Mohammad Tahaei*, Tianshi Li, and Kami Vaniea

Understanding Privacy-Related Advice on Stack Overflow

Abstract: Privacy tasks can be challenging for developers, resulting in privacy frameworks and guidelines from the research community which are designed to assist developers in considering privacy features and applying privacy enhancing technologies in early stages of software development. However, how developers engage with privacy design strategies is not yet well understood. In this work, we look at the types of privacy-related advice developers give each other and how that advice maps to Hoepman's privacy design strategies.

We qualitatively analyzed 119 privacy-related accepted *answers* on *Stack Overflow* from the past five years and extracted 148 pieces of advice from these answers. We find that the advice is mostly around compliance with regulations and ensuring confidentiality with a focus on the `inform`, `hide`, `control`, and `minimize` of the Hoepman's privacy design strategies. Other strategies, `abstract`, `separate`, `enforce`, and `demonstrate`, are rarely advised. Answers often include links to official documentation and online articles, highlighting the value of both official documentation and other informal materials such as blog posts. We make recommendations for promoting the under-stated strategies through tools, and detail the importance of providing better developer support to handle third-party data practices.

Keywords: software developers, stack overflow, usable privacy, privacy by design, privacy design strategies

DOI Editor to enter DOI

Received ..; revised ..; accepted ...

***Corresponding Author: Mohammad Tahaei:** University of Bristol, E-mail: mohammad.tahaei@bristol.ac.uk. The work was conducted and manuscript prepared while the author was at University of Edinburgh.

Tianshi Li: Carnegie Mellon University, E-mail: tianshil@cs.cmu.edu

Kami Vaniea: University of Edinburgh, E-mail: kami.vaniea@ed.ac.uk

1 Introduction

Privacy regulations such as the General Data Protection Regulation (GDPR) [46] and the California Consumer Privacy Act (CCPA) [43] have been enacted in recent years, obliging software developers to take actions to protect users' privacy. However, privacy violations are still common in today's apps [22, 35, 38, 42, 49, 56, 62, 69]. Meanwhile, researchers have created frameworks to help synthesize guidelines and strategies for incorporating privacy values from the initial stages of design, such as Privacy by Design [13, 14, 28]. Efforts have been made to translate Privacy by Design principles into concrete design guidelines, such as the *privacy design strategies* by Hoepman [27, 28]. However, the extent of adoption of privacy design strategies and their associated privacy enhancing technologies by developers is yet unknown. The contrast between the proliferation of privacy framework research and the pervasiveness of privacy violations in the wild calls for systematic research into developers' awareness, the usefulness, and operationality of these frameworks [60, 64].

Developers, on the front line of building apps and services, oftentimes discuss programming-related issues in online forums (e.g., Stack Overflow [59] and Reddit [33]). These informal discussions have become one of the resources of practical knowledge of software development [4, 9, 47], which suggests information circulating on these platforms has significant impact on how developers handle privacy and security in practice [1, 21, 33, 59]. Prior analysis of privacy-related questions on Stack Overflow shows that developers find it challenging to handle privacy requirements including writing and maintaining privacy policies and also dealing with access control requirements [59]. The impact of advice on decision-making, particularly around security and privacy, has been studied from an end-user perspective (e.g., choose a strong PIN for your phone [10, 48]) and also from a security correctness perspective by looking at the impact of Stack Overflow's security-related code on mobile apps [1], but not yet regarding privacy from a developer's point of view. Therefore, we do not yet know what privacy advice developers give one an-

BY-NC-ND

other to address privacy challenges and how the advice would fit into existing privacy frameworks.

In this paper, we present an analysis of privacy-related posts on Stack Overflow using a new angle: focusing on the accepted *answers* to privacy-related questions on Stack Overflow. Studying privacy answers offers us a window into what methods developers employ to resolve privacy issues and how they apply them in various contexts to strengthen our understanding of the gap between privacy in the research community and privacy in practice. Specifically, we aim to investigate two research questions to contribute to the evolving field of developer-centered privacy:

RQ1: How does the privacy advice developers give each other relate to the privacy design strategies by Hoepman [27, 28]?

RQ2: What advice/information do developers give one another on Stack Overflow to address privacy issues?

We qualitatively analyzed 119 privacy-related accepted answers on Stack Overflow and extracted 148 pieces of advice that developers give one another to accomplish privacy-related tasks and provide privacy to their users. We find that some privacy design strategies, `inform`, `hide`, `control`, and `minimize`, are advised frequently by developers, and some strategies, `abstract`, `separate`, `enforce`, and `demonstrate`, are rarely advised. Our results suggest that the under-stated privacy design strategies need to be promoted by improving privacy education to increase developers' awareness of these strategies and building tools to help developers adopt these strategies in software development.

Our findings show that *complying with regulations* and their consequences, and approaches to *ensuring confidentiality* are two common themes of advice. Most advice was related to relatively traditional privacy enhancing technologies such as asking for user consent, access control, encryption, and stripping personal information to de-identify data. On the contrary, novel technologies such as differential privacy and federated learning were rarely proposed as solutions to privacy-related software development questions on Stack Overflow.

Furthermore, existing privacy frameworks are often focused on mitigating privacy risks related to first-party data practices, while we observed a large portion of discussions regarding practical techniques to protect privacy when using third-party services or libraries. We hence reflect on existing privacy design frameworks and privacy enhancing technologies. We suggest that the importance and challenges of handling third-party data practices need to be emphasized in these frameworks, and practical developer tools are needed to help developers understand and control third-party data practices.

2 Related Work

Our work contributes to the evolving field of developer-centered privacy [2, 3, 28, 30, 32–34, 52, 55, 59, 60]. We detail the relationship and difference between our work and two lines of related research below.

2.1 Privacy by Design for Developers

Originally proposed in 2009 [13], Privacy by Design has become a widely acknowledged reference framework for building privacy-friendly systems [12]. The fundamental principle of Privacy by Design is that developers should consider privacy requirements throughout the entire development process and take proactive measures to avoid privacy risks rather than remedy them after they have occurred [12–14]. Accordingly, researchers have explored two directions to help developers achieve the high-level standards set by the Privacy by Design framework.

One line of work is focused on the design process, which proposes and studies design patterns to help designers and system developers translate the Privacy by Design framework into design requirements and guidelines before diving into the implementation phase [16, 26–28]. In our work, we use the privacy design strategies proposed by Hoepman [27, 28], because it is directed to developers and designers, as well as being well-cited. It details eight privacy design strategies:

- `Minimize`: "limit as much as possible the processing of personal data."
- `Separate`: "separate the processing of personal data as much as possible."
- `Abstract`: "limit as much as possible the detail in which personal data is processed."
- `Hide`: "protect personal data, or make it unlinkable or unobservable. Make sure it does not become public or known."
- `Inform`: "inform data subjects about the processing of their personal data in a timely and adequate manner."
- `Control`: "provide data subjects adequate control over the processing of their personal data."
- `Enforce`: "commit to processing personal data in a privacy-friendly way, and adequately enforce this."

– `Demonstrate`: "demonstrate you are processing personal data in a privacy-friendly way."

Another line of research investigates privacy enhancing technologies at the implementation level to assist developers in achieving Privacy by Design requirements [27]. For example, a privacy threat analysis framework can guide developers to select appropriate privacy enhancing technologies to achieve soft privacy properties such as policy and consent compliance and hard privacy properties such as unlinkability, anonymity, and plausible deniability [15]. Recently, the research about different types of privacy enhancing technologies is on the upswing, including usable privacy research that investigates the design and privacy notice [19] and control [68], conventional technical privacy enhancing technologies such as authentication and access control, and novel privacy enhancing technologies such as differential privacy [17] and homomorphic encryption [41].

Efforts have been made to understand developers adoption of privacy frameworks and privacy enhancing technologies. Developers' language in conceptualizing privacy is often limited to the security vocabulary [25, 60] and they prefer technical measures like data anonymization over providing privacy policies [53]. Experienced developers in novel privacy enhancing technologies (e.g., homomorphic encryption, differential privacy, and secure multi-party) state that the mathematical and computational complexity of these technologies made it difficult to explain these technologies not only to end-users but also to developers, investors, product managers, and policymakers [2].

Our work extends this literature by providing a complementary analysis that looks at developer discussions on Stack Overflow instead of self-reported interviews and surveys which may suffer from social desirability bias where subjects may not report unorthodox behaviors or thoughts that they think the interviewer or society may judge them negatively for. Rather than directly asking developers, we leveraged the privacy advice from Stack Overflow as a proxy to empirically study the adoption of these strategies by developers in the wild. Using this different method, we are able to identify privacy design strategies that are suggested by Hoepman's guide but rarely adopted by developers (e.g., `separate` and `abstract`), which calls for improvement in how we make developers aware of privacy strategy options, the usability of developer tools, privacy laws, and platform policies, operationality of privacy frameworks, as well as a consideration of what parts of the framework are better suited for organizational-level audiences rather than developers, for example, `enforce` and `demonstrate` may be more challenging for developers to do on their own.

2.2 Privacy Discussions in Developer Forums

Online developer forums are a type of community of practice where developers informally discuss programming-related issues and learn from one another [33]. They serve as a major source of knowledge for developers [4, 9, 47] and have therefore provided a window into how developers handle programming-related tasks in the real world. Specifically, researchers have investigated these forums to identify popular topics of security and privacy and challenges for fulfilling security and privacy requirements [23, 33, 54, 59, 66].

For example, Tahaei et al. [59] studied privacy-related questions on Stack Overflow, and found that common privacy-related topics discussed on the platform include privacy policies, access control, and encryption. Developers find it challenging to write privacy policies required by software development platforms such as Apple and Google as well as adhere to the platform's other privacy requirements. To do so, they ask questions on Stack Overflow, where they asked other programming questions, to get help and fix errors and exceptions raised by the platforms (e.g., including a privacy policy in app's website or including a description for requested permissions).

As another example, Li et al. [33] identified potential issues that could hinder developers from building privacy-friendly apps by analyzing posts that mentioned personal data use from /r/androiddev, a subreddit themed around Android development. Privacy-related discussions rarely emerged spontaneously with regard to troubleshooting or improving the privacy of specific apps. Instead, they were mainly triggered by privacy requirements from the Android operating system, app store policies, and privacy laws. Developers had trouble understanding privacy requirements and complained about the inconvenience and lack of support for complying with these requirements.

Unlike prior work that studied privacy questions to extract developers' privacy challenges [33, 59, 66], our work employs a different angle, focusing on accepted answers to privacy questions to examine the information sources, coverage, and level of details included in the privacy advice offered (and likely used) by the developer community in practice. Our findings help gain

insights into how developers use privacy enhancing techniques to address privacy issues in the wild and identify developers' knowledge blind spots and misconceptions.

3 Method

We qualitatively analyzed 170 privacy-related posts in Stack Overflow that had an accepted answer and were posted in the past five years.

3.1 Dataset

We collected 170 privacy-related posts on Stack Overflow created between April 2016 to April 2021 using Stack Exchange's API with two conditions: the post must have a "privacy" tag and it must have an accepted answer. We selected posts with a privacy tag as we were interested in posts Stack Overflow users define as privacy-related rather than looking for specific keywords or terms related to privacy. No sampling was used and all posts that passed on criteria were included.

As we were interested in the solution that the asker judged as fixing their problem, we only focused on the accepted answers. Stack Overflow defines accepted answers as: "As the asker, you have a special privilege: you may accept the answer that you believe is the best solution to your problem" [45]. Out of the 170 posts, 124 posts only had an accepted answer. 46 of the posts had 2 or more answers, the accepted answer had the highest or equal number of votes in 35 of these and in 11 posts a non-accepted answer was voted highest. When a non-accepted answer was highest voted, the difference between it and the accepted answer's votes was only 3 ($SD = 5.1$) on average.

Initially, we did not limit the data to a specific date (465 posts). However, after reading some posts, we realized many of the older posts in the dataset had obsolete information and no longer provided useful information (e.g., "uniqueIdentifier property is deprecated in iOS5 and you should not use it now. As an alternative you can generate your own unique ID" [1476155]). Therefore, we limited our data to the past five years (170 posts).

3.2 Ethics

The research was approved through our institute's ethics procedures. We followed Stack Overflow's guidelines for running academic research. Stack Overflow encourages researchers to use its data to produce academic papers [8] and requires researchers to give attribution to posts using a direct link to them [6]; therefore, we use hyperlinks to link our quotes to the original answers.

All posts on Stack Overflow are under Creative Commons with the following requirements [7]: "You are free to Share — to copy, distribute, and transmit the work to Remix — to adapt the work Under the following conditions Attribution — You must attribute the work in the manner specified by the author or licensor (but not in any way that suggests that they endorse you or your use of the work). Share Alike — If you alter, transform, or build upon this work, you may distribute the resulting work only under the same or similar license to this one."

3.3 Analysis

We qualitatively coded the accepted answers and also read the questions for context. Two authors first read a 5% random subset of the data and identified initial interesting themes and concepts to code. Then, they met to discuss potential concepts for further coding while also discussing the findings with the third author and getting inspirations from similar works in analyzing online forums directed to reverse engineers [63], Stack Overflow privacy-related questions [59], Reddit privacy-related subs [33], and security advice on the Internet [48]. We then decided to look for the *level of detail* the answerer provides and which *privacy design strategies* the answer would fit into. The level of detail in the answers was selected for coding because we observed a difference between posts that provided information about a technology without a solution and others that suggested a solution to a technical problem, for example, by giving a code sample. For privacy design strategies, we used the strategies and suggestions by Hoepman's "Privacy Design Strategies" [27, 28]: `minimize`, `separate`, `abstract`, `hide`, `inform`, `control`, `enforce`, and `demonstrate` (See Section 2.1 for details) because it is primarily directed to designers and engineers. We built our initial codebook based on these strategies.

We also open coded four aspects of the answer: (1) *given advice* or solution using imperatives for the codes such as "delete parts of data," "use regular expressions," and "use an encryption algorithm," (2) a potential corrected *misconception*, because some answers addressed and corrected a misconception in the question, (3) the *information sources* and provided links, and (4) who is the *target audience* of the advice, developer or the end-

user, as in some cases askers were looking for a solution for their own privacy issues such as protecting their privacy in Git commits by removing their emails.

Two authors then independently coded another four random sets with 5% of data with the initial codebook and calculated the Gwet's AC1 [24]. This measure was a suitable measure for our data because we had a high agreement in some codes, such as *separate* and *target audience*, and Cohen's Kappa may not work well in these situations resulting in "Cohen's Kappa Paradox" (i.e., having a high agreement but a low Kappa) [24, 65, 67]. Disagreements were then resolved through discussion, and minor changes were made to the codebook. In the last iteration, the coders achieved an average agreement of 88% across the codebook which is considered good agreement [50]. Then, they split the data into two parts and separately coded the data. To do a final check, they both coded the last 5% of data independently and again calculated Gwet's AC1, which resulted in a good agreement (81%). Multiple codes were allowed per answer and the reported agreements are an average of agreement for all codes across the codebook.

After coding all the data, three authors together used affinity diagrams [11, 31] to construct themes around the open-codes (a code may appear in multiple themes). Section 4.3 and Section 4.1.1 are based on the resulting themes from the affinity diagram.

3.4 Limitations

While our study looks at one source of information and advice for developers, other resources such as Twitter, Reddit, and LinkedIn are also examples of forums and Q&A websites that developers may look for advice on. Since Stack Overflow is one of the information sources that developers use to build apps it can impact apps' security aspects [21]; we believe our study can provide insights into privacy-related advice that developers give one another on the Internet. However, as developers come from a wide background and working situations (e.g., large companies vs. smaller companies) the results may not be generalizable to all developers. Future research may want to look at other resources and conduct a comprehensive review.

We analyzed accepted answers as we believe those solutions fixed the asker's problem. Another choice could have been to focus on answers with the highest number of votes. We chose to not use vote count though because such answers might have evolved over time as technology changed or might have been proposed sometime later based on the comments or other users' interactions. Most accepted answers were also the highest voted, with 124 out of the 170 posts having only an accepted answer and 46 had at least one more answer in addition to the accepted answer.

Using privacy as a keyword limited our research to posts that developers considered a privacy-related challenge and concern. One caveat of this decision is that there are very likely posts on Stack Overflow that are related to privacy conceptually (e.g., information minimization) but the asker had not tagged it with privacy. Our research is partially inspired by privacy frameworks which would only be sought out or used by a developer who knew they had a privacy-related issue. So we purposely limited our focus to askers who believe that their question pertains to privacy. This approach also allowed us to have the broadest definition of "privacy" by allowing the askers to indicate what they felt was privacy-related, rather than the researchers making that judgment. Future research may want to take a random sample from a developer forum without filtering for specific keywords and thematically analyze posts to determine what topics are related to privacy that the asker has not tagged as privacy.

4 Findings

When looking at the target audience of the post, we observed two types of posts, one focusing on developers trying to protect their own privacy ($N = 48$), and the other focusing on their users' privacy ($N = 119$), similar to the findings in prior research [59]. Because our research questions were targeted at how developers advise one another to protect their users' privacy, our results were built on top of the latter type of post (users' privacy, $N = 119$). We also removed three posts because they were about public/private variables and not related to privacy. Throughout this section, we refer to the answers with their unique identifier on Stack Overflow.

4.1 Answerers and Their Answers

The accepted answers in our dataset where provided by 94 unique answerers. On average, they provided 1.2 accepted answers each ($SD = 1.7$, $max = 17$). The continents associated with the 65 answerers who included a location on their public profile were: Europe: 31, North America: 18, Asia: 13, South America: 2, and Oceania:

1. Despite a decrease in the number of accepted answers in 2019, we observed an overall increase in the number of accepted answers over the past five years (Figure 1).

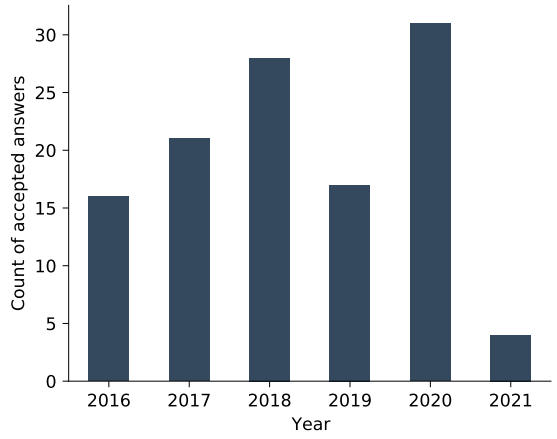

Fig. 1. Count of accepted answers per year. 2021 is partial and covers up to the month April.

4.1.1 Information Sources

We open coded situations where the answerer provided specific links or references to external resources, resulting in 84 answers containing this type of data. They provided links to the official documentation of tools and APIs (41/84) (e.g., Apple and Google), articles (21/84) (e.g., personal and business blog posts, Wikipedia, news articles, tutorials from books, and Internet Engineering Task Force Request for Comments), privacy tools (10/84), GitHub repositories (9/84), and other Stack Overflow posts (4/84). These numbers are in line with prior research on privacy-related questions on Stack Overflow [59].

While in the case of security, having access to official documentation may result in writing secure code, they are difficult to use and may result in less functional code [1]. The majority of answers pointing to official documentation highlights the perceived value developers place on that type of documentation suggesting that providing high-quality and usable documentation to developers is valued. It may also signal that there is growing body of unofficial "shadow documentation" [47, 59] building on Stack Overflow for privacy issues to clarify, add context, and provide examples in parallel to official documentation. In the case of privacy, future research may want to explore the usability of privacy-related code in official documentation for developers.

4.1.2 Provided Level of Details

We observed three levels of detail in the answers, which are high-level opinions and general feedback (24/119), starting points (69/119), and step-by-step guidance (26/119). Only 26 out of the 119 answers (21%) provided step-by-step guidance that developers can directly apply to solve the problem. Conversely, 69 out of 119 answers (58%) provided solely starting points, which may speak to the complexity or context knowledge requirements of implementing a detailed solution. However, the lack of clear directions on how to solve the issue can create a barrier to developers adoption of the privacy advice, as they need to do further research on their own.

High-level opinions and generic feedback (24/119)
Answerers spoke about their view of the problem without providing much detail about how to do a certain task and instead explaining why a certain technology is used or why certain behavior happens. An example quote is "Cookies can be set by response headers, therefore any website resource outside your control can set his cookie. Of course cookie will be visible/accessible only for its domain (not Yours)." [50856878]

Starting points (69/119)
These answers included links to external resources (Section 4.1.1), proposed the setting of required flags and small code snippets, and directions about how to accomplish a task. An asker would need to do some research or some changes to the provided code snippet to get to the solution:

> If the data you are handling is at all sensitive, you should conduct a data protection impact assessment (DPIA) and/or privacy impact assessment . . . . CNIL (the French data protection office) has an excellent PIA app [hyperlinked] that walks you through the process. [54408565]

Step-by-step guidance (26/119)
These answers tended to contain specific details, code samples, and how-to dos. The level of details also provided are generally enough that an asker could follow

the answer and get their task done with minimal tweaks and changes or further research:

> To store data like password and key, azure recommend us to use azure key vault [hyperlinked]. So I suggest you to use key vault store your password and key, and then get the key vault in your logic app . . . . [provides a list of screenshots]. [65351254]

4.2 Applied Privacy Design Strategies

Out of the 119 answers, we coded 111 of them with the eight privacy design strategies (RQ1, Table 1). We could not extract strategies from eight answers (See Section 5.2 for details). The most common strategy was `inform` (48/111), which may come from developers trying to adhere to the requirements imposed by platforms, such as informing users about permissions, including a privacy policy, and asking for user consent. `Hide` (45/111) was the second most commonly used strategy, which applies methods, including encryption and hashing, to ensure confidentiality. While the earlier mentioned `inform` strategy only informs users about actions or decisions, the `control` (35/111) strategy also provides options and control to the user to take action; consent pop-ups may fall into this category if they provide control. `Minimize` (33/111) was also used to provide anonymity to users by collecting less data or not collecting data at all.

Other strategies were not as common, and only a few answers referred to `abstract` (5/111), `separate` (3/111), `enforce` (2/111), and `demonstrate` (2/111). An example of the `abstract` strategy was removing parts of data and masking to ensure anonymity: ". . . you are probably looking for Dynamic Data Masking [hyperlinked]." [41999638] `Separate` suggests storing and processing data in multiple places instead of using a central system; in our dataset, it is related to using blockchain technologies such as Hyperledger. The other two strategies, `enforce` and `demonstrate`, primarily look at organizational level privacy. We hypothesize that we observed these two strategies rarely because they sit at a higher organizational-level while Stack Overflow is targeted at developers and programming problems.

We found that answers often provided solutions for protecting users' privacy; however, in a few posts, we saw instances of privacy-unfriendly practices, such as using the camera when the app is running in the background and appending parameters to URLs for tracking.

Table 1. Number of occurrences per privacy design strategy in the accepted answers.

Privacy design strategy	Occurrences
Inform	48 (43.2%)
Hide	45 (40.5%)
Control	35 (31.5%)
Minimize	33 (29.7%)
Abstract	5 (4.5%)
Separate	3 (2.7%)
Enforce	2 (1.8%)
Demonstrate	2 (1.8%)

4.3 Advice for Applying Privacy Design Strategies

We extracted 148 pieces of advice from the accepted answers (RQ2), 11 of which were hard to interpret because of lack of context or being too broad. We also found 21 misconceptions where the answerer explicitly corrected the asker's understanding of the problem, two of these were removed because they were hard to interpret. Therefore, the results of this section are built on 137 open-coded advice and 19 open-coded misconceptions. The main themes that came out of the data were regarding legal issues, third-party services, confidentiality, data collection, and a miscellaneous theme where we clustered smaller sets of advice that did not belong to other themes (Table 2). In each theme, we also report misconceptions, if there were any.

4.3.1 Be Compliant With Regulations and Their Consequences (advice=53/137, misconceptions=4/19)

The largest group of advice was around legal, regulation issues, or platform requirements that were also introduced by regulations such as GDPR and CCPA.

Check if your privacy policy is compliant (17/137)
Answers here suggested including a privacy policy and letting users know what happens to their data, using third-party tools to create a privacy policy, following the specifications and guidelines required to build a privacy policy, and checking the privacy policy of integrated services: "If you have any third party dependencies you need to check the documentation for those (or possi-

Table 2. Constructed themes around the open-coded advice with their occurrences (total number of open-codes=137).

Theme	Occurrences
Be Compliant With Regulations and Their Consequences	*53 (38.7%)*
Check if your privacy policy is compliant	19 (13.9%)
Inform users when requesting permissions	14 (10.2%)
Ask for user consent	9 (6.6%)
Be aware of data practices of third-party services	5 (3.6%)
Seek legal guidance	4 (2.9%)
Run a data protection impact assessment	2 (1.5%)
Ensure Confidentiality	35 *(25.5%)*
Control who has access	17 (12.4%)
Use encryption	9 (6.6%)
Use HTTPS	6 (4.4%)
Use hashing	3 (2.2%)
Avoid Collecting Data	*25 (18.2%)*
Strip personal information or use fake data	15 (10.9%)
Avoid having or storing data	6 (4.4%)
Use the correct configuration to avoid data collection	4 (2.9%)
Read About Third-Party Services	*15 (10.9%)*
Analytics code may need extra permissions	7 (5.1%)
Don't use analytics or use a privacy-friendly alternative	6 (4.4%)
Check ad networks' policies and permissions	2 (1.5%)
Miscellaneous	*5 (3.6%)*
Use custom solutions	3 (2.2%)
No action is needed	2 (1.5%)

bly use the web browser's debug tools to verify) that it does not store cookies nor send off request to third parties." [62406722] The percentage of answers in this theme (13.9%) is consistent with prior work on questions about managing privacy policies on Stack Overflow (13% of privacy-related questions [59]).

It is also notable that platforms may require disclosure about sensitive data use beyond privacy policies. This situation led to one misconception that including a privacy policy is the only requirement to avoid violating Google Play's policies for a money management app, when actually further *in-app* disclosure about the use of data was also required.

Inform users when requesting permissions (14/137)

These answers primarily suggested the asker include a description in the configuration files to tell their users why they are requesting this specific type of permission and how they would use it: "Before requesting permission you could display an alert that explains why you are going to request permission. You can use whatever text you like in this alert." [50323089]

This advice is in line with the recommendations of usable privacy research to explain the purpose of data use to users [36]. However, we observed that the advice was mostly given to fix the error of a missing required data use string field in iOS apps to satisfy the requirement imposed by the operating system, rather than trying to improve users' awareness of data use.

One corrected misconception was that the original poster believed they could change the purpose description of a permission request during runtime, while the description must be statically defined in configuration files and therefore can not be changed during runtime.

Ask for user consent (9/137)

This theme is about answers that discussed asking for user consent before collecting sensitive data. Some answers stated that it is necessary to ask for consent explicitly due to legal and embedded system requirements (e.g., data collection by mobile APIs is protected by permissions). We also observed a misconception here that the original poster had only provided key descriptions in the info.plist but did not know they need to call the authorization request APIs to trigger the permission alert programmatically. The accepted answer corrected this misconception by saying: "Based on your comments below, you need to know how to programmatically request authorization so the alerts can be responded to." [40494067]

On the other hand, a few answers suggested that consent is not always needed for collecting data from a compliance perspective. For example, the following answer discussed the legal basis of data collection from GDPR's viewpoint and discussed when consent should or should not be used:

> Consent is normally applied to optional things – for example opting-in to marketing emails while buying something – where the additional processing is not a requirement for the primary purpose of the data collection. Consent should not be used unnecessarily because unlike the other bases for processing, it can be withdrawn unilaterally at any point by the data subject. [54364758]

Be aware of data practices of third-party services (5/137)

Frequently, developers use services provided by multiple companies to build their apps (e.g., using libraries, servers, and analytics services). Using multiple services

may cause some issues with how these companies handle users' data. Answers here argued that developers had the responsibility to figure out how third-party services may affect users' privacy. However, there is no one-size-fits-all solution, and developers have to read the terms of services and privacy policies of these services to figure out the implications and make sure that these services are in line with what developer are required to provide to their users. The following example highlights the complexity of modern software development and the burden it imposes to developers:

> . . . because Heroku is a managed app service, it means that they get more access than a typical VM would have. You then need to read their privacy policy [hyperlinked], which presents a problem: Heroku is owned by Salesforce.com, who have taken a belligerent Facebook-style head-in-sand denial approach to recent court verdicts in this doc [hyperlinked]. [65431027]

The quote also highlights a tone observed from answerers that the full privacy story is not always possible to find in documentation or privacy policies. Though from a legal compliance perspective these might be reliably used. But from a moral or user-protection perspective, the third-party's track record on privacy issues should also be considered when making decisions around where to send data or what to include in programs or apps.

Seek legal guidance (4/137)

These suggestions were around seeking legal guidance either by asking a lawyer or reading the regulation documents: "You can not take this word to word [provides screenshots of a privacy policy], you need legal advice." [44116236] Legal matters can be complicated and beyond the normal training of a developer, "seek legal advice" seemed to be a phrase to use when the questions were too involved or complex.

Run a data protection impact assessment (2/137)

A few answers suggested running a privacy assessment in some instances:

> In special category situations, GDPR requires that you conduct a data protection impact assessment (DPIA) and/or a privacy impact assessment (PIA) before implementing solutions, so that you you are able to justify your decisions should an information commissioner ask for it. [61716297]

Such assessments are intended to provide a set of checklists to understand, identify, and minimize data protection risks. They are recommended when a project is likely to cause risks to individuals [44].

4.3.2 Ensure Confidentiality (advice=35/137, misconceptions=5/19)

While trying to look after users' privacy, developers discussed using techniques such as access control, encryption, and hashing to ensure their users' confidentiality.

While prior work shows 40% of privacy-related questions on Stack Overflow were about access control [59], in our dataset, 12.4% of the accepted answers cover this topic which may signal that over half of the access control questions may not have an accepted answer.

It is further notable that in prior work 3% of privacy-related questions were about encryption [59] compared to our finding of 6.6% accepted answers being about encryption. If we combine our three encryption-related themes (i.e., encryption, HTTPS, and hashing) into one, then we have 13.1% of accepted answers involving encryption compared to 3% of encryption-related questions from Tahaei et al. [59] which may suggest that answerers are recommending encryption-related solutions to the questions that do not mention encryption, HTTPS, or hashing.

Control who has access (17/137)

These answers suggested using correct configurations to make limited access to sensitive resources, building multiple versions of an app for local/private and public use respectively, and storing data internally in the app to preserve privacy. This type of answers usually demonstrated in-depth understanding about a particular platform or framework, as the implementation of access control is highly dependent on the corresponding platform and framework. For example, an answer explained how to share private files with other apps on Android 10 or newer versions referred to specific flags and methods:

> You should be crashing with a `FileUriExposedException` on Android 7.0+, so you already applied a hack to get around that. It's just that now, on Android 10+, that hack has limited value. So, replace `Uri.fromFile(file)` with uri. Also, include `addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION)` as part of your apply {} lambda. And find where in your code you are configuring `StrictMode` and have it complain about `FileUriExposedException`, at least on debug builds. [61343223]

Four misconceptions around access control here are about not knowing what data would be available publicly, and missing the security risk that giving access to a resource could help attackers access to other resources. For example, the answerer here points out security issues that may happen if source maps are made public:

> . . . your code should be secure against an attacker that has the source code, this will not always be the case in reality, there will be vulnerabilities, and those will be easier to discover for an attacker if source maps are available. [44336792]

Use encryption (9/137)
We observed answers that put forth using encryption as a general security recommendation to protect databases, passwords, and audio/video streams, as well as answers that discussed specific issues related to the implementation of encryption such as performance issues, key management method, and which encryption library to use. For example, an answer recommended using symmetric-key encryption for encrypting stored data due to the performance benefits:

> Devices typically contain enough storage that needs protection to warrant the use of a symmetric key algorithm. Public key crypto is way too slow for large amounts of data. If it's e.g. a harddisk, even a block chaining of the encryption is quite counterproductive. [42239048]

Use HTTPS (6/137)
We created a separate theme for the use of HTTPS, as it was frequently mentioned as advice to build privacy-friendly web-based services. Possibly because it is relatively easy to implement and a good way to encrypt data in transit. For example: "Make sure you have set all applicable HTTP security headers [hyperlinked], and (if you're not already) you should be using HTTPS, even for a static site." [52497207]

Use hashing (3/137)
We found three pieces of advice that suggested using hashing to obfuscate personal data such as email address, phone number or passwords. Some was just high-level advice and did not provide concrete guidance for implementation. For example: "Hashing the email address or phone number means that you've effectively put that data "beyond use". So long as you delete all the other data relating to it, it does not represent "personal data" in the GDPR sense." [60231117] Some recommended specific hashing algorithms, for example: "Passwords should be secured (hopefully with bcrypt [hyperlinked]) because if Alice has used the same password on Bob's Things as she has on Gmail then any attacker gaining access to the database on Bob's . . ." [38770303]

4.3.3 Avoid Collecting Data (advice=25/137, misconceptions=7/19)

Answerers recommended removing parts of data to preserve users' anonymity, not storing or collecting data to avoid further complications, and setting the correct configurations to avoid data collection in the first place.

Under this theme, we saw seven corrected misconceptions around what data is collected and how to handle data collection. For example, the original poster suspected a library would always send the data to third-party servers while the library actually allows the developer to deploy the backend on their own machines; the original poster mistakenly believed that there was a comprehensive taxonomy to help define what is personally identifiable information from privacy laws; and the original poster mistakenly believed that they needed to keep the template language provided by a privacy policy generator which claimed more than what they actually collected to be *"on the safe side legally."*

Strip personal information or use fake data (15/137)
These answers provided options, code samples, and advice to minimize data collection by removing parts of data, masking, setting constant values, anonymizing, proxies, and tools to block tracking. Note that these answers both discussed how to achieve this goal for first-party data practices and third-party data practices. For example, an answer offered detailed suggestions for how to handle the storage of IP addresses of a visited website for legal compliance:

> Strictly speaking, your web logs may contain personal data in the form of IP addresses and user agent strings. That data can be reasonably kept for a short period, say 10-30 days, for the purposes of combating abuse, but after that you should either truncate logs or strip out data that can be associated with any individual. [52497207]

The suggestions for third-party data practices were diverse and related to various third-party services. The challenge was that developers both needed to know what data might be collected by third-party services and how to prevent the data from being collected. For example,

this answer shows a nested feature that strips personal data before data is sent to Google Analytics:

> You can use the Fields to Set option (Variables -> Google Analytics Settings -> More Settings -> Fields to Set) to set the location and other Google Analytics parameters. You will need to create a variable (eg sanitisedLocation as Custom JavaScript to return the value, and use that for your field. [50246739]

Avoid having or storing data (6/137)

Some solutions suggest not collecting at all to avoid further processing and issues with storing data:

> There is one very simple way of avoiding all the negative consequences of third-party cookies: don't have any. It's possible to do a great many things without them, it means you may not need to display cookie notifications or seek consent. [56055698]

Use the correct configuration to avoid data collection (4/137)

The difference between this sub-theme and the previous sub-theme is that the previous one looks at reducing active data collection, while this sub-theme looks at avoiding unexpected data collection. For example, the default option of some data logging system does not guarantee that minimum data is collected, so developers need to actively change configurations to restrict data collection. For example, this answer suggested enabling the `Secure Outputs` option of Azure to avoid logging sensitive information like passwords in a clear format:

> ... Click the button in the upper right corner of "Get secret" action, click "Settings". Enable "Secure Outputs". After that, you can use the password value in your next actions and we can't see the password value in the run history. [65338257]

4.3.4 Read About Third-Party Services (advice=15/137, misconceptions=2/19)

Similar to the advice for checking data practices of third-parties (Section 4.3.1) but different in where and when the advice is applicable, this theme looks at various issues that may come out of using libraries, services, browser extensions, and tools that are built by others. The advice to tackle these issues are often around reading about and understanding the extra required permissions, using an alternative privacy-friendly service, or completely removing the third-party service. Two corrected misconceptions here were about where the data goes when using ad networks and browser extensions. In the following answer, the answerer corrects the asker's initial thought that Google and Matomo both transfer the data to their servers:

> Matomo is a different matter, because it's usually self-hosted and so is not sending data to anyone but yourself. That said, it usually does so via a javascript tracker plugin, and may set cookies. However, it will also work purely with log analytics which require neither of those things. [64995157]

Analytics code may need extra permissions (7/137)

While analytics tools provide insights into how users interact with developers' apps and services; they may also collect unnecessary data from users, which may require asking users for extra permissions that are not part of the developer's main app's permissions list:

> GA [Google Analytics] & GTM [Google Tag Manager] are extremely difficult to make GDPR compliant. You should not even load the scripts before getting consent. EU courts have already ruled that analytics does not constitute an "required" service, and thus does require consent, with all the baggage that goes with that. [57716738]

Don't use analytics or use a privacy-friendly alternative (6/137)

Another solution to third-party code was to completely remove the analytics code or use a more privacy-friendly option such as Matomo (i.e., an open-source alternative to Google Analytics that claims to be GDPR and CCPA compliant [37]):

> [provides a code snippet to find code that uses location data] Once you find the offending lib, you can try to figure out what purpose location data has and then decide whether you can get rid of it . . . [56779282]

Check ad networks' policies and permissions (2/137)

Two of the answers specifically addressed issues with ad networks and how they deal with permissions and location of the user: "I'm pretty sure that you're using some ad network or dependency that request such permissions. As an example several ad networks relies on `READ_PHONE_STATE` permission which could also trigger such notice from Google." [42203563]

4.3.5 Miscellaneous (advice=5/137, misconceptions=1/19)

A few answers were still worthy of mentioning but did not fit into other themes: using custom solutions and no action is needed.

Use custom solutions (3/137)
These answers suggested sample code or directions to build custom solutions for blocking tracking content in the emails and websites and building a custom view that does not require access to specific resources, and consequently, does not require the user's permission.

No action is needed (2/137)
When either the action is managed automatically, or no data is transferred to the servers, therefore, no action is required by the asker: ". . . the package will attempt to connect to the Internet only to download stuff in case something is not present or up to date. No data is uploaded anywhere." [66300609] One misconception was that the original poster thought when programming iOS apps they needed to manually display a symbol when using the camera to alert users about it, while the system would generate a visual indicator automatically if the camera is in use.

5 Discussion and Future Work

We qualitatively analyzed 119 Stack Overflow's privacy-related accepted answers. We extracted 148 pieces of advice from these posts, 21 corrected misconceptions, their information sources, and how they are related to Hoepman's privacy design strategies.

5.1 Privacy Frameworks

In our analysis, we observed that the eight privacy design strategies were mentioned at different levels of frequency. The observation has several likely causes. First, it could be that developers have varying levels of awareness and usage of the strategies, leading them to recommend the more familiar solutions. It could also be that problems that best associate with these solutions are more confusing or simply come up more frequently, leading to more questions in these areas. The more commonly advised strategies may also be simpler to implement and opperationalize. Stack Overflow rewards answerers who can provide clear explanations with examples, so there is some bias towards solutions that can be expressed that way. Irregardless of the reason, answerers clearly recommend some strategies more often than others which quite likely is also impacting the choices made by others who read these questions and answers when trying to solve their own problems.

In the following, we summarize the strategies into three groups based on how frequently they were mentioned in Stack Overflow privacy answers, and then speculate the causes of the unbalanced mentioning of different strategies, discuss their implications on developers and end-users, and discuss future directions to promote the underused strategies.

5.1.1 Most Frequent: `Inform` and `Control`

We find that most of the advice is around compliance with regulations and requirements imposed by software development platforms, which often relate to `inform` and `control` privacy design strategies, likely caused by the emphasis regulations like GDPR and CCPA put on informed consent. Regulators tend to put more pressure on big players like Apple and Google to ensure compliance on their platforms which leads in turn to such platforms creating requirements for developers to adhere to. Our finding highlight the impact platforms have on the types of questions developers ask as well as the answers given since approaches like `inform` are mandated to be used (i.e., privacy policies) while others like `separate` are not directly required. Such findings are also consistent with prior work about privacy discussion on developer forums [33, 57, 59] which also observed that how platforms present requirements impacts what developers discuss. Interestingly though, researchers interviewing developers about privacy conceptualizations observed that privacy approaches like "notice" are not mentioned by developers [25]. Its possible the difference is caused by developers' natural focus on technical solutions which are more inclined towards approaches like encryption leading them to focus on these areas in interviews. But the involvement of platforms in the development process forces them to engage with more legal and human focused approaches which they are less familiar with leading to questions.

Noting that `inform` is more common than `control` may also signal that users are not receiving as much control and are only informed about privacy practices

(e.g., requested permissions) more than having the controls and options to accept or reject a privacy-related feature. Although usable privacy researchers have proposed and evaluated various designs of usable and effective privacy notices and controls [20, 51], only the formats required by laws and platforms such as privacy policies and permissions were mentioned in Stack Overflow answers, which may suggest that the academic usable privacy work may still require effort to make it usable to developers. Future work may want to promote these designs through building an open-source, easy-to-use and integrate, and customizable consent pop up (or in general, notifications with controls).

5.1.2 Less Frequent: `Hide` and `Minimize`

The second and third most common type of advice is to ensure confidentiality, focusing on the `hide` strategy and avoiding data collection, which is directly related to the `minimize` strategy. As both of these approaches are easy to understand and reasonably easy to implement we were somewhat surprised that they were less common, though it is likely caused by platforms only indirectly encouraging developers to consider these approaches. If data is not collected then consent is not needed and privacy policies are not required. Some answerers recommended these strategies as a way of avoiding complex issues legal issues.

Prior work with developer interviews found that confidentiality came up several times, but minimization did not [25]; one potential reason might be the impact of study methodology. Interviews tend to be retrospective and over-sample for memorable events and general attitudes. Our approach instead focuses on situations where developers encountered problems even if those problems were not memorable. The combination of the findings suggests that developers may be applying these strategies reactively either to solve other problems (e.g., unwanted consent dialogues) or as a natural part of system design when they have to engage in activities like database structuring.

5.1.3 Rarely Mentioned: `Abstract`, `Separate`, `Enforce`, and `Demonstrate`

We found the other privacy design strategies, including `abstract`, `separate`, `enforce`, and `demonstrate` to be rarely advised in Stack Overflow's answers. The rare mention of `enforce` and `demonstrate` is unsurprising since they are organizational-level strategies which might be overlooked by developers who work at technical levels. Their absence is a bit concerning though since some questions highlight the effort some developers put into protecting user privacy but with little focus on demonstrating this to users.

Conversely, the `abstract` and `separate` are more technical yet still rarely recommended in Stack Overflow answers. The `abstract` strategy can provide anonymity to users, for example, by using k-anonymity or coarse data instead of precise data, and `separate` suggests the distribution of storing and processing data [28]. Some potential explanations may be (1) tools that can provide these techniques are not as readily available, (2) developers are not yet aware of them, (3) Stack Overflow is not the right place to find information about these techniques, (4) askers do not tag and associate these techniques with privacy, (5) clients do not ask for them, and (6) software platforms and operating systems do not yet offer these techniques as their core services. These findings also emphasize the value in making privacy enhancing technologies accessible and usable by the users of them (developers from our study's viewpoint). Such findings also echo findings of Agrawal et al. [2] where they study two privacy enhancing technologies, secure multi-party computation (`separate` strategy) and differential privacy (`abstract` strategy). They find that these strategies are not yet usable by developers because of the gaps between theory and practice. One future direction is to look at tools and coverage of these techniques in the software development ecosystem to find obstacles and barriers to developers' adoption.

5.2 Where to Find Privacy Advice?

When looking at the information sources and provided links in the answers, online articles are second after official documentation. While there is literature that looks at the impacts of Stack Overflow's security posts on software security [21], future direction may look at the content of resources available to developers beyond official documentation and Stack Overflow such as websites, tutorials, and blogs, that provide privacy advice.

We noticed a type of post (8/119) that does not try to apply a privacy design strategy but tries to know and understand what is going on with a product, service, or regulation, which is similar to "abstract/conceptual" question type that Tahaei et al. found in privacy-related questions [59]. Example questions included: where data goes if the developer uses a particular API or library,

is this data a piece of personal information or sensitive data from GDPR's point of view, and can third-parties set cookies on websites. The tone in these posts is investigative and curious. The accepted answers in line with the this type of questions are explanatory and descriptive (e.g., explaining how a technology works) rather than providing a how-to dos. Combined with information sources, it appears that Stack Overflow may partially shape how developers think about and conceptualize privacy. Therefore, looking more into these answers' validity and providing solutions to improve answers can be a valuable future research avenue.

Two posts about privacy policies were closed (although they have an accepted answer and are still accessible online) because the community considered them off-topic. While we did not formally analyze the comments, some comments for these posts were particularly interesting: "I'm voting to close this question as off-topic because this is a legal question, not a programming question" and "I'm voting to close this question as off-topic because a privacy policy in itself is not programming related." While several other questions around privacy policies were answered and not closed, parts of Stack Overflow's community appear to not welcome questions related to the regulatory aspects of programming even though this topic is one of the major pain points for developers who ask questions about privacy topics on Stack Overflow [59].

Such reactions opens up another research direction to look at what modern programming means, what skills are required to publish an app on Google Play, for example, and how developers can be trained and supported in these tasks. Knowing about the classic abilities, such as maintainability, dependability, efficiency, and functionality, may not be sufficient for a computer science graduate to develop software in today's software ecosystem. Academic education may be an option to teach privacy topics to a portion of future software developers; suggested methods include talking about online personal information and consequences of data sharing [18], or creating games to make students aware of sensitive decisions [55]. We suggest incorporating privacy values into security courses too. As suggested in prior work, defining what could happen if secure programming measures are not taken into consideration may help students understand the value of using secure approaches [5]. We recommend combining security vulnerability consequences with privacy consequences for the users and the society to teach students about the larger consequences of their choices instead of immediate functional requirements, as having people with the right mindset for privacy may be more productive than having a hard set of guidelines for implementing Privacy by Design [29].

5.2.1 Community Privacy Champions

Some privacy answerers in Stack Overflow provided multiple answers, and we view them as informal privacy champions in the online communities that spend their time for free to educate and inform others about privacy. Privacy champions in software teams motivate others and promote privacy values in their teams by having informal conversations about privacy and running tailored workshops around privacy [60]. The software ecosystem can benefit from community privacy champions to educate and promote developers through peer discussion on online developer forums. A future direction may involve conducting interviews with these informal online privacy champions to find out their motivations, information sources, how to best leverage their knowledge, and also how to best support them.

5.3 Third-Party Data Practices

We noticed that much of the advice in Stack Overflow was about how to understand and control third-party data practices (Sections 4.3.1, 4.3.3 and 4.3.4). This is not surprising as including third-party code is one of main causes of GDPR violation in Android apps [42]. While we expected to see more privacy concerns about ad networks and their privacy practices, we observed another entity, analytics services, to be more concerning from developers' perspective. Future research may look at analytics' privacy interfaces to understand how they present privacy information to developers and, more broadly, how privacy controls can and should be presented to developers to assist them in making informed decisions for their apps and users' privacy. Recent research shows that ad networks' privacy interfaces may include dark patterns to nudge developers into making privacy-unfriendly decisions [58, 61]. Future research may look at similar patterns in other software development platforms such as analytics services.

While developers do not often discuss privacy issues about ad networks, empirical analysis of apps shows that ad networks (e.g., Facebook and Unity) are one of the primary collectors of personal data from users through Android apps [42]. We hypothesize that developers may not see ad networks' privacy practices as concerning or, as prior works shows [40], they may not see

themselves responsible for privacy practices of ad networks. Either way, future research may look at privacy interfaces directed to developers and understand how privacy information in these platforms are presented to developers, whether developers are able to understand the privacy consequences of their choices for their users' privacy, and in general, how platforms support developers in being compliant with privacy regulations.

Finally, our empirical observations also give us an opportunity to reflect on the coverage of the privacy frameworks for achieving Privacy by Design and the research on privacy enhancing technologies. Although the existing frameworks and privacy enhancing technologies mostly focus on addressing privacy issues related to first-party data use, we observed that currently developers have to spend a lot of time and effort digging into specific third-party services and tools to understand and control third-party data practices. Our findings suggest that there have yet to be practical tools to support developers during this process. The main resource developers have available is the terms of service and privacy policies of these services (Section 4.3.1), which are often lengthy, vague, and full of legalese [39]. Presentation of privacy information and how data is controlled varies across platforms (Section 4.3.3), which suggests developers have to use ad-hoc approaches to solving the problem and can not transfer the knowledge learned from using one platforms to another. Given these challenges, we argue that handling third-party data practices should be emphasized more in privacy frameworks and practical developer tools are needed to provide sufficient support.

6 Conclusion

We qualitatively analyzed 119 privacy-related accepted answers on Stack Overflow and extracted 148 pieces of advice that developers give one another to accomplish privacy-related tasks and provide privacy to their users. We find that, developers most commonly provide answers that recommend using `inform`, `hide`, `control`, and `minimize` strategies while other strategies such as `abstract` and `separate` are rarely suggested. Similar to prior work on Stack Overflow questions [59], we find that the requirements that platforms like Google Play enforce impact the types of questions, and consequently answers, observed on Stack Overflow leading to strategies like `inform` and `control` being common in accepted answers. Future research may look at ways to improve the privacy ecosystem and empower developers by thinking about the usability of the less common privacy framework strategies as well as the usability of approaches like differential privacy which have promise but are currently challenging for developers to use.

Acknowledgments

We thank the reviewers whose constructive feedback helped improve the paper greatly. This work was sponsored in part by Microsoft Research through its Ph.D. Scholarship Program. Tianshi Li was supported in part by the CMU CyLab Presidential Fellowship.

References

[1] Yasemin Acar, Michael Backes, Sascha Fahl, Doowon Kim, Michelle L Mazurek, and Christian Stransky. You Get Where You're Looking for: The Impact of Information Sources on Code Security. In *2016 IEEE Symposium on Security and Privacy (SP)*, pages 289–305. IEEE, May 2016. 10.1109/SP.2016.25.

[2] Nitin Agrawal, Reuben Binns, Max Van Kleek, Kim Laine, and Nigel Shadbolt. Exploring Design and Governance Challenges in the Development of Privacy-Preserving Computation. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. ACM. 10.1145/3411764.3445677.

[3] Sami Alkhatib, Jenny Waycott, George Buchanan, Marthie Grobler, and Shuo Wang. Privacy by Design in Aged Care Monitoring Devices? Well, Not Quite Yet! In *32nd Australian Conference on Human-Computer Interaction*, OzCHI '20, page 492–505, New York, NY, USA, 2020. ACM. 10.1145/3441000.3441049.

[4] Miltiadis Allamanis and Charles Sutton. Why, when, and what: Analyzing Stack Overflow questions by topic, type, and code. In *2013 10th Working Conference on Mining Software Repositories (MSR)*, pages 53–56. IEEE, May 2013. 10.1109/MSR.2013.6624004.

[5] Majed Almansoori, Jessica Lam, Elias Fang, Kieran Mulligan, Adalbert Gerald Soosai Raj, and Rahul Chatterjee. How Secure Are Our Computer Systems Courses? In *Proceedings of the 2020 ACM Conference on International Computing Education Research*, ICER '20, page 271–281, New York, NY, USA, 2020. ACM. 10.1145/3372782.3406266.

[6] Jeff Atwood. Attribution Required, 2009. URL https://stackoverflow.blog/2009/06/25/attribution-required/.

[7] Jeff Atwood. Stack Overflow Creative Commons Data Dump, 2009. URL https://stackoverflow.blog/2009/06/04/stack-overflow-creative-commons-data-dump/.

[8] Jeff Atwood. Academic Papers Using Stack Overflow Data, 2010. URL https://stackoverflow.blog/2010/05/31/academic-papers-using-stack-overflow-data/.

[9] Anton Barua, Stephen W Thomas, and Ahmed E Hassan. What are developers talking about? An analysis of topics

and trends in Stack Overflow. *Empirical Software Engineering*, 19(3):619–654, 2014. 10.1007/s10664-012-9231-y.

[10] Maia J. Boyd, Jamar L. Sullivan Jr., Marshini Chetty, and Blase Ur. Understanding the Security and Privacy Advice Given to Black Lives Matter Protesters. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. ACM. 10.1145/3411764.3445061.

[11] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3(2): 77–101, 2006. 10.1191/1478088706qp063oa.

[12] Julio C. Caiza, Yod-Samuel Martín, Danny S. Guamán, Jose M. Del Alamo, and Juan C. Yelmo. Reusable Elements for the Systematic Design of Privacy-Friendly Information Systems: A Mapping Study. *IEEE Access*, 7:66512–66535, 2019. 10.1109/ACCESS.2019.2918003.

[13] Ann Cavoukian. Privacy by Design: The 7 Foundational Principles. *Information and privacy commissioner of Ontario, Canada*, 5, 2009. URL https://iab.org/wp-content/IAB-uploads/2011/03/fred_carter.pdf.

[14] Ann Cavoukian, Scott Taylor, and Martin E. Abrams. Privacy by Design: essential for organizational accountability and strong business practices. *Identity in the Information Society*, 3(2):405–413, August 2010. 10.1007/s12394-010-0053-z.

[15] Mina Deng, Kim Wuyts, Riccardo Scandariato, Bart Preneel, and Wouter Joosen. A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements. *Requirements Engineering*, 16(1):3–32, 2011. 10.1007/s00766-010-0115-7.

[16] Nick Doty and Mohit Gupta. Privacy Design Patterns and Anti-Patterns, 2013. URL https://cups.cs.cmu.edu/soups/2013/trustbusters2013/Privacy_Design_Patterns-Antipatterns_Doty.pdf.

[17] Cynthia Dwork. Differential privacy: A survey of results. In *International conference on theory and applications of models of computation*, pages 1–19. Springer, 2008. 10.1007/978-3-540-79228-4_1.

[18] Serge Egelman, Julia Bernd, Gerald Friedland, and Dan Garcia. The Teaching Privacy Curriculum. In *Proceedings of the 47th ACM Technical Symposium on Computing Science Education*, SIGCSE '16, page 591–596, New York, NY, USA, 2016. ACM. 10.1145/2839509.2844619.

[19] Pardis Emami-Naeini, Yuvraj Agarwal, Lorrie Faith Cranor, and Hanan Hibshi. Ask the Experts: What Should Be on an IoT Privacy and Security Label? In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 447–464. IEEE, 2020. 10.1109/SP40000.2020.00043.

[20] Yuanyuan Feng, Yaxing Yao, and Norman Sadeh. A Design Space for Privacy Choices: Towards Meaningful Privacy Control in the Internet of Things. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, New York, NY, USA, 2021. ACM. 10.1145/3411764.3445148.

[21] Felix Fischer, Konstantin Böttinger, Huang Xiao, Christian Stransky, Yasemin Acar, Michael Backes, and Sascha Fahl. Stack Overflow Considered Harmful? The Impact of Copy Paste on Android Application Security. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 121–136. IEEE, May 2017. 10.1109/SP.2017.31.

[22] Imane Fouad, Cristiana Santos, Feras Al Kassar, Nataliia Bielova, and Stefano Calzavara. On Compliance of Cookie Purposes with the Purpose Specification Principle. In *IWPE 2020 - International Workshop on Privacy Engineering*, pages 1–8, Genova, Italy, September 2020. Inria. URL https://hal.inria.fr/hal-02567022.

[23] Daniel Greene and Katie Shilton. Platform privacies: Governance, collaboration, and the different meanings of "privacy" in iOS and Android development. *New Media & Society*, 20 (4):1640–1657, 2018. 10.1177/1461444817702397.

[24] Kilem Li Gwet. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48, 2008. 10.1348/000711006X126600.

[25] Irit Hadar, Tomer Hasson, Oshrat Ayalon, Eran Toch, Michael Birnhack, Sofia Sherman, and Arod Balissa. Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering*, 23(1):259–289, February 2018. 10.1007/s10664-017-9517-1.

[26] Thomas Heyman, Koen Yskout, Riccardo Scandariato, and Wouter Joosen. An analysis of the security patterns landscape. In *Third International Workshop on Software Engineering for Secure Systems (SESS'07: ICSE Workshops 2007)*, pages 3–3. IEEE, 2007. 10.1109/SESS.2007.4.

[27] Jaap-Henk Hoepman. Privacy Design Strategies. In Nora Cuppens-Boulahia, Frédéric Cuppens, Sushil Jajodia, Anas Abou El Kalam, and Thierry Sans, editors, *ICT Systems Security and Privacy Protection*, pages 446–459, Berlin, Heidelberg, 2014. Springer Berlin Heidelberg. 978-3-642-55415-5_38.

[28] Jaap-Henk Hoepman. *Privacy Design Strategies (The Little Blue Book)*. Radboud University, 2019. URL https://cs.ru.nl/~jhh/publications/pds-booklet.pdf.

[29] Bert-Jaap Koops and Ronald Leenes. Privacy regulation cannot be hardcoded. a critical comment on the 'privacy by design' provision in data-protection law. *International Review of Law, Computers & Technology*, 28(2):159–171, 2014. 10.1080/13600869.2013.801589.

[30] Blagovesta Kostova, Seda Gürses, and Carmela Troncoso. Privacy Engineering Meets Software Engineering. On the Challenges of Engineering Privacy By Design, 2020. URL https://arxiv.org/abs/2007.08613.

[31] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. Chapter 8 - Interviews and focus groups. In Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser, editors, *Research Methods in Human Computer Interaction*, pages 187–228. Morgan Kaufmann, Boston, second edition edition, 2017. 10.1016/B978-0-12-805390-4.00008-X.

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

[33] Tianshi Li, Elizabeth Louie, Laura Dabbish, and Jason I. Hong. How Developers Talk About Personal Data and What It Means for User Privacy: A Case Study of a Developer Forum on Reddit. *Proc. ACM Hum.-Comput. Interact.*, 4 (CSCW3), January 2021. 10.1145/3432919.

[34] Tianshi Li, Elijah B. Neundorfer, Yuvraj Agarwal, and Jason I. Hong. Honeysuckle: Annotation-guided code gen-

eration of in-app privacy notices. *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, 5(3), September 2021. 10.1145/3478097.

[35] Ilaria Liccardi, Monica Bulger, Hal Abelson, Daniel Weitzner, and Wendy Mackay. Can apps play by the COPPA Rules? In *2014 Twelfth Annual International Conference on Privacy, Security and Trust*, pages 1–9. IEEE, 2014. 10.1109/PST.2014.6890917.

[36] Jialiu Lin, Shahriyar Amini, Jason I. Hong, Norman Sadeh, Janne Lindqvist, and Joy Zhang. 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, page 501–510, New York, NY, USA, 2012. ACM. 10.1145/2370216.2370290.

[37] Matomo. Google Analytics alternative that protects your data, 2021. URL https://matomo.org.

[38] Celestin Matte, Nataliia Bielova, and Cristiana Santos. Do Cookie Banners Respect my Choice? : Measuring Legal Compliance of Banners from IAB Europe's Transparency and Consent Framework. In *2020 IEEE Symposium on Security and Privacy (SP)*, pages 791–809. IEEE, 05 2020. 10.1109/SP40000.2020.00076.

[39] Aleecia M McDonald and Lorrie Faith Cranor. The Cost of Reading Privacy Policies. *I/S: A Journal of Law and Policy for the Information Society (ISJLP)*, 4:543, 2008. URL https://heinonline.org/HOL/P?h=hein.journals/isjlpsoc4&i=563.

[40] Abraham H. Mhaidli, Yixin Zou, and Florian Schaub. "We Can't Live Without Them!" App Developers'Adoption of Ad Networks and Their Considerations of Consumer Risks. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*, Santa Clara, CA, August 2019. USENIX Association. URL https://www.usenix.org/conference/soups2019/presentation/mhaidli.

[41] Michael Naehrig, Kristin Lauter, and Vinod Vaikuntanathan. Can homomorphic encryption be practical? In *Proceedings of the 3rd ACM Workshop on Cloud Computing Security Workshop*, CCSW '11, page 113–124, New York, NY, USA, 2011. ACM. 10.1145/2046660.2046682.

[42] Trung Tin Nguyen, Michael Backes, Ninja Marnau, and Ben Stock. Share first, ask later (or never?) studying violations of gdpr's explicit consent in android apps. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 3667–3684. USENIX Association, August 2021. URL https://www.usenix.org/conference/usenixsecurity21/presentation/nguyen.

[43] State of California Department of Justice. California Consumer Privacy Act (CCPA), 2018. URL https://oag.ca.gov/privacy/ccpa.

[44] Information Commissioner's Office. Data protection impact assessments, 2021. URL https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/accountability-and-governance/data-protection-impact-assessments/.

[45] Stack Overflow. What should I do when someone answers my question?, 2021. URL https://stackoverflow.com/help/someone-answers.

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

[47] Chris Parnin, Christoph Treude, Lars Grammel, and Margaret-Anne Storey. Crowd documentation: Exploring the coverage and the dynamics of API discussions on Stack Overflow. *Georgia Institute of Technology, Tech. Rep*, 11, 2012. URL http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.371.6263.

[48] Elissa M. Redmiles, Noel Warford, Amritha Jayanti, Aravind Koneru, Sean Kross, Miraida Morales, Rock Stevens, and Michelle L. Mazurek. A Comprehensive Quality Evaluation of Security and Privacy Advice on the Web. In *29th USENIX Security Symposium (USENIX Security 20)*, pages 89–108. USENIX Association, August 2020. URL https://www.usenix.org/conference/usenixsecurity20/presentation/redmiles.

[49] Irwin Reyes, Primal Wijesekera, Joel Reardon, Amit Elazari Bar On, Abbas Razaghpanah, Narseo Vallina-Rodriguez, and Serge Egelman. "Won't Somebody Think of the Children?" Examining COPPA Compliance at Scale. *Proceedings on Privacy Enhancing Technologies*, 2018(3): 63–83, 2018. 10.1515/popets-2018-0021.

[50] Neil Salkind. *Encyclopedia of Research Design*. SAGE Publications, Inc, June 2020. 10.4135/9781412961288.

[51] Florian Schaub, Rebecca Balebako, Adam L. Durity, and Lorrie Faith Cranor. A Design Space for Effective Privacy Notices. In *Proceedings of the Eleventh USENIX Conference on Usable Privacy and Security*, SOUPS '15, page 1–17, USA, 2015. USENIX Association. URL https://www.usenix.org/system/files/conference/soups2015/soups15-paper-schaub.pdf.

[52] Awanthika Senarath and Nalin A. G. Arachchilage. Why Developers Cannot Embed Privacy into Software Systems?: An Empirical Investigation. In *Proceedings of the 22Nd International Conference on Evaluation and Assessment in Software Engineering 2018*, EASE'18, pages 211–216, New York, NY, USA, 2018. ACM. 10.1145/3210459.3210484.

[53] Swapneel Sheth, Gail Kaiser, and Walid Maalej. Us and Them: A Study of Privacy Requirements Across North America, Asia, and Europe. In *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, pages 859–870, New York, NY, USA, 2014. ACM. 10.1145/2568225.2568244.

[54] Katie Shilton and Daniel Greene. Linking Platforms, Practices, and Developer Ethics: Levers for Privacy Discourse in Mobile Application Development. *Journal of Business Ethics*, 155(1):131–146, March 2019. 10.1007/s10551-017-3504-8.

[55] Katie Shilton, Donal Heidenblad, Adam Porter, Susan Winter, and Mary Kendig. Role-Playing Computer Ethics: Designing and Evaluating the Privacy by Design (PbD) Simulation. *Science and Engineering Ethics*, PP(PP), July 2020. 10.1007/s11948-020-00250-0.

[56] Laura Shipp and Jorge Blasco. How private is your period?: A systematic analysis of menstrual app privacy policies. *Proceedings on Privacy Enhancing Technologies*, 2020(4): 491–510, October 2020. 10.2478/popets-2020-0083.

[57] Mohammad Tahaei and Kami Vaniea. A Survey on Developer-Centred Security. In *2019 IEEE European Symposium on Security and Privacy Workshops (Eu-*

roS&PW), pages 129–138. IEEE, June 2019. 10.1109/EuroSPW.2019.00021.

[58] Mohammad Tahaei and Kami Vaniea. "Developers Are Responsible": What Ad Networks Tell Developers About Privacy. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems Extended Abstracts*, CHI '21 Extended Abstracts, pages 1–12, New York, NY, USA, 2021. ACM. 10.1145/3411763.3451805.

[59] Mohammad Tahaei, Kami Vaniea, and Naomi Saphra. Understanding Privacy-Related Questions on Stack Overflow. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, page 1–14. ACM, 2020. 10.1145/3313831.3376768.

[60] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. Privacy Champions in Software Teams: Understanding Their Motivations, Strategies, and Challenges. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, CHI '21, pages 1–15. ACM, 2021. 10.1145/3411764.3445768.

[61] Mohammad Tahaei, Alisa Frik, and Kami Vaniea. Deciding on Personalized Ads: Nudging Developers About User Privacy. In *Seventeenth Symposium on Usable Privacy and Security (SOUPS 2021)*, pages 573–596. USENIX Association, August 2021. URL https://www.usenix.org/conference/soups2021/presentation/tahaei.

[62] Christine Utz, Martin Degeling, Sascha Fahl, Florian Schaub, and Thorsten Holz. (Un) Informed Consent: Studying GDPR Consent Notices in the Field. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, page 973–990. ACM, 2019. 10.1145/3319535.3354212.

[63] Daniel Votipka, Mary Nicole Punzalan, Seth M Rabin, Yla Tausczik, and Michelle L Mazurek. An Investigation of Online Reverse Engineering Community Discussions in the Context of Ghidra. In *IEEE European Symposium on Security and Privacy (EuroS&P)*. IEEE, 2021.

[64] Richmond Y. Wong and Deirdre K. Mulligan. Bringing Design to the Privacy Table: Broadening "Design" in "Privacy by Design" Through the Lens of HCI. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 262:1–262:17. ACM, 2019. 10.1145/3290605.3300492.

[65] Nahathai Wongpakaran, Tinakon Wongpakaran, Danny Wedding, and Kilem L. Gwet. A comparison of Cohen's Kappa and Gwet's AC1 when calculating inter-rater reliability coefficients: a study conducted with personality disorder samples. *BMC Medical Research Methodology*, 13(1):61, April 2013. 10.1186/1471-2288-13-61.

[66] Xin-Li Yang, David Lo, Xin Xia, Zhi-Yuan Wan, and Jian-Ling Sun. What Security Questions Do Developers Ask? A Large-Scale Study of Stack Overflow Posts. *Journal of Computer Science and Technology*, 31(5):910–924, September 2016. 10.1007/s11390-016-1672-0.

[67] Slavica Zec, Nicola Soriani, Rosanna Comoretto, and Ileana Baldi. High Agreement and High Prevalence: The Paradox of Cohen's Kappa. *The open nursing journal*, 11:211–218, October 2017. 10.2174/1874434601711010211.

[68] Eric Zeng and Franziska Roesner. Understanding and improving security and privacy in multi-user smart homes: a design exploration and in-home user study. In *28th USENIX Security Symposium (USENIX Security 19)*, pages 159–176, 2019.

[69] Sebastian Zimmeck, Peter Story, Daniel Smullen, Abhilasha Ravichander, Ziqi Wang, Joel Reidenberg, N. Cameron Russell, and Norman Sadeh. MAPS: Scaling Privacy Compliance Analysis to a Million Apps. *Proceedings on Privacy Enhancing Technologies*, 2019(3):66–86, 2019. 10.2478/popets-2019-0037.