like workshops, seminars, hackathons, and lightning talks provide additional opportunities to Privacy Champions to promote privacy values and share knowledge: ‘That’s how I share with the company what’s new, and what we’re doing to promote user privacy’ (P9).

Design and Code Reviews

Privacy Champions find a good opportunity to promote privacy values during design and code review process (10/12), prior and after development: ‘Much like many companies have security reviews early in product scoping sessions, data management and privacy reviews can go a long way’ (P7). These processes help to ‘block off’ time for privacy, and think through practical challenges and applied solutions: ‘When someone has to take in some feedback and then actually think through proposed mitigation and have a discussion around how we can change that mitigation to make it more workable. They’re actually deeply involved into the particular problem’ (P6).

Echoing the Privacy by Design philosophy, some participants believe that privacy reviews are more effective when conducted before development (at the requirements stage) than after: ‘Whether or not there are more privacy-preserving ways to build that feature. Those ways never get implemented after the fact, because at that point, the feature’s done and the team’s moved on to something else’ (P12).

Moreover, Privacy Champions suggest that open-ended questions are more helpful than compliance checklists or privacy impact assessment scales in triggering a more profound discussion about the privacy implications of a software product:

4. *Privacy Champions in Software Teams*

‘What user data goes through your service? What can you learn about the user from this? Very basic questions give a lot of the leverage’ (P1).

Some participants mentioned that it’s beneficial if everyone in the company, in addition to the developers and data protection experts, can engage in the reviews of requirements and specifications of the new features.

Documentation and Guidelines

Many Privacy Champions (8/12) believe that documentation and guidelines are helpful in promoting implementing privacy in product design and software development. Our participants frequently mentioned internal documentation, organisational policies, formalised processes, and internalised risk management strategies. Less often, participants mentioned external guidelines and standards, such as: ‘General guidelines, like GDPR, you can get some stuff from the ISO 27000’ (P5).

Lack of formalised and standardised policies may lead to product incompatibilities, inconsistencies, and engineers’ frustrations about time wasted on implementing sub-optimal privacy mitigation solution. The value of formal processes is especially critical in reconciling the disagreements among experts about best practices and advice: ‘We recognised the value of having the standard . . . to synchronise our thoughts on something before we provide someone with a recommendation’ (P6). Formal procedures and policies also leverage Privacy Champions’ ability to advocate privacy features.

On the other hand, preparing documentation and reviews takes time and creates friction between teams: ‘Nobody wants to be audited or write documentation that much if they could write code instead’ (P8). Therefore, combining formal procedures with informal roles of Privacy Champions or other privacy experts offers a balanced solution to promoting privacy: ‘That was seen as the advantage of this role. That this dissemination of knowledge that was the goal would happen organically rather than formally’ (P2). At the end of the day, some participants believe that documentation cannot substitute human involvement and expertise in providing customised guidance and help, emphasising the benefits of moderating role of Privacy Champions or other privacy experts: ‘There are tonnes of documents, but basically, they point you to the right people to talk to . . . you have to talk with someone who understands . . . your problem better’ (P3).

Training and Mentoring

Privacy Champions (8/12) talked about the role of training and mentoring in promoting privacy values, however, their opinions about its effectiveness were

nuanced. For instance, in addition to shifting attitudes and raising sensitivity to privacy issues,

Privacy Champions believe that training should provide practically useful information on how to implement privacy principles to be a valued resource for developers: ‘If I talk to someone out of blue about this . . . maybe they’re not so interested, but when they actually have to use this data they are more receptive into what I have to tell them about it’ (P3).

For the same reason, design and code reviews can have a better educational effect than formal training, due to their practical relevance: ‘The developer education seems to be more effective once they’ve had a review and they see how we think about things, and they start to change’ (P6). Similarly, delivering information about organisational privacy documentation ‘*that includes the security and privacy checklist*’ during on-boarding training for new hires may be ‘the wrong time to do that’ (P6).

Moreover, training targeted to the specific audience or topic that is ‘relevant to those people’s technical jobs’ (P11) is more effective and motivating for the engineers than general privacy awareness programs: ‘It was better to have a subject matter expert come in and teach people within the team or within close by teams, rather than have everyone know everything’ (P2).

On the other hand, mandatory training applied selectively to the teams can be perceived as punishment, e.g., for the mistakes they made in implementing privacy. To mitigate this, P1 recommends to change the tone of the purpose for training assignment, approach the team lead and offer a privacy session tailored specifically for the target team with the examples relevant to their product, rather than positioning it as a behaviour correction measure: ‘They’re likely to show up to that anyway because you made it exciting to them’ (P1). Even more generally, P1 believes that punishing and shaming developers for not being concerned about users’ privacy are not effective approaches for instruction and behaviour change in the organisations; instead it may make developers defensive and secretive about privacy issues: ‘They go into this, ‘How do I make sure my team doesn’t get in trouble with the privacy team?’’ (P1).

Additionally, mentoring can be effective in educating developers about privacy: ‘We do have a strong internal mentorship programme both formal with expectations or pairing junior developers with more senior developers and senior managers’ (P7).

Tools and Libraries

Privacy Champions use or build tools and libraries to assist others in developing privacy-preserving products, testing, and vulnerability discovery (7/12), in

4. Privacy Champions in Software Teams

addition to using such common approaches as cryptography, k-anonymity, and differential privacy. For instance, libraries can offer choices that are privacy-preserving by default, and built in the best data protection practices hence, minimising the chances of making mistakes for developers:

Give people libraries, tools etc that are already built in a way that tries to minimise data . . . You're limiting the choices that are available, to only the choices that are deemed to provide enough privacy or enough security (P2).

Data flow modelling and data annotation techniques further assist developers in thinking about privacy implications:

I have seen people look at designs for how they're planning to store data and go, 'Oh, we actually don't need all this sensitive data. Dealing with sensitive data is annoying, we can design this feature so that we use public data to solve this problem'. (P1)

Our participants mentioned some automated tools that detect vulnerabilities: 'There is a lot of automated systems in the company and most of them work when you push a code to GitHub . . . It would prevent you from merging code if it said 'really high vulnerability'' (P8). However, most of the mentioned automated tools are focused on security; indeed, P6 expressed the need 'to have more automation' (P6) for discovering *privacy vulnerabilities* and provided an example how 'to prevent other third-parties from learning about our users, we proxy all requests to third-party services, like for example, Google Safe Browsing' (P6).

External Factors

External factors, outside of the company, may influence the adoption of privacy principles within the organisations (8/12). One of these factors is political and regulatory support (e.g., EU GDPR, CCPA, FERPA): 'Because you had that soft power and influence and buy in from people that comes from not just inside the company but from the whole society' (P1).

Privacy champions believe that academic work also influences organisational privacy practices, however, academic research is not always practically applicable: 'They are the kinds of things that people publish papers about in web privacy are mostly often tales and novel and not actually useful' (P10). Finally, public critique in mass media or through the open-access and public-facing documentation encourages 'transparency and accountability' (P7).

Criteria for Assessing the Effectiveness of a Strategy

We asked Privacy Champions to tell us how they know if a strategy or a communication channel is effective or ineffective. Many Privacy Champions often mentioned practical usefulness (8/12): ‘We were able to do these [data flow modelling] and come back with very, very definite, very concrete requirements which were really appreciated by the engineering staff’ (P11); especially if the proposed privacy approach can save developers’ time: ‘Developers really want to have code in production as soon as possible, so, any kind of benefit to that is a massive win for them’ (P2);

or reduce the tension between teams: ‘We started to be more consistent about doing spec reviews and inviting people to publish their specs earlier, and we’ve had a lot less fights with people at the implementation level’ (P6).

Positive impact on end-users and developers’ decisions and attitudes, or lack of that impact, is another factor that Privacy Champions use to estimate the effectiveness of a privacy-promoting strategy (8/12).

Given the lack of standardisation and evaluation metrics, discussed in Section 4.4.4, the ability to measure the impact of a strategy or approach, or define the minimum requirements is especially appreciated by Privacy Champions (4/12): ‘I and a couple of other people are working on some equipment privacy metric and I think that will be enormously useful in prioritising and motivating the development of certain features’ (P10).

Finally, relevance of a strategy or information, e.g., training content (see Section 4.4.5), to a particular audience is another criteria Privacy Champions (3/12) suggest considering when defining its effectiveness. For example, broadcasting messages or company-wide training may not be as effective as information targeted to a certain audience ‘because people tend to read it and then quickly forget about it’ (P6).

4.4.6. Information Resources

We asked Privacy Champions how they keep up with the latest in privacy. Online resources, including articles on the Internet, general media, news, and blogs are the most common online resource about privacy among our participants (9/12): ‘Knowing what they’re saying about privacy on NBC and CNN and Fox News and the New York Times can really give me a sense of what the general population is seeing’ (P1). Online social networks, such as Twitter and LinkedIn groups, Reddit, other fora, and newsletters are also popular sources of information. Half of our participants (6/12), in positions related and not related to research, read academic papers and attend conferences to keep up with the latest in privacy.

Privacy Champions also learn about latest achievements, best practices, and mistakes in privacy domain from the experiences of other companies (4/12). Some even have ‘shared channels with other companies’ (P6) to exchange information.

In-person communications with peers, attending industry events, workshops, and working groups help Privacy Champions (3/12) stay tuned as well: ‘A series of workshops, I went to one where they were gathering feedback on their privacy framework, and learned a ton there, and also got to contribute to that conversation’ (P12). Internal organisational channels, such as Slack, are common and useful resources for both finding and promoting information about privacy (2/12): ‘We have a Slack channel, where everybody shares articles that they’ve encountered’ (P6).

4.5. Discussion

Privacy engineering is a challenging task for developers [141, 295, 315, 330]. Our interviews demonstrate that, similarly to Innovation Champions in other domains, including cybersecurity [144, 146, 147], Privacy Champions are promising facilitators of the privacy transition in software teams. However, they need support from organisations and peers to succeed in their efforts.

4.5.1. How to Motivate Privacy Champions?

Given the promising role of Privacy Champions, the logical question arises: how to find, retain, and support motivation of Privacy Champions? We found that self-motivated Privacy Champions seek employment in companies with strong privacy culture and like-minded colleagues, and avoid companies with weak privacy values (see Section 4.4.3). This finding suggests that Privacy Champions may be especially concentrated in a handful of privacy-focused companies and be rather rare or muted in other companies. Therefore, putting privacy values at the forefront of the company’s mission would not only strengthen the competitive advantage at the user market, but also help attract and retain Privacy Champions.

Privacy Champions are motivated by personal and organisational values, similar to other champions of innovation [160, 277]. Like security advocates [59], Privacy Champions’ attitudes often form from personal experience with privacy risks. In contrast to the security domain [147], privacy has a strong connection to social norms and ethical values; Privacy Champions see privacy as a fundamental human right and feel personal responsibility to protect it and satisfaction from creating benefits for society. This passion explains why many of our participants continue being the voices of privacy despite their efforts not being officially recognised or compensated. Therefore, the recruitment efforts (within or outside of the

organisation) directed at Privacy Champions need to emphasise their positive impact on users and society, possibly with the supporting examples from media, creating a sense of purpose and mission, which has been proved as effective driver in psychology and management [75, 77, 367].

Privacy Champions, like other engineers [142], enjoy solving challenging tasks and appreciate the recognition of their efforts (see Section 4.4.3). Thus, organisations and peers should stimulate their curiosity, encourage them to use their unique expertise to find privacy-preserving solutions for technical issues, provide intellectual freedom and resources for exploring new approaches and ideas [55], and acknowledge their efforts not only via explicit positive feedback, but also via career promotions and fair compensation for the additional (often voluntary) work they do.

4.5.2. Support the Motivations

Privacy Champions often face developers' low motivation to address privacy issues in software design due to indifference and negative privacy attitudes (see Section 4.4.4). Similarly, Security Champions often have to overcome apathy towards security by making it tangible and relatable using stories and analogies to help team members understand [146]. While security has objective tangible benefits, the value of privacy is hard to measure thus it is more subject to diverse personal attitudes (see Section 4.4.4).

To address such negative privacy attitudes among members of software development and product design teams, it is important to improve organisational privacy culture. To achieve that, Privacy Champions in our study recommend encouraging formal and informal discussions about privacy implications for end-users. The discussions about privacy can take a variety of forms, from seminars and lightning talks, specialised channels (e.g. Slack groups or message threads) dedicated to discussing privacy questions and exchanging resources on the topic, to motivational speeches during all-hands company meetings, where management can show their support and recognition of the importance of privacy values and leverage 'social influences' [71].

The technical complexity associated with designing and implementing privacy-preserving solutions (Section 4.4.4), can be leveraged to increase the motivation of engineers, who find solving difficult challenges rewarding (see Section 4.4.3). Companies could emphasise the prestige of privacy engineering work due to the level of expertise it requires, and praise developers and provide them with tangible rewards, e.g., career promotions or additional compensation, for improving privacy in their products.

Moreover, it is important to improve communication between teams, aligning priorities of different stakeholders, and increase diversity in the teams to invite the variety of opinions to the table. Similarly to Security Champions [326], Privacy Champions can leverage their multidisciplinary knowledge and skills to facilitate the communications between legal and development teams. Participants acknowledged the value of special interest groups that focus on privacy and are comprised of members of different teams to facilitate the transfer of knowledge between teams and ensure that each team has an expert they can consult about privacy matters. In contrast to findings about Security Champions [147], Privacy Champions in our interviews did not find it useful to punish or shame developers for not addressing privacy concerns (see Section 4.4.3), and recommend employing a rather collaborative approach.

External influence, e.g., media stories, public critique and privacy regulations, can also increase developers' awareness of users' concerns, reinforce privacy norms and social values, and provide basis for judging privacy-related misconduct. Privacy regulations also establish penalties for privacy violations, motivating companies to include this aspect in the cost-benefit analysis. Open-source documentation further supports the corporate and individual developers' accountability and responsibility over designing privacy-preserving systems and solutions that respect privacy norms.

4.5.3. Support the Opportunities

Privacy Champions in our study reported that software developers are more likely to push back the engineering goals related to privacy when their opportunity to work on these issues competes with other technical or business priorities (e.g., primary product functionalities, performance, and revenue), and is limited by time and financial resources (see Section 4.4.4). Similarly, security also doesn't receive as much developers' attention as functional requirements [326].

To provide developers an opportunity to think about the privacy implications of their software throughout the development, privacy considerations should become an integral part of software development process, so that the project timelines and deadlines account for the additional time required to address privacy concerns, and project headcounts include engineers whose responsibilities involve such work. The principles of Privacy by Design (PbD) [78] provide a useful framework and a starting point for incorporating privacy considerations throughout the software development life cycle. In line with PbD, our participants repeatedly mentioned the importance of thinking about privacy impact early in the process, starting from the design reviews during the requirements stage. Design and code reviews offer a good opportunity to supervise the progress on a project, check the quality of implemented safeguards, and detect vulnerabilities. Moreover, opportunity to

comment on design should be offered to all employees, instead of limiting it to a specific team, to check in with the interests of other stakeholders, take advantage of the diversity of perspectives, and further encourage strong privacy culture. As security reviews are already common, privacy reviews can piggyback on them by adding to their templates a block of criteria for evaluating privacy.

Finally, companies could organise privacy-focused hackatons, which could encourage engineers to both identify the current issues and compete for finding the best and novel solutions for them.

4.5.4. Support the Capabilities

Our participants acknowledged that they and engineers they work with sometimes lack the knowledge about privacy and how to implement it. To overcome the technical challenges of privacy engineering, we propose to increase developers' knowledge, awareness, and skills around privacy and facilitate the task itself.

Increase the Knowledge, Awareness, and Skills

Prior work has shown the value of University-type education in improving privacy and security skills of software developers [22, 301, 325]. In most computer science programs, computer security is not a mandatory course [22] and privacy engineering programs are rare [88]. However, modern software developers need to think not only about the functionalities but also about the ethics of their products, encouraging to include the topics of privacy and ethics in the curriculum. This does not mean that every software developer needs to be an *expert* in privacy; if most developers in a team have at least a basic understanding of privacy requirements and ethical values, Privacy Champions and other privacy experts can assist with the nuances of its implementation.

At the workplace, when deploying privacy training, our participants recommend teaching engineers practical skills relevant to building privacy-preserving systems and targeted to their roles rather than raising their general privacy awareness and concerns. In terms of timing, our participants find privacy training to be rather ineffective during the on-boarding process for new hires, as new employees lack the familiarity with the specifics of the product they will be working on to properly contextualise their knowledge. Instead, they recommend integrating it directly into the development work. For instance, in addition to advantages discussed in Section 4.5.3, design and code reviews can educate developers about the company's values and the concept of privacy using practical examples from their own work.

Mentoring programs is an alternative way to integrate practical privacy education throughout the development process. However, to be effective, mentors need certain guidance themselves on how to best supervise someone's work, deliver critique and advice, and encourage critical thinking of their apprentices.

As Privacy Champions often rely on online resources and academic work for learning about privacy, we encourage researchers to share their work not only in academic venues, but also in blogs, online social networks (e.g., Twitter and LinkedIn), professional newsletters, and general media outlets and news sites (see Section 4.4.6). Privacy Champions may be instrumental in sharing this knowledge with the development teams. Companies can also create more opportunities to exchange their experiences, success stories, and mistakes in addressing privacy issues, for example, through newsletters, meetup groups, workshops, and company blogs.

Alleviate the Complexity of Privacy Engineering

In addition to design and code reviews, to incorporate privacy considerations into formal processes, our findings suggest using verified libraries that do not contain privacy threats as well as tools that help to annotate data sets, map data flows, and automate the detection of privacy threats. Such tools should be practically useful and effective, and save developers' time without introducing additional burden [296]. Since security tools are already commonly used in the organisations, the new privacy features can be incorporated into those existing tools to further facilitate adoption.

Providing developers with regulation-compliant and user-friendly privacy consent templates, and code samples for its integration could help follow the best compliance and user consent practices and avoid mistakes. Our participants acknowledged that recommendations that help interpret legal documentation and translate it into technical requirements would also help developers incorporate privacy in software design, and facilitate communication between different stakeholders, including engineers, regulators and lawyers, and business management.

Several participants find it difficult to measure privacy risks, and effectiveness of mitigation strategies. Over 80 privacy metrics to measure privacy aspects of a system were proposed in academic research, such as, time that it takes an attacker to violate user privacy or how much information an attacker can gain [356]. Nevertheless, only a few metrics (e.g., k-anonymity and differential privacy) were mentioned by our participants. Increasing awareness of the existing metrics and developing new practical and robust privacy metrics could provide reliable tools for demonstrating the benefits of addressing and costs of not addressing a specific privacy issue, and aligning various conflicting corporate priorities.

4.6. Conclusion and Future Work

We show that Privacy Champions play an important role and have strong personal and organisational motivations to promote privacy values in software development teams, despite the challenges they face. We discuss the main strategies and communication channels that Privacy Champions use to overcome those challenges, and resources they use to learn about privacy matters. We discuss how organisations and team members could assist Privacy Champions by providing organisational support, resources, and simply acknowledging their efforts.

Future research is called for to quantify the prevalence of identified challenges to adoption of privacy practices in organisations, evaluate the effectiveness of strategies, develop robust and standardised taxonomies of privacy risks, detailed practical guidelines and privacy engineering recommendations on how to technically address privacy issues, explore the reasons why existing privacy metrics are not widely adopted in software development industry, and propose solutions to those issues.

Acknowledgements

We thank all the participants for their time and valuable inputs, Julie M. Haney for sharing materials from their study on cybersecurity advocates and helping with the recruitment, Mary Ellen Zurko for helping with the recruitment, and Adam Jenkins for his feedback on earlier versions of this paper. We also thank the anonymous reviewers whose comments helped improve the paper greatly. This work was supported by the Center for Long-Term Cybersecurity at UC Berkeley, National Science Foundation grants CNS-1514211 and CNS-1528070, Microsoft Research through its PhD Scholarship Program, and a Google Research Award.

Back Cover

Privacy champions can provide invaluable input to developers and by just having conversations about privacy with other developers promote a privacy culture in software companies. In the context of developer-centred security, prior research has shown success in incorporating security champions in software teams, I believe that having a separate role as a privacy champion can further strengthen privacy values in software teams and potentially bring competitive advantage to companies that are looking to provide privacy to their customers in the digital market. Another observation was that privacy champions are motivated by several factors including having some educational background in privacy and security. In the next chapter (Chapter 5), I follow this direction and conduct interviews with computer science students to understand their privacy and security mindsets.

שְׁמֵרָה (ציי)

5. 'I Don't Know Too Much About It': On the Privacy and Security Mindsets of Computer Science Students

Front Cover

In the interviews with privacy champions, I found that some of them are motivated to champion for privacy because they had privacy and security courses during their time as undergraduates or postgraduates. Therefore, I decided to better understand how computer science students who are one of the main parts of the incoming workforce for software teams think about privacy and security, and how would they consider privacy and security features while thinking about design of a hypothetical app.

This paper investigates the privacy and security perceptions, experiences, and practices of current computer science students at the graduate and undergraduate level using semi-structured interviews. My co-authors and I find that the attitudes of students already match many of those that have been observed in professional level developers. Students have a range of hacker and attack mindsets, lack of experience with security APIs, a mixed view of who is in charge of privacy and security in the software life cycle, and a tendency to trust other peoples' code as a convenient approach to rapidly build software. We discuss the impact of our results on both curriculum development and support for professional developers. Our observations show that developers need more support from academia so that they can accomplish privacy and security tasks in their future career.

From the privacy champions study we learned about many misconceptions and concerns fellow developers have. Its notable that many of those expressed concerns are in-line with what we were observing from students, irregardless of if they had or had not take an privacy or security course. So while taking such a course may be motivational to a privacy champion, current courses do not necessarily dispel common misunderstandings among students which may in turn lead to those same misunderstandings existing in development teams.

5.1. Introduction

Software developers can impact millions of lives with seemingly small security decisions that have a large impact on the people using the technologies. One example is the case of the dating site Ashley Madison, where a strong cryptographic algorithm was used to store passwords but was implemented incorrectly [199].

Even for apps where security is not a primary feature, it is a requirement needed for stability and safety of operation. Therefore, software developers need to be keenly aware of the security implications of their design decisions. Ideally, they should have strong support from their tools to avoid privacy and security issues in their resulting code.

Basic tools such as cryptographic libraries (OpenSSL) and federated authentication (OAuth) exist partially to assist developers in integrating common security needs into their projects without needing to know all the complex details. There are also efforts to help raise awareness of common coding and design issues such as the IEEE top ten security flaws [7, 32].

Yet, security remains a pervasive problem in deployed code. In 2013 alone, 88% of apps (out of 11,748) analysed on Google Play had at least one mistake in how the developer used a cryptographic API [106]. Code that they write goes into security-critical applications such as banking software [128] as well as software with less obvious security implications such as Internet connected kettles [224].

Non-usable APIs are a key point of failure for most developers [5, 135, 162, 265, 376]. Providing manuals is not enough. A usability evaluation of programming security in Android found that developers created code with security errors even when they were provided with official documentation [3]. Perhaps more importantly, developer understanding of security is also problematic. Interviews with professional developers show a range of concern about privacy and security knowledge [42]. The situation is exacerbated when developers make non-obvious errors when implementing security which results in believing that code is secure when it is actually not secure [2].

One potential opportunity for changing developers' security attitudes and practices is during their training. In this work, we investigate the privacy and security mindsets of a group of twenty graduate and undergraduate computer science (CS) students on a variety of career trajectories, and with a range of exposure to formal security training. Our research questions are:

RQ1: What are students' comprehension of privacy and security related concepts?

RQ2: To what extent do students consider privacy and security while coding applications, and how do they implement it?

Within the context of developer-centred security, our study highlights the extent to which students already have similar mindsets and practices as have been found in professional developers, suggesting that these may form and consolidate early. We conclude that, while early educational intervention would be ideal, we also need to provide developers with usable tools, such as APIs, and easily accessible training, which can be used both by trainees and professionals.

5.2. Related Work

Creating secure software correctly is quite challenging even for professional developers, often resulting in unintended security vulnerabilities [5, 135, 376]. The OWASP organisation publishes the top ten most critical web application security risks every few years. A review of their last three reports covering seven years terrifyingly that the most common issues are quite stable [253], with common and highly damaging vulnerabilities such as code injection and broken authentication continuously remaining in the top ten.

Arce et al. observed that many of the OWASP vulnerabilities represent unintentional errors or mistakes rather than planned actions and therefore are minimally helpful to someone trying to design a secure system [32]. Instead they propose a set of top ten *security design flaws*, that is security issues that are a planned element of the software. Their list is much higher-level and contains issues such as ‘earn or give, but never assume, trust’ [32, p. 9].

The problem of code vulnerabilities in live software is further exacerbated by the steady reduction of the barriers to entry for new software creators. While generally a good thing, the ‘anyone can code’ movement has also led to an increase in the number of software creators with minimal formal training in software development and even less training in security. Unsurprisingly, this group also has difficulty creating secure software [248, 265].

Neither of these groups is, or should be, expected to be security experts, but the decisions they make can still have serious security impacts. In an effort to better support these software creators, several tools and libraries have been proposed such as OpenSSL, PyCrypto, and cryptography.io which encapsulate many of the security decisions, theoretically making development easier.

Unfortunately, many of these tools still suffer from usability issues, such as confusing API designs [2, 106, 113, 128, 178, 192, 343] or poorly designed documentation [3, 227]. Official documentations are often not easy to use, hence developers prefer online resources which may not offer valid and secure solutions. While Stack Overflow, for example, helps with getting code working quickly, the suggested solutions may also result in less secure code [3, 115].

Security is also challenging for developers because it causes no obvious visual effect, making it difficult to identify when an unintended state has occurred [112, 113]. A common example of invisible security effects is SSL/TLS. When used incorrectly, a connection is still formed, but that connection might not be encrypted, or it might be encrypted, but without certificate validation. This results in a vulnerability to man-in-the-middle (MITM) attacks during connection setup. Fahl et al. observed how challenging this can be for developers to spot. One of their developers even used Wireshark to ‘test’ the SSL/TLS connection and, because the data was garbled looking, incorrectly concluded things were working even though no certificate checking was happening [113].

Georgiev et al. similarly conducted an analysis of SSL certificate validation in multiple non-browser platforms and showed that many applications on the market are open to a MITM attack where data can be read and modified in transit because developers accidentally or intentionally configure their code to not validate the certificate source [128]. Such problems arise when developers are expected to understand the implications of the different settings of SSL, which is exacerbated by APIs that do not offer a helpful level of abstraction [192].

Security is also not a well-established requirement in the software development workflow. Without a dedicated developer in charge, security becomes a hot potato which is passed between groups because no one wants to deal with it [42, 245, 267, 366]. In interviews with security experts, Thomas et al. found that security auditing is seen as a separate task from software development. While security auditing is performed by the rare breed that are security experts, it is then the developer’s job to fix the security issues found [335].

Many future software developers were once Computer Science (CS) students. A survey by Stack Overflow in 2019 showed that 62.4% (75,614 responses) of developers have a degree in CS, computer engineering, or software engineering [95]. Given the importance of this group, many researchers study them to either address gaps between academia and industry [73, 167, 271, 272, 317] or to suggest educational tools to improve their skill and abilities [234, 320, 369]. Research shows that CS students often work under misconceptions which can lead to bad practice. For example, when it comes to software engineering processes and teamwork [317], many think that working alone is a quicker way of working on a software project, which goes against established industry best practice. Here we study the privacy and security mindsets of CS students with a view to identifying what they know and think about privacy and security, and what misconceptions exist.

5.3. Methodology

We used semi-structured interviews to explore how a range of students from undergraduate to PhD think about privacy and security. The semi-structured approach allowed us to probe students' privacy and security mindsets in detail and investigate how they relate to their own practices as developers.

5.3.1. Interview Design

After informed consent, we explicitly invited participants to talk as much as they wanted on the various topics discussed. The interview began with an open question on academic and professional background and general questions about coding and software development experience. Questions about demographics were asked at the end of the interview in order to minimise stereotype threat. The full interview script is included in the Appendix B.

We began the privacy and security discussion by asking participants to consider creating 'a new group discussion app for in-class discussions'. They were then asked to free-list the app's features on paper and after they finished they were asked to circle those that were privacy and security related.

Next, we examined participants' understandings around threats and hackers. We started by asking participants about the hypothetical app: 'Who is most likely to try and attack this system? What are they likely going to try and do?' We then moved on to talk about hackers, because work on security folk models has found them to be an important part of how people think about security [361]. We elicited participants' definitions of the term hacker, and their views on hackers' intentions, goals, and background.

We then moved on to considering who was responsible for privacy and security in software development practice. The discussion was grounded in participants' own experience of writing software, in particular problems with (security) APIs.

Finally, we asked participants about personal privacy and security practices. First, participants were asked to list the words and concepts they associated with 'computer security' on paper. We followed up with questions about good security practices, and their own security practices.

Since prior negative experiences can impact future choices [350], we also asked about prior experiences with compromise, prompting them with examples such as 'getting a virus on your computer, losing your password, having an email sent from your account, or loss of data about you' if needed. We explored how the experience was resolved, and what participants learned.

5.3.2. Recruitment

We recruited participants through mailing lists associated with a large Russell Group University in the United Kingdom, Facebook groups, and word of mouth. Advertisements asked for Computer Science students (BSc, MSc and PhD) to participate in an interview about opinions and attitudes around software development, particularly around the handling of requirements prioritisation. All advertisements avoided privacy and security related words to limit self-selection and priming.

5.3.3. Participants

Our sample, shown in Table 5.1, includes twenty students (6 BSc, 11 MSc, and 3 PhD students), participants who previously took a computer security course at any University are indicated with 'PS' instead of 'P'. The sample contains five female, and fifteen male students with an average age of 24 years old ($range = 20 - 37$, $SD = 3.8$, $median = 23$). They come from various countries and have diverse CS-related educational backgrounds. Interviews were conducted in English. Our sample reflects both the diversity seen in the tech industry [41, 131], and the culturally diverse classrooms found in many computer science departments.

The interviews were advertised to be 60 to 90 minutes long with a compensation of £10 in cash. In practice, interviews took an average of 68 minutes ($range = 41 - 108$, $SD = 18.4$, $median = 65.5$) and were completed in July 2018. All interviews were audio recorded with participant consent. The study was conducted in accordance with the Ethics procedures of the School where the students were recruited (cert ID 2870).

We interviewed students over the summer. This meant that the Masters students were in their dissertation phase, and had completed the course work part of their 12-month degree. PhD students in the UK have typically completed a Masters before starting a PhD and are not necessarily required to take courses, pass a qualifying exam, or be a Teaching Assistant, though many choose to take additional courses and tutor. Therefore, beyond teaching and thesis work, PhD students are unlikely to be impacted by security courses taught at the University.

5.3.4. Pilot

We conducted seven pilot interviews with Masters and PhD students, six of which were associated with our research lab but unfamiliar with the work. These interviews were used to iteratively refine the interview script as well as adjust the number and content of questions to keep interviews at about 60 minutes. The pilot contained some students with no security background to help ensure the phrasing

of security questions was clear. Feedback was also sought about the structure, clarity, and accuracy of the interview schedule. Pilot interviewees and interviews were not used in our final analysis.

5.3.5. Interview Analysis

Interview analysis focused on uncovering students' mindsets of privacy and security as they relate to the software development process. Relevant themes were extracted using a three stage process. First, two researchers listened to the full audio of four interviews which had been selected by the interviewer to cover a wide range of participants, identified relevant parts for more detailed analysis and transcription, and outlined an initial topic guide for coding [212, 284]. Audio was used because it provides a richer record of the original interview than a standard transcript. In the second stage, the researchers performed open coding of the transcripts based on the topic guide [212, 284].

In the third stage, the open codes were analysed using an affinity diagram [181] to yield a set of seven themes, which are discussed in the Results Section 5.4 below. While some authors suggest reporting how many participants mention each theme [181], we chose to follow standard qualitative research reporting practice and focus on describing and contextualising our themes [276, 361].

Table 5.1.: Interview study demographics. P = participant without computer security background; PS = participant who self-describes as having taken a computer security course in the past.

Participant	Gender	Nationality	Expected Degree	Age
PS01	M	EU	PhD	29
P02	M	EU	MSc	28
PS03	F	Asia	MSc	22
PS04	M	Asia	MSc	24
PS05	M	Asia	PhD	25
P06	F	Asia	MSc	23
P07	M	Asia	BSc	22
PS08	M	UK	MSc	21
PS09	M	Asia	MSc	25
P10	M	Asia	BSc	21
P11	M	EU	BSc	22
PS12	M	Asia	MSc	23
PS13	M	EU	BSc	21
P14	M	EU	BSc	20
PS15	M	EU	PhD	25
PS16	M	Asia	MSc	37
P17	F	EU	BSc	25
P18	F	Asia	MSc	23
P19	M	UK	MSc	24
P20	F	Asia	MSc	20

5.4. Results

All participants all had some form of prior programming experience ranging from classroom projects, internships, and prior employment in industry. Since our participants included a large number of Masters students, they also had classroom experience from prior universities, with several expressing that they had worked in industry either as interns or full time before coming back for a Masters or PhD. Half had taken a computer security course at some point in their education. We did not ask about the details of these courses.

5.4.1. 'Computer Security' Word Association Results

Mid-way through the interview participants were asked to free-list words associated with 'computer security'. The words were grouped into topics by the lead researcher with a bottom-up approach. A second researcher then reviewed the groupings and disagreements were resolved through discussion. Table 5.2 shows the resulting eleven topics.

Participants' understanding of the term 'computer security' was broad, with participants who wrote words providing an average of 9.6 words ($range = 2 - 19$, $SD = 4.2$). Listed words included standard security topics such as encryption, attacks, and system security which are readily found in most security text books. Participants also listed company names that are either associated with security (Norton) or that had been discussed recently in the news in relation to security (Facebook [243, 354]). Two participants (P02 & P20) were not able to list any words, suggesting uncertainty with the term. 'It is all very flimsy' (P02), 'To be honest I do not know too much about it' (P20).

Of the participants who provided words, participants listed words from an average of 4.2 topics ($range = 1 - 7$, $SD = 1.8$). The topics cover a wide range, but each individual participant had less range, with at most seven topics mentioned by one participant. Most notable is the lack of a single common topic amongst participants. For example, the most common word 'privacy' was mentioned by only 40% of participants. Common security topics such as passwords, authentication, and encryption also appeared. Some of these topics are similar to what professional developers associate with security, for example, encryption, user control, and user access [141].

Table 5.2.: Topics mentioned during free-listing, number of words participants listed associated with that topic, number of unique participants listing at least one word associated with the topic, and a set of sample words representing the range.

Topic	Example Words	#Words	#Participants
Encryption	End-to-end, hash, RSA, public/private key, SSL, symmetric.	28	11
Authentication	Passwords, permissions, 2FA, tokens, access controls, emails.	28	9
Privacy	Anonymity, right to be forgotten, visibility, cookies.	27	10
Attacks	Reconnaissance, phishing, buffer overflows, DoS, MITM.	25	8
System security	Protocols, database, Unix, system calls, TCP/IPs.	13	5
Social	Regulations, roles, responsibilities, public knowledge.	13	7
Finance	PayPal, Apple Pay, Bitcoin, online payments.	8	4
Defending	Anti-virus/malware, penetration testing, logging, bounties.	7	5
Security holes	Failures, physical access, loopholes.	5	4
Companies	Facebook, Google, Norton, Red Hat.	5	3
Trade offs	Usable security, features vs security, easy to use UX.	4	3

5.4.2. Interview Themes

Security Mindsets

Participants varied substantially in their understanding of privacy and security. While some participants had a strong up-front understanding of security which varied minimally during the interview, others had clearly not thought much about the topic before resulting in them re-thinking their opinions mid-interview. This is to be somewhat expected as many people have not previously devoted extensive time to assessing their own understanding of the topic [34].

This theme provides rich additional context to the initial topics identified through free association. Those with a more sophisticated understanding of privacy and security tended to use more definitive language, had more stable descriptions of attacker motivations, and were more likely to be sure that their statements were accurate, and to describe less intuitive or extreme scenarios. For example, PS15, a cryptography PhD student, explains that ‘in crypto, we assume that the attacker is any code, literally any Turing machine’ (PS15).

Those with an initially less sophisticated understanding of privacy and security showed signs of forming their opinions as the interview progressed. Often, this would involve contradictions in thoughts as they finally reached a definition for themselves. This was most notable for the hacking theme. Participants with less developed models exhibited less self-assurance around motivations, or definitions of attack scenarios. ‘I think [HTTPS] is standard by now, don’t they? The more encryption the better? . . . Like exchange of data that’s not encrypted at all. I don’t think that’s happening anymore. I’m not sure but I don’t think it is’ (P17).

Similar to non-tech savvy users [361, 389], some of our participants think they are not a target for attackers. ‘We are just average people. It is ok to have small security measures’ (P11), ‘I am also very boring computer user. I just do my courses and I watch movie on Netflix. So I don’t really do anything that could put me in front of a virus’ (P20). Conversely, some participants had high awareness of potential attacks, though they still did not perceive themselves as at risk.

I am running a server at home, which has an SSH access available. There you can see a lot of stuff going on, there are just bots or so whatever trying to get into. That is even a bit scary if you see that happen all the time, but I think my pass has been strong enough to keep them out. (PS13)

Participants clearly evidenced their own internal struggle over what privacy and security actually was and when it was or was not needed, which might partially explain its lack of inclusion in initial requirements. ‘[My address] is not so important, because every website is required. Maybe because I live in a dormitory, if it is in my home that is different’ (P06).

While participants understood that private data should be protected, they struggled with what ‘private data’ actually meant. Even when talking about privacy and security in their private lives, participants had mixed opinions about how problematic it was for data like bank transactions to be leaked.

So the data [leak] was about the full info about the bank accounts, the transactions, in and out, the current amount in it. For me it was normal . . . to have these transactions. But for some people it was an issue, because they receive money from hidden source, so it was an issue for them. (PS16)

Who Are Hackers and What Do They Want?

Some participants’ definitions of hackers were well articulated. ‘Really theoretical let’s say, the adversary we say in crypto is literally anyone that has a computer and some access to your systems’ (PS15). Other participants had a more general understanding. ‘The images that you have in your head are from Hollywood. Super smart kids sitting in the corner of a room then CIA calls upon them to solve a problem’ (P02).

We found a wide range of imagined intentions for hacking, such as financial, personal, political, and just for fun. All four types of previously observed models of hackers from Wash’s work [361] were mentioned by our participants:

Graffiti, which is a mischief causing attacker with technical background: ‘Want to try what they learn from the class. They may write some code to hack some system of the school to show their ability’ (PS12); Burglar, who commits crimes

using computers mostly with financial motivations: 'There is nothing but personal interest. Personal gain. Personal satisfaction. And of course they are who just do it for financial gain. Stealing identities, pictures, personal info. Just to sell it afterwards, to like black market' (PS01); Big fish, who looks only for high valued targets: 'Political incentive that certain countries fund a lot of hacking and cracking to gain power depending how important or how famous you are there might be people who want to get access to your account' (PS13); and Contractor, a Graffiti hacker with financial/criminal motivations: 'Trained people who are trained to do this kind of stuff. Either by some governments to hack other governments. Or to break the encryption or security mechanism' (PS05).

The Role of Security When Planning Software

When participants were asked about what features they would consider in an in-class discussion app, they commonly mentioned functional requirements including task management, calendar, question/answering, recording classes, and assignment management. Many of these features currently exist in course management software with which the students are familiar, such as Blackboard LEARN and Piazza.

Only four participants (PS08, PS15, PS16, and P19) mentioned privacy and security in their initial design and feature list, a somewhat small number since ten of our participants had previously taken a security course. Only two of the participants proactively brought up privacy issues. 'First thing that comes to my mind is privacy. Definitely in terms of features. Presumably, the School will wish to host it locally rather than to have some sort of central cloud back service' (P19), while PS08 noted the connection between privacy and ethics: 'There is some ethical questions involved in the area of student privacy' (PS08).

Security of the data was also a concern, particularly in terms of information leaks. 'I will make sure of the safety and security because [no one] wants to use the tool if he feel he is vulnerable his info may leak to any unwanted person' (PS16). PS15 was also able to pull on prior experience and identify specific attacks and solutions that needed to be addressed:

For sure I put HTTPS and TLS around it. So that would be safe. Because still, I would leave a lot of surface for attacks, because the big applications have more surface for attacks . . . All those places where there is user input we basically talk about security, and we have to remember SQL injection and stuff like that. (PS15)

Some participants turned to more authoritative sources such as laws, regulations, and public policies as a guide for what should and had to be built into the system. 'You have designed an app I guess you also think about security. But you also think about engagement. Does a certain security feature if it is option not legally

required, how does it sort of effecting the engagement' (P02). Some mentioned the EU *General Data Protection Regulation* (GDPR, enforcement date: 25 May 2018) [127], either as a convenience tool for end-users or from a regulation perspective for companies. 'Do we have to be GDPR compliance? Probably, I'm guessing' (P11) was mentioned by a participant when answering a question about what privacy and security features his hypothetical classroom app might need.

Requirements and Responsibilities: Playing Hot Potato

Several participants recognised security as an *explicit requirement*. They consider the developers' job to be transforming requirements into code. Therefore if security is an explicit requirement, then they have to take it into account during design and development.

So as a software engineer, if I am already given a certain requirement, I should not care about anything else outside the specs. You are employed as software engineer, you just write your. You are given a list, you just have to code it. Right? Unless you can do that. You are still doing your job. (PS01)

On the other hand, other participants see security as an *implied requirement* that is always present.

When the requirement is out but [privacy] has to be taken care of at every single step here. If someone comes to me asking for something then I assume that I do security for all the requirements. Wherever applicable security should be. (PS04)

Security was also sometimes seen as a problem or requirement that should be solved by a designated entity within their workflow. For some participants, this entity was the *operating system* 'Android, it is responsible. Because Android restricts my way of developing an application. So it should provide sufficient security mechanism for me to rely on' (PS05), 'Mostly the OS is the one that should provide security' (PS15). Others considered that a *security team* in the software development workflow should be responsible. 'There should be a security team. Which takes care of that. Just like any other team inside the company. Like UI, testing team' (PS04).

Many interviewees thought that the *company* as a legal entity is responsible for privacy and security, and some highlighted the role of legislation and government. 'We are [responsible]. Not me personally but the company that I work for as a legal entity' (P17). Moreover, a few saw *end-users* having some responsibility as part of the larger privacy and security ecosystem. 'There should be a certain amount of onus on the user, they should be responsible for like managing their password' (PS08).

General Attitudes to APIs

Participants saw APIs as a useful and handy tool, especially in terms of code re-use ‘GUI stuff in python, here you can just call functions without write whole part of code yourself. It’s always handy’ (PS03). APIs also allowed them to lean on the knowledge of others and not need to understand all the concepts themselves.

It is quite useful and simple to import the library from platform. Before I used that library I need to learn each algorithm one by one mathematically. In terms of math the algorithm is quite hard. With library I just can, I import them from Internet. With one or two lines code I can use them. I can focus more on main procedure of neural network and data manipulation, so I can save a lot of time with the library. (PS12)

Other peoples’ code was a large theme when discussing APIs, particularly examples posted online or documentation-like guidance from others. ‘Sometimes just some posts either forums or some question and answer community like Stack Overflow. There are people show you how to use in their answers, kind of you can copy paste and modify that to suit your needs’ (PS05). APIs also tended to be designed in such a way that they were easy to start using. ‘Maybe it is just experience, that makes it easier, because I was using APIs for so long so it is easy now to just come and start’ (PS01). APIs also made it easy to get code running quickly, especially if the documentation was good and contained examples. ‘If you pick a certain thing, you read the documentation, hopefully the documentation is done well, by done well I mean by examples. That you can get something to run as fast as possible because that keeps you motivated’ (PS13).

Security APIs

When asked about a ‘security API’ participants struggled to understand what that could even be, falling back on areas commonly associated with security, like finance. ‘What do you mean by security APIs? Something like payment gateway?’ (PS04). Only one participant had a hands on experience with a security API which was problematic.

There is no feedback [in Android certificate validation]. It is a complete nightmare, various very long complicated classes archaic options that you are supposed to set. All and all was 40-50 lines of code. This was just a block of imperative commands for doing something basic like I’d like to validate against certificate file please. Absolutely crazy. (P19)

While only observed from one participant, his comments closely match what other researchers have observed from professional developers [33, 106, 113, 128].

P19, who has industry experience both as a developer and an intern, was one of the few people who discussed issues around secure programming, such as buffer overflow and functions with known security issues. 'buffer overflows, system calls are an issue of languages, actually more than anything else. We still use C this is an atrocity, we shouldn't be using C anymore' (P19). He is referring to common C function calls like gets which are impossible to use in a secure way, but are still commonly used due to being part of core C [170].

Trusting Other Peoples' Code

Using APIs and examples from the Internet was convenient for our participants, but it also required them to trust people they had never met. Some were concerned about blindly trusting code from unknown sources, but many had no problem instead choosing to trust in collective intelligence.

If I download, I am often downloading source codes myself from the Internet and then building it. And again I don't have the time or the skill to audit say a code base that has millions of lines. I perhaps trust a little bit too much the crowd of people. If I look at the code base and see something on Github and it has let's say 2000 stars. Few hundred people watching it. The code is all open. I tend to perhaps foolishly I assume that if this many people have looked at it and if there was something up. Surely someone would do have said something. Download the code and build it. So it is possible that I have exposed myself to security issues as a result of that. (PS08)

PS08 is referring to the 'many eyeballs' idea in open source software which is an indicator of security and reliability of code for some developers [156].

Trust is an inherent component of open source, that is code is open for everyone to read.

As one of the reason I really want an open source app to do this is that this kind of app is allowed to access a lot of info. I don't trust any closed source software. I could use them but I don't trust them. Open source is the only way I could trust software. Although open source you could still add malicious code to open source in hope that people wouldn't discover that. But this is the only way. (PS05)

Trust in open source reaches to its highest level when people prefer to write less code and reuse others' code instead.

So my idea is that the least I code the better. As long as [hypothetical app] is still maintained and supported regularly and I do update that regularly. Then I think I will be fine. Because tools that are widely used are very exposed to criticism so their maintainers usually patch up and

correct their mistakes as fast as they can. So I'd very aware of what of dangers of the whole thing. And I would be careful to following news. But I'd avoid writing my own code. (PS15)

is a comment on open source software while the participant was discussing her hypothetical classroom app.

5.5. Discussion

Security Mindsets

Mindsets are likely to influence actions and decision making [196, 201, 361]. We found that most students did not have a clearly developed concept of security. In fact, some participants even struggled to come up with words that could be associated with the term 'computer security'. When it comes to threats such as hackers, what they can do, their intentions and capabilities is another point which needs improvements, we observe the similar patterns and folk models in CS students that others have seen in home users [208, 361].

Mental models could be partially rooted in media [123]; participants cited media plot elements when describing hackers. End-user security has seen success in teaching users to copy existing mental models such as viruses or home safety to better understand and reason about privacy and security [74]. Our results suggest that similar approaches may work in the educational context to improve the mental models of students.

Application Programming Interfaces

When it comes to APIs, our results closely mirror what related work has shown for professional developers. They often use a combination of online resources to learn and use APIs. They prefer to use easier to use resources, and because official documentation is often not easy to use they tend to go for online resources like Stack Overflow [3]. Professional developers (like our student sample) prefer documentation with examples and matching API scenarios [259, 279]. Therefore, API designers are a significant element in secure software development ecosystem particularly industry API designers who have a large impact on developers. By designing usable APIs [135] and easy to understand documentation [279] they can help students and developers learn and use APIs correctly which could result in building secure software.

Division of Labour

Who is in charge of doing security at organisations has long been a problem point with different units often thinking that security is the job of another team [32]. A view shared by several of our participants. Though such a tendency is considered to be a 'key inhibitor toward secure software development practices' [380, p. 164].

In the work place, security auditors are in charge of checking code for issues which developers are then in charge of fixing [335]. However this system has some downsides. First, auditing takes time during which developers work on other projects and lose the working memory they had about particular code segments. And second, fixing the code requires an understanding of the security issue in order to properly address it, and as has been previously shown, developers have difficulty interacting with security technologies like cryptography libraries due to misunderstandings around how cryptography works [106].

In industry [36] it is necessary to create a security culture where basic security is everyone's responsibility and the security team is a component of that culture rather than the only people who 'do' security. In education such a culture might be facilitated by providing student with code samples that are secure by default and by having them use code checking tools in IDEs that check for problems, such as static analysis tools which teach them not only that they should look for these issues, but also how.

Companies with high security standards make security as a commitment, do not satisfy security because of complexity, and they follow strict formal development and testing processes [148]. Universities can benefit these best practices and teach CS students how to become developers that care about privacy and security.

Security as a Requirement

There are several similarities between the students' views and general industry practices. Student developers' treatment of security as an implied requirement is in line with findings that security is often treated as a non-functional feature in agile methods [58, 118], and that the requirement is not explicitly stated [52, 66]. When asked to describe the features of the classroom discussion app, which had been intentionally chosen as an example of a task with implicit privacy and security requirements, many students did not consider privacy and security as an initial priority. For some students, this might be an artefact of their classroom development experience, where they tend to work on well formed projects that are unlikely to have security as an explicit requirement.

Poor and inconsistent understanding of privacy and security among CS students is likely to cause conflicts between real and best practices in the software industry. For example, when choosing a framework developers do not consider security

as a deciding factor which contradicts secure development best practices [35]. In alignment with other aspects of software development, there is a need to synchronise the development approaches taught in the classroom with those used by industry. That synchronisation needs to occur in both directions such that students are taught industry best standards which they are then able to apply.

Internships

Internships are a way to engage students in the topic as well as prepare them for future careers [63, 81]. Although they require investments from industry [158] we believe that the shortage of privacy and security professionals [124] cannot be solved without involving every player. Hence, we encourage industry to offer more internships to CS students in privacy and security fields to improve the number of students graduating with that type of experience.

5.6. Limitations

Our population includes only students at a single Russell Group university in the UK. Even though our sample was diverse, it was not balanced for gender or security experience. Moreover, only two of our participants were native speakers of English, and we might have obtained more finely differentiated views and opinions if we had been able to interview each participant in their native language. Since we conducted the study during summer vacation time, this resulted in a participant pool biased towards Masters and PhD students, since undergraduate students are not normally present at University in the summer months. Possibly some of our potential participants were in their hometown and could not take part in this study.

5.7. Future Work

We plan to expand our study to other universities with a large scale survey to investigate differences and similarities across curriculum, universities and countries. Extending outcomes of this research to industry and professional developers and comparing results is also a path that could lead to valuable insights. Another interesting avenue for future work is to investigate the impact of open source and code reuse in system security. It also remains to question how developers trust in others' code and import code from different resources without knowing their source and coder.

5.8. Conclusion

In this work we reported on a qualitative analysis of twenty semi-structured interviews with CS students. We find that the attitudes of students match many of those observed by other researchers looking at professional level developers. Students have a range of hacker/attack mindsets, lack of experience with security APIs, a mixed view of who is in charge of privacy and security in the software life cycle, and a tendency to trust other peoples' code as a convenient approach to rapidly build software. We further give recommendations for both industry and academia to improve software privacy and security ecosystem.

Acknowledgements

Thanks to all participants for their time and everyone associated with the TULiPS Lab at the University of Edinburgh for helpful discussions and feedback. We also thank the anonymous reviewers whose comments helped improve the paper greatly. This work was sponsored in part by Microsoft Research through its PhD Scholarship Programme.

Back Cover

Computer science students rarely talk about privacy and security features in design of a hypothetical app, while having a background in privacy and security may result in developers becoming privacy champions as described in Chapter 4. This begs the question of does academia provide computer science students the right skills and mindsets for today's software development tasks? I argue including more privacy and professional ethics courses in the computer science could result in having developers who at minimum may consider including some privacy measures in their software development, potentially resulting in building a privacy oriented culture in the software ecosystem. In the next chapters, I study software development platforms, in particular ad networks, and show that how they impact developers' choices and potentially decision making process while trying to do privacy tasks. Results of this chapter shows that providing computer science students with ethics courses may be a good starting point to inform future developers about the ethical consequences of their choices.



6. ‘Developers Are Responsible’: What Ad Networks Tell Developers About Privacy

Front Cover

The Stack Overflow analysis shows that developers are heavily impacted by the software development platforms. Many of their privacy-related questions are influenced by the requirements imposed by the platforms. For example, if Apple introduces a new privacy requirement, developers try to satisfy it. Also, from the computer science students, I found that students rarely talk about privacy features if they are not prompted about privacy. Therefore, I decided to look at a particular type of platform, ad networks, because they have privacy consequences for users and prior research shows that some users find them creepy, annoying, and discriminating [13, 26, 219, 303, 312, 341], to understand how these interfaces can impact developers’ decisions for users’ privacy, and how can we nudge developers about privacy in such platforms.

My co-authors and I did a walkthrough of four popular ad network guidance pages with a senior Android developer by looking at the privacy-related information presented to developers. We found that information is focused on complying with legal regulations, and puts the responsibility for such decisions on the developer. Also, sample code and settings often have privacy-unfriendly defaults laced with dark patterns to nudge developers’ decisions towards privacy-unfriendly options such as sharing sensitive data to increase revenue. We conclude by discussing future research around empowering developers and minimising the negative impacts of dark patterns. This line of work shows the potential impact of platforms on developers’ decision making process and that privacy-related data is not readily available to help them make choices. The work contributes to the field of developer-centred privacy by providing empirical evidence to the research community.

6.1. Introduction

Mobile ads are one of the most popular models of monetising apps [28, 143, 183, 298, 331], about 77% of free Android apps contain an ad library [151, 164]. With about 3 million apps in the Google Play Store alone [239], ad networks collect massive amounts of data about users on a daily basis. Developers who build these apps may be able to decide what to include or exclude in their apps, but these decisions are not always fully informed, and developers tend to pick the default options provided by the ad networks (‘status quo bias’) potentially endangering user privacy [211].

Developers may have privacy concerns for users and look for options that can protect user privacy [108, 190, 324, 330]. For example, when given the option, developers chose coarse over fine grain location information [163]; suggesting that developers do consider privacy. Other developers, though, may just use ‘industry standard’ content provided by large companies which may not be in the best interest of users [108]. Ad networks’ documentation is also one of the primary resources of developers when choosing an ad network [211], making presented information particularly important.

To understand the default ad networks’ configurations and what privacy-related information they provide to developers which can effect developers’ choices and consequently user privacy, we conducted a study with a usable privacy and security researcher and a senior Android developer who reviewed the quick start guides and linked information, of four popular ad networks. We find that most of the privacy information presented is framed around legal compliance, casting developers as the responsible entity—which contradicts developers’ view that ad networks are responsible for user privacy [211]. The information is also provided in a variety of places sometimes on the main path or included in the libraries by default and sometimes linked from hard-to-notice places which makes finding the information highly inconsistent between ad networks.

6.2. Background

Developers and user privacy. Although developers acknowledge the value of user privacy, they find it difficult to understand what information is collected and how it is addressed by platforms [108]. Furthermore, developers’ user privacy attitudes and actions may contradict; while they may say that they care about user privacy, their decisions and final app may not be privacy favourable [211]. Developer concerns about privacy are reflected in questions they ask on Stack Overflow [330], and given the options, they pick more privacy-preserving alternatives [163]. Developers tend to follow guidelines and requirements provided by the platforms [300,

330]. Our study primarily focuses on the available privacy guidelines for developers in ad networks, as one of the main resources for choosing an ad network [211].

Ad networks and Android app development. Android developers must request for permissions from the operating system, and sometimes the user, to access certain resources. These permissions are defined by the developer in a manifest file `AndroidManifest.xml` [97]. Permissions could be ‘normal permissions’ like ‘time zone’ that do not require user consent, or ‘dangerous permissions’ such as access to contacts, location, and read/write messages that requires explicit user consent [263, 313]. Permissions requested by the app are shared within the project, and libraries do not need to ask for second permission from the user or the developer to access shared resources [275, 310]. Third-party libraries not only collect data in free apps but also from paid apps [48, 143], opening up the question of why developers make such choices? Our study expands ad network literature by studying developer-facing privacy information and options in ad networks.

Dark patterns. Dark patterns are ‘instances where designers use their knowledge of human behavior (e.g., psychology) and the desires of users to implement deceptive functionality that is not in the user’s best interest’ [134, p. 1]. Some common dark patterns are (1) *nagging*: when an interface interrupts user workflow consistently and asks for a certain action from them, (2) *preselection*: values set to defaults that are in the best interest of the provider prior to user interaction, (3) *aesthetic manipulation*: graphical elements used to deceive the user into taking an action that may be in favour of service provider rather than the user, (4) *forced action*: users are given only one option to follow even if that is not what they prefer to do, (5) *toying with emotion*: elements, colours, or language to provoke user’s emotion to get user make a decision that is in favour of the service provider, and (6) *false hierarchy*: options that are in the best interest of the service provider are in higher positions [134]. As such patterns become prevalent in the digital world [65, 70, 93, 99, 117, 134, 203, 231, 238, 355], we are keen to explore the presence of them in tools and libraries that developers use, specifically, in ad networks.

6.3. Method

We conducted a walkthrough with two reviewers (similar to a pair-programming activity) of four highly popular Android ad networks [16]: *Google AdMob (GAM)*, *Amazon Mobile ad network (AMN)*, *Facebook Audience Network (FAN)*, and *Twitter MoPub (TMP)*. Reviewers started by searching for ‘[ad library’s name] Android’. on Google which is one of the primary tools developers use to find information [209]. Doing so produced the official guidance on how to integrate the library into an app for all four ad networks. This guidance was then accessed and followed to

integrate interstitial ads into a hypothetical app, including creating an account. While stepping through the guidance, the reviewers noted any material provided that related to privacy as well as any links to other materials that might be privacy-related which were then visited later. The two reviewers discussed all material as they went through it and agreed on the observations. All websites were visited on a Firefox v79.0 with a UK-based IP address in August 2020. In total, we spent 15.5 hours on the four ad networks; 6 hours on GAM, 3 hours on AMN, 2.5 hours on FAN, and 4 hours on TMP. GAM took most of the time since it had the most materials.

Privacy. In advance of the review, the researcher reviewed prior work on how developers think about privacy on Stack Overflow [330], an ad networks study with developers [211], and in the privacy by design framework [157]. Developers tend to associate privacy with permissions, data collection and management, information disclosure, privacy policies, and laws and regulations associated with privacy such as the General Data Protection Regulation (GDPR) [127], California Consumer Privacy Act (CCPA) [72], and Children’s Online Privacy Protection Act (COPPA) [80].

Reviewers. A researcher with four years of experience in usable privacy and security research and four years of experience in software engineering, and a senior software engineer who we will call *Abi* who has a computer science degree and 11 years of experience in Android development, went through all the content. *Abi* has written over 40 apps for corporations that have users from hundreds to millions, creates online Android programming video tutorials, and is fluent in Java. Because he develops apps for corporations, he had not previously worked with ad networks and was, therefore, able to look at the pages with experienced, but fresh, eyes.

Limitations. Both *Abi* and the researcher have extensive experience in their respective fields, reducing the chances of missing the relevant information. However, we note that the results are not generalisable to all developers and ad networks. We focused on the developer-facing information and did not go through the legal documents, terms of services, and privacy policies. We are not aware of studies on developers behaviour when dealing with a privacy policy, but the general public’s attitude towards privacy policies is to skip or spend less than 90 seconds reading them [242]. Notably, we conducted the study during COVID-19 pandemic when many businesses were either closed or doing remote work which may have impacted the resources that ad network companies spend on their documentation, websites, and guidelines.

6.4. Findings

This section includes the information that was found in *guides*, linked-to content (*supplemental documentation*), and in the developer's *dashboard*. Section 6.4.5 consists of dark patterns that the research team found during discussions after reviewing the screenshots taken during the review procedure. Table C.1 in the Appendix provides an overview of the available privacy information and where they are located, Appendix C.1 shows the screenshots.

6.4.1. Google AdMob (GAM)

GAM provides developers with a clear step-by-step guide and also a consistently-visible sidebar with many links to other materials ranging from how to handle custom events to CCPA. We found the documentation easy to navigate with an everything-in-one-place tone to the user interface.

Guide. A step-by-step guide is included in [Get Started](#) page with videos, sample code, and some minimal explanation text. Privacy wise, it has a warning under initialising mobile ads about obtaining consent 'from users in the European Economic Area (EEA)' and directs the developer to set request-specific flags such as 'tagForChildDirectedTreatment' or take other actions before initialising the SDK because it may preload ads. None of the terms in the warning were linked (Figure C.1).

Supplemental Documentation. GAM provides a fair amount of extra privacy-related information, most of which is linked directly off the sidebar. [CCPA Preparation](#) appears just below the 'Get Started' and 'Test Ads' items on the sidebar, so it is relatively prominent. It starts with a link to another set of instructions that explains CCPA and provides guidance on how to restrict data processing via the developer's account page. The CCPA Preparation page itself provides code examples of how to restrict data processing in code via either the Google RDP signal or the IAB ('consortium charged with producing and helping companies implement global industry technical standards' [1]) signal. Notably, the Google option defaults to restricting data processing, but the IAB code example has only a placeholder and requires the developer to open IAB specifications to construct the parameter string. The in-code setting also overrides any setting in the developer's account page, which may be confusing to some.

The [EU Consent](#) option takes the user to the User Messaging Platform SDK which causes all the AdMob branding and sidebar to vanish. The page provides step-by-step instructions on how to use the SDK with many code examples. Notably,

the SDK appears to handle user-facing messaging itself so developers cannot easily change it. The example code also puts user consent in a loop so if the user dismisses the popup, it just reloads (Figure C.2). A comment in the code states: 'Handle dismissal by reloading form'.

The *Precise Location Data Policy* page first links to *Google Publisher Policies* and notes that there are 'notice and consent requirements for publishers who pass users' precise location data to Google, for ads-related purposes'. Then provides sample code which asks the user for consent to use their location. There are several points about this code sample (Figure C.3). First, Abi immediately spotted the 'we may use your location . . . for the purposes of personalized advertising' which is misleading because the location data is definitely being used for this purpose. Second, the popup only provides an 'OK' option. Third, the text provides a URL to 'our' privacy policy that leads to a Chinese Android app market page. Presumably, the developer is meant to link to their own privacy policy which will explain how GAM will use location information. Though, we were unable to find any guidance about what a developer should write into such a policy. Fourth, Abi further pointed out that showing multiple popups that do not even belong to his app will annoy users and is the last thing he would do while building an app.

The sidebar also contains several pages on *Mediation* which is a GAM service that lets developers load ads from other ad networks through GAM. None of the sidebar options are obviously about privacy. However, information about GDPR and other regulations does appear in the instructions under 'Optional steps' for some ad network partners. For example, the guidance for *Facebook Audience Network* discusses GDPR and CCPA but provides no example code. The guidance for both *AdColony* and *AdLovin* tells the developer that they are obliged to get user consent and then provides sample code that indicates that the user has given consent.

Dashboard. During account and app creation procedures, GAM encourages developers to share data with other Google services like Google Analytics and Firebase to 'optimize your app's user experience and your ad revenue'. These items are also preselected to maximum data sharing (Figures C.4, C.5, and C.6). 'Blocking controls' provides several privacy options such as sensitive categories, ad content rating, CCPA, EU user consent, and ad networks (Figure C.8). Sensitive categories are all allowed by default using grey toggle switches (e.g., 'References to Sex' and 'Religion') except for 'Gambling & Betting (18+)' which is blocked using a blue toggle switch. On the content rating page (Figure C.7), the setting is set to 'Mature Audiences' with a bar that allows developers to change the audiences to 'Teens', 'Parental Guidance', and 'General Audiences'. When trying to lower the setting, it provides a red box saying: 'Est. impact of changing MA to PG: -29 to -57% impressions, -31 to -64% revenue . . . '.

Developers further can limit ads personalisation for California and European users (Figure C.9). Defaults for these pages are set to 'Don't restrict data processing', 'Personalised ads', and use all common advertising partners. Under the ad networks page (Figure C.10), a list of partners is shown with over 5,000 partners that GAM shares data with; they are all set to 'allowed' with grey looking toggle switches. The only option that is set to on is 'Automatically allow new Google-certified ad networks' with a blue toggle switch that does not have an 'allowed' or 'blocked' text like others. The funding choices, a service provide by GAM to assist developers in building a consent popup, includes two choice, the first choice does not include a 'Do not consent' button, while the second nearly-identical choice does (Figures C.11 and C.12).

6.4.2. Amazon Mobile Ad Network (AMN)

AMN's top search result contained no step-by-step page and instead directed us to their main [Mobile Ads](#) page which had a 'getting started' section of links. Going through the process of integrating the library involved pages that were filled with links to other pages, which in turn also contained many links. The number of pages necessary could easily be overwhelming to a developer.

Guide. We treat the 'Get started' section on the [Mobile Ads](#) page as the guide. It contained four pages: download SDK, FAQ, account sign up, and publishing apps. To add ads to an app, all the pages except FAQ would need to be gone through, so technically FAQ could be skipped. However, the FAQ was the first thing Abi wanted to read in the hope that it will contain some useful information. He found all the provided links confusing and not related to 'how to add an ad'. Unlike the GAM pages, the AMN pages were very text-heavy and had no clear set of steps for developers to follow. Only the [FAQ](#) page contained any privacy-related information.

The FAQ page had questions on CCPA, GDPR, monetizing EU traffic, managing what ads appear, geolocation from EU, and users' ability to opt-out of tailored ads. COPPA was also briefly mentioned in an answer. AMN does have a [Quick Start Guide](#) linked off of the FAQ page. However, its omission from the main Mobile Ads page and its low position on Google search make it unclear if AMN considers the page a primary entry point for developers. The page is a step-by-step guide to incorporating the API. It also contains information about how to optionally set up both coarse and fine grain location permissions to enable 'relevant targeted ads' and points out that doing so will likely result in higher revenue (Figure C.13).

Supplemental Documentation. AMN locates nearly all their privacy information on the FAQ page and directs developers elsewhere to find general information on the CCPA transparency framework and to find specifics about IAB standards and targeting options. The questions for CCPA says that ‘you can pass us the user choice signal via the instructions below so that we can honor that choice’ and then provides a sample code that sets `us_privacy` to `1---` and says in the comment ‘example privacy string value’. When we looked at the linked IAB documentation (Figure C.14) we realised that ‘-’ means ‘Not Applicable’. The sample code also sets the location tracking on by `enableGeoLocation(true)` (Figure C.15). The GDPR question only asks to set two flags for GDPR purposes without providing any other materials.

Dashboard. AMN provides a minimal set of privacy settings in the account page allowing developers to ‘Block Product Categories’ where all the items are set to on by default (Figure C.16). It also includes an option to ‘Include Ads From 3rd Party Networks’ with a ‘Yes/No’ radio button (default is set to ‘Yes’) without giving a list of partners. These settings are located in a tab bar next to ‘My account’, ‘Tax Identity’, and ‘Company Profile’.

6.4.3. Facebook Audience Network (FAN)

FAN’s guidance was very prescribed with clear step-by-step instructions, lots of screenshots, and example code. Similar to GAM they had a consistent sidebar, but with a deep auto-collapsed hierarchy. So a developer can easily see where they are but might have to expand several times to find specific content.

Guide. FAN provides a [Get Started with Android](#) guide to developers along with a guide to adding interstitial ads. Neither guide provides any privacy information.

Supplemental Documentation. The FAN sidebar has no privacy-related terms visible at the default expansion. The ‘Guides’ sidebar option opens to show options for pages about COPPA and CCPA but not GDPR. The [COPPA](#) page provides guidance on what ‘child directed’ means and what flags to set, though when we looked up the stated flags, they did not exist in the linked API’s documentation. The [CCPA](#) page provides code to use for both manual and library-detected setting of location.

Dashboard. ‘Blocking’ is on the sidebar of ‘Monetisation Manager’ next to pricing and performance. It provides options to block ad categories in sensitive and general categories (Figure C.17) that are all unchecked (allowed) by default (e.g., ‘Associated with violence’, ‘Gambling’, and ‘Mature apps’). It also provides an option to limit data use but it is set to off by default (Figure C.18).

6.4.4. Twitter MoPub (TMP)

Get Started with MoPub provides step-by-step instructions, many of which require a visit another page to complete the step. However, all the pages appear with the same sidebar, and the UI shows where the developer is in the site organisation as well as providing a clear ‘Get Started’ link at the top of the sidebar so they can easily return to the main guide page. Each step also ends with encouraging statements like ‘Terrific: you’ve completed Step 4 of 7’.

Guide. The TMP guide *Get Started with MoPub* has seven steps each of which contain a mix of text and links to other necessary guides, such as guides for integrating MoPub into Android, iOS, and Unity. Many of these are clearly on the critical path for a developer trying to integrate ads, but the guide text also contains recommended steps to do things like ‘refer to our best practices’ with links. The *Integrate the MoPub SDK for Android* warns developers at the top that if they are upgrading they may have to do extra steps for GDPR and links to GDPR guidance which is also linked off the sidebar. The current SDK’s behaviour is to auto-detect the user’s coarse location using the truncated IP address and then automatically asks for consent from EU users without the developer needing to take action, hence the primary guidance does not directly cover GDPR. The sample code provides optional permissions with fine location data collection (Figure C.19).

Supplemental Documentation. The first line in the *GDPR* page, described above, tells developers to read another page first; making it difficult to follow the instructions: ‘Do not start this article until you read our *GDPR Publisher Integration Guide* to understand the flow of events that you will implement below’. Otherwise, TMP provided no other privacy guidance, and terms like CCPA and COPPA are not mentioned in guide pages.

Dashboard. The app application process requires developers to agree to a statement that their app does not target children younger than 13 years old. When clicking on our account name up in the right corner, content blocking shows up next to account settings, and log out. Some items are blocked by default (e.g., ‘Spyware/Malware’, ‘Hate Content’, and ‘Extreme Graphic/Explicit Violence’) and cannot be unblocked (Figure C.20).

6.4.5. Defaults Laced With Dark Patterns

While ad networks make it clear that it is developers' responsibility to make choices and be compliant with the regulations, ad networks make use of range of known dark patterns to nudge developers to make choices that are in the best interest of ad networks.

Developer-facing dark patterns. Ad networks use *toying with emotion* by hinting that developers get higher revenue or better analytics by sharing more data with ad networks and enabling options like higher content ratings (e.g., mature audiences). *Preselection* used by all ad networks: regulation defaults are set to off, data collection is not limited, personalised ads are allowed, user consent is by default set to true in sample code, and content categories are all set to on (except for TMP that has a few categories set to off by default).

GAM uses *aesthetic manipulation* in the content categories UI by having a blue toggle represent blocked items and a grey one for allowed items; TMP also uses a similar pattern with blue for blocked items. GAM uses *false hierarchy* by making the first option on the consent popup builder not include a 'Do not consent' option, while the second nearly-identical choice does. Moreover, privacy information is hard to find in all the ad networks, representing the *hidden information* dark pattern. Abi pointed out that if privacy requirements are buried under sub-pages, advanced options, FAQs, or called 'optional', it is not realistic to expect developers to fulfil those requirements. Privacy options should be part of the workflow, included in the step-by-step ad building guides like the other steps.

User-facing dark patterns Dark patterns in ad networks also target users. Sample code provided by GAM continues to ask for user consent even if they decline it, which is a clear example of *nagging* behaviour. Other examples in GAM include notifying the user about using location without giving them any options to refuse, or providing consent popups that do not have a 'I do not consent' button; both of these instances represent a *forced action* dark pattern that could end up in developers' apps.

6.5. Discussion and Future Work

6.5.1. Drivers of Privacy

The need to comply with *legal requirements* like GDPR and CCPA drove much of the privacy-related content presented by ad networks. The need for *users* to consent to permission usage was also a large driver. Most mobile *operating systems* require

user consent before apps can access specific data and resources, like location. This requirement seems to have compelled ad networks to provide instructions around enabling the permissions and getting the user to consent to their usage. Prior work on privacy-related questions developers ask on Stack Overflow [330] observed developers rarely asking regulation-related privacy questions. However, a fairly large number of questions were related to construction and consequences of accessing resources in privacy policies. The overlap between the two works suggests that ad networks are interested in following regulations by providing flags, settings, and consent examples to developers, while developers are looking for advice on how to write privacy policies that properly express the impacts of including third-party content; information that ad networks do not currently provide.

A potential solution for the challenging privacy tasks may be providing tools to developers which can assist them in making more privacy-friendly decisions by making easy-to-use options available to them [163]. But there are few tools that help developers write things like user-friendly consent popups and accurate privacy policies, and even fewer that take into account both regulations and the current behaviours of common third-party APIs [318, 326], like ad networks. A line of future research would be to look into the practicalities of creating such a tool, potentially learning from usability studies in security APIs [128, 165] and notifications [329].

Ad networks also implied that making the more privacy-friendly choices would negatively impact developers' ability to make money from the ads. Since the only real benefit of adding an ad network to an app is financial, these comments may have an impact on developer decisions. Future research is called to look at the impact of choice framing and *nudging* on developers' decisions, and also the financial impact of such choices. Such studies, if presented in developer-friendly language, would have the potential to allow developers to make more informed trade-off decisions.

6.5.2. 'Developers Are Responsible': Following Regulations Is a Developer Choice and Responsibility

The ad networks' language implied that following regulations was the *developer's responsibility*, not the ad networks', which is in contradiction with what developers think: it is ad networks' responsibility to protect user privacy [211]. While some ad networks provide code samples of how to handle legal requirements, they also take care to emphasise that it is the developer's responsibility to make sure they are complying with regulations appropriately. Abi was continuously confused and frustrated with these requirements as he (the developer) was the one who would be blamed for things like permission requests, even though they were being

requested by the ad networks. For example, AMN provides brief documentation for CCPA and says: ‘We realize that you will determine what the CCPA means for your Amazon Mobile Ads integration’ making the developer responsible but not providing an adequate guide. When Abi saw that he had to set flags (e.g., *Google’s RDP* and *IABUSPrivacy_String*) in his app’s `SharedPreferences` and not in the SDK, he was not sure how these changes might impact his liability, because typically he sets flags in the libraries and not in his app’s shared space and it was odd that the regulation-related flags were located in a different place than the other API flags.

6.5.3. Here Be Dragons: Each Ad Network Has Its Own Unknowns

Ad networks present privacy information in a different location, use different language and options, and have different ways to control privacy options. They also explain how to handle legal requirements differently as well as differing on who is responsible (developer vs ad network) to do checks. For example, GAM provides consent popups but asks the developer to present it to the user, as opposed to TMP’s consent that is handled by TMP. The result for developers is that each ad network is effectively its own uncharted area, requiring a fair bit of time to go through and understand how it handles privacy issues, often also requiring the developer to search beyond what is presented in the quick start guide.

6.5.4. Privacy Run Around

IAB is being used to standardise privacy requirements. However, it is also being used as an information black hole; developers are sent to IAB to find documentation on settings and flags that does not exist. AMN page contained information on CCPA, including links to IAB’s guidance, relevant flags, and a sample code that shows how to set IAB flags by setting CCPA as not applicable and turning on location tracking. The FAQ question on GDPR provides several concrete flags that the developer can set but does not link to documentation on the setting options. We tried looking for guidance on the ‘consent string’ called `IABTCF_TCString` but could not find its correct formatting on AMN or IAB. The FAQ page makes it clear that the IAB consent must be used if the developer wants to make money from EU traffic. GAM also asks developers to use IAB flags, but does not say how. Other black holes that ad networks send developers to visit are main regulation pages under the government sites or privacy policy pages of the parent corporations. Abi was not sure about the usefulness of such links for developers who do not have a legal background and said that he needs a lawyer to understand all these

acronyms, terms, and conditions. Future research is needed to build and evaluate a usable framework for presenting privacy information to developers.

6.5.5. Can Developers Make Informed Choices?

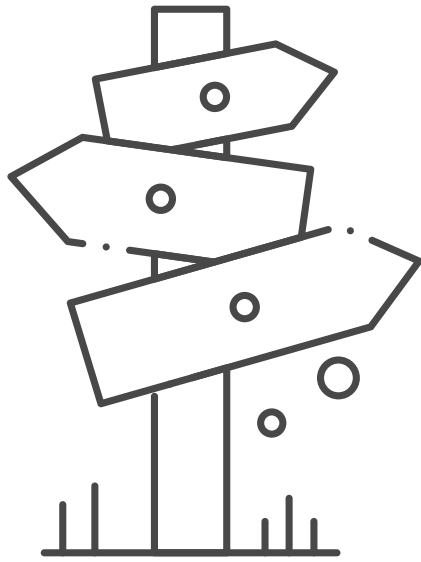
We find that ad networks use dark patterns to nudge developers to choose personalised ads and share more data, much like users resulting in a ‘*control theatre*’ rather than giving developers a chance to make informed choices. We hypothesise that the low rate of GDPR-compliant consent popups in websites [120, 206, 346] and the abundance of non-compliant Android apps [189, 278, 302, 388] may partially be because developers are not making *informed decisions* and are either not aware of the consequences of their choices on users or not aware of how to do a better job than the defaults suggest; hence, not because of their ignorance for user privacy. Opinions about dark patterns are mixed; they are viewed as an ethical issue or a violation of law [355, 358]. A recent study by Norwegian Consumer Council shows that instances of data sharing in ad networks (e.g., TMP) appear to be illegal under GDPR [249]. Future research could look at ways to, at minimum, inform developers about these patterns, and regulators could work towards enforcing regulations beyond satisfying requirements like having a privacy policy or terms of service that only a few people may pay attention to.

Acknowledgements

We thank Yashar PourMohammad for doing the walkthrough with us, and everyone associated with the TULiPS-Lab at the University of Edinburgh for helpful discussions and feedback. This work was sponsored in part by Microsoft Research through its PhD Scholarship Program and a Google Research Award.

Back Cover

The findings of this chapter shows that the presentation of privacy information on ad networks is highly inconsistent, which may result in developers not being able to easily navigate and find privacy-related information. I also find that privacy regulations are addressed differently in these platforms making it challenging for developers to to follow the regulations. Findings of this chapter builds a groundwork for the next chapter (Chapter 7), in which I design an experiment to test various types of choice framing and wording for presenting privacy options to developers and measure the impact of these changes on developers' user privacy decisions.



7. Deciding on Personalised Ads: Nudging Developers About User Privacy

Front Cover

In the previous chapter, I find that ad networks make use of dark patterns to nudge developers into sharing more of their users' data with platforms and privacy-related information is presented in a highly inconsistent way. I decided to run an experiment to understand whether nudging developers into privacy-friendly options would result in developers making more privacy-friendly choices.

My co-authors and I conducted a survey-based online experiment with 400 participants with experience in mobile app development. Across six conditions, we varied the choice framing of options around ad personalisation. Participants in the condition where privacy consequences of ads personalisation are highlighted in the options are significantly (11.06 times) more likely to choose non-personalised ads compared to participants in the Control condition with no information about privacy. Participants' choices of an ad type are driven by impact on revenue, user privacy, and relevance to users. Our findings suggest that developers are impacted by interfaces and need transparent options. This chapter contributes to the developer-centred privacy by showing that developers do need support from platforms and including developers in the design process of these platforms may be an effective way to build usable documentation for developers.

7.1. Introduction

Mobile advertising networks play an intermediary role of matching the advertisers (companies that want to advertise their products) with the publishers (apps that want to generate revenue by hosting advertising). They are a popular monetisation approach [28, 143, 183, 298, 331], with about 77% of free Android apps containing an ad library [151, 164]. To show personalised ads, ad networks collect data from app users, which raises privacy concerns [130, 344, 360]. Targeted ads can also seem intrusive and discriminating to some users [197, 241, 270, 385]. Major operating systems give users an option to limit these ads and associated tracking. However, behavioural research shows that due to status quo bias, people rarely change the default configurations [10, 166, 269, 286], and poor usability makes it hard for users to opt out of behavioural advertising and tracking [139, 182, 283]. Thus, developers' decisions regarding the defaults for their apps have implications for user privacy. Specifically, when configuring ad networks, developers can choose in the developer dashboard between personalised and non-personalised ads. Here again, status quo bias may not play out in favour of user privacy: if ad networks set personalised ads that imply more extensive personal data collection as default choices, it might nudge developers to stick to those privacy-unfriendly defaults [108, 211].

With about 24 million software developers (estimated to go up to 28.7 million by 2024) [240], who are in charge of building apps for personal smart devices, cars, and large industries, it is essential to understand how services they use impact their decisions. Indeed, studies of privacy-related questions on Stack Overflow [330] and Reddit Android forums [186] show that developers' privacy concerns are heavily driven by large platforms such as Google and Apple. Moreover, there is a growing use of dark patterns that persuade users into make decisions that are in favour of platforms; for example, by using preselected default options, or sneaking a small product or service into users shopping basket without informing users, such as adding travel insurance during the plane ticket purchasing [134, 203, 231]. The use of dark patterns in the context of software development may have negative implications for users, as developers' choices will effect all users of their apps. For example, collecting location data, showing unrestricted ads categories, and displaying personalised ads are often allowed by default in popular ad networks [211, 327].

Similarly, given that ads tailored to users' preferences have a higher value [200], ad networks have incentive to nudge developers into choosing personalised ads over non-personalised ones, without necessarily acknowledging the trade-offs between revenue, user privacy and experience. In addition to status quo bias leveraged by default choices, salience effect can be leveraged to further facilitate the nudging [46, 293]. For example, while an emphasis on user privacy may steer developers' decisions towards non-personalised ads, an emphasis on potentially larger revenue may nudge developers to choose personalised ads which is used by

some ad networks through including statements like ‘including personalised ads may likely result in higher revenue’ in their documentation, quick start guides, and blog posts [327, 332].

In this study, we aim to understand how choice framing in ad networks affect developers’ decision making. Our research question are:

RQ1: How does choice framing in ad networks impact developers’ decisions about ad personalisation?

RQ2: What are the reasons behind developers’ choices of personalised or non-personalised ads?

To answer our research questions, we conducted an online survey-based experiment with 400 participants with app development experience. In a hypothetical scenario, we asked them to make a series of choices to integrate ads in a personal finance management app and a gaming app. The main decision of interest was regarding the choice between personalised and non-personalised ads. The framing of those choices was manipulated between one control and five experimental conditions to emphasise implications for framing around data processing restrictions, user-facing descriptions, user privacy, developer’s revenue, and both user privacy and developer’s revenue. To help further contextualise and interpret the results, we also surveyed participants’ opinions and attitudes about personalised ads, ad networks, and privacy regulations.

We find that although on average the majority of participants decided to integrate the personalised ads, choice framing significantly impacted their decisions. When user privacy implications were made salient, participants were 11.06 times more likely to select non-personalised ads than when the neutral framing was used (control condition). When a framing emphasised data processing restrictions, participants were 3.45 more likely to select the non-personalised ads than in the control condition. Other nudges—emphasising the consequences of ads on an app’s revenue, presenting participants with an explicit choice between user privacy and app’s revenue, and telling participants that users will be able to see whether the app is using ads based on their personal data or not—did not significantly changed participants decisions compared to the control condition.

The analysis of open-ended responses revealed a variety of reasons for developers’ choices, ranging from maximising the app’s revenue and relevance of ads to the uses, to concerns about user privacy and regulation compliance, and implications for user experience. From the exit survey, we found that even when upper- and middle- management choose the ad networks and app’s business models, developers still feel involved in this decision-making process. However, developers generally believe that they do not have full control over ad networks’ data collection, and believe users have even less control. By illustrating the potential impact of choice framing on ad personalisation decisions during app development, our

results inform regulators about the need to enforce greater control over ad networks' data collection and analysis practices, discourage the use of dark patterns, and encourage ad networks to adopt interfaces for developers that may assist them in making informed decisions about user privacy.

7.2. Related Work

Ad Networks

Ad networks are a popular mobile app monetisation approach [28, 143, 183, 298, 331]. Over half of Android apps include ad network libraries [28, 151, 164, 331], which often offer both personalised and non-personalised ads. Personalised ads attract more user attention than non-personalised ads [50, 197], generating higher engagement and therefore revenue. To provide ads tailored to a specific user, ad networks collect personal information from users such as age, gender, and location [249, 316], not only in free apps that rely mostly on ads to generate revenue, but also in paid apps [48, 143]. However, personalised ads have some negative consequences for users. For example, some users find them discomfoting [197, 385], discriminating [266], and intrusive [241, 270].

Options Provided by Ad Networks to Users and Developers

Both users and developers can limit data collection and turn off ad personalisation. After the introduction of the General Data Protection Regulation (GDPR) [127] and the California Consumer Privacy Act (CCPA) [72], the prevalence of these options particularly increased [154].

On the user side, self-regulatory programs (e.g., Digital Advertising Alliance opt-out [100]), smartphone operating systems, service providers, and browsers offer settings that allow opting out of ad personalisation [205], and at minimum, request user consent to show personalised ads. Research shows limited effectiveness, usefulness, legal compliance [120, 140, 206, 346], and usability [206, 238] of these methods.

On the developer side, ad networks provide an interface for configuring personalisation and data collection for specific apps and geographic regions. These interfaces often use defaults that are not in favour of user privacy [211, 327]. Developers tend to keep the defaults, follow industry standards, guidelines, and requirements provided by the platforms built by large tech companies [135, 186, 300, 330] without fully considering all the options and consequences of their choices on user privacy [87, 108, 211]. Developers generally acknowledge the value of user privacy [108, 211, 299], but find it challenging to understand what information

is collected, how it is used by platforms [108, 211], and how to protect user privacy [186, 330]. Hence, some poor user privacy elements in how apps integrate ad networks may be caused by the way ad networks are framing choices and nudging developers through defaults.

Nudging

Humans can be nudged towards making certain actions through the use of specific wordings, framing, colours, and default values [10, 76]. *Choice framing*, in particular, uses the activation of salience effects [46, 293] and status quo bias [166, 269, 286], to effectively nudge the privacy choices of users [10, 44]. For example, priming survey respondents about privacy using words like ‘privacy-sensitive’ and ‘potential privacy risks’ increases the reported privacy concerns [68] and making privacy information salient drives more privacy-preserving choices in user experiments [340]. We believe that similar effects can be achieved in the context of software development, where choice framing in tools and interfaces may affect developers’ decision making.

Nudges can be used to encourage users to make decisions that are favourable to service providers (e.g., ad networks) but not necessarily favourable to themselves. Such practices are often referred to *dark patterns*—‘instances where designers use their knowledge of human behavior (e.g., psychology) and the desires of users to implement deceptive functionality that is not in the user’s best interest’ [134, p. 1]. In the context of privacy, the examples of dark patterns include privacy consent forms that do not provide a ‘reject all’ button [238] and hard-to-find (or completely absent) options for deleting accounts [65]. Similar patterns are also visible in ad networks’ developer dashboards where the default values are all set to personalised ads and location data is often collected by default [211, 327].

Our Contribution. We extend the literature on developer-facing privacy interfaces by looking at the privacy nudges directed at developers and exploring the impact of choice framing in ad networks’ developer dashboards.

7.3. Method

To answer our research questions, we conducted an online survey-based between-subject experiment with 400 participants with mobile development experience administered using Qualtrics. The study received ethical approval from our institute. All participants provided informed consent before completing the study. We describe the study protocol below, and the full survey text is in Appendix D.1.2.

After screening for app development experience (Section 7.3.2), participants were randomly assigned to one of six conditions (Section 7.3.1), and asked to complete the main survey. Each participant was presented with two hypothetical scenarios in a random order: one was about a *gaming app*, another one was about a *financial app* for personal finance management. We chose these app categories, because personal finance management has obvious privacy implications (e.g., developers reported more sensitive variables for the financial category compared to other app categories [45]), and gaming is the most popular category on both Apple App Store and Google Play [220, 221].

Participants were asked to imagine that they were the default a shareholder in a software development company, and together with a small team, they created a (financial or gaming) app, which will be published in Europe and the United States and is mainly targeted towards adults above the age of 18. Then, we asked them to answer questions posed by the ‘Acme Assistant’, a tool for an imaginary ad network that helps with integrating the ad network into the app. The Assistant was inspired by MoPub Integration Suite, a new service by Twitter’s MoPub ad network for an easy app integration [218]. The Assistant asked five multiple-choice questions about ad formats (e.g., banner and interstitial), level of graphics (high-quality and moderate-quality), platforms (e.g., Android and iOS), types of ads (personalised and non-personalised), and the regulations that apply to the app (e.g., GDPR, CCPA). After making the choices, they were also asked an open-ended question about the primary reason for choosing the personalised or non-personalised ad type.

After completing the above for both the financial and gaming apps, they were sent to an exit survey with question about: how they would go about asking for user consent for the personalised ads, how the choice of ad type would affect an app’s revenue or number of users, what role does user privacy play in their daily development routines, and how much users and developers have control over data collected by ad networks. The exit survey provided additional insights about participants’ opinions, knowledge, and attitudes, and helped to further contextualise and interpret experimental results. Finally, they answered software and mobile development, and demographics questions.

7.3.1. Experimental Conditions

All participants were randomly assigned to one of six conditions including one Control group and five treatment groups. The only difference among the conditions was the framing of the choice about personalised or non-personalised ads. The order of all options was randomised. Each choice consisted of a short label phrase followed by a longer description.

Control–Minimal Information ($N = 66$): (1) *Personalised ads: Acme can show personalised ads to your users.* (2) *Non-personalised ads: Acme will show only non-personalised ads to your users.* This framing was inspired by Google AdMob’s developer dashboard to help developers build GDPR-compliant apps for European users (Figure D.1 in the Appendix). It used neutral wording about ad types without mentioning any information about collection and processing of user data.

Data Processing Restrictions ($N = 67$): (1) *Ads with unrestricted data processing: Acme can show personalised ads to your users based on a user’s past behaviour, such as previous visits to sites or apps or where the user has been.* (2) *Ads with restricted data processing: Acme will show only non-personalised ads to your users based on contextual information, such as the content of your site or app, restricting the use of certain unique identifiers and other data.* This framing was inspired by Google AdMob’s developer dashboard to help developers build CCPA-compliant apps for California users (Figure D.2 in the Appendix) and it explicitly hinted at the types of data used for ad personalisation, which may indirectly encouraged developers to consider privacy implications of such data processing. We based two of our conditions on Google AdMob because it is the most common mobile ad network in apps [16, 17, 129].

User-Facing Descriptions ($N = 68$): (1) *Ads with ‘Personalised Ads’ tag displayed to users: Acme can show personalised ads to your users. Users will see the ‘Personalised Ads’ tag next to the ‘Install’ button and the following text in your app description in the App Store or Google play ‘This app shows ads personalised based on your personal information’.* (2) *Ads with ‘Non-personalised Ads’ tag displayed to users: Acme will show only non-personalised ads to your users. Users will see the ‘Non-personalised Ads’ tag next to the ‘Install’ button and the following text in your app description in the App Store or Google play ‘This app shows ads not personalised based on your personal information’.* This condition aimed at leveraging transparency and nudging developers’ accountability and responsibility to users. The framing was inspired by the recent additions to the Apple App Store called ‘Privacy Details’ to ‘help users better understand an app’s privacy practices before they download the app on any Apple platform’ [29] and prior work’s recommendation about including privacy features of apps in the app stores to softly nudge developers to consider user privacy in their apps [186].

Privacy Focused ($N = 67$): (1) *Ads with lower user privacy: Acme can show personalised ads to your users based on their past behaviour, such as previous visits to sites or apps or where the user has been.* (2) *Ads with higher user privacy: Acme will show only non-personalised ads to your users based on contextual information, such as the content of your site or app.* This condition is aimed at leveraging salience effects [46, 293], by making privacy implications prominent in the choice option descriptions.

Revenue Focused ($N = 65$): (1) *Ads with higher revenue: Acme can show personalised ads to your users, which may yield higher revenue than non-personalised ads.* (2) *Ads with lower revenue: Acme will show only non-personalised ads to your users, which may*

yield lower revenue than personalised ads. This condition aimed at leveraging salience effects [46, 293], by making revenue implications prominent in the choice option descriptions.

Privacy vs Revenue ($N = 67$): (1) *Ads with higher revenue: Acme can show personalised ads to your users, which may yield higher revenue than non-personalised ads.* (2) *Ads with higher user privacy: Acme will show only non-personalised ads to your users which may increase your users' privacy.* This condition aimed at exploring what choices the participants would make if they were faced with an explicit trade-off between the user privacy and revenue.

7.3.2. Recruitment and Screening

In January 2021, we used Prolific, GitHub, and LinkedIn groups to recruit the participants. On average, the survey took 19 minutes ($SD = 89$) to complete.

Prolific. Using Prolific's exclusion criteria, we recruited 1,288 participants who were fluent in English, had computer programming skills, and an approval rate of at least 90%. They responded to a 1-minute screening survey (Appendix D.1.1) to assess their software development experience, and received £0.15 compensation. Those who worked on at least one app in the past three years ($N = 466$) were invited to the main survey and were paid £1.50 for completing it. Of the invited participants, 372 respondents started the main survey, but eight did not complete it. We removed two respondents because they had worked on over eighty apps while having less than three years of mobile development experience, one respondent who finished the survey in less than three minutes, and one respondent who did not pass the attention check question. In total, we received 328 valid responses from Prolific.

GitHub. We sent emails to GitHub users who contributed to the top 1,000 GitHub repositories (sorted by the number of stars) written either in (1) Java (with 'Android' as an additional keyword), or (2) Objective-C or Swift (with 'iOS' as an additional keyword). In total, we sent out 33,675 emails, out of which 128 started the survey, 51 respondents did not finish the survey, and five had not developed apps in the past three years. Other checks did not result in removing any additional responses. In total, we received 72 valid responses from GitHub emails. These participants were offered to provide an email to enter into a raffle for a £30 gift card for each 20 participants; 57 participants decided to enter the raffle, out of which three random participants received a gift card.

Other Channels. We made an effort to recruit women and minority groups by posting the survey in 20 LinkedIn groups specific to these populations. 14 respondents started the survey, seven did not finish the survey, and the other seven had not worked on any apps in the past three years. Therefore, we did not receive any valid responses from these channels.

The anonymised dataset for multiple-choice responses, excluding the open-ended responses (per participant consent), for the 400 valid participants is available online at DOI: [10.7488/ds/3045](https://doi.org/10.7488/ds/3045).

7.3.3. Data Analysis

Quantitative Analysis

We fitted a generalised linear mixed model with the binary value of choice between personalised (coded as 0) and non-personalised ads (coded as 1) as the dependent variable. The model consisted of the six conditions (with Control as the baseline), app category (with gaming as the baseline), and several demographics as fixed effects, and participants as random effects, given that we had two data points per participant (gaming and financial apps) [217].

Qualitative Analysis

The count of words in the three open-ended questions showed that the answers were brief (on average 20 words, $SD = 16$) and enabled us to use affinity diagrams to analyse them [67, 180]. We used the virtual collaboration platform Miro [216] to create separate boards for each open-ended question and posted virtual sticky notes with participants' responses. During a half-day virtual session with five security and privacy researchers with a minimum Master's degree in computer science, and one senior Android developer, we identified the common themes through group affinity diagram building.

7.3.4. Limitations

As with any self-reported data, respondents' survey answers may be subject to social desirability bias [116] and may differ from actual behaviours (so called, privacy paradox [173]). However, our use of role-playing scenarios and questions about intentions (rather than only attitudes) partially mitigates these biases, as intentions are shown to significantly correlate with behaviours [18, 107]. Our work complements and extends other privacy-related studies with developers [186, 324, 330] by conducting a controlled study with high internal validity which provides

a foundation for future validation work. The results show a promising effect which will need further field experiments to fully test the generalisability.

Compared to other studies using similar recruitment strategies, the response rate for GitHub emails in our study is 0.21%, which is similar to 0.31% in [329] and lower than 1.3% in [6]. However, we were able to recruit a sufficient number of participants through Prolific. Moreover, mentioning ad networks in the recruitment email could deter people concerned about user privacy or ad networks. However, our results do not support that worry, demonstrating a wide variety of opinions about ad networks and user privacy.

Due to the demographic composition of the Prolific participant pool [111], our sample is predominately European, which could result in participants being more aware of European privacy laws, i.e. GDPR. However, GDPR's jurisdiction applies worldwide and many developers create apps for different geographic markets, mitigating this concern. To geographically balance our sample, we used additional Prolific screening criteria to exclude European countries for 274 respondents of the screening survey. The diverse geographic background of GitHub participants also added diversity to our sample. While our results may not be generalisable to all populations, it provides useful insights on the impact of various nudges on developers' decisions. Future research is encouraged to validate the results with other populations.

7.4. Results

We first report participants' demographics in Section 7.4.1, then the main experimental effects in Section 7.4.2 and Section 7.4.3, and finally the additional findings about participants' opinions and attitudes about ads personalisation in Section 7.4.4 to contextualise and interpret the main results.

7.4.1. Participants

Our participants are mostly European (66%), male (82%), have on average 5.1 years of experience in software development ($SD = 5.3$), 2.7 years of experience in mobile development ($SD = 2.6$), on average worked on 3.5 apps in the past three years ($SD = 4.2$), 73% work in software teams (e.g., developer, tester, or manager), and 46% hold a software development position.¹ Table D.1 in the appendices further summarise participants' demographics.

¹Over 90% of Google Play developers have one to nine apps under their account (as of 2015) [359], suggesting that our sample represents a portion of mobile developers. More than half (57%) of participants have used at least one ad network in their apps. Google AdMob (48%), Facebook Audience Network (20%), and Unity Ads (20%) were the most popular ad networks.

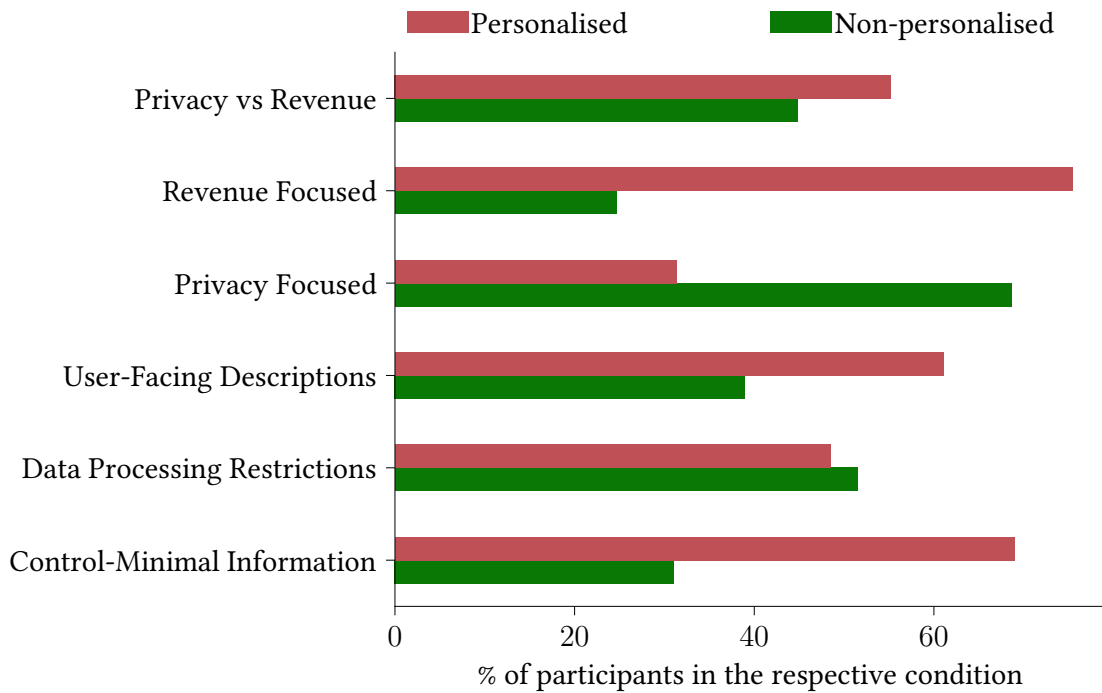


Figure 7.1.: Participants' choices between personalised and non-personalised ads across the six conditions.

7.4.2. Choices Between Personalised and Non-Personalised Ads

As shown in Figure 7.1, the majority of participants chose personalised ads in the Revenue Focused (75%), Control (69%), and User-Facing Description (61%) conditions, and non-personalised ads in the Privacy Focused condition (69%). In the Data Processing Restrictions and Privacy vs Revenue conditions, the choices between the two types of ads were split almost equally, with 49% and 55% respectively choosing the personalised ads.

The regression analysis (Table 7.1) confirms that the choice framing does impact participants' choices (RQ1). The strongest effect was in the Privacy Focused condition: using framing that explicitly mentions the implication for user privacy and what data will be used nudged participants to be 11.06 times ($p < .001$) more likely to choose non-personalised ads over personalised ads, compared to the Control condition. In the Data Processing Restrictions condition, framing that emphasised data restrictions associated with the choice of ads nudges participants to be 3.45 times ($p = .011$) more likely to choose the non-personalised ads compared to the Control condition. The results in the Revenue Focused, User-Facing Descriptions, and Privacy vs Revenue conditions were not significantly different from the Control condition. In other words, using the neutral framing about personalised and non-personalised ads (Control condition), emphasising the consequences of personalised ads on app's revenue (Revenue Focused condition), leveraging the user-facing description to provide transparency to users about whether app uses personalised ads based on users' personal data or not (User-

Facing Description condition), and providing an explicit choice between user privacy and app's revenue (Privacy vs Revenue) similarly affect participants' choices to integrate predominantly personalised ads in the apps.

Table 7.1.: Generalised linear mixed model regression. Outcome variable is the binary choice between personalised (coded as 0) and non-personalised ads (coded as 1). OR: odds ratios, CI: confidence intervals, conditional R^2 : .614 (represents how much of the variance is explained by the model [195]), No. observations: 800.

Independent Variables	ORs	CI (95%)	<i>p</i> -value
Condition			
Control–Minimal Information		Reference	
Data Processing Restrictions	3.45	1.32–8.98	.011
User-Facing Descriptions	1.38	0.54–3.50	.502
Privacy Focused	11.06	3.97–30.75	<.001
Revenue Focused	0.50	0.19–1.33	.164
Privacy vs Revenue	2.48	0.97–6.35	.058
App Category			
Gaming app		Reference	
Financial app	1.02	0.70–1.49	.923
Given Priority to Privacy in Development Routines			
Low priority		Reference	
Not a priority	1.27	0.11–15.04	.851
Medium priority	1.84	0.75–4.51	.184
High priority	3.94	1.59–9.75	.003
Essential	10.33	3.43–31.11	<.001
Main Income Source			
Salary, not dependent on app revenue		Reference	
Don't make money from app development	2.63	1.23–5.66	.013
Salary, partially dependent on app revenue	0.57	0.27–1.17	.126
Direct app revenue	0.73	0.32–1.66	.447
Other	0.90	0.07–11.17	.934
Years of experience in software development	1.08	1.02–1.14	.007
Number of developed apps in the past three years	0.92	0.86–0.99	.033
(Intercept)	0.09	0.03–0.3	<.001

Impact of App Category: Financial vs Gaming

Participants' choices between the app categories were not different; 57% of participants chose personalised ads in both categories. Thus, our expectation that the financial app would trigger more privacy-preserving choices (non-personalised ads) because it carries obvious privacy risks for users is not supported by the data. In Section 7.5.3, we explore the potential reasons behind this effect based on participants' open-ended answers.

Impact of Demographics

We also included the demographic variables in the model that improved the model's fit. We found that participants, who consider privacy an essential or high priority are 10.33 ($p < .001$) and 3.94 times ($p = .003$), respectively, more likely to choose non-personalised ads compared to those who consider privacy a low priority in daily development routines (we selected the low priority as the reference category here because the not a priority category only had five responses making the category sizes highly unbalanced). Participants, who do not make money from software or apps, are 2.63 times ($p = .013$) more likely to choose the non-personalised ads compared to those whose income is from software/app development but is not dependant on app revenue.

Each additional year of experience in software development increases the likelihood of choosing non-personalised ads by 8% ($p < .001$), but each additional app that participants developed in the past three years decreases the odds of choosing the non-personalised ads by 8% ($p = .033$).

The inverse relation between the number of developed apps and the choice of non-personalised ads may be related to the participants getting used to the status quo in that area as they develop more apps. More years of experience may also increase developers' awareness about other app monetisation methods. Inclusion of other variables, such as years of experience in mobile development, did not improve the model fit, thus we did not include them in the final model.

7.4.3. Reasons Behind the Ad Type Choices

Using affinity diagrams, as discussed in Section 7.3.3, we constructed themes around participants' responses to the question: 'What was the biggest reason that made you pick the ad type: [their choice]' (RQ2). Table 7.2 shows the resulting themes. We provide the unique count of participants that mention each theme at all (out of 400) as well as the number of responses that mention a theme (out of 800) as each participant provided a response for each of the two apps. Quotes are labelled with P or NP based on the participant's choice for personalised or

7. Nudging Developers About User Privacy

non-personalised ads. Theme frequencies are provided to give a sense of scale, but should not be used for generalisation since they only measure what participants thought to mention. Statistics are also not used in this section for the same reason.

Table 7.2.: Constructed themes from participants' answers about the primary reason for choosing the ad type.

Theme	Condition (participants, N = 400)						Ad Type Choices (occurrences, N = 800)			
	Control	Data Processing Restrictions	User-Facing Descriptions	Privacy Focused	Revenue Focused	Privacy vs Revenue	Total	Personalised	Non-Personalised	Total
Impact on revenue	32 (8.0%)	16 (4.0%)	29 (7.2%)	18 (4.5%)	46 (11.5%)	25 (6.2%)	166 (41.5%)	232 (29.0%)	24 (3.0%)	256 (32.0%)
User privacy	13 (3.2%)	34 (8.5%)	23 (5.8%)	48 (12.0%)	11 (2.8%)	32 (8.0%)	161 (40.2%)	24 (3.0%)	269 (33.6%)	293 (36.6%)
Sensitive data	1 (0.2%)	9 (2.2%)	4 (1.0%)	11 (2.8%)	1 (0.2%)	6 (1.5%)	32 (8.0%)	-	35 (4.4%)	35 (4.4%)
User trust	2 (0.5%)	6 (1.5%)	5 (1.2%)	3 (0.8%)	7 (1.8%)	7 (1.8%)	30 (7.5%)	5 (0.6%)	40 (5.0%)	45 (5.6%)
Compliance	-	6 (1.5%)	1 (0.2%)	2 (0.5%)	2 (0.5%)	1 (0.2%)	12 (3.0%)	3 (0.4%)	13 (1.6%)	16 (2.0%)
Competitive advantage	-	3 (0.8%)	-	3 (0.8%)	-	4 (1.0%)	10 (2.5%)	-	12 (1.5%)	12 (1.5%)
Users don't care about privacy	-	-	-	6 (1.5%)	-	1 (0.2%)	7 (1.8%)	7 (0.9%)	-	7 (0.9%)
Security reasons	1 (0.2%)	2 (0.5%)	1 (0.2%)	2 (0.5%)	-	1 (0.2%)	7 (1.8%)	1 (0.1%)	8 (1.0%)	9 (1.1%)
Privacy & ethics trade-off	-	-	1 (0.2%)	3 (0.8%)	1 (0.2%)	1 (0.2%)	6 (1.5%)	4 (0.5%)	4 (0.5%)	8 (1.0%)
Relevance to users	33 (8.2%)	26 (6.5%)	33 (8.2%)	11 (2.8%)	30 (7.5%)	23 (5.8%)	156 (39.0%)	197 (24.6%)	29 (3.6%)	226 (28.2%)
User experience	8 (2.0%)	9 (2.2%)	17 (4.2%)	12 (3.0%)	11 (2.8%)	3 (0.8%)	60 (15.0%)	48 (6.0%)	27 (3.4%)	75 (9.4%)
Category-related reasons	7 (1.8%)	9 (2.2%)	6 (1.5%)	18 (4.5%)	5 (1.2%)	15 (3.8%)	60 (15.0%)	18 (2.2%)	89 (11.1%)	107 (13.4%)
Finance-related	3 (0.8%)	9 (2.2%)	4 (1.0%)	13 (3.2%)	5 (1.2%)	8 (2.0%)	42 (10.5%)	7 (0.9%)	72 (9.0%)	79 (9.9%)
Gaming-related	3 (0.8%)	2 (0.5%)	3 (0.8%)	7 (1.8%)	-	8 (2.0%)	23 (5.8%)	10 (1.2%)	20 (2.5%)	30 (3.8%)
Specificity of a target audience	2 (0.5%)	-	1 (0.2%)	5 (1.2%)	4 (1.0%)	5 (1.2%)	17 (4.2%)	10 (1.2%)	10 (1.2%)	20 (2.5%)
Users should decide	2 (0.5%)	2 (0.5%)	5 (1.2%)	4 (1.0%)	2 (0.5%)	2 (0.5%)	17 (4.2%)	18 (2.2%)	8 (1.0%)	26 (3.2%)
Easier to develop	-	3 (0.8%)	1 (0.2%)	-	4 (1.0%)	-	8 (2.0%)	1 (0.1%)	9 (1.1%)	10 (1.2%)
Everyone does it	-	1 (0.2%)	1 (0.2%)	3 (0.8%)	1 (0.2%)	1 (0.2%)	7 (1.8%)	7 (0.9%)	-	7 (0.9%)
Unclear responses	5 (1.2%)	4 (1.0%)	2 (0.5%)	4 (1.0%)	1 (0.2%)	3 (0.8%)	19 (4.8%)	15 (1.9%)	11 (1.4%)	26 (3.2%)

We identified three major reasons for choosing personalised or non-personalised ads: expected impact on revenue, user privacy, and relevance to users. Participants in the Privacy Focused condition mentioned privacy most often, and participants in the Revenue Focused condition mentioned monetisation most often as a reason for their ads choices.

Impact on Revenue

A main reason for choosing a certain ad type was related to monetisation goals and impact on revenue, mentioned by 41.5% of participants (166/400). Those, who chose personalised ads, were especially likely to relate their choice to expected positive impact on revenue (232/800): 'To ensure most people click on the ad, increasing the apps revenue' (P309). Less often participants chose non-personalised ads with the expectations of positive impact on revenue (24/800): 'I believe that providing non-customized ads would help to increase consumption regardless of the type of ad' (NP68).

User Privacy

Out of participants who chose non-personalised ads, most did it because of user privacy (269/800), for example, to protect users' sensitive data (35/800), gain their trust (40/800), comply with privacy regulations (13/800), or gain a competitive advantage (12/800): 'App doesn't have personalized information about the user.

Also, it is easier to comply with GDPR rules that way' (NP213), 'Given Apple's latest privacy changes, users are more aware of apps that invade their privacy and as a result, could be less likely to download these apps' (NP224). Some mentioned the long-term benefits of user trust over the short-term gains from violating user privacy:

Users trust in protecting the privacy is the most valuable good for a developer (besides quality of content). Aiming at a one-hit-wonder one wouldn't care about it, but with long time plans this is the only manageable compromise for all stakeholders (NP135).

Participants, who mentioned privacy in relation to their choice of personalised ads (24/800), mostly assumed that users do not care about privacy (7/800): 'Just like it is with facebook and odther [sic.] big ad circulators, It's proven that people only care about their privacy on a surface level' (P202).

Several participants acknowledged the trade-off between user privacy, trust, and other considerations such as revenue (6/400):

I was torn. On the one hand, personalized ads in the context of ones finances are going to have a *much* higher CPM and I would like to capitalize on that. However, because I'm running an app whose data is sensitive and where I am more dependent on long term trust from my users, I decided to make the ads less personalized to start so that I can have fewer scary disclosures and consent screens. If the app is successful, I can always explore personalizing them later (NP197).

Participants also expressed struggling with the trade-off between revenue and user privacy: 'Desire to protect customers privacy. This was a tough one and I waffled back and forth. If it offered higher payout I would have selected this option' (NP317).

Only seven participants mentioned the potential security risks associated with personalised ads: 'This type of app wants to give the user a sense of security so personalised ads might put someone off from using this app to manage their finances' (NP473).

Relevance to Users

Many participants believe that ads should be interesting, relevant, engaging, and useful to the users (156/400). On the one hand, they believe that such ads are beneficial to the users: 'Personalised ads are appealing to the user, a person interested in a specific topic would rather see/read more about it than a random ad' (P169). Given that personalised ads are targeted to users' potential interests, most participants driven by that reason selected the personalised ads, than non-personalised ones (197/800 vs 29/800). A smaller group of participants chose

non-personalised ads because they considered them relevant to users: ‘Using non-personalised ads, you have the luxury of inserting different ads of which some may get the attention of the users further increasing the interaction’ (NP163). Some participants were even worried that relevant ads may distract users’ attention away from the app, driving the engagement down: ‘. . . you would get distracted if you saw a product that you like, the user could easily close the app and search that product’ (NP42).

Participants in the Privacy Focused condition were least likely to mention the relevance of ads to the users (11/400), but we did not observe much difference among the other conditions (23–33/400).

User Experience

Some participants (60/400) mentioned the impact of ads on user experience as a reason for their choice. In contrast to the theme about relevance of ads emphasising their utility and benefits to the users, this theme emphasises the emotional and experiential impact of ads.

Participants who chose personalised ads (48/800) thought that they are less annoying, more enjoyable, and of higher quality: ‘To avoid frustrating customers with irrelevant to their interests ads that they will be forced to watch throw [sic.] to play the game for free personalized ads are a great choice to make fun the rewarded video ad format’ (P493), ‘. . . I would like the ads to feel native to the app so it is a more professional experience for the user and as such high quality and personalized ads would fit better for such an app’ (P333). Participants, who chose non-personalised ads (27/800) believed that they are less invasive and creepy: ‘I feel that personalized ads are too intrusive and creepy, so I would rather opt for non-personalized ads. . . . I don’t want to scare away users’ (NP330). Some participants preferred to reduce the number of ads in general to minimise the interruption of the main interaction with the app, especially in the gaming context: ‘Gaming isn’t a prime state to be in to think about purchases. As someone with experience, ads feel like a break in action in games and I would say its not worth the extra money overall’ (NP396).

Category-Related

Some participants said their choice of ad type partially depends on the app category, the data it collects, or the specific user audience it targets (60/400). For instance, we already discussed earlier that perceived sensitivity of user data may raise privacy and trust concerns, especially in the context of a financial app, leading participants to choose non-personalised ads: ‘We’re building a financial app after all. The data in there is sensitive and if there have to be ads, they should

in no way track the user. Otherwise we'll lose trust faster than we can build the app' (NP136). Similarly, some participants thought that the data collected in the gaming app is not sensitive, justifying the use of personalised ads: 'The information shared with a gaming type of application may be not as important to the consumer' (P301). Others thought that the data collected in the gaming app does not reveal personal information, and thus cannot be used for targeting, leading to the choice of non-personalised ads: 'A Gaming app should not have any access to personal data, so personalized advertising is just not possible' (NP192).

On the other hand, a few participants (6/400) thought that the target audience of a financial app is particularly valuable to advertisers, due to their higher buying power, thus, promising a particularly high return on personalised advertising: 'The target market for the app is an older and more affluent audience, therefore it is worth exploring to show the personalized ads to yield a higher revenue' (P474).

Other Themes

These themes were mentioned by a few participants, but still provide interesting insights. For instance, 17 participants said that they prefer to let *users* decide what types of ads they want to see. For example, participant P39 shifted the responsibility to users assuming that they know what information was used for customising the ad, what are the privacy implications of such targeting, and what the appropriate tools are for controlling online tracking:

Because I bet on the smart mind of my client, he/she should know how ads work and should know whether if the ad is shown after seeing custom profiling data or not and to offer the choice to get tracked or not (P39).

Participant NP299 acknowledged that there is currently little transparency about the data practices in app stores, and that users may not pay attention to the disclosures with poor usability:

Somehow in google play they do not give at least warnings and most users install without first reading labels. The case is to leave that label so that the user reads or does not read it is aware of the type of advertising that is included with the application (NP299).

Eight participants expected that it will be easier and faster to implement non-personalised ads: 'Helps to get app on stores, we are not collecting personal information and it helps to pass faster' (NP12). Seven participants chose personalised ads simply because it is common and it is the status quo in app advertising: 'Many of the apps that I use have this type of ad' (P484).

7.4.4. Opinions About Ad Networks, Privacy Regulations, and Consent

In this section we report the results from the exit survey that helped us further contextualise and interpret the main treatment effects, as later discussed in Section 7.5.

Perceived Control Over Ads

While the choices about ad networks' and apps' business models are often made by upper-level and middle management (Figure D.3 in the Appendix), our participants feel involved in that decision-making process. Many participants have been involved at least a moderate amount in choosing ad networks (36%), configuring ads (46.7%), and integrating the code to enable in-app ads (47.5%) (Figure D.4 in the Appendix). However, despite the involvement in selecting ad networks, participants mostly agree that developers have moderate (40.25%) or very little (32.75%) control over the data collection by those networks (Figure D.5 in the Appendix); and end-users have even less of such control (Wilcoxon signed-rank test of perceived end-user control relative to developer control: $U = 8409$, $p < .001$)

Reasons for Not Including an Ad Network

More than half (69%) of participants have used at least one ad network in their apps. We asked the remaining 123 participants to explain why they did not include any ad networks in their apps and constructed themes around participants' answers (Table 7.3), as discussed in Section 7.3.

Forty percent of these participants (50/123) did not integrate ad networks because there was no need to use ads to monetise the app, for instance, because it was free or open-source, or relied on other sources of revenue. About 20% of participants (25/123) did not aim for a broad audience and public use, but used instead for small personal projects, learning experience, homework, or academic research. Some participants (18/123) considered ads intrusive and damaging to user experience: 'I've always found it less intrusive for the end-users and a much smoother experience for them overall so buying a premium version would be preferred as a way to monetise the apps' (P131). Others (16/123) said that they did not have control over that decision, e.g., because they were developing an app for a client. A few participants said that they did not know how to integrate an ad network (5/123), it was someone else's responsibility to do it (7/123), or the project was still in the early development stage for ad integration (4/123). Only four participants explicitly mentioned concerns about user privacy: 'Ad networks

Table 7.3.: Constructed themes around participants' reasons for not including ad networks in their apps ($N = 123$).

Reason for Not Including Ad Networks	#Participants
No need to monetise the app	50 (40.65%)
Generic reasons	31 (25.2%)
Paid apps	12 (9.8%)
Open-source or free apps	7 (5.7%)
Apps not intended for public audience	25 (20.3%)
Small and personal projects	17 (13.8%)
Academic projects	8 (6.5%)
Expected negative impact on user experience	18 (14.6%)
Decision was made by others	16 (13.0%)
It's a responsibility of others	7 (5.7%)
Don't know how to do it	5 (4.1%)
User privacy	4 (3.3%)
Still in early development stages	4 (3.3%)
Unclear responses	4 (3.3%)

are not transparent and can't be audited. I can't control the amount of information fetch from my users' (P201).

Perceived Impact of Personalised Ads on Revenue and User Base

We asked participants how choosing personalised ads over non-personalised ads is likely to affect the revenue and number of users (Figure D.6). The majority of participants expected an increase in revenue in both app categories, but no or little decrease in the user base. Specifically, almost half of participants expected an increase in revenue by up to 40%. Slightly more participants believed that the user base won't change in the gaming app compared to financial app (43% vs 32.5%). However, 16-18% of participants believed that deploying personalised ads will not change the revenue at all, or even *decrease* the revenue in both app categories, and decrease the user base by up to 40% in financial (32%) and gaming (23%) apps.

Beliefs About Privacy Regulations

In the survey scenarios, we told participants that the apps will be published in Europe and the United States and are mainly targeted towards adults above age of 18. For both apps, we asked participants to select the regulations that

would apply to each app, providing both full names and abbreviations of all regulation options. Most participants (70.5%) correctly chose GDPR, while the American privacy regulation CCPA was not chosen as often (26%), although the app descriptions explicitly mentioned that the apps will be published in both European and American markets. Moreover, specialised American regulations—Children’s Online Privacy Protection Act (COPPA) [80] and Health Insurance Portability and Accountability Act (HIPAA) [152]—were chosen by 22.8% and 9.9%, respectively, although the described apps were not directed at children and did not collect health-related information.

It is possible that the participants, most of which are from Europe, are more familiar with the European regulations than the American ones, however, we did not find a significant difference between the answers about applicable regulations between the European and North American residents (Mann-Whitney test: $U = 98708.0$, $p = 0.174$). Finally, 22.8% of participants did not know what regulations apply to the apps, and 2.9% thought that none of them apply. These results show that developers may not be familiar with privacy regulations outside their home country and may not know which regulations are applicable to their apps. It also echos the findings of interviews with developers that they rarely know about privacy guidelines and required measures for privacy [42].

Opinions About User Consent

In the exit survey, we asked participants how they would ask for user consent, assuming they had decided to use personalised ads (Table D.1 in the Appendix). The majority (32%) selected the consent form provided by our imaginary Acme ad network. Others preferred to rely on the consent forms provided by leading tech companies (22.5%), such as Facebook or Google, or not-for-profit organisations (10.7%), such as Mozilla or Electronic Frontier Foundation, or use their own consent forms (17.7%). Only 9.75% said they will not ask for user consent at all, assuming that ad network or someone else in the team will take care of it, or because they find the process difficult, unfamiliar, unimportant, or simply not required. Finally, 6% said they would consult the specialised companies providing compliance services.

We asked the 71 participants, who indicated they would use their own consent form, what *information sources* they would use to build it (Table 7.4). After constructing themes around open-ended responses using affinity diagrams, we found that almost a third (29.6%) of participants would still fall back on the existing consent forms built by other teams, apps, companies, non-for-profit organisations, or ready-to-use templates, when building their own forms. Another 19.7% would use general guidelines, such as regulatory policies and recommendations; four participants mentioned using user experience guidelines and best practices when

Table 7.4.: Constructed themes around participant’s information sources for building their consent forms ($N = 71$).

Information Source	#Participants
Reuse available materials	21 (29.6%)
From other companies and not-for-profits	17 (23.9%)
Ready-to-use templates	4 (5.6%)
Guidelines	14 (19.7%)
Legal policies (e.g., GDPR)	10 (14.1%)
UX guidelines	4 (5.6%)
Online search	9 (12.7%)
Legal teams	7 (9.9%)
Relying on own knowledge	6 (8.5%)
Don’t know	6 (8.5%)
Unclear responses	12 (16.9%)

building consent forms: ‘Existing UX research on consent forms and how to maximize consent with storytelling’ (P224).

Other participants said they would search for information about consent forms on the Internet (12.7%), rely on the legal teams or lawyers (9.9%), and their own knowledge or ‘common sense’ (8.5%). However, what constitutes ‘common sense’ for the developer may not necessarily represent what is ‘common sense’ for users. For instance, P277 said that they would tell users that their app uses ads, but would refrain from disclosing that those ads are based on personal information about them: ‘I’d be upfront about including ads but not state that they dig into people’s history’ (P277). Finally, 8.5% said they do not know what information they would rely on when building consent forms.

7.5. Discussion and Future Work

Prior work suggests the importance of improving usability of *security*-related interfaces for developers, for example, through security APIs [135], security notifications [329], and providing secure code examples [213–215]. Our study highlights the importance of *privacy* interfaces as well by looking at the impact of choice framing on developers’ decisions about user privacy while interacting with ad networks. We hypothesise that the low rate of GDPR-compliant consent forms on websites [120, 206, 346] and the abundance of non-compliant Android apps [189, 278, 302, 388] may partially be caused by developers’ low awareness about or consideration of consequences of their decisions on user privacy. We find

that incorporating nudges in the design of developers' tools may assist developers in making decisions that consider user privacy in their software development processes.

7.5.1. Provide Information About Privacy Implications of Ad Personalisation

The choice framing that described data processing as being restricted to contextual information instead of past behaviours produced positive but weaker effects compared to the explicit use of privacy labels (11.06 vs 3.45 times increase in the likelihood to choose non-personalised ads). We believe that this is because in the former case participants had to evaluate themselves the implications of using contextual vs behavioural targeting on user privacy, while labels that clearly indicated the positive and negative privacy consequences simplified this task. We hypothesise that developers may not fully understand the differences between contextual and behavioural targeting and associated privacy implications; future work is called to explore this hypothesis.

Thus, we recommend ad networks to include information to help developers evaluate privacy implications of their decisions in a transparent, concise, and direct way, by including clear privacy labels to the choices about the ad types. Including these options in the documentation and quick start guides as part of developers' workflow for ads integration may also assist developers in considering user privacy as part of their app development procedure. Additional information on users' concerns about behavioural targeting (e.g., discomfoting [197, 385], discriminating [266], and intrusive [241, 270]) might facilitate developers' assessment of privacy implications or support the claims about their relative privacy invasiveness; future work is needed to study how to effectively integrate this information without making the choice text options longer, and whether the manipulation is effective in nudging developers' choices in a less controlled setting.

7.5.2. Improve the Effectiveness of User-Facing Privacy Descriptions

Prior work recommends emphasising privacy features in the app stores [186], for instance, the recent inclusion of 'Privacy Details' in the Apple App Store aimed at explaining apps' privacy practices before users download them [29]. However, our experiment did not find evidence that adding user-facing descriptions (with our choice framing) of app's ad targeting practices would nudge participants to integrate less invasive non-personalised ads. Participants' open-ended comments suggest a potential explanation: most participants do not expect personalised ads to reduce their app's user base; they also believe that personalised ads are more

relevant and less annoying to the users. In other words, some participants believed that telling users that an app shows ads tailored to their personal information will not discourage users from downloading it, and indeed, may even attract users who prefer ads relevant and customised to their interests. However, prior work shows that some users do not like behaviourally targeted ads, find them invasive and creepy, and try to avoid or block such ads [13, 26, 219, 303, 312, 341].

Future work is called for to explore more efficient ways to nudge developers to consider privacy implications of their in-app ad choices. For instance, studying how to best provide evidence to developers about user opinions around ads, privacy preferences, and the impact of app-store presented information, would all help better inform developers' choices. Moreover, future work may test and improve the effectiveness of the existing ways to increase transparency and developers' responsibility to users' regarding their privacy, such as adding 'Privacy Details' in the Apple App Store [29], potentially from a privacy nutrition labels perspective [169].

7.5.3. Reconcile Contradicting Beliefs

As we explained in Section 7.4.2, the app category did not impact the decisions between the personalised and non-personalised ads, and the number of participants in each group differed only slightly. The analysis of category-related reasons (Section 7.4.3) provides a potential explanation why we might have not observed a difference. Specifically, it revealed the contradicting beliefs about the same app category that lead to different ad type choices, potentially cancelling out the effects of app category. For example, while some participants preferred non-personalised ads for financial apps to avoid raising privacy and trust concerns among users, others preferred to maximise profit from showing the personalised ads to this affluent user group, particularly valued by the advertisers. In the gaming context, because presumably the app does not collect sensitive information, some chose personalised ads as they believed it would not raise privacy concerns, others chose non-personalised ads as it would not be possible to customise ads due to the lack of personal information.

Similar contradictions are revealed in the experimental conditions. When we emphasised privacy implications, the majority of participants chose more privacy-friendly non-personalised ads. When we emphasised the implications on app's revenue, the majority chose revenue-maximising personalised ads. However, when faced with an explicit choice between user privacy and app's revenue, the choices between two types of ads split almost equally, with a small preference for non-personalised ads. This finding suggests the balance between the contradicting values is fragile and can be easily manipulated. Similar to users' privacy decisions being context-dependent [9, 12, 235], developers' decisions may also be driven

by contextual factors. As some of our participants clarified in the open-ended responses, this choice may change depending on the associated impact on revenue or user privacy. For instance, if the promised increase in revenue is high enough, developers may choose it over user privacy; if they believe that the data collected by the app or context of the app in general is particularly sensitive to raise user concerns, they may be more prone to choose user privacy over profit.

Developers may integrate ad networks primarily because they see it as the only feasible way to monetise the app [211]. The current choice framing in the ad networks also favours the revenue and uses a language that nudges developers into choosing the personalised ads [327]. However, there are also hidden costs of mobile ads that many developers do not consider in weighing the trade-offs, such as frequent updating of ad-related code, and increased consumption of energy and network data on users' phone and subsequent decrease in app's use [137]. Future work could suggest ways to provide transparency about such trade-offs by looking at proposed frameworks for improving the equilibrium between the revenue and user privacy in smartphones by adjusting the level of privacy protection in response to ad-generated revenue [183].

Our results also inform regulators that slight changes in ad networks' interface design for developers may affect the fragile balance between the contradicting values of personalised ads and significantly affect developers' choices to benefit platform's interests in profit maximisation. We recommend regulators build clear technical recommendations for providing choices to users, and to enforce that ad networks and other platforms use the mandated framing to promote users' welfare, and avoid effects driven by platforms' sole interests. Future work could provide inputs to the regulators by studying the usability of developer-facing interfaces (e.g., the privacy dashboard on Google AdMob), to inform the design of such interfaces and to provide suggestions to regulators on how to minimise the use of dark patterns in these interfaces.

7.5.4. Increase Developers' and Users' Control Over Data and Transparency

Many participants said that they do not have full control over ad networks' data collection and processing for ad personalisation, and that users have even less control over it. We recommend ad networks, and app stores in particular, to increase the transparency about data practices, accountability to users, and developers' and users' control over data. For instance, Google Play's privacy nudges for permissions has shown success in reducing the number of permissions that developers request [260]. This model might be used to make information about third-party libraries such as ad networks more specific. We suggest app stores to scan for ad libraries and inform developers about their privacy implications

during the automatic reviews of the apps (as they currently do for other purposes such as displaying third-party apps [31]).

Some of our participants said that they prefer to let users decide what types of ads they want to see (personalised or non-personalised). However, this line of thought is not completely fair to the users in the environment of information asymmetry, where users are poorly informed about the data practices of apps and ad networks, and personal data flows are not transparent to the users [19, 43, 61, 225]. Thus, providing means for users to see what ad networks are being used in apps when installing a new app [82], what types of ads do the apps serve, and what personal information is used to customise them, as well as other improvement in user interfaces described in Section 7.5.2, might be effective. Prior results from user research may also help build usable privacy interfaces for developers and increase transparency and control. For instance, several elements of the labels such as data collection, purpose, and data sharing [109, 169] might be reused to inform developers about an ad network's data collection. Other proposed interfaces that visually represent permissions, purposes, data leaks [191, 348], data flows, the effects of removing and adding libraries [347], and integrating privacy checks into programming interfaces [185] might further inform developers about the privacy consequences of their choices. Not-for-profit organisations could build open-source services and easy to integrate privacy consent mechanisms to facilitate consent integration, and offer alternatives to for-profit large companies consent forms. Future work could also evaluate the effectiveness of various types of information sources on developers' success in building compliant and user-friendly consent forms (Table 7.4).

7.6. Conclusion

We present the results of a survey-based online experiment with 400 participants with mobile app development experience on their decisions regarding configuring ads for hypothetical apps. We tested the impact of six conditions where we slightly changed the choice framing between personalised and non-personalised ads. We find that the choice framing significantly impacts developers' decisions. When user privacy implications and data processing restrictions were made salient, participants were 11.06 and 3.45 times more likely to select the non-personalised ads than when the neutral framing was used. Other nudges—emphasising the consequences of ads on app's revenue, presenting participants with an explicit choice between user privacy and app's revenue, and telling participants that users will be able to see whether the app is using ads based on their personal data or not—did not significantly changed participants decisions compared to the Control condition. We also find that participants have different opinions about ads

personalisation that lead to contrasting choices, such as their impact on revenue, user privacy, user experience, and what type of ads users eventually prefer.

We find that the choice framing in ad networks significantly impacts developers' choices and subsequently privacy of millions of users. Thus, more control and transparency should be provided to developers and users in choosing the type of ads and data collection practices. Moreover, some of our participants incorrectly identified what privacy regulations would apply to the apps, and many said they rely on ad networks and examples of tech companies, when building user consent forms. This means that those companies are not only responsible to their own users, but also set example for other smaller companies and independent developers, further illustrating the large impact of ad network platform's design and choice framing on data practices in app development. Our results have implications for ad networks, app stores, and regulators by giving them grounds for promoting user privacy by improving the usability of developer-facing interfaces to empower developers in making informed decisions for their users.

Acknowledgements

This work was sponsored in part by Microsoft Research through its PhD Scholarship Program and a Google Research Award, and in part by the National Security Agency's Science of Security program. Opinions, findings, and conclusions are those of the authors and do not necessarily reflect the views of the funders.

Back Cover

The experiment in this chapter shows the significant impact of software development platforms on developers' decisions regarding their users' privacy. As highlighted in Chapter 3, developers' privacy challenges are primarily driven by platforms and this chapter provides empirical evidence for this impact. Platforms have the capacity to sway developers in either direction, sharing more of users' data or protecting users' data. I believe that it is the regulators' task to make sure that corporations are not only thinking about their profit but also consider users' benefits. Developers on the other hand need more support from academia to provide them with facts and awareness about consequences of their choices on their users.

ÉPILOGUE



8. Final Thoughts

This thesis provides empirical evidence that the *developer factor* impacts software privacy. Developers' educational background, colleagues, social interactions, tools they use, and interfaces they interact with, all influence software privacy by impacting the developer factor. In this chapter, first, I discuss the implications of some of the main findings from all the papers provided in this thesis, such as the value of including privacy courses in computer science curricula, the need for usable privacy metrics, and the usability of developer-facing privacy interfaces. The chapter then follows with future directions and ends with a conclusion section.

8.1. Discussion

Developers' educational background may drive them to become champions of privacy in software teams, emphasising the value of including privacy courses in computer science curricula (Chapter 5). A side benefit of including such courses in universities is to facilitate privacy champions efforts. A challenge for privacy champions is to overcome the 'I've got nothing to hide' argument. This argument is extensively discussed by Daniel J. Solove [309]. One approach suggested by Solove is to disintegrate privacy into smaller understandable elements such as 'secondary use', 'exclusion', and 'breach of confidentiality'. Including such elements with the potential consequences of not considering privacy to both users and society in the computer science curricula may impact the future of the software development ecosystem by raising privacy-aware developers who may have some level of privacy concerns.

While abundant in the research community, privacy metrics are rarely mentioned by the experts on the ground, highlighting the value of impactful research that puts developers in the centre and provides privacy measures that can be used in software teams. *Privacy by Design* as a high-level framework was mentioned in the studies. It provides a starting point for discussions around privacy and what could be done. However, similar to privacy itself, it is challenging to translate Privacy by Design into technical requirements. Workshops and conferences related to privacy technicalities are a promising place to provide further roadmaps and directions for understanding how to include privacy in software design. The value of one-to-one conversations is also much appreciated by the privacy champions showing that the human factor and the human interaction influence developers' mindsets around privacy (Chapter 4). Software companies that look to add privacy into their core values may benefit from starting conversations about privacy by organising motivating speeches by the managers, running custom-designed

privacy workshops tailored to teams' specific needs, and embedding privacy champions in teams to provide a bottom-up approach for supporting privacy features.

Looking at the graphical developer-facing interfaces (e.g., ad networks' quick start pages and developers' dashboards) shows that developers are influenced by the choice framing and wording of the options designed by the platforms (Chapter 6 and Chapter 7). While our study focuses on one specific type of platform, I hypothesise the same effect may exist in other platforms such as app stores and analytics tools. Developers may unintentionally select options that may negatively influence the users, yet they are not fully informed about them. UK's Information Commissioner's Office recently introduced 'Age Appropriate Design Code' in which they refer to minimising the use of nudges in child-directed apps [15]. I recommend minimising the use of nudges in all interfaces for all types of users. People deserve to make choices without being nudged about specific options and actions. Like other users, developers also need to have transparent options to make an informed decision for their apps. Making nudges privacy-focused/friendly may also not be the best option because developers' decisions may impact their income as well and the relation between financial gain/loss and privacy gain/loss is yet not clear. Future research could look at how can we present both financial and privacy (and perhaps other factors) consequences of options in software development tools to developers in a usable way.

When looking at the questions developers ask on Stack Overflow, it is noteworthy that developers do have privacy concerns, and they look for privacy-friendly options (Chapter 3). Therefore, the assumption that developers may be ignorant about user privacy may not be valid. At least a portion of developers need support from tools builders and platform owners to consider privacy in their products. A step towards supporting developers could be designing privacy technologies by taking a developer-centred approach that considers developers in the design stages of these systems like any other everyday objects recommended by Don Norman [237]. Computer scientists are working to make programming accessible to everyone, for example, by offering 'what you see what you get' design methods [171]. Similar approaches are required in the privacy domain to make them accessible to all developers because developers from a wide range of backgrounds and show struggles in privacy-related tasks. A hypothetical tool may ask developers about what level of privacy they need (e.g., critical, moderate, and low) and reports back that the system can tolerate what types of attacks and is robust for what types of applications such as financial systems, children-directed apps, and a calendar app. For more advanced users, it could provide lower-level access with instructions about the associated risks. It could also provide relative privacy measures with standard reference points. For example, it may provide privacy levels relative to NASA's or an offline chess game's privacy measures.

Challenges in recruiting and finding developers after about a decade of research in this area still remain the same. Computer science students are one of the proxies for an accessible population, however, controversies still remain in the community about whether the results from a student population study are applicable and generalisable to a wider developer population. I believe software development companies can be an influential player and assist researchers to find participants for running studies. Collaboration between academia and industry can provide invaluable inputs to the whole software ecosystem, not only benefiting one company by running internal studies.

8.2. Future Directions

8.2.1. Privacy Policies

Privacy policies are a pain point for developers. A developer whose background is not in law and until only a few years ago had to work at the technical level is now required to know about terms and policies that they were never asked for before. Developers find it difficult to understand what is required, what terms they should include, and how to frame the policies. The language difference between developers' technical language and legal jargon also exacerbates the problem. Future work may look at alternative solutions such as providing a checkbox list that developers can go through and select the permissions and data they use. Then, the tool can generate a standard privacy policy. A human can double-check the final policy to satisfy all the requirements. Such tools, later on, can be integrated into developers' toolboxes like integrated development environments.

Another perspective is to view developers as users of privacy policies and terms of services of software platforms and study developers' understanding and attitudes about policies and terms directed to developers. The general public's attitude towards privacy policies is to skip or spend less than 90 seconds reading them [242]. Developers also have to read these policies and terms and translate them into a policy that can be used in their apps. A qualitative study with developers could be a starting point to understand developers' attitudes and understanding of such policies. A follow-up study could be an online experiment asking developers to go through a hypothetical software development service that shows a privacy policy before using the service could show how much of the policies are read and understood by the participants. Comparing these results with end-users privacy policies could provide insights into the usability and value of these pages.

8.2.2. Integration of Privacy Tasks in Developers' Workflows

Looking at the ad networks' documentation as an example shows that privacy requirements may be buried in the documentation and they may not be given the same priority as other tasks. Future research may provide recommendations about how to integrate privacy tasks into developer-facing documentation. A line of research could look at how developers look and search for information, how libraries should present information to developers, and how lessons learned from general programming libraries could be applied to the privacy domain. A first step may be a lab study where participants see the current documentation and asked to talk about privacy information they observed and how much they understood the documents' privacy-related information. A second study may modify the documentation by putting the privacy information amid code samples and the steps that a developer may take to accomplish their task (e.g., integrating an ad in their app). A third study may test the prototype with a larger population to find the solution's generalisability. This research direction could be a starting point for understanding how privacy requirements might be best presented to be seen as essential features of software systems rather than so-called non-functional requirements.

8.2.3. Privacy Evaluation Metrics

Privacy champions asked for metrics and measures to enable them to argue for privacy. Future research may first collect the available industry privacy metrics and then compare those with what is suggested in academia. A follow-up study may look at why there are over eighty metrics in the research community for privacy, but the industry has not yet adopted them. It may also provide inputs for what metrics are expected by industry and how academia can respond. Interviews with the people in charge of such evaluations could be a starting point.

8.2.4. Identifying and Supporting Privacy Champions

Champions are a well-studied group in the information systems' literature. They are also studied in the security domain [144, 145, 147], and this thesis extends the literature to the privacy domain. While we identified and brought attention to privacy champions, there is still a need to add structure to the privacy champions' work. In the security domain, organisations like OWASP provide a playbook for structuring the role of security champions by recognising their role and supporting them [254]. Similar approaches may be investigated in the privacy domain to support privacy champions' work further. A first step could be understanding what approaches organisations can use to support and recognise their champions internally. While compensation like titles and raises could work, there may also

be less obvious support methods like getting access to training typically only provided to privacy staff or being given time to attend privacy-focused conferences. Currently, we have little understanding of what approaches to recommend to organisations that want to encourage their privacy champions, and there is room for future work here.

8.2.5. Transparent Privacy Options for Developers

Our findings show that nudging developers about user privacy impacts their decisions and consequently impacts their users' privacy [323]. Dark patterns directed to developers could result in unwanted consequences such as excessive data collection from users by third-parties. Research in the end-user community is currently looking at the ethics, challenges, and potential solutions for minimising the effects of dark patterns [204]. I expect to see similar concerns rising in developer-facing interfaces as well. Future research may study other developer-facing interfaces such as analytics tools and how they may intentionally or unintentionally use dark patterns to nudge developers to collect data from users or share data with third-parties. Follow-up studies could look at developer-facing interfaces such as code samples. While providing a code sample is a good way to assist developers in achieving their tasks because they can copy and paste the code, they should provide defaults favouring users or providing neutral options. Future research may find ways to integrate neutral options in the code samples and investigate methods to include privacy-friendly nudges in the code samples, potentially as comments and privacy to-dos. This direction may include ways to design nudges that also considers financial impacts of such choices for developer as well, as it is yet not clear how much developers may lose or gain by including personalised ads in their apps. Currently, developers may only think that one of the easiest (and perhaps lucrative) way of monetising their app is to include ads, however, this may not be true, and future research could provide insights into other monetisation methods for apps with a smaller audience groups.

8.3. Conclusion

In this thesis, I looked at several elements that can support developers in privacy tasks, such as academic support, peer-support (either online or in-person), tool-based support from static analysis tools, and libraries such as ad networks. My findings show that developers need support from all these elements, and this list can be expanded to many other elements, such as cryptographic libraries, continuous integration tools, and coding boot camps, to name a few. I find that there have been efforts towards including the developer factor in the design of

8. *Final Thoughts*

privacy technologies; however, there are still several research gaps to be filled to fully support developers in performing privacy tasks.

The concept of *Privacy by Design* cannot be achieved unless the *developer factor* is considered in the design process of privacy technologies. Academics, regulators, and software development platform owners need to work together to provide usable tools and interfaces both in the code and graphical levels. Such tools and interfaces should provide transparent and easy-to-use options, integrate privacy tasks in the developers' workflows just like any other programming tasks, and provide defaults that favour users best interests instead of platforms.

REFERENCES

Bibliography

- [1] *About IAB Tech Lab*. IAB Technology Laboratory (Tech Lab). 2018. URL: <https://wiki.iabtechlab.com> (visited on 09/2020) (cited on page 125).
- [2] Yasemin Acar, Michael Backes, Sascha Fahl, Simson Garfinkel, Doowon Kim, Michelle L Mazurek, and Christian Stransky. ‘Comparing the Usability of Cryptographic APIs’. In: *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2017, pp. 154–171. DOI: [10.1109/SP.2017.52](https://doi.org/10.1109/SP.2017.52) (cited on pages 20, 26, 28, 30, 38, 100, 101).
- [3] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L Mazurek, and Christian Stransky. ‘You Get Where You’re Looking for: The Impact of Information Sources on Code Security’. In: *2016 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2016, pp. 289–305. DOI: [10.1109/SP.2016.25](https://doi.org/10.1109/SP.2016.25) (cited on pages 14, 20, 26, 31–33, 38, 39, 100, 101, 114).
- [4] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L Mazurek, and Christian Stransky. ‘How Internet Resources Might Be Helping You Develop Faster but Less Securely’. In: *IEEE Security Privacy 15.2* (Mar. 2017), pp. 50–60. DOI: [10.1109/MSP.2017.24](https://doi.org/10.1109/MSP.2017.24) (cited on page 39).
- [5] Yasemin Acar, Sascha Fahl, and Michelle L Mazurek. ‘You are Not Your Developer, Either: A Research Agenda for Usable Security and Privacy Research Beyond End Users’. In: *Cybersecurity Development (SecDev)*, IEEE. 2016, pp. 3–8. DOI: [10.1109/SecDev.2016.013](https://doi.org/10.1109/SecDev.2016.013) (cited on pages 100, 101).
- [6] Yasemin Acar, Christian Stransky, Dominik Wermke, Michelle L. Mazurek, and Sascha Fahl. ‘Security Developer Studies with GitHub Users: Exploring a Convenience Sample’. In: *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. Santa Clara, CA: USENIX Association, July 2017, pp. 81–95. URL: <https://www.usenix.org/conference/soups2017/technical-sessions/presentation/acar> (cited on pages 20, 21, 30–32, 146).
- [7] Yasemin Acar, Christian Stransky, Dominik Wermke, Charles Weir, Michelle L Mazurek, and Sascha Fahl. ‘Developers Need Support, Too: A Survey of Security Advice for Software Developers’. In: *2017 IEEE Cybersecurity Development (SecDev)*. 2017, pp. 22–26. DOI: [10.1109/SecDev.2017.17](https://doi.org/10.1109/SecDev.2017.17) (cited on page 100).
- [8] *Accessing Protected Resources*. Apple. 2019. URL: https://developer.apple.com/documentation/uikit/protecting_the_user_s_privacy/accessing_protected_resources (visited on 09/2019) (cited on page 60).

- [9] Mark Ackerman, Trevor Darrell, and Daniel J Weitzner. 'Privacy in context'. In: *Human-Computer Interaction* 16.2-4 (2001), pp. 167–176. doi: [10.1207/S15327051HCI16234_03](https://doi.org/10.1207/S15327051HCI16234_03) (cited on page 159).
- [10] Alessandro Acquisti, Idris Adjerid, Rebecca Balebako, Laura Brandimarte, Lorrie Faith Cranor, Saranga Komanduri, Pedro Giovanni Leon, Norman Sadeh, Florian Schaub, Manya Sleeper, Yang Wang, and Shomir Wilson. 'Nudges for Privacy and Security: Understanding and Assisting Users' Choices Online'. In: *ACM Computing Surveys* 50.3 (Aug. 2017). doi: [10.1145/3054926](https://doi.org/10.1145/3054926) (cited on pages 9, 138, 141).
- [11] Alessandro Acquisti and Jens Grossklags. 'What can behavioral economics teach us about privacy'. In: *Digital privacy: theory, technologies and practices* 18 (2007), pp. 363–377. doi: [10.1201/9781420052183.ch18](https://doi.org/10.1201/9781420052183.ch18) (cited on page 68).
- [12] Alessandro Acquisti, Leslie K. John, and George Loewenstein. 'What Is Privacy Worth?' In: *The Journal of Legal Studies* 42.2 (2013), pp. 249–274. doi: [10.1086/671754](https://doi.org/10.1086/671754) (cited on page 159).
- [13] *Ad-Blocking: A deep-dive into ad-blocking trends*. Tech. rep. GlobalWebIndex, 2018. URL: <https://www.globalwebindex.com/hubfs/Downloads/Ad-Blocking-trends-report.pdf> (visited on 02/2021) (cited on pages 121, 159).
- [14] Anne Adams and Martina Angela Sasse. 'Users Are Not the Enemy'. In: *Communications of the ACM* 42.12 (Dec. 1999), pp. 40–46. doi: [10.1145/322796.322806](https://doi.org/10.1145/322796.322806) (cited on pages 3, 14).
- [15] *Age appropriate design: a code of practice for online services*. Information Commissioner's Office. 2020. URL: <https://ico.org.uk/for-organisations/guide-to-data-protection/key-data-protection-themes/age-appropriate-design-code/> (visited on 03/2021) (cited on page 168).
- [16] Md Ahasanuzzaman, Safwat Hassan, Cor-Paul Bezemer, and Ahmed E. Hassan. 'A longitudinal study of popular ad libraries in the Google Play Store'. en. In: *Empirical Software Engineering* 25.1 (Jan. 2020), pp. 824–858. doi: [10.1007/s10664-019-09766-x](https://doi.org/10.1007/s10664-019-09766-x) (cited on pages 123, 143).
- [17] Md Ahasanuzzaman, Safwat Hassan, and Ahmed E. Hassan. 'Studying Ad Library Integration Strategies of Top Free-to-Download Apps'. In: *IEEE Transactions on Software Engineering* PP (Mar. 2020), pp. 1–1. doi: [10.1109/TSE.2020.2983399](https://doi.org/10.1109/TSE.2020.2983399) (cited on page 143).
- [18] Icek Ajzen. 'From Intentions to Actions: A Theory of Planned Behavior'. In: *Action Control: From Cognition to Behavior*. Ed. by Julius Kuhl and Jürgen Beckmann. Berlin, Heidelberg: Springer Berlin Heidelberg, 1985, pp. 11–39. doi: [10.1007/978-3-642-69746-3_2](https://doi.org/10.1007/978-3-642-69746-3_2) (cited on page 145).
- [19] Urs-Vito Albrecht. 'Transparency of health-apps for trust and decision making.' eng. In: *Journal of medical Internet research* 15.12 (Dec. 2013). doi: [10.2196/jmir.2981](https://doi.org/10.2196/jmir.2981) (cited on page 161).

- [20] Alexa. *stackoverflow.com*. 2019. URL: <https://www.alexam.com/siteinfo/stackoverflow.com> (visited on 09/2019) (cited on page 39).
- [21] Miltiadis Allamanis and Charles Sutton. 'Why, when, and what: Analyzing Stack Overflow questions by topic, type, and code'. In: *2013 10th Working Conference on Mining Software Repositories (MSR)*. May 2013, pp. 53–56. DOI: [10.1109/MSR.2013.6624004](https://doi.org/10.1109/MSR.2013.6624004) (cited on pages 39, 40, 42, 45, 46).
- [22] Majed Almansoori, Jessica Lam, Elias Fang, Kieran Mulligan, Adalbert Gerald Soosai Raj, and Rahul Chatterjee. 'How Secure Are Our Computer Systems Courses?' In: *Proceedings of the 2020 ACM Conference on International Computing Education Research*. ICER '20. Virtual Event, New Zealand: Association for Computing Machinery, 2020, pp. 271–281. DOI: [10.1145/3372782.3406266](https://doi.org/10.1145/3372782.3406266) (cited on page 93).
- [23] Teresa M. Amabile. 'Motivational synergy: Toward new conceptualizations of intrinsic and extrinsic motivation in the workplace'. In: *Human Resource Management Review* 3.3 (1993), pp. 185–201. DOI: [10.1016/1053-4822\(93\)90012-5](https://doi.org/10.1016/1053-4822(93)90012-5) (cited on page 78).
- [24] Leo St. Amour and W. Michael Petullo. 'Improving Application Security through TLS-Library Redesign'. In: *Security, Privacy, and Applied Cryptography Engineering*. Ed. by Rajat Subhra Chakraborty, Peter Schwabe, and Jon Solworth. Cham: Springer International Publishing, 2015, pp. 75–94. DOI: [10.1007/978-3-319-24126-5_5](https://doi.org/10.1007/978-3-319-24126-5_5) (cited on page 14).
- [25] Le An, Ons Mlouki, Foutse Khomh, and Giuliano Antoniol. 'Stack Overflow: A code laundering platform?' In: *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*. Feb. 2017, pp. 283–293. DOI: [10.1109/SANER.2017.7884629](https://doi.org/10.1109/SANER.2017.7884629) (cited on page 39).
- [26] Mimi An. *Why People Block Ads (And What It Means for Marketers and Advertisers)*. HubSpot. 2020. URL: <https://blog.hubspot.com/marketing/why-people-block-ads-and-what-it-means-for-marketers-and-advertisers> (visited on 02/2021) (cited on pages 121, 159).
- [27] Benjamin Andow, Samin Yaseer Mahmud, Wenyu Wang, Justin Whitaker, William Enck, Bradley Reaves, Kapil Singh, and Tao Xie. 'PolicyLint: Investigating Internal Privacy Policy Contradictions on Google Play'. In: *28th USENIX Security Symposium (USENIX Security 19)*. Santa Clara, CA: USENIX Association, Aug. 2019, pp. 585–602. URL: <https://www.usenix.org/conference/usenixsecurity19/presentation/andow> (cited on page 59).
- [28] *Android Ad Network statistics and market share*. AppBrain. 2020. URL: <https://www.appbrain.com/stats/libraries/ad-networks> (visited on 09/2020) (cited on pages 122, 138, 140).

- [29] *App privacy details on the App Store*. Apple. 2021. URL: <https://developer.apple.com/app-store/app-privacy-details/> (visited on 02/2021) (cited on pages 143, 158, 159).
- [30] *App Review*. Apple. 2019. URL: <https://developer.apple.com/app-store/review> (visited on 09/2019) (cited on page 60).
- [31] *App Store Review Guidelines*. Apple. 2021. URL: <https://developer.apple.com/app-store/review/guidelines/#unacceptable> (visited on 02/2021) (cited on page 161).
- [32] Iván Arce, Kathleen Clark-Fisher, Neil Daswani, Jim DelGrosso, Danny Dhillon, Christoph Kern, Tadayoshi Kohno, Carl Landwehr, Gary McGraw, Brook Schoenfeld, et al. 'Avoiding the Top 10 Software Security Design Flaws'. In: *Technical report, IEEE Computer Societys Center for Secure Design (CSD)* (2014). URL: <https://cybersecurity.ieee.org/blog/2015/11/13/avoiding-the-top-10-security-flaws/> (cited on pages 100, 101, 115).
- [33] Steven Arzt, Sarah Nadi, Karim Ali, Eric Bodden, Sebastian Erdweg, and Mira Mezini. 'Towards Secure Integration of Cryptographic Software'. In: *2015 ACM International Symposium on New Ideas, New Paradigms, and Reflections on Programming and Software (Onward!)* Onward! 2015. Pittsburgh, PA, USA: Association for Computing Machinery, 2015, pp. 1–13. DOI: [10.1145/2814228.2814229](https://doi.org/10.1145/2814228.2814229) (cited on page 112).
- [34] Farzaneh Asgharpour, Debin Liu, and L. Jean Camp. 'Mental Models of Security Risks'. In: *Financial Cryptography and Data Security*. Ed. by Sven Dietrich and Rachna Dhamija. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 367–377. DOI: [10.1007/978-3-540-77366-5_34](https://doi.org/10.1007/978-3-540-77366-5_34) (cited on page 108).
- [35] Hala Assal and Sonia Chiasson. 'Security in the Software Development Lifecycle'. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 281–296. URL: <https://www.usenix.org/conference/soups2018/presentation/assal> (cited on pages 20, 23, 24, 41, 116).
- [36] Hala Assal and Sonia Chiasson. 'Think Secure from the Beginning': A Survey with Software Developers'. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland UK: ACM, 2019, 289:1–289:13. DOI: [10.1145/3290605.3300519](https://doi.org/10.1145/3290605.3300519) (cited on pages 41, 115).
- [37] Hala Assal, Sonia Chiasson, and Robert Biddle. 'Cesar: Visual representation of source code vulnerabilities'. In: *2016 IEEE Symposium on Visualization for Cyber Security (VizSec)*. IEEE, 2016, pp. 1–8. DOI: [10.1109/VIZSEC.2016.7739576](https://doi.org/10.1109/VIZSEC.2016.7739576) (cited on pages 20, 28, 38).

- [38] Oshrat Ayalon, Eran Toch, Irit Hadar, and Michael Birnhack. 'How Developers Make Design Decisions about Users' Privacy: The Place of Professional Communities and Organizational Climate'. In: *Companion of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. CSCW '17 Companion. Portland, Oregon, USA: Association for Computing Machinery, 2017, pp. 135–138. DOI: [10.1145/3022198.3026326](https://doi.org/10.1145/3022198.3026326) (cited on page 68).
- [39] Nathaniel Ayewah, David Hovemeyer, J. David Morgenthaler, John Penix, and William Pugh. 'Using Static Analysis to Find Bugs'. In: *IEEE Software* 25.05 (Sept. 2008), pp. 22–29. DOI: [10.1109/MS.2008.130](https://doi.org/10.1109/MS.2008.130) (cited on pages 20, 27, 28).
- [40] Nathaniel Ayewah and William Pugh. 'A Report on a Survey and Study of Static Analysis Users'. In: *Proceedings of the 2008 Workshop on Defects in Large Software Systems*. DEFECTS '08. Seattle, Washington: Association for Computing Machinery, 2008, pp. 1–5. DOI: [10.1145/1390817.1390819](https://doi.org/10.1145/1390817.1390819) (cited on pages 20, 27, 28).
- [41] Mohammad Azhar, Sajal Bhatia, Greg Gagne, Chadi Kari, Joseph Maguire, Xenia Mountroudou, Liviana Tudor, David Vosen, and Timothy T. Yuen. 'Securing the Human: Broadening Diversity in Cybersecurity'. In: *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE '19. Aberdeen, Scotland UK: Association for Computing Machinery, 2019, pp. 251–252. DOI: [10.1145/3304221.3325537](https://doi.org/10.1145/3304221.3325537) (cited on page 104).
- [42] Rebecca Balebako and Lorrie Cranor. 'Improving App Privacy: Nudging App Developers to Protect User Privacy'. In: *IEEE Security Privacy* 12.4 (2014), pp. 55–58. DOI: [10.1109/MSP.2014.70](https://doi.org/10.1109/MSP.2014.70) (cited on pages 41, 100, 102, 156).
- [43] Rebecca Balebako, Jaeyeon Jung, Wei Lu, Lorrie Faith Cranor, and Carolyn Nguyen. "'Little Brothers Watching You": Raising Awareness of Data Leaks on Smartphones'. In: *Proceedings of the Ninth Symposium on Usable Privacy and Security*. SOUPS '13. Newcastle, United Kingdom: Association for Computing Machinery, 2013. DOI: [10.1145/2501604.2501616](https://doi.org/10.1145/2501604.2501616) (cited on page 161).
- [44] Rebecca Balebako, Pedro G Leon, Hazim Almuhiemedi, Patrick Gage Kelley, Jonathan Mugan, Alessandro Acquisti, Lorrie Cranor, and Norman Sadeh-Koniecpol. 'Nudging users towards privacy on mobile devices'. In: *CHI 2011 Workshop on Persuasion, Nudge, Influence and Coercion*. Carnegie Mellon University, 2011. DOI: [10.1184/R1/13028258.v1](https://doi.org/10.1184/R1/13028258.v1) (cited on page 141).
- [45] Rebecca Balebako, Abigail Marsh, Jialiu Lin, Jason I Hong, and Lorrie Cranor. 'The privacy and security behaviors of smartphone app developers'.

- In: *Workshop on Usable Security (USEC'14)*. Internet Society, 2014. DOI: [10.14722/usec.2014.23006](https://doi.org/10.14722/usec.2014.23006) (cited on pages 20, 27, 38, 41, 142).
- [46] Rebecca Balebako, Florian Schaub, Idris Adjerid, Alessandro Acquisti, and Lorrie Cranor. 'The Impact of Timing on the Salience of Smartphone App Privacy Notices'. In: *Proceedings of the 5th Annual ACM CCS Workshop on Security and Privacy in Smartphones and Mobile Devices*. SPSM '15. Denver, Colorado, USA: Association for Computing Machinery, 2015, pp. 63–74. DOI: [10.1145/2808117.2808119](https://doi.org/10.1145/2808117.2808119) (cited on pages 138, 141, 143, 144).
- [47] Derek E. Bambauer. 'Privacy versus Security'. In: *Journal of Criminal Law and Criminology* 103 (2013), pp. 667–684. URL: <https://ssrn.com/abstract=2208824> (cited on pages 38, 59).
- [48] Kenneth A Bamberger, Serge Egelman, Catherine Han, Amit Elazari Bar On, and Irwin Reyes. 'Can You Pay For Privacy? Consumer Expectations and the Behavior of Free and Paid Apps'. In: *Berkeley Technology Law Journal* 35 (2020). DOI: [10.15779/Z38XP6V40J](https://doi.org/10.15779/Z38XP6V40J) (cited on pages 123, 140).
- [49] Kenneth A Bamberger and Deirdre K Mulligan. *Privacy on the Ground: Driving Corporate Behavior in the United States and Europe*. MIT Press, 2015. URL: <https://mitpress.mit.edu/books/privacy-ground> (cited on page 78).
- [50] Hyejin Bang and Bartosz W. Wojdyski. 'Tracking users' visual attention and responses to personalized advertising based on task cognitive demand'. In: *Computers in Human Behavior* 55 (2016), pp. 867–876. DOI: [10.1016/j.chb.2015.10.025](https://doi.org/10.1016/j.chb.2015.10.025) (cited on page 140).
- [51] Catherine Barrett. 'Are the EU GDPR and the California CCPA becoming the de facto global standards for data privacy and protection?' In: *Scitech Lawyer* 15.3 (2019), pp. 24–29. URL: <https://search.proquest.com/docview/2199825726> (cited on page 69).
- [52] Steffen Bartsch. 'Practitioners' Perspectives on Security in Agile Development'. In: *Proceedings of the 2011 Sixth International Conference on Availability, Reliability and Security*. ARES '11. USA: IEEE Computer Society, 2011, pp. 479–484. DOI: [10.1109/ARES.2011.82](https://doi.org/10.1109/ARES.2011.82) (cited on pages 20, 24, 26, 29, 115).
- [53] Anton Barua, Stephen W. Thomas, and Ahmed E. Hassan. 'What are developers talking about? An analysis of topics and trends in Stack Overflow'. In: *Empirical Software Engineering* 19.3 (June 2014), pp. 619–654. DOI: [10.1007/s10664-012-9231-y](https://doi.org/10.1007/s10664-012-9231-y) (cited on pages 39, 42, 45, 46).
- [54] Aniqua Baset and Tamara Denning. 'IDE Plugins for Detecting Input-Validation Vulnerabilities'. In: *2017 IEEE Security and Privacy Workshops (S&PW)*. 2017, pp. 143–146. DOI: [10.1109/SPW.2017.37](https://doi.org/10.1109/SPW.2017.37) (cited on page 33).
- [55] Cynthia Mathis Beath. 'Supporting the Information Technology Champion'. In: *MIS Quarterly* 15.3 (1991), pp. 355–372. DOI: [10.2307/249647](https://doi.org/10.2307/249647) (cited on page 91).

- [56] Ingolf Becker, Simon Parkin, and M. Angela Sasse. 'Finding Security Champions in Blends of Organisational Culture'. In: *Proceedings 2nd European Workshop on Usable Security*. Paris, France: Internet Society, 2017. doi: [10.14722/eurousec.2017.23007](https://doi.org/10.14722/eurousec.2017.23007). (Visited on 03/27/2020) (cited on page 70).
- [57] Kathrin Bednar, Sarah Spiekermann, and Marc Langheinrich. 'Engineering Privacy by Design: Are engineers ready to live up to the challenge?' In: *The Information Society* 35.3 (2019), pp. 122–142. doi: [10.1080/01972243.2019.1583296](https://doi.org/10.1080/01972243.2019.1583296) (cited on pages 40, 41).
- [58] Laura Bell, Michael Brunton-Spall and Rich Smith, and Jim Bird. *Agile Application Security: Enabling Security in a Continuous Delivery Pipeline*. O'Reilly Media, 2017 (cited on page 115).
- [59] Odette Beris, Adam Beutement, and M. Angela Sasse. 'Employee Rule Breakers, Excuse Makers and Security Champions: Mapping the Risk Perceptions and Emotions That Drive Security Behaviors'. In: *Proceedings of the 2015 New Security Paradigms Workshop*. NSPW '15. Twente, Netherlands: Association for Computing Machinery, 2015, pp. 73–84. doi: [10.1145/2841113.2841119](https://doi.org/10.1145/2841113.2841119) (cited on pages 70, 90).
- [60] Karin Bernsmed and Martin Jaatun. 'Threat modelling and agile software development: Identified practice in four Norwegian organisations'. In: *2019 International Conference on Cyber Security and Protection of Digital Services (Cyber Security)*. IEEE, 2019, pp. 1–8. doi: [10.1109/CyberSecPODS.2019.8885144](https://doi.org/10.1109/CyberSecPODS.2019.8885144) (cited on page 70).
- [61] Jan Hendrik Betzing, Matthias Tietz, Jan vom Brocke, and Jörg Becker. 'The impact of transparency on mobile privacy decision making'. In: *Electronic Markets* 30.3 (Sept. 2020), pp. 607–625. doi: [10.1007/s12525-019-00332-3](https://doi.org/10.1007/s12525-019-00332-3) (cited on page 161).
- [62] Stefanie Beyer and Martin Pinzger. 'A Manual Categorization of Android App Development Issues on Stack Overflow'. In: *2014 IEEE International Conference on Software Maintenance and Evolution*. Sept. 2014, pp. 531–535. doi: [10.1109/ICSME.2014.88](https://doi.org/10.1109/ICSME.2014.88) (cited on pages 38–40, 42, 45, 46, 49).
- [63] Jens F. Binder, Thom Baguley, Chris Crook, and Felicity Miller. 'The academic value of internships: Benefits across disciplines and student backgrounds'. In: *Contemporary Educational Psychology* 41 (2015), pp. 73–82. doi: [10.1016/j.cedpsych.2014.12.001](https://doi.org/10.1016/j.cedpsych.2014.12.001) (cited on page 116).
- [64] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 'Latent Dirichlet Allocation'. In: *Journal of Machine Learning Research* 3 (Mar. 2003), pp. 993–1022. URL: <http://dl.acm.org/citation.cfm?id=944919.944937> (cited on page 41).

- [65] Christoph Bösch, Benjamin Erb, Frank Kargl, Henning Kopp, and Stefan Pfattheicher. 'Tales from the Dark Side: Privacy Dark Strategies and Privacy Dark Patterns'. In: *Proceedings on Privacy Enhancing Technologies* 2016.4 (2016), pp. 237–254. DOI: [10.1515/popets-2016-0038](https://doi.org/10.1515/popets-2016-0038) (cited on pages 123, 141).
- [66] Jonathan P. Bowen, Mike Hinchey, Helge Janicke, Martin Ward, and Hussein Zedan. 'Formality, Agility, Security, and Evolution in Software Development'. In: *Computer* 47.10 (Oct. 2014), pp. 86–89. DOI: [10.1109/MC.2014.284](https://doi.org/10.1109/MC.2014.284) (cited on page 115).
- [67] Virginia Braun and Victoria Clarke. 'Using thematic analysis in psychology'. In: *Qualitative Research in Psychology* 3.2 (2006), pp. 77–101. DOI: [10.1191/1478088706qp0630a](https://doi.org/10.1191/1478088706qp0630a) (cited on page 145).
- [68] Alex Braunstein, Laura Granka, and Jessica Staddon. 'Indirect Content Privacy Surveys: Measuring Privacy without Asking about It'. In: *Proceedings of the Seventh Symposium on Usable Privacy and Security*. SOUPS '11. Pittsburgh, Pennsylvania: Association for Computing Machinery, 2011. DOI: [10.1145/2078827.2078847](https://doi.org/10.1145/2078827.2078847) (cited on page 141).
- [69] Robert L Brennan and Dale J Prediger. 'Coefficient Kappa: Some Uses, Misuses, and Alternatives'. In: *Educational and psychological measurement* 41.3 (1981), pp. 687–699. DOI: [10.1177/001316448104100307](https://doi.org/10.1177/001316448104100307) (cited on page 73).
- [70] Harry Brignull. *Dark Patterns*. 2020. URL: <https://darkpatterns.org> (visited on 09/2020) (cited on page 123).
- [71] Fei Bu, Nengmin Wang, Bin Jiang, and Huigang Liang. "'Privacy by Design" implementation: Information system engineers' perspective'. In: *International Journal of Information Management* 53 (2020), p. 102124. DOI: [10.1016/j.ijinfomgt.2020.102124](https://doi.org/10.1016/j.ijinfomgt.2020.102124) (cited on page 91).
- [72] *California Consumer Privacy Act (CCPA)*. State of California Department of Justice. 2018. URL: <https://oag.ca.gov/privacy/ccpa> (visited on 09/2020) (cited on pages 4, 69, 124, 140).
- [73] Volkan Cambazoglu and Neena Thota. 'Computer Science Students' Perception of Computer Network Security'. In: *2013 Learning and Teaching in Computing and Engineering*. 2013, pp. 204–207. DOI: [10.1109/LaTiCE.2013.19](https://doi.org/10.1109/LaTiCE.2013.19) (cited on page 102).
- [74] L. Jean Camp. 'Mental models of privacy and security'. In: *IEEE Technology and Society Magazine* 28.3 (2009), pp. 37–46. DOI: [10.1109/MTS.2009.934142](https://doi.org/10.1109/MTS.2009.934142) (cited on page 114).
- [75] Andrew Campbell and Sally Yeung. 'Creating a sense of mission'. In: *Long Range Planning* 24.4 (1991), pp. 10–20. DOI: [10.1016/0024-6301\(91\)90002-6](https://doi.org/10.1016/0024-6301(91)90002-6) (cited on page 91).

- [76] Ana Caraban, Evangelos Karapanos, Daniel Gonçalves, and Pedro Campos. ‘23 Ways to Nudge: A Review of Technology-Mediated Nudging in Human-Computer Interaction’. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI ’19. Glasgow, Scotland UK: Association for Computing Machinery, 2019, pp. 1–15. doi: [10.1145/3290605.3300733](https://doi.org/10.1145/3290605.3300733) (cited on page 141).
- [77] Susan Cartwright and Nicola Holmes. ‘The meaning of work: The challenge of regaining employee engagement and reducing cynicism’. In: *Human Resource Management Review* 16.2 (2006). The New World of Work and Organizations, pp. 199–208. doi: [10.1016/j.hrmr.2006.03.012](https://doi.org/10.1016/j.hrmr.2006.03.012) (cited on page 91).
- [78] Ann Cavoukian. ‘Privacy by Design: The 7 Foundational Principles’. In: *Information and privacy commissioner of Ontario, Canada* 5 (2009). URL: https://iab.org/wp-content/uploads/2011/03/fred_carter.pdf (cited on pages 40, 68, 69, 92).
- [79] Ann Cavoukian, Scott Taylor, and Martin E. Abrams. ‘Privacy by Design: essential for organizational accountability and strong business practices’. In: *Identity in the Information Society* 3.2 (Aug. 2010), pp. 405–413. doi: [10.1007/s12394-010-0053-z](https://doi.org/10.1007/s12394-010-0053-z) (cited on page 40).
- [80] *Children’s Online Privacy Protection Rule (COPPA)*. Federal Trade Commission. 1998. URL: <https://www.ftc.gov/enforcement/rules/rulemaking-regulatory-reform-proceedings/childrens-online-privacy-protection-rule> (visited on 09/2020) (cited on pages 124, 156).
- [81] Shiona Chillias, Abigail Marks, and Laura Galloway. ‘Learning to labour: an evaluation of internships and employability in the ICT sector’. In: *New Technology, Work and Employment* 30.1 (2015), pp. 1–15. doi: [10.1111/ntwe.12041](https://doi.org/10.1111/ntwe.12041) (cited on page 116).
- [82] Saksham Chitkara, Nishad Gothoskar, Suhas Harish, Jason I. Hong, and Yuvraj Agarwal. ‘Does This App Really Need My Location? Context-Aware Privacy Management for Smartphones’. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.3 (Sept. 2017). doi: [10.1145/3132029](https://doi.org/10.1145/3132029) (cited on page 161).
- [83] Maria Christakis and Christian Bird. ‘What Developers Want and Need from Program Analysis: An Empirical Study’. In: *Proceedings of the 31st IEEE/ACM International Conference on Automated Software Engineering*. ASE 2016. Singapore, Singapore: Association for Computing Machinery, 2016, pp. 332–343. doi: [10.1145/2970276.2970347](https://doi.org/10.1145/2970276.2970347) (cited on pages 20, 27).
- [84] Maria da Conceição Freitas and Miguel Mira da Silva. ‘GDPR Compliance in SMEs: There is much to be done’. In: *Journal of Information Systems Engineering & Management* 3.4 (2018), p. 30. doi: [10.20897/jisem/3941](https://doi.org/10.20897/jisem/3941) (cited on page 69).

- [85] Kovila P. L. Coopamootoo and Thomas Groß. 'Cyber Security and Privacy Experiments: A Design and Reporting Toolkit'. In: *Privacy and Identity Management. The Smart Revolution: 12th IFIP WG 9.2, 9.5, 9.6/11.7, 11.6/SIG 9.2.2 International Summer School, Ispra, Italy, September 4-8, 2017, Revised Selected Papers*. Ed. by Marit Hansen, Eleni Kosta, Igor Nai-Fovino, and Simone Fischer-Hübner. Cham: Springer International Publishing, 2018, pp. 243–262. doi: [10.1007/978-3-319-92925-5_17](https://doi.org/10.1007/978-3-319-92925-5_17) (cited on pages 17, 23, 34).
- [86] Catherine Courage and Kathy Baxter. 'APPENDIX F - AFFINITY DIAGRAM'. In: *Understanding Your Users*. Interactive Technologies. San Francisco: Morgan Kaufmann, 2005, pp. 714–721. doi: [10.1016/B978-155860935-8/50050-6](https://doi.org/10.1016/B978-155860935-8/50050-6) (cited on page 17).
- [87] Joseph Cox. *How the U.S. Military Buys Location Data from Ordinary Apps*. VICE. 2020. URL: <https://www.vice.com/en/article/jgqm5x/us-military-location-data-xmode-locate-x> (visited on 02/2021) (cited on page 140).
- [88] Lorrie Cranor and Norman Sadeh. 'A Shortage of Privacy Engineers'. In: *IEEE Security & Privacy* 11.2 (2013), pp. 77–79. doi: [10.1109/MSP.2013.25](https://doi.org/10.1109/MSP.2013.25) (cited on page 93).
- [89] Adéle Da Veiga and Jan HP Eloff. 'A framework and assessment instrument for information security culture'. In: *Computers & Security* 29.2 (2010), pp. 196–207. doi: [10.1016/j.cose.2009.09.002](https://doi.org/10.1016/j.cose.2009.09.002) (cited on page 68).
- [90] Duy Dang-Pham, Siddhi Pittayachawan, and Vince Bruno. 'Applications of social network analysis in behavioural information security research: Concepts and empirical analysis'. In: *Computers & Security* 68 (2017), pp. 1–15. doi: [10.1016/j.cose.2017.03.010](https://doi.org/10.1016/j.cose.2017.03.010) (cited on page 70).
- [91] Duy Dang-Pham, Siddhi Pittayachawan, and Vince Bruno. 'Applying network analysis to investigate interpersonal influence of information security behaviours in the workplace'. In: *Information & Management* 54.5 (2017), pp. 625–637. doi: [10.1016/j.im.2016.12.003](https://doi.org/10.1016/j.im.2016.12.003) (cited on page 70).
- [92] Duy Dang-Pham, Siddhi Pittayachawan, and Vince Bruno. 'Investigation into the formation of information security influence: Network analysis of an emerging organisation'. In: *Computers & Security* 70 (Sept. 2017), pp. 111–123. doi: [10.1016/j.cose.2017.05.010](https://doi.org/10.1016/j.cose.2017.05.010). (Visited on 04/03/2020) (cited on page 70).
- [93] *Deceived by design*. Forbrukerrådet. 2018. URL: <https://forbrukerradet.no/undersokelse/no-undersokelsekategori/deceived-by-design/> (visited on 09/2020) (cited on page 123).

- [94] Erik Derr, Sven Bugiel, Sascha Fahl, Yasemin Acar, and Michael Backes. 'Keep Me Updated: An Empirical Study of Third-Party Library Updatability on Android'. In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. CCS '17. Dallas, Texas, USA: ACM, 2017, pp. 2187–2200. doi: [10.1145/3133956.3134059](https://doi.org/10.1145/3133956.3134059) (cited on pages 20, 27).
- [95] *Developer Survey Results*. StackOverflow. 2019. URL: <https://insights.stackoverflow.com/survey/2019> (visited on 08/2019) (cited on pages 39, 102).
- [96] *Developer Survey Results*. Stack Overflow. 2020. URL: <https://insights.stackoverflow.com/survey/2020> (visited on 09/2020) (cited on page 73).
- [97] Google Developers. *Application Fundamentals | Android Developers*. 2019. URL: <https://developer.android.com/guide/components/fundamentals> (visited on 09/2020) (cited on page 123).
- [98] Google Developers. *Manifest.permission*. 2019. URL: <https://developer.android.com/reference/android/Manifest.permission.html> (visited on 09/2019) (cited on page 60).
- [99] Linda Di Geronimo, Larissa Braz, Enrico Fregnan, Fabio Palomba, and Alberto Bacchelli. 'UI Dark Patterns and Where to Find Them: A Study on Mobile Applications and User Perception'. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. Honolulu, HI, USA: Association for Computing Machinery, 2020, pp. 1–14. doi: [10.1145/3313831.3376600](https://doi.org/10.1145/3313831.3376600) (cited on pages 68, 123).
- [100] *Digital Advertising Alliance (DAA)*. Digital Advertising Alliance (DAA). 2021. URL: <https://digitaladvertisingalliance.org> (visited on 02/2021) (cited on page 140).
- [101] Lisa Nguyen Quang Do, Karim Ali, Benjamin Livshits, Eric Bodden, Justin Smith, and Emerson Murphy-Hill. 'Just-in-time Static Analysis'. In: *Proceedings of the 26th ACM SIGSOFT International Symposium on Software Testing and Analysis*. ISSTA 2017. Santa Barbara, CA, USA: ACM, 2017, pp. 307–317. doi: [10.1145/3092703.3092705](https://doi.org/10.1145/3092703.3092705) (cited on pages 20, 28, 32, 38).
- [102] Paul Dourish and Ken Anderson. 'Collective Information Practice: Exploring Privacy and Security as Social and Cultural Phenomena'. In: *Human-Computer Interaction* 21.3 (2006), pp. 319–342. doi: [10.1207/s15327051hci2103_2](https://doi.org/10.1207/s15327051hci2103_2) (cited on page 38).
- [103] Nora A Draper and Joseph Turow. 'The corporate cultivation of digital resignation'. In: *New Media & Society* 21.8 (2019), pp. 1824–1839. doi: [10.1177/1461444819833331](https://doi.org/10.1177/1461444819833331) (cited on page 68).

- [104] Line Dubé and Guy Paré. ‘Rigor in Information Systems Positivist Case Research: Current Practices, Trends, and Recommendations’. In: *MIS Quarterly* 27.4 (Nov. 2003), pp. 597–635. DOI: [10.2307/30036550](https://doi.org/10.2307/30036550) (cited on pages 17, 19, 22).
- [105] Anne Edmundson, Brian Holtkamp, Emanuel Rivera, Matthew Finifter, Adrian Mettler, and David Wagner. ‘An Empirical Study on the Effectiveness of Security Code Review’. In: *Proceedings of the 5th International Conference on Engineering Secure Software and Systems*. ESSoS’13. Paris, France: Springer-Verlag, 2013, pp. 197–212. DOI: [10.1007/978-3-642-36563-8_14](https://doi.org/10.1007/978-3-642-36563-8_14) (cited on pages 20, 26, 30).
- [106] Manuel Egele, David Brumley, Yanick Fratantonio, and Christopher Kruegel. ‘An Empirical Study of Cryptographic Misuse in Android Applications’. In: *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security*. CCS ’13. Berlin, Germany: ACM, 2013, pp. 73–84. DOI: [10.1145/2508859.2516693](https://doi.org/10.1145/2508859.2516693) (cited on pages 14, 38, 100, 101, 112, 115).
- [107] Serge Egelman, Marian Harbach, and Eyal Peer. ‘Behavior Ever Follows Intention? A Validation of the Security Behavior Intentions Scale (SeBIS)’. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2016, pp. 5257–5261. DOI: [10.1145/2858036.2858265](https://doi.org/10.1145/2858036.2858265) (cited on page 145).
- [108] Anirudh Ekambaranathan, Jun Zhao, and Max Van Kleek. ‘Understanding Value and Design Choices Made by Android Family App Developers’. In: *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI EA ’20. Honolulu, HI, USA: Association for Computing Machinery, 2020, pp. 1–10. DOI: [10.1145/3334480.3383064](https://doi.org/10.1145/3334480.3383064) (cited on pages 122, 138, 140, 141).
- [109] Pardis Emami-Naeini, Yuvraj Agarwal, Lorrie Faith Cranor, and Hanan Hibshi. ‘Ask the Experts: What Should Be on an IoT Privacy and Security Label?’ In: *2020 IEEE Symposium on Security and Privacy (SP)*. 2020, pp. 447–464. DOI: [10.1109/SP40000.2020.00043](https://doi.org/10.1109/SP40000.2020.00043) (cited on page 161).
- [110] Stack Exchange. *Stack Exchange Data Explorer*. 2019. URL: <https://data.stackexchange.com> (visited on 09/2019) (cited on page 42).
- [111] *Explore our participant pool demographics*. Prolific. 2021. URL: <https://www.prolific.co/demographics/> (visited on 02/2021) (cited on page 146).
- [112] Sascha Fahl, Marian Harbach, Thomas Muders, Lars Baumgärtner, Bernd Freisleben, and Matthew Smith. ‘Why Eve and Mallory Love Android: An Analysis of Android SSL (in)Security’. In: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*. CCS ’12. Raleigh, North Carolina, USA: Association for Computing Machinery, 2012, pp. 50–61. DOI: [10.1145/2382196.2382205](https://doi.org/10.1145/2382196.2382205) (cited on page 102).

- [113] Sascha Fahl, Marian Harbach, Henning Perl, Markus Koetter, and Matthew Smith. 'Rethinking SSL Development in an Appified World'. In: *Proceedings of the 2013 ACM SIGSAC Conference on Computer & Communications Security. CCS '13*. Berlin, Germany: Association for Computing Machinery, 2013, pp. 49–60. DOI: [10.1145/2508859.2516655](https://doi.org/10.1145/2508859.2516655) (cited on pages 4, 20, 30, 41, 101, 102, 112).
- [114] Pietro Ferrara and Fausto Spoto. 'Static Analysis for GDPR Compliance'. In: *Proceedings of the Second Italian Conference on Cyber Security (ITASEC 2018), Milan, Italy*. Vol. 2058. CEUR Workshop Proceedings. CEUR-WS.org, 2018. URL: <http://ceur-ws.org/Vol-2058/paper-10.pdf> (cited on page 69).
- [115] Felix Fischer, Konstantin Böttinger, Huang Xiao, Christian Stransky, Yasemin Acar, Michael Backes, and Sascha Fahl. 'Stack Overflow Considered Harmful? The Impact of Copy Paste on Android Application Security'. In: *2017 IEEE Symposium on Security and Privacy (SP)*. May 2017, pp. 121–136. DOI: [10.1109/SP.2017.31](https://doi.org/10.1109/SP.2017.31) (cited on pages 39, 101).
- [116] Robert J Fisher. 'Social desirability bias and the validity of indirect questioning'. In: *Journal of consumer research* 20.2 (1993), pp. 303–315. DOI: [10.1086/209351](https://doi.org/10.1086/209351) (cited on page 145).
- [117] Dan Fitton and Janet C. Read. 'Creating a Framework to Support the Critical Consideration of Dark Design Aspects in Free-to-Play Apps'. In: *Proceedings of the 18th ACM International Conference on Interaction Design and Children. IDC '19*. Boise, ID, USA: Association for Computing Machinery, 2019, pp. 407–418. DOI: [10.1145/3311927.3323136](https://doi.org/10.1145/3311927.3323136) (cited on page 123).
- [118] Brian Fitzgerald and Klaas-Jan Stol. 'Continuous software engineering: A roadmap and agenda'. In: *Journal of Systems and Software* 123 (2017), pp. 176–189. DOI: [10.1016/j.jss.2015.06.063](https://doi.org/10.1016/j.jss.2015.06.063) (cited on page 115).
- [119] Linda Flower and John R. Hayes. 'A Cognitive Process Theory of Writing'. In: *College Composition and Communication* 32.4 (1981), pp. 365–387. DOI: [10.2307/356600](https://doi.org/10.2307/356600) (cited on page 32).
- [120] Imane Fouad, Cristiana Santos, Feras Al Kassar, Nataliia Bielova, and Stefano Calzavara. 'On Compliance of Cookie Purposes with the Purpose Specification Principle'. In: *IWPE 2020 - International Workshop on Privacy Engineering*. Genova, Italy, Sept. 2020, pp. 1–8. URL: <https://hal.inria.fr/hal-02567022> (cited on pages 133, 140, 157).
- [121] Rita Francese, Carmine Gravino, Michele Risi, Giuseppe Scanniello, and Genoveffa Tortora. 'Mobile App Development and Management: Results from a Qualitative Investigation'. In: *Proceedings of the 4th International Conference on Mobile Software Engineering and Systems. MOBILESoft '17*. Buenos Aires, Argentina: IEEE Press, 2017, pp. 133–143. DOI: [10.1109/MOBILESoft.2017.33](https://doi.org/10.1109/MOBILESoft.2017.33) (cited on page 70).

- [122] Alisa Frik, Leysan Nurgalieva, Julia Bernd, Joyce S. Lee, Florian Schaub, and Serge Egelman. 'Privacy and Security Threat Models and Mitigation Strategies of Older Adults'. In: *Proceedings of the Fifteenth USENIX Conference on Usable Privacy and Security*. SOUPS'19. Santa Clara, CA, USA: USENIX Association, Aug. 2019, pp. 21–40. URL: <https://www.usenix.org/conference/soups2019/presentation/frik> (cited on page 68).
- [123] Kelsey R. Fulton, Rebecca Gelles, Alexandra McKay, Yasmin Abdi, Richard Roberts, and Michelle L. Mazurek. 'The Effect of Entertainment Media on Mental Models of Computer Security'. In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. Santa Clara, CA: USENIX Association, Aug. 2019. URL: <https://www.usenix.org/conference/soups2019/presentation/fulton> (cited on page 114).
- [124] Steven Furnell, Pete Fischer, and Amanda Finch. 'Can't get the staff? The growing need for cyber-security skills'. In: *Computer Fraud & Security 2017.2* (2017), pp. 5–10. DOI: [10.1016/S1361-3723\(17\)30013-1](https://doi.org/10.1016/S1361-3723(17)30013-1) (cited on page 116).
- [125] Trevor Gabriel and Steven Furnell. 'Selecting security champions'. In: *Computer Fraud & Security 2011.8* (2011), pp. 8–12. DOI: [10.1016/S1361-3723\(11\)70082-3](https://doi.org/10.1016/S1361-3723(11)70082-3) (cited on page 70).
- [126] Simson Garfinkel and Heather Richter Lipford. 'Usable Security: History, Themes, and Challenges'. In: *Synthesis Lectures on Information Security, Privacy, and Trust 5.2* (2014), pp. 1–124. DOI: [10.2200/S00594ED1V01Y201408SPT011](https://doi.org/10.2200/S00594ED1V01Y201408SPT011) (cited on page 3).
- [127] *General Data Protection Regulation (GDPR)*. The European parliament and the council of the European union. 2018. URL: <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679> (visited on 09/2020) (cited on pages 4, 33, 69, 111, 124, 140).
- [128] Martin Georgiev, Subodh Iyengar, Suman Jana, Rishita Anubhai, Dan Boneh, and Vitaly Shmatikov. 'The Most Dangerous Code in the World: Validating SSL Certificates in Non-Browser Software'. In: *Proceedings of the 2012 ACM Conference on Computer and Communications Security*. CCS '12. Raleigh, North Carolina, USA: Association for Computing Machinery, 2012, pp. 38–49. DOI: [10.1145/2382196.2382204](https://doi.org/10.1145/2382196.2382204) (cited on pages 4, 41, 100–102, 112, 131).
- [129] Phillipa Gill, Vijay Erramilli, Augustin Chaintreau, Balachander Krishnamurthy, Konstantina Papagiannaki, and Pablo Rodriguez. 'Follow the Money: Understanding Economics of Online Aggregation and Advertising'. In: *Proceedings of the 2013 Conference on Internet Measurement Conference*. IMC '13. Barcelona, Spain: Association for Computing Machinery, 2013, pp. 141–148. DOI: [10.1145/2504730.2504768](https://doi.org/10.1145/2504730.2504768) (cited on page 143).

- [130] Avi Goldfarb. ‘What is Different About Online Advertising?’ In: *Review of Industrial Organization* 44.2 (Mar. 2014), pp. 115–129. doi: [10.1007/s11151-013-9399-3](https://doi.org/10.1007/s11151-013-9399-3) (cited on page 138).
- [131] Google. *Google diversity annual report*. 2018. URL: <http://diversity.google/annual-report> (visited on 09/2019) (cited on page 104).
- [132] Google. *Privacy & Terms*. 2019. URL: <https://policies.google.com/terms> (visited on 09/2019) (cited on page 60).
- [133] Peter Leo Gorski, Luigi Lo Iacono, Dominik Wermke, Christian Stransky, Sebastian Möller, Yasemin Acar, and Sascha Fahl. ‘Developers Deserve Security Warnings, Too: On the Effect of Integrated Security Advice on Cryptographic API Misuse’. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 265–281. URL: <https://www.usenix.org/conference/soups2018/presentation/gorski> (cited on pages 20, 28).
- [134] Colin M. Gray, Yubo Kou, Bryan Battles, Joseph Hoggatt, and Austin L. Toombs. ‘The Dark (Patterns) Side of UX Design’. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI ’18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–14. doi: [10.1145/3173574.3174108](https://doi.org/10.1145/3173574.3174108) (cited on pages 68, 123, 138, 141, 232).
- [135] Matthew Green and Matthew Smith. ‘Developers Are Not the Enemy!: The Need for Usable Security APIs’. In: *IEEE Security and Privacy* 14.5 (Sept. 2016), pp. 40–46. doi: [10.1109/MSP.2016.111](https://doi.org/10.1109/MSP.2016.111) (cited on pages 4, 14, 38, 100, 101, 114, 140, 157).
- [136] Daniel Greene and Katie Shilton. ‘Platform privacies: Governance, collaboration, and the different meanings of “privacy” in iOS and Android development’. In: *New Media & Society* 20.4 (2018), pp. 1640–1657. doi: [10.1177/1461444817702397](https://doi.org/10.1177/1461444817702397) (cited on pages 40, 41, 60).
- [137] Jiaping Gui, Stuart Mcilroy, Meiyappan Nagappan, and William G. J. Halfond. ‘Truth in Advertising: The Hidden Cost of Mobile Ads for Software Developers’. In: *Proceedings of the 37th International Conference on Software Engineering - Volume 1*. ICSE ’15. Florence, Italy: IEEE Press, 2015, pp. 100–110. doi: [10.1109/ICSE.2015.32](https://doi.org/10.1109/ICSE.2015.32) (cited on page 160).
- [138] Seda Gürses, Carmela Troncoso, and Claudia Diaz. ‘Engineering privacy by design’. In: *Computers, Privacy & Data Protection* 14.3 (2011), p. 25. URL: <https://software.imdea.org/~carmela.troncoso/papers/Gurses-CPDP11.pdf> (cited on page 40).
- [139] Hana Habib, Sarah Pearman, Jiamin Wang, Yixin Zou, Alessandro Acquisti, Lorrie Faith Cranor, Norman Sadeh, and Florian Schaub. ‘“It’s a Scavenger Hunt”: Usability of Websites’ Opt-Out and Data Deletion Choices’. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*.

- CHI '20. Honolulu, HI, USA: Association for Computing Machinery, 2020, pp. 1–12. doi: [10.1145/3313831.3376511](https://doi.org/10.1145/3313831.3376511) (cited on page 138).
- [140] Hana Habib, Yixin Zou, Aditi Jannu, Neha Sridhar, Chelse Swoopes, Alessandro Acquisti, Lorrie Faith Cranor, Norman Sadeh, and Florian Schaub. 'An Empirical Analysis of Data Deletion and Opt-Out Choices on 150 Websites'. In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. Santa Clara, CA: USENIX Association, Aug. 2019. URL: <https://www.usenix.org/conference/soups2019/presentation/habib> (cited on page 140).
- [141] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. 'Privacy by designers: software developers' privacy mindset'. In: *Empirical Software Engineering* 23.1 (Feb. 2018), pp. 259–289. doi: [10.1007/s10664-017-9517-1](https://doi.org/10.1007/s10664-017-9517-1) (cited on pages 33, 38, 40, 41, 58, 59, 68, 69, 73, 77, 90, 107).
- [142] Tracy Hall, Helen Sharp, Sarah Beecham, Nathan Baddoo, and Hugh Robinson. 'What Do We Know about Developer Motivation?' In: *IEEE Software* 25.4 (2008), pp. 92–94. doi: [10.1109/MS.2008.105](https://doi.org/10.1109/MS.2008.105) (cited on page 91).
- [143] Catherine Han, Irwin Reyes, Álvaro Feal, Joel Reardon, Primal Wijesekera, Amit Elazari, Kenneth A Bamberger, and Serge Egelman. 'The Price is (Not) Right: Comparing Privacy in Free and Paid Apps'. In: *Privacy Enhancing Technologies Symposium (PETS 2020)*. 2020, p. 21. doi: [10.2478/popets-2020-0050](https://doi.org/10.2478/popets-2020-0050) (cited on pages 122, 123, 138, 140).
- [144] Julie M Haney and Wayne G Lutters. 'Skills and Characteristics of Successful Cybersecurity Advocates'. In: *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. USENIX Association, 2017, p. 7. URL: <https://www.usenix.org/conference/soups2017/workshop-program/wsiw2017/haney> (cited on pages 70, 83, 90, 170).
- [145] Julie M. Haney and Wayne G. Lutters. 'The Work of Cybersecurity Advocates'. In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA '17. Denver, Colorado, USA: Association for Computing Machinery, 2017, pp. 1663–1670. doi: [10.1145/3027063.3053134](https://doi.org/10.1145/3027063.3053134) (cited on pages 70, 170).
- [146] Julie M. Haney and Wayne G. Lutters. "'It's Scary. . . It's Confusing. . . It's Dull": How Cybersecurity Advocates Overcome Negative Perceptions of Security'. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 411–425. URL: <https://www.usenix.org/conference/soups2018/presentation/haney-perceptions> (cited on pages 68–73, 90, 91).

- [147] Julie M. Haney and Wayne G. Lutters. ‘Motivating Cybersecurity Advocates: Implications for Recruitment and Retention’. In: *Proceedings of the 2019 on Computers and People Research Conference*. SIGMIS-CPR ’19. Nashville, TN, USA: Association for Computing Machinery, 2019, pp. 109–117. doi: [10.1145/3322385.3322388](https://doi.org/10.1145/3322385.3322388) (cited on pages 70–73, 90, 92, 170).
- [148] Julie M. Haney, Mary Theofanos, Yasemin Acar, and Sandra Spickard Prettyman. “‘We make it a big deal in the company’”: Security Mindsets in Organizations that Develop Cryptographic Products’. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 357–373. URL: <https://www.usenix.org/conference/soups2018/presentation/haney-mindsets> (cited on pages 20, 25, 115).
- [149] Julie M. Haney, Mary Theofanos, Yasemin Acar, and Sandra Spickard Prettyman. “‘We make it a big deal in the company’”: Security Mindsets in Organizations that Develop Cryptographic Products’. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 357–373. URL: <https://www.usenix.org/conference/soups2018/presentation/haney-mindsets> (cited on page 70).
- [150] Hamza Harkous, Kassem Fawaz, Rémi Leuret, Florian Schaub, Kang G. Shin, and Karl Aberer. ‘Polisis: Automated Analysis and Presentation of Privacy Policies Using Deep Learning’. In: *27th USENIX Security Symposium (USENIX Security 18)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 531–548. URL: <https://www.usenix.org/conference/usenixsecurity18/presentation/harkous> (cited on page 59).
- [151] Boyuan He, Haitao Xu, Ling Jin, Guanyu Guo, Yan Chen, and Guangyao Weng. ‘An Investigation into Android In-App Ad Practice: Implications for App Developers’. In: *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*. 2018, pp. 2465–2473. doi: [10.1109/INFOCOM.2018.8486010](https://doi.org/10.1109/INFOCOM.2018.8486010) (cited on pages 122, 138, 140).
- [152] *Health Insurance Portability and Accountability Act of 1996 (HIPAA)*. U.S. Department of Health & Human Services. 1996. URL: <https://www.cdc.gov/phlp/publications/topic/hipaa.html> (visited on 02/2021) (cited on page 156).
- [153] Michael S.H Heng, Eileen M Trauth, and Sven J Fischer. ‘Organisational champions of IT innovation’. In: *Accounting, Management and Information Technologies* 9.3 (1999), pp. 193–222. doi: [10.1016/S0959-8022\(99\)00008-9](https://doi.org/10.1016/S0959-8022(99)00008-9) (cited on page 71).
- [154] Maximilian Hils, Daniel W. Woods, and Rainer Böhme. ‘Measuring the Emergence of Consent Management on the Web’. In: *Proceedings of the ACM Internet Measurement Conference*. IMC ’20. Virtual Event, USA: Association

- for Computing Machinery, 2020, pp. 317–332. doi: [10.1145/3419394.3423647](https://doi.org/10.1145/3419394.3423647) (cited on page 140).
- [155] Michael Hilton, Nicholas Nelson, Timothy Tunnell, Darko Marinov, and Danny Dig. ‘Trade-Offs in Continuous Integration: Assurance, Security, and Flexibility’. In: *Proceedings of the 2017 11th Joint Meeting on Foundations of Software Engineering*. ESEC/FSE 2017. Paderborn, Germany: Association for Computing Machinery, 2017, pp. 197–207. doi: [10.1145/3106237.3106270](https://doi.org/10.1145/3106237.3106270) (cited on pages 19, 20, 22, 26).
- [156] Scott A. Hissam, Daniel Plakosh, and C Weinstock. ‘Trust and vulnerability in open source software’. In: *IEEE Proceedings - Software* 149.1 (2002), pp. 47–51. doi: [10.1049/ip-sen:20020208](https://doi.org/10.1049/ip-sen:20020208) (cited on page 113).
- [157] Jaap-Henk Hoepman. *Privacy Design Strategies (The Little Blue Book)*. Radboud University, 2019. url: <https://cs.ru.nl/~jhh/publications/pds-booklet.pdf> (cited on pages 38, 40, 56, 62, 68, 124).
- [158] Lance Hoffman, Diana Burley, and Costis Torgas. ‘Holistically Building the Cybersecurity Workforce’. In: *IEEE Security & Privacy* 10.2 (2012), pp. 33–39. doi: [10.1109/MSP.2011.181](https://doi.org/10.1109/MSP.2011.181) (cited on page 116).
- [159] Christopher Horn and Anita D’Amico. ‘Measuring Application Security’. In: *Advances in Human Factors in Cybersecurity*. Ed. by Tareq Z. Ahram and Denise Nicholson. Vol. 782. Cham: Springer International Publishing, 2019, pp. 44–55. doi: [10.1007/978-3-319-94782-2_5](https://doi.org/10.1007/978-3-319-94782-2_5). (Visited on 04/03/2020) (cited on page 70).
- [160] Jane M Howell and Christine M Shea. ‘Individual differences, environmental scanning, innovation framing, and champion behavior: key predictors of project performance’. In: *Journal of Product Innovation Management* 18.1 (2001), pp. 15–27. doi: [10.1016/S0737-6782\(00\)00067-9](https://doi.org/10.1016/S0737-6782(00)00067-9) (cited on pages 78, 90).
- [161] Nasif Imtiaz, Akond Rahman, Effat Farhana, and Laurie Williams. ‘Challenges with Responding to Static Analysis Tool Alerts’. In: *Proceedings of the 16th International Conference on Mining Software Repositories*. MSR ’19. Montreal, Quebec, Canada: IEEE Press, 2019, pp. 245–249. doi: [10.1109/MSR.2019.00049](https://doi.org/10.1109/MSR.2019.00049) (cited on page 39).
- [162] Soumya Indela, Mukul Kulkarni, Kartik Nayak, and Tudor Dumitras. ‘Toward Semantic Cryptography APIs’. In: *2016 IEEE Cybersecurity Development (SecDev)*. 2016, pp. 9–14. doi: [10.1109/SecDev.2016.014](https://doi.org/10.1109/SecDev.2016.014) (cited on page 100).
- [163] Shubham Jain and Janne Lindqvist. ‘Should I Protect You? Understanding Developers’ Behavior to Privacy-Preserving APIs’. In: *Workshop on Usable Security (USEC’14)*. Internet Society, 2014. doi: [10.14722/usec.2014.23045](https://doi.org/10.14722/usec.2014.23045) (cited on pages 20, 22, 29, 41, 68, 122, 131).

- [164] Ling Jin, Boyuan He, Guangyao Weng, Haitao Xu, Yan Chen, and Guanyu Guo. 'MAdLens: Investigating into Android In-App Ad Practice at API Granularity'. In: *IEEE Transactions on Mobile Computing* PP.PP (2019), pp. 1–1. doi: [10.1109/TMC.2019.2953609](https://doi.org/10.1109/TMC.2019.2953609) (cited on pages 122, 138, 140).
- [165] Brittany Johnson, Yoonki Song, Emerson Murphy-Hill, and Robert Bowdidge. 'Why Don't Software Developers Use Static Analysis Tools to Find Bugs?' In: *Proceedings of the 2013 International Conference on Software Engineering*. ICSE '13. San Francisco, CA, USA: IEEE Press, 2013, pp. 672–681. doi: [10.1109/ICSE.2013.6606613](https://doi.org/10.1109/ICSE.2013.6606613) (cited on pages 20, 27, 28, 38, 131).
- [166] Eric J Johnson, Steven Bellman, and Gerald L Lohse. 'Defaults, Framing and Privacy: Why Opting In-Opting Out¹'. In: *Marketing letters* 13.1 (Feb. 2002), pp. 5–15. doi: [10.1023/A:1015044207315](https://doi.org/10.1023/A:1015044207315) (cited on pages 138, 141).
- [167] Keith Jones, Akbar Siami Namin, and Miriam Armstrong. 'What Should Cybersecurity Students Learn in School? Results from Interviews with Cyber Professionals (Abstract Only)'. In: *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*. SIGCSE '17. Seattle, Washington, USA: Association for Computing Machinery, 2017, p. 711. doi: [10.1145/3017680.3022407](https://doi.org/10.1145/3017680.3022407) (cited on page 102).
- [168] Donna Kelley and Hyunsuk Lee. 'Managing Innovation Champions: The Impact of Project Characteristics on the Direct Manager Role*'. In: *Journal of Product Innovation Management* 27.7 (2010), pp. 1007–1019. doi: [10.1111/j.1540-5885.2010.00767.x](https://doi.org/10.1111/j.1540-5885.2010.00767.x) (cited on page 71).
- [169] Patrick Gage Kelley, Joanna Bresee, Lorrie Faith Cranor, and Robert W. Reeder. 'A "Nutrition Label" for Privacy'. In: *Proceedings of the 5th Symposium on Usable Privacy and Security*. SOUPS '09. Mountain View, California, USA: Association for Computing Machinery, 2009. doi: [10.1145/1572532.1572538](https://doi.org/10.1145/1572532.1572538) (cited on pages 159, 161).
- [170] Brian W Kernighan and Dennis M Ritchie. *The C Programming Language*. Prentice Hall, 2006 (cited on page 113).
- [171] Andrew J. Ko, Robin Abraham, Laura Beckwith, Alan Blackwell, Margaret Burnett, Martin Erwig, Chris Scaffidi, Joseph Lawrance, Henry Lieberman, Brad Myers, Mary Beth Rosson, Gregg Rothermel, Mary Shaw, and Susan Wiedenbeck. 'The State of the Art in End-User Software Engineering'. In: *ACM Computing Surveys* 43.3 (Apr. 2011). doi: [10.1145/1922649.1922658](https://doi.org/10.1145/1922649.1922658) (cited on pages 14, 168).
- [172] Andrew J. Ko, Thomas D. Latoza, and Margaret M. Burnett. 'A Practical Guide to Controlled Experiments of Software Engineering Tools with Human Participants'. In: *Empirical Software Engineering* 20.1 (Feb. 2015), pp. 110–141. doi: [10.1007/s10664-013-9279-3](https://doi.org/10.1007/s10664-013-9279-3) (cited on pages 17, 19, 21, 34).

- [173] Spyros Kokolakis. ‘Privacy attitudes and privacy behaviour: A review of current research on the privacy paradox phenomenon’. In: *Computers & Security* 64 (2017), pp. 122–134. doi: [10.1016/j.cose.2015.07.002](https://doi.org/10.1016/j.cose.2015.07.002) (cited on page 145).
- [174] Bert-Jaap Koops, Jaap-Henk Hoepman, and Ronald Leenes. ‘Open-source intelligence and privacy by design’. In: *Computer Law & Security Review* 29.6 (2013), pp. 676–688. doi: [10.1016/j.clsr.2013.09.005](https://doi.org/10.1016/j.clsr.2013.09.005) (cited on page 40).
- [175] Kat Krol, Jonathan M. Spring, Simon Parkin, and M. Angela Sasse. ‘Towards Robust Experimental Design for User Studies in Security and Privacy’. In: *The LASER Workshop: Learning from Authoritative Security Experiment Results (LASER 2016)*. San Jose, CA: USENIX Association, May 2016, pp. 21–31. url: <https://www.usenix.org/conference/laser2016/program/presentation/krol> (cited on pages 17, 19).
- [176] J. Richard Landis and Gary G. Koch. ‘The Measurement of Observer Agreement for Categorical Data’. In: *Biometrics* 33.1 (1977), pp. 159–174. doi: [10.2307/2529310](https://doi.org/10.2307/2529310) (cited on page 73).
- [177] Marc Langheinrich. ‘Privacy by Design — Principles of Privacy-Aware Ubiquitous Systems’. In: *Ubicomp 2001: Ubiquitous Computing*. Ed. by Gregory D. Abowd, Barry Brumitt, and Steven Shafer. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 273–291. doi: [10.1007/3-540-45427-6_23](https://doi.org/10.1007/3-540-45427-6_23) (cited on page 33).
- [178] David Lazar, Haogang Chen, Xi Wang, and Nickolai Zeldovich. ‘Why Does Cryptographic Software Fail? A Case Study and Open Problems’. In: *Proceedings of 5th Asia-Pacific Workshop on Systems*. APSys ’14. Beijing, China: Association for Computing Machinery, 2014. doi: [10.1145/2637166.2637237](https://doi.org/10.1145/2637166.2637237) (cited on page 101).
- [179] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. ‘Chapter 11 - Analyzing qualitative data’. In: *Research Methods in Human Computer Interaction*. Ed. by Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. Second Edition. Boston: Morgan Kaufmann, 2017, pp. 299–327. doi: [10.1016/B978-0-12-805390-4.00011-X](https://doi.org/10.1016/B978-0-12-805390-4.00011-X) (cited on page 73).
- [180] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. ‘Chapter 8 - Interviews and focus groups’. In: *Research Methods in Human Computer Interaction*. Ed. by Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. Second Edition. Boston: Morgan Kaufmann, 2017, pp. 187–228. doi: [10.1016/B978-0-12-805390-4.00008-X](https://doi.org/10.1016/B978-0-12-805390-4.00008-X) (cited on pages 22, 46, 47, 145).
- [181] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research Methods in Human-Computer Interaction*. Morgan Kaufmann, 2017 (cited on page 105).

- [182] Pedro Leon, Blase Ur, Richard Shay, Yang Wang, Rebecca Balebako, and Lorrie Cranor. 'Why Johnny Can't Opt out: A Usability Evaluation of Tools to Limit Online Behavioral Advertising'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: Association for Computing Machinery, 2012, pp. 589–598. DOI: [10.1145/2207676.2207759](https://doi.org/10.1145/2207676.2207759) (cited on page 138).
- [183] Ilias Leontiadis, Christos Efstratiou, Marco Picone, and Cecilia Mascolo. 'Don't Kill My Ads! Balancing Privacy in an Ad-Supported Mobile Application Market'. In: *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*. HotMobile '12. San Diego, California: Association for Computing Machinery, 2012. DOI: [10.1145/2162081.2162084](https://doi.org/10.1145/2162081.2162084) (cited on pages 122, 138, 140, 160).
- [184] He Li, Lu Yu, and Wu He. 'The Impact of GDPR on Global Technology Development'. In: *Journal of Global Information Technology Management* 22.1 (2019), pp. 1–6. DOI: [10.1080/1097198X.2019.1569186](https://doi.org/10.1080/1097198X.2019.1569186) (cited on page 69).
- [185] Tianshi Li, Yuvraj Agarwal, and Jason I. Hong. 'Coconut: An IDE Plugin for Developing Privacy-Friendly Apps'. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.4 (Dec. 2018). DOI: [10.1145/3287056](https://doi.org/10.1145/3287056) (cited on pages 41, 60, 161).
- [186] Tianshi Li, Elizabeth Louie, Laura Dabbish, and Jason I. Hong. 'How Developers Talk About Personal Data and What It Means for User Privacy: A Case Study of a Developer Forum on Reddit'. In: *Proc. ACM Hum.-Comput. Interact.* 4.CSCW3 (Jan. 2021). DOI: [10.1145/3432919](https://doi.org/10.1145/3432919) (cited on pages 138, 140, 141, 143, 145, 158).
- [187] Yuanchun Li, Fanglin Chen, Toby Jia-Jun Li, Yao Guo, Gang Huang, Matthew Fredrikson, Yuvraj Agarwal, and Jason I. Hong. 'PrivacyStreams: Enabling Transparency in Personal Data Processing for Mobile Apps'. In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1.3 (Sept. 2017), 76:1–76:26. DOI: [10.1145/3130941](https://doi.org/10.1145/3130941) (cited on page 41).
- [188] Gaoqi Liang, Steven Weller, Junhua Zhao, Fengji Luo, and Z. Dong. 'The 2015 Ukraine Blackout: Implications for False Data Injection Attacks'. In: *IEEE Transactions on Power Systems* 32.4 (2017), pp. 3317–3318. DOI: [10.1109/TPWRS.2016.2631891](https://doi.org/10.1109/TPWRS.2016.2631891) (cited on page 14).
- [189] Ilaria Liccardi, Monica Bulger, Hal Abelson, Daniel Weitzner, and Wendy Mackay. 'Can apps play by the COPPA Rules?' In: *2014 Twelfth Annual International Conference on Privacy, Security and Trust*. 2014, pp. 1–9. DOI: [10.1109/PST.2014.6890917](https://doi.org/10.1109/PST.2014.6890917) (cited on pages 133, 157).

- [190] Dirk van der Linden, Pauline Anthonysamy, Bashar Nuseibeh, Thein Than Tun, Marian Petre, Mark Levine, John Towse, and Awais Rashid. ‘Schrödinger’s Security: Opening the Box on App Developers’ Security Rationale’. In: *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering*. ICSE ’20. Seoul, South Korea: Association for Computing Machinery, 2020, pp. 149–160. DOI: [10.1145/3377811.3380394](https://doi.org/10.1145/3377811.3380394) (cited on page 122).
- [191] Bin Liu, Mads Schaarup Andersen, Florian Schaub, Hazim Almuhammedi, Shikun (Aerin) Zhang, Norman Sadeh, Yuvraj Agarwal, and Alessandro Acquisti. ‘Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions’. In: *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, June 2016, pp. 27–41. URL: <https://www.usenix.org/conference/soups2016/technical-sessions/presentation/liu> (cited on page 161).
- [192] Luigi Lo Iacono and Peter Leo Gorski. ‘I Do and I Understand. Not Yet True for Security APIs. So Sad’. In: *European Workshop on Usable Security*. Apr. 2017. DOI: [10.14722/eurosec.2017.23015](https://doi.org/10.14722/eurosec.2017.23015) (cited on pages 20, 30, 101, 102).
- [193] Tamara Lopez, Thein Tun, Arosha Bandara, Mark Levine, Bashar Nuseibeh, and Helen Sharp. ‘An Anatomy of Security Conversations in Stack Overflow’. In: *Proceedings of the 41st International Conference on Software Engineering: Software Engineering in Society*. ICSE-SEIS ’10. Montreal, Quebec, Canada: IEEE Press, 2019, pp. 31–40. DOI: [10.1109/ICSE-SEIS.2019.00012](https://doi.org/10.1109/ICSE-SEIS.2019.00012) (cited on pages 38, 39).
- [194] Tamara Lopez, Thein T. Tun, Arosha Bandara, Mark Levine, Bashar Nuseibeh, and Helen Sharp. ‘An Investigation of Security Conversations in Stack Overflow: Perceptions of Security and Community Involvement’. In: *Proceedings of the 1st International Workshop on Security Awareness from Design to Deployment*. SEAD ’18. Gothenburg, Sweden: ACM, 2018, pp. 26–32. DOI: [10.1145/3194707.3194713](https://doi.org/10.1145/3194707.3194713) (cited on page 39).
- [195] Daniel Lüdecke, Dominique Makowski, Philip Waggoner, and Indrajeet Patil. *performance: Assessment of Regression Models Performance*. R package. 2020. DOI: [10.5281/zenodo.3952174](https://doi.org/10.5281/zenodo.3952174) (cited on page 148).
- [196] Janosch Maier., Arne Padmos., Mortaza S. Bargh., and Wolfgang Wörndl. ‘Influence of Mental Models on the Design of Cyber Security Dashboards’. In: *Proceedings of the 12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 3: IVAPP, (VISIGRAPP 2017)*. INSTICC. SciTePress, 2017, pp. 128–139. DOI: [10.5220/0006170901280139](https://doi.org/10.5220/0006170901280139) (cited on page 114).

- [197] Miguel Malheiros, Charlene Jennett, Sneha Patel, Sacha Brostoff, and Martina Angela Sasse. 'Too Close for Comfort: A Study of the Effectiveness and Acceptability of Rich-Media Personalized Advertising'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: Association for Computing Machinery, 2012, pp. 579–588. DOI: [10.1145/2207676.2207758](https://doi.org/10.1145/2207676.2207758) (cited on pages 138, 140, 158).
- [198] Lena Mamykina, Bella Manoim, Manas Mittal, George Hripcsak, and Björn Hartmann. 'Design Lessons from the Fastest Q&A Site in the West'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. Vancouver, BC, Canada: ACM, 2011, pp. 2857–2866. DOI: [10.1145/1978942.1979366](https://doi.org/10.1145/1978942.1979366) (cited on page 38).
- [199] Steve Mansfield-Devine. 'The Ashley Madison affair'. In: *Network Security* 2015.9 (2015), pp. 8–16. DOI: [10.1016/S1353-4858\(15\)30080-5](https://doi.org/10.1016/S1353-4858(15)30080-5) (cited on page 100).
- [200] Miriam Marciel, Jose Gonzalez, Yonas Mitike Kassa, Roberto Gonzalez, and Mohamed Ahmed. 'The Value of Online Users: Empirical Evaluation of the Price of Personalized Ads'. In: *2016 11th International Conference on Availability, Reliability and Security (ARES)*. 2016, pp. 694–700. DOI: [10.1109/ARES.2016.89](https://doi.org/10.1109/ARES.2016.89) (cited on page 138).
- [201] Heike Märki, Miriam Maas, Michaela Kauer-Franz, and Marius Oberle. 'Increasing Software Security by Using Mental Models'. In: *Advances in Human Factors in Cybersecurity*. Ed. by Denise Nicholson. Cham: Springer International Publishing, 2016, pp. 347–359. DOI: [10.1007/978-3-319-41932-9_29](https://doi.org/10.1007/978-3-319-41932-9_29) (cited on page 114).
- [202] Guy Martin, Paul Martin, Chris Hankin, Ara Darzi, and James Kinross. 'Cybersecurity and healthcare: how safe are we?' In: *BMJ* (2017). DOI: [10.1136/bmj.j3179](https://doi.org/10.1136/bmj.j3179) (cited on page 14).
- [203] Arunesh Mathur, Gunes Acar, Michael J. Friedman, Elena Lucherini, Jonathan Mayer, Marshini Chetty, and Arvind Narayanan. 'Dark Patterns at Scale: Findings from a Crawl of 11K Shopping Websites'. In: *Proceedings of the ACM on Human-Computer Interaction* 3.CSCW (Nov. 2019). DOI: [10.1145/3359183](https://doi.org/10.1145/3359183) (cited on pages 123, 138).
- [204] Arunesh Mathur, Jonathan Mayer, and Mihir Kshirsagar. 'What Makes a Dark Pattern... Dark? Design Attributes, Normative Considerations, and Measurement Methods'. In: CHI '21 (2021), pp. 1–27. DOI: [10.1145/3411764.3445610](https://doi.org/10.1145/3411764.3445610) (cited on pages 8, 171).
- [205] Arunesh Mathur, Jessica Vitak, Arvind Narayanan, and Marshini Chetty. 'Characterizing the Use of Browser-Based Blocking Extensions To Prevent Online Tracking'. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 103–116.

URL: <https://www.usenix.org/conference/soups2018/presentation/mathur> (cited on page 140).

- [206] Celestin Matte, Nataliia Bielova, and Cristiana Santos. 'Do Cookie Banners Respect my Choice? : Measuring Legal Compliance of Banners from IAB Europe's Transparency and Consent Framework'. In: *2020 IEEE Symposium on Security and Privacy (SP)*. May 2020, pp. 791–809. doi: [10.1109/SP40000.2020.00076](https://doi.org/10.1109/SP40000.2020.00076) (cited on pages 133, 140, 157).
- [207] Roy Maxion. 'Making Experiments Dependable'. In: *Dependable and Historic Computing: Essays Dedicated to Brian Randell on the Occasion of His 75th Birthday*. Ed. by Cliff B. Jones and John L. Lloyd. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 344–357. doi: [10.1007/978-3-642-24541-1_26](https://doi.org/10.1007/978-3-642-24541-1_26) (cited on pages 17, 19).
- [208] Michelle L. Mazurek, J. P. Arsenault, Joanna Bresee, Nitin Gupta, Iulia Ion, Christina Johns, Daniel Lee, Yuan Liang, Jenny Olsen, Brandon Salmon, Richard Shay, Kami Vaniea, Lujjo Bauer, Lorrie Faith Cranor, Gregory R. Ganger, and Michael K. Reiter. 'Access Control for Home Data Sharing: Attitudes, Needs and Practices'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2010, pp. 645–654. doi: [10.1145/1753326.1753421](https://doi.org/10.1145/1753326.1753421) (cited on page 114).
- [209] Michael Meng, Stephanie Steinhardt, and Andreas Schubert. 'Application Programming Interface Documentation: What Do Software Developers Want?' In: *Journal of Technical Writing and Communication* 48.3 (2018), pp. 295–330. doi: [10.1177/0047281617721853](https://doi.org/10.1177/0047281617721853) (cited on page 123).
- [210] Abraham H. Mhaidli, Yixin Zou, and Florian Schaub. "'We Can't Live Without Them!' App Developers' Adoption of Ad Networks and Their Considerations of Consumer Risks'. In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. Santa Clara, CA: USENIX Association, Aug. 2019. URL: <https://www.usenix.org/conference/soups2019/presentation/mhaidli> (cited on pages 38, 63).
- [211] Abraham H. Mhaidli, Yixin Zou, and Florian Schaub. "'We Can't Live Without Them!' App Developers' Adoption of Ad Networks and Their Considerations of Consumer Risks'. In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*. Santa Clara, CA: USENIX Association, Aug. 2019. URL: <https://www.usenix.org/conference/soups2019/presentation/mhaidli> (cited on pages 122–124, 131, 138, 140, 141, 160).
- [212] Matthew Miles and Michael Huberman. *Qualitative Data Analysis: A Methods Sourcebook*. Sage, 1994 (cited on page 105).

- [213] Kai Mindermann, Philipp Keck, and Stefan Wagner. ‘How Usable Are Rust Cryptography APIs?’ In: *2018 IEEE International Conference on Software Quality, Reliability and Security (QRS)*. IEEE, July 2018, pp. 143–154. DOI: [10.1109/qrs.2018.00028](https://doi.org/10.1109/qrs.2018.00028) (cited on page 157).
- [214] Kai Mindermann and Stefan Wagner. ‘Usability and Security Effects of Code Examples on Crypto APIs’. In: *2018 16th Annual Conference on Privacy, Security and Trust (PST)*. IEEE, Aug. 2018, pp. 1–2. DOI: [10.1109/PST.2018.8514203](https://doi.org/10.1109/PST.2018.8514203) (cited on page 157).
- [215] Kai Mindermann and Stefan Wagner. ‘Fluid Intelligence Doesn’t Matter! Effects of Code Examples on the Usability of Crypto APIs’. In: *Proceedings of the ACM/IEEE 42nd International Conference on Software Engineering: Companion Proceedings. ICSE ’20*. Seoul, South Korea: Association for Computing Machinery, 2020, pp. 306–307. DOI: [10.1145/3377812.3390892](https://doi.org/10.1145/3377812.3390892) (cited on page 157).
- [216] Miro | Online Whiteboard for Visual Collaboration. Miro. 2021. URL: <https://miro.com/> (visited on 02/2021) (cited on page 145).
- [217] *Mixed Effects Logistic Regression*. UCLA: Statistical Consulting Group. 2020. URL: <https://stats.idre.ucla.edu/stata/dae/mixed-effects-logistic-regression/> (visited on 02/2021) (cited on page 145).
- [218] *MoPub Integration Suite*. Twitter MoPub. 2021. URL: <https://developers.mopub.com/publishers/integrate/> (visited on 02/2021) (cited on page 142).
- [219] Lymari Morales. *U.S. Internet Users Ready to Limit Online Tracking for Ads*. Gallup Polls. 2010. URL: <https://news.gallup.com/poll/145337/internet-users-ready-limit-online-tracking-ads.aspx> (visited on 02/2021) (cited on pages 121, 159).
- [220] *Most popular Apple App Store categories in August 2020, by share of available apps*. Statista. 2020. URL: <https://www.statista.com/statistics/270291/popular-categories-in-the-app-store/> (visited on 02/2021) (cited on page 142).
- [221] *Most popular Google Play app categories as of 3rd quarter 2020, by share of available apps*. Statista. 2020. URL: <https://www.statista.com/statistics/279286/google-play-android-app-categories/> (visited on 02/2021) (cited on page 142).
- [222] Xenia Mountrouidou, David Vosen, Chadi Kari, Mohammad Q. Azhar, Sajal Bhatia, Greg Gagne, Joseph Maguire, Liviana Tudor, and Timothy T. Yuen. ‘Securing the Human: A Review of Literature on Broadening Diversity in Cybersecurity Education’. In: *Proceedings of the Working Group Reports on Innovation and Technology in Computer Science Education. ITiCSE-WGR ’19*. Aberdeen, Scotland UK: Association for Computing Machinery, 2019, pp. 157–176. DOI: [10.1145/3344429.3372507](https://doi.org/10.1145/3344429.3372507) (cited on page 83).

- [223] Deborah Mrazek and Michael Rafeld. 'Integrating Human Factors on a Large Scale: Product Usability Champions'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '92. Monterey, California, USA: Association for Computing Machinery, 1992, pp. 565–570. doi: [10.1145/142750.142989](https://doi.org/10.1145/142750.142989) (cited on page 70).
- [224] Ken Munro. *Hacking kettles & extracting plain text WPA PSKs. Yes really!* 2015. URL: <https://www.pentestpartners.com/security-blog/hacking-kettles-extracting-plain-text-wpa-psks-yes-really> (visited on 08/2019) (cited on pages 14, 100).
- [225] Patrick Murmann. 'Usable Transparency for Enhancing Privacy in Mobile Health Apps'. In: *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct*. MobileHCI '18. Barcelona, Spain: Association for Computing Machinery, 2018, pp. 440–442. doi: [10.1145/3236112.3236184](https://doi.org/10.1145/3236112.3236184) (cited on page 161).
- [226] Brad Myers, Amy Ko, Thomas LaToza, and YoungSeok Yoon. 'Programmers Are Users Too: Human-Centered Methods for Improving Programming Tools'. In: *Computer* 49.07 (July 2016), pp. 44–52. doi: [10.1109/MC.2016.200](https://doi.org/10.1109/MC.2016.200) (cited on page 14).
- [227] Sarah Nadi, Stefan Krüger, Mira Mezini, and Eric Bodden. 'Jumping through Hoops: Why Do Java Developers Struggle with Cryptography APIs?' In: *Proceedings of the 38th International Conference on Software Engineering*. ICSE '16. Austin, Texas: Association for Computing Machinery, 2016, pp. 935–946. doi: [10.1145/2884781.2884790](https://doi.org/10.1145/2884781.2884790) (cited on pages 20, 22, 26, 30, 38, 39, 101).
- [228] Alena Naiakshina, Anastasia Danilova, Eva Gerlitz, Emanuel von Zezschwitz, and Matthew Smith. "'If You Want, I Can Store the Encrypted Password": A Password-Storage Field Study with Freelance Developers'. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland UK: Association for Computing Machinery, 2019, pp. 1–12. doi: [10.1145/3290605.3300370](https://doi.org/10.1145/3290605.3300370) (cited on page 32).
- [229] Alena Naiakshina, Anastasia Danilova, Christian Tiefenau, Marco Herzog, Sergej Dechand, and Matthew Smith. 'Why Do Developers Get Password Storage Wrong? A Qualitative Usability Study'. In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. CCS '17. Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 311–328. doi: [10.1145/3133956.3134082](https://doi.org/10.1145/3133956.3134082) (cited on pages 20, 29, 31, 32).
- [230] Alena Naiakshina, Anastasia Danilova, Christian Tiefenau, and Matthew Smith. 'Deception Task Design in Developer Password Studies: Exploring a Student Sample'. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 297–313.

- URL: <https://www.usenix.org/conference/soups2018/presentation/naiakshina> (cited on pages 20, 30–32).
- [231] Arvind Narayanan, Arunesh Mathur, Marshini Chetty, and Mihir Kshirsagar. ‘Dark Patterns: Past, Present, and Future’. In: *Queue* 18.2 (Apr. 2020), pp. 67–92. DOI: [10.1145/3400899.3400901](https://doi.org/10.1145/3400899.3400901) (cited on pages 123, 138).
- [232] Seyed Mehdi Nasehi, Jonathan Sillito, Frank Maurer, and Chris Burns. ‘What makes a good code example?: A study of programming Q A in StackOverflow’. In: *2012 28th IEEE International Conference on Software Maintenance (ICSM)*. Sept. 2012, pp. 25–34. DOI: [10.1109/ICSM.2012.6405249](https://doi.org/10.1109/ICSM.2012.6405249) (cited on pages 39, 40, 49).
- [233] Duc Cuong Nguyen, Dominik Wermke, Yasemin Acar, Michael Backes, Charles Weir, and Sascha Fahl. ‘A Stitch in Time: Supporting Android Developers in Writing Secure Code’. In: *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. CCS ’17*. Dallas, Texas, USA: Association for Computing Machinery, 2017, pp. 1065–1077. DOI: [10.1145/3133956.3133977](https://doi.org/10.1145/3133956.3133977) (cited on pages 20, 28, 31, 33, 38).
- [234] Seth James Nielson. ‘PLAYGROUND: preparing students for the cyber battleground’. In: *Computer Science Education* 26.4 (2016), pp. 255–276. DOI: [10.1080/08993408.2016.1271526](https://doi.org/10.1080/08993408.2016.1271526) (cited on page 102).
- [235] Helen Nissenbaum. ‘Privacy as contextual integrity’. In: *Wash. L. Rev.* 79 (2004), p. 119 (cited on page 159).
- [236] Helen Nissenbaum. *Privacy in context: Technology, policy, and the integrity of social life*. Stanford University Press, 2009. URL: <http://www.sup.org/books/title/?id=8862> (cited on page 75).
- [237] Don Norman. *The Design of Everyday Things: Revised and Expanded Edition*. Basic Books, 2013 (cited on page 168).
- [238] Midas Nouwens, Ilaria Liccardi, Michael Veale, David Karger, and Lalana Kagal. ‘Dark Patterns after the GDPR: Scraping Consent Pop-Ups and Demonstrating Their Influence’. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems. CHI ’20*. Honolulu, HI, USA: Association for Computing Machinery, 2020, pp. 1–13. DOI: [10.1145/3313831.3376321](https://doi.org/10.1145/3313831.3376321) (cited on pages 123, 140, 141).
- [239] *Number of available applications in the Google Play Store from December 2009 to June 2020*. Statista. 2020. URL: <https://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/> (visited on 09/2020) (cited on page 122).
- [240] *Number of software developers worldwide in 2018, 2019, 2023 and 2024*. Statista. 2020. URL: <https://www.statista.com/statistics/627312/worldwide-developer-population/> (visited on 02/2021) (cited on page 138).

- [241] Katie O'Donnell and Henriette Cramer. 'People's Perceptions of Personalized Ads'. In: *Proceedings of the 24th International Conference on World Wide Web*. WWW '15 Companion. Florence, Italy: Association for Computing Machinery, 2015, pp. 1293–1298. doi: [10.1145/2740908.2742003](https://doi.org/10.1145/2740908.2742003) (cited on pages 138, 140, 158).
- [242] Jonathan A. Obar and Anne Oeldorf-Hirsch. 'The biggest lie on the Internet: ignoring the privacy policies and terms of service policies of social networking services'. In: *Information, Communication & Society* 23.1 (2020), pp. 128–147. doi: [10.1080/1369118X.2018.1486870](https://doi.org/10.1080/1369118X.2018.1486870) (cited on pages 124, 169).
- [243] Information Commissioner's Office. *Investigation into the use of data analytics in political campaigns*. 2018. URL: <https://ico.org.uk/media/action-weve-taken/2259371/investigation-into-data-analytics-for-political-purposes-update.pdf> (visited on 08/2019) (cited on page 107).
- [244] Chitu Okoli. 'A Guide to Conducting a Standalone Systematic Literature Review'. In: *Communications of the Association for Information Systems* 37.43 (Nov. 2015). doi: [10.17705/1CAIS.03743](https://doi.org/10.17705/1CAIS.03743) (cited on pages 15, 19).
- [245] Daniela Oliveira, Marissa Rosenthal, Nicole Morin, Kuo-Chuan Yeh, Justin Cappos, and Yanyan Zhuang. 'It's the Psychology Stupid: How Heuristics Explain Software Vulnerabilities and How Priming Can Illuminate Developer's Blind Spots'. In: *Proceedings of the 30th Annual Computer Security Applications Conference*. ACSAC '14. New Orleans, Louisiana, USA: Association for Computing Machinery, 2014, pp. 296–305. doi: [10.1145/2664243.2664254](https://doi.org/10.1145/2664243.2664254) (cited on pages 20, 29, 102).
- [246] Daniela Seabra Oliveira, Tian Lin, Muhammad Sajidur Rahman, Rad Akefirad, Donovan Ellis, Eliany Perez, Rahul Bobhate, Lois A. DeLong, Justin Cappos, and Yuriy Brun. 'API Blindspots: Why Experienced Developers Write Vulnerable Code'. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 315–328. URL: <https://www.usenix.org/conference/soups2018/presentation/oliveira> (cited on pages 20, 29).
- [247] Marten Oltrogge, Yasemin Acar, Sergej Dechand, Matthew Smith, and Sascha Fahl. 'To Pin or Not to Pin Helping App Developers Bullet Proof Their TLS Connections'. In: *Proceedings of the 24th USENIX Conference on Security Symposium*. SEC'15. Washington, D.C.: USENIX Association, 2015, pp. 239–254. URL: <https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/oltrogge> (cited on pages 20, 30).
- [248] Marten Oltrogge, Erik Derr, Christian Stransky, Yasemin Acar, Sascha Fahl, Christian Rossow, Giancarlo Pellegrino, Sven Bugiel, and Michael Backes. 'The Rise of the Citizen Developer: Assessing the Security Impact of Online

- App Generators'. In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 2018, pp. 634–647. doi: [10.1109/SP.2018.000005](https://doi.org/10.1109/SP.2018.000005) (cited on page 101).
- [249] *Out of Control - How consumers are exploited by the online advertising industry*. Forbrukerrådet. 2020. URL: <https://www.forbrukerradet.no/side/new-study-the-advertising-industry-is-systematically-breaking-the-law> (visited on 09/2020) (cited on pages 133, 140).
- [250] Stack Overflow. *About*. 2019. URL: <https://stackoverflow.com/company> (visited on 09/2019) (cited on page 39).
- [251] Stack Overflow. *What does it mean when an answer is "accepted"?* 2019. URL: <https://stackoverflow.com/help/accepted-answer> (visited on 09/2019) (cited on page 47).
- [252] OWASP. *Top 10 Most Critical Web Application Security Risks*. 2010. URL: <https://owasp.org/www-project-top-ten/> (visited on 08/2020) (cited on page 14).
- [253] OWASP. *The Ten Most Critical Web Application Security Risks*. 2017. URL: <https://owasp.org/www-project-top-ten/> (visited on 08/2020) (cited on pages 40, 101).
- [254] OWASP. *Security Champions*. 2019. URL: https://wiki.owasp.org/index.php/Security_Champions (visited on 03/2021) (cited on page 170).
- [255] Leysia Palen and Paul Dourish. 'Unpacking "Privacy" for a Networked World'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '03. Ft. Lauderdale, Florida, USA: Association for Computing Machinery, 2003, pp. 129–136. doi: [10.1145/642611.642635](https://doi.org/10.1145/642611.642635) (cited on page 75).
- [256] Chris Parnin, Christian Bird, and Emerson Murphy-Hill. 'Java Generics Adoption: How New Features Are Introduced, Championed, or Ignored'. In: *Proceedings of the 8th Working Conference on Mining Software Repositories*. MSR '11. Waikiki, Honolulu, HI, USA: Association for Computing Machinery, 2011, pp. 3–12. doi: [10.1145/1985441.1985446](https://doi.org/10.1145/1985441.1985446) (cited on page 70).
- [257] Chris Parnin, Christian Bird, and Emerson Murphy-Hill. 'Adoption and use of Java generics'. In: *Empirical Software Engineering* 18.6 (Dec. 2013), pp. 1047–1089. doi: [10.1007/s10664-012-9236-6](https://doi.org/10.1007/s10664-012-9236-6). (Visited on 04/03/2020) (cited on page 70).
- [258] Chris Parnin, Christoph Treude, Lars Grammel, and Margaret-Anne Storey. 'Crowd documentation: Exploring the coverage and the dynamics of API discussions on Stack Overflow'. In: *Georgia Institute of Technology, Technical Report 11* (2012). URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.371.6263> (cited on page 61).

- [259] Nikhil Patnaik, Joseph Hallett, and Awais Rashid. 'Usability Smells: An Analysis of Developers' Struggle With Crypto Libraries'. In: *Fifteenth Symposium on Usable Privacy and Security (SOUPS)*. USENIX Association, Aug. 2019. URL: <https://www.usenix.org/conference/soups2019/presentation/patnaik> (cited on pages 39, 114).
- [260] Sai Teja Peddinti, Igor Bilogrevic, Nina Taft, Martin Pelikan, Úlfar Erlingsson, Pauline Anthonysamy, and Giles Hogben. 'Reducing Permission Requests in Mobile Apps'. In: *Proceedings of the Internet Measurement Conference*. IMC '19. Amsterdam, Netherlands: Association for Computing Machinery, 2019, pp. 259–266. DOI: [10.1145/3355369.3355584](https://doi.org/10.1145/3355369.3355584) (cited on page 160).
- [261] Mariana Peixoto, Dayse Ferreira, Mateus Cavalcanti, Carla Silva, Jéssyka Vilela, João Araújo, and Tony Gorschek. 'On Understanding How Developers Perceive and Interpret Privacy Requirements Research Preview'. In: *Requirements Engineering: Foundation for Software Quality*. Ed. by Nazim Madhavji, Liliana Pasquale, Alessio Ferrari, and Stefania Gnesi. Cham: Springer International Publishing, 2020, pp. 116–123. DOI: [10.1007/978-3-030-44429-7_8](https://doi.org/10.1007/978-3-030-44429-7_8) (cited on page 73).
- [262] Giancarlo Pellegrino, Constantin Tschürtz, Eric Bodden, and Christian Rossow. 'jÄk: Using Dynamic Analysis to Crawl and Test Modern Web Applications'. In: *Research in Attacks, Intrusions, and Defenses*. Ed. by Herbert Bos, Fabian Monrose, and Gregory Blanc. Cham: Springer International Publishing, 2015, pp. 295–316. DOI: [10.1007/978-3-319-26362-5_14](https://doi.org/10.1007/978-3-319-26362-5_14) (cited on page 14).
- [263] *Permissions overview | Android Developers*. Google Developers. 2020. URL: <https://developer.android.com/guide/topics/permissions/overview> (visited on 09/2020) (cited on page 123).
- [264] Hiep Cong Pham, Linda Brennan, Lukas Parker, Nhat Tram Phan-Le, Irfan Ulhaq, Mathews Zanda Nkhoma, and Minh Nhat Nguyen. 'Enhancing cyber security behavior: an internal social marketing approach'. In: *Information & Computer Security* 28.2 (Oct. 2019), pp. 133–159. DOI: [10.1108/ICS-01-2019-0023](https://doi.org/10.1108/ICS-01-2019-0023) (cited on page 70).
- [265] Olgierd Pieczul, Simon Foley, and Mary Ellen Zurko. 'Developer-centered Security and the Symmetry of Ignorance'. In: *Proceedings of the 2017 New Security Paradigms Workshop*. NSPW 2017. Santa Cruz, CA, USA: ACM, 2017, pp. 46–56. DOI: [10.1145/3171533.3171539](https://doi.org/10.1145/3171533.3171539) (cited on pages 14, 38, 100, 101).
- [266] Angelisa C. Plane, Elissa M. Redmiles, Michelle L. Mazurek, and Michael Carl Tschantz. 'Exploring User Perceptions of Discrimination in Online Targeted Advertising'. In: *26th USENIX Security Symposium (USENIX Security 17)*. Vancouver, BC: USENIX Association, Aug. 2017, pp. 935–

951. URL: <https://www.usenix.org/conference/usenixsecurity17/technical-sessions/presentation/plane> (cited on pages 140, 158).
- [267] Andreas Poller, Laura Kocksch, Sven Türpe, Felix Anand Epp, and Katharina Kinder-Kurlanda. 'Can Security Become a Routine?: A Study of Organizational Change in an Agile Software Development Group'. In: *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing*. CSCW '17. Portland, Oregon, USA: ACM, 2017, pp. 2489–2503. DOI: [10.1145/2998181.2998191](https://doi.org/10.1145/2998181.2998191) (cited on pages 20, 23–25, 102).
- [268] Lutz Prechelt. 'Plat_Forms: A Web Development Platform Comparison by an Exploratory Experiment Searching for Emergent Platform Properties'. In: *IEEE Transactions on Software Engineering* 37.01 (Jan. 2011), pp. 95–108. DOI: [10.1109/TSE.2010.22](https://doi.org/10.1109/TSE.2010.22) (cited on pages 20, 30, 31).
- [269] Isaac Prilleltensky. 'Psychology and the status quo.' In: *American Psychologist* 44.5 (1989), pp. 795–802. DOI: [10.1037/0003-066X.44.5.795](https://doi.org/10.1037/0003-066X.44.5.795) (cited on pages 138, 141).
- [270] *Public Attitudes Towards Online Targeting*. Centre for Data Ethics and Innovation. 2020. URL: <https://www.gov.uk/government/publications/cdei-review-of-online-targeting> (visited on 02/2021) (cited on pages 138, 140, 158).
- [271] Alex Radermacher and Gursimran Walia. 'Gaps between Industry Expectations and the Abilities of Graduates'. In: *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*. SIGCSE '13. Denver, Colorado, USA: Association for Computing Machinery, 2013, pp. 525–530. DOI: [10.1145/2445196.2445351](https://doi.org/10.1145/2445196.2445351) (cited on page 102).
- [272] Alex Radermacher, Gursimran Walia, and Dean Knudson. 'Investigating the Skill Gap between Graduating Students and Industry Expectations'. In: *Companion Proceedings of the 36th International Conference on Software Engineering*. ICSE Companion 2014. Hyderabad, India: Association for Computing Machinery, 2014, pp. 291–300. DOI: [10.1145/2591062.2591159](https://doi.org/10.1145/2591062.2591159) (cited on page 102).
- [273] Chaiyong Ragkhitwetsagul, Jens Krinke, Matheus Paixao, Giuseppe Bianco, and Rocco Oliveto. 'Toxic Code Snippets on Stack Overflow'. In: *IEEE Transactions on Software Engineering* (2019), pp. 1–22. DOI: [10.1109/TSE.2019.2900307](https://doi.org/10.1109/TSE.2019.2900307) (cited on page 39).
- [274] Akond Rahman, Asif Partho, Patrick Morrison, and Laurie Williams. 'What Questions Do Programmers Ask About Configuration As Code?' In: *Proceedings of the 4th International Workshop on Rapid Continuous Software Engineering*. RCoSE '18. Gothenburg, Sweden: ACM, 2018, pp. 16–22. DOI: [10.1145/3194760.3194769](https://doi.org/10.1145/3194760.3194769) (cited on page 39).

- [275] Joel Reardon, Álvaro Feal, Primal Wijesekera, Amit Elazari Bar On, Narseo Vallina-Rodriguez, and Serge Egelman. '50 Ways to Leak Your Data: An Exploration of Apps' Circumvention of the Android Permissions System'. In: *28th USENIX Security Symposium (USENIX Security 19)*. Santa Clara, CA: USENIX Association, Aug. 2019, pp. 603–620. URL: <https://www.usenix.org/conference/usenixsecurity19/presentation/reardon> (cited on page 123).
- [276] Karen Renaud, Melanie Volkamer, and Arne Renkema-Padmos. 'Why Doesn't Jane Protect Her Privacy?' In: *Privacy Enhancing Technologies*. Ed. by Emiliano De Cristofaro and Steven J. Murdoch. Cham: Springer International Publishing, 2014, pp. 244–262. DOI: [10.1007/978-3-319-08506-7_13](https://doi.org/10.1007/978-3-319-08506-7_13) (cited on page 105).
- [277] Jaco Renken and Richard Richard. 'Champions of IS Innovations'. In: *Communications of the Association for Information Systems* 44 (2019), pp. 811–851. DOI: [10.17705/1CAIS.04438](https://doi.org/10.17705/1CAIS.04438) (cited on pages 7, 68, 70, 71, 78, 90).
- [278] Irwin Reyes, Primal Wijesekera, Joel Reardon, Amit Elazari Bar On, Abbas Razaghpanah, Narseo Vallina-Rodriguez, and Serge Egelman. "'Won't Somebody Think of the Children?'" Examining COPPA Compliance at Scale'. In: *Proceedings on Privacy Enhancing Technologies* 2018.3 (2018), pp. 63–83. DOI: [10.1515/popets-2018-0021](https://doi.org/10.1515/popets-2018-0021) (cited on pages 133, 157).
- [279] Martin P. Robillard and Robert Deline. 'A Field Study of API Learning Obstacles'. In: *Empirical Software Engineering* 16.6 (Dec. 2011), pp. 703–732. DOI: [10.1007/s10664-010-9150-8](https://doi.org/10.1007/s10664-010-9150-8) (cited on page 114).
- [280] Christoffer Rosen and Emad Shihab. 'What are mobile developers asking about? A large scale study using stack overflow'. In: *Empirical Software Engineering* 21.3 (June 2016), pp. 1192–1223. DOI: [10.1007/s10664-015-9379-3](https://doi.org/10.1007/s10664-015-9379-3) (cited on pages 38–40, 42, 49).
- [281] Per Runeson, Martin Host, Austen Rainer, and Bjorn Regnell. *Case Study Research in Software Engineering: Guidelines and Examples*. 1st. Wiley Publishing, 2012. URL: <https://dl.acm.org/doi/book/10.5555/2361717> (cited on pages 17, 19).
- [282] SAFECODE. *Software Security Takes a Champion - A Short Guide on Building and Sustaining a Successful Security Champions Program*. Tech. rep. SAFECODE, 2019. URL: <http://safecode.org/wp-content/uploads/2019/02/Security-Champions-2019-.pdf> (cited on page 70).
- [283] Takahito Sakamoto and Masahiro Matsunaga. 'After GDPR, Still Tracking or Not? Understanding Opt-Out States for Online Behavioral Advertising'. In: *2019 IEEE Security and Privacy Workshops (SPW)*. 2019, pp. 92–99. DOI: [10.1109/SPW.2019.00027](https://doi.org/10.1109/SPW.2019.00027) (cited on page 138).
- [284] Johnny Saldaña. *The Coding Manual for Qualitative Researchers*. Sage, 2015 (cited on page 105).

- [285] Neil Salkind. *Encyclopedia of Research Design*. SAGE Publications, Inc, June 2020 (cited on page 46).
- [286] William Samuelson and Richard Zeckhauser. 'Status quo bias in decision making'. In: *Journal of risk and uncertainty* 1.1 (Mar. 1988), pp. 7–59. doi: [10.1007/BF00055564](https://doi.org/10.1007/BF00055564) (cited on pages 138, 141).
- [287] M Angela Sasse and Ivan Flechais. 'Usable security: Why do we need it? How do we get it?' In: *Security and Usability Designing Secure Systems that People Can Use*. O'Reilly, 2005. URL: <https://discovery.ucl.ac.uk/id/eprint/20345/> (cited on page 14).
- [288] Mark Saunders, Philip Lewis, and Adrian. Thornhill. *Research Methods for Business Students*. Prentice Hall, 2012 (cited on page 21).
- [289] Yukiko Sawaya, Mahmood Sharif, Nicolas Christin, Ayumu Kubota, Akihiro Nakarai, and Akira Yamada. 'Self-Confidence Trumps Knowledge: A Cross-Cultural Study of Security Behavior'. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: Association for Computing Machinery, 2017, pp. 2202–2214. doi: [10.1145/3025453.3025926](https://doi.org/10.1145/3025453.3025926) (cited on page 34).
- [290] Gabe Scelta, Hamid Rashid, Hoi Wai Jackie Cheng, Marcelo LaFleur, Mariangela Parra-Lancourt, Alex Julca, Nicole Hunt, S. Islam, and Hiroshi Kawamura. 'Data Economy: Radical transformation or dystopia?' In: *Frontier Technology Quarterly* 1 (Jan. 2019). URL: https://www.un.org/development/desa/dpad/wp-content/uploads/sites/45/publication/FTQ_1_Jan_2019.pdf (cited on page 68).
- [291] Florian Schaub, Rebecca Balebako, and Lorrie Faith Cranor. 'Designing Effective Privacy Notices and Controls'. In: *IEEE Internet Computing* 21.3 (May 2017), pp. 70–77. doi: [10.1109/MIC.2017.75](https://doi.org/10.1109/MIC.2017.75) (cited on page 59).
- [292] Stuart Schechter. *Common Pitfalls in Writing about Security and Privacy Human Subjects Experiments, and How to Avoid Them*. Tech. rep. Microsoft, Jan. 2013. URL: <https://www.microsoft.com/en-us/research/publication/common-pitfalls-in-writing-about-security-and-privacy-human-subjects-experiments-and-how-to-avoid-them/> (cited on pages 17, 23).
- [293] Deborah H Schenk. 'Exploiting the Salience Bias in Designing Taxes'. In: *Yale J. on Reg.* 28 (2011), p. 253. doi: [10.2139/ssrn.1661322](https://doi.org/10.2139/ssrn.1661322) (cited on pages 138, 141, 143, 144).
- [294] Donald A. Schon. 'Champions for Radical New Inventions'. In: *Harvard Business Review* 41 (1963), pp. 77–86. URL: <https://id.lib.harvard.edu/ead/c/bak00203c02144/catalog> (cited on page 70).

- [295] Awanthika Senarath and Nalin A. G. Arachchilage. 'Why Developers Cannot Embed Privacy into Software Systems?: An Empirical Investigation'. In: *Proceedings of the 22Nd International Conference on Evaluation and Assessment in Software Engineering 2018*. EASE'18. Christchurch, New Zealand: ACM, 2018, pp. 211–216. DOI: [10.1145/3210459.3210484](https://doi.org/10.1145/3210459.3210484) (cited on pages 34, 40, 90).
- [296] Awanthika Senarath, Marthie Grobler, and Nalin Asanka Gamagedara Arachchilage. 'Will They Use It or Not? Investigating Software Developers' Intention to Follow Privacy Engineering Methodologies'. In: *ACM Transactions on Privacy and Security* 22.4 (Nov. 2019). DOI: [10.1145/3364224](https://doi.org/10.1145/3364224) (cited on page 94).
- [297] Awanthika R. Senarath and Nalin Asanka Gamagedara Arachchilage. 'Understanding user privacy expectations: A software developer's perspective'. In: *Telematics and Informatics* 35.7 (2018), pp. 1845–1862. DOI: [10.1016/j.tele.2018.05.012](https://doi.org/10.1016/j.tele.2018.05.012) (cited on page 34).
- [298] *Share of global smartphone shipments by operating system from 2014 to 2023*. Statista. 2020. URL: <https://www.statista.com/statistics/272307/market-share-forecast-for-smartphone-operating-systems/> (visited on 09/2020) (cited on pages 122, 138, 140).
- [299] Swapneel Sheth, Gail Kaiser, and Walid Maalej. 'Us and Them: A Study of Privacy Requirements Across North America, Asia, and Europe'. In: *Proceedings of the 36th International Conference on Software Engineering*. ICSE 2014. Hyderabad, India: ACM, 2014, pp. 859–870. DOI: [10.1145/2568225.2568244](https://doi.org/10.1145/2568225.2568244) (cited on pages 19, 20, 22, 26, 27, 140).
- [300] Katie Shilton and Daniel Greene. 'Linking Platforms, Practices, and Developer Ethics: Levers for Privacy Discourse in Mobile Application Development'. In: *Journal of Business Ethics* 155.1 (Mar. 2019), pp. 131–146. DOI: [10.1007/s10551-017-3504-8](https://doi.org/10.1007/s10551-017-3504-8) (cited on pages 41, 122, 140).
- [301] Katie Shilton, Donal Heidenblad, Adam Porter, Susan Winter, and Mary Kendig. 'Role-Playing Computer Ethics: Designing and Evaluating the Privacy by Design (PbD) Simulation'. In: *Science and Engineering Ethics* PP.PP (July 2020). DOI: [10.1007/s11948-020-00250-0](https://doi.org/10.1007/s11948-020-00250-0) (cited on page 93).
- [302] Laura Shipp and Jorge Blasco. 'How private is your period?: A systematic analysis of menstrual app privacy policies'. In: *Proceedings on Privacy Enhancing Technologies* 2020.4 (Oct. 2020), pp. 491–510. DOI: [10.2478/popets-2020-0083](https://doi.org/10.2478/popets-2020-0083) (cited on pages 133, 157).
- [303] Ashish Kumar Singh and Vidyasagar Potdar. 'Blocking Online Advertising - A State of the Art'. In: *Proceedings of the 2009 IEEE International Conference on Industrial Technology*. ICIT '09. USA: IEEE Computer Society, 2009, pp. 1–10. DOI: [10.1109/ICIT.2009.4939739](https://doi.org/10.1109/ICIT.2009.4939739) (cited on pages 121, 159).

- [304] H. Jeff Smith, Tamara Dinev, and Heng Xu. 'Information Privacy Research: An Interdisciplinary Review'. In: *MIS Quarterly* 35.4 (Dec. 2011), pp. 989–1016. DOI: [10.2307/41409970](https://doi.org/10.2307/41409970) (cited on page 40).
- [305] Justin Smith, Brittany Johnson, Emerson Murphy-Hill, Bill Chu, and Heather Richter Lipford. 'Questions Developers Ask While Diagnosing Potential Security Vulnerabilities with Static Analysis'. In: *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*. ESEC/FSE 2015. Bergamo, Italy: ACM, 2015, pp. 248–259. DOI: [10.1145/2786805.2786812](https://doi.org/10.1145/2786805.2786812) (cited on pages 20, 28).
- [306] Justin Smith, Brittany Johnson, Emerson Murphy-Hill, Bill Chu, and Heather Richter Lipford. 'How Developers Diagnose Potential Security Vulnerabilities with a Static Analysis Tool'. In: *IEEE Transactions on Software Engineering* 45.9 (Sept. 2019), pp. 877–897. DOI: [10.1109/TSE.2018.2810116](https://doi.org/10.1109/TSE.2018.2810116) (cited on page 33).
- [307] Daniela Soares Cruzes, Martin Gilje Jaatun, Karin Bernsmed, and Inger Anne Tondel. 'Challenges and Experiences with Applying Microsoft Threat Modeling in Agile Development Projects'. In: *2018 25th Australasian Software Engineering Conference (ASWEC)*. Adelaide, SA: IEEE, Nov. 2018, pp. 111–120. DOI: [10.1109/ASWEC.2018.00023](https://doi.org/10.1109/ASWEC.2018.00023). (Visited on 04/03/2020) (cited on page 70).
- [308] Daniel J Solove. 'A taxonomy of privacy'. In: *University of Pennsylvania Law Review* 154 (2005), pp. 477–560. URL: <https://ssrn.com/abstract=667622> (cited on pages 40, 62, 75).
- [309] Daniel J Solove. "'I've Got Nothing to Hide" and Other Misunderstandings of Privacy'. In: *San Diego Law Review* 44 (2007), p. 745. URL: <https://ssrn.com/abstract=998565> (cited on pages 80, 167).
- [310] Sooel Son, Daehyeok Kim, and Vitaly Shmatikov. 'What Mobile Ads Know About Mobile Users'. In: *Network and Distributed System Security Symposium (NDSS)*. 2016. DOI: [10.14722/ndss.2016.23407](https://doi.org/10.14722/ndss.2016.23407) (cited on page 123).
- [311] spaCy. *spaCy - Industrial-strength Natural Language Processing in Python*. 2019. URL: <https://spacy.io> (visited on 09/2019) (cited on page 45).
- [312] *Special Eurobarometer 431 "Data protection"*. Tech. rep. European Commission, 2015. URL: https://ec.europa.eu/commfrontoffice/publicopinion/archives/ebs/ebs_431_en.pdf (visited on 02/2021) (cited on pages 121, 159).
- [313] Chad Spensky, Jeffrey Stewart, Arkady Yerukhimovich, Richard Shay, Ari Trachtenberg, Rick Housley, and Robert K. Cunningham. 'SoK: Privacy on Mobile Devices – It's Complicated'. In: *Proceedings on Privacy Enhancing Technologies* 2016.3 (2016), pp. 96–116. DOI: [10.1515/popets-2016-0018](https://doi.org/10.1515/popets-2016-0018) (cited on page 123).

- [314] Sarah Spiekermann and Lorrie Faith Cranor. 'Engineering Privacy'. In: *IEEE Transactions on Software Engineering* 35.1 (Jan. 2009), pp. 67–82. doi: [10.1109/TSE.2008.88](https://doi.org/10.1109/TSE.2008.88) (cited on pages 40, 62).
- [315] Sarah Spiekermann, Jana Korunovska, and Marc Langheinrich. 'Inside the Organization: Why Privacy and Security Engineering Is a Challenge for Engineers'. In: *Proceedings of the IEEE* 107.3 (2019), pp. 600–615. doi: [10.1109/JPROC.2018.2866769](https://doi.org/10.1109/JPROC.2018.2866769) (cited on page 90).
- [316] Ryan Stevens, Clint Gibler, Jon Crussell, Jeremy Erickson, and Hao Chen. 'Investigating User Privacy in Android Ad Libraries'. In: *Workshop on Mobile Security Technologies (MoST)*. 2012. URL: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.298.7556> (cited on page 140).
- [317] Leigh Ann Sudol and Ciera Jaspán. 'Analyzing the Strength of Undergraduate Misconceptions about Software Engineering'. In: *Proceedings of the Sixth International Workshop on Computing Education Research*. ICER '10. Aarhus, Denmark: Association for Computing Machinery, 2010, pp. 31–40. doi: [10.1145/1839594.1839601](https://doi.org/10.1145/1839594.1839601) (cited on page 102).
- [318] Ruoxi Sun and Minhui Xue. 'Quality Assessment of Online Automated Privacy Policy Generators: An Empirical Study'. In: *Proceedings of the Evaluation and Assessment in Software Engineering*. EASE '20. Trondheim, Norway: Association for Computing Machinery, 2020, pp. 270–275. doi: [10.1145/3383219.3383247](https://doi.org/10.1145/3383219.3383247) (cited on page 131).
- [319] Yung Sype and Walid Maalej. 'On lawful disclosure of personal user data: What should app developers do?' In: *2014 IEEE 7th International Workshop on Requirements Engineering and Law (RELAW)*. 2014, pp. 25–34. doi: [10.1109/RELAW.2014.6893479](https://doi.org/10.1109/RELAW.2014.6893479) (cited on page 59).
- [320] Madiha Tabassum, Stacey Watson, Bill Chu, and Heather Richter Lipford. 'Evaluating Two Methods for Integrating Secure Programming Education'. In: *Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. SIGCSE '18. Baltimore, Maryland, USA: Association for Computing Machinery, 2018, pp. 390–395. doi: [10.1145/3159450.3159511](https://doi.org/10.1145/3159450.3159511) (cited on page 102).
- [321] Madiha Tabassum, Stacey Watson, and Heather Richter Lipford. 'Comparing Educational Approaches to Secure programming: Tool vs. TA'. In: *Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. Santa Clara, CA: USENIX Association, July 2017. URL: <https://www.usenix.org/conference/soups2017/workshop-program/wsiw2017/tabassum> (cited on pages 20, 29, 31, 33).
- [322] Mohammad Tahaei. "'I Don't Know Too Much About It": On the Security Mindsets of Future Software Creators'. In: *Proceedings of the 2019 ACM Conference on Innovation and Technology in Computer Science Education*. ITiCSE

- '19. Aberdeen, Scotland UK: Association for Computing Machinery, 2019, p. 350. doi: [10.1145/3304221.3325592](https://doi.org/10.1145/3304221.3325592) (cited on page viii).
- [323] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 'Deciding on Personalized Ads: Nudging Developers About User Privacy'. In: *Seventeenth Symposium on Usable Privacy and Security (SOUPS '21)*. USENIX Association, 2021, pp. 1–24 (cited on page 171).
- [324] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. 'Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges'. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21. New York, NY, USA: Association for Computing Machinery, 2021, pp. 1–15. doi: [10.1145/3411764.3445768](https://doi.org/10.1145/3411764.3445768) (cited on pages 122, 145).
- [325] Mohammad Tahaei, Adam Jenkins, Kami Vaniea, and Maria K. Wolters. "'I Don't Know Too Much About It": On the Security Mindsets of Computer Science Students. 9th International Workshop, STAST 2019, Luxembourg City, Luxembourg, September 26, 2019, Revised Selected Papers'. In: *Socio-Technical Aspects in Security and Trust*. Ed. by Thomas Groß and Tryfonas Theo. First Edition. Springer International Publishing, June 2021. doi: [10.1007/978-3-030-55958-8](https://doi.org/10.1007/978-3-030-55958-8) (cited on page 93).
- [326] Mohammad Tahaei and Kami Vaniea. 'A Survey on Developer-Centred Security'. In: *2019 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. June 2019, pp. 129–138. doi: [10.1109/EuroSPW.2019.00021](https://doi.org/10.1109/EuroSPW.2019.00021) (cited on pages 4, 6, 10, 42, 68–70, 92, 131).
- [327] Mohammad Tahaei and Kami Vaniea. "'Developers Are Responsible": What Ad Networks Tell Developers About Privacy'. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems Extended Abstracts*. CHI '21 Extended Abstracts. New York, NY, USA: Association for Computing Machinery, 2021, pp. 1–12. doi: [10.1145/3411763.3451805](https://doi.org/10.1145/3411763.3451805) (cited on pages 138–141, 160).
- [328] Mohammad Tahaei and Kami Vaniea. 'Code-Level Dark Patterns: Exploring Ad Networks' Misleading Code Samples with Negative Consequences for Users'. In: *What Can CHI Do About Dark Patterns? Workshop at CHI '21*. 2021, pp. 1–5. url: <http://hdl.handle.net/20.500.11820/ea71877b-4def-4c2c-aa45-e148122b4f36> (cited on page vii).
- [329] Mohammad Tahaei, Kami Vaniea, Beznosov Konstantin, and Maria K. Wolters. 'Security Notifications in Static Analysis Tools: Developers' Attitudes, Comprehension, and Ability to Act on Them'. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. CHI '21. New York, NY, USA: Association for Computing Machinery, 2021, pp. 1–17. doi: [10.1145/3411764.3445616](https://doi.org/10.1145/3411764.3445616) (cited on pages 4, 131, 146, 157).

- [330] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 'Understanding Privacy-Related Questions on Stack Overflow'. In: *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. CHI '20. Honolulu, HI, USA: Association for Computing Machinery, 2020, pp. 1–14. DOI: [10.1145/3313831.3376768](https://doi.org/10.1145/3313831.3376768) (cited on pages 70, 90, 122, 124, 131, 138, 140, 141, 145).
- [331] *The State of Mobile in 2020*. App Annie. 2020. URL: <https://www.appannie.com/en/insights/market-data/state-of-mobile-2020/> (visited on 09/2020) (cited on pages 122, 138, 140).
- [332] *The Value of Personalized Ads to a Thriving App Ecosystem*. Facebook. 2020. URL: <https://developers.facebook.com/blog/post/2020/06/18/value-of-personalized-ads-thriving-app-ecosystem/> (visited on 02/2021) (cited on page 139).
- [333] Tyler Thomas, Bill Chu, Heather Lipford, Justin Smith, and Emerson Murphy-Hill. 'A study of interactive code annotation for access control vulnerabilities'. In: *2015 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. Los Alamitos, CA, USA: IEEE Computer Society, Oct. 2015, pp. 73–77. DOI: [10.1109/VLHCC.2015.7357200](https://doi.org/10.1109/VLHCC.2015.7357200) (cited on pages 20, 28).
- [334] Tyler W. Thomas, Heather Lipford, Bill Chu, Justin Smith, and Emerson Murphy-Hill. 'What Questions Remain? An Examination of How Developers Understand an Interactive Static Analysis Tool'. In: *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, June 2016. URL: <https://www.usenix.org/conference/soups2016/workshop-program/wsiw16/presentation/thomas> (cited on pages 20, 27).
- [335] Tyler W. Thomas, Madiha Tabassum, Bill Chu, and Heather Lipford. 'Security During Application Development: An Application Security Expert Perspective'. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: ACM, 2018, 262:1–262:12. DOI: [10.1145/3173574.3173836](https://doi.org/10.1145/3173574.3173836) (cited on pages 20, 24–26, 68, 70, 102, 115).
- [336] Kerry-Lynn Thomson, Rossouw Von Solms, and Lynette Louw. 'Cultivating an organizational information security culture'. In: *Computer fraud & security 2006.10* (2006), pp. 7–11. DOI: [10.1016/S1361-3723\(06\)70430-4](https://doi.org/10.1016/S1361-3723(06)70430-4) (cited on page 68).
- [337] Inger Anne Tøndel, Martin Jaatun, and Daniela Cruzes. 'IT Security Is From Mars, Software Security Is From Venus'. In: *IEEE Security & Privacy* 18.04 (July 2020), pp. 48–54. DOI: [10.1109/MSEC.2020.2969064](https://doi.org/10.1109/MSEC.2020.2969064) (cited on pages 68–70).

- [338] Christoph Treude, Ohad Barzilay, and Margaret-Anne Storey. 'How Do Programmers Ask and Answer Questions on the Web? (NIER Track)'. In: *Proceedings of the 33rd International Conference on Software Engineering. ICSE '11*. Waikiki, Honolulu, HI, USA: ACM, 2011, pp. 804–807. doi: [10.1145/1985793.1985907](https://doi.org/10.1145/1985793.1985907) (cited on pages 38–40, 42, 45, 46, 49).
- [339] Omer Tripp, Salvatore Guarnieri, Marco Pistoia, and Aleksandr Aravkin. 'ALETHEIA: Improving the Usability of Static Security Analysis'. In: *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security. CCS '14*. Scottsdale, Arizona, USA: Association for Computing Machinery, 2014, pp. 762–774. doi: [10.1145/2660267.2660339](https://doi.org/10.1145/2660267.2660339) (cited on page 14).
- [340] Janice Y. Tsai, Serge Egelman, Lorrie Cranor, and Alessandro Acquisti. 'The Effect of Online Privacy Information on Purchasing Behavior: An Experimental Study'. In: *Info. Sys. Research* 22.2 (June 2011), pp. 254–268. doi: [10.1287/isre.1090.0260](https://doi.org/10.1287/isre.1090.0260) (cited on page 141).
- [341] Joseph Turow, Jennifer King, Chris Hoofnagle, Amy Bleakley, and Michael Hennessy. 'Americans Reject Tailored Advertising and Three Activities That Enable It'. In: (Sept. 2009). doi: [10.2139/ssrn.1478214](https://doi.org/10.2139/ssrn.1478214) (cited on pages 121, 159).
- [342] Sven Türpe, Laura Kocksch, and Andreas Poller. 'Penetration Tests a Turning Point in Security Practices? Organizational Challenges and Implications in a Software Development Team'. In: *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, June 2016. URL: <https://www.usenix.org/conference/soups2016/workshop-program/wsiw16/presentation/turpe> (cited on pages 20, 24).
- [343] Martin Ukrop and Vashek Matyas. 'Why Johnny the Developer Can't Work with Public Key Certificates'. In: *Topics in Cryptology – CT-RSA 2018*. Ed. by Nigel P. Smart. Cham: Springer International Publishing, 2018, pp. 45–64. doi: [10.1007/978-3-319-76953-0_3](https://doi.org/10.1007/978-3-319-76953-0_3) (cited on page 101).
- [344] Blase Ur, Pedro Giovanni Leon, Lorrie Faith Cranor, Richard Shay, and Yang Wang. 'Smart, Useful, Scary, Creepy: Perceptions of Online Behavioral Advertising'. In: *Proceedings of the Eighth Symposium on Usable Privacy and Security. SOUPS '12*. Washington, D.C.: Association for Computing Machinery, 2012. doi: [10.1145/2335356.2335362](https://doi.org/10.1145/2335356.2335362) (cited on page 138).
- [345] Akond Ashfaqur Rahman and Laurie Williams. 'Software Security in DevOps: Synthesizing Practitioners' Perceptions and Practices'. In: *Proceedings of the International Workshop on Continuous Software Evolution and Delivery. CSED '16*. Austin, Texas: Association for Computing Machinery, 2016, pp. 70–76. doi: [10.1145/2896941.2896946](https://doi.org/10.1145/2896941.2896946) (cited on pages 20, 26).

- [346] Christine Utz, Martin Degeling, Sascha Fahl, Florian Schaub, and Thorsten Holz. '(Un) Informed Consent: Studying GDPR Consent Notices in the Field'. In: *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. CCS '19. London, United Kingdom: Association for Computing Machinery, 2019, pp. 973–990. DOI: [10.1145/3319535.3354212](https://doi.org/10.1145/3319535.3354212) (cited on pages 133, 140, 157).
- [347] Max Van Kleek, Reuben Binns, Jun Zhao, Adam Slack, Sauyon Lee, Dean Ottewell, and Nigel Shadbolt. 'X-Ray Refine: Supporting the Exploration and Refinement of Information Exposure Resulting from Smartphone Apps'. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. Montreal QC, Canada: Association for Computing Machinery, 2018, pp. 1–13. DOI: [10.1145/3173574.3173967](https://doi.org/10.1145/3173574.3173967) (cited on page 161).
- [348] Max Van Kleek, Ilaria Liccardi, Reuben Binns, Jun Zhao, Daniel J. Weitzner, and Nigel Shadbolt. 'Better the Devil You Know: Exposing the Data Sharing Practices of Smartphone Apps'. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. CHI '17. Denver, Colorado, USA: Association for Computing Machinery, 2017, pp. 5208–5220. DOI: [10.1145/3025453.3025556](https://doi.org/10.1145/3025453.3025556) (cited on page 161).
- [349] Kami Vaniea and Yasmeeen Rashidi. 'Tales of Software Updates: The Process of Updating Software'. In: *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. CHI '16. San Jose, California, USA: Association for Computing Machinery, 2016, pp. 3215–3226. DOI: [10.1145/2858036.2858303](https://doi.org/10.1145/2858036.2858303) (cited on page 15).
- [350] Kami E. Vaniea, Emilee Rader, and Rick Wash. 'Betrayed by Updates: How Negative Experiences Affect Future Security'. In: *Proceedings of the 32Nd Annual ACM Conference on Human Factors in Computing Systems*. CHI '14. Toronto, Ontario, Canada: ACM, 2014, pp. 2671–2674. DOI: [10.1145/2556288.2557275](https://doi.org/10.1145/2556288.2557275) (cited on page 103).
- [351] Ismini Vasileiou and Steven Furnell. 'Personalising Security Education: Factors Influencing Individual Awareness and Compliance'. In: *Information Systems Security and Privacy*. Ed. by Paolo Mori, Steven Furnell, and Olivier Camp. Cham: Springer International Publishing, 2019, pp. 189–200. DOI: [10.1007/978-3-030-25109-3_10](https://doi.org/10.1007/978-3-030-25109-3_10) (cited on page 70).
- [352] Veracode. *State of Software Security*. Tech. rep. Veracode, CA Technologies, 2018. URL: <https://www.veracode.com/state-of-software-security-report/> (cited on page 14).
- [353] Daniel Votipka, Rock Stevens, Elissa M. Redmiles, Jeremy Hu, and Michelle L. Mazurek. 'Hackers vs. Testers: A Comparison of Software Vulnerability Discovery Processes'. In: *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, May 2018, pp. 374–391. DOI: [10.1109/SP.2018.00003](https://doi.org/10.1109/SP.2018.00003) (cited on page 70).

- [354] Vox. *The Cambridge Analytica Facebook scandal*. 2018. URL: <https://www.vox.com/2018/4/10/17207394> (visited on 08/2019) (cited on page 107).
- [355] Ben Wagner, Krisztina Rozgonyi, Marie-Therese Sekwenz, Jennifer Cobbe, and Jatinder Singh. 'Regulating Transparency? Facebook, Twitter and the German Network Enforcement Act'. In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. FAT* '20. Barcelona, Spain: Association for Computing Machinery, 2020, pp. 261–271. DOI: [10.1145/3351095.3372856](https://doi.org/10.1145/3351095.3372856) (cited on pages 123, 133).
- [356] Isabel Wagner and David Eckhoff. 'Technical Privacy Metrics: A Systematic Survey'. In: *ACM Computing Surveys* 51.3 (June 2018). DOI: [10.1145/3168389](https://doi.org/10.1145/3168389) (cited on page 94).
- [357] Ari Ezra Waldman. 'Designing without privacy'. In: *Houston Law Review* 55 (2018), p. 659. URL: <https://ssrn.com/abstract=2944185> (cited on pages 68, 69, 73).
- [358] Ari Ezra Waldman. 'Cognitive biases, dark patterns, and the 'privacy paradox''. In: *Current Opinion in Psychology* 31 (2020), pp. 105–109. DOI: <https://doi.org/10.1016/j.copsyc.2019.08.025> (cited on page 133).
- [359] Haoyu Wang, Zhe Liu, Yao Guo, Xiangqun Chen, Miao Zhang, Guoai Xu, and Jason Hong. 'An Explorative Study of the Mobile App Ecosystem from App Developers' Perspective'. In: *Proceedings of the 26th International Conference on World Wide Web*. WWW '17. Perth, Australia: International World Wide Web Conferences Steering Committee, 2017, pp. 163–172. DOI: [10.1145/3038912.3052712](https://doi.org/10.1145/3038912.3052712) (cited on page 146).
- [360] Ying Wang, Ebru Genc, and Gang Peng. 'Aiming the Mobile Targets in a Cross-Cultural Context: Effects of Trust, Privacy Concerns, and Attitude'. In: *International Journal of Human-Computer Interaction* 36.3 (2020), pp. 227–238. DOI: [10.1080/10447318.2019.1625571](https://doi.org/10.1080/10447318.2019.1625571) (cited on page 138).
- [361] Rick Wash. 'Folk Models of Home Computer Security'. In: *Proceedings of the Sixth Symposium on Usable Privacy and Security*. SOUPS '10. Redmond, Washington, USA: Association for Computing Machinery, 2010. DOI: [10.1145/1837110.1837125](https://doi.org/10.1145/1837110.1837125) (cited on pages 33, 103, 105, 109, 114).
- [362] Jaana Wäyrynen, Marine Bodén, and Gustav Boström. 'Security Engineering and eXtreme Programming: An Impossible Marriage?' In: *Extreme Programming and Agile Methods - XP/Agile Universe 2004*. Ed. by Carmen Zannier, Hakan Erdogmus, and Lowell Lindstrom. Berlin, Heidelberg, 2004, pp. 117–128. DOI: [10.1007/978-3-540-27777-4_12](https://doi.org/10.1007/978-3-540-27777-4_12) (cited on page 14).
- [363] Charles Weir, Ingolf Becker, James Noble, Lynne Blair, Angela Sasse, and Awais Rashid. 'Interventions for Software Security: Creating a Lightweight Program of Assurance Techniques for Developers'. In: *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering*

- in Practice (ICSE-SEIP). IEEE, 2019, pp. 41–50. doi: [10.1109/ICSE-SEIP.2019.00013](https://doi.org/10.1109/ICSE-SEIP.2019.00013) (cited on page 70).
- [364] Charles Weir, Ingolf Becker, James Noble, Lynne Blair, M. Angela Sasse, and Awais Rashid. ‘Interventions for long-term software security: Creating a lightweight program of assurance techniques for developers’. In: *Software: Practice and Experience* 50.3 (2020), pp. 275–298. doi: [10.1002/spe.2774](https://doi.org/10.1002/spe.2774) (cited on page 70).
- [365] Charles Weir, Ben Hermann, and Sascha Fahl. ‘From Needs to Actions to Secure Apps? The Effect of Requirements and Developer Practices on App Security’. In: *29th USENIX Security Symposium (USENIX Security 20)*. Boston, MA: USENIX Association, Aug. 2020. url: <https://www.usenix.org/conference/usenixsecurity20/presentation/weir> (cited on page 70).
- [366] Charles Weir, Awais Rashid, and James Noble. ‘How to Improve the Security Skills of Mobile App Developers? Comparing and Contrasting Expert Views’. In: *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*. Denver, CO: USENIX Association, June 2016. url: <https://www.usenix.org/conference/soups2016/workshop-program/wsiw16/presentation/weir> (cited on pages 20, 24, 102).
- [367] Sara J Weston, M Teresa Cardador, Patrick L Hill, Ted Schwaba, Jennifer Lodi-Smith, and Susan K Whitbourne. ‘The Relationship Between Career Success and Sense of Purpose: Examining Linkages and Changes’. In: *The Journals of Gerontology: Series B* PP (Sept. 2020). doi: [10.1093/geronb/gbaa162](https://doi.org/10.1093/geronb/gbaa162) (cited on page 91).
- [368] *Where Developers Learn, Share, & Build Careers*. Stack Overflow. 2019. url: <https://stackoverflow.com> (visited on 09/2019) (cited on page 38).
- [369] Michael Whitney, Heather Lipford-Richter, Bill Chu, and Jun Zhu. ‘Embedding Secure Coding Instruction into the IDE: A Field Study in an Advanced CS Course’. In: *Proceedings of the 46th ACM Technical Symposium on Computer Science Education*. SIGCSE ’15. Kansas City, Missouri, USA: Association for Computing Machinery, 2015, pp. 60–65. doi: [10.1145/2676723.2677280](https://doi.org/10.1145/2676723.2677280) (cited on pages 20, 29, 102).
- [370] Alma Whitten and J. D. Tygar. ‘Why Johnny Can’t Encrypt: A Usability Evaluation of PGP 5.0’. In: *Proceedings of the 8th Conference on USENIX Security Symposium - Volume 8*. SSYM’99. Washington, D.C.: USENIX Association, 1999, p. 14. url: <https://www.usenix.org/conference/8th-usenix-security-symposium/why-johnny-cant-encrypt-usability-evaluation-pgp-50> (cited on page 3).
- [371] John E. Williams, J. Kenneth Morland, and Walter L. Underwood. ‘Connotations of Color Names in the United States, Europe, and Asia’. In: *The Journal of Social Psychology* 82.1 (1970), pp. 3–14. doi: [10.1080/00224545.1970.9919925](https://doi.org/10.1080/00224545.1970.9919925) (cited on page 228).

- [372] Jim Witschey, Shundan Xiao, and Emerson Murphy-Hill. 'Technical and Personal Factors Influencing Developers' Adoption of Security Tools'. In: *Proceedings of the 2014 ACM Workshop on Security Information Workers*. SIW '14. Scottsdale, Arizona, USA: Association for Computing Machinery, 2014, pp. 23–26. DOI: [10.1145/2663887.2663898](https://doi.org/10.1145/2663887.2663898) (cited on pages 20, 23).
- [373] Jim Witschey, Olga Zielinska, Allaire Welk, Emerson Murphy-Hill, Chris Mayhorn, and Thomas Zimmermann. 'Quantifying Developers' Adoption of Security Tools'. In: *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*. ESEC/FSE 2015. Bergamo, Italy: Association for Computing Machinery, 2015, pp. 260–271. DOI: [10.1145/2786805.2786816](https://doi.org/10.1145/2786805.2786816) (cited on pages 20, 23, 27).
- [374] Richmond Y. Wong and Deirdre K. Mulligan. 'Bringing Design to the Privacy Table: Broadening "Design" in "Privacy by Design" Through the Lens of HCI'. In: *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*. CHI '19. Glasgow, Scotland UK: ACM, 2019, 262:1–262:17. DOI: [10.1145/3290605.3300492](https://doi.org/10.1145/3290605.3300492) (cited on page 40).
- [375] Yuhao Wu, Shaowei Wang, Cor-Paul Bezemer, and Katsuro Inoue. 'How do developers utilize source code from stack overflow?' In: *Empirical Software Engineering* 24.2 (Apr. 2019), pp. 637–673. DOI: [10.1007/s10664-018-9634-5](https://doi.org/10.1007/s10664-018-9634-5) (cited on page 39).
- [376] Glenn Wurster and P. C. van Oorschot. 'The Developer is the Enemy'. In: *Proceedings of the 2008 New Security Paradigms Workshop*. NSPW '08. Lake Tahoe, California, USA: Association for Computing Machinery, 2008, pp. 89–97. DOI: [10.1145/1595676.1595691](https://doi.org/10.1145/1595676.1595691) (cited on pages 14, 100, 101).
- [377] Shundan Xiao, Jim Witschey, and Emerson Murphy-Hill. 'Social Influences on Secure Development Tool Adoption: Why Security Tools Spread'. In: *Proceedings of the 17th ACM Conference on Computer Supported Cooperative Work & Social Computing*. CSCW '14. Baltimore, Maryland, USA: Association for Computing Machinery, 2014, pp. 1095–1106. DOI: [10.1145/2531602.2531722](https://doi.org/10.1145/2531602.2531722) (cited on pages 20, 23, 24, 27).
- [378] Jing Xie, Bill Chu, Heather Richter Lipford, and John T. Melton. 'ASIDE: IDE Support for Web Application Security'. In: *Proceedings of the 27th Annual Computer Security Applications Conference*. ACSAC '11. Orlando, Florida, USA: Association for Computing Machinery, 2011, pp. 267–276. DOI: [10.1145/2076732.2076770](https://doi.org/10.1145/2076732.2076770) (cited on pages 20, 28).
- [379] Jing Xie, Heather Lipford, and Bei-Tseng Chu. 'Evaluating Interactive Support for Secure Programming'. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '12. Austin, Texas, USA: Association for Computing Machinery, 2012, pp. 2707–2716. DOI: [10.1145/2207676.2208665](https://doi.org/10.1145/2207676.2208665) (cited on pages 19, 20, 28, 31).

- [380] Jing Xie, Heather Richter Lipford, and Bill Chu. ‘Why do programmers make security errors?’ In: *2011 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. Sept. 2011, pp. 161–164. DOI: [10.1109/VLHCC.2011.6070393](https://doi.org/10.1109/VLHCC.2011.6070393) (cited on pages 20, 23, 29, 41, 115).
- [381] Xin-Li Yang, David Lo, Xin Xia, Zhi-Yuan Wan, and Jian-Ling Sun. ‘What Security Questions Do Developers Ask? A Large-Scale Study of Stack Overflow Posts’. In: *Journal of Computer Science and Technology* 31.5 (Sept. 2016), pp. 910–924. DOI: [10.1007/s11390-016-1672-0](https://doi.org/10.1007/s11390-016-1672-0) (cited on pages 38–40, 42).
- [382] Robert K Yin. *Case study research and applications: Design and methods*. Sage Publications, 2018 (cited on page 19).
- [383] Koen Yskout, Riccardo Scandariato, and Wouter Joosen. ‘Does Organizing Security Patterns Focus Architectural Choices?’ In: *Proceedings of the 34th International Conference on Software Engineering*. ICSE ’12. Zurich, Switzerland: IEEE Press, 2012, pp. 617–627. DOI: [10.1109/ICSE.2012.6227155](https://doi.org/10.1109/ICSE.2012.6227155) (cited on pages 20, 25).
- [384] Koen Yskout, Riccardo Scandariato, and Wouter Joosen. ‘Do Security Patterns Really Help Designers?’ In: *Proceedings of the 37th International Conference on Software Engineering - Volume 1*. ICSE ’15. Florence, Italy: IEEE Press, 2015, pp. 292–302. DOI: [10.1109/ICSE.2015.49](https://doi.org/10.1109/ICSE.2015.49) (cited on pages 20, 25).
- [385] Jay (Hyunjae) Yu and Brenda Cude. ‘Hello, Mrs. Sarah Jones! We recommend this product!’ Consumers’ perceptions about personalized advertising: comparisons across advertisements delivered via three different types of media’. In: *International Journal of Consumer Studies* 33.4 (2009), pp. 503–514. DOI: [10.1111/j.1470-6431.2009.00784.x](https://doi.org/10.1111/j.1470-6431.2009.00784.x) (cited on pages 138, 140, 158).
- [386] Jun Zhu, Heather Richter Lipford, and Bill Chu. ‘Interactive Support for Secure Programming Education’. In: *Proceeding of the 44th ACM Technical Symposium on Computer Science Education*. SIGCSE ’13. Denver, Colorado, USA: Association for Computing Machinery, 2013, pp. 687–692. DOI: [10.1145/2445196.2445396](https://doi.org/10.1145/2445196.2445396) (cited on pages 20, 28, 29, 31).
- [387] Jun Zhu, Jing Xie, Heather Richter Lipford, and Bill Chu. ‘Supporting secure programming in web applications through interactive static analysis’. In: *Journal of Advanced Research* 5.4 (2014). Cyber Security, pp. 449–462. DOI: [10.1016/j.jare.2013.11.006](https://doi.org/10.1016/j.jare.2013.11.006) (cited on pages 20, 28).
- [388] Sebastian Zimmeck, Peter Story, Daniel Smullen, Abhilasha Ravichander, Ziqi Wang, Joel Reidenberg, N. Cameron Russell, and Norman Sadeh. ‘MAPS: Scaling Privacy Compliance Analysis to a Million Apps’. In: *Proceedings on Privacy Enhancing Technologies* 2019.3 (2019), pp. 66–86. DOI: [10.2478/popets-2019-0037](https://doi.org/10.2478/popets-2019-0037) (cited on pages 59, 133, 157).

- [389] Yixin Zou, Abraham H. Mhaidli, Austin McCall, and Florian Schaub. “‘I’ve Got Nothing to Lose’’: Consumers’ Risk Perceptions and Protective Actions after the Equifax Data Breach’. In: *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*. Baltimore, MD: USENIX Association, Aug. 2018, pp. 197–216. URL: <https://www.usenix.org/conference/soups2018/presentation/zou> (cited on page 109).

APPENDIX

A. Appendices for Privacy Champions Study

A.1. Screening Survey

[After the participant read the participant information sheet and consent form, and agreed to participate in the study.]

1. What is your current employment status? (Check all that apply).
 - Full time employee (or contractor equivalent) • Part-time employee (or contractor equivalent) • Freelance/consultant • Furloughed (temporarily laid off) or on leave • Unemployed • Student • Retired
2. Please select the statement that best describes your primary role at your current or most recent job.
 - Jobs NOT related to computer science, informatics, computer engineering, or related fields • Designing products (e.g. UI designer, interaction designer) • Developing software (e.g. programmer, developer, web developer, software engineer) • Testing software (e.g. tester, quality analyst, automation engineer) • Managing software development (e.g. project manager, IT manager, scrum master) • Privacy and/or security engineering (e.g. security engineer, privacy engineer, penetration tester, ethical hacker, cryptographer) • Other
3. What is your job title? (Free text)
4. How many members are there in your team that you work with directly? (Free text)
5. How many employees work in your organisation?
 - 1-9 employees • 10-99 employees • 100-999 employees • 1,000-9,999 employees • 10,000 or more employees
6. Overall, how many years have you worked in roles related to software development or IT? (Free text)
7. Where did you mainly learn to program and develop software? (Choose all that apply.)
 - Self-taught • High school courses • College or university courses • Online courses • Industry or on-the-job training • Other
8. Which of the following sectors most closely matches the one in which you are employed?
 - Business • Academia/education • Government • Non-profit • Other
9. Which one best describes your English proficiency level?
 - Basic Knowledge • Conversational/Functional • Proficient • Fluent/Native speaker

10. In which country do you currently reside? (List of countries)
11. What is your gender?
 - Male • Female • Non-binary • Prefer not to say • Prefer to self describe
12. How old are you? (Free text)
13. If you'd like to participate in the study, what email address should we use to contact you? (Free text)
14. What software would you prefer to use for the interview? (You can keep the video camera turned off).
 - Zoom • Google Hangouts Meet • Teams • Skype • Other
15. Do you have any comments or questions about the study? (Optional)

If you are selected for the interview, you will be notified over email within 2 weeks from today. Please keep an eye on the email inbox for the address that you provided in this survey.

A.2. Interview Script

[After the interviewer has introduced themselves, and obtained verbal consent.]

1. Can you tell me just briefly about what you do in your job?
2. Before the interview, we asked other people in your organisation to tell us who they think promotes user privacy, and among other people, they nominated you. Why do you think they consider you to be playing this role?
3. Could you define the term 'privacy' as you normally use it in your work context?
 - ▶ In your opinion, what is the difference between privacy and security?
4. What motivates you to promote user privacy in your work, formally or informally?
5. What do you find most rewarding about promoting user privacy?
6. What do you find most challenging or frustrating about promoting user privacy?
7. Think about formal or informal strategies that you use to promote or support users' privacy in product design and development:
 - ▶ Which ones do you find most effective? Why? How do you know it's effective?
 - ▶ Which ones do you find least effective? Why? How do you know it's ineffective?
8. In addition to your role, what other strategies in your organisation have you found most effective in promoting users' privacy?
 - ▶ Which strategies have you found to be least effective?

9. What communication channels for promoting privacy specifically do you think are the most effective and least effective? Why?
10. How are your efforts for promoting user privacy valued by other people within your team? Within the organisation?
 - ▶ What kind of feedback do you get?
 - ▶ Can you talk about any times when you felt that what you said or did wasn't appreciated?
11. How do you keep up with the latest in privacy?
12. Is there anything else you'd like to add with respect to what we've talked about today?

A.3. Codebook

1. Conceptualisations of privacy
 - Data management / control • Transparency / trust • Human right / ethical value as definition • Protect access to personal information • Legal compliance • Relationship between privacy and security • Complex / contextual term • Approaches to privacy (e.g. PbD and differential privacy)
2. Motivations
 - Organisational • Personal • Sense of responsibility
3. Rewards and positive feedback
 - Challenging task • Seeing shift / change in the company culture • Official promotion / incentives • Impact on end-user / society
4. Challenges and negative feedback
 - Attitudes • Communications issues • Dominant conceptualisation • Tension between priorities • Technical complexity
5. Strategies
 - External influence • Improving company culture values • Relying on instinct / being careful • Tools, APIs, and libraries • Training • Punishment • Documentation • Reviews / review meetings
6. Communication channels
 - Special events • Communication / productivity platforms • Verbal / messaging channels • Written communications
7. Criteria for (in)effectiveness of a strategy or a communication channel
 - Experience / intuition • Practical usefulness of processes / procedures • Impact on end products and decisions • Auditability / transparency / accountability • Fewer arguments / disagreements • Measurability • Relevance / targetedness • Difficulty to find / browse
8. Information resources
 - In-person networking • Online resources • Experiences of other companies • Academic research • Internal organisational channels

B. Appendices for Computer Science Students Study

B.1. Interview Script

1. Background
 - Can you tell me about yourself? Your academic and professional background? • Can you tell me about your dream job?
2. App scenario

Let's say you were asked to create a new group discussion app for in-class discussions. • Free list: what features would you consider in this app? • Here is a red pen. Can you circle the features that are privacy and security related? Or where you might have to consider privacy and security when building them? • Why these ones? • Who is most likely to try and attack this system? What are they likely going to try and do?
3. Threats and attacks
 - Can you tell me who hackers are, in your opinion? • Their intentions? • What are hackers trying to get? • Their background?
4. Responsibility attribution
 - Who is responsible for providing privacy and security to end-users?
5. Prior coding experiences
 - Tell me about the last piece of software you wrote. • Did you consider security while building your project? If not this one, any other projects? • Can you tell me an example of an API/library? Can you give me some experiences you have had with them? Any experience with security APIs in particular? • What was good about it? Why did you like it? • What was confusing about it?
6. Personal security/privacy practices

Now we are going to switch to talking about how you handle privacy and security personally as an end-user. • Free list: What words and concepts do you associate with computer security? • Can you give me an example of a good computer security practice? What about something you have done yourself? • Have you ever experienced a security or privacy compromise such as getting a virus on your computer, losing your password, having an email sent from your account, or loss of data about you? • How did you find out about the issue? • How did you correct it? • What did you learn from the experience? • Can you tell me some about the experiences you have had with passwords?
7. Background and demographics

B. Appendices for Computer Science Students Study

- How old are you?
- What is your degree title?
- Which year of the program are you in?
- What programming languages do you know?
- What programming courses have you taken?
- What security courses have you taken?
- What is your nationality?
- Where did you study your undergraduate, Masters, or other degrees?

C. Appendices for What Ad Networks Tell Developers About Privacy Study

C.1. Screenshots & Summary of Presented Privacy-Related Information

Here, we provide a list of screenshots (as of Jan 2021), that have a privacy element or a dark pattern. Table C.1 provides an overview of the available privacy information and where they are located.

Table C.1.: Presented privacy-related information on the ad networks' pages.

Ad Network	GDPR	CCPA	COPPA	Block Categories	Block Certain Domains	List of Vendors	Consent Popup	Location Permission	Other
GAM	Sidebar & warning in guide	Sidebar	Warning in guide	Dashboard	Dashboard	Dashboard	Customisable	Sidebar	Content rating (dashboard)
AMN	FAQ	FAQ	FAQ	Dashboard	Dashboard	-	-	Within guide	-
FAN	-	Sidebar	Sidebar	Dashboard	Dashboard	-	-	-	-
TMP	Sidebar & within guide	-	While creating an app	Dashboard	Dashboard	-	Provided	Within guide	Personal data passing (sidebar)

Warning: Ads may be preloaded by the Mobile Ads SDK or mediation partner SDKs upon calling `MobileAds.initialize()`. If you need to obtain consent from users in the European Economic Area (EEA), set any request-specific flags (such as `tagForChildDirectedTreatment` or `tag_for_under_age_of_consent`), or otherwise take action before loading ads, ensure you do so before initializing the Mobile Ads SDK.

Figure C.1.: GAM's warning about obtaining consent from users in the European Economic Area in the *Get Started* page.

```
public void loadForm(){
    UserMessagingPlatform.loadConsentForm(
        this,
        new UserMessagingPlatform.OnConsentFormLoadSuccessListener() {
            @Override
            public void onConsentFormLoadSuccess(ConsentForm consentForm) {
                MainActivity.this.consentedForm = consentForm;
                if(consentInformation.getConsentStatus() == ConsentInformation.ConsentStatus.REQUIRED) {
                    consentForm.show(
                        MainActivity.this,
                        new ConsentForm.OnConsentFormDismissedListener() {
                            @Override
                            public void onConsentFormDismissed(@Nullable FormError formError) {
                                // Handle dismissal by reloading form.
                                loadForm();
                            }
                        });
                }
            }
        },
        new UserMessagingPlatform.OnConsentFormLoadFailureListener() {
            @Override
            public void onConsentFormLoadFailure(FormError formError) {
                /// Handle Error.
            }
        }
    );
}
```

Figure C.2.: *Obtaining Consent with the User Messaging Platform* page in GAM provided a sample code for obtaining consent from users that constantly shows the popup to the user until they consent. Developers who use this sample code spread a 'nagging' dark pattern in their apps.

```
protected void presentConsentOverlay(Context context) {
    new AlertDialog.Builder(context)
        .setTitle("Location data")
        .setMessage("We may use your location, " +
            "and share it with third parties, " +
            "for the purposes of personalized advertising, " +
            "analytics, and attribution. " +
            "To learn more, visit our privacy policy " +
            "at https://myapp.com/privacy.")
        .setNeutralButton("OK", new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
                // TODO: replace the below log statement with code that specifies how
                // you want to handle the user's acknowledgement.
                Log.d("MyApp", "Got consent.");
            }
        })
        .show();
}

// To use the above method:
presentConsentOverlay(this);
```

Figure C.3.: *Precise Location Data Policy* page in GAM provided a sample code for obtaining location consent from users without providing a 'I do not consent' or 'No' button. Developers who use this sample code spread a 'forced action' dark pattern in their applications.

Configure Google Analytics

Analytics location ⓘ
United States

Data sharing settings and Google Analytics terms

Use the default settings for sharing Google Analytics data. [Learn more](#)

- Share your Analytics data with Google to improve Google Products and Services
- Share your Analytics data with Google to enable Benchmarking
- Share your Analytics data with Google to enable Technical Support
- Share your Analytics data with Google Account Specialists

I accept the [Measurement Controller-Controller Data Protection terms](#) and acknowledge I am subject to the [EU End User Consent Policy](#). This is required when sharing Google Analytics data to improve Google Products and Services. [Learn more](#)

I accept the [Google Analytics terms](#)

Upon project creation, a new Google Analytics property will be created and linked to your Firebase project. This link will enable data flow between the products. Data exported from your Google Analytics property into Firebase is subject to the Firebase terms of service, while Firebase data imported into Google Analytics is subject to the Google Analytics terms of service. [Learn more](#)

[Previous](#) [Create project](#)

Figure C.4.: Google Analytics had the default option on to share our data with Google (GAM). Turning off the default option would result in seeing a list of sub by default on permissions for all the grey items (e.g. Google products, Benchmarking). 'Preselection' dark patterns happens here as the default setting to share information between multiple services is not in the best interest of user privacy.

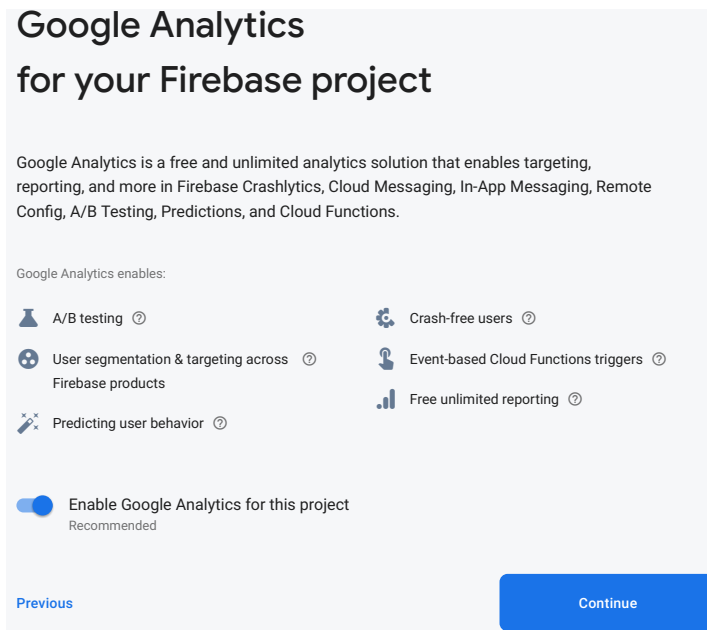


Figure C.5.: Google Analytics is turned on by default when creating an account on Firebase (GAM). ‘Preselection’ dark patterns happens here as the default setting to share information between multiple services is not in the best interest of user privacy.

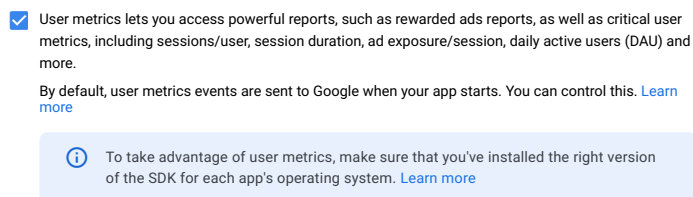


Figure C.6.: When creating an app on GAM, we were asked to enable users metrics for powerful reports. The box was pre-ticked. ‘Preselection’ happens here as sharing user data with multiple services is not in favour of user privacy.

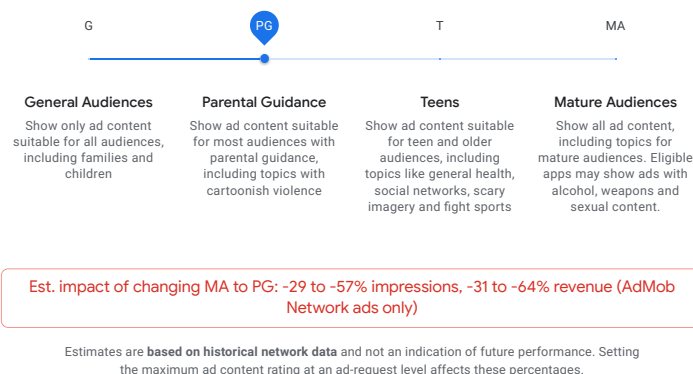


Figure C.7.: In the GAM account page under *Blocking controls* we could change the ad content rating. The default value was on the MA. ‘Toying with emotion’ has been applied to encourage developers stay with the MA ratings.





















Standard categories  0 blocked categories (15 remaining)	
Category name 	Status
Astrology and Esoteric 	Allowed <input type="checkbox"/>
Cosmetic Procedures & Body Modification 	Allowed <input type="checkbox"/>
Dating 	Allowed <input type="checkbox"/>
Downloadable Utilities 	Allowed <input type="checkbox"/>
Drugs & Supplements 	Allowed <input type="checkbox"/>
Get Rich Quick 	Allowed <input type="checkbox"/>
Politics 	Allowed <input type="checkbox"/>
References to Sex 	Allowed <input type="checkbox"/>
Religion 	Allowed <input type="checkbox"/>
Sensationalism 	Allowed <input type="checkbox"/>
Sexual & Reproductive Health 	Allowed <input type="checkbox"/>
Significant Skin Exposure 	Allowed <input type="checkbox"/>
Social Casino Games 	Allowed <input type="checkbox"/>
Video Games (Casual & Online) 	Allowed <input type="checkbox"/>
Weight Loss 	Allowed <input type="checkbox"/>
Restricted categories  1 blocked category (0 remaining)	
Category name 	Status
Gambling & Betting (18+) 	Blocked <input checked="" type="checkbox"/>

Figure C.8.: In the GAM account page under *Blocking controls* we could ‘allow’ or ‘block’ certain categories. The use of grey and blue colour to use ‘aesthetic manipulation’ dark pattern is easily visible. Grey commonly has a passive and negative tone whereas blue is known to have a positive tone [371].

Restricted data processing

You can choose from two options for users that Google determines are in California. If you want to continue to show personalised ads, tell us the partners that you want to monetise your ads with below. By default, data processing isn't restricted and personalised ads will continue to show.

Don't restrict data processing
 Google continues to show personalised ads to eligible users in California. Personalised ads are based on a user's past behaviour, such as previous visits to sites or apps or where the user has been.

Restrict data processing
 Google restricts how it uses certain unique identifiers and other data. Google only shows non-personalised ads from Google demand to eligible users in California. Non-personalised ads are based on contextual information, such as the content of your site or app.

Select the type of ads that you want to show

You can choose from two ad serving options. If you don't make any changes, personalised ads will continue to show for EEA and UK users. Your selection will not affect mediation.

Personalised ads
 Google can show personalised ads to your users in the EEA and the UK. ⓘ

Non-personalised ads
 Google will show only non-personalised ads to your users in the EEA and the UK. ⓘ

Figure C.9.: CCPA and GDPR sections of GAM have pre-selected items for information processing and personalised ads. 'Preselection' dark pattern occurs here because GAM by default collect information and also shows personalised ads (hence collects more information as well).

Preferences

Google-certified ad networks may be notified of your allow & block settings (including advertiser URLs and categories), so that they may comply and show appropriate ads. These networks compete for your ad inventory, potentially increasing your total AdSense revenue. ⓘ

Automatically allow new Google-certified ad networks

All ad networks 5009 networks (0 blocked)

Ad network ↑	Status
(Do Not Use) Assembly	Allowed <input type="checkbox"/>
(DO NOT USE) Dentsu-One Global w/ DBM	Allowed <input type="checkbox"/>
(Dont use) Leadbolt	Allowed <input type="checkbox"/>
(SGD) Amnet Asia SG w/Adobe	Allowed <input type="checkbox"/>
(USD) Amnet Asia SG w/Adobe	Allowed <input type="checkbox"/>
(주) 디엠씨미디어	Allowed <input type="checkbox"/>
+Don't use	Allowed <input type="checkbox"/>
-	Allowed <input type="checkbox"/>

Figure C.10.: In the *funding choices* service we could 'allow' and 'block' certain ad vendors. All vendors were 'Allowed' by default. GAM pre-ticked the box for automatically adding new vendors to list. 'Preselection' dark pattern spreads via this interface to end-users if developers do not make an effort to change the defaults.

- Select the consent choices that your users see ⓘ
- Consent/Manage options**
Users can consent to your vendors and their purposes, or manage options to customise their choices.
 - Consent/Do not consent/Manage options**
Users can consent or not consent to your vendors and their purposes, or manage options to customise their choices.

Figure C.11.: *Funding choices* is a service from GAM to create consent popups. It provides two ready-to-use consent popups to developers, the first option does not have a 'Do not consent' button.

The screenshot displays a vertical list of seven data processing categories, each with a title, a brief description, and two toggle options: 'Consent' and 'Legitimate interest'. The 'Consent' toggle is currently off (grey), and the 'Legitimate interest' toggle is currently on (blue). Each category also includes a 'View details' link.

- Measure content performance**
The performance and effectiveness of content that you see or interact with can be measured. [View details](#)
Consent: Off
Legitimate interest ⓘ: On
- Apply market research to generate audience insights**
Market research can be used to learn more about the audiences who visit sites/apps and view ads. [View details](#)
Consent: Off
Legitimate interest ⓘ: On
- Develop and improve products**
Your data can be used to improve existing systems and software, and to develop new products. [View details](#)
Consent: Off
Legitimate interest ⓘ: On
- Ensure security, prevent fraud, and debug** ⓘ
Your data can be used to monitor for and prevent fraudulent activity, and ensure systems and processes work properly and securely. [View details](#)
- Technically deliver ads or content** ⓘ
Your device can receive and send information that allows you to see and interact with ads and content. [View details](#)
- Match and combine offline data sources** ⓘ
Data from offline data sources can be combined with your online activity in support of one or more purposes. [View details](#)
- Link different devices** ⓘ
Different devices can be determined as belonging to you or your household in support of one or more of purposes. [View details](#)

Figure C.12.: *Funding choices* is a service from GAM to create consent popups. Several items could be customised for users. These are some of the default values. 'Preselection' dark pattern spreads via this interface to end-users if developers do not make an effort to change the defaults.

```

<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />

```

Figure C.13.: AMN in the *Quick Start Guide* page asks developers to add internet, network, wifi access, coarse and fine location for higher revenues for developers. While location permissions are called as ‘optional’ the sample code includes both fine and coarse location permissions. ‘Sneak into basket’ dark pattern is present here because developers may copy paste this code without fully being informed about what the sample code does.

Examples

The following examples provide a sample US Privacy String that represents the stated conditions. In all but the last example, a digital property has determined to use a US Privacy String and that CCPA applies to the transaction.

Example 1 meets the following conditions:

- Version 1 of the US Privacy string is being used. (1)
- The digital property has provided explicit user notice. (Y)
- The user has NOT made a choice to opt out of sale. (N)
- The digital property is not operating under the Limited Service Provider Agreement. (N)

1YNN

Example 2 meets the following conditions:

- Version 1 of the US Privacy string is being used. (1)
- The digital property has NOT provided explicit user notice. (N)
- The user has made a choice to opt out of sale. (Y)
- The digital property is not operating under the Limited Service Provider Agreement. (N)

1NYN

Example 3: Digital property outsources string creation

In this example the digital property has asked a vendor to create a US Privacy String on their behalf, knowing only whether the user has opted of sale of personal data.

- Version 1 of the US Privacy string is being used. (1)
- The status of provided explicit user notice is unknown. (-)
- The user has made a choice to opt out of sale. (Y)
- The status of operating under the Limited Service Provider Agreement is unknown. (-)

1-Y-

Example 4: CCPA does not apply

In this example, a digital property has determined to use a US Privacy String and that CCPA does not apply to the transaction.

1---

Figure C.14.: IAB’s sample values for CCPA’s *US Privacy String*.

```
final JSONObject pjObject = new JSONObject();
try {
    pjObject.put("us_privacy", "1---"); // example privacy string value
}
catch (JSONException ex)
{
    Log.e(LOGTAG, ex.getMessage());
}
final AdTargetingOptions adOptions = new AdTargetingOptions();
adOptions.enableGeoLocation(true);
adOptions.setAdvancedOption("pj", pjObject.toString());
this.adView.loadAd(adOptions);
```

Figure C.15.: AMN’s sample code in their [FAQ](#) page. `enableGeoLocation` is switched on in the sample code (‘sneaking’ [134]). ‘1---’ is provided as an example privacy string value. ‘1’ means version 1 and ‘-’ means ‘Not Applicable’. Two dark patterns are visible here, if developers copy paste this code ‘sneak into basket’ occurs and ‘preselection’ also happens because the defaults are not in favour of user privacy (see Figure C.14 for IAB code samples.)

Sensitive (0)

<input type="checkbox"/> Select All	<input type="checkbox"/> Adult	<input type="checkbox"/> Alcohol
<input type="checkbox"/> Firearms	<input type="checkbox"/> Foreign Language Ads	<input type="checkbox"/> Free Offers
<input type="checkbox"/> Gambling & Sports Betting	<input type="checkbox"/> General	<input type="checkbox"/> Get Rich Quick
<input type="checkbox"/> Sexually Suggestive	<input type="checkbox"/> Surveys & Questionnaires	<input type="checkbox"/> Sweepstakes
<input type="checkbox"/> System Dialog / Windows Error Ads	<input type="checkbox"/> Tobacco	<input type="checkbox"/> UGC
<input type="checkbox"/> Violence (War/Terrorism/Murder)		

Figure C.16.: AMN lets developers to block ad categories. By default no categories are blocked. ‘Preselection’ dark pattern happens here as developers may never visit this page and the defaults are not in the best interest of end-users.

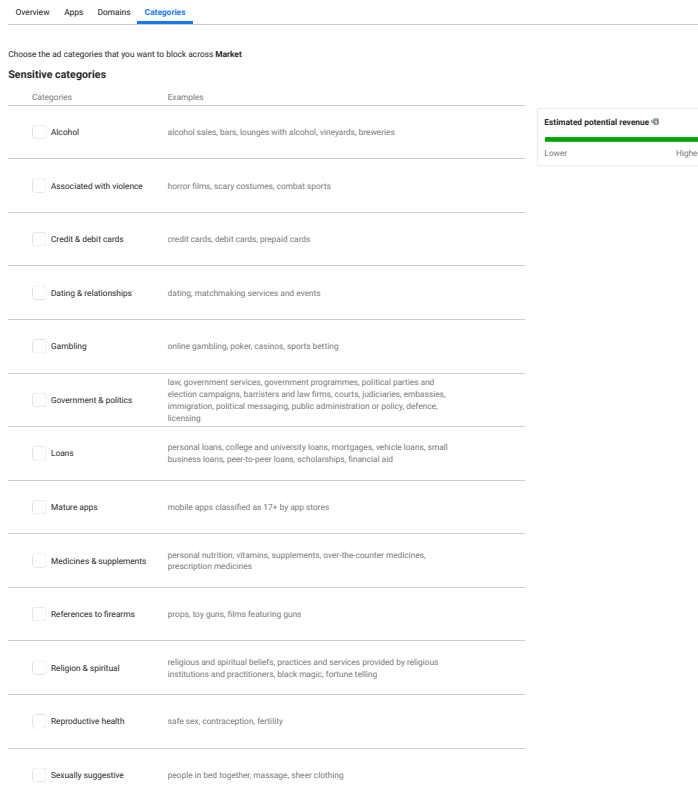


Figure C.17.: FAN’s sensitive categories are all active by default. The two blocked options are blocked by us. ‘Preselection’ dark pattern happens here as developers may never visit this page and the defaults are not in the best interest of end-users. The green bar on the top right corner will change as developers pick several items, hinting a loss of revenue as they block more categories.

Limited Data Use - Default Behavior

Facebook is offering a "Limited data use" flag in Audience Network SDK version 5.10 and above to control how California personal information is used in our systems. Publishers should implement the "Limited data use" flag as instructed in our [developer docs](#). Publishers using mediation partners should note that they must set the "Limited data use" flag before initialising the Mediation SDK for us to receive it. There will be a transition period to allow you to [implement the flag](#). During this time, we will limit data use on all unflagged events in California by default, meaning that **Facebook won't be able to serve ads or otherwise use specified data to the full extent described in our Audience Network Terms**. If you don't require this transition period, you can immediately enable full data use from this business ID whenever a request doesn't have a "Limited data use" flag.

- Enable full use of Specified Data under our Audience Network Terms from this Business ID whenever a request does not have a Limited Data Use flag

By enabling this option, you're permitting Facebook to make full use of specified data from this business ID whenever a request doesn't have a "Limited data use" flag.

You may want to enable this option if:

 - (a) Your business is not subject to the applicable law
 - (b) You are complying with applicable law in another way (e.g. filtering events before sending them to Facebook)
 - (c) You've completed implementation of "Limited data use" for this business ID

Figure C.18.: FAN users’ data policy is under developer’s setting page, next to roles and permissions and notification. It is disabled by default to no limit for data use. ‘Preselection’ dark pattern happens here as data collection which is not in favour of user privacy is turned on by default.

```

<!-- Required permissions -->
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />

<!-- Optional permissions. Will pass Lat/Lon values when available. Choose either Coarse or Fine -->
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    
```

Figure C.19.: TMP’s permissions for older versions of Android in the *Integrate the MoPub SDK for Android* page.

Note that blocking too broadly may negatively impact your revenue

[Expand all categories](#)

- > Arts & Entertainment (IAB1)
- > Automotive (IAB2)
- > Business (IAB3)
- > Careers (IAB4)
- > Education (IAB5)
- > Family & Parenting (IAB6)
- > Health & Fitness (IAB7)
- > Food & Drink (IAB8)
- > Hobbies & Interests (IAB9)
- > Home & Garden (IAB10)
- > Law, Gov't & Politics (IAB11)
- > News (IAB12)
- > Personal Finance (IAB13)
- > Society (IAB14)
- > Science (IAB15)
- > Pets (IAB16)
- > Sports (IAB17)
- > Style & Fashion (IAB18)
- > Technology & Computing (IAB19)
- > Travel (IAB20)
- > Real Estate (IAB21)
- > Shopping (IAB22)
- > Religion & Spirituality (IAB23)
- Uncategorized (IAB24)
 - > Non-Standard Content (IAB25)
 - Unmoderated UGC (IAB25-1)
 - Extreme Graphic/Explicit Violence (IAB25-2)
 - Pornography (IAB25-3)
 - Profane Content (IAB25-4)
 - Hate Content (IAB25-5)
 - Under Construction (IAB25-6)
 - Incentivized (IAB25-7)
 - > Illegal Content (IAB26)
 - Illegal Content (IAB26-1)
 - Warez (IAB26-2)
 - Spyware/Malware (IAB26-3)
 - Copyright Infringement (IAB26-4)

Figure C.20.: TMP's content categories. Default blocked categories cannot be changed and are greyed out. Blocked items by the developer are highlighted by blue.

D. Appendices for Nudging Developers About User Privacy Study

D.1. Survey Instruments

D.1.1. Screening Survey

1. Please select the statement that best describes your primary role at your current or most recent job.
 - I'm not employed
 - Jobs NOT related to computer science, informatics, computer engineering, or related fields
 - Designing products (e.g. UI designer, interaction designer)
 - Developing software (e.g. programmer, developer, web developer, software engineer)
 - Testing software (e.g. tester, quality analyst, automation engineer)
 - Managing software development (e.g. project manager, IT manager, scrum master)
 - Privacy and/or security engineering (e.g. security engineer, privacy engineer, penetration tester, ethical hacker, cryptographer)
 - Others (please specify)
2. How many years of experience do you have in software development? (numbers only)
3. How many years have you worked in mobile app development, specifically? (numbers only)
4. How many mobile apps have you worked on in the last 3 years? (numbers only)

D.1.2. Main Survey

[After the participant read the participant information sheet and consent form, and agreed to participate in the study.]

1. How many mobile apps have you worked on in the last 3 years? (numbers only)
2. [*Scenario description.*] Imagine that you are a shareholder in a software development company. Together with a small team, you created a [personal finance management/gaming] app. The app will be published in Europe and the United States and is mainly targeted towards adults (above age of

18). To monetise the app, you have decided to use the 'Acme' ad network to show ads to your users.

The Acme ad network offers a step-by-step Assistant – a graphical user interface that provides various configuration choices for integrating ads into your [personal finance management/gaming] app. The Assistant asks the developer several questions and then provides ad network configuration code based on the answers that can be imported directly into an app with minimal additional coding required.

The following are the 5 questions asked by Acme's Assistant, please answer them as if you were developing the [personal finance management/gaming] app.

I Which ad formats are you integrating?

- Banner: A basic ad format that appears at the top & bottom of the device screen.
- Interstitial: full-page ads appear at natural breaks & transitions, such as level completion. Supports video content.
- Rewarded Video: ads reward users for watching short videos and interacting with playable ads and surveys. Good for monetising free-to-play users. Supports video content.
- Native: customisable ad format that matches the look & feel of your app. Ads appear inline with app content. Supports video content.

II What level of graphics do you want for your ads?

- Ads with highest graphics quality. These ads will work best on newer phones with the latest operating systems.
- Ads with moderate to low graphics quality. These ads will work on most phones.

III Which platform are you integrating Acme ad network on?

- Android
- iOS
- Unity
- Windows Phone

IV Select the type of ads that you want to show. [Participants were asked to choose between the personalised and non-personalised ads described according to the condition, to which they were randomly assigned. See the text of the options in section 7.3.1.]

V Which of the following regulations apply to this app?

- GDPR (General Data Protection Regulation)
- CCPA (California Consumer Privacy Act)
- COPPA (Children's Online Privacy Protection Act)
- HIPAA (Health Insurance Portability and Accountability Act)
- None of the above
- I don't know

VI What was the biggest reason that made you pick the ad type: [chosen ads type]? (Please provide at much as details you can. Your response helps us better understand the reasons behind your choices.) [Open-ended question]

[Repeat the above questions for the second scenario.]

3. Assume that you decided to use personalised ads in both the gaming and financial management apps described earlier. How do you imagine you would go about asking for user consent for the personalised ads?

- I'd use my own consent form
 - I'd use the consent form provided by the Acme ad network
 - I'd use a third-party consent form provided by a leading tech company (e.g., Facebook, Google, Amazon, Twitter)
 - I'd use a third-party consent form provided by a not-for-profit organisation (e.g., Mozilla, Electronic Frontier Foundation)
 - I'd use a third-party consent form provided by other companies providing compliance services (e.g. OneTrust)
 - I won't ask for user consent because I don't think it's required
 - I won't ask for user consent because I don't think it's important
 - I won't ask for user consent because someone else in the team should take care of it
 - I won't ask for user consent because it's hard to do so
 - I won't ask for user consent because I'm not familiar with the consent process
 - I won't ask for user consent because the Acme ad network will take care of it
 - Other (please explain)
4. *[If 'I'd use my own consent form' chosen.]* What information sources, if any, would you use to build your own consent form? [Open-ended question]
 5. How, if at all, would your app's **revenue** change if you chose personalised ads over non-personalised ads in the [**personal financial management/gaming**] app described earlier? [Participants were asked about both app categories, in randomised order.]
 - Decrease by more than 81%
 - Decrease by 61%-80%
 - Decrease by 41%-60%
 - Decrease by 21%-40%
 - Decrease by 1%-20%
 - It won't change
 - Increase by 1%-20%
 - Increase by 21%-40%
 - Increase by 41%-60%
 - Increase by 61%-80%
 - Increase by more than 81%
 6. How, if at all, would the number of **users** of your app change if you chose personalised ads over non-personalised ads in the [**personal financial management/gaming**] app described earlier? [Participants were asked about both app categories, in randomised order.]
 - Decrease by more than 81%
 - Decrease by 61%-80%
 - Decrease by 41%-60%
 - Decrease by 21%-40%
 - Decrease by 1%-20%
 - It won't change
 - Increase by 1%-20%
 - Increase by 21%-40%
 - Increase by 41%-60%
 - Increase by 61%-80%
 - Increase by more than 81%
 7. How much priority do you give to privacy improvement and maintenance tasks in your daily development routines?
 - Not a priority
 - Low priority
 - Medium priority
 - High priority
 - Essential
 8. As a developer, how much control do **you** generally have over the amount of data collected by ad networks?
 - No control at all
 - Very little control
 - Moderate control
 - A lot of control
 - Full control
 9. How much control do **users** generally have over the amount of data collected by ad networks?
 - No control at all
 - Very little control
 - Moderate control
 - A lot of control
 - Full control
 10. What platforms have you previously developed apps for?

- Android • iOS • Blackberry • Windows Phone
11. How involved have you been in in-app advertising activities? [Options were: Not at all, A little, A moderate amount, A lot, A great deal]
 - Choosing an advertising partner or advertising network for an app. • Configuring the types of in-app ads shown in an app (e.g., where to place ads, what categories of ads to show, etc.) • Integrating the necessary code into an app to enable in-app advertising. • Other (please specify)
 12. Regarding mobile apps, have you used or worked with any advertising networks?
 - AdColony • Amazon Mobile Ad Network • Facebook Audience Network • Flurry • Google AdMob • InMobi • Millennial media • Twitter MoPub • Unity Ads • Vungle • Greyfriars Bobby • I have never included any ad networks in my mobile apps
 13. [*If 'I have never included any ad networks in my mobile apps' chosen.*] What are the primary reasons that you never included any ad networks in your apps? (Please provide at much as details you can. Your response helps us better understand your reasons behind your choices.)
 14. What is the revenue model of the apps that you typically develop?
 - Free with In-App Advertising, users cannot pay a fee to remove advertisements • Free with In-App Advertising, users can pay a fee to remove advertisements • Freemium model (app is free, certain features cost user's money) • Paid download • In-App purchases (selling physical or virtual goods through the app) • Subscription (similar to Freemium, except instead of paying for extra features, users must pay for extra content) • My apps are completely free • Cannot remember • Other (please specify)
 15. Who decides what revenue model to use in the apps that you develop?
 - Only me • Developer(s) / Programmer(s) • Project manager(s) • CEO and/or other upper-level management • Investor(s) • Other (please specify) • I do not know who was involved in the decision process
 16. Who decides what advertisement network to use in the apps that you develop?
 - Only me • Developer(s) / Programmer(s) • Project manager(s) • CEO and/or other upper-level management • Investor(s) • Other (please specify) • I do not know who was involved in the decision process
 17. What is your main source of income in software or mobile development?
 - I don't make money from software or mobile development • Salary, not dependent on software/app revenue • Primarily salary and bonuses, partially dependent on software/app revenue • Primarily direct software/app revenue • Other (please specify)
 18. What type of employment best describes your most recent app development experience?
 - Full time employee (or contractor equivalent) • Part-time employee (or contractor equivalent) • Freelance/consultant • Furloughed (temporarily

laid off) or on leave • Unemployed • Student • Retired • Other (please specify)

19. Please select the statement that best describes your primary roles at your most recent job.
 - I'm not employed
 - Jobs NOT related to computer science, informatics, computer engineering, or related fields
 - Designing products (e.g. UI designer, interaction designer)
 - Developing software (e.g. programmer, developer, web developer, software engineer)
 - Testing software (e.g. tester, quality analyst, automation engineer)
 - Managing software development (e.g. project manager, IT manager, scrum master)
 - Privacy and/or security engineering (e.g. security engineer, privacy engineer, penetration tester, ethical hacker, cryptographer)
 - Others
20. How many years of experience do you have in software development? (numbers only)
21. How many years have you worked in mobile app development specifically? (numbers only)
22. Where did you mainly learn to program and develop software?
 - Self-taught
 - High school courses
 - College or university courses
 - Online courses
 - Industry or on-the-job training
 - Others
23. How many people were employed in the organisation for which you worked on the app development most recently?
 - 1-9 employees
 - 10-99 employees
 - 100-999 employees
 - 1,000-9,999 employees
 - 10,000+ employees
24. How many members were in the team that you have directly worked with most recently? (numbers only)
25. How old are you? (numbers only)
26. In which country do you currently reside? [List of countries]
27. If you can't find your country in the above question options, please enter it here. [Open-ended question]
28. What is your gender?
 - Male
 - Female
 - Non-binary
 - Prefer not to say
 - Prefer to self describe
29. If you'd like to be included in the raffle, please provide your email address.
30. Do you have comments or anything to say about the survey or study in general? (optional)

D.2. Ads Options on Google AdMob Developer Dashboard

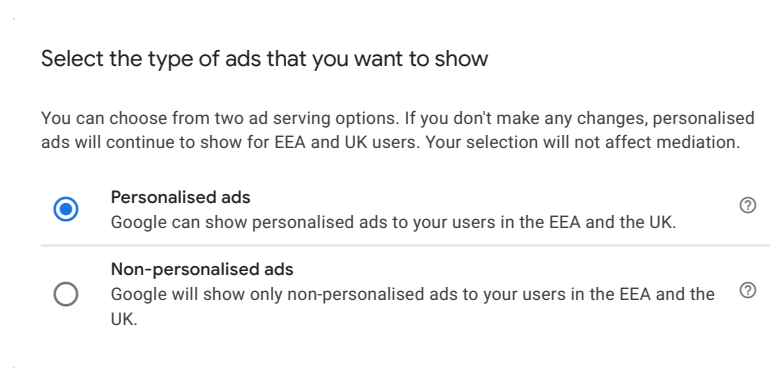


Figure D.1.: Screenshot from Google AdMob developer dashboard: Blocking controls -> Manage EU user consent (as of Jan'21).

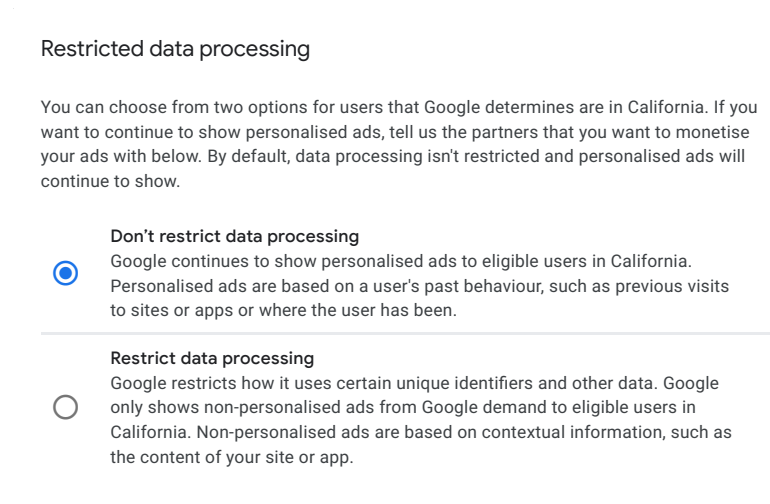


Figure D.2.: Screenshot from Google AdMob developer dashboard: Blocking controls -> Manage CCPA settings (as of Jan'21).

D.3. Participants' Demographics and Opinions About Ad Networks

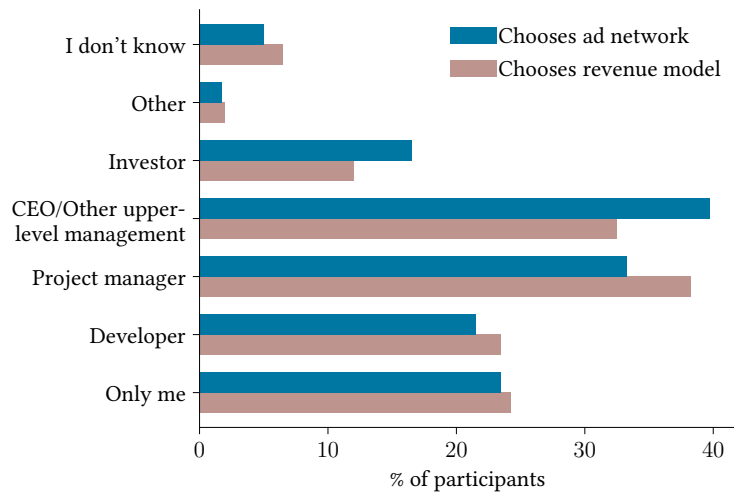


Figure D.3.: Responses about who decides what revenue model and ad network to use in the apps participants develop.

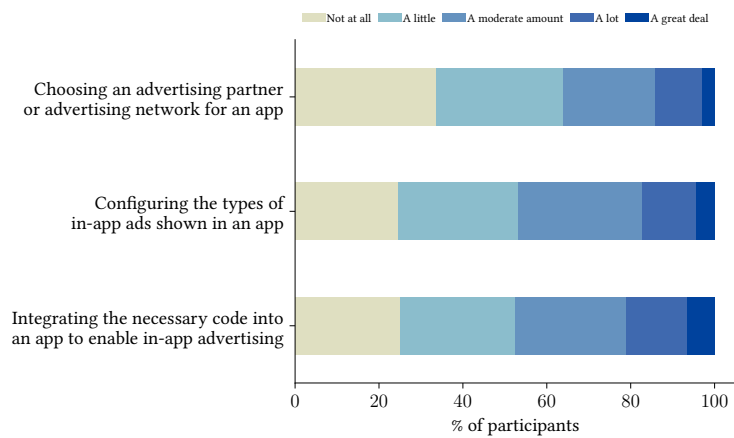


Figure D.4.: Involvement in in-app advertising activities.

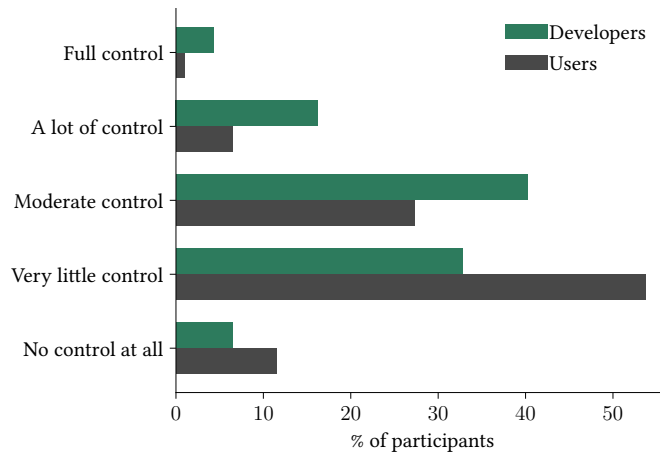


Figure D.5.: Perceived control over ad networks' data collection.

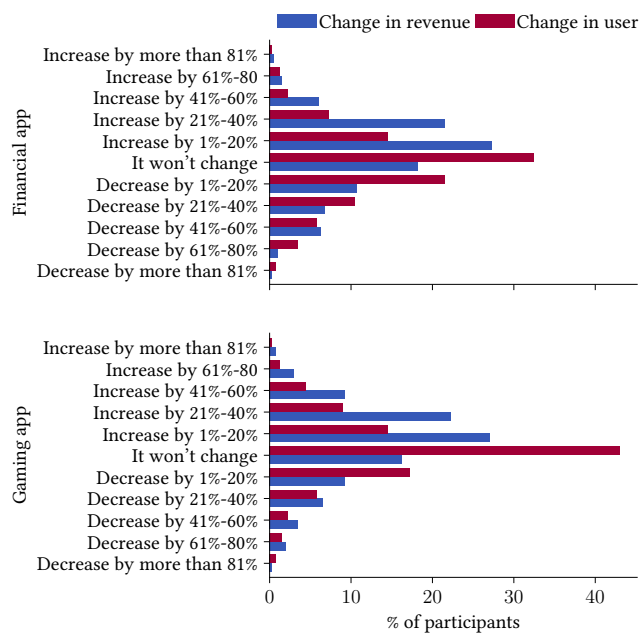


Figure D.6.: Expected change in app's revenue and number of users if personalised ads are chosen over non-personalised ads.

Table D.1.: Summary of participants’ demographics and prior experience with ads ($N = 400$, unless otherwise specified).

	#Participants		#Participants
Age	$\mu = 27.4, \sigma = 8$	Revenue Models	
Gender		Free with In-App Advertising	120 (30.0%)
Male	330 (82%)	Completely free	103 (25.8%)
Female	58 (14%)	Freemium model	103 (25.8%)
Prefer not to say	11 (3%)	In-App purchases	83 (20.8%)
Non-binary	1 (<1%)	Free with In-App Advertising	82 (20.5%)
Current Continent of Residence		Subscription	54 (13.5%)
Europe	265 (66%)	Paid download	43 (10.8%)
North America	75 (19%)	Other	11 (2.8%)
Asia	24 (6%)	Can't remember	8 (2.0%)
Oceania	15 (4%)	Which Ad Networks Used in the Past	
South America	11 (3%)	Google AdMob	191 (47.8%)
Africa	7 (2%)	Never included any ad networks in apps	123 (30.8%)
Prefer not to say	3 (1%)	Facebook Audience Network	117 (29.2%)
Employment Status		Unity Ads	81 (20.2%)
Full-time	147 (37%)	Amazon Mobile Ad Network	64 (16.0%)
Student	107 (27%)	AdColony	33 (8.2%)
Freelance/consultant	75 (19%)	Twitter MoPub	27 (6.8%)
Part-time	54 (14%)	Flurry	15 (3.8%)
Unemployed	10 (2%)	InMobi	12 (3.0%)
Temporarily laid off	3 (1%)	Other	11 (2.8%)
Other	2 (<1%)	Vungle	9 (2.2%)
Retired	2 (<1%)	Millennial media	7 (1.8%)
Number of Employees		Sources of User Consent Forms	
1–9 employees	170 (42%)	The Acme ad network's form	128 (32.0%)
10–99 employees	142 (36%)	Leading tech company's form	90 (22.5%)
100–999 employees	49 (12%)	My own consent form (see Table 7.4)	71 (17.8%)
1,000–9,999 employees	21 (5%)	Not-for-profit organization's form	43 (10.8%)
10,000 or more employees	18 (4%)	Won't ask for user consent because:	39 (9.75%)
Team Members	$\mu = 7.3, \sigma = 10.3$	Acme ad network will take care of it	14 (3.5%)
Years of Experience		Someone else in the team should do it	14 (3.5%)
In software development	$\mu = 5.1, \sigma = 5.3$	Not familiar with the consent process	6 (1.5%)
In mobile development	$\mu = 2.7, \sigma = 2.6$	It's not important	2 (0.5%)
Number of Developed Apps in the Past Three Years	$\mu = 3.5, \sigma = 4.2$	It's hard to do so	2 (0.5%)
Software-Related Roles ($N = 291$)		It's not required	1 (0.2%)
Developing software	186 (64%)	Companies providing compliance services	24 (6.0%)
Testing software	37 (13%)	Other	5 (1.2%)
Managing software development	32 (11%)	Given Priority to Privacy in Development Routines	
Designing products	30 (10%)	High priority	144 (36%)
Privacy & security engineering	5 (2%)	Medium priority	136 (34%)
Main Income Source		Essential	61 (15%)
Salary, not dependent on software/app revenue	172 (43%)	Low priority	54 (14%)
Salary, partially dependent on software/app revenue	85 (21%)	Not a priority	5 (1%)
I don't make money from software/app dev.	80 (20%)	Where Learned to Develop Software	
Direct software/app revenue	58 (14%)	Self-taught	248 (62.0%)
Other	5 (1%)	College or university courses	237 (59.2%)
		Online courses	170 (42.5%)
		Industry or on-the-job training	103 (25.8%)
		High school courses	70 (17.5%)
		Other	3 (0.8%)

