Qatrunnada Ismail*, Tousif Ahmed, Kelly Caine, Apu Kapadia, and Michael Reiter

To Permit or Not to Permit, That is the Usability Question: Crowdsourcing Mobile Apps' Privacy Permission Settings

Abstract: Millions of apps available to smartphone owners request various permissions to resources on the devices including sensitive data such as location and contact information. Disabling permissions for sensitive resources could improve privacy but can also impact the usability of apps in ways users may not be able to predict. We study an efficient approach that ascertains the impact of disabling permissions on the usability of apps through large-scale, crowdsourced user testing with the ultimate goal of making recommendations to users about which permissions can be disabled for improved privacy without sacrificing usability.

We replicate and significantly extend previous analysis that showed the promise of a crowdsourcing approach where crowd workers test and report back on various configurations of an app. Through a large, between-subjects user experiment, our work provides insight into the impact of removing permissions within and across different apps (our participants tested three apps: Facebook Messenger (N=218), Instagram (N=227), and Twitter (N=110)). We study the impact of removing various permissions within and across apps, and we discover that it is possible to increase user privacy by disabling app permissions while also maintaining app usability.

Keywords: privacy, crowdsourcing, mobile apps, permissions

DOI 10.1515/popets-2017-0041
Received 2017-02-28; revised 2017-06-01; accepted 2017-06-02.

1 Introduction

Various smartphone app marketplaces enable users to install numerous applications from various categories such as social networking, messaging, games, and utility apps. These apps usually demand access to various sensitive resources such as a device's location, camera, calendar, and address book. Access to such resources may seem unnecessary for many apps — for instance, Felt et al. found that around 33% of 940 Android apps ask to access more resources than they need [18]. PrivacyGrade [39], a research project that uses crowdsourcing to assign grades to apps based on users' expectations and actual permissions requested [32, 33], finds that apps' privacy ratings run the gamut with several popular apps receiving the lowest 'D' rating [30, 40, 43].

Access to such sensitive resources, in general, raises privacy concerns for users who may not have the ability to make informed decisions about whether granting these permissions is necessary or understand the consequences of denying permissions. Ostensibly to give users more control over privacy, smartphone operating systems enable users to control whether apps can access different sensitive resources in devices. In Android Lollipop and older, apps request various permissions to access resources, and users must accept *all* of them to be able to install the app or deny all of them and not install it. On the other hand, in iOS and in the newest versions of Android (Marshmallow and Nougat), users are prompted to grant specific permissions the first time an app attempts to use a sensitive resource. Furthermore, users can explicitly disable these permissions by accessing the system settings panel.

To help users decide which permissions to disable, in prior work we proposed a crowdsourced approach where crowd workers test different configurations of apps (with different combinations of permissions disabled) and report on their usability [24]. Based on the crowd's feedback, suitable configurations can be identified that offer higher privacy (with some sensitive permissions disabled) without sacrificing (much) usability. Furthermore, we proposed a "lattice-based" approach (see Figure 1) to efficiently explore the search space of 2^n configurations, where n is the number of permissions being explored. Our main hypothesis was that usability scores generally only decrease as the set of removed permissions grows (i.e., along paths in the lattice). Thus, if an unusable configuration is encountered, all configurations disabling a superset of those permissions can be ignored by the crowd, resulting in a more structured exploration of the search space.

Our previous work showed the promise of the lattice-based approach through a user study where we found that scores were generally decreasing when more permissions are removed, as we had hypothesized. Although our work was an important first step, our previous study was exploratory with a small number of participants (N=26), and we tested only a

***Corresponding Author: Qatrunnada Ismail:** Indiana University Bloomington, King Saud University, Riyadh, Saudi Arabia, E-mail: qismail@indiana.edu
Tousif Ahmed: Indiana University Bloomington, E-mail: touahmed@indiana.edu
Kelly Caine: Clemson University, E-mail: caine@clemson.edu
Apu Kapadia: Indiana University Bloomington, E-mail: kapadia@indiana.edu
Michael Reiter: University of North Carolina at Chapel Hill, E-mail: reiter@cs.unc.edu

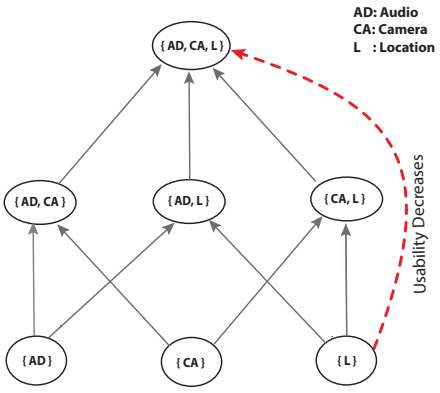


Fig. 1. In the lattice-based approach [24], each node denotes the set of permissions *removed*. Two nodes are connected if one node is a strict superset of the other. Usability scores are expected to drop along paths in the lattice, e.g., the usability of a configuration with both 'location' and 'camera' access disabled is expected to have usability equal to or worse than configurations with only 'location' or only 'camera' removed.

single app (Instagram), which limited the generalization of our findings.

In this paper, we not only report on a large scale replication of the previous work, but we also incorporate an improved methodology and conduct a more extensive analysis on the structure of the lattice across *different* apps. Whereas prior work used a within-subjects experimental design (individual participants tested multiple configurations), we use a between-subjects design (each participant tested only one configuration) across three apps in different categories: Facebook Messenger (N=218), Instagram (N=227), and Twitter (N=110). We also use established and validated usability scales (Single Ease Question [SEQ] and System Usability Scale [SUS] [47, 48]) to measure usability, whereas our prior work measured "acceptability" using an unvalidated scale. Finally, we validate the lattice-based approach for multiple apps and analyze the impact of removing permissions both within and across different apps. In our previous work, we also sought to validate the use of collaborative filtering to recommend versions based on user similarities. We do not attempt to replicate those findings in this work as it would necessitate a separate within-subjects design where participants test multiple configurations.

In summary, we sought to answer the following broad research questions (we list specific hypotheses in Section 3) through a more extensive user study:

R1: *Is there always a trade-off between usability and privacy? Specifically, can we identify more private versions of an app while maintaining the same level of usability?*
Even though there are many apps that ask for permissions that are not needed for their main functionality, and thus removing those will not adversely affect usability, we study the effect of removing permissions that are actually needed or used by the apps. We seek to not only show the existence of such situations in which removing needed permissions does not greatly affect usability but to also *find* usable versions that are more private than the original apps in terms of removed permissions.

R2: *How does disabling various sets of permissions affect the usability of an app, and at what point do apps become unusable as permissions are removed?*
It is possible to find cases in which removing only one permission causes an app to be unusable. In other cases, removing multiple other permissions does not adversely affect the usability of the app. Therefore, we seek to study the structures of how the usability of apps vary as we remove different combinations and numbers of permissions.

R3: *How is the usability of apps from different categories affected by disabling various permissions?*
Although it may seem obvious that removing the microphone permission will make voice-chatting apps useless because their main task (making calls) is based on that permission, in other apps that have several tasks (e.g, Twitter), the effect of disabling permissions is not so predictable. Therefore, we seek to compare the usability impact of removing different permissions on a variety of apps (both within and across apps).

R4: *To what extent do different permissions affect the usability of apps?*
To shed further light on R3, we would like to *quantify* the overall impact of removing individual permissions from apps, which can be useful to isolate and estimate the importance (for usability) of various permissions for an app. Such estimates can be used to improve the state of the art for recommendation interfaces to assist users in picking suitable permission configurations by explaining the privacy and usability trade-offs to users.

Our contributions. Beyond the replication component, we study the lattice structures of three different apps and show that it is possible to increase user privacy by disabling app permissions while maintaining app usability across multiple apps. We also study the *impact* of removing permissions within and across apps. Our analysis shows how we can quantify the potential usability impact of disabling certain permissions with implications for the design of interfaces that guide users to make better privacy choices while configuring permissions for apps without having to guess the usability impact of their decisions. Overall, we provide deeper insight into how a crowdsourcing strategy for exploring app configurations can be ap-

plied to perform a guided search for configurations that improves privacy without sacrificing usability.

2 Related Work

Crowdsourcing

Crowdsourcing refers to "the act of taking a job traditionally performed by a designated agent (usually an employee) and outsourcing it to an undefined, generally large group of people in the form of an open call" [23]. This technique has been used in several contexts in the literature, but we focus on its usage to improve users' privacy and security. In particular, we consider splitting the task of exploring various permission configurations among a large group of crowd workers instead of a single user for whom the task would be too onerous.

Crowdsourcing security in general. Crowdsourcing has long been explored by the security community, e.g., for spam detection where email recipients can flag emails as 'spam' [10, 62], validating certificates [42, 58], 'peer patching' where collaborating nodes can attain better security by sharing information [7, 14], and improved security and privacy through a focus on communities [15, 21]. Other examples of crowdsourcing-based systems are the Crowdroid malware-detection framework that dynamically analyzes apps' behaviors [6], and the Super-Ego framework that resides between the smartphone OS and apps and leverages crowdsourcing to decide whether to grant apps access to location [55].

Crowdsourcing and smartphone app permissions. Several recent works have employed crowdsourcing in the context of smartphone apps and privacy. Crowdsourcing was used to capture users' expectations and reactions about privacy-related behaviors and permission usage in Android apps to determine whether apps demand more permissions than warranted by most users [2, 32, 33]. Also, ProtectMyPrivacy (PMP) [1] and RecDroid [41] employ crowdsourcing to provide privacy setting recommendations to users based on the crowd's preferences. However, unlike our study, they do not examine the usability implications of the privacy preferences. Our approach is to instead focus on users' feedback through actual use to assess the impact of selectively disabling various permissions on the app's usability [24] (we significantly expand on our previous work as described in Section 1).

Permission purposes and users' perceptions

Previous studies show that many users cannot make informed decisions when it comes to granting permissions to apps because of interface issues as well as a lack of understanding of permissions in general [4, 19, 20, 27]. Kelley et al. found that more privacy information about permissions, as well as the timing of when this information is displayed, are helpful for users to make better privacy decisions [28]. Wang et al. used text-based features from apps' code and machine learning techniques to infer the purpose of the permissions requested by apps, which can be used to improve users' privacy through informed decisions [57]. Personalized Privacy Assistant (PPA) assigns privacy profiles to users based on their own privacy preferences and uses these profiles to recommend changes in permission settings [34]. Moreover, in the context of third-party apps for cloud services, Harkous et al. proposed a new permission model that informs users about what apps can infer about them based on the data that they can access [22].

In iOS and the newest Android versions (Marshmallow and Nougat), a user is asked to grant a permission the *first* time an app needs it. The "ask-on-first-use" technique in requesting permission may not be adequate because the context in which apps use permissions may change after the first permission request [59]. Instead, a better technique may be needed to infer users' permission preferences and prompt them in the cases when they may find the permission requests concerning [59]. In User-Driven Access Control (UDAC), the operating system grants permissions to apps by capturing users' intentions from their actions (such as tapping on a trusted button to take a photo). However, it is unclear how this approach applies to the background use of permissions [44]. Alternatively, we advocate an approach where users test app configurations to find permissions that can be removed while maintaining reasonable usability. Our approach can thus indicate to users the usability impacts of removing various permissions.

Binary analysis of permission usage

When disabling apps' permissions, there is a trade-off between the gain of privacy and the loss of usability. Some researchers have studied this problem by analyzing runtime behaviors of apps. For example, Kennedy et al. designed Pyandrazzi, a system that automatically evaluates the run-time effects of removing permissions from Android apps [29]. While Pyandrazzi evaluates the effects of permission removal on applications automatically, here we evaluate application usability based on actual usage by users. AppScanner uses automation to analyze and learn applications' behaviors to find privacy-related issues and then uses crowdsourcing to learn "what kinds of privacy concerns and surprises people have" [2]. Stowaway is an automated tool that detects unnecessary permissions requested by apps [18]. XManDroid analyzes communications across applications at run time and, based on specified security rules, it detects and prevents privilege escalation attacks at the application level [5].

3 Hypotheses

In this section, we describe the specific hypotheses that we seek to test in this paper.

H1: *Increasing privacy (operationalized as removing permissions) does not affect usability (operationalized as equal SEQ scores).*
Although there are many factors that can affect mobile users' privacy, in this study we focus on disabling "dangerous" permissions that are known to impact privacy [11]. So, as the number of disabled permissions increases, the amount of private information that apps can access is reduced.

H2: *The lattice relationship holds, which means that whenever an unusable version is found, as we remove more permissions and move upwards in the lattice, the usability scores (operationalized as SEQ scores) of the other versions do not increase.*
We seek to verify the lattice relationship using three different exemplar apps. The point of studying the differing lattice structures is to understand the *gradient* with which usability decreases for different categories of apps and how 'high' in the lattice we can find a version that does not significantly affect usability while (perhaps greatly) increasing privacy.

H3: *There is an interaction between the type of permission disabled (e.g., camera) and the type of app (e.g., messaging apps) on usability. Removing various types of permissions affects different kinds of apps differently.*
Although different apps need different permissions for their functionality, these permissions differ in their level of importance to the apps. We seek to quantify and study to what degree different permissions are important to different classes of apps.

4 Method

We conducted a 3 (app) x 16 (permission configuration) between-subjects experiment with participants recruited via Amazon Mechanical Turk. We tested three social-networking apps from different categories as described in Section 4.1: Facebook Messenger, Instagram, and Twitter. We generated multiple versions of each app by disabling different subsets of permissions.

Each participant was assigned a random version of one app to install in his/her Android device. The between-subjects design ensured that each participant interacted with only one version of one app. We then provided a list of tasks for each participant to perform in that version of the app (these tasks were fixed for each app category). Participants were asked to provide us with their feedback about its usability using both post-task and post-test questionnaires (i.e., the standardized SEQ and SUS scales [47, 48] described in Section 4.3). These scales measure usability as defined by the ISO, which defines usability as: "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use" [25]. The ISO definition is agnostic about the underlying reason (e.g., a bug, error, bad UI design) for poor usability. Thus, we measure usability based on users' experience and perception of the usability of the version of the app they were assigned.

4.1 App Selection

To test our hypotheses and compare the impact of removing permissions from different apps, we chose three popular social-networking apps that span three categories: messaging, photography, and microblogging. Although all three apps chosen are social networking apps, they are exemplars of very specific functionality. Primarily, Facebook Messenger is used for exchanging messages between friends, Instagram for sharing photos, and Twitter for microblogging. As discussed in more detail in Section 7, these apps are actively used by more than a billion users combined each month [52–54], allowing us to feasibly test a smaller selection of apps relevant to a large population without having to expend considerable resources that would be needed to study a larger selection of apps.

Messaging. Messaging apps enable users to exchange text messages, images, and videos. They differ from the standard text/multimedia messages (SMS/MMS) in that they can use the data plan or wifi instead of relying solely on the cellular network. In addition to avoiding per-message charges, these apps have risen in popularity for their expanded set of features such as message delivery notifications, location sharing, and the ability to create group conversations. Facebook Messenger is the second most popular messaging app after WhatsApp [51]. We chose Facebook Messenger over WhatsApp because it uses emails to identify users instead of phone numbers, which made it more convenient to conduct our experiment using test accounts.

Photography. Several photography apps enable users to take photos and/or videos, edit them, and then share them with others. The ability to enhance the images and videos in novel ways before sharing them with established social networks has made these apps popular despite the existence of default camera apps on the phone. We used Instagram, which is one of the top three photography apps across platforms [3, 13].

Microblogging. Microblogging apps enable users to quickly share brief messages with others with optionally attached content (e.g., images and videos). We chose Twitter since it is by far the most popular microblogging app [56].

4.2 Generating Versions for each App

For each app in our study, we created several different versions with different combinations of permissions disabled.

Disabling permissions. In Android apps, the permissions are listed in the manifest file. We used Apktool to unpack the apps, access the manifest files, modify the list of permissions requested by the apps, and then repackage the apps [9]. We did *not* modify any application code.

Permissions used in our experiment. The apps that we selected request various permissions in their manifest files. According to the Android Developer website, these permissions vary in their protection level (Normal vs. Dangerous) based on the risks they introduce to users' privacy [11]. We specifically focus on the "dangerous" permissions that request access to private user data such as contacts and location. Focusing on dangerous permissions allows us to maximize the potential privacy gain to users. To explore the entire lattice space, we focused on four dangerous permissions that were common to the three apps we selected. We used the permissions that request access to the following resources and data in users' devices: 1) microphone (`RECORD_AUDIO`), 2) camera (`CAMERA`), 3) location (`ACCESS_COARSE_LOCATION, ACCESS_FINE_LOCATION`), and 4) contacts (`READ_CONTACTS, WRITE_CONTACTS`). Using these four permissions, we generated $2^4 = 16$ customized versions of each app corresponding to each possible subset of permissions (including the original app where no permissions were removed). In Twitter, we used only eight of these versions, always leaving the location permission enabled, because we discovered from our pilot study that disabling the location permission cripples the app completely.

4.3 Questionnaire

After consenting to participate in our study, participants were a) asked screening questions to insure their eligibility to participate in the study; b) asked questions about their demographics and how often they performed various activities in the app; c) given a link to an app version to install on their phones; and d) given a list of tasks to perform using that version. Following each task, participants were asked to assess the ease of performing the task by answering a Single Ease Question (SEQ) [48]. At the end of all tasks, participants were asked to answer a System Usability Scale (SUS) [47]. The complete survey can be found in Appendix B.

Screening. Participants were required to be at least 18 years old, to be living in the United States, to have an Android device with Internet connectivity, and to be a current user of the test app. Since MTurkers vary in reliability, we followed the suggested practice of restricting participation to MTurkers with a 95% successful task-completion rate [38] and to exclude data from participants who failed attention-check questions for which the answers are known [31].

Demographics and activity level. We asked participants different demographic questions including their gender, age, highest level of education, and their primary racial or ethnic background. We also asked them about their activity level in the app and what features they usually use. These questions were presented before the tasks (described below) and were used to validate the choice of tasks as well as to study how much a feature's use affects usability ratings for those tasks.

Version installation. For each participant, we assigned one version of an app at random to minimize order effects. Each participant tested only one version of only one app. Since installing our customized version is a very important step to be able to complete the study, we provided technical support via email to help participants who faced problems while installing the version (about 9–10% of participants required such support, which typically involved 2–4 email exchanges).

Testing each version through tasks. For each app in our study, we determined a set of salient features related to the permissions that we varied. We created a list of all conceivable tasks that can be performed with those features. We then picked a representative set from these tasks, combined that list with some general tasks common to the apps that are not affected by the permissions (e.g., sending text messages in Facebook Messenger and liking pictures in Instagram) to create our task list. We then pilot tested the list, and our participants reported that these tasks corresponded to common use cases. Our tasks, therefore, correspond to the popular features and uses of these apps and are representative of common usage patterns for the apps we selected. Figure 2 shows a sample task in which we asked users to take a picture using the Instagram app's camera and share it. The full list of tasks can be found in Appendix B. These tasks were presented to participants in random order to minimize order effects.

Single Ease Question (SEQ). As shown in Figure 2, after each task, we asked participants whether they successfully performed that task followed by a Single Ease Question (SEQ) [48]. The SEQ is a post-task usability measure that has a single 7-point Likert scale question about task difficulty (1 is 'very difficult' and 7 is 'very easy'). It is known to be reliable and easy to respond to [46]. Following our pilot study, we modified the question to include a short description of the

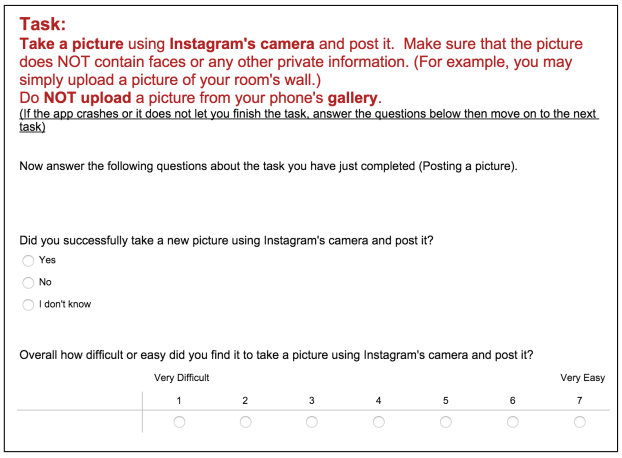


Fig. 2. A sample task.

task for clarity. For example, instead of asking "Overall how difficult or easy did you find this task?" for sharing a photo, we asked "Overall how difficult or easy did you find it to take a picture using Instagram's camera and post it?"

System Usability Scale (SUS). After all tasks, participants were asked to rate the usability of the customized app version on a System Usability Scale (SUS) [47]. The SUS is a post-test usability measure that has 10 questions, each with 5-point Likert scale ranging from 'strongly agree' to 'strongly disagree.' To avoid confusion, we modified the wording in the questions to ask about the "customized version of the app" instead of using the word "system."

Exclusion criteria and attention-check questions. We used four compliance and attention check questions. For example, when we created the different versions of the apps, we modified the name that appears in the app list to be "appname-N" where 'appname' is the original app's name, and N is the version number, which ranged between 0–15. We asked participants to report the name of the installed app to ensure that they use our customized app version and not the original one from the app store. We also added attention-check questions such as "Please answer 'Rarely' to this question." Finally, we chose to exclude data from participants who answered "I don't know" to three or more of the questions that ask whether they successfully performed a task.

4.4 Pilot Studies

We ran multiple pilot studies for each one of the apps to identify issues with the apps and survey questions. For example, the pilot studies revealed how removing the location permission cripples Twitter. Also, we re-worded our survey to indicate that it is expected for some tasks to fail, because some participants were concerned about their work being rejected if they couldn't complete some of the tasks. Moreover, pilot studies allowed us to validate that the tasks we picked are common among participants. They also helped us to validate that the survey can be finished within 30 minutes.

4.5 Study Procedures

Power analysis: Determining the sample size. To estimate the number of participants needed for each version of each app to test our hypotheses, we conducted a prospective power analysis. Assuming effect size = 0.25 (to be able detect small and medium effects), $\alpha = 0.05$, power = 0.85, and if interaction is expected (we would like to compare the effects of different removed permissions on different apps), *at least 11 participants are needed for each version*. Assuming a drop-out rate of 5 because of the exclusion criteria, we therefore aimed to recruit 16 per version, for an estimated 256, 256 and 128 users for Facebook Messenger, Instagram and Twitter respectively.

Ethical considerations. This study was approved by our institution's ethics board. To protect participants' privacy and to comply with Amazon Mechanical Turk's terms of service, we provided credentials for participants to use in the test apps instead of using their own profiles. For tasks that required participants to add location tags, we instructed them to tag any arbitrary location. For tasks that involved taking photos or videos, we instructed participants to not capture anything identifiable (e.g., by recording an image/video of a room's wall). For the tasks that required access to the contacts, because adding friends from contacts in the test apps results in uploading the contacts to their servers, we made this task optional (47% of our participants opted out of this task due to privacy concerns). Finally, our institution's general counsel approved our modification of the three apps' manifest files in this study based on a fair use analysis.

Compensation. Participants who missed at most one out of four attention-check questions were paid $3.00. Those who missed two or more were not compensated. Since our survey was designed to be finished in 30 minutes (and validated through our pilot studies), we picked this price to be commensurate with what many workers in Mechanical Turk consider to be ethical and fair ($0.10 per minute) [60].

4.6 Statistical Analysis

We verified the underlying assumptions of the parametric tests against our data. Each data point is independent; Shapiro-Wilk test result ($W=0.99$, $p>0.05$) verifies the normality assumption; and Levene's test result ($F=2.01$, $p>0.05$) verifies the homoscedasticity assumption. For the hypotheses **H1** and **H2**,

we use ANOVAs and t-tests. Whenever we find a significant difference, we calculate the effect size using Cohen's D which helps to measure the magnitude of the observed effect and provides an objective measure of its importance. For **H3**, to test the interaction between the type of app and the type of removed permission, we use factorial ANOVA where the app and removed permissions are the main effects (independent variables), and the SEQ score is the dependent variable. Where appropriate, we use the Benjamini Hochberg (BH) correction for multiple comparisons with $\alpha = 0.05$. We use BH instead of Bonferroni because the latter is more conservative and has less statistical power especially when the number of comparisons is large, which is the case in our study. Finally, to isolate the effect of removing each permission on apps' usability, we performed multiple linear regression analysis in which the SEQ score is the outcome and the permissions are the independent variables. We coded the permissions as dummy categorical variables with a value of 0 if the permission is removed, and 1 if the permission is present.

5 Findings

We now describe our findings based on our sample of 555 participants across the three apps.

5.1 Participants and Demographics

We ran our study between December 2015 and February 2016 with a total of 633 participants enrolled (260 for Facebook Messenger, 243 for Instagram, and 130 for Twitter). After applying our exclusion criteria (see Section 4.3), we ended up with a total of N=555 participants (218 for Facebook Messenger, 227 for Instagram, and 110 for Twitter). Each version was tested by 13–18 participants, thus meeting the 11-participant threshold established by the power analysis (see Section 4.5).

Our final sample consisted of 239 women (43.06%) and 309 men (55.68%). Their ages ranged from 18–65 years, although most were between 18–49 (97.29%) years old. Most had either finished high school (41.80%) or received an undergraduate degree (48.64%). All participants used Android 5 (Lollipop) or older. At the time of our study, Android 6 (Marshmallow) was not yet widely adopted as we ran our study soon after its announcement in October 2015. The majority of participants were familiar with the Android platform and were frequent users of the apps they were assigned. Most participants had used an Android device for more than three years and reported using their assigned app at least once a week. Among all participants, 52.79% reported checking permissions requested by an app frequently ("Almost every time" or "Every time") before installing an app. Most (70%) reported stopping an installation of an app due to the requested permissions at least once. Since we asked participants to temporarily change their security settings to be able to install our customized app, there is a possibility that people who were more concerned about their privacy dropped out of the study before installing the app, which may have resulted in a bias in our sample. Still, as indicated above, our sample included a considerable fraction of privacy-conscious participants.

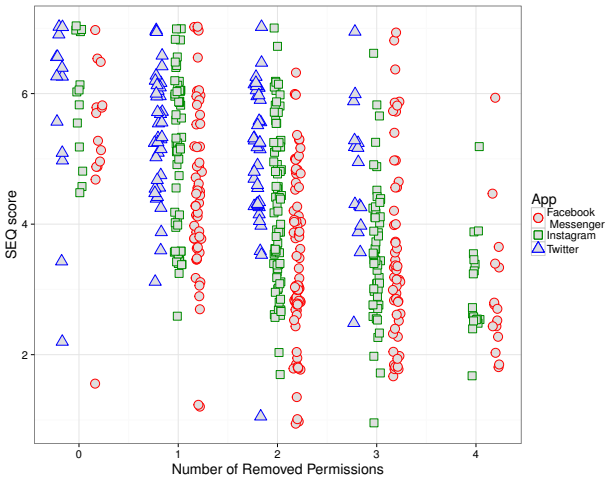


Fig. 3. A scatter plot of the SEQ scores for the three apps.

5.2 Usability Measures

As mentioned in Section 4.3, we assessed usability from users' perception with two measures, the post-task SEQ and post-test SUS. The SEQ and SUS scales are known to be correlated [49] and were correlated in our results (r=0.645, p<0.001). Since we are interested in measuring the impact of disabling specific permissions and collected the SEQ associated with each permission removed, we will focus on the SEQ as the usability measure for our analysis.

5.3 The Number of Removed Permissions and Usability

We begin our analysis by first considering if and how the number of permissions removed impacts usability. For each version of each app, we generated a usability score by calculating the mean value of the SEQ scores across tasks. Using this measure, we find a relationship between usability and the number of removed permissions. Our results indicate that the difference in usability as we remove more permissions is statis-

No. of Removed permissions	Facebook Messenger	Instagram	Twitter
0 – 1	0.37	$0.01*_{L}$	0.54
0 – 2	$0.01*_{L}$	$< 0.001^{***}{}_{VL}$	0.20
0 – 3	$0.01*_{L}$	$< 0.001^{***}{}_{VL}$	0.14
0 – 4	$0.01*_{VL}$	$< 0.001^{***}{}_{VL}$	
1 – 2	$< 0.001^{***}{}_{L}$	$< 0.001^{***}{}_{M}$	0.08
1 – 3	$< 0.001^{***}{}_{L}$	$< 0.001^{***}{}_{VL}$	0.08
1 – 4	$< 0.001^{***}{}_{VL}$	$< 0.001^{***}{}_{VL}$	
2 – 3	0.82	$< 0.001^{***}{}_{M}$	0.54
2 – 4	0.09	$< 0.001^{***}{}_{VL}$	
3 – 4	0.09	$0.02*_{M}$	

Statistical significance: *** p <0.001, ** p <0.01, * p <0.05
Effect size: M: Medium, L: Large, VL: Very Large

Table 1. Adjusted p-values along with the effect sizes for pairwise t-test for all combinations of the number of removed permissions.

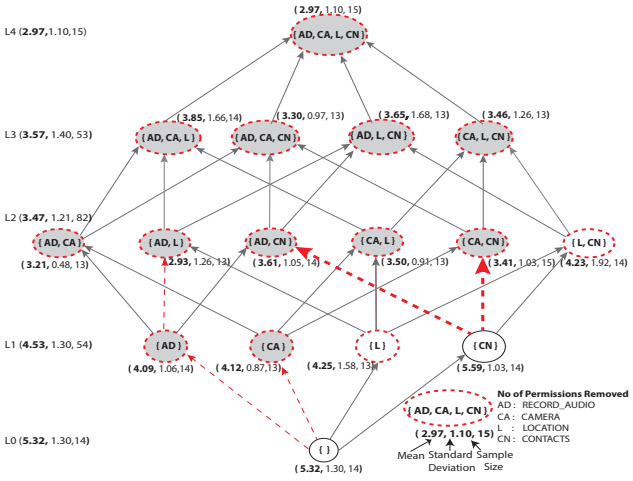
tically significant in the three apps; the ANOVA results are as follows: Facebook Messenger ($F(4)=11.9$, $p<0.001$), Instagram ($F(4)=25.5$, $p<0.001$), Twitter ($F(3)=2.8$, $p=0.04$). Figure 3 shows a scatter plot of the SEQ scores.

To investigate differences at each level, we conducted post-hoc t-tests. The results for the post-hoc t-tests (with BH correction) to find which values differ from each other are shown in Table 1. The pairs in the first column are the number of removed permissions. For instance, "1 – 2" refers to the difference in usability scores for when 1 vs. 2 permissions are removed. We also indicate the effect sizes based on Cohen's D[1] (the exact values are provided in Table 5 in Appendix A).

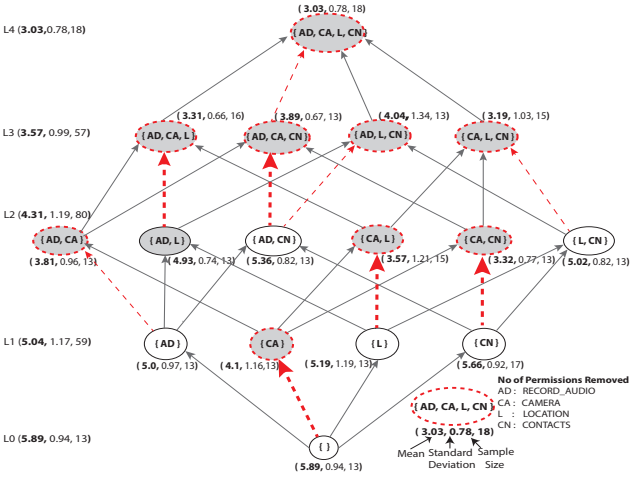
Based on this analysis, we find that it is possible to remove permissions while maintaining the same level of usability. Table 1 shows multiple instances where there was no difference in usability between versions of apps that had permissions removed, which provides partial support for **H1**. For example, there was no difference in usability in the version of Facebook Messenger or Twitter that had 0–1 permission removed. Although in some cases increasing privacy (by removing permissions) does negatively affect usability, in other cases it does not. This suggests that there are indeed conditions where there is no privacy-usability trade-off and that apps can be designed to maintain usability while providing a high degree of privacy.

5.4 Usability Scores in the Lattice

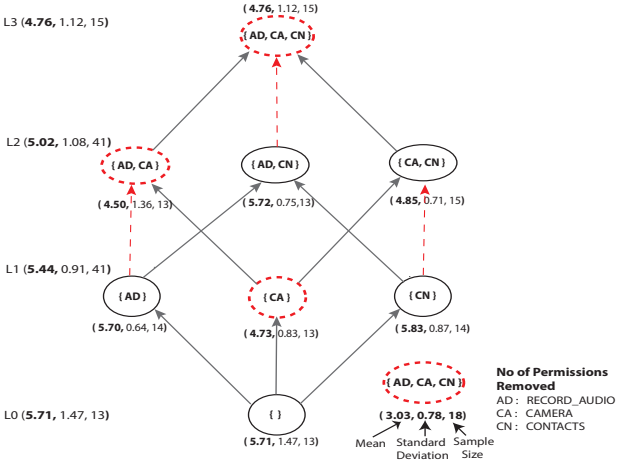
Within each app, we examined the usability differences between the versions that we created. Our ANOVA results indicate that, for each app, there is a significant difference in usability scores for different versions: Facebook Messenger ($F(15)=4.9$, $p<0.001$), Instagram ($F(15)=16.3$, $p<0.001$), and Twitter ($F(7)=4.2$, $p<0.001$). We performed further analysis



(a). Facebook Messenger



(b). Instagram



(c). Twitter

Fig. 4. The lattice structures for the tested apps. The dashed red arrows indicate significant differences in the usability scores between the connected nodes where the arrow thickness represents the effect size (large or very large). The nodes with dashed red borders indicate less usable versions due to below-average usability, and the shaded nodes represent significant drops in usability compared to the original app.

1 We use the following thresholds to interpret the effect size: 0.20: small, 0.50: medium , 0.80: large, and 1.30: very large [16]

of the usability scores using three approaches. First, to determine version-specific differences, in Section 5.4.1, we tested the differences in usability between each pair of versions that have subset relationships, i.e., connected with edges in the lattice structure. In Section 5.4.2, we used a threshold based on average scores for the SEQ scale in general. We labeled a version to be 'less usable' if its score is below the average, and 'more usable' if its score is equal to or more than the average. Finally, in Section 5.4.3, we compared the usability scores of different versions of each app with the original, unmodified version as a baseline.

5.4.1 How Usability Changes through the Lattice

For each arrow in the lattice, where each version is represented as a node, we performed t-tests (with BH correction) to check whether the difference between scores is significant. As shown in Figure 4, for each significant result, we indicate the effect size using Cohen's D (the exact values are provided in Table 6 in Appendix A) — **the dashed red arrows** indicate significant differences in usability scores between the connected versions. The effect sizes that we found are 'large' and 'very large' and are shown by thin and thick dashed red arrows respectively. We can see in the three figures that whenever there is a significant difference between nodes in the lattice, the usability scores are decreasing, which supports our hypothesis **H2** since all non-significant differences are expected to have a small effect size based on the statistical power of our experiment.

Moreover, the first four rows of Table 1 show that for Instagram and Facebook Messenger, whenever the usability drops significantly, removing more permissions also results in significant drops in usability. The lattice relationship [24] holds, and we accept our hypothesis **H2**.

5.4.2 Versions with Below-average Usability

For the SEQ, it is recommended to consider anything below 4.8 as 'below average' [36, 48]. In Figure 4, the **nodes with dashed red borders** indicate below-average usability and thus are labled as 'less usable'. We can see that whenever a node is less usable due to below-average usability, all of its successors in the upper levels are also less usable (with only one exception in Figure 4c). This validates the pruning strategy proposed in our prior study [24] for efficiently exploring the lattice and provides further support for **H2**. That is, whenever we find a less usable node, we stop exploring its successors above it in the lattice because they are also expected to be less usable.

Removed Permissions	Facebook Messenger	Instagram	Twitter
{AD}	0.01^{*}_{L}	0.06	0.98
{CA}	0.01^{*}_{L}	$<0.001^{***}_{VL}$	0.12
{L}	0.07	0.10	
{CN}	0.54	0.51	0.98
{AD,CA}	$<0.001^{***}_{VL}$	$<0.001^{***}_{VL}$	0.12
{AD,L}	$<0.001^{***}_{VL}$	0.01^{*}_{L}	
{AD,CN}	0.001^{**}_{VL}	0.15	0.98
{CA,L}	$<0.001^{***}_{VL}$	$<0.001^{***}_{VL}$	
{CA,CN}	$<0.001^{***}_{VL}$	$<0.001^{***}_{VL}$	0.12
{L,CN}	0.07	0.06	
{AD,CA,L}	0.01^{*}_{L}	$<0.001^{***}_{VL}$	
{AD,CA,CN}	$<0.001^{***}_{VL}$	$<0.001^{***}_{VL}$	0.12
{AD,L,CN}	0.01^{*}_{L}	$<0.001^{***}_{VL}$	
{CA,L,CN}	0.001^{**}_{VL}	$<0.001^{***}_{VL}$	
{AD,CA,L,CN}	$<0.001^{***}_{VL}$	$<0.001^{***}_{VL}$	

Statistical significance: *** p<0.001, ** p<0.01, * p<0.05
AD: Audio, CA: Camera, L: Location, CN: Contacts
Effect size: M: Medium, L: Large, VL: Very Large

Table 2. The adjusted p-values along with the effect sizes for the t-tests between the baseline (the original app without any removed permissions) and all other versions.

5.4.3 Usability Compared to the Original App

For each app, we used the original version as a baseline and performed t-tests (with BH correction) to compare its usability score with all other versions. This comparison detects significant drops in usability compared to the baseline. **The shaded nodes** in Figure 4 represent 'large' or 'very large' drops in usability (Cohen's D values are shown in Table 4 in Appendix A). As shown in Table 2:

Facebook Messenger. All of the versions have significantly lower usability compared to the original app except {L}, {CN} and {L,CN}. This means that usability does not have to be traded off for privacy in these cases which provides further support for **H1**.

Instagram. All versions have significantly lower usability compared to the original app except {AD}, {L}, {CN}, {AD,CN} and {L,CN}. Removing these permissions does not have a significant effect on the usability of the app again supporting **H1**.

Twitter. None of the versions have a significant difference in usability compared to the original app. This means that in Twitter, removing any of the three permissions (camera, audio, and contacts) does not introduce a trade-off between privacy and usability, thus supporting **H1**.

5.5 Impact of Removing Permissions across Apps

To study the impact of removing permissions on the usability of different apps, we tested the interaction between the app and the removed permissions using the following model:

$$\text{SEQ} \sim \text{removed permissions} + \text{app} + \text{app} * \text{removed permissions}$$

Removed Permissions	DF	F value	Adjusted p-value
{ }	2	0.97	0.50
{AD}	2	7.63	0.002**
{CA}	2	1.41	0.39
{L}	1	5.21	0.04*
{CN}	2	0.18	0.88
{AD,CA}	2	4.52	0.03*
{AD,L}	1	21.74	<0.001***
{AD,CN}	2	14.48	<0.001***
{CA,L}	1	0.03	0.88
{CA,CN}	2	9.09	<0.001***
{L,CN}	1	4.48	0.06
{AD,CA,L}	1	1.86	0.30
{AD,CA,CN}	2	6.31	0.006**
{AD,L,CN}	1	0.83	0.51
{CA,L,CN}	1	0.43	0.63
{AD,CA,L,CN}	1	0.02	0.89

Statistical significance: *** p <0.001, ** p <0.01, * p <0.05
AD: Audio, CA: Camera, L: Location, CN: Contacts

Table 3. The adjusted p-values for the effect of app type on each combination of removed permissions.

The factorial ANOVA showed significant results for the main effects: removed permissions ($F(15, 515)=15.49$, $p<0.001$), and app ($F(2, 515)=34.50$, $p<0.001$). It also revealed that there is a significant interaction ($F(22, 515)=2.5$, $p<0.001$). These results show a significant effect of removing some permissions on the usability scores and each app is impacted differently; so we accept hypothesis **H3**.

Simple Effects Analysis. Since the interaction was significant, we performed a follow-up simple effects analysis which helps to look at the effect of one independent variable (app) at each level of the other independent variable (removed permissions). Our results (shown in Table 3) suggest that removing the following permission combinations has a significant different impact of usability scores on the three apps: {AD}, {L}, {AD,CA}, {AD,L}, {AD,CN}, {CA,CN}, {AD,CA,CN}. In particular, removing the camera or contacts permissions affects all apps relatively uniformly, whereas removing the audio or location permissions affects the apps differently.

5.6 Quantifying the Impact of Permissions within Apps

As shown in the following equation, we use multiple linear regression analysis to isolate and quantify (in SEQ units) the effect of removing each permission on apps' usability:

$$SEQ = \beta_0 + \beta_1 AD + \beta_2 CA + \beta_3 L + \beta_4 CN$$

β_0 yields the baseline mean of an app's usability (in SEQ units) with all permissions removed (and quantifies other factors that affect usability other than the four permissions). Each subsequent β_n value indicates the expected *increase* in usability by adding the corresponding permission.

Facebook Messenger. The usability coefficients are:

$$SEQ = 3.05 + 0.89AD + 0.71CA + 0.48L + 0.21CN$$

The baseline mean of Facebook Messenger's usability (in SEQ units) with all permissions removed is $\beta_0 = 3.05$ ($p < 0.001$). The usability of Facebook Messenger is predicted to increase by $\beta_1 = 0.89$ ($p < 0.001$) when the Audio permission is added. For the Camera permission, the expected usability increase is $\beta_2 = 0.71$ ($p < 0.001$) and for the Location permission $\beta_3 = 0.48$ ($p = 0.01$). The Contacts permission did not yield a significant result. The overall model fit is $R^2 = 0.17$. Our results indicate that the Audio and Camera permissions have the highest impact on usability for Facebook Messenger, which supports our findings in the previous subsections.

Instagram. For this app, the usability coefficients are:

$$SEQ = 3.17 + 0.29AD + 1.74CA + 0.62L + 0.33CN$$

The baseline mean of Instagram's usability (in SEQ units) with all permissions removed is $\beta_0 = 3.17$ ($p < 0.001$). The usability of Instagram is predicted to increase by $\beta_1 = 0.29$ ($p = 0.02$) when the Audio permission is added. For the Camera permission, the expected usability increase is $\beta_2 = 1.74$ ($p < 0.001$); Location permission $\beta_3 = 0.62$ ($p < 0.001$); and Contacts permission $\beta_4 = 0.33$ ($p = 0.01$). The overall model fit is $R^2 = 0.51$. Our results indicate the Camera permission has the highest impact on usability for Instagram, which supports our findings in the previous subsections.

Twitter. For this app, the usability coefficients are:

$$SEQ = 4.99 + 0.08AD + 1.04CA - 0.14CN$$

The baseline mean of Twitter's usability (in SEQ units) with all permissions removed is $\beta_0 = 4.99$ ($p < 0.001$). Although both the Audio and Contacts permissions did not yield significant results, the usability of Twitter is predicted to increase by $\beta_2 = 1.04$ ($p < 0.001$) in SEQ units when the Camera permission is added. The overall model fit is $R^2 = 0.25$. Our results indicate the Camera permission has the highest impact on usability for Twitter, which supports our findings in the previous subsections. We note that in Twitter, when users try to find friends from contacts, if the version has the contacts permission, Twitter displays suggestions based on the user contacts. If the version does not have the contacts permission, Twitter displays suggestions based on the user's followers and who follows them. This may explain why usability is relatively unaffected by removing the contacts permission.

6 Discussion and Implications

There are three primary contributions of this work: 1) we discover that it is possible to increase user privacy by disabling app permissions while maintaining app usability for the tested apps; 2) we confirm that the lattice-based approach efficiently identifies less usable paths, allowing us to limit the number of app versions that require testing; 3) we find that removing certain types of permissions impacts the usability of various apps differently. We also study and quantify how permissions impact apps' usability and find that removing certain permissions (e.g., access to the camera) disproportionately impacts the usability of apps compared to removing other permissions (e.g., access to contacts). Our findings have several implications as discussed next.

Privacy and usability not always in tension

One of the primary goals of this work was to determine whether we could enhance user privacy (by disabling some permissions) without degrading app usability. For each of the three apps tested, we found at least one configuration where removing one or more permissions resulted in an equivalent level of usability. In other words, we identified specific instances where an explicit privacy-usability trade-off was absent. Although previous works proposed solutions to improve the trade-off between privacy and usability in different contexts [12, 17, 50, 61], we demonstrate cases in which we increase privacy without trading off usability. We even find that, in the case of Twitter, it is possible for the app to maintain a high degree of usability even when three sensitive permissions are disabled. Our findings have important implications for both theory and practice of privacy-enhancing technologies: if there does not always have to be a privacy-usability trade-off, then designers can strive to find ways to increase user privacy without sacrificing usability.

Lattice based approach for crowd management

Our prior work proposed a lattice-based approach as a crowd management strategy to explore the state space of permissions efficiently [24]. This work demonstrated that by exploring configurations with only one permission removed, one can identify any unusable versions and use these to prune the search space by ignoring all descendants of that version in the lattice. We replicate and extend prior work by presenting evidence about the efficacy of a lattice- and crowd-based management approach across multiple apps.

We found that there are cases in which removing only one permission causes an app to be unusable, whereas in others, removing multiple permissions does not affect the usability of the app. The point of studying the differing lattice structures is to understand the *gradient* with which usability decreases (even if we know that it eventually drops), and how 'high' in the lattice can we find a version that has not significantly affected usability while (perhaps greatly) increasing privacy.

We analyzed the usability scores in the lattice using three different approaches presented in Section 5.4. They all show that we can find cases in which we remove permissions without introducing a strict trade-off between usability and privacy, and they confirm that the lattice-based approach efficiently identifies less usable paths. However, there are differences between the results of the three approaches. For instance, in Twitter, removing the camera permission results in a usability score that is below average using the threshold-based approach, but the usability drop is not significant compared to the baseline (original app). Further experimentation, as future work, is needed to compare these approaches to determine how best to prune lattice paths during crowd based exploration.

Understanding how disabling permissions affect apps

Another contribution of this work is that we show to what extent various permissions affect apps. Our scope is larger than just understanding the effect of removing individual permissions. We seek to determine how removing *combinations* of permissions affects an app's usability. We found that the types of permissions disabled affect each app differently (e.g., disabling the camera permission affects Instagram much more than disabling the location permission). Moreover, although it is trivial to predict the impact of removing some permissions from some apps, there are situations in which the effect of disabling permissions is not so predictable. For example, for a Flashlight app, it may seem obvious that disabling the camera permission will make it useless because it blocks the app from accessing the flash, but it is not trivial to know that disabling the location or contacts permissions makes it unusable even though these are not connected to its main functionality.

Using linear regression, we quantified the impact of removing permissions in terms of usability units. To help users decide which permissions to disable while maintaining an apps' usability, the impact scores – through such linear regressions – can be integrated with systems that augment permission interfaces with recommendations [1, 34]. These scores could also be used to generate recommendations based on a usability threshold set by users.

Also, since the newest Android versions (Marshmallow and Nougat) give users the ability to control what permissions to enable or disable, developers will be challenged to make apps robust when different combinations of permissions are disabled. Therefore, our study can be useful to developers who

wish to infer the usability of an app under different combinations of disabled permissions.

The current permission model, however, does not separate between use for core functionality vs. other uses. Users may indeed disable permissions because they don't know what *else* the app does with the information, even if it affects core functionality. Our work highlights the usability impact of disabling such permissions and demonstrates some apps handle this more gracefully than others. However, much work remains to be done to automatically determine contextually appropriate uses of permissions and that direction is promising.

7 Limitations

Number of tested apps. Our study is based on only three apps, albeit carefully chosen ones. We chose to limit the apps in this study because our goal was to test the usefulness of this approach, and it would take an incredible number of resources to test the entire library of phone apps (and would be ill-advised before knowing whether the approach worked). To simultaneously limit our study and still impact as large a number of users as possible, we decided to study popular, exemplar apps actively used by more than a billion users combined (1.2 billion active monthly users for Facebook Messenger, 700 million active monthly users for Instagram, and 328 million active monthly users for Twitter) [52–54]. Choosing exemplars from each category allowed us to compare how lattice structures/gradients might be structured within categories and how they differ across categories. Despite these strengths, we advise caution when generalizing our findings to other apps. In particular, the three apps we used in our study are popular apps in the overarching 'social networking' category, and thus, our findings are most applicable to that category. We encourage future work on studying an extensive set of apps, and we would like to see a more extensive analysis of inter-app differences through experimentation with numerous apps. Finally, we note that even if we made an extremely limiting assumption that our results could only be applied to these three apps, the results would still apply to billions of existing users.

The list of tested tasks. In this work, we chose a list of popular and common tasks associated with each app (validated through the pilot study and eventually through the experiment). This set of tasks was picked for being representative and so that participants could test apps in a reasonable time period. The usability scores are representative only of those sets of popular tasks, and the scores may differ for other tasks. Nevertheless, since all versions for a given app were tested with the same set of app-specific tasks, the relative differences still yield meaningful findings for popular uses of those apps. As future work, it would be interesting to compare the results that we got from our current controlled experiment using fixed tasks with having participants use the apps as they normally would for a reasonable amount of time. Thus, more work is needed to study the impact of permission removal on different users based on their individual usage patterns (our previous study shed light on this aspect but would need a different experimental design than we perform in this study).

Study population. Since we are exploring crowdsourcing (and Amazon Mechanical Turk is a crowdsourcing platform), our sample was picked from crowd workers ("MTurkers"). We restricted participation to U.S.-only MTurkers. Although there is evidence that data collected from Amazon Mechanic Turk is reflective of real-world behavior in different contexts [8, 37], some researchers have shown how the MTurk population differs from the U.S. population: the professional survey takers [45] are younger, better educated, more savvy with social media, and more privacy conscious [26]. We carefully designed our study to remove privacy biases by making tasks uniform for all MTurkers and assessing only usability. Moreover, while the quality of results from tests on MTurk are not as good as lab-based testing, Liu et al. [35] found that MTurk was a viable platform for usability testing. A general concern with using platforms such as MTurk is gaming by the workers. Peer et al. found that using higher reputation workers (with 95% approval rating) or attention-check questions improved the quality of data [38]. In our pilot testing, we found that higher reputation workers were still failing attention check questions, which led us to use both approaches.

8 Conclusions

We analyzed a "lattice based" crowdsourcing strategy for identifying which permissions can be disabled for apps while still maintaining their usability. Through an experiment-based user study with 555 participants, we shed light on how permissions are structured within and across apps to better guide a crowdsourced exploration of app configurations. Building on our previous work, we not only replicate previous findings through a larger sample size, validated measures, and multiple apps, but we also determine to what extent removing various permissions impact different apps. Our findings have implications for the design of crowdsourcing strategies in discovering privacy enhanced configurations of mobile apps and making privacy recommendations to users when choosing and installing apps. Our techniques can be used to predict the usability impact of disabling permissions and provide feedback to users on how their decisions to limit permissions may impact the user experience.

Unlike most other approaches, our work addresses the *impact on usability* of privacy decisions, which requires human involvement in contrast to automated approaches that analyze execution of apps. We believe we have made a significant step forward in understanding the application of crowdsourcing with efficient crowd management to the privacy and usability of mobile apps. We advocate further research to study more categories of apps as well as larger sets of permissions. Finally, although our approach can be easily integrated into existing app stores where, for example, user ratings are collected along with configurations selected by the users, further research into incentivizing the crowd is necessary for a more structured exploration of configurations.

9 Acknowledgments

This material is based upon work supported in part by the National Science Foundation under grants 1228364, 1228471, and 1252697. We thank Henry Akaeze, Serigne Sene, and Jeremiah Stevens for their help in creating our surveys; and JangDong Seo for his assistance with the statistical analysis.

References

[1] Yuvraj Agarwal and Malcolm Hall. 2013. ProtectMyPrivacy: detecting and mitigating privacy leaks on iOS devices using crowdsourcing. In *Proceeding of the 11th annual international conference on Mobile systems, applications, and services*. ACM, 97–110.

[2] Shahriyar Amini, Jialiu Lin, Jason I Hong, Janne Lindqvist, and Joy Zhang. 2013. Mobile Application Evaluation Using Automation and Crowdsourcing. In *Workshop on Privacy Enhancing Tools*.

[3] Karissa Bell. 2015. The 7 best iPhone photography apps of all time. (2015). **http://mashable.com/2015/12/13/best-iphone-photo-apps-of-all-time/#NruakC5LTuqF**.

[4] K. Benton, L. J. Camp, and V. Garg. 2013. Studying the effectiveness of Android application permissions requests. In *Security and Social Networking (SESOC), 2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops)*. 291–296.

[5] Sven Bugiel, Lucas Davi, Alexandra Dmitrienko, Thomas Fischer, Ahmad-Reza Sadeghi, and Bhargava Shastry. 2012. Towards Taming Privilege-Escalation Attacks on Android. In *19th Network & Distributed System Security Symposium*.

[6] Iker Burguera, Urko Zurutuza, and Simin Nadjm-Tehrani. 2011. Crowdroid: behavior-based malware detection system for android. In *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices*. ACM, 15–26.

[7] L.Jean Camp and Allan Friedman. 2004. Peer Patching – Rapid Response in Distributed Systems. In *24th Army Science Conference*.

[8] Scott Clifford, Ryan M Jewell, and Philip D Waggoner. 2015. Are samples drawn from Mechanical Turk valid for research on political ideology? *Research & Politics* 2, 4 (2015), 2053168015622072.

[9] Ryszard Wiśniewski Connor Tumbleson. 2015. A tool for reverse engineering Android apk files. (2015). **https://ibotpeaches.github.io/Apktool/**.

[10] E. Damiani, S. De Capitani Di Vimercati, S. Paraboschi, and P. Samarati. 2004. P2P-Based Collaborative Spam Detection and Filtering. In *Fourth IEEE Conference on P2P*. **http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1334945**

[11] Developers. 2016. Requesting Permissions. (2016). **https://developer.android.com/guide/topics/permissions/requesting.html**.

[12] Irit Dinur and Kobbi Nissim. 2003. Revealing information while preserving privacy. In *Proceedings of the twenty-second ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*. ACM, 202–210.

[13] Alex Dobie. 2015. The best photography apps for Android. (2015). **http://www.androidcentral.com/best-photography-apps-android**.

[14] Zheng Dong and L Jean Camp. 2012. PeerSec: Towards Peer Production and Crowdsourcing for Enhanced Security.. In *HotSec*.

[15] Zheng Dong, Vaibhav Garg, Jean Camp, and Apu Kapadia. 2012. Pools, Clubs and Security: Designing for a Party Not a Person. In *Proceedings of The New Security Paradigms Workshop (NSPW)*. 77–86. DOI:**http://dx.doi.org/10.1145/2413296.2413304**

[16] Paul D. Ellis. 2009. Thresholds for Interpreting Effect Sizes. (2009). **http://www.polyu.edu.hk/mm/effectsizefaqs/thresholds_for_interpreting_effect_sizes2.html**.

[17] Kassem Fawaz and Kang G Shin. 2014. Location privacy protection for smartphone users. In *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 239–250.

[18] Adrienne Porter Felt, Erika Chin, Steve Hanna, Dawn Song, and David Wagner. 2011a. Android permissions demystified. In *18th ACM Conference on Computer and Communications Security*. 627–638. DOI:**http://dx.doi.org/10.1145/2046707.2046779**

[19] Adrienne Porter Felt, Kate Greenwood, and David Wagner. 2011b. The effectiveness of application permissions. In *2nd USENIX Conference on Web Application Development*. 75–86.

[20] Adrienne Porter Felt, Elizabeth Ha, Serge Egelman, Ariel Haney, Erika Chin, and David Wagner. 2012. Android Permissions: User Attention, Comprehension, and Behavior. In *8th Symposium on Usable Privacy and Security*. Article 3, 14 pages. DOI:**http://dx.doi.org/10.1145/2335356.2335360**

[21] Vaibhav Garg, Sameer Patil, Apu Kapadia, and L. Jean Camp. 2013. Peer-produced Privacy Protection. In *IEEE International Symposium on Technology and Society (ISTAS)*. 147–154. DOI:**http://dx.doi.org/10.1109/ISTAS.2013.6613114**

[22] Hamza Harkous, Rameez Rahman, Bojan Karlas, and Karl Aberer. 2016. The Curious Case of the PDF Converter that Likes Mozart: Dissecting and Mitigating the Privacy Risk of Personal Cloud Apps. *arXiv preprint arXiv:1608.05661* (2016).

[23] Jeff Howe. 2006. Crowdsourcing: A Definition. (2006). **http://crowdsourcing.typepad.com/cs/2006/06/crowdsourcing_**

a.html.

[24] Qatrunnada Ismail, Tousif Ahmed, Apu Kapadia, and Michael K Reiter. 2015. Crowdsourced exploration of security configurations. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*. ACM, 467–476.

[25] ISO. 2016. Usability of consumer products and products for public use. (2016). https://www.iso.org/obp/ui/#iso:std:iso:ts:20282:-2:ed-2:v1:en.

[26] Ruogu Kang, Stephanie Brown, Laura Dabbish, and Sara Kiesler. 2014. Privacy Attitudes of Mechanical Turk Workers and the U.S. Public. In *Symposium On Usable Privacy and Security (SOUPS '14)*. 37–49.

[27] Patrick Gage Kelley, Sunny Consolvo, Lorrie Faith Cranor, Jaeyeon Jung, Norman Sadeh, and David Wetherall. 2012. A conundrum of permissions: Installing applications on an Android smartphone. In *Financial Cryptography and Data Security*. Springer, 68–79.

[28] Patrick Gage Kelley, Lorrie Faith Cranor, and Norman Sadeh. 2013. Privacy as part of the app decision-making process. In *2013 ACM Conference on Human Factors in Computing Systems*. 3393–3402.

[29] Kristen Kennedy, Eric Gustafson, and Hao Chen. 2013. Quantifying the Effects of Removing Permissions from Android Applications. In *IEEE Mobile Security Technologies*.

[30] Kim Komando. 2015. These 7 apps are among the worst at protecting privacy. (2015). http://www.usatoday.com/story/tech/columnist/komando/2015/09/18/apps-protecting-privacy/32563419/.

[31] Robert Kosara and Caroline Ziemkiewicz. 2010. Do Mechanical Turks dream of square pie charts?. In *Proceedings of the 3rd BELIV'10 Workshop: Beyond time and errors: novel evaluation methods for Information Visualization*. ACM, 63–70.

[32] Jialiu Lin, Bin Liu, Norman Sadeh, and Jason I Hong. 2014. Modeling users' mobile app privacy preferences: Restoring usability in a sea of permission settings. In *Symposium On Usable Privacy and Security (SOUPS 2014)*. 199–212.

[33] Jialiu Lin, Norman Sadeh, Shahriyar Amini, Janne Lindqvist, Jason I Hong, and Joy Zhang. 2012. Expectation and purpose: Understanding users' mental models of mobile app privacy through crowdsourcing. In *2012 ACM Conference on Ubiquitous Computing*. 501–510.

[34] Bin Liu, Mads Schaarup Andersen, Florian Schaub, Hazim Almuhimedi, Shikun Aerin Zhang, Norman Sadeh, Yuvraj Agarwal, and Alessandro Acquisti. 2016. Follow My Recommendations: A Personalized Assistant for Mobile App Permissions. In *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*.

[35] Di Liu, Randolph G. Bias, Matthew Lease, and Rebecca Kuipers. 2012. Crowdsourcing for usability testing. *Proceedings of the American Society for Information Science and Technology* 49, 1 (2012), 1–10. DOI:http://dx.doi.org/10.1002/meet.14504901100

[36] Craig M. MacDonald, Sean Fitzell, Megan Koontz, Kate Merlie, Alana Miller, Samantha Raddatz, Tal Rozen, April Siqueiros, and Susan Young. 2013. Usability Report. (2013). http://www.craigmacdonald.com/wp-content/uploads/2013/08/CCPS-Usability-Report-Summer-2013.pdf.

[37] Michelle L Mazurek, Saranga Komanduri, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Patrick Gage Kelley, Richard Shay, and Blase Ur. 2013. Measuring password guessability for an entire university. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security*. ACM, 173–186.

[38] Eyal Peer, Joachim Vosgerau, and Alessandro Acquisti. 2014. Reputation as a sufficient condition for data quality on Amazon Mechanical Turk. *Behavior Research Methods* 46, 4 (2014), 1023–1031.

[39] PrivacyGrade. 2014. PrivacyGrade: Grading The Privacy Of Smartphone Apps. (2014). http://www.privacygrade.org/.

[40] John Patrick Pullen. 2015. Your Favorite Apps Know More About You Than You Realize. (2015). http://time.com/3857380/apps-security-privacy-trivia-crack/.

[41] Bahman Rashidi, Carol Fung, and Tam Vu. 2015. Dude, ask the experts!: Android resource access permission recommendation with RecDroid. In *Integrated Network Management (IM), 2015 IFIP/IEEE International Symposium on*. 296–304. DOI:http://dx.doi.org/10.1109/INM.2015.7140304

[42] M. K. Reiter and S. G. Stubblebine. 1998. Resilient authentication using path independence. *IEEE Trans. Comput.* 47, 12 (1998), 1351–1362.

[43] Meredith Rizzo. 2014. How Well Do Your Apps Protect Your Privacy? (2014). http://www.npr.org/sections/health-shots/2014/11/20/363342736/how-well-do-your-apps-protect-your-privacy.

[44] Franziska Roesner, Tadayoshi Kohno, Alexander Moshchuk, Bryan Parno, Helen J Wang, and Crispin Cowan. 2012. User-driven access control: Rethinking permission granting in modern operating systems. In *Security and privacy (SP), 2012 IEEE Symposium on*. IEEE, 224–238.

[45] Joel Ross, Lilly Irani, M. Six Silberman, Andrew Zaldivar, and Bill Tomlinson. 2010. Who Are the Crowdworkers?: Shifting Demographics in Mechanical Turk. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems (CHI EA '10)*. ACM, New York, NY, USA, 2863–2872.

[46] Jeff Sauro. 2010. If You Could Only Ask One Question, Use This One. (2010). http://www.measuringu.com/blog/single-question.php.

[47] Jeff Sauro. 2011. Measuring Usability With The System Usability Scale (SUS). (2011). http://www.measuringu.com/sus.php.

[48] Jeff Sauro. 2012. 10 Things To Know About The Single Ease Question (SEQ). (2012). http://www.measuringu.com/blog/seq10.php.

[49] Jeff Sauro and Joseph S Dumas. 2009. Comparison of three one-question, post-task usability questionnaires. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 1599–1608.

[50] Umesh Shankar and Chris Karlof. 2006. Doppelganger: Better browser privacy without the bother. In *Proceedings of the 13th ACM conference on Computer and communications security*. ACM, 154–167.

[51] Statista. 2016. Most popular global mobile messenger apps as of January 2016, based on number of monthly active users (in millions). (2016). http://www.statista.com/statistics/258749/most-popular-global-mobile-messenger-apps/.

[52] Statista. 2017a. Number of monthly active Facebook Messenger users from April 2014 to April 2017 (in millions). (2017). https://www.statista.com/statistics/417295/facebook-messenger-monthly-active-users/.

[53] Statista. 2017b. Number of monthly active Instagram users from January 2013 to April 2017 (in millions). (2017). https://www.statista.com/statistics/253577/number-of-

monthly-active-instagram-users/.

[54] Statista. 2017c. Number of monthly active Twitter users worldwide from 1st quarter 2010 to 1st quarter 2017 (in millions). (2017). https://www.statista.com/statistics/282087/number-of-monthly-active-twitter-users/.

[55] Eran Toch. 2014. Crowdsourcing privacy preferences in context-aware applications. *Personal and ubiquitous computing* 18, 1 (2014), 129–141.

[56] Voidcan. 2015. Top 10 Microblogging Sites for 2015. (2015). http://www.voidcan.org/top-10-microblogging-sites-for-2015/.

[57] Haoyu Wang, Jason Hong, and Yao Guo. 2015. Using text mining to infer the purpose of permission use in mobile apps. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. ACM, 1107–1118.

[58] D. Wendlandt, D. G. Andersen, and A. Perrig. 2008. Perspectives: Improving SSH-style host authentication with multi-path probing. In *Proceedings of USENIX Annual Technical Conference*.

[59] Primal Wijesekera, Arjun Baokar, Ashkan Hosseini, Serge Egelman, David Wagner, and Konstantin Beznosov. 2015. Android permissions remystified: a field study on contextual integrity. In *24th USENIX Security Symposium (USENIX Security 15)*. 499–514.

[60] Wiki. 2016. Fair Payment. (2016). http://wiki.wearedynamo.org/index.php?title=Fair_payment.

[61] Rebecca N Wright, L Jean Camp, Ian Goldberg, Ronald L Rivest, and Graham Wood. 2002. Privacy tradeoffs: myth or reality?. In *Financial Cryptography*. Springer, 147–151.

[62] M. Yuen, I. King, and K. Leung. 2011. A Survey of Crowdsourcing Systems. In *in Proceedings the 3rd SocialCom*. http://www.cse.cuhk.edu.hk/~king/PUB/SocialCom2011-Yuen.pdf

A Additional Tables

Removed Permissions	Facebook Messenger	Instagram
{AD}	1.0	
{CA}	1.0	1.6
{L}		
{CN}		
{AD,CA}	2.1	2.1
{AD,L}	1.8	1.1
{AD,CN}	1.4	
{CA,L}	1.6	2.1
{CA,CN}	1.6	2.9
{L,CN}		
{AD,CA,L}	0.9	3.2
{AD,CA,CN}	1.7	2.4
{AD,L,CN}	1.1	1.5
{CA,L,CN}	1.4	2.7
{AD,CA,L,CN}	1.9	3.3

AD: Audio, CA: Camera, L: Location, CN: Contacts

Table 4. Cohen D values for the effect size for the significant differences between the baseline and versions.

Number of Removed permissions	Facebook Messenger	Instagram
0 – 1		0.74
0 – 2	1.2	1.3
0 – 3	1.0	2.3
0 – 4	1.4	3.3
1 – 2	0.9	0.6
1 – 3	0.8	1.3
1 – 4	1.3	1.8
2 – 3		0.6
2 – 4		1.3
3 – 4		0.5

Table 5. Cohen D values for the effect size for the significant differences in usability as a results of the number of removed permissions

Removed Permissions	Facebook Messenger	Instagram	Twitter
{ } – {AD}	1.0		
{ } – {CA}	1.0	1.6	
{AD} – {AD,CA}		1.2	1.1
{AD} – {AD,L}	0.9		
{L} – {CA,L}		1.3	
{CN} – {AD,CN}	1.8		
{CN} – {CA,CN}	2.1	2.7	1.2
{AD,L} – {AD,CA,L}		2.2	
{AD,CN} – {AD,CA,CN}		1.9	0.9
{AD,CN} – {AD,L,CN}		1.1	
{L,CN} – {CA,L,CN}		0.9	
{AD,CA,CN} – {AD,CA,L,CN}		1.1	

AD: Audio, CA: Camera, L: Location, CN: Contacts

Table 6. Cohen D values for the effect size for the significant differences between nodes in Figures 4a, 4b and 4c

B The survey

Display the consent form.

Then display screening questions.

Non-eligible User: If any of the following conditions were received from the participant: Do not have an Android device, or do not have a data plan or wifi connectivity , or have and Android version older than 4.0 in Twitter and Instagram, and other than 5.0 for Facebook Messenger, or never use the App-Name, or do not live in the United States or younger than 18 years old. Then the survey recipient is sent to the end of the survey. Other than that, we display the questions bellow.

What is your Mechanical Turk ID?

Please select your gender

- Male
- Female
- Would prefer not to answer
- Other:

Please select the highest level of education that you have achieved.

- No High School
- High School
- Undergraduate Degree

– Master's Degree
– Professional (MD, JD/PhD)

What is your primary racial or ethnic background? Please select all that apply.

– Hispanic or Latino
– American Indian or Alaska Native
– Asian
– Black or African American
– Native Hawaiian or Other Pacific Islander
– White
– Other:

For Twitter: Which of the following tasks do you regularly perform in Twitter? (Select all that apply)

– Post tweets with pictures
– Post tweets with videos
– Post tweets with location tags
– Post tweets with text only
– Re-tweet other people's tweets
– Favorite other people's tweets
– Send direct messages
– Find friends from your contacts

*For Instagram:*Which of the following tasks do you regularly perform in Instagram? (Select all that apply)

– Post pictures
– Post videos
– Tag your location on posts
– Comment on posts
– like posts
– Send direct messages
– Find friends from your contacts

For Facebook Messenger: Which of the following tasks do you regularly perform in Facebook Messenger? (Select all that apply)

– Send pictures
– Send videos
– Send Audio messages
– Send text messages
– Share your location
– Find friends from your contacts

App installation:

We highly recommend that you connect to a free wifi network to avoid extra charges from your cellular provider.Uninstall the App-Name from your phone. You can accomplish that by going to Settings >Apps or Application manager>Find App-Name in the list and click on it>click Uninstall.Allow installation of non-Market apps. You can accomplish that by going to Settings >Apps OR security>make sure that the Unknown sources choice is checked. Install the following customized App-Name on your phone: Enter this link into your phone's browser and make sure to type it as it is exactly because it is case sensitive: http://URL.TO.APP.VERSION
OR
Scan the following QR Code with a bar code scanner:

To be able to continue the survey, you are required to successfully install the app. If you are unable to install the app, you may choose to abandon this task now and quit the study. OR you may contact us for technical help. Please email us at ourEmail@something.edu What is your current status?

– I successfully installed the customized App-Name
– I cannot install the app and I sent an email to receive technical help
– I cannot install the app and I would like to quit the study

If the third option, "I cannot install the app and I would like to quit the study," is selected, the participant is then sent to the end of the survey.
What is the name of the customized app as it appears in your app list?(we show an example)
Log in using the following credentials:
Username: a user name
Password: a password
To be able to continue the survey, you are required to successfully login. If you are unable to login, please email us at ourEmail@something.edu. What is your current status?

– I **successfully logged in**
– I **cannot login** and I would like to **quit** the study

If the second option "I **cannot login** and I would like to **quit** the study" is selected, the participant is sent to the end of the survey.

Twitter Tasks:

Task 1: Tag Location: **Post a new tweet** and **tag any location** on it (it does NOT have to be your own location). The tweet can be as simple as "Hello there". The location tag can be added by clicking on the location icon that appears with the options in the tweet editor. (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task)
Now answer the following questions about the task you have just completed (Tagging a location). Did you successfully tag a location on a tweet and post it?
-Yes -No -I don't know
Overall how difficult or easy did you find it to tag a location on a tweet and post it?

If the score is 4 or below: Why was it difficult to tag a location on a tweet and post it?

Task 2: Post a Video: Post a new tweet **with a new, short video.** Use **Twitter's camera** to record a new video. Make sure that the video does not contain faces or any other private information. (For example, you may simply record your room's wall.) Do **NOT upload** a video from your phone's **gallery**. (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task)

Now answer the following questions about the task you have just completed (Posting a tweet with a video).

Did you successfully tag a location on a tweet and post it?

-Yes -No -I don't know

If "Yes", then: Please describe the video that you posted:

Overall how difficult or easy did you find it to record a short video using Twitter's camera and post it?

If the score is 4 or below: Why was it difficult to record a short video using Twitter's camera and post it?

Task 3: Post a Picture: Post a new tweet **with a new picture. Take a new picture** using **Twitter's camera** and post it. Make sure that the picture does NOT contain faces or any other private information. (For example, you may simply take a picture of your room's wall.)

Do **NOT upload** a picture from your phone's gallery. (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task)

Now answer the following questions about the task you have just completed (Posting a tweet with a picture).

Did you successfully take a picture using Twitter's camera and post it?

-Yes -No -I don't know

If "Yes", then: Please describe the picture that you posted?

Overall how difficult or easy did you find it to take a picture using Twitter's camera and post it?

If the score is 4 or below: Why was it difficult to take a picture using Twitter's camera and post it?

Task 4: Viewing Contacts: Performing the following step will allow Twitter to access the contacts on your phone. If you are not comfortable with this, you may skip this step.

Click on "**find people**". Depending on your Twitter's version, this feature can be reached by either clicking on the find people icon that can be found in Twitter's top menu. Alternatively, you can click on on the top menu to reach "find people" option. Click on continue, and then skip the question that asks you for your phone number. Now, answer the following questions about your list of suggested friends. (If the app crashes or it does not let you finish the task, answer the question below then move on to the next task)

Now answer the following questions about the task you have just completed (Viewing suggested friends from your contacts).

How many friends did Twitter suggest?

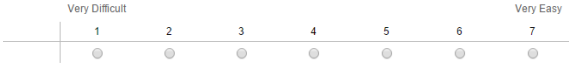
- 0-10 friends suggested
- More than 10 friends suggested
- Unable to count
- I would rather not perform this task

If the answer is "Unable to count", then: Why you were unable to count the number of suggested friends from your contacts in Twitter?

- The app stopped working
- Other

If "I would rather not perform this task", then: Why did you choose not to perform this task?

Overall how difficult or easy did you find it to count the number of suggested friends from your contacts in Twitter?



If the score is 4 or below: Why was it difficult to count the number of suggested friends from your contacts in Instagram?

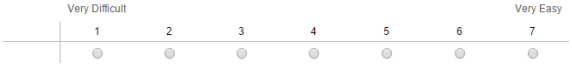
Task 5: Send a DM Go to messages, create a new direct message saying "Hello there" and send it to UserX. (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task)

Now answer the following questions about the task you have just completed (Sending a direct message).

Did you successfully send a direct message to UserX?

-Yes -No -I don't know

Overall how difficult or easy did you find it to send a direct message to UserX?



If the score is 4 or below:

Why was it difficult to send a direct message to UserX?

Task 6: Retweet: Go to **UserX** account and retweet or Undo retweet for any tweet. (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task) Now answer the following questions about the task you have just completed (retweet or undo retweet).

Did you successfully retweet or undo retweet any tweet from UserX's account?

-Yes -No -I don't know

Overall how difficult or easy did you find it to retweet or undo

retweet any tweet from UserX's account?

If the score is 4 or below:
Why was it difficult to retweet or undo retweet any tweet from UserX's account?

Task 7: Comment with a Random Number: Post the following tweet: "**Hello number XXXX**" (XXXX is a randomly generated number) (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task) Now answer the following questions about the task you have just completed (Posting a tweet).
Did you successfully post the tweet?
-Yes -No -I don't know
Overall how difficult or easy did you find it to post the tweet?

If the score is 4 or below: Why was it difficult to post the tweet?

Facebook Messenger Tasks:

Task 1: Tag Location: (This task is similar to Twitter's Task 1. In Facebook Messenger's context, we ask to share a location with a friend in the list).

Task 2: Post a Video: (This task is similar to Twitter's Task 2. In Facebook Messenger's context, we ask to send a video recorded using Facebook Messenger's camera to a friend in the list).

Task 3: Post a Picture: (This task is similar to Twitter's Task 3. In Facebook Messenger's context, we ask to send a picture taken using Facebook Messenger's camera to a friend in the list).

Task 4: Viewing Contacts: (This task is similar to Twitter's Task 4).

Task 5: Send an Audio Message: Find **"UserX"** in your friends list and **send** him **a short, new audio message. Record** the new audio message using **Facebook Messenger's microphone** and send it. Make sure that the message does NOT contain any private information. (For example, you may simply record a message that says "Hello there") (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task) Now answer the following questions about the task you have just completed (Sending an audio message).
Did you successfully send an audio message to UserX?
-Yes -No -I don't know
Overall how difficult or easy did you find it to send an audio message to UserX?

If the score is 4 or below: Why was it difficult to send an audio message to UserX?

Task 6: Send a Message with a Random Number: (This task is similar to Twitter's Task 7. In Facebook Messenger's context, we ask to send a specific random number to a friend in the list).

Instagram Tasks:

Task 1: Tag Location: (This task is similar to Twitter's Task 1. In Instagram's context, we ask to upload a picture and tag any location on it).

Task 2: Post a Video: (This task is similar to Twitter's Task 2. In Instagram's context, we ask to post a video recorded using Instagram's camera).

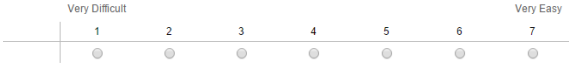
Task 3: Post a Picture: (This task is similar to Twitter's Task 3. In Instagram's context, we ask to post a picture taken using Instagram's camera).

Task 4: Viewing Contacts: (This task is similar to Twitter's Task 4).

Task 5: Send a DM: Take a picture using Instagram's camera and send it as direct message to **UserX.** (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task) Now answer the following questions about the task you have just completed (Sending a direct message).
Did you successfully take a picture using Instagram's camera and send it as a direct message to UserX?
-Yes -No -I don't know
Overall how difficult or easy did you find it to take a picture using Instagram's camera and send it as a direct message to UserX?

If the score is 4 or below: Why was it difficult to take a picture using Instagram's camera and send it as a direct message to UserX?

Task 6: Like a Picture: Go to **UserX** account and Like or Unlike any picture. (If the app crashes or it does not let you finish the task, answer the questions below then move on to the next task) Now answer the following questions about the task you have just completed (Liking or Unliking a picture).
Did you successfully like or unlike any picture from UserX's account?
-Yes -No -I don't know
Overall how difficult or easy did you find it to like or unlike any picture from UserX's account?



If the score is 4 or below: Why was it difficult to like or unlike any picture from UserX's account?

Task 7:Comment with a Random Number: (This task is similar to Twitter's Task 7. In Instagram's context, we ask to find a

specific picture and add a comment to it with a specific random number).

System Usability Scale:

Please answer these questions:

	1- Strongly Disagree	2	3	4	5- Strongly Agree
I think that I would like to use this version of the app frequently.					
I found this version of the app unnecessarily complex.					
I thought this version of the app was easy to use.					
I think that I would need the support of a technical person to be able to use this version of the app.					
I found the various functions in this version of the app were well integrated.					
I thought there was too much inconsistency in this version of the app.					
I would imagine that most people would learn to use this version of the app very quickly.					
I found this version of the app very cumbersome to use.					
Please answer option "4" to this question					
I felt very confident using this version of the app.					
I needed to learn a lot of things before I could get going with this version of the app.					

Technical Questions:

How long have you had an Android smartphone?

- Less than a year
- 1-2 years
- 3-4 years
- More than 4 years

Please answer "Rarely" to this question:

- Never
- Rarely
- Occasionally
- Almost every time
- I don't know

How often do you check the permissions requested by an app before installing it on your phone?

- Never
- Rarely
- Occasionally
- Almost every time
- Every time
- I don't know

If "Never", then: Why?

- I never thought about it.
- I don't care about what permissions apps request.
- I trust all apps on the Google Play store
- I usually check the reviews instead
- Other

If "Rarely", "Occasionally", "Almost every time", "Every time", or "Never": Have you ever stopped installation of an app because of the permissions it requested?

- Yes
- No
- I don't know

If "Yes", then: Why?

- I didn't like the permissions
- There were too many permissions
- I thought the app did not need them
- I don't know