The Simpler, the Better? Presenting the COPING Android Permission-Granting Interface for Better Privacy-related Decisions

Paul Gerber [corresponding author]

Work and Engineering Psychology Research Group, Institute of Psychology, Technische Universitaet Darmstadt, Darmstadt, Germany

gerber@psychologie.tu-darmstadt.de

Melanie Volkamer

Professor for Usable Privacy and Security, Karlstad University, Karlstad, Sweden

melanie.volkamer@secuso.org

Karen Renaud

Usable Security & Privacy Lead, School of Computing Science, University of Glasgow, Glasgow, Scotland

karen.renaud@glasgow.ac.uk

Abstract

One of the great innovations of the modern world is the Smartphone app. The sheer multitude of available apps attests to their popularity and general ability to satisfy our wants and needs. The flip side of the functionality these apps offer is their potential for privacy invasion. Apps can, if granted permission, gather a vast amount of very personal and sensitive information. App developers might exploit the combination of human propensities and the design of the Android permission-granting interface to gain permission to access more information than they really need. This compromises personal privacy. The fact that the Android is the globally dominant phone means widespread privacy invasion is a real concern.

We, and other researchers, have proposed alternatives to the Android permission-granting interface. The aim of these alternatives is to highlight privacy considerations more effectively during app installation: to ensure that privacy becomes part of the decision-making process. We report, here, on a study with 344 participants that compared the impact of a number of permission-granting interface proposals, including our own (called the **COPING** interface — **CO**mprehensive **P**erm**I**ssio**N** **G**ranting) and two Android interfaces. To conduct the comparison we carried out an online study with a mixed-model design.

Our main finding is that the focus in these interfaces ought to be on improving the *quality* of the provided information rather than merely simplifying the interface. The intuitive approach is to reduce and simplify information, but we discovered that this actually impairs the quality of the decision. Our recommendation is that further investigation is

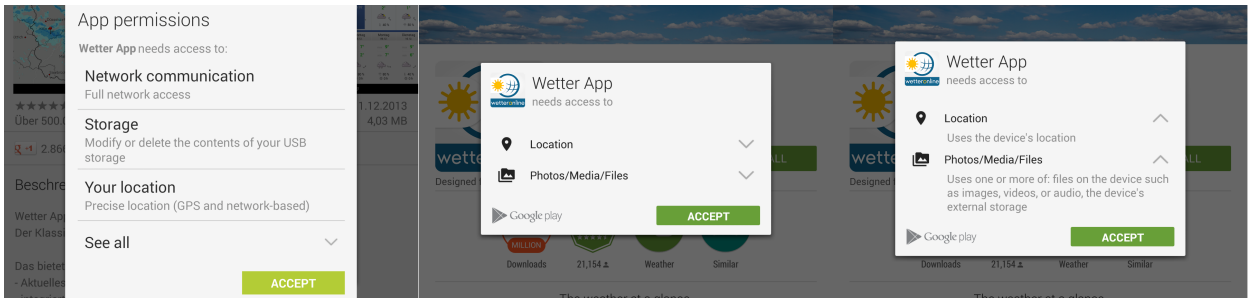

Figure 1: (a) On the left: depiction of permissions of the legacy version, i.e. Google Play Store until version 4.6.17 including (b) In the middle: corresponding interface of the recent version, i.e. since version 4.8.19 (c) on the right: corresponding interface of the recent version with revealed details.

required in order to find the "sweet spot" where understandability *and* comprehensiveness are maximised.

Keywords: Android Permission-Granting Interface; Heuristics; Interface comprehensiveness; Privacy-related behavior

1. Introduction

Smartphones, in addition to facilitating traditional communication, provide much more functionality than feature phones, and nearly a third of mankind owns one [1]. All of the provided functionality relies on, and produces, sensitive and personal data. This is often stored on the device, augmented by data that is collected by the phone itself, including usage data, location and biometrics.

The majority of all global Smartphones are Android (83% of all sold Smartphones in the second quarter of 2015 [2]). Access rights to data, sensors, and interfaces are granted by means of the Android permissions interface. Three such interfaces exist: (1) pre-2014, (2) June 2014-November 2015 (used on the majority of Androids [3]), and (3) post-November 2015. We will refer to these interfaces as (1) *legacy*, (2) *recent* and (3) *iPhone-similar*.

The legacy interface is depicted in Figure 1 (a). See Figure 1 (b) for an example screenshot of the *recent* interface. A number of issues make the legacy and recent Android permission granting process sub-optimal in terms of supporting privacy-aware decision making. One significant drawback is the on-off atomic process of permission-granting, essentially a choice between "Grant all" or "Abort installation". The fact that this decision has to be made post-installation is problematical since it requires post-commitment abandonment of a predefined course of action. These interfaces also tend to use overly technical jargon and complex wording [4].

Researchers have proposed a number of alternative permission-granting interfaces in order to inform users about possible privacy implications more effectively. Many of these were proven superior to the legacy Android interface [5, 6, 7, 8]. Since different study designs were used a comparison of all alternatives to each other is not possible.

To our knowledge no one has, thus far, compared and contrasted these interfaces to each other in order to determine which proposal maximises decision quality with respect to privacy-aware installation. Therefore, our contributions are:

- Proposing a more comprehensive alternative interface (see Figure 2) in order to more effectively inform users about possible privacy implications when installing the corresponding app, since possible interactions between permissions are shown.
- Comparing our alternative interface to other proposed alternatives [5, 6, 7] with varying comprehensiveness, as well as to the *recent* and *legacy* Android permission-granting interfaces in an online study with a mixed-model design.
- Drawing conclusions about Android permission interfaces and informing the development of privacy-setting interfaces in general.

Our results indicate that Smartphone owners are capable of reading, understanding and making an informed decision, even with very information-rich interfaces. The focus when designing new permission-granting alternatives, as well as privacy-setting interfaces, should be on improving the *quality* of the provided information (including benefits and consequences of granting access) rather than merely simplifying interfaces.

Section 2 provides an overview of the related literature and alternative interfaces provided therein. Section 3 describes the construction of our alternative interface we add to the comparison while Section 4 describes our study. Section 5 presents the study results and Section 6 reflects on our findings and suggests directions for future work. Section 7 concludes.

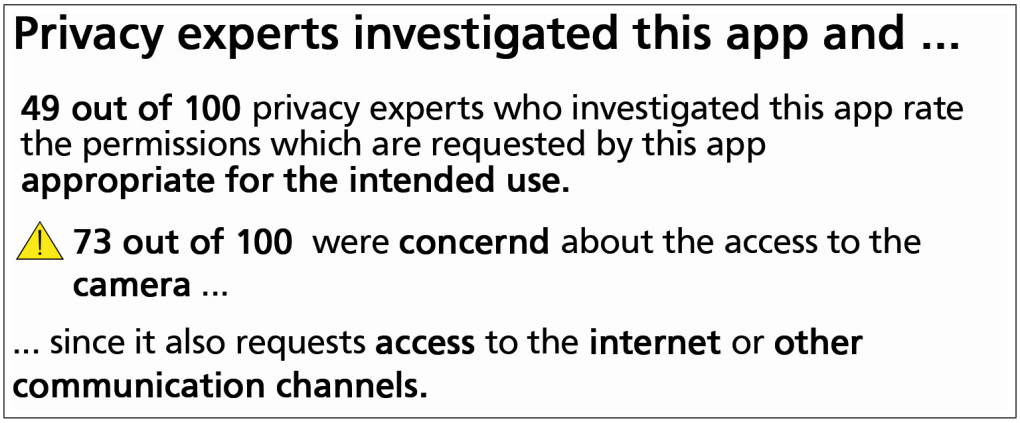

Figure 2: The COPING Interface

2. Background

First, we take a look at the different Android permission interfaces and then review proposed alternatives to the Android interface. We commenced our literature search by

combing through the CHI[1] and SOUPS[2] proceedings from the last few years. These are the top two conferences from Human Computer Interaction and Usable Security and Privacy respectively. We then conducted backward and forward reference searches. We targeted publications dealing with access permissions in the context of the app decision-making process with a particular focus on alternative interface proposals.

2.1. Android Permission-Granting Interface

In this subsection, we describe the legacy and recent Android interfaces[3]. Although not included in the comparison, we also describe and discuss the changes to the Android permission-granting interface which have recently resulted in the iPhone-similar interface (Android 6 Marshmallow).

Legacy Android Permission-Granting Interface.
Up until June 2014 Android displayed a vertical list of potentially-sensitive permissions being requested by an app (see Figure 1 (a)). To see a full list of all requested permissions the user had to touch the "see all" button beneath the list. Users could only install an app if they granted **all** requested permissions.

Recent Android permission-granting interface.
With the update to Play Store Version 4.8.19 in June 2014, all Android permissions were allocated to one of thirteen permission groups[4]. The relevant group names are displayed after the user clicks on the 'install' button. The name of each group is listed, accompanied by an explanatory pictogram (see Figure 1 (b) and (c)), if the app requests any, or all, permissions from that group. There is no visual difference in the permission interface between an app that requests *one* permission from a group and an app that requests *all* permissions in the group. While the legacy Android permission interface provides the 'see all' option, this is not the case for the recent interface. Note, the thirteenth permission group, 'Other', does not appear. At installation, users cannot see whether the app in question is requesting 'Other' permissions unless they check the detailed permission interface, which is only accessible from the PlayStore page.

Users can only install an app if they grant all requested permissions, or, in essence, entire groups of permissions. However, in contrast to the legacy interface, the user implicitly grants all permissions within the requested groups, not only single permissions. Hence, app updates do not need to re-request any permissions in the group, even if new ones are now required, as long as the user has granted permission for that group during installation (see Figure 3). For a detailed discussion of the changes Google made to the permission interface, and the privacy implications thereof, consult [9].

iPhone-similar Permission-Granting Interface.

[1] http://www.sigchi.org/conferences
[2] https://www.usenix.org/conference/soups2017
[3] The recent interface is also used as a fallback in Android 6 for apps programmed using the old SDK.
[4] https://support.google.com/googleplay/answer/6014972? hl=de; last retrieved 04/3/2016

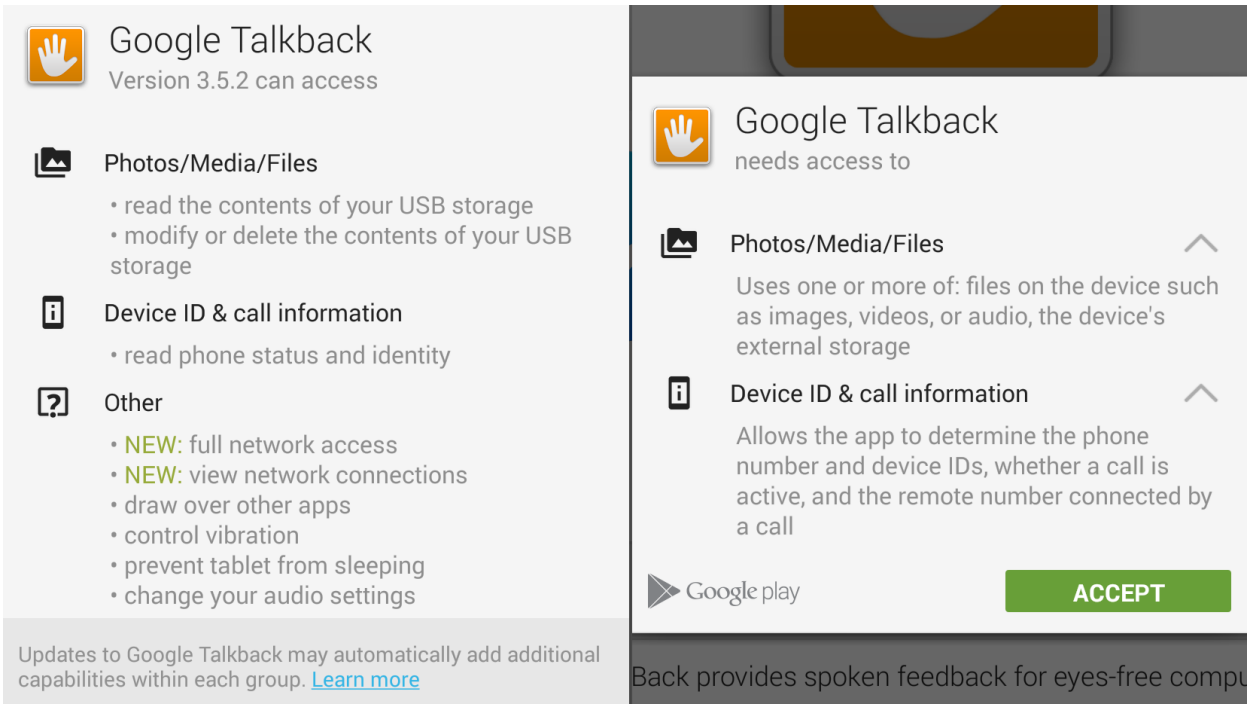

Figure 3: App update with newly requested permissions in the detailed and the regular permission interface

With the release of Android 6, Google changed the permission granting to a 'grant at runtime' model similar to the iPhone paradigm. All apps developed using the new SDK, and running on a device with Android 6.0 or newer, will no longer ask for permissions to be granted at installation. Instead, permission will be requested as and when required and the user can grant or deny access, or deny previously-granted access. In the corresponding dialog, only the permission that is needed at that moment is requested. If multiple permissions are requested, multiple dialogs appear sequentially, instead of being summarised in a single interface. Permissions that have previously been granted are not redisplayed. Permissions Google considers to be 'normal'[5] (such as changing the time zone) are granted by default.

2.2. Shortcomings of the Legacy and Recent Android Permission-Granting Interfaces

A number of researchers have been studied the Android permission-granting interfaces. The findings are:

2.2.1. Dismissing the Permission Interface

The permissions interface is likely to be treated as an annoying hurdle, something to be dismissed in order to achieve the goal of installing and using the app. Users have become habituated to security warnings of all kinds [10], and the permissions interface may be considered to be "yet another warning" to be dismissed. If this happens, this interface will

[5]http://developer.android.com/guide/topics/security/permissions.html#normal-dangerous - last retrieved on 4/3/2016

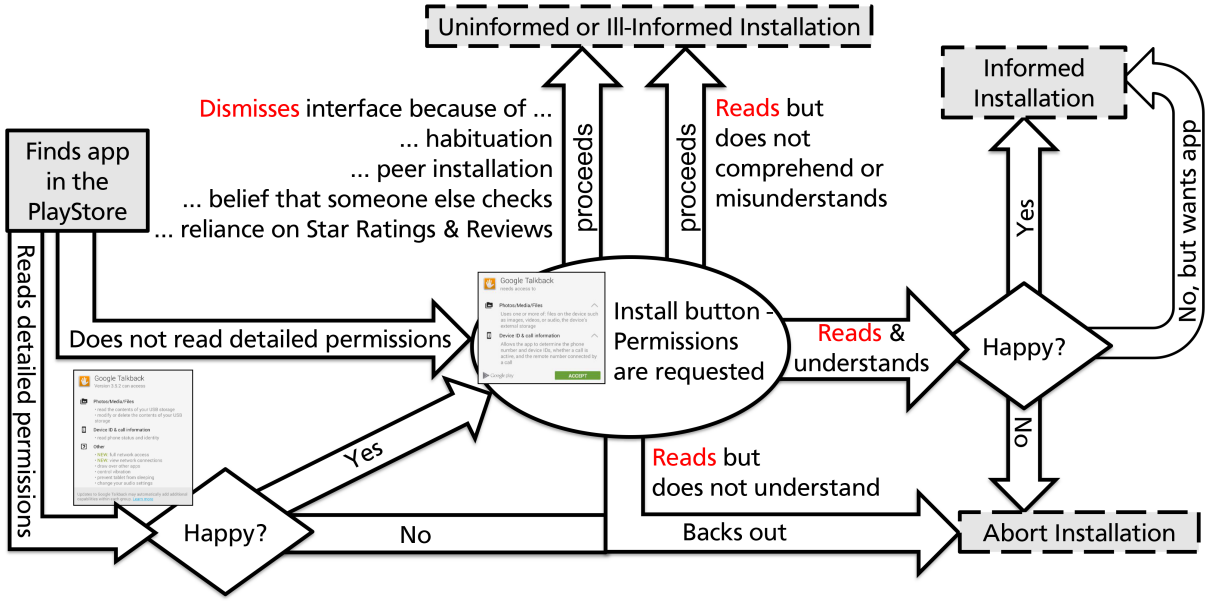

Figure 4: How do Smartphone Owners decide whether or not to grant Permissions?

fall prey to the habitation effect created by myriad applications people use daily. The fact that many people are unaware that permissions even exist for mobile apps seems to confirm that this might be happening [11].

Some people believe that it is someone else's responsibility to ensure that apps respect their privacy [4] and consequently skip through the permissions interface without taking much notice. Finally, there is evidence that many rely heavily on star ratings, full text reviews or word of mouth when deciding whether to install an app or not [4], so when they see the interface they simply grant the permission without thinking about it.

2.2.2. Considering Permissions but Installing due to Social Pressure

If people do take note of the permissions dialogue, they might still install an app, despite its invasiveness and gratuitous permissions, because their peers have installed it without qualms [12]. This kind of unthinking behaviour has also been observed in other contexts [13].

2.2.3. Problems when Considering Permission

There are a number of problems facing those who *do* consider permissions. They can only make an informed decision if they understand the permissions mechanism and the associated risk(s) [11]. They also need to be informed enough to rely on the signal-bearing indicators in deciding whether to grant access or not. A number of researchers have concluded that this understanding is not necessarily a given [4, 14, 15, 16]. Kelley *et al.* [4], for example, declare the permission descriptions to be '*at best vague, and at worst confusing, misleading, jargon-filled, and poorly grouped*' (p.78).

Even if people do understand the extent of the requested permissions, the atomicity of the process is problematical. Users cannot grant a subset of the requested permissions: it is all or none. If they choose the latter they have to abort the installation. This is the case for legacy and recent Android versions. Other potential issues are the timing of the appearance of the permissions interface, since permissions are requested *after* the owner has decided to install. To withdraw at that stage is cognitively expensive, disappointing and loss-inducing, something humans are averse to [17]. These issues probably combine to maximise the incidence of unwise installations.

2.2.4. Summary

Figure 4 summarises the app installation decision-making process. When researchers propose alternative interfaces, they primarily address those cases where alternative apps are indeed available and target people who would take the time to peruse permission interfaces. The fact that permissions are requested *after* a decision is made to install should be addressed by providing the information in a more accessible place in the Google Playstore, together with the ratings and the description. We do not address the timing issue but rather focus on the interface itself.

2.3. Alternative Interfaces

Researchers have proposed a number of alternative layouts to address the issues identified in the previous subsection. The alternatives were tested in one or more user studies, as compared to the legacy Android permission interface. All alternatives were superior in encouraging users to choose apps with fewer privacy invasions (usually they had to select one out of a set of apps all providing the same functionality and with roughly the same ratings).

Kelley *et al.* [5] proposed a layout called 'privacy facts' (see Figure 5 (A)). They cluster the requested permissions into two categories. First, the category 'This app collects your' with eight bullet points. Second, the category 'This app uses' with two bullet points for 'Advertising' and 'Analytics'. They used automated tools to gather this information.

Kraus *et al.* [6] provided a visualisation of *permission-related statistical data* to enable the users to assess the app in question in relation to other apps in the same category (see Figure 5 (B)). They used a horizontal risk slider to visually link the actual number of requested permissions with the mean, minimum and maximum number of requested permissions in the same app category.

Lin *et al.* [7] investigated a *crowd-based approach* to determine which permissions other users would expect a given app to request. More precisely they provide them with reasons why access should be granted, based on source code analyses. Possible reasons were "core functionality", "sharing and tagging" or "advertising/market analysis". From the answers the following type of statements were deduced: "95% of users were surprised this app sent their approximate location to mobile ad providers". The corresponding interface is displayed in Figure 6 (A))

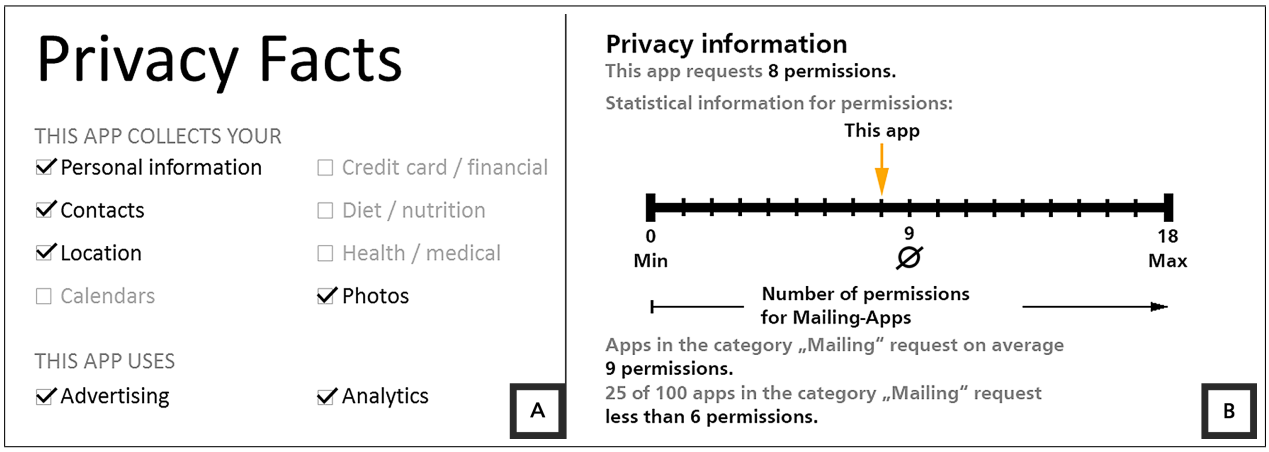

Figure 5: (A) on the left 'Privacy facts' by Kelley *et al.* (B) on the right the permission interface proposed by Kraus *et al* [6] (translation).

Harbach *et al.* [8] present the permissions together with personal information such as photos stored on the particular Smartphone or the owner's actual location (see Figure 6 (B)). The idea is to help people to imagine the consequences of requested permissions, rather than merely listing them.

Further approaches and summary. Other papers address the permissions problem and propose providing additional information. Table 1 presents an overview of the literature and the provided information. Since all propose providing additional information or substituting different information, and do not propose a particular interface to present them to the user, they are not discussed any further.

A final note about the iPhone-similar interface. The latest Android interface deals with the granularity of permission granting in Android in a way that seems superior to the legacy and recent interfaces. On the downside, it does not give the installer the option of considering all requested permissions at once in order to detect suspicious or unwanted combinations. The latter is usually a reliable indicator of malicious apps [24, 25]. We are unaware of any publication that investigates this type of interface in terms of privacy-related decisions. Judging permissions at runtime, instead of judging at installation, did not match the purpose of our study and we thus did not include this interface in our study.

3. The COPING Interface

We first consider the information that could be provided by a permission-granting interface. The list of the permissions being requested by an app is core. There should also be an assessment of the requested permissions, e.g. whether they are risky and/or needed, and the interaction between different permissions.

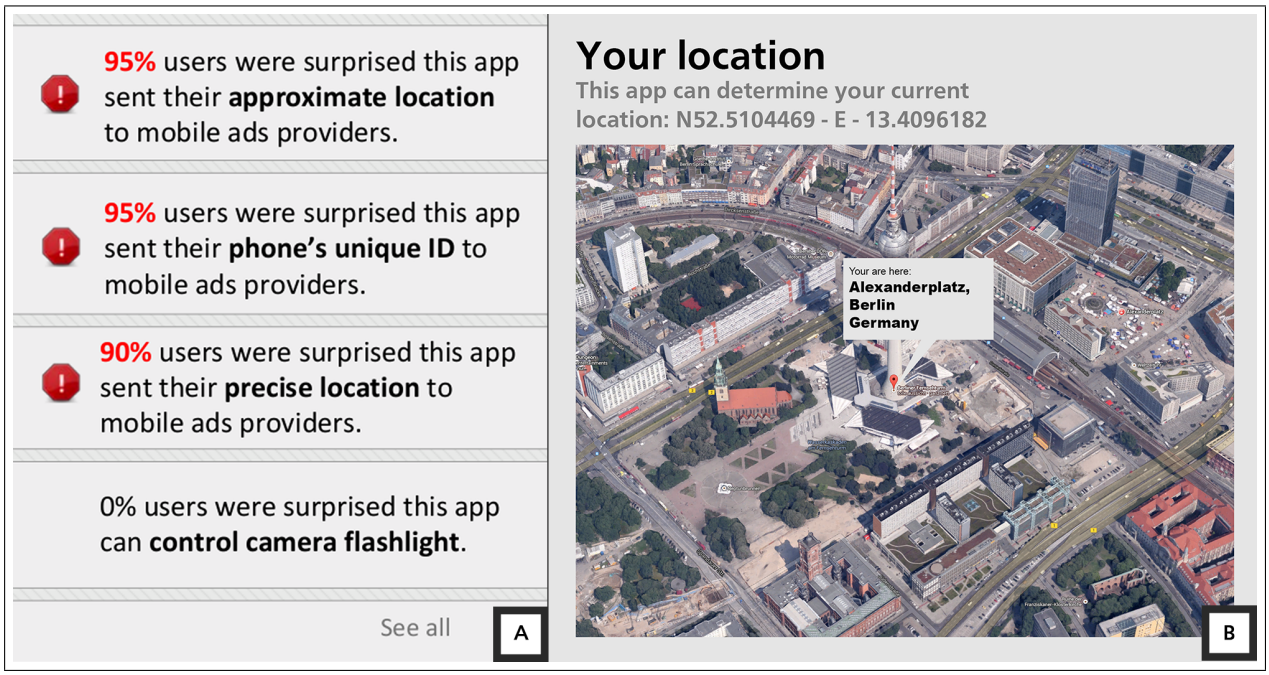

Figure 6: (A) on the left permission interface by Lin *et al.* [7] (B) on the right an example of the 'Your location' permission as it was used by Harbach *et al.* [8]

Since the interface proposed by Lin *et al.* [7] already listed all permissions, and accompanied this with an assessment thereof we based our own proposal on theirs. However, we included even more information about the permission-requesting behavior to enhance comprehensiveness. We changed the term 'User' to 'Expert' to more effectively differentiate this rating from normal user star ratings. We added more details, in particular:

1. At first the information interface is introduced with an overall judgment about the full permission set requested by the app. Thus, a user could make a decision based on a single glance.
2. A ranked and grouped list of the requested permissions is then provided based on the percentage of experts who considered the particular permission concerning in the context of the app's functionality. The grouping is based on the reasons cited by experts for their rating of the permissions as being concerning, or not. Note that instead of using a standard probability format to present the expert assessment as Lin *et al.* did, we used a natural sampling of frequencies. This is easier for users to understand [26].
3. A warning sign was displayed, as proposed by Lin *et al.* if the percentage exceeds a threshold (50 out of 100).
4. Possible permission interactions, and the associated risks thereof, was also provided. For instance, data collection is much more critical if an app can also access communication channels which could mean that such data can be easily transmitted to a server for archiving, analysis and sharing.

Table 1: Overview of usable data types and sources, with related publications

Information to Present	Data Source	Related Publication(s)
Permission Types	Android manifest	(already implemented)
Statistical data about permission requests in the same category (mean number, min, max etc.)	Android manifest of apps in the same category	Kuehnhausen and Frost [18]
Intercorrelated permission usage in the same app category	Android manifest of apps in the same category analyzed via machine learning methods	Barrera *et al.* [19]
Over-privileged status	Automated testing tools	Felt *et al.* [14]
Reasons for request	Natural language processing of app description	Pandita *et al.* [20]
	Automated testing tools	Enck *et al.* [21]
	Crowd sourcing	Lin et al. [7]
	Explicit information by the developer(s)	Maseberg *et al.*[22]
Information leakage	Automated testing tools	Lortz *et al.* [23]

All the tested interfaces, and their classification with respect to comprehensiveness, are displayed in Figure 7.

4. Study

We conducted an online study using a mixed-model design to compare the alternative interfaces as described above. All proposed interfaces were compared to the "legacy" Android interface. This interface was chosen as the baseline because the other researcher-proposed alternatives were all compared to this interface.

Every participant was randomly assigned one interface alternative. They were required to choose one out of three proposed apps for three different app types. This approach follows the study design used by Kelley *et al.* [5] who, in turn, adapted the design used by [27]. The apps to choose from are described in more detail in section 4.4. All figures and questions provided have been translated from the original German. Before describing the hypotheses and the study design, we present some pre-considerations.

4.1. Pre-Considerations

Comprehensiveness. The permission interfaces discussed in the previous sections differ in terms of comprehensiveness. The legacy permission interface provides a linear list of all permissions and we compare all others to this interface in terms of lesser or greater comprehensiveness in order to propose a ranking to support analysis.

Compared to the legacy permission interface, the interfaces proposed by Kraus *et al.* [28] and Kelley *et al.* [5], as well as the recent Android interface, provide less comprehensive

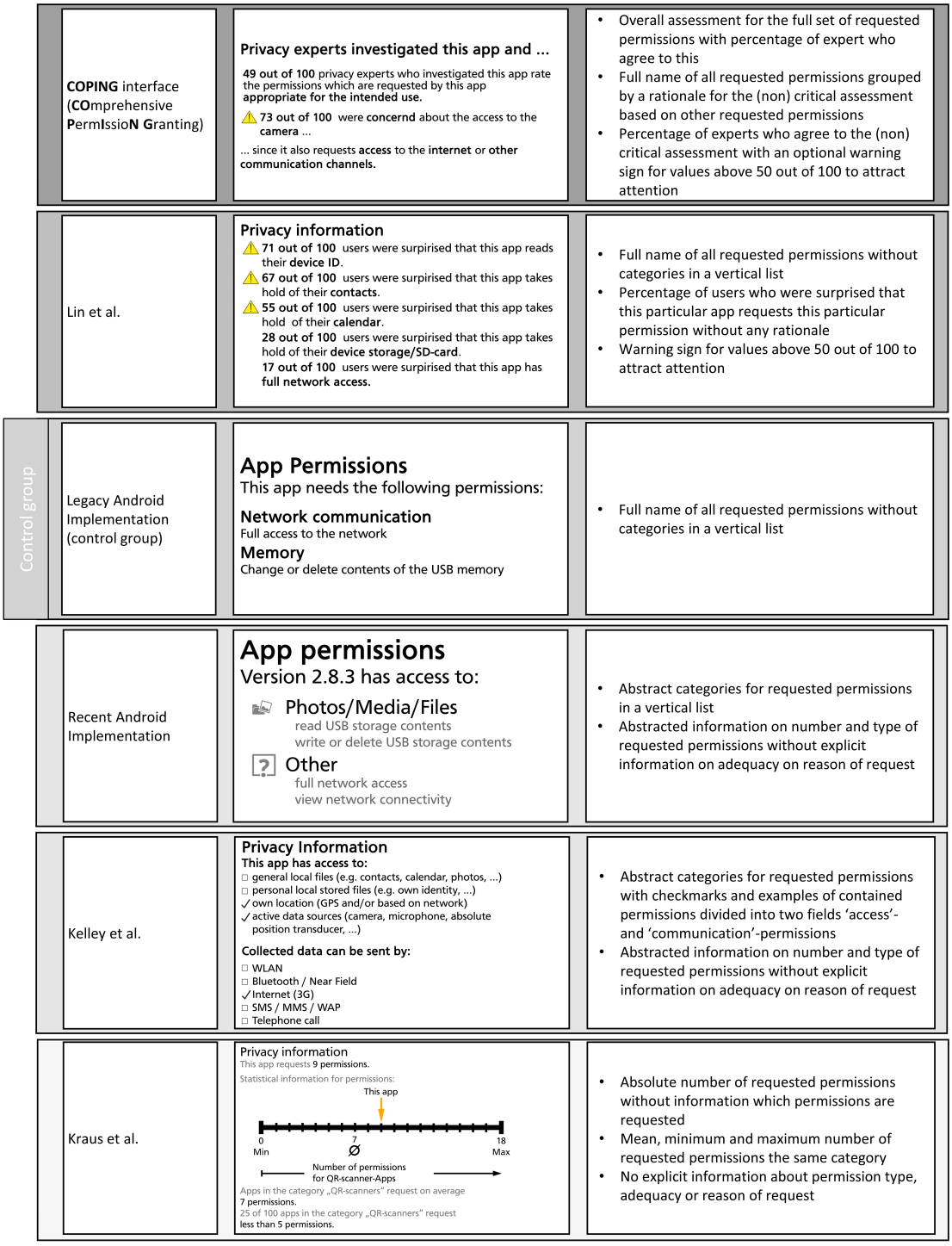

Figure 7: Overview of Tested Permission Interfaces

information. The proposals by Lin *et al.* [7] and Harbach *et al.* [8], on the other hand, provide more comprehensive information than the legacy interface, as does the COPING proposed interface.

Considered interfaces. We decided not to include the alternative interface proposed by Harbach *et al.* [8] for two reasons. Firstly, the main thrust of their proposal was to provide examples of actual private data from the personal Smartphone. This was not possible to provide in an online survey. In the second place, the information offered by their interface is somehow tangential to the kind of information the other interfaces provide, and could feasibly be used to improve the other interfaces.

As our focus was on permissions and not assuming apps are analysed with corresponding tools, we made some changes to the proposals by Kelley *et al.* and Lin *et al.*, ensuring that the core ideas were retained. We adapted Kelley *et al.*'s proposed interface, as shown in Figure 7. We discarded all information in the "collecting" area, which cannot be acquired solely by assessing permissions or other meta-data provided by the Play Store or the specific app. The same is true for "advertising" and "analytics" in the "usage" area. Note, the term "analytics" was generally not well understood by their study's subjects. To retain their main idea of collecting data and using (after sending) data, we included, in addition to options for 'This app has access to', a number of options for 'Collected data can be sent by'.

Similar to Kelley *et al.*'s approach we omitted information that would not be available via permission analyses, as performed by Lin *et al.*. This results in the omission of the reason statement, which could be "core functionality", "sharing and tagging" or "advertising/market analysis".

Integration into the Play Store. All screenshots used in the study were integrated into an adaptation of the Google Play Store interface and also depicted a logo, name, description, and user ratings.

4.2. Hypotheses

We tested the following hypotheses based on the findings from the literature:

H1 Participants perform differently based on the permission-granting interface they were assigned to.

- **H1**$_1$ Participants using the interface proposal by Kraus *et al.* perform better[6] than participants who see the *legacy* interface.
- **H1**$_2$ Participants using the interface proposal by Kelley *et al.* perform better than participants who see the *legacy* interface.
- **H1**$_3$ Participants using the interface proposal by Lin *et al.* perform better than participants who see the *legacy* interface.

[6]We consider performance to be *better* if participants select the most privacy friendly app.

H1$_4$ Participants using *the COPING interface* perform better than participants who see the *legacy* interface.

H1$_5$ Participants using the *recent* Android interface perform better than participants who see the *legacy* interface.

H2 Subjective helpfulness ratings of the interface will be different based on the permission-granting interface participants see.

H3 The decision time will be different based on the permission-granting interface the participants see.

H3$_1$ Participants using the interface proposal by Kraus *et al.* will make decisions more quickly than participants who see the legacy interface.

H3$_2$ Participants using the interface proposal by Kelley *et al.* will make decisions more quickly than participants who see the legacy interface.

H3$_3$ Participants using the interface proposal by Lin *et al.* will make decisions more quickly than participants who see the legacy interface.

H3$_4$ Participants using the COPING interface will make decisions more quickly than participants who see the legacy interface.

H3$_5$ Participants using the recent interface will make decisions more quickly than participants who see the legacy interface.

4.3. Study procedure

Our study was hosted by SoSciSurvey[7], a survey provider located in Germany. The study consists of the following phases:

Phase 0: Welcome and introduction. The study commenced with a welcome interface, which contained a short description of our research group, the supposed goal of the actual study, the anticipated duration as well as privacy-related information. The stated goal of the study was the evaluation of user interaction with a Smartphone, especially with apps. Neither privacy nor security was mentioned. The participant was prompted to imagine he or she got a new Android Smartphone and had to choose some new apps from the Google Play Store. Participants were told about the actual purpose of the study after phase 3 was concluded. There was also a short note about a lottery in which participants could win a prize for participating in the study.

Phase 1: Android experience. Participants were asked whether or not they owned an Android device and how long they had been using it. Based on their answer to this

[7]to be provided

question, they got just the normal instruction for the next phase or an additional short explanation of Android and the Google Play Store.

Phase 2: Decision Situations. All participants were asked three times to choose their preferred app out of three apps. Figure 8 depicts an example of the decision situation for one category. The categories were Sudoku, e-mail and QR-code-scanning app. For more details refer to section 4.4. For every category we offered three alternative apps. Tailoring of app-selection-interface (as shown in figure 8) for each of the decisions was randomized, i.e. app-name and app description as a bundle and each of those bundles (three for each category) were randomly assigned for every decision situation with one out of three app logos, developers, user ratings, privacy-levels and positions (left, middle, right). We included user ratings since this is provided by all Android app-selection-interfaces (as well as the app description and name); leaving it out would compromise validity. To control the potential side-effects we limited the variance of the user ratings (between 4.1 and 4.3) as well as fix the total sum of reviews to an equal magnitude. The sequence of the categories and the permission interface type were also randomized for every participant. The decision time for every decision and every participant was recorded.

Phase 3: Questions about usual app choice behavior. Participants were asked if they recognized one or more of the presented apps. In addition, they were prompted to estimate how often they used the Google Play Store, how many apps they have installed on their own device and how many different apps they use per week. We also asked whether they had noticed any differences to the normal Play Store while choosing the apps in our study, to indicate whether the integration of the permission interface was noticed or not.

Phase 4: Questions about the displayed permission interfaces. After finishing phase 3, participants were given a short explanation of the real purpose of our study, accompanied by an example image with a red rectangle to mark our added permission interface in case they didn't realise it in phase 2. They were asked whether or not they had read the given permission interface, whether or not they consulted the provided information to choose the app, whether the information was understandable and helpful and whether or not they believed that such an interface would help them to better protect their privacy. (We did not ask whether privacy was important to them, or not, since such questions are normally affirmed without much impact on real world behavior.)

Phase 5: General demographics. To check for demographic-driven effects we asked our participants to provide some general information such as gender, age, occupation and educational levels.

Phase 6: Debriefing. Participants were asked whether or not they wanted their names entered into the lottery and whether or not they wanted to be informed about the study results. We also gave them some information about the latest update to the permission interface in the Play Store, and the associated risks thereof.

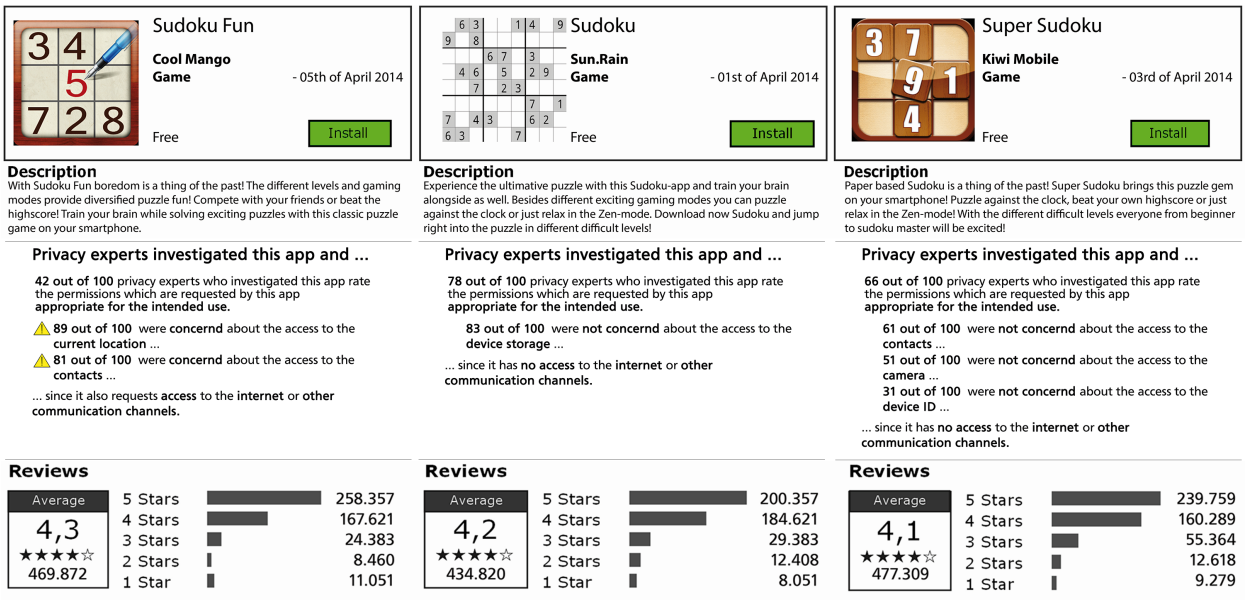

Figure 8: Example interface for the decision situation for a Sudoku app in our study (translation)

4.4. Decision Scenarios

Some permission interfaces reduce comprehensiveness, essentially providing summaries [28, 5]. Others provide extra information, improving the comprehensiveness [7]. To assess efficacy we constructed three situations where participants would be asked to make a choice in order to assess decision quality[8]. These are contrived to mirror real world app choices. They are represented by one category of app each (Table 2). In each category apps were picked randomly from the less-popular apps in the Play Store to minimise prior knowledge. We edited the app descriptions to present an equal level of functionality in every decision situation and randomly assigned an app name and logo.

The first situation, which is represented by three Sudoku (a logic game) apps, represents a *simple decision* in terms of privacy. "Easy" in this case means that it is obvious that a Sudoku app does not need any permissions to perform its main functionality. Any requested access to private data or communication channels does not support the app's main task and therefore, with respect to privacy, the best choice is the alternative which requests the fewest permissions.

The second situation, which is represented by three mailing apps, represents a *complex decision*. The functionality of the mailing app suggests a requirement to access private data (contacts) and communication channels (the Internet connection) to provide the core functionality (as opposed to the Sudoku apps). The proposed alternatives were contrived

[8]In this paper we consider decision quality to depict the best possible decision from a privacy perspective, while controlling for functionality aspects of the given app alternatives.

Table 2: Summary of decision situations

App category	Main characteristics of decision situation	Best decision wrt. privacy
Game app (Sudoku) - simple decision	Simple app, low number of permissions, easy pattern of 'acceptable permissions' since no functional need for accessing private data	app requesting the lowest number of permissions
Mailing app - complex decision	Complex app with complex and many permissions since functionality needs access to private data and/or communication channels	app requesting the lowest number of permissions
QR Code scanner app - deceptive decision	Medium complex app but complex choice scenario, since the app with the lowest number of requested permissions has a privacy invasive combination of permissions	app with no privacy invasive permission combinations (in our case the app without access to any communication channels such as WiFi or the mobile Internet)

so that that the privacy-friendly decision is also the one which requests the fewest permissions. The additional permissions requested by the more privacy-invasive apps are clearly not required for the app to deliver its core functionality.

The third situation, which is represented by three QR-Code scanning apps, represents a *deceptive decision* context. The complexity of the app in question is moderate, since the functionality is more complex than in the Sudoku apps (which need no permissions) but less functionality is delivered than by emailing clients which genuinely require access to private data and/or communication channels. A QR-scanning app will undeniably require access to the camera to fulfill its core function. At least in Android there is no reasonable need to access to communication channels since an app doesn't need a special permission to communicate with other apps. So a link extracted from a QR-Code could be sent via the operating system to a browser app without the need for full network access (and all the corresponding risks) for the app itself. The most privacy-friendly choice is no longer the app which requests the fewest permissions but rather the app which doesn't request any access to communication channels. To fix on the best alternative the optimal combination of the requested permissions must be evaluated by the participants. So mere counting is not enough. A true informed decision must be made.

4.5. Ethics

Ethical requirements with respect to respondents' informed consent and data privacy were in line with the university's ethical guidelines. Participants first read an information page they were assured that their data would not be linked to their identity and that the responses would only be used for study purposes. Furthermore, SoSciSurvey stores all data in Germany and is thus subject to strict EU data protection law. Contact data, namely an email address, was stored in a separate data file, used for a lottery and subsequently deleted. At the end of the study participants were debriefed. We provided some additional information about the newest permission interface in the Play Store and the privacy risks related to this.

4.6. Recruitment

The recruitment was carried out over online channels such as mailing lists, forums with various thematic scopes, university groups in Facebook and other social networks. We also used offline channels to distribute the link to our study like malls, billboards and flyers in canteens. In addition we recruited participants through a platform called "Workhub"[9] which is a German equivalent to Amazon Mechanical Turk. Every participant from Workhub received €3 as compensation for their participation.

4.7. Sample

From September to October 2014 344 participants completed our survey. On average, they were 25.7 years old with a standard deviation of 8.3 years. The youngest participant was 18, the oldest 75. Other relevant demographic information is summarized in Table 3. 255 (74.1%) participants owned an Android device. On average they had used it for 13.8 months with a standard deviation of 15.4 months. Out of 344 participants 59 (17.15%) had a high IT affiliation based on their stated occupation.

[9] www.workhub.com

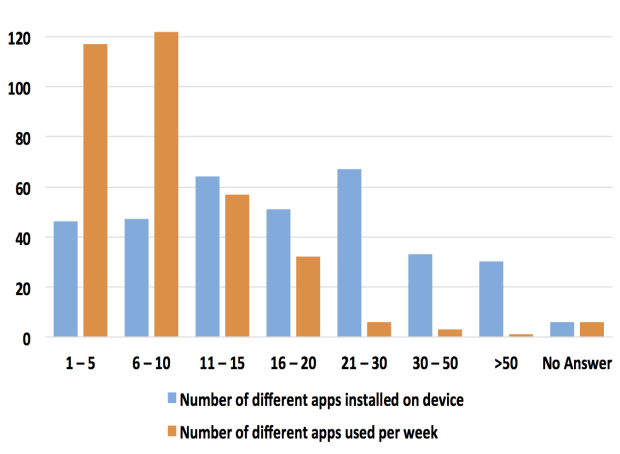

Figure 9: Apps installed and used

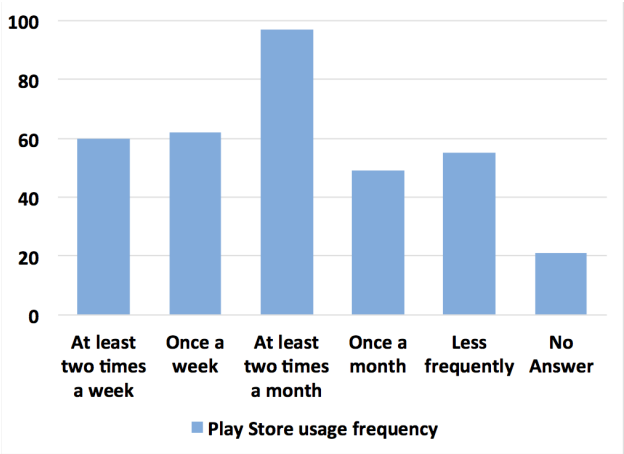

Figure 10: Frequency of Play Store Usage

Table 3: Demographic data about our sample

		N	%
Gender	Male	197	57.3
	Female	139	40.4
	Other	8	2.3
Education	Pre high school	23	6.8
	High school or equivalent	77	22.7
	College or equivalent	156	45.9
	Bachelor or equivalent	40	11.8
	Master or equivalent	29	8.5
	PhD or equivalent	7	2.1
	Other	12	3.5

Table 4: Mean frequency of correct decision for every permission interface and decision situation (standard deviation in parentheses; down arrow indicates a significant lower, up arrow a significantly higher value as compared to the legacy Android permission interface)

Frequency correct decision for	**Sudoku simple situation**	**Mail-app complex situation**	**QR-Scanner deceptive situation**	**Overall**
Legacy Android	.75 (.78)	.75 (.44)	.22 (.42)	1.72 (.78)
Recent Android	.71 (.46)	↓.50 (.50)	.17 (.38)	1.38 (.75)
Kraus *et al.*	↓.57 (.50)	↓.55 (.50)	↑.52 (.50)	1.64 (1.15)
Kelley *et al.*	↓.53 (.50)	.62 (.49)	.31 (.47)	1.46 (.82)
Lin *et al.*	.72 (.45)	.75 (.44)	.14 (.35)	1.61 (.76)
COPING	.71 (.46)	.76 (.43)	↑.71 (.46)	↑2.18 (.84)

219 (67.8%) participants visited the Play Store at least twice a month. 215 (62.5%) participants had installed between ten and fifty apps on their device. Figures 9 and 10 depict the app usage behavior of our sample.

253 (73.55%) of participants didn't recognize any of the given apps. 12 (3.49%) knew one or more of the Sudoku apps, 42 (12.21%) knew one or more of the mailing apps and 61 (17.73%) knew one or more of the proposed QR-Code scanning apps. The detailed sample demographics, especially about usage behavior, are integrated to enable other researchers to replicate our study with comparable samples and/or to compare their results cross culture with ours.

5. Results

The tests for the different hypotheses are presented in the corresponding subsections.

5.1. Hypothesis H1: Overall number of privacy-friendly decisions

Since the 'privacy friendliness' of app alternatives is not on an interval scale, it could not be directly used as a dependent variable in an analysis of variance (ANOVA) testing all participants and decision situations. We first calculated the frequency of correct decision, i.e. choosing the privacy-friendly alternative, for every participant as an interval-scaled measurement of the effectiveness of the used permission interface. To test our first hypothesis we coded every decision of every participant (i.e. three per participant), whether he or she chose the most privacy-friendly app (=1) or not (=0). We then calculated the overall frequency of a correct decision for every participant; i.e. how often he or she picked the app which was least privacy invasive.

This frequency value was used as our dependent variable in a one-way analysis of variance with our six different permission interfaces as treatment groups (i.e. 'permission interface type' was the independent variable). To test the hypotheses we then calculated the simple contrast (legacy Android interface was the control) for *a priori* hypotheses. Every custom interface as well as the new Android interface, was compared to the legacy Android interface as the control group and baseline, which all tested interfaces were shown to be superior to.

The last column of Table 4 shows the means and standard deviations for all six interfaces. ↑ indicates a significantly better, ↓ a significantly worse performance, as compared to the control group (legacy Android interface). Only the COPING interface performed significantly better ($p = 0.005$) as compared to the legacy Android interface, over all three different choice situations. Hypothesis 1 is only true for the most comprehensive interface. The other sub-hypotheses must be rejected since all other interfaces deliver no significant improvement over the legacy one. A closer look at the results reveals that the recent Android interface, contrary to our expectations, performed very close to being significantly worse than the legacy interface ($p = 0.052$).

5.2. Hypothesis H1: Number of privacy-friendly decisions per choice situation

Besides the analysis of the overall performance we conducted an analysis of variance with repeated measures to investigate whether our six permission interfaces performs differently in our three decision situations. The within-subjects factor was the three decision situations

Table 5: Frequency, mean and standard deviation values for the subjective assessments of all six permission interfaces (down arrow indicate a significant lower, up arrow a significant higher value compared to the legacy Android permission interface). U=Understandable; H=Helpful; P=Protective

T	**Read**		**Used**		**U**		**H**		**P**	
	Y	**N**	μ	σ	μ	σ	μ	σ	μ	σ
Legacy Android	48	3	5.78	(1.63)	5.60	(1.29)	5.65	(1.51)	6.15	(1.31)
Recent Android	36	11	↓4.88	(2.22)	5.02	(1.61)	↓4.96	(1.80)	↓5.36	(1.78)
Kraus *et al.*	47	11	↓4.65	(2.15)	5.17	(1.72)	↓4.90	(1.79)	↓5.38	(1.98)
Kelley *et al.*	45	9	5.76	(1.74)	5.78	(1.41)	5.89	(1.38)	5.85	(1.37)
Lin *et al.*	62	6	5.44	(1.97)	5.73	(1.52)	5.58	(1.51)	5.68	(1.66)
COPING	43	4	5.71	(1.74)	5.67	(1.26)	5.82	(1.38)	6.25	(1.20)

and the between-subject factor was the permission interface type. We found a significant interaction between the two ($F = 6.237$; $p < 0.001$). Table 4 provides the detailed mean and standard deviation values for every interface in every situation. To further investigate this we conducted a multivariate analysis of variance[10] (MANOVA) with the decision situation and the permission interface type as independent variables and the frequency of the correct decision as dependent variable. Simple contrasts were calculated to test both hypotheses for each of our situations.

In the simple decision situation, where our participants had to choose a Sudoku app the two less comprehensive custom interfaces, they performed significantly more poorly than the control group ($p = 0.042$ and $p = 0.015$). The two more comprehensive interfaces deliver no significant difference. This is also true for the new Android permission interface. In the situation with the more complex mailing app the Kraus *et al.* interface ($p = 0.026$) and the recent Android implementation ($p = 0.007$) perform significantly worse than the control group. The other three custom interfaces deliver no significant differences. In the deceptive situation, where the best decision was the QR-code-scanning app without any access to communication channels (not the one with lowest overall number of requested permissions), the Kraus *et al.* ($p < 0.001$) and the COPING interface ($p < 0.001$) deliver a significantly better performance than the control group. The COPING interface outperformed the interface proposed by Kraus *et al.*. The other interfaces perform as well as the control group.

5.3. Hypothesis H2

Our second hypothesis postulates a significant difference between the legacy Android interface and all other interfaces in terms of the subjective helpfulness. The detailed ratings for every permission interface are displayed in Table 5. We carried out a MANOVA to test this. Dependent variables were the seven-point Likert scale values for 'the permission interface...' 'was used', 'was understandable', 'was helpful' and 'seemed protective of individual privacy'. The ratings were collected during phase 4 of our study. This analysis revealed significantly worse usage ($p=0.026$), helpfulness ($p=0.025$) and protectiveness ($p=0.016$) ratings for the recent Android interface. There were significant differences for the Kraus *et al.* interface ($p=0.002$ / $p=0.011$ / $p=0.011$) as compared to the legacy Android interface as illustrated by Table 5. The more comprehensive interfaces deliver no improvement as compared to each other and to the control group.

5.4. Hypothesis H3

Since our tested interfaces have different levels of comprehensiveness, it seems plausible that the mean decision time varies as well. Figure 11 depicts the mean decision time for every permission interface. Although the absolute difference between the fastest (the recent Android interface with 26.31s) and slowest (the COPING interface with 51.55s) seems large,

[10]Since we conducted our analyses in SPSS version 21 there was no possibility to calculate those contrasts directly in the aforementioned ANOVA with repeated measures. Since our contrasts each compare the interface type within a given decision situation we conducted a MANOVA to avoid as best as possible the cumulating of first error probability

variance for all interfaces was high. We analysed the decision time with a univariate ANOVA where the permission interface type was the independent variable and the mean overall decision time was the dependent variable. The only significant difference we observed was between the aforementioned two extremes ($F = 2.60$; $p = 0.03$) with a low effect size ($\eta^2 = 0.037$). Hence Hypothesis 3 is only partly confirmed since only two interfaces delivered significant differences.

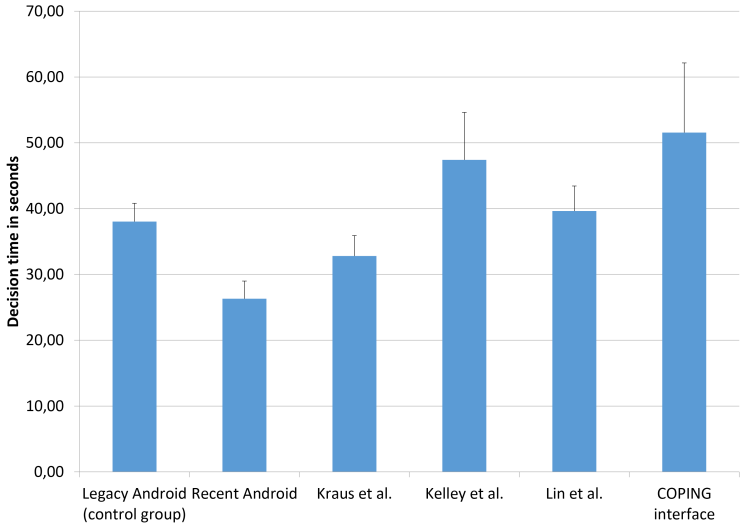

Figure 11: Mean decision time per interface type (in seconds) with marked standard error

6. Discussion

We were unable to replicate the findings of the researchers who reported improvements, as compared to the legacy Android interface. Only our own proposal – the most comprehensive interface – performed better, as compared to the legacy interface, whereas all others performed equally or more poorly. A closer look at the performance results in the three different decision situations reveals that the major differences arise in the deceptive choice situation. Here the best choice can not easily be ascertained by merely counting the requested permissions. The kinds, as well as combinations of permissions, have to be evaluated and understood by the user.

With this fact in mind it seems plausible that the information abstraction from the legacy Android implementation which is applied in order to formulate the two less comprehensive interfaces essentially leads to poorer performance in terms of privacy. The least comprehensive interface provides only statistical data without explicitly mentioning any of the requested permissions. Thus, privacy-invasiveness can only be assessed by comparing the number of permissions with the mean or quartile number of other apps in the same category. Our results show a significantly poorer performance as compared the legacy Android

interface, even in the simple decision situation. Significantly poorer ratings, in terms of helpfulness and protectiveness, also seem to indicate that this kind of information visualization is poorly suited for everyday use.

The second least comprehensive interface [5] was also outperformed by the legacy Android interface. The distinct categories with checkmarks seem to encourage tallying instead of careful consideration as Harbach *et al.* [8] already pointed out. Similar to the recent Android permission interface, it becomes hard for the user to distinguish between apps with a different number of requested permissions in a particular category. It seems that people cannot reliably identify privacy-invasive apps if the permission set is knowingly assembled by the developer. This is demonstrated in our deceptive decision situation, where the best choice, privacy-wise, was actually not the one with the fewest checkmarks.

The second most comprehensive interface delivered a performance comparable to the legacy Android version. The additional information provided by user assessment of the single permission and warning signs for unexpected permissions did not foster user understanding of the permission set. Our participants seemed simply to compare the permission list length and often chose the one with the shortest list (as those with the legacy Android interface do).

Only our proposal – the most comprehensive interface – provided enough understandable information about the permission set to enable our participants to assess the deceptive situation properly. This interface alternative delivers a significantly improved performance over all three analysed decision situations, as compared to the legacy Android version. It outperforms all other tested interfaces, since they all perform equally. The costs do not increase significantly, since we found no significantly increased decision times as compared to the other interfaces, with the exception of the recent Android interface, which performs significantly more poorly. Since the mean decision times were the highest, as compared to the other interfaces, it seems plausible that our participants did not trust the expert ratio in the beginning (whether the permission set seemed appropriate or not) but read the more detailed explanations about why the set seemed appropriate (or not).

Felt *et al.* [11] reported that around 20% of their sample were 'expert' users and concluded that they could help other consumers by writing reviews when they encountered inexplicable permission requests. This will undoubtedly help those who rely on review scores rather than trying to assess permissions themselves. To generate the data needed for such an interface, a crowd-sourced approach, such as the one used by Lin *et al.* [7] could be feasible. Knowledge of the permission system should be tested, like Felt *et al.* did in their study. Apps should be displayed to determine whether the requested permission set seems appropriate in terms of delivering the functionality, or not.

Since many users do not understand Android permissions very well, as confirmed by several authors (e.g. [11]), many proposals reduce comprehensiveness in order to simplify the interface. Google's recent permission interface seems designed along these lines leading to a reduced number of permissions only presenting abstract data. Instead of empowering the user with better tools to understand associated risks and make an informed decision, the abstract visualizations led to a situation where most of the users just counted checkmarks or compared the length of the permission lists. Such a 'grant the fewest' heuristic, which is

best illustrated by the choice behavior in our deceptive situation, could easily be abused by any entity which knowingly picks a permission set to match those abstract representations, but which actually provides access to a great deal of private data.

This is especially true for the recent Android interface which delivered the poorest performance in our study. It also received relatively poor subjective ratings in terms of helpfulness, usage and how protective it seemed. The decision time was significantly shorter for this interface, as compared to the most comprehensive interfaces, but this clearly impairs decision quality. It should be noted that our evaluation of the permission interface is in the nature of a best-case scenario, since we used the more detailed interface that is only available via a small link at the bottom of the app details page (pre-installation). The installation interface, displayed during 'installation' or 'update' does not include the 'other' category, which contains many of the more privacy-invasive permissions such as "full network access" or "use credentials". Even if an update to a new app version requests additional permissions from this category it is still not displayed [9]. This also applies to the iPhone-similar interface introduced with Android 6 "Marshmallow", where no permission-granting interface is shown during installation (at least for apps programmed with the most recent SDK). Requesting during runtime delivers benefits such as better association between functionality and permission, but there are also drawbacks. Previously granted permissions are assumed for subsequent execution. So it seems very hard to assess whether a permission is a threat to the individual's privacy since possible interactions can not be re-assessed.

6.1. Limitations

The first limitation is the absence of real-world behavior, since the participants did not have to make app installation decisions or install apps on their own devices. We also provided a side-by-side comparison of three apps which is, to the best of our knowledge, not available in any extant app store. On the other hand, since app markets are not solely accessible via small smartphone screens (at least the play store), but also have versions which can be used on PCs or laptops to choose apps and push them directly to mobile devices, even our side-by-side interface used in the study is not out of the question. For the single app assessment scenario, which is more likely for smartphones, the COPING interface used the same horizontal space and font size as all the other interfaces, including the two Android interfaces. So, there is no obvious reason why it should not fit on a smaller screen as the other interfaces do. Besides this, our participants also didn't have a real interactive app details page as provided by the Google Play Store.

Due to the technical requirements of the chosen survey platform all app detail pages were static images without the supported interaction possibilities. We also shortened all app descriptions so they only sketched the core functionality of the apps. Other parts, such as feature lists or change logs for the last app versions, were not part of our study since they normally provide no meaningful privacy information.

We only tested a small sample of available apps and app categories. This limits the applicability of our results. We also had to leave out the iPhone and Android 6 interfaces because they change the paradigm from 'judge at installation' to 'judge at runtime', this switch in paradigm warrants a completely different study.

Finally, our sample consists only of German language speakers and therefore cultural effects could be confounding factors.

We discovered that the most comprehensive interface delivered the best results. An even more comprehensive might perform even better. We thus make no claims as to the optimality of the COPING interface, only that it worked better than the others we compared it to.

6.2. Future work

Our results suggest a number of promising directions for future research. First of all we are already working on a prototype of a web form to collect crowd-based data to inform reviews for a number of real apps in the style of our most comprehensive interface. Thereafter, we plan to implement an assessment of permission understanding followed by a guided evaluation of a given app. We are also developing a prototype app to provide privacy-related data to users in addition to the normal meta-data from the Google Play Store. Since offering an alternative permission interface is impossible without Google's cooperation, our app should work as a proof-of-concept by providing an aggregated user evaluation of real-world behavior. This will enable us to conduct a field study to observe user behavior with far more realistic decision scenarios.

Our finding is that merely reducing comprehensiveness, and thereby complexity, does not necessarily lead to privacy-aware decisions. Since the dimensions of understandability and comprehensiveness of an interface follow somehow opposing trends — as one increases the other will probably decrease — it seems reasonable to investigate whether an ideal balance with respect to privacy-related decision quality can be identified (Figure 12).

For example, a very simple interface is easy to understand but provides little, vague or very abstract information, so the potential decision quality is impaired by the paucity of the provided information (as shown by the smaller red area compared to the green area). Increasing comprehensiveness also potentially makes it harder for the person to comprehend all the information. In a very complex interface (right-hand side of 12), all relevant information to support a decision is provided, but the decision quality might be hindered by complexity. To maximise decision quality it seems best to balance understandability and comprehensiveness of an interface, as Figure 12 suggests (the point where the red area is maximized). Such a 'sweet spot' represents the hypothetical point where the user is not overwhelmed by the amount of information but is also adequately informed. Since in our study the most comprehensive interface performed best in terms of decision quality it seems promising to search for the sweet spot by increasing comprehensiveness until it becomes clear that complexity is impairing decision making.

In line with this assumption the permission interface should evolve to a more sophisticated 'privacy summary' providing more information. Even though it performs well, there were a significant number of people (25 to 30%) who did not choose the most privacy-friendly app. There is thus room for improvement. One approach we want to pursue is to integrate data based on dynamic code analysis to visualize data flow, i.e. mapping functionality to the requested permissions. For this purpose the research project Zertapps[11], supported

[11] http://www.zertapps.de/

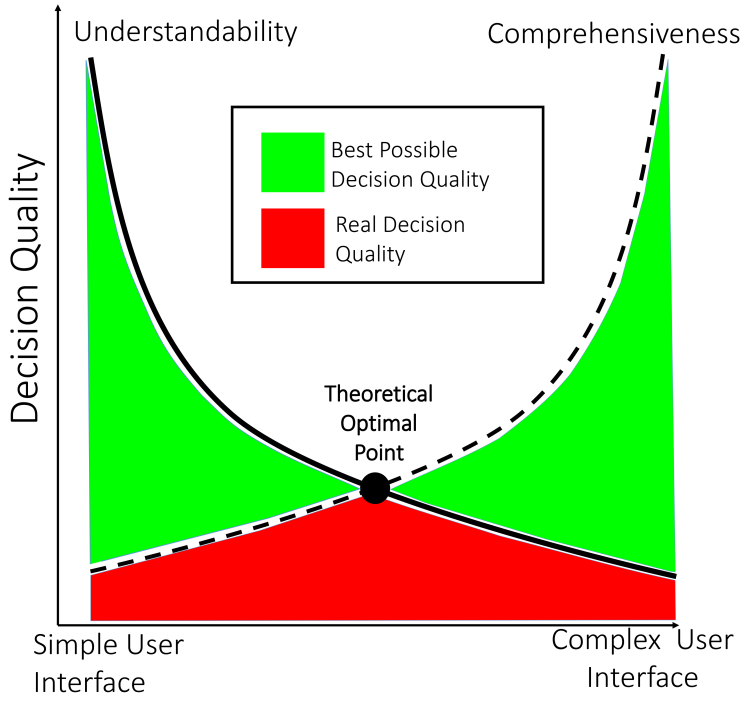

Figure 12: Balancing comprehensiveness and understandability to maximise decision quality; green area marks the theoretical best possible decision quality, while red area marks the real decision quality due to impairment by on of the factors

by the Federal Ministry of Education and Research in Germany, is currently developing a lightweight certification process for Android apps.

7. Conclusion

We tested a number of different permission interfaces, including the new COPING interfaces and compared them to the legacy and recent Android interfaces. We recorded whether participants chose the most privacy-friendly app or not.

Our results show that people have the ability to read and understand relatively complex interfaces. The intuitive approach of merely reducing comprehensiveness to improve decision quality seems, on reflection, an unsuitable strategy for improving privacy-related decisions. Our results suggest the idea of the existence of a sweet spot between comprehensiveness and understandability where user understanding of given situations is maximized without undue effort being expended. Future work in this area will attempt to develop an interface that approaches this sweet spot more closely than any of the current interfaces.

8. Acknowledgements

This article was supported by the German Federal Ministry of Education and Research in the framework of the project “Zertapps”.

References

[1] statista.com, Number of smartphone users* worldwide from 2014 to 2019 (in millions) (2015).
URL http://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/

[2] statista.com, Global market share held by smartphone operating systems from 2009 to 2015 (2016).
URL http://www.statista.com/statistics/263453/global-market-share-held-by-smartphone-\
operating-systems/

[3] Google, Dashboard with relative number of devices that share a certain characteristic - last access on September 2nd (2016).
URL http://developer.android.com/about/dashboards/index.html

[4] P. G. Kelley, S. Consolvo, L. F. Cranor, J. Jung, N. Sadeh, D. Wetherall, A Conundrum of Permissions: Installing Applications on an Android Smartphone, in: J. Blyth, S. Dietrich, L. J. Camp (Eds.), Financial Cryptography and Data Security, Vol. 7398 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 68–79. doi:10.1007/978-3-642-34638-5_6.

[5] P. G. Kelley, L. F. Cranor, N. Sadeh, Privacy as part of the app decision-making process, ACM Press, New York, New York, USA, 2013, p. 3393. doi:10.1145/2470654.2466466.

[6] L. Kraus, I. Wechsung, S. Moller, Using statistical information to communicate android permission risks to users, in: Socio-Technical Aspects in Security and Trust (STAST), 2014 Workshop on, IEEE, 2014, pp. 48–55.

[7] J. Lin, S. Amini, J. I. Hong, N. Sadeh, J. Lindqvist, J. Zhang, Expectation and purpose: understanding users' mental models of mobile app privacy through crowdsourcing, in: Proceedings of the 2012 ACM Conference on Ubiquitous Computing, ACM, 2012, pp. 501–510.

[8] M. Harbach, M. Hettig, S. Weber, M. Smith, Using personal examples to improve risk communication for security & privacy decisions, Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14 (2014) 2647–2656doi:10.1145/2556288.2556978.

[9] P. Gerber, M. Volkamer, K. Renaud, Usability versus privacy instead of usable privacy: Google's balancing act between usability and privacy, SIGCAS Comput. Soc. 45 (1) (2015) 16–21. doi:10.1145/2738210.2738214.

[10] S. Egelman, L. F. Cranor, J. Hong, You've been warned: an empirical study of the effectiveness of web browser phishing warnings, in: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, ACM, 2008, pp. 1065–1074.

[11] A. P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, D. Wagner, Android Permissions : User Attention , Comprehension , and Behavior, in: Symposium on Usable Privacy and Security (SOUPS) 2012, Washington, DC, USA, 2012.

[12] O. Kulyk, P. Gerber, M. El Hanafi, B. Reinheimer, K. Renaud, M. Volkamer, Encouraging Privacy-Aware Smartphone App Installation: What Would the Technically-Adept Do?, in: Usable Security Workshop (USEC), 2016.

[13] M. Walrave, W. Heirman, L. Hallam, Under pressure to sext? Applying the theory of planned behaviour to adolescent sexting, Behaviour & Information Technology 33 (1) (2014) 86–98.

[14] A. P. Felt, E. Chin, S. Hanna, D. Song, D. Wagner, Android permissions demystified, ACM Press, New York, New York, USA, 2011, p. 627. doi:10.1145/2046707.2046779.

[15] J. Lin, B. Liu, N. Sadeh, J. I. Hong, Modeling Users Mobile App Privacy Preferences : Restoring Usability in a Sea of Permission Settings, in: Proceedings of the tenth Symposium on Usable Privacy and Security, Vol. 1, Menlo Park, CA., 2014, pp. 1–14.

[16] S. Egelman, A. P. Felt, D. Wagner, Choice architecture and smartphone privacy: There's a price for that, in: The economics of information security and privacy, Springer, 2013, pp. 211–236.

[17] A. Tversky, D. Kahneman, Loss aversion in riskless choice: A reference-dependent model, The quarterly journal of economics (1991) 1039–1061.

[18] M. Kuehnhausen, V. S. Frost, Trusting smartphone apps? to install or not to install, that is the question, in: Cognitive Methods in Situation Awareness and Decision Support (CogSIMA), 2013 IEEE International Multi-Disciplinary Conference on, IEEE, 2013, pp. 30–37.

[19] D. Barrera, H. G. u. . Kayacik, P. C. van Oorschot, A. Somayaji, A methodology for empirical analysis

of permission-based security models and its application to android, in: Proceedings of the 17th ACM conference on Computer and communications security - CCS '10, no. 1, ACM Press, New York, New York, USA, 2010, p. 73. doi:10.1145/1866307.1866317.

[20] R. Pandita, X. Xiao, W. Yang, W. Enck, T. Xie, Whyper: Towards automating risk assessment of mobile applications., in: USENIX Security, Vol. 13, 2013.

[21] W. Enck, P. Gilbert, S. Han, V. Tendulkar, B.-G. Chun, L. P. Cox, J. Jung, P. McDaniel, A. N. Sheth, Taintdroid: an information-flow tracking system for realtime privacy monitoring on smartphones, ACM Transactions on Computer Systems (TOCS) 32 (2) (2014) 5.

[22] S. Maseberg, E. Bodden, M. Kus, A. Brucker, S. Rasthofer, B. Berger, S. Huber, K. Sohr, P. Gerber, M. Volkamer, Zertifizierteapps, in: 14th German IT Security Congress, no. TUD-CS-2015-0022, Stadthalle Bonn-Bad Godesberg, 2015.

[23] S. Lortz, H. Mantel, A. Starostin, T. Bähr, D. Schneider, A. Weber, Cassandra: Towards a certifying app store for android, in: Proceedings of the 4th ACM Workshop on Security and Privacy in Smartphones & Mobile Devices, SPSM '14, ACM, New York, NY, USA, 2014, pp. 93–104. doi:10.1145/2666620.2666631.

[24] W. Enck, M. Ongtang, P. McDaniel, On lightweight mobile phone application certification, in: Proceedings of the 16th ACM conference on Computer and communications security, ACM, 2009, pp. 235–245.

[25] A. P. Felt, M. Finifter, E. Chin, S. Hanna, D. Wagner, A survey of mobile malware in the wild, in: Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices, ACM, 2011, pp. 3–14.

[26] G. Gigerenzer, U. Hoffrage, How to improve bayesian reasoning without instruction: frequency formats., Psychological review 102 (4) (1995) 684.

[27] N. Good, R. Dhamija, J. Grossklags, D. Thaw, S. Aronowitz, D. Mulligan, J. Konstan, Stopping spyware at the gate: a user study of privacy, notice and spyware, in: Proceedings of the 2005 symposium on Usable privacy and security, ACM, 2005, pp. 43–52.

[28] L. Kraus, I. Wechsung, S. Möller, Using Statistical Information to Communicate Android Permission Risks to Users, in: G. Lenzini, G. Bella (Eds.), Proc. of 4th Int. Worshop on Socio-Technical Aspects in Security and Trust (STAST), IEEE, 2014.

9. Appendix

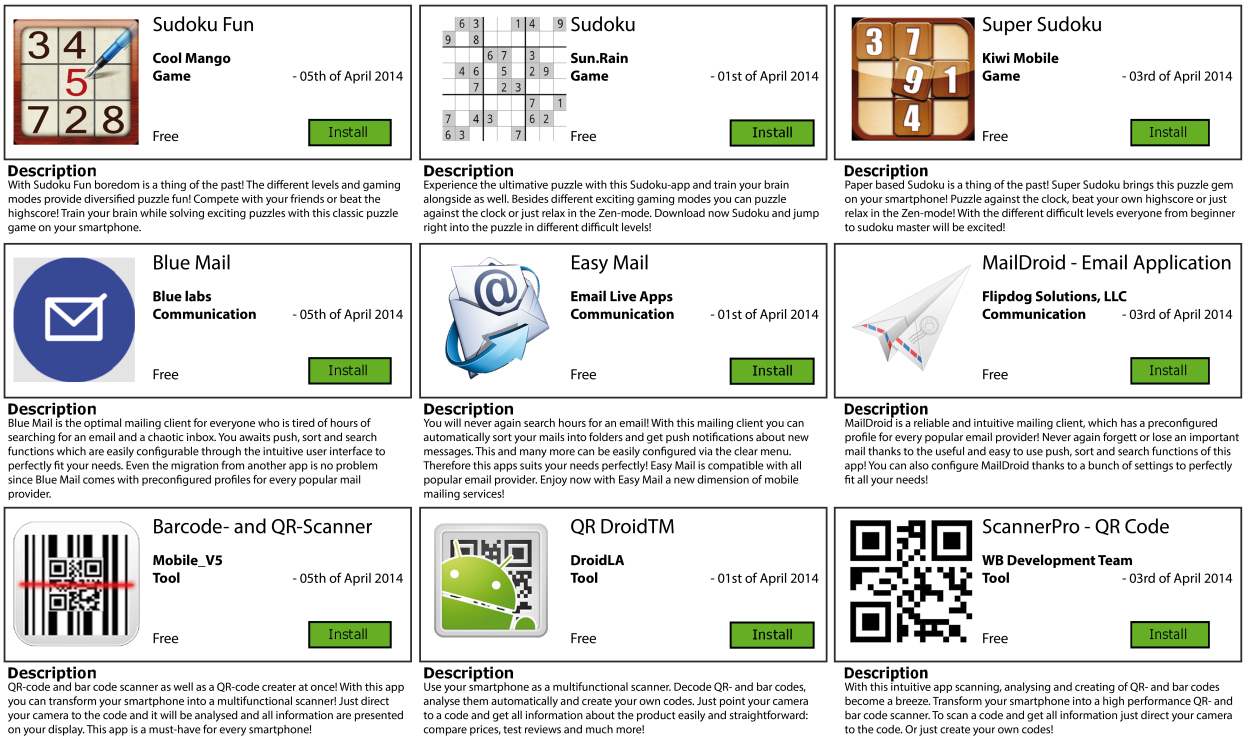

Figure 13: Overview of all used App names, logos and descriptions in the study; each of them was randomly assigned for every participant with a user rating and a permission interface

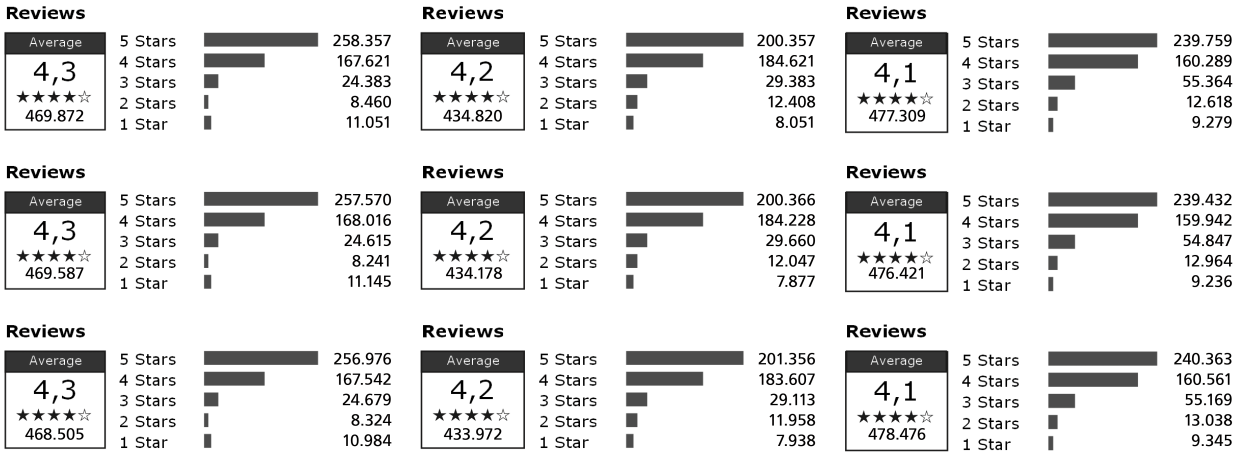

Figure 14: Overview of all used user ratings; each app alternative was randomly assigned to a user rating

Figure 15: Overview of all used permission interfaces in the decision situation "mail" with the best alternative in the first row and with ascending comprehensiveness from left to right; each participant was randomly assigned to one interface type (column) for all three decision situations

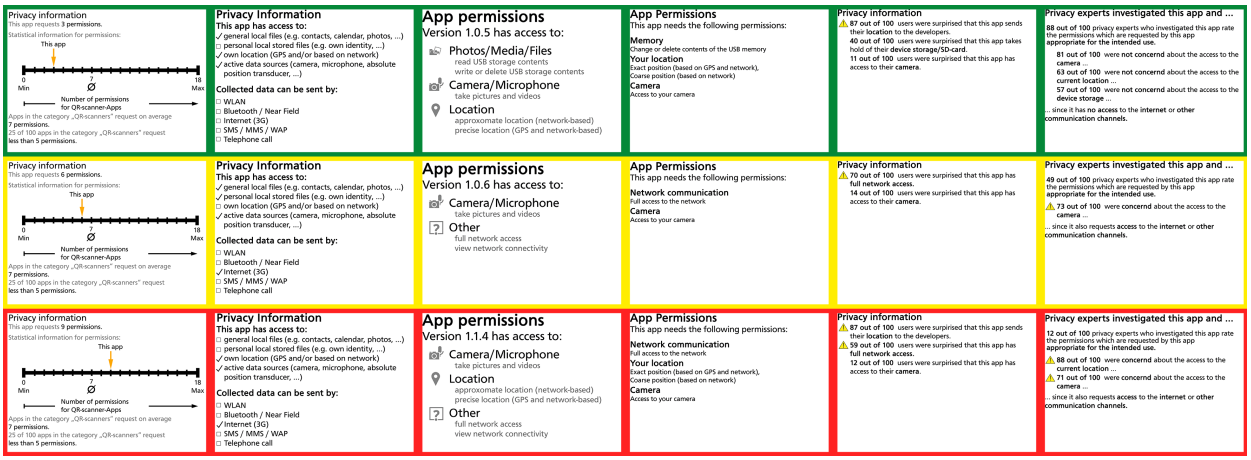

Figure 16: Overview of all used permission interfaces in the decision situation "QR-Code-Scanner" with the best alternative in the first row and with ascending comprehensiveness from left to right; each participant was randomly assigned to one interface type (column) for all three decision situations

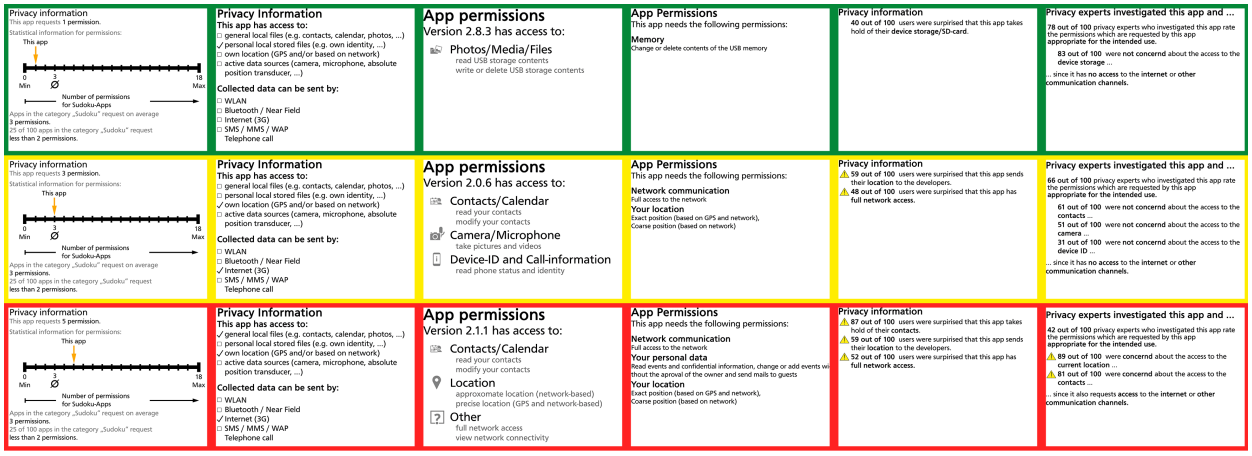

Figure 17: Overview of all used permission interfaces in the decision situation "Sudoku" with the best alternative in the first row and with ascending comprehensiveness from left to right; each participant was randomly assigned to one interface type (column) for all three decision situations