How Developers Talk About Personal Data and What It Means for User Privacy: A Case Study of a Developer Forum on Reddit

TIANSHI LI, Carnegie Mellon University, USA
ELIZABETH LOUIE, Carnegie Mellon University, USA
LAURA DABBISH, Carnegie Mellon University, USA
JASON I. HONG, Carnegie Mellon University, USA

While online developer forums are major resources of knowledge for application developers, their roles in promoting better privacy practices remain underexplored. In this paper, we conducted a qualitative analysis of a sample of 207 threads (4772 unique posts) mentioning different forms of personal data from the /r/androiddev forum on Reddit. We started with bottom-up open coding on the sampled posts to develop a typology of discussions about personal data use and conducted follow-up analyses to understand what types of posts elicited in-depth discussions on privacy issues or mentioned risky data practices. Our results show that Android developers rarely discussed privacy concerns when talking about a specific app design or implementation problem, but often had active discussions around privacy when stimulated by certain external events representing new privacy-enhancing restrictions from the Android operating system, app store policies, or privacy laws. Developers often felt these restrictions could cause considerable cost yet fail to generate any compelling benefit for themselves. Given these results, we present a set of suggestions for Android OS and the app store to design more effective methods to enhance privacy, and for developer forums (e.g., /r/androiddev) to encourage more in-depth privacy discussions and nudge developers to think more about privacy.

CCS Concepts: • **Human-centered computing → Human computer interaction (HCI)**; **Empirical studies in collaborative and social computing**; • **Security and privacy → Human and societal aspects of security and privacy**; **Software security engineering**.

Additional Key Words and Phrases: Privacy; Software Development; Community of Practice; Android; Qualitative Analysis; Reddit

ACM Reference Format:
Tianshi Li, Elizabeth Louie, Laura Dabbish, and Jason I. Hong. 2020. How Developers Talk About Personal Data and What It Means for User Privacy: A Case Study of a Developer Forum on Reddit. *Proc. ACM Hum.-Comput. Interact.* 4, CSCW3, Article 220 (December 2020), 28 pages. https://doi.org/10.1145/3432919

1 INTRODUCTION

In recent years, concerns about data privacy have grown. People struggle with excessive data collection, unexpected data sharing, and difficulty understanding and managing how their data is used by others. Despite the introduction of strict privacy laws such as GDPR and CCPA, many developers still fail to comply with privacy best practices [16, 32].

Authors' addresses: Tianshi Li, Carnegie Mellon University, Pittsburgh, USA, tianshil@cs.cmu.edu; Elizabeth Louie, Carnegie Mellon University, Pittsburgh, USA, elouie2@andrew.cmu.edu; Laura Dabbish, Carnegie Mellon University, Pittsburgh, USA, dabbish@andrew.cmu.edu; Jason I. Hong, Carnegie Mellon University, Pittsburgh, USA, jasonh@cs.cmu.edu.

2573-0142/2020/12-ART220 $15.00
https://doi.org/10.1145/3432919

We argue that a better understanding of developers' current data practices and the challenges they face can help inform effective solutions to privacy issues. Online developer forums are a special type of community of practice which offer a place for developers to informally help others solve development-related problems and share developer news. These websites are a major resource of knowledge for developers in general [6, 8], making them also a potential place to disseminate knowledge about data use and privacy. Furthermore, developers sometimes write posts that detail the data practices of their own applications to provide background for a question or solicit suggestions about app design, making these sites a window into how developers use personal data and handle privacy risks in the real world.

Previous work has examined how developers talk about privacy in the context of the popular developer Q&A site Stack Overflow [31]. Tahaei and colleagues examined posts that specifically mention the word "privacy" (for web, smartphone, and other technologies discussed on the site), and found that developers did turn to Stack Overflow for support about privacy issues, and the largest driver of privacy-related questions was "personal concerns, client, or company requirements".

In this paper, we offer a complementary analysis, focusing on `/r/androiddev`, a developer forum on Reddit focused on Android development. Android is the most popular smartphone platform today, taking roughly 87% of the global market share. `/r/androiddev` started in June 2009 and now has over 144k members, with roughly 12 new threads and 175 new posts per day. Unlike Stack Overflow, which is designed to answer technical questions, `/r/androiddev` allows and encourages in-depth discussion on a broad range of Android-development-related issues, such as giving feedback on high-level app designs, suggesting useful libraries, and discussing news for Android developers. This difference could potentially give rise to more interesting discussions with respect to privacy.

Rather than focusing specifically on the term "privacy" as in prior work [15, 31], we consider discussions of personal data use more broadly, looking at when and how privacy concerns arise in these discussions. The concept of "personal data", defined as any information that is related to an *identified* or *identifiable* person [2], is more concrete and related to all aspects of data privacy, such as data collection, data sharing, data storage, user-facing notices, and user control of their data. In addition, developers sometimes discuss privacy concerns without using the word privacy. For example, developers might discuss privacy aspects of a data use case (e.g., only collecting the minimum data needed) without mentioning the word "privacy".

With this new angle, we aim to investigate the following research questions:

RQ1 What types of discussions do developers have about personal data in an online community of practice?

RQ2 When developers talk about personal data, how do they discuss privacy-related issues?

RQ3 What risky data practices (e.g., sharing data with third parties) are discussed by developers?

We conducted a qualitative analysis of 207 threads with 4772 posts from the `/r/androiddev` sub-forum mentioning different forms of personal data. We performed bottom-up open coding on the sampled posts to build a typology of discussions on personal data use, and conducted follow-up analyses to understand what types of posts generated privacy concerns or mentioned risky data privacy practices.

Overall, this paper makes the following contributions:

- We found that privacy-related issues were occasionally discussed on this developer forum (about 20% threads that discussed personal data also discussed privacy). However, most discussions of privacy were triggered by external events (e.g., restrictions to enhance privacy from the operating system and app store policies), showing that developers passively react to privacy requirements most of the time. Furthermore, developers often expressed that complying with these requirements incurred a high cost with little benefit for themselves.

- We showed that developers frequently mentioned risky data practices such as sending data out of the device and sharing data with third parties when discussing the design and implementation of specific apps. However, they rarely discussed privacy issues that these risky data practices may involve.
- We offer a set of design suggestions for Android OS, app store, and developer forums to promote better privacy practices in Android app development. For example, Android OS should be more upfront about the design rationale when introducing new API designs for privacy; Android OS and app stores may want to complement the current data-restriction-based approaches with clear and publicly visible privacy metrics to encourage developers to adopt better privacy practices; and developer forums can help posters frame questions in a way that is more likely to prompt feedback on privacy.

2 RELATED WORK

In this section, we first discuss developers' important role in protecting user privacy. We then review studies aiming to understand how developers handle privacy. Finally, we discuss prior work that also studied online developer communities to investigate how knowledge was curated and transferred on these platforms, and how previous researchers leveraged them as a resource to study other developer practices.

2.1 Analyzing Developers' Personal Data Use and Privacy Practices

In this subsection, we first clarify the relationship between two prominent concepts in this paper, namely *privacy* and *personal data*, and offer an overview of past work on developers' practices with respect to personal data and privacy.

Privacy is a complex, abstract concept. Many privacy theories try to provide concrete definitions of privacy by prescribing how personal data should be used by other individuals and organizations. Classic definitions of privacy demands "limited access to" personal data by other individuals and organization [9], or "control over" one's personal data [28]. More recently, the seminal privacy theory Contextual Integrity [25] defines privacy as the appropriateness of data flow under certain contexts.

Accordingly, there is a body of work that applied some of these definitions to analyze how well different aspects of privacy are handled in the real world. For example, Habib et al. [16] studied the usefulness and usability of opt-out and data deletion choices provided in 150 English-language websites and found that the location and design of these privacy choices made it difficult for users to find and use these choices. Andow et al. [1] analyzed the consistency between the privacy policies and the detected data flows in 13,796 Android apps and found that 42.4% of the apps did not correctly disclose their data practices in the privacy policies. Chitkara et al. [11] discovered that a set of 30 third-party libraries could account for half of the accesses to sensitive personal data on an Android phone. These studies suggest that there are still many privacy issues that need to be addressed by developers.

However, there are a few fundamental challenges that limit the effectiveness of automated analysis approaches like these. For example, machine-based data flow analysis can not easily distinguish malicious disclosures from legitimate data disclosures and often has to rely on further manual examination [24]. The use of app obfuscation technology further increases the difficulty to infer what the data is used for [36]. Moreover, when data is uploaded to the backend server of the app, only developers can potentially know how the data will be used and who the data will be further shared with.

For these aspects that are too subtle or too inaccessible to be analyzed at scale, developers' own knowledge and willingness to comply with privacy requirements become especially important to understand. Towards this end, we investigated what caused developers to discuss privacy-related issues when talking about personal data, as well as their attitudes manifested in these discussions, in the context of /r/androiddev. We identify problems such as how developers often acted passively to handle privacy, and propose suggestions to address the problem for main stakeholders of the development ecosystem.

2.2 Research Aimed to Understand How Developers Handle Privacy

Given the important role that developers play in protecting user privacy, there has been a growing interest in using a human-centered approach to study how developers handle privacy in practice. One line of previous research studied developers' knowledge and attitudes of privacy using interviews [7, 20], surveys [30], or observing developers' personal data practices in a lab-based setting [20, 29]. Li et al. [20] directly interviewed developers about the concept of privacy, and found that although some developers cared about user privacy, they only had a partial understanding of this concept and do not always keep an accurate track of the data practices in their own apps. Sheth et al. [30] showed a list of factors such as "Data Sharing", "Data Breaches" "Privacy Policy", "Anonymizing all data" and asked survey respondents to rate their importance. They found that developers clearly prefer improving privacy using technical measures like data anonymization over privacy laws and policies. Senarath and Arachchilage [29] requested developers to accomplish a software design task for a hypothetical health app and to embed privacy into their design. They observed some common issues developers were facing during this process, including design requirements contradicting privacy requirements and lacking knowledge about privacy theories.

This type of research features a formal and direct query about the concept of privacy. While this body of research has led to important findings, these methods could not characterize developers' attitudes towards and practices regarding privacy in a natural condition when not explicitly prompted about the concept. In contrast, we studied a log of online discussions that happened naturally in a developer forum, which can offer us a better understanding of what developers actually think about privacy in practice.

Another line of work used a similar method of studying discussions on online developer forums [15, 31]. Greene and Shilton [15] studied posts that contained the keyword "privacy" in an iOS developer forum and an Android developer forum, and suggested that developers of different platforms had different interpretation of the meaning of "privacy". Tahaei et al. [31] analyzed questions and answers on the developer Q&A forum Stack Overflow that mentioned "privacy" in the title or tags, and then identified related topics such as privacy policies, access control, and version changes.

Although we also studied online developer forums, our methods differ in several ways from this prior work. We also examine privacy and use of personal data from a different perspective and contribute new findings. First, instead of using "privacy" as a keyword, we used a set of keywords derived from the concept of personal data, which may be more concrete and familiar to average developers. Second, prior work using "privacy" as the only keyword may result in sampling bias because it could not help us understand situations when developers discussed privacy-related issues without referring to the exact keyword or when developers discussed sensitive data practices but did not mention the corresponding privacy risks at all. By directly focusing on personal data practices, our method can address both of these issues. Third, unlike prior work that studied Q&A sites like StackOverflow [31], the platform studied in this work (/r/androiddev) has encouraged more open-ended discussions such as reactions to developer news and extensive discussions about the design philosophies of a new OS release. The breadth of the discussion topics allowed us to study privacy discussions under many contexts (e.g., during app development, triggered by external events) and to compare privacy discussions among these contexts.

2.3 Online Developer Forums as Communities of Practice (CoP) of Software Development

Communities of Practice (CoP) are groups of people informally bound together by shared expertise and passion of a topic [35]. They support knowledge sharing, learning, solving problems, and promoting the spread of best practices [35].

Online developer forums are a special type of CoP that serve as a major source of knowledge for software developers [6, 8]. There has been a lot of research on these forums, especially Stack Overflow. For example, Abdalkareem et al. [4] analyzed the topics of development questions on Stack Overflow, and found that knowledge about technical concepts (e.g., programming languages, API use, etc.) and documenting bugs are the most popular topics that developers sought help for on Stack Overflow. Prior work also showed Stack Overflow had a significant effect on developers' coding activities, such as reusing source code from Stack Overflow [37] and open-source development work on GitHub [33]. In particular, code reuse has a negative impact on code security [13], which is an important component of privacy. Developer forums have been studied for particular sub-areas in software development, such as deep learning [19] and mobile app development [22]. Specifically, prior work has also studied privacy in software development by analyzing iOS and Android developer forums [15] and Stack Overflow [31] (as discussed in the previous sub-section).

Privacy is an important yet rarely studied topic in this line of research, possibly due to its abstract nature and the fact that it has received less attention from developers than other functional requirements such as fixing bugs, improving usability, and optimizing performance. This makes it harder to capture posts of interest. Correspondingly, one contribution of our work is to present an alternative approach to study this problem: focusing on personal data rather than the "privacy" keyword. In this way, we were able to identify many low-level discussions revealing privacy-related challenges that developers encountered in practice, which could not be detected using the keyword "privacy". To our knowledge, our work is also the first to identify the low visibility issue of privacy on developer communities of practice and discuss potential solutions for these forums to address this issue.

3 METHODOLOGY

In this section, we provide a brief overview of `/r/androiddev`, and then present how we collected data from this subreddit, how we sampled threads that contained personal data use discussions, and how we conducted qualitative coding analysis and sentiment analysis to answer our three research questions.

3.1 Research Setting: A Developer Forum on Reddit About Android Development (/r/androiddev)

The `/r/androiddev` forum is a subreddit themed around Android app development. This is an active community that has more than 144,000 subscribed readers as of May 2020, with 12 new threads and 175 new posts created per day on average. Moreover, this forum has two special community rules which make it a good fit for the goals of our study. The first rule "Must be related to Android Development" helps us obtain a dataset mostly dedicated to Android-development-related discussion, and the second rule "No easily searched/specific dev questions" differentiates it from other programmer Q&A sites (e.g., Stack Overflow) by encouraging more high-level discussions about "architecture, performance optimizations, or design" of apps, which give us more opportunities to dig into developers' design decisions, design rationales, and the problems facing them when personal data is collected and used in their apps.

A non-scientific survey conducted in 2017 by the moderators of this forum[1] provides a glimpse into the demographics of this forum. There were 386 responses collected in total. In the responses to the question "Which of the following describes you?", 274 (71%) self-identified as "Android Developer - Professional", 178 (46%) as "Android Developer - Personal", and 67 (17%) as "Learning Android Development". The responses to the question "Where are you located" suggests that our dataset contains a global set of developers. The top 4 countries were United States (90, 23%), United Kingdom (33, 8%), Germany (32, 8%), and India (19, 5%). This diversity is important for a study on privacy, since some privacy laws target users in certain regions (e.g., GDPR for Europe, CCPA for California).

3.2 Data Collection

We pulled posts in `/r/androiddev` created between March, 2009 (the start of this online forum) and Feb, 2020 using the services provided by pushshift.io [2] We did not include posts that were empty, removed by the moderator (the text is "[removed]") or deleted by the poster (the text is "[deleted]"). The dataset contains 46,254 threads and 666,676 posts, with 9.3 posts per thread on average ($median = 5.0$, $std = 18.6$, including the original post). On average, there were 12.1 new threads ($median = 10.0$, $std = 9.4$) and 174.6 new posts ($median = 176.0$, $std = 127.7$) created per day.

3.3 Sampling a Set of Threads That Contain Discussions Related to Personal Data

In this sub-section, we present the process of generating a sample that contained personal data use discussions for the main analyses. The general method was we first filtered candidates of relevant threads using a list of keywords of personal data curated based on important privacy laws, and then manually removed false positives in a random sample of potentially relevant threads (e.g., removing hiring posts that used "location" in "job location").

3.3.1 Keywords of Personal Data. As personal data is a rather broad concept, we could not exhaustively enumerate all possible data types to examine in our study. Therefore, we drew on the definition of personal data from a set of crucial privacy laws announced in recent years, including CCPA, CalOPPA, COPPA, and GDPR. The first three laws protects the privacy rights of residents of California, and the last one protects the privacy rights of European users (not all European countries).

Each of these laws provided a general definition of personal data along with a list of personal data examples. For example, in GDPR, personal data is defined in the following way (with keywords we extracted from the text in bold):

> "personal data" means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a **name**, an **identification number**, **location** data, an **online identifier** or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person;

[1] http://web.archive.org/web/20200531071603/https://www.reddit.com/r/androiddev/comments/64a56m/randroiddev_survey_time/

[2] https://web.archive.org/web/20200530201525/https://pushshift.io/

[3] SSAID is an alternative name of Android ID, which is an Android-specific device ID.

[4] Line1 is a code name of phone number in Android

Table 1. The four types of personal data that we studied and the corresponding keywords used to locate posts that mentioned personal data. This keyword list is curated based on our analysis of the definitions of personal data in four privacy laws: CCPA, CalOPPA, COPPA, and GDPR.

Data type category	Related keywords used in privacy laws
Unique identifier	`first name, last name, real name, identification number, id number, social security number, ssn, license number, passport number, screen name, user name, account name, user id, username, userid, online identifier, imei, device serial number, advertising id, android id, ssaid`[3]`, mac address, imsi, instance id, guid, internet protocol address, ip address, email address, telephone number, phone number, line1`[4]
Photo and Video	`video recording, camera, gallery`
Audio	`audio recording, microphone, voice`
Location	`location, physical street address, home address, physical address, street name, city name, postal address`

We identified 57 unique keywords/key phrases from the these four laws. Then we removed keywords/key phrases that refer to data that is only sensitive when used or stored with other data (e.g., “birthday”, “height”, “weight”), and removed keywords/key phrases that are too specific to a particular use case and less likely to be discussed on a developer forum (e.g., “professional or employment-related information”). To help match more relevant posts on an Android development forum, we expanded the list by adding terms used in Android development that have a similar meaning to one of the keywords/key phrases, such as “line1” for telephone number. We also added abbreviations (e.g., “ip address” is an abbreviation of “internet protocol address”) and popular variations of keywords/key phrases (e.g., “username” and “userid”)

Finally, we generated a list of 44 keywords/key phrases summarized into four categories of personal data: *Unique identifier*, *Photo and Video*, *Audio*, and *Location*. The list is presented in Table 1.

3.3.2 Keyword-Based Filtering Process. In this process, we aimed to only keep threads that contained at least one of the 44 keywords/key phrases (both singular and plural forms) listed in Table 1. To improve matching efficiency, all text was converted to lower case; for hyperlinks, we only used the alternative text if available, and otherwise kept the hyperlink.

Then the matching process was conducted at word level rather than character level to avoid mismatches like matching “guid” to “guide”. We made sure that key phrases (containing more than one word) would not be matched across more than one sentence. Finally, We obtained 6827 unique threads that contained at least one keyword/key phrase, comprising 14.8% threads of the entire data set.

3.3.3 Manual Refinement Process. We drew a random sample of 600 threads from the 6827 threads that potentially contain discussion regarding personal data of users. As we went through this sample, we realized that sometimes the keyword was not used to represent the personal data type that we were looking for. For example, the word “location” was intended to match discussion related to geolocation, while it could also be used to refer to “file location” or “job location”; the word “voice” was intended to match discussion related to audio, while it could also be used as a verb (e.g., “voice my opinion”); “email address” can be used to refer to “developer email address” rather than user email address. Therefore, two researchers (the first and second author) collectively reviewed the occurrences of keywords and removed all these false positives.

We also noticed some special threads opened by moderators to gather discussion on similar topics in one place, such as "Weekly Questions Thread - November 18, 2019." Since the first-level comments were the actual discussion starters, we split up posts and considered them as separate threads if they had different first-level comments or were replying to different first-level comments.

3.3.4 Final Sample. The final sample contained 329 unique threads (207 used for developing the typology of RQ1 and conducting other qualitative and quantitative analysis; the other 122 used to validate the generated typology), with 23.1 posts ($median = 7.0$, $std = 72.0$) per thread on average. The 329 threads included first-level comments of weekly question threads that matched the keywords. This suggests that threads that elicited personal data use discussions received more replies than other threads in general.

3.4 Qualitative Analysis of Data Use Discussions (RQ1)

To study the first research question "What types of discussions do developers have about personal data in an online community of practice", we qualitatively analyzed the discussion starters of the 207 threads in our final sample, using a bottom-up process to identify the common topics in the discussion starters and develop a proposed typology, and then used the other 122 threads to validate the typology. We then conducted a sentiment analysis on the threads to compare sentiment of different data use discussions types.

3.4.1 Typology Development Process. Following previous guidelines [12, 27], a single researcher (the first author) conducted all of the coding using an inductive process where major themes emerge through interpretation of the data. During this process, the coder first reviewed the discussion starters of the 207 threads, and developed an initial set of codes that characterized the themes of these posts. Then the coder analyzed the frequencies of and the relevance among these codes, and revised and merged the codes accordingly. Finally, we renamed the final codes to create a typology for topics of discussion starters (also determining the topic of the thread) that elicited personal data use discussions, and then updated the labels of categories for each thread. The codebook of the initial set of codes is included in the appendix (Appendix A). In all, five key-categories and ten sub-categories were developed during this process.

3.4.2 Typology Validation. We applied the developed typology to the rest of the sample (122 threads). Two researchers (the first and the second author) discussed the definitions of the categories and the examples from the 207 threads used to develop the typology, and then independently coded the discussion starters of the 122 threads using this typology. Cohen's Kappa was 0.88 for both labels of the five key categories and the ten sub-categories.

3.5 Qualitative Analysis for In-Depth Privacy Discussions (RQ2)

In order to address our second research question "When developers talk about personal data, how do they discuss privacy-related issues", two researchers (the first and second author) collectively examined all 4772 posts in the 207 threads used for developing the typology in the analysis of RQ1, and labeled threads that contained some in-depth discussion related to privacy. During this process, the thread topic category information was not visible to the coders to reduce any potential bias.

Since developers may discuss privacy-related issues without mentioning the keyword "privacy," we used a more systematic method to identify in-depth privacy discussions inspired by the Fair Information Practice Principles (FIPPs) [3]. We looked through these posts to find discussions related to the following five aspects: Collection/Sharing, Notice/Consent, Choice/Participation, Integrity/Security, Enforcement/Accountability (see Table 2).

Table 2. This table presents five aspects of privacy that we used as criteria to identify threads that contain in-depth privacy discussions, inspired by the Fair Information Practice Principles (FIPPs).

Privacy discussion aspect	Definition
Collection/Sharing	The post discusses issues related to data collection limitation, purpose limitation, and sharing limitation.
Notice/Consent	The post discusses issues related to providing users with notice about what data is collected and how it is used by the app, and requesting user consent before collecting data.
Choice/Participation	The post discusses issues related to providing users with control over their own data.
Integrity/Security	The post discusses design and implementation considerations with respect to data security, such as data authentication, access control.
Accountability	The post discusses the accountability of developers to comply with privacy laws and app store policies for privacy.

The two coders first independently labeled all the selected threads with respect to whether each contained discussions in any of the five aspects (multiple labels could apply to each thread). Then the same two coders went through all initial labels together to discuss any difference in the labeling results and finalized a binary "privacy-related" label for each thread. Then one researcher (the first author) conducted a thematic analysis on threads that were privacy-related and identified common themes in the in-depth privacy discussions. The four authors held regular meetings to review the analysis results and to discuss the themes generated from the analysis.

3.6 Qualitative Analysis of Risky Data Practices (RQ3)

We next did a bottom-up coding analysis in order to examine the third research question: "What risky data practices (e.g., sharing data with third parties) are discussed by developers". In Android programming, there are data practices that are more likely to cause privacy harms than others (e.g., keeping all data on device) and are therefore considered as risky data practices. For example, covert data practices such as collecting data in the background and sharing sensitive data with third parties may not be expected by average users; sending data out of the phone and storing data (infinitely) may be susceptible to risks like security breaches and privacy concerns like future data use with different purposes than users gave consent for. Based on these criteria, two researchers (the first and second author) collectively examined the 207 threads used for developing the data use discussion typology (for RQ1), and labeled 109 threads mentioning risky data practices. The two coders first independently coded all the sensitive data practices that emerged from the selected threads and obtained 35 initial codes (multiple codes can apply to each thread). Next the same two coders held meetings to merge the initial codes into six high-level categories of risky data practices (presented in Table 4) and relabeled the threads (each thread could contain multiple categories).

After finalizing the labels of threads containing sensitive data practices, one researcher (the first author) conducted a thematic analysis on discussions that mentioned risky data practices to identify common themes. The four authors held regular meetings to review the analysis results and to discuss the themes generated from the analysis.

4 RQ1 RESULTS: A TYPOLOGY OF TOPICS OF THREADS THAT CONTAINED DISCUSSIONS OF PERSONAL DATA

4.1 Developers Discussed Personal Data Use on `/r/androiddev` Around Five Topics

The final typology of topics of threads that contained discussions of personal data (Table 3) contains five key-categories and 10 sub-categories. Interestingly, the first three categories happen to match three main phases of app development, which are before development (*Planning app development*), during development (*Solving technical issues*), and after development (*Seeking feedback*).

4.1.1 Planning App Development. This is the largest category in our sample (96 threads). For this category, developers often presented the general idea of an app or a certain feature that may require some personal data to implement. Sometimes, the original poster already had some idea about what data to use. For example, a person started a new thread to request help with a bus tracking app he/she/they planned to build. The keywords "locations" and "username" showed up in the discussion starter as they were the type of data that the person envisioned was needed for the app:

> *I want to build a bus tracking app for a local school where each parent would have a username and password that would allow them to track only the bus their child is on. I'll be using a mobile phone as the tracker. From what I understand, i need to send the instantaneous location of the tracker phone to a server then have the parent's app only access the **locations** after supplying the **username** and password. How do I get the device to send its location and how do I get the parent's app to view the location as it moves? I was hoping I'd have something similar to how the Uber app or Whatsapp live location works.*

At other times, developers only had a high-level goal in mind, and were looking for concrete suggestions about how to implement it. Accordingly, personal data may only appear in the replies as part of the design suggestions. For example, a person asked how to fight bots in their app, and one commenter proposed an idea of using phone number to achieve the goal, which was not mentioned in the original post:

> *If it's an android to android message app create an account I'd based on a number of factors and have the user validate at least one of those factors. Such as get them to validate the **phone number** attached to the SIM card for example and make it so only validated devices can access the service for real, shadow-ban the rest.*

In both situations, the presence of personal data keywords suggest that developers likely planned to collect these types of data for their apps.

4.1.2 Solving Technical Problems During App Development. This category characterizes threads that were created with a very specific goal: solving technical issues encountered while programming. The personal data keywords that appeared here are usually related to the problem that the developer was trying to solve. For example, the following quote describes a syntax error the developer ran into when constructing queries with users' location data:

> *I'm working on an app that queries a database with a set of longitude and latitude coordinates to get the nearest landmark to the user's current **location**. I'm using rawquery so I can get more out of my query statements, however, I seem to have a peculiar syntax error and I hope someone can help me figure it out.*

[5] This is related to a recent change in Google Play policy that restricts the the use of high risk or sensitive permissions, including the SMS and Call Log permission groups.

Table 3. A typology of topics of threads in `/r/androiddev` that mentioned certain types of personal data. The first three categories happen to correspond to three main phases of app development: before development (Planning), during development (Solving technical issues), and after development (Seeking feedback). The fourth category characterizes a common situation in Android app development when developers need to react to several external events. The fifth category characterizes threads that share development knowledge and opinions about Android programming (not focused on developing a specific app).

Thread topic (Count)	Definitions	Excerpts of the original post
Planning app development (96)		
Discussing app ideas (5)	Looking for new app ideas and/or discussing the ideas (e.g. in terms of the development cost)	"Need ideas for simple applications to practice"
Seeking help for app implementation (91)	Seeking advice and help about implementing an app idea or a feature	"I have a great idea for an App, but I'm a novice. Will you guys help me think this out?"
Solving technical problems during app development (38)		
Solving technical problems (38)	Seeking help to address technical issues encountered during app development.	"Trouble querying database with GPS coordinates."
Seeking feedback on finished apps (17)		
Seeking app feedback (17)	Posting app store links and source code and asking for feedback	"My first Android App attempt - feedback greatly appreciated"
Reacting to external events (28)		
New OS release (14)	Discussing actions required by changes in the API design of the latest operating system	"Android 8 Background Location.. thoughts?"
App store policy updates and enforcement notice (6)	Discussing how to comply with new policies of the Google Play store (the official app store) and fix violations of the policies	"I'm building an app that uses sms as a fallback for no data access to send location data using a travel/check in app. Should I be worried?"[5]
User reviews (5)	Discussing how to handle negative reviews	"Tired of these fake troll reviews and even Google won't remove them if I report them"
Privacy law updates (3)	Discussing how to comply with new privacy laws	"For CCPA compliance, if I choose 'Restrict data processing', will that put me on safe side?"
Sharing knowledge and opinions (30)		
Sharing useful resources (22)	Requesting/Sharing interesting technologies, new hardware, etc.	"Introducing Firebase App Distribution"
Opinion poll (8)	Sharing opinions about Android programming, e.g., giving feedback on API design	"What is the worst part of Android development in your opinion?"

4.1.3 *Seeking Feedback on Finished Apps.*

This category characterizes threads that were created to seek feedback on an app built by the original poster. The personal data keywords usually showed up when the original poster was describing features supported by the data, or when the commenter provided feedback on the design or implementation of a certain feature using the data. For example, in this sampled thread, the keyword "location" was first mentioned by original poster when introducing a location-sharing feature in the discussion starter:

> *Hey all - I just finished my latest version of my first attempt at an android app and would greatly appreciate all the feedback I can get. May I introduce [App name removed for anonymization] - Tip with confidence with this app that asks you questions about your dining experience and provides you with suggestions on how much you should tip. Find your current* ***location*** *and share the story of your dining experience with your facebook friends! Works on large and normal screens supporting android 2.3 and up.*

And then another person commented on this thread with some suggestions to improve the design of this feature:

> *Here are some things I'd fix: The find* ***location*** *button in the action bar - Add an icon. Some people like the text buttons, but I feel like they are too attention grabbing on a phone sized screen.*

4.1.4 Reacting to External Events. As shown in Table 3, this category is related to how developers reacted to some unexpected external events. Many of these events were new requirements for enhancing privacy in Android apps, including updates to system API designs in a recent OS release, changes in privacy laws, and modifications to app store policies. These changes affected what personal data could be used in Android apps, for example:

> *As we know that our apps can't access* ***IMEI*** *numbers of devices running Android 10 unless the app is a system app. (reference https://developer.android.com/about/versions/10/privacy/changes#non-resettable-device-ids), I wanted to know if any other substitutes can be used to identify a device uniquely for the entire lifetime of the device.*

Another type of external events was user reviews. In our sample, developers were mostly concerned about negative user reviews they received on the app store regarding personal data collected by the app or other practices of the app related to the use of personal data (e.g., implementing consent dialogs required by GDPR). For example, one person posted a screenshot of an app user giving a 1-star rating because the app collected their **IP address**, which annoyed the developer because the transmission of IP address was perceived as normal and necessary by the developer for any app that is connected to the Internet.

4.1.5 Sharing Knowledge and Opinions. In posts in this category, developers shared resources such as useful libraries, articles that introduce Android development knowledge, and their personal feelings about specific aspects of Android development. The personal data used here is mostly regarding a general condition rather than a specific app. For example, one of the threads was sharing an article introducing a security flaw of Android that may allow location tracking with only storage permission by making the app read the location metadata of photos stored on the smartphone.

> *Almost all pictures taken with any camera app tags the image with* ***location*** *metadata (if this was enabled). For proof, just open Google's Photos app, tap on the last image you took, and scroll up. If you enabled* ***location*** *tagging, you'll see a mini map of where you took this picture. This has its uses, as it allows for organizing photos based on a special place you visited, by city, or by trip. However, any app that has Storage Permissions not only has access to all your photos, but access to this same* ***location*** *metadata.*

4.2 How Active Were the Discussions of These Categories?

We calculated the average thread length (i.e., number of posts per thread) and the average number of unique posters that participated in each thread to get a basic understanding of how people participated in discussions of these threads.

For thread length, there was a statistically significant difference between topics as determined by a one-way ANOVA ($F(4, 202) = 7.584, p < .001$). The categories “Sharing knowledge and opinions” ($mean$ = 78.2, $median$ = 8.0, std = 163.2) and “Reacting to external events” ($mean$ = 46.0, $median$ = 17.0, std = 67.6) had threads that yielded the most active discussion, with average and median thread length much higher than those of the entire dataset. In contrast, the category “Solving problems during app development” had the fewest posts per thread ($mean$ = 5.7, $median$ = 4.0, std = 5.3), lower than the average thread length of the entire dataset. The category “Planning app development” ($mean$ = 8.5, $median$ = 6.0, std = 8.4) and “Seeking feedback on apps” ($mean$ = 9.3, $median$ = 5.0, std = 10.6) in general had similar average and median thread length as other posts in this dataset.

Similar trends were observed in the average number of people that participated in the discussion. There was a statistically significant difference between topics as determined by one-way ANOVA ($F(4, 202) = 8.791, p < .001$). The category “Sharing knowledge and opinions” ($mean$ = 37.7, $median$ = 5.5, std = 73.1) and “Reacting to external events” ($mean$ = 18.7, $median$ = 8.0, std = 21.6) had the most posters participated in each thread on average, and “Planning app development” ($mean$ = 5.0, $median$ = 4.0, std = 4.6), “Seeking feedback on apps” ($mean$ = 4.8, $median$ = 3.0, std = 4.6), “Solving problems during app development” ($mean$ = 3.0, $median$ = 2.0, std = 2.5) had fewer people participated.

5 RQ2 RESULTS: IN-DEPTH DISCUSSIONS OF PRIVACY RARELY HAPPENED UNLESS STIMULATED BY EXTERNAL EVENTS

In this section, we present our analysis results regarding RQ2: “When developers talk about personal data, how do they discuss privacy-related issues?”. By qualitatively analyzing the 207 threads, we identified 43 of them (21%) contained in-depth discussions of privacy (see Section 3.5 for detailed definitions of in-depth privacy discussions and the analysis process)

5.1 What Aspects of Privacy Were Discussed When Discussing Personal Data Use?

We found that all five dimensions of privacy issues, as previously introduced in Section 3.5, were mentioned in developers’ discussions. The most frequently used label was “Collection/Sharing”, which was assigned to 24 threads. Three types of privacy issues were discussed in these threads, which are issues related to purpose limitation (only collecting data with a clear purpose), collection limitation (collecting minimum amount of data to achieve the purpose), and sharing limitation (only sharing user data with external parties after obtaining users’ explicit consent).

In some cases, stricter restrictions on data access enforced by the operating system or the app store, such as forbidding access to some device IDs and accessing location data in the background, may cause developers to discuss their sensitive data use more openly on this forum. When developers asked for alternative ways to collect sensitive data that can circumvent these restrictions, some other developers may question the legitimacy of such request. For example, a post warned the developer who looked for a substitute for IMEI (a device ID) which was banned in Android 10:

> *First, ask if your use-case actually warrants usage of any unique, permanent ID of the device. If you really need it (and there are only a handful of use cases, say, an MDM), those use cases are usually covered by existing solutions/APIs. There’s a reason why Google is restricting access to these identifiers, and don’t forget that such circumvention can result*

> *in the ban of your app from the Play Store (and possibly even added to Play Protect as a malicious actor).*

Sometimes the original poster directly asked questions regarding best privacy practices. For example, a person created a thread asking whether there were any advertising networks that collect less data:

> *I'm looking to add some mediation networks to my admob banner (and interstitial) ads. I want the ads to only need the internet permission. I don't want some dodgy permissions like Writing External storage or getting location data or getting phone numbers etc. Just Internet.*

The "Notice/Consent" and "Choice/Participation" aspects of privacy were usually associated with issues about dealing with the permission system in Android. For example, regarding the run-time permission system released since Android M (6.0), a person posted a question about best practices of privacy notice in reaction to this change:

> *What are some best practices to explain why you need permissions on Android 6.0 onwards? I like to use shouldShowRequestPermissionRationale in ActivityCompat. It lets me know if the system thinks I should show an explanation of that permission. All the permissions in the apps I build are pretty obvious (like you hit the take photo button and it asks for the camera permission) so I never really have to explain too much. Otherwise, if the call returns ""true"" to show the rationale I just do a simple AlertDialog.*

The "Accountability" aspect of privacy was usually associated with discussions around privacy laws. For example, when CCPA was released in Janurary, 2020, developers who used Admob (an advertising library developed by Google) felt unclear about what they should do to stay compliant with the new privacy law:

> *Currently, for AdMob user, it isn't clear what are required to be done from developer, so that we can stay compliance to new CCPA. I was wondering, if we just select "Restrict data processing", will this put us in safe side, without performing any app code change? (I don't mind to have reduced earning, if that is a safe way to comply to new CCPA)*

To our surprise, security was not the most frequently mentioned aspect of privacy in our sample. However, after exploring the 14 threads that mentioned the security aspect of privacy, we learned that it was not because developers did not care about security, but mostly because there were less threads about new system designs or policies that were related to security. This is probably because security updates are usually more transparent to developers, which required little extra work for them. In addition, we noticed that developers seemed to be naturally sensitive to security issues in development planning, technical problem solving, and app feedback threads, especially when the related app involves data encryption, user authentication, and access control, as we later discuss in Section 6.2.1.

5.2 When Did Developers Attend to Privacy-Related Issues When Discussing Personal Data Use?

We analyzed the ratios of privacy-related threads within each of the five key-categories of data use thread topics discussed in Section 4. In all, the category "Reacting to external events" had the highest ratio of privacy-related threads (81.4%), while the other four categories had only around 10% of threads related to privacy discussion (Figure 1). This result suggests that Android developers in this forum tended to act passively in handling privacy issues in their apps. They rarely discussed privacy implications of their app when discussing app design, solving technical issues, or giving app feedback, which indicates a significant challenge in promoting developers to apply best privacy practices to design and build their apps.

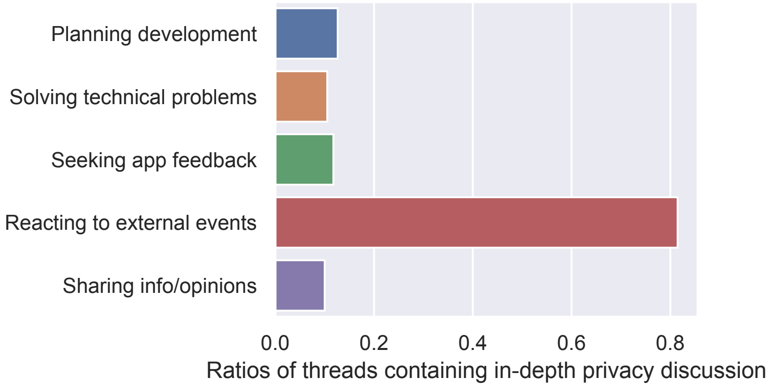


Fig. 1. More than 80% of threads in the category "Reacting to external events" had in-depth privacy discussions, while only around 10% of threads in the other four categories had in-depth privacy discussions. This suggests that Android developers in this forum tended to act passively to handle privacy in their apps, with most of them only discussing privacy-related issues when stimulated by external events such as API design change, policy change, receiving app removal notices and user reviews.

5.3 What May Affect the Efficacy of Privacy Enhancement Measures Built in the Operating System, the App Store Policies, and Privacy Laws?

In Section 4, we found that most in-depth privacy discussions were triggered by external requirements with the aim of improving privacy by the operating system, the app store, and privacy laws. Below, we present themes that emerged in our analysis of privacy-related threads of the "Reacting to external events" category.

5.3.1 Developers May Not Perceive These Solutions As Helpful to Enhance Privacy. For each new version of Android, the official documentation often provided a list summarizing privacy changes in this version. However, this documentation only focused on introducing the behaviors of the new APIs and how they will be enforced in future systems. What was missing was an upfront explanation of the rationale for these changes, especially why they would help with privacy. This often directly resulted in confusion about why bother to switch to the new APIs:

> *Scoped storage - Hey, wait?! What exact problem they are trying to solve with this?*

Even if developers knew the new design was for enhancing privacy, they may not be fully convinced. For example, one developer, commenting about the new storage framework, mentioned that they did not believe simply giving users more choices can better protect privacy:

> *Apps need to make files available for other apps - one option is to allow this legacy behavior for an app as a user choice - perhaps with a switch in android settings. Of course it comes back to the same thing - you cannot have user choice and also protect naive users from app misbehavior.*

It is worth noting that not all of these comments were factually correct, but they do represent how some developers felt about these privacy-enhancing proposals. Without efficient communication with developers that can persuade them to believe the benefits will be worth the cost, developers may have little motivation in complying with them.

5.3.2 Developers Found Some Restrictions on Accessing User Data Too Rigid, Which Can Hurt Legitimate Data Use.

Android 10 introduced more restrictions on accessing location data in the background, including forbidding background services to access location data and throttling foreground services to do so. Although the original goal of these restrictions was to prevent malicious apps from stealing users' data (e.g., surreptitiously tracking locations in the background), many developers were worried that they would also affect the legitimate functions of normal apps. For example, a person expressed concerns about Android P disabling foreground services to access location when the battery saver is turned on and screen is off:

> *That sounds reasonable, but it's really not. Think of a user of a navigation app, for example. As a user, you'd want to turn power saving mode to prolong battery life. That's understandable. However, if you decide to turn navigation on, you would like it to actually work. Regardless of device settings and the state of the screen. Maybe you even intentionally turn the screen off and only listen to navigation. Expecting that the user will understand that power saving needs to be disabled for navigation to work is ridiculous. And when navigation doesn't work, guess who's app gonna get 1-star angry review?'*

Since these restrictions would only be imposed at app development level and remain transparent to users, users may blame developers for the loss, change, and performance degradation of app functionality affected by the changes. One developer explicitly expressed this concern:

> *Google needs to inform users of reduction in features in Android Q (just like new features). Good business practice dictates that Google inform Oreo users switching to Q - failure to do so will be willful misrepresentation on the part of Google... Changes due in Android Q will shock users - Google needs to inform users (as good business practice) so users do not buy Android Q devices with presumption that local storage will behave as before.*

5.3.3 Developers Blamed Google for Not Providing Sufficient Support.

Contrary to the strict restrictions on accessing data caused by the app store policies and API designs, Google did not improve the usability of their developer-facing system, which aggravated the situation. For example, the uninformative and unpredictable app review process of the Google Play app store was a pain to many developers. A person described their experience of going through the app review process to request for approval to use a restricted permission. They first received an email saying their request was approved, and three days later, they received another email saying the same request was rejected. In the end, this person concluded that: *"The review process is complete mess. Looks like anyway they can remove your app any time...I'm fed to the teeth with this. It is too risky to develop Android apps for me now."*

Meanwhile, developers also mentioned some issues in system API design. For example, the lack of clarity of the permission system may deepen the miscommunication between users and developers, which makes it even harder for users to make informed decisions to use apps that request sensitive permissions and eventually has negative impact on privacy:

> *The whole permission system is retarded. For example you need to write simple file be it an image, sound or whatever you need storage permission. The dialog for storage permission says ' access photo media and files on your devices' which can be scary. Completely unrelated and very misleading.*

Developers also expected more support to help them comply with new privacy laws such as GDPR and CCPA. All three threads of the "Privacy law updates" sub-category of our sample was related to compliance with these laws when their apps integrated the Google's advertising network, Admob. For example, when GDPR was initially released in 2018, developers found that the data practices of Admob was too opaque for them to implement the consent dialog required by the law:

> *At this point, I'm not even sure Google knows what's happening with the user data. There is literally zero documentation about it: a) Exactly which user data is collected for Admob? b) When and by what means is this user data collected? c) Does Admob itself own and process the user data? d) What exactly does the Admob SDK do to collect user data? e) When does the processing of user data happen, ad hoc or in advance? Without this information, no developer can implement a consent dialog, because he wouldn't have a clue what's going on.*

Later Google released tools like "Google Mobile Ads Consent SDK" to help developers handle the consent requirement of GDPR. However, as more privacy laws for specific regions were enacted in recent years (e.g., CCPA that protects privacy rights of California residents), it was not clear to developers whether using this type of SDK was sufficient to help them stay compliant with the latest privacy laws. A developer mentioned that they hope there could be some generic tool for helping them handle the complicated law requirements for users from different regions (e.g., Europe, California):

> *Hopefully the consent SDK gets updated to be generic enough that any new privacy law/regulation requirements can be added without an update. From a developer point of view, this is only going to get worse.*

5.3.4 The Broken Compatibility Issue Caused Fundamental Challenges Complying With New Requirements. New privacy requirements breaking backward and forward compatibility of an app across different versions was a recurring theme in many in-depth privacy discussions on the forum. In addition, developers also found some new APIs were released without fully testing the compatibility with other parts of the system. A representative example was an API that required IMSI to retrieve network usage data. As Android 10 banned apps from accessing IMSI, this API could not function normally, yet was not updated either.

> *...many changes are not forward compatible. If your code runs on Pie, it may not run in Q. For example, IMSI number is required for querying data in NetworkStatsManager class (used for getting internet Data used and other similar stuff). But in Q, they consider IMSI number as personal identifier (which is true) and devs cannot use it in any way. Not even by asking permissions from the user. They completely blocked it from user apps and only System apps can use IMSI number. So that means NetworkStatsManager class is broken and useless in Q? I still didn't get an answer to this.*

5.3.5 There Were Also People Who Support These Changes and Try to Offer Constructive Suggestions to Help Others Adapt to Them. Although many developers had concerns about these privacy-enhancing changes, we also observed other developers that seemed to support these changes. They reacted to a thread discussing the above problems and clarify why some changes are needed to improve privacy and correct misunderstandings of some restrictions. Some of them also asked whether the restricted data access actually prevent important features from being implemented, which aligned with privacy principles such as purpose limitation and minimum data collection.

6 RQ3 RESULTS: RISKY DATA PRACTICES IN THE POSTS

As suggested in prior work [17, 20] and the results of our previous analysis of in-depth privacy-related discussions (Section 5), many developers did not know what privacy issues could occur in their apps and did not pay much attention to privacy during app development. As a result, a study of the actual data practices that developers chose to adopt in their apps could provide a complementary understanding of potential privacy issues developers didn't know about.

In this section, we present analyses to address RQ3: "What risky data practices (e.g., sharing data with third parties) are discussed by developers?" Altogether six types of risky data use emerged from 109 threads (53%) of the same sample of 207 threads as used in addressing RQ1 and RQ2.

6.1 What Risky Data Practices Were Mentioned in the Sample?

Table 4 summarizes the six risky data practices identified in our sample. The most frequently mentioned data practice was "Send data out of the device" (mentioned in 44 threads), and the least frequently mentioned data practice was "Collect data in the background" (mentioned in 10 threads). The possible risks of these data practices include violating user expectations of data use (e.g., when the data is collected in the background, or shared with external parties without explicit consent), data leaks due to security breaches, and secondary data use (e.g., when the data is uploaded to and stored on a remote server)

We noticed that the majority of these risky data practices were built into the apps for legitimate reasons. For example, certain use cases such as sharing data among users, user authentication, and fraud detection required data being sent to the server; third-party libraries sometimes were needed to address the fragmentation problem of Android for certain APIs (e.g., Camera APIs have different behavior on different devices); and in many cases, data sharing, data storage, location tracking (in the background) were part of the main functionality of the app and cannot be easily removed.

6.2 When Did Developers Bring Up Risky Data Practices in a Certain App When Discussing Personal Data Use?

We analyzed the ratios of threads that mentioned risky data practices within each of the five key-categories of data use thread topics discussed in Section 4. As showed in Figure 2, three categories had higher ratios of threads that mentioned risky data practices of a certain app, which are "Planning app development" (67%), "Seeking app feedback" (53%) and "Solving technical problems" (47%). Conversely, the other two categories had lower ratios of threads that mentioned risky data practices, which are "Reacting to external events" (26%) and "Sharing info/opinions" (13%). This is possibly because the first three categories had topics related to the development process of a specific app, while the last two categories discussed more general issues and therefore caused fewer disclosures about concrete app behaviors.

6.2.1 Security Issues of These Risky Data Practices Were More Frequently Recognized During Normal Development Activities. Among the entire 109 threads that mentioned risky data practices, 97 were threads that fell in the first three categories about normal app development activities. 12 out of the 97 threads discussed privacy-related issues and 11 out of the 12 threads were related to security issues. Conversely, none of the threads of the last two categories (about external events and sharing information) contained security-related discussions.

In our sample, most of the security-related discussions were triggered by questions regarding app designs aimed to protect user security, such as user login, fraud detection, building an app that can securely back up users' sensitive information such as photos, notes and passwords. Interestingly, some of these threads elicited discussions related to other aspects of privacy as well. For example, in an app feedback thread about an app designed for securely storing users' sensitive information and asked for feedback, there was a comment related to the "Notice/Consent" aspect of privacy, which reads: *"Privacy policy is not working, for this kind of app it is a must have"*. In another thread, a developer asked about the best practices for handling user login, and other developers not only made security-related suggestions (*"Don't ever persist a user's password (in plain text form) for*

Table 4. Six risky data practices (and the corresponding thread count) were identified in our sample. These data practices were mentioned in the posts as part of the description of a concrete app the poster worked on, or part of the suggestions of how to achieve a certain feature. The possible risks include violating user expectations of data use, data leaks due to security breaches, secondary data use on the remote server, by the third party libraries/web services, or other users.

Risky data practices (Count)	Example in the sample	Possible risks
Send data out of the device (44)	"For example, if I have retrofit set up to log the body of all of my api calls, which includes the **username** and password fields during authentication, what's the best way to be sure I'm not exposing that to any third parties."	If data is sent to the developers' own server, there is potential risk of data leak due to security breach or secondary data use without user consent if the data is also stored.
Share data with third-party libraries/services (34)	"Record the call on the phone. Push the **recording** to Googles api to put the audio into text."	How these third parties will use personal data shared with them is unclear or even intentionally obfuscated [34].
Store data on remote server (26)	"I currently have a server setup that is storing events (with time, date, **location**, etc.)"	Remote data storage has potential risk of data leak due to security breach or secondary data use without user consent.
Store data on device (25)	"I created a class called User which saves **username**, password and authorization token... The second class is called SharedManager and it takes care of everything that has to do with saving the account to SharedPreferences."	On-device data storage has potential risk of allowing malicious apps on the same device to steal the data. Sometimes the data will not be automatically removed if the app is uninstalled, which may cause violation of users' expectations of data retention period.
Share data with other users (13)	"I want the user to be able to share this **voice recording** with other users of the app"	The other user may use the data for other purposes without noticing the data subject.
Collect data in background (10)	"My app heavily relies on background tasks that need the **location** services. In this case, I'm using the workmanager API to perform this background tasks."	Background data collection may violate users' expectations of data use.

security reasons."), but also reminded the original poster about the importance of only collecting the minimum amount of data (*"For sign-up, think about what data you actually need, and don't require first name / last name, unless really necessary. Don't ask for data you don't need."*) and being transparent about the data practices to users (*"Make sure that there is a privacy policy / terms and conditions accessible from the login/sign-up screen."*)

6.2.2 Risky Data Practices Could Be Introduced for Practical Reasons Without Easy Alternatives. We found that developers often had legitimate reasons to justify the risky data practices that were used in their apps or suggested to other developers. For example, there is an app feedback thread about an app designed for tracking family member locations in real time. Risky data practices such as

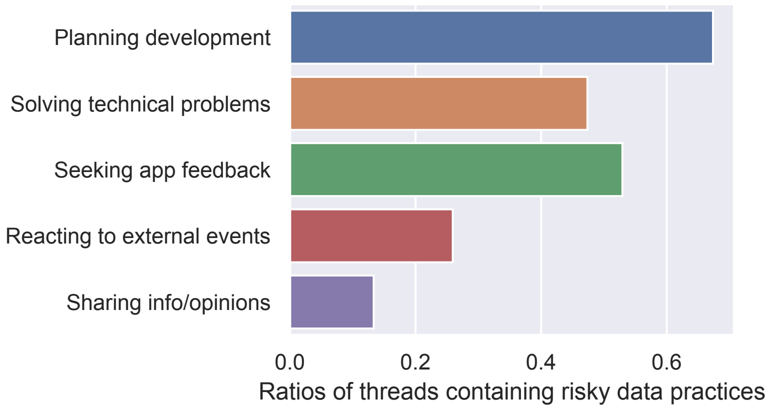


Fig. 2. The ratios of threads that contained risky data practices of a certain app were higher in the "Planning development" (67%), "Seeking app feedback (53%)" and the "Solving technical problems" (47%) categories, possibly because these topics are about the development process of a specific app.

sending location data out of the phone and processing data in the background were mentioned in the original post as they supported the main functionality of the app:

> *I have developed a family tracking concept based application where let you add the members of your family and track where they are as real-time location fetching occurs. The motivation behind this project is to* ***apply never-ending background services*** *on android with a clean architecture and* ***working with some back-end operations by using Firebase****. Check out the source code below and do not hesitate to give me feedback.*

Using third-party libraries/services can raise severe privacy concerns [11, 21] because they often resulted in unexpected data use both to users and developers [7, 20]. However, our data showed that there were many practical reasons for developers to share personal data with third-party libraries and web services. First, some standard system APIs were too difficult to use and had inconsistent behaviors across different devices (e.g., Camera APIs), so developers turned to third-party libraries to help them streamline data collection. Second, some common data processing tasks such as barcode scanning and object detection were complicated to implement from scratch, and developers preferred to use existing libraries rather than reinventing the wheel. Third, sometimes personal data had to be shared with third-party services to retrieve useful information required by the app functionality. For example, a weather app needed to share the current location of the app with a third-party web API to get local weather information (*"It's an app that retrieves the weather from openweathermap.org according to your location."*).

Although most of the risky data practices seemed to be required for legitimate purposes, there are many other things that developers need to take into consideration to conform to the best privacy practices (e.g., the FIPPs [3]). For example, clear privacy notices and effective control of data must be provided to users when the app relies heavily on sensitive personal data to function; the data collected and stored for achieving a certain purpose should be deleted when it is no longer needed. However, discussions along these lines were rarely observed on this forum.

7 DISCUSSION

Developers can have a great impact on user privacy by making different design decisions around personal data use. Current technology does not provide enough auditing capabilities to detect all privacy issues, leaving the privacy of an app heavily reliant on whether developers themselves comply with privacy requirements [24, 36].

Our analysis of /r/androiddev shows that privacy is still a challenging task for developers, with many of the privacy-enhancing measures in current Android systems failing to generate the expected values (Section 5), and potentially fewer discussions around privacy issues posted on developer forums than the actual risks (Section 5 and 6).

In the following, we discuss our findings in more depth and also talk about the implications of our results on different entities of the Android development ecosystem and offer suggestions of alternative designs for each of them to help promote better privacy practices.

7.1 Privacy Has Low Visibility in Online Developer Forums

Our analysis revealed that although developers talked about personal data use in multiple phases of app development (Section 4) and even frequently mentioned risky data practices (Section 6), they rarely discussed how to build a specific app that collects users' personal data in a privacy-respectful way (Section 5). However, there were more discussions around privacy prompted by external events for enhancing privacy (e.g., new privacy APIs, new app store policies and privacy laws) forced developers to collect less data or provide more privacy notices (Section 5.2).

This finding could lead to two main takeaways. On the one hand, despite the release of stricter privacy laws like GDPR and more privacy-enhancing restrictions imposed by the operating system and the app store, it is still challenging for developers to weave privacy considerations into their normal development activities. On the other hand, as many developers build their technical knowledge opportunistically using online resources such as developer forums [10], the low visibility of privacy issues could potentially reinforce less attention to privacy during app development.

We speculate two reasons for this problem. First, privacy is a non-functional requirement and is fundamentally less visible than other requirements such as usability and performance. Privacy API changes made privacy requirements much more visible to developers, while in a painful way which caused backlash among developers. Second, developers lack a comprehensive understanding of what potential privacy issues may occur due to their data use, which makes it unlikely for them to ask the right questions to resolve these issues. In our analysis, privacy-related discussions that spontaneously emerged in normal development activity (e.g., planning development, solving technical problems, seeking app feedback) were highly skewed towards discussions around security issues (Section 6.2.1), echoing findings of prior work that other aspects of privacy such as transparency and control have been overlooked [17, 20].

7.2 Mismatch Between the Actual Privacy Risks and Developer-Perceived Risks

An interesting observation was that personal-data-discussion categories with fewer in-depth privacy discussions had the most risky data practices. For example, 67% of threads that discussed around "Planning app development" mentioned some risky data practices such as data sharing and data retention (see Section 6), while only 12.6% of threads in this category had in-depth privacy discussions (see Section 5).

In the posts about normal development activities (i.e., planning development, solving problems, seeking feedback) that both mentioned risky data practices and privacy-related issues, most discussions were around security, which echoes prior work [17, 20] that suggested developers had a partial understanding of privacy and cared more about the security aspect of it. Furthermore, although risky data practices were motivated by seemingly legitimate purposes, developers rarely discussed important implementation details corresponding to best privacy practices such as providing effective privacy notices and controls, limiting data sharing, and reducing data retention periods [3]. The limited discussions about these best practices corroborates prior findings about the violation of best privacy practices in mobile and web apps [1, 5, 11, 16, 23].

This mismatch between potential privacy issues implied by the risky data practices and the actual discussions on privacy issues suggests that more in-depth privacy discussions are needed to help developers identify opportunities to improve privacy in their apps as early as possible in the development process. In addition, given that developers may lack a comprehensive understanding of what privacy issues could arise in their apps, it may be helpful if forums could provide concrete guidelines to help developers describe their personal data practices to elicit discussions on different aspects of privacy that can generate concrete suggestions to mitigate privacy risks of sensitive data use required by legitimate purposes.

7.3 How Can Android OS and App Stores Act to Enhance Privacy

In Section 5.3, we showed that developers raised many concerns regarding the current privacy-enhancing measures of the Android OS and the Google Play app store because these measures caused a lot more work while offering little benefit to developers. Accordingly, we want to discuss opportunities the OS and the app store may leverage to address some of the issues in two directions: increasing developers' motivation to enhance privacy and reducing the difficulty of handling privacy requirements, corresponding to two main factors of persuasive design [14].

The first two suggestions are about increasing developers' motivations to comply with the new privacy requirements. First, the maintainers of the Android operating system should consider providing a clear, upfront explanation *to app developers* of the rationale of the changes when introducing new API designs, including what problem they are trying to resolve, who will be affected, and what benefits developers can obtain by switching to the new API. This may alleviate the problem observed in our sample that some developers felt confused about why a new framework was proposed to replace the old framework or felt unconvinced about the benefits of privacy.

Second, the OS should also highlight *to end users* the privacy benefits and any potential utility costs (e.g., loss of features) due to any system-level API changes. For example, the new storage access framework (SAF) stops allowing apps to store files in certain areas of the external storage that will remain even if the app is uninstalled. Although the privacy benefits are straightforward, it may also break users' expectations of many apps' behavior that used to keep the files when reinstalling an app. In these situations, the operating system and the app store could embed clear notices about what app functionality may change before users upgrading to a new system or downloading a new app and framing it in a positive way by highlighting the privacy values. This could address developers' concerns about receiving negative user reviews due to the different behavior of the new framework and help them comfortably adapt their apps to the new framework.

The third suggestion is about reducing unnecessary hassles in complying with new privacy requirements, such as conducting comprehensive testing about the compatibility of the privacy updates with other parts of the system (see the IMSI example in Section 5.3.4); improving the OS-level permission system to facilitate user-developer communication; and improving the app review process on Google Play store. In general, we argue that more effort is needed for improving the UI/UX of developer-facing systems for developers such as the app review system. Although these systems are used by fewer people than end-user-facing systems, they impact many more people indirectly.

Lastly, stepping back, managing privacy often means extra work for developers, with very little upside for those developers and their organizations. We want to speculate about a possible new direction to consider for future systems: Can we complement current restriction-based approaches with *softer nudges* [18] that provide developers with *extra rewards* if they do better in privacy? For example, the ranking algorithm of the app store could take privacy-related metrics into account. These metrics can be very simple and generic, such as the amount of data collected and the third-party library used. This design can motivate developers to minimize their data use to compete with other apps that have similar features. This idea is similar to prior work of showing permission use of other similar apps to developers for comparison after submitting their app to the app store [26]. To further motivate developers to adopt those privacy-friendly measures, we argue it is worth incorporating those metrics to the user-facing part of the system as well.

7.4 How to Design Online Developer CoP to Promote Respectful Privacy Practices

Designing developer community of practice (CoP) to encourage more discussion of privacy is not only beneficial to the specific people who ask questions. As these websites may eventually turn into repositories of software development knowledge [6, 8], higher visibility of privacy discussions can potentially inform more future viewers of how to apply best privacy practices in a context that is relevant to their goals, and inspire them to think more about privacy.

To encourage more privacy-related discussion, forums could potentially design concrete guidelines, or even modify or create specific threads (e.g., see the "App Feedback Thread" from `/r/androiddev` that is created to gather all app feedback requests within a certain period of time, in Figure 3) to help the original posters frame their data practices in a way that can prompt more feedback from multiple aspects of privacy. For example, when asking questions about how to implement a certain feature, development forums should guide developers to include how to achieve best data practices, such as minimizing data collection, as part of their question.

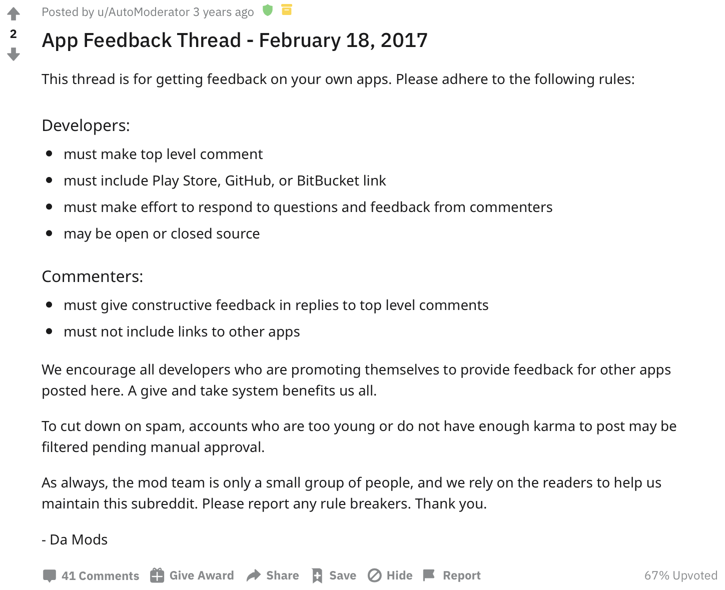
In fact, the current FAQ of `/r/androiddev`[6] already contains some privacy-related questions that are recommended to be answered *before* posting about app ideas, which are: *"Where will the data come from/be stored? How will it be used? Do I have a proper privacy policy developed?"* However, instead of just reminding developers to treat privacy as a problem that they should address before asking questions, we argue that it would be more effective if these guidelines could also encourage developers to speak up about how they are using or plan to use personal data in their apps, and try to figure out the best privacy practices for the specific use case through discussions, just like how other functional requirements such as UI design and performance optimization are being discussed on developer forums.

7.5 Limitations

There are a few limitations of our methods that may affect the generalizability of the results. First, we chose a developer forum on Reddit that is focused on Android development, therefore our conclusion may only apply to developers on this forum and to Android development. According to the demographic information from the non-scientific survey conducted on this forum introduced in Section 3.1, 71% of participants on this forum identified themselves as professional developers and only 17% identified themselves as beginners. This also corresponds to the fact that we have observed a lot of in-depth analysis of privacy features in Android on this forum. Therefore, our results may be more applicable to situations when many of the participants of a forum are expert developers.

Android and iOS had very different design philosophies from Day One. Android tends to give developers more flexibility and iOS tends to have more control over what developers can achieve. Because Android developers may be initially attracted by the flexibility, they may also backfire

[6] http://web.archive.org/web/20200609004523/https://www.reddit.com/r/androiddev/wiki/index



Posted by u/AutoModerator 3 years ago

2

App Feedback Thread - February 18, 2017

This thread is for getting feedback on your own apps. Please adhere to the following rules:

Developers:

- must make top level comment
- must include Play Store, GitHub, or BitBucket link
- must make effort to respond to questions and feedback from commenters
- may be open or closed source

Commenters:

- must give constructive feedback in replies to top level comments
- must not include links to other apps

We encourage all developers who are promoting themselves to provide feedback for other apps posted here. A give and take system benefits us all.

To cut down on spam, accounts who are too young or do not have enough karma to post may be filtered pending manual approval.

As always, the mod team is only a small group of people, and we rely on the readers to help us maintain this subreddit. Please report any rule breakers. Thank you.

- Da Mods

Fig. 3. A screenshot from /r/androiddev showing the app feedback thread was created to gather all app feedback requests within a certain period of time. We envision that the forum may be able to design a similar thread particularly for privacy suggestions and offer guidelines to help requesters frame their questions.

more when the flexibility is taken away, as in these privacy-enhancing restrictions. Therefore, our findings may only be applicable to Android development.

Since some themes were not very common, several sub-categories of the typology only got a few instances (e.g., “User reviews”, “Privacy law updates”). This prevented us from conducting systematic analysis at the sub-category level.

Another limitation of our work is related to the difference between discussion content and behavior, namely – we could only learn about what developers *said* but not verify it against what they actually *did*. This may require additional caution when interpreting our results and applying our conclusions to inspire policy and toolkit designs for real-world development activities.

8 CONCLUSIONS

We present a qualitative study of /r/androiddev, a popular subforum on Reddit focused on Android development. We studied how developers discussed personal data use and privacy using bottom-up open coding on sampled posts to develop a typology of discussions about personal data use and then conducted follow-up analyses to understand what types of posts elicited in-depth discussions on privacy issues or mentioned risky data practices. We found that developers rarely discussed privacy issues in personal data use unless stimulated by external events such as new system API designed for privacy, new app store policy updates, privacy laws, and user reviews (mostly negative reviews) about personal data use. We also found that developers frequently mentioned risky data practices related to specific app development activities but rarely discussed the potential privacy issues or how to address these issues. With our findings, we provide a set of suggestions for related stakeholders including Android OS, app store, and the developer forum to promote better privacy practices and encourage app developers to think more about privacy during app development.

ACKNOWLEDGMENTS

This material is based on research sponsored by Air Force Research Laboratory under agreement number FA8750-15-2-0281. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of Air Force Research Laboratory or the U.S. Government. This material is also based upon work supported by the National Science Foundation under Grant No. (NSF 1704087).

The authors would like to thank Jackie Yang for providing feedback on our paper. The authors would also like to thank the anonymous reviewers for their constructive feedback.

A CODEBOOK OF THE DISCUSSION STARTER TOPIC TYPOLOGY

Code	Definitions	Excerpts of the original post
App feedback	Seeking feedback on their app, self-promotion	"Can i get some feedback on my design?"
Seeking high-level programming advice	Seeking advice on design decisions, high-level issues when planning for how to implement the app, a certain feature, or achieve a certain purpose	"How to provide users 7 days free trial without asking them to signup."
Seek app ideas	Looking for suggestions on interesting ideas to work on	"Any relatively simple (a weekend or two worth of programming) app ideas?"
Discussing idea cost	Questions about estimating cost of ideas	"Cost for an idea I have, an Uber type app?"
Discussing idea feasibility	Questions about whether it is possible to implement something	"Is there a way / Why is there no way to code a voicemail app?"
Discussing idea legitimacy	Questions about whether it is legitimate/proper to implement something	"It is legal to share user contacts with another users?"
Finding tester	Finding app testers	"Looking for some volunteers for my Final Year Project."
Finding partner developers	Finding collaborators for a certain idea or project	"Who wants to create an open source Google Voice app?"
Solving technical problems	Seeking answers to a concrete, low-level technical question encountered while building an app that uses personal data, which implies that personal data needs to be used in the app that is built or going to be built by the original poster or their team	"Android Emulator - Camera Testing"
Asking about library behaviors	Asking questions about the behavior of a certain library	"How come my admob ads seem to know my location when I haven't given it any location data?"
App feedback	Seeking feedback on their app, self-promotion	"Can i get some feedback on my design?"
Asking about a new API	Asking questions about a new API	"Will this keep working? I keep hearing about a new ID that we have to use and the old one will stop serving ads after 31st August. So, what exactly is it?"

Google Policy change	Seeking advice on how to deal with google play policy change	“I’m building a travel/peace of mind app for location tracking abroad and it will have a feature of being able to use sms when data is not available to send location updates with latitude and longitude. I’ve read on here about the sms/call permissions problems happening now. Should I be worried?”
Discussing Google Policy	Discussing Google developer policy/play store policy	“Credit where it’s due, Android Play Store policy enforcement has come a long way since launch. Great works Google!”
Seeking advice on app store issues	Seeking advice on how to deal with problems encountered when releasing the app to the play store, such as policy violations, negative user reviews	“’Play Store Console: "You can’t edit this app until you create a new app release declaring sensitive permissions”
Seeking advice about new version release	Seeking advice on how to deal with new Android version release	“New permissions dialog at Android Q and Workmanager or background tasks”
Discussing new version release	Discussing new version release	“Android P will Prevent Background Apps from Accessing the Camera”
Discussing new hardware	Discussing new hardware, such as multi-camera, new form factors	“Get your app ready for foldable phones”
Discussing hardware manufacturers	Discussing issues related to manufacturers	“Huawei ban and use of 3rd party libraries for development”
Seeking advice about privacy laws	Seeking advice on how to deal with (new) privacy law requirements	“I was wondering, if we just select "Restrict data processing", will this put us in safe side, without performing any app code change? (I don’t mind to have reduced earning, if that is a safe way to comply to new CCPA)”
Discussing new privacy laws	Discussing new privacy laws	“At this point, I’m not even sure Google knows what’s happening with the user data. There is literally zero documentation about it”
Sharing resources	Sharing resources (e.g., new lib), development ideas, and development knowledge (e.g., security flaws in certain design)	“CameraKit - one of the hardest Android APIs made into a high level and easy to use library”
Opinion poll	Opinion poll about Android development, android engineering team asking for feedback for API design	“We’re on the engineering team for Android P. Ask us Anything! (starts July 19)”
Discussing other apps	Discussing the behavior of other apps	“Why does Facebook still get away with not targeting sdk 23+?”
Comparing iOS and Android	Comparing what can be achieved on iOS and Android	“Android vs. iOS Capabilities”
Discussing the behavior of some APIs	Discussing the behavior of some APIs	“Exactly how accurate is the GPS?”

REFERENCES

[1] 2020. Actions Speak Louder than Words: Entity-Sensitive Privacy Policy and Data Flow Analysis with PoliCheck. In *29th USENIX Security Symposium (USENIX Security 20)*. USENIX Association, Boston, MA. https://www.usenix.org/conference/usenixsecurity20/presentation/andow

[2] 2020. Art. 4 GDPR – Definitions | General Data Protection Regulation (GDPR). http://web.archive.org/web/20200530095018/https://gdpr-info.eu/art-4-gdpr/. (Accessed on 05/30/2020).

[3] 2020. Fair Information Practice Principles. http://web.archive.org/web/20200309081014/https://iapp.org/resources/article/fair-information-practices/. (Accessed on 05/31/2020).

[4] Rabe Abdalkareem, Emad Shihab, and Juergen Rilling. 2017. What Do Developers Use the Crowd For? A Study Using Stack Overflow. *IEEE Software* 34, 2 (mar 2017), 53–60. https://doi.org/10.1109/ms.2017.31

[5] Vitalii Avdiienko, Konstantin Kuznetsov, Alessandra Gorla, Andreas Zeller, Steven Arzt, Siegfried Rasthofer, and Eric Bodden. 2015. Mining Apps for Abnormal Usage of Sensitive Data. In *2015 IEEE/ACM 37th IEEE International Conference on Software Engineering*. IEEE. https://doi.org/10.1109/icse.2015.61

[6] Alberto Bacchelli, Luca Ponzanelli, and Michele Lanza. 2012. Harnessing Stack Overflow for the IDE. In *2012 Third International Workshop on Recommendation Systems for Software Engineering (RSSE)*. IEEE. https://doi.org/10.1109/rsse.2012.6233404

[7] Rebecca Balebako, Abigail Marsh, Jialiu Lin, Jason Hong, and Lorrie Faith Cranor. 2014. The Privacy and Security Behaviors of Smartphone App Developers. In *Proceedings 2014 Workshop on Usable Security*. Internet Society. https://doi.org/10.14722/usec.2014.23006

[8] Anton Barua, Stephen W. Thomas, and Ahmed E. Hassan. 2012. What are developers talking about? An analysis of topics and trends in Stack Overflow. *Empirical Software Engineering* 19, 3 (nov 2012), 619–654. https://doi.org/10.1007/s10664-012-9231-y

[9] Helena Béjar and Slssela Bok. 1987. "Secrets" (On the Ethics of Concealment and Revelation). *Reis* 37 (1987), 248. https://doi.org/10.2307/40183271

[10] Joel Brandt, Philip J. Guo, Joel Lewenstein, Mira Dontcheva, and Scott R. Klemmer. 2009. Two studies of opportunistic programming: interleaving web foraging, learning, and writing code. In *Proceedings of the 27th international conference on Human factors in computing systems - CHI 09*. ACM Press. https://doi.org/10.1145/1518701.1518944

[11] Saksham Chitkara, Nishad Gothoskar, Suhas Harish, Jason I. Hong, and Yuvraj Agarwal. 2017. Does this App Really Need My Location?: Context-Aware Privacy Management for Smartphones. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 1, 3 (sep 2017), 1–22. https://doi.org/10.1145/3132029

[12] Norman K Denzin and Yvonna S Lincoln. 2008. *Strategies of qualitative inquiry*. Vol. 2. Sage.

[13] Felix Fischer, Konstantin Bottinger, Huang Xiao, Christian Stransky, Yasemin Acar, Michael Backes, and Sascha Fahl. 2017. Stack Overflow Considered Harmful? The Impact of Copy&Paste on Android Application Security. In *2017 IEEE Symposium on Security and Privacy (SP)*. IEEE. https://doi.org/10.1109/sp.2017.31

[14] BJ Fogg. 2009. A behavior model for persuasive design. In *Proceedings of the 4th International Conference on Persuasive Technology - Persuasive '09*. ACM Press. https://doi.org/10.1145/1541948.1541999

[15] Daniel Greene and Katie Shilton. 2017. Platform privacies: Governance, collaboration, and the different meanings of "privacy" in iOS and Android development. *New Media & Society* 20, 4 (apr 2017), 1640–1657. https://doi.org/10.1177/1461444817702397

[16] Hana Habib, Sarah Pearman, Jiamin Wang, Yixin Zou, Alessandro Acquisti, Lorrie Faith Cranor, Norman Sadeh, and Florian Schaub. 2020. "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*. ACM. https://doi.org/10.1145/3313831.3376511

[17] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. 2017. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering* 23, 1 (apr 2017), 259–289. https://doi.org/10.1007/s10664-017-9517-1

[18] David Halpern. 2015. *Inside the nudge unit: How small changes can make a big difference*. Random House.

[19] Junxiao Han, Emad Shihab, Zhiyuan Wan, Shuiguang Deng, and Xin Xia. 2020. What do Programmers Discuss about Deep Learning Frameworks. *Empirical Software Engineering* 25, 4 (apr 2020), 2694–2747. https://doi.org/10.1007/s10664-020-09819-6

[20] Tianshi Li, Yuvraj Agarwal, and Jason I. Hong. 2018. Coconut: An IDE Plugin for Developing Privacy-Friendly Apps. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2, 4 (dec 2018), 1–35. https://doi.org/10.1145/3287056

[21] Jialiu Lin, Norman Sadeh, Shahriyar Amini, Janne Lindqvist, Jason I. Hong, and Joy Zhang. 2012. Expectation and purpose: understanding users' mental models of mobile app privacy through crowdsourcing. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12*. ACM Press. https://doi.org/10.1145/2370216.2370290

[22] Mario Linares-Vasquez, Bogdan Dit, and Denys Poshyvanyk. 2013. An exploratory analysis of mobile development issues using stack overflow. In *2013 10th Working Conference on Mining Software Repositories (MSR)*. IEEE. https:

//doi.org/10.1109/msr.2013.6624014

[23] Xueqing Liu, Yue Leng, Wei Yang, Wenyu Wang, Chengxiang Zhai, and Tao Xie. 2018. A Large-Scale Empirical Study on Android Runtime-Permission Rationale Messages. In *2018 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)*. IEEE. https://doi.org/10.1109/vlhcc.2018.8506574

[24] Kangjie Lu, Zhichun Li, Vasileios P. Kemerlis, Zhenyu Wu, Long Lu, Cong Zheng, Zhiyun Qian, Wenke Lee, and Guofei Jiang. 2015. Checking More and Alerting Less: Detecting Privacy Leakages via Enhanced Data-flow Analysis and Peer Voting. In *Proceedings 2015 Network and Distributed System Security Symposium*. Internet Society. https://doi.org/10.14722/ndss.2015.23287

[25] Helen Nissenbaum. 2009. *Privacy in context: Technology, policy, and the integrity of social life*. Stanford University Press.

[26] Sai Teja Peddinti, Igor Bilogrevic, Nina Taft, Martin Pelikan, Úlfar Erlingsson, Pauline Anthonysamy, and Giles Hogben. 2019. Reducing Permission Requests in Mobile Apps. In *Proceedings of the Internet Measurement Conference*. ACM. https://doi.org/10.1145/3355369.3355584

[27] Johnny Saldaña. 2015. *The coding manual for qualitative researchers*. Sage.

[28] Ferdinand David Schoeman. 1984. *Philosophical dimensions of privacy: An anthology*. Cambridge University Press.

[29] Awanthika Senarath and Nalin A. G. Arachchilage. 2018. 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*. ACM Press. https://doi.org/10.1145/3210459.3210484

[30] Swapneel Sheth, Gail Kaiser, and Walid Maalej. 2014. 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*. ACM Press. https://doi.org/10.1145/2568225.2568244

[31] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. 2020. Understanding Privacy-Related Questions on Stack Overflow. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. ACM. https://doi.org/10.1145/3313831.3376768

[32] Christine Utz, Martin Degeling, Sascha Fahl, Florian Schaub, and Thorsten Holz. 2019. (Un)informed Consent: Studying GDPR Consent Notices in the Field. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. ACM. https://doi.org/10.1145/3319535.3354212

[33] Bogdan Vasilescu, Vladimir Filkov, and Alexander Serebrenik. 2013. StackOverflow and GitHub: Associations between Software Development and Crowdsourced Knowledge. In *2013 International Conference on Social Computing*. IEEE. https://doi.org/10.1109/socialcom.2013.35

[34] Nicolas Viennot, Edward Garcia, and Jason Nieh. 2014. A measurement study of google play. In *The 2014 ACM international conference on Measurement and modeling of computer systems - SIGMETRICS '14*. ACM Press. https://doi.org/10.1145/2591971.2592003

[35] Etienne C Wenger and William M Snyder. 2000. Communities of practice: The organizational frontier. *Harvard business review* 78, 1 (2000), 139–146.

[36] Dominik Wermke, Nicolas Huaman, Yasemin Acar, Bradley Reaves, Patrick Traynor, and Sascha Fahl. 2018. A Large Scale Investigation of Obfuscation Use in Google Play. In *Proceedings of the 34th Annual Computer Security Applications Conference*. ACM. https://doi.org/10.1145/3274694.3274726

[37] Yuhao Wu, Shaowei Wang, Cor-Paul Bezemer, and Katsuro Inoue. 2018. How do developers utilize source code from stack overflow? *Empirical Software Engineering* 24, 2 (jul 2018), 637–673. https://doi.org/10.1007/s10664-018-9634-5

Received June 2020; accepted July 2020