Enhancing the Usability of Android Application Permission Model

Zeeshan Haider Malik(✉), Habiba Farzand, and Zahra Shafiq

Computer Science Department, Forman Christian College
(A Chartered University), Lahore, Pakistan
zeeshanmalik@fccollege.edu.pk,
habiba.farzand@gmail.com, zahrashafiq7@gmail.com

Abstract. Visualization is an encouraging tool for the study and understanding of text. Effective Visualization refers to visualization of data in a way that it requires minimal training for the user and is easy to understand. Keeping in consideration the user privacy and concern when it comes to android application permission model, the textual representation of permissions is transformed into visualization and the effect is examined deeply. The results depict that the purpose of visualization has been achieved. With the use of technique of visualization, users read, understand, acknowledge and are more aware about the permissions being accessed by the application.

Keywords: Android permission model · Textual vs. visualization
Usable security and privacy · Human-centric computing

1 Introduction

Android allows the installation of third party apps. A permission model is used to restrict the rights to the user's information available on the smartphone. Prior to installing the app, the user is prompted with a list of permissions that an app requires and the user is given the option to accept or reject the permissions. After accepting these permissions the installation continues and these permissions are accepted by the system. Giving access to users without having full knowledge about permissions can lead to security incidents. It has been noticed that most of the regular users do not even bother to read that list of permissions in order to install a particular application rather they just install it for their desired comfort and functionality. So, the users tend to try out the applications while not paying much attention to the permission dialog.

Changes have been made in the android application permission model. Now, instead of prompting for permissions prior to installation, the app permission model allows the user to download the app and when the user launches the app for the first time, then permissions are prompted one by one. The user then may select some/all permissions. This change has been applied to some of the applications available on the android market. Even with this change, the need for improvement remains as the user will only review permissions for one app or two but when the user becomes used to this model, the user will stop giving attention to the permissions and just accept the permissions without acknowledging what permission they are acquiring. Users do not pay

K. Arai et al. (Eds.): FICC 2018, AISC 887, pp. 236–255, 2019.
https://doi.org/10.1007/978-3-030-03405-4_16

attention to the permissions being asked at the time of installation or using an application for the first time. Due to which user's privacy and security is at risk.

In this paper we discuss the usability of android application permission model and the impact factor of visualization vs textual representation of permissions through laboratory study and survey.

2 Literature Review

2.1 Follow My Recommendations: A Personalized Privacy Assistant for Mobile App Permissions Maintaining the Integrity of the Specifications

A research by Liu et al. presents Personalized Privacy Assistant (PPA) app which includes privacy profiles for permission settings and customizing these profiles through PPA. They conducted two field studies. The first field study was conducted with 84 participants. The second field study was conducted with 72 participants. The application permissions were placed under three headings: app categories, app permissions and purposes associated with each permission [1].

The purpose of PPA was to use information about the apps which are installed on the mobile phone of the users so that user's privacy preference can be known and then used to recommend on how to configure associated permission settings. They designed an interactive profile assignment dialog, in which the PPA generates questions that helps the users to match to the privacy profile that suits their preferences and later used to provide recommendations on which permissions to deny.

Two field studies were conducted. One included the permission settings by PPA (n = 84). This was conducted to discover the permission settings of the users in order to build the privacy profiles and for this purpose permission manager was given to the users.

The first step of this study was to ask the users to fill in a survey form and then they were provided with a link to download permission manager and a user name was provided in for activating the permission manager. In the first week, the participants could use the permission manager to deny or allow permissions. The app, Permission Manager, also collected the frequencies of permission requests for installed apps, which were shown in the permission manager. In the second week, the participants received a privacy nudge each day for once only. After this the participants were asked to complete an online survey. The online survey technique was used as an exit ticket.

The next step taken was inviting all participants to an optional interview, in which the reasons for restricting or allowing different permissions, the comfort level concerning the permission settings, and the usability of the enhanced permission manager and privacy nudges was discussed.

The second field study included between-subjects field study (n = 72). Both had different participants. 78.7% of the recommendations made by the PPA were accepted by participants. Only 5.1% of recommended permission settings were later changed by the participants.

This research says that the app category and the permission type are significant predictors for an individual's judgment for allowing or denying the permission, whereas demographics, privacy concerns, the app name, access frequency and purpose information are not much weighty.

This is a good and effective way to help users in protecting their privacy but it has some reservations. The first drawback is that this application works after the app which the user wants to use has been installed on the phone. After the user has installed the app, the app has received the consent to mess with user's smartphone. Along with this, the app will require some space on the SD Card of the phone which means there has to be some reserved space for this app to function. The third drawback is that the user has to open the PPA and follow a series of steps in order to allow/disallow some/all permissions granted by the application which is time consuming and the user might not like doing it for every app he installs. Hence, there arises a need for a different and better system which brings improvement in the methodology of protecting the privacy of the user.

2.2 Analysis of Android Applications' Permissions

Second by second, applications are being added into the android market. A research by Johnson, Wang, Gagnon, and Stavrou analyzes the android application permissions. In their research, they downloaded 141,372 applications from official android market and examined the permissions prompted at the time of installation [2].

Out of 141,372, 54.01% applications demand extra permissions which are not required by the application. Out of 141,372, 49.95% applications demands less permission then are required. Out of 141,372, only 19.95% applications demand exact permissions, i.e. which they intend to use. These stats clearly depict that how important it is for the user to know what permissions are being granted to the application in order to protect privacy and for this purpose, a new methodology needs to be introduced which safeguards the privacy of the user.

2.3 Studying the Effectiveness of Android Application Permissions Requests

A research was done at Indiana University in March 2013 on studying the effectiveness of Android application permissions requests. They investigated five questions i.e. Do users understand what permissions granted to an application can do? Does additional extended text based explanation of permissions affect the understanding of permissions? Are visual warnings more effective than textual explanation? Do excessive permissions demanded by an application affect the user to think upon installation? Does the application's download count affect the decision to accept risk? Their study comprised of experimental portion followed by a survey. They basically had four hypotheses that are as follows:

(1) Most of the users are not aware of the function of common android permissions.
(2) An additional text explanation of the permission will improve the understanding.
(3) Excessive permissions demanded by an application have no impact on the installation rate.
(4) An application's popularity affects the installation rates.

There were 200 Android user respondents and they were asked to visit a web page with Android device to get access code required to start the survey. Survey took place on Amazon's mechanical Turk.

This research concludes that permission requests are appeared to be ineffective because greater number of participants admitted and were analyzed that they were not aware of the implications of the permissions and they intended to uninstall the application after understanding the permissions. Moreover, to improve the permission model, they also introduced extended textual explanation of the permission after the standard permission dialogue box but it was noticed that it did not have much significant impact, whereas when they introduced a non-technical visual warning for risky permissions it was noticed that it has more impact than extended textual explanation [3].

This research presents an effective way of examining the permissions in terms of its experimental design and survey i.e. it divided the respondents into groups and provided them with normal and custom models to carry out the experiments. This research shows that visual is more impactful then textual and visual can fulfill the purpose of making users aware of the permissions being granted to the applications. At this point, a more visualized method would make a perfect entry into the world of user security and privacy.

2.4 Presenting Risks Introduced by Android Application Permissions in a User-Friendly Way

Another study conducted in 2014, deals with a new method of letting user know that the installed applications are harmful or not. The proposed method in this paper consists of two parts. First part calculates the risk based on permissions the application requires and for that the permissions are distributed among color groups on the basis of risk that those permissions overall pose [4].

Red was assigned to high critical risk. Yellow was assigned to indicate moderate risk. Orange color was assigned to slight risk. Whereas Green to little or no risk. The results depicted that apps from Finance category are with highest risk, i.e. 37.21% and apps from Communication category requires maximum permissions.

With the rapid increase of advancement in technology, more and more finance related apps are being added on playstore; thus making finance related work easy for users. Along with this, the risk factor of apps is also increasing. Evil-minded parties might want to play around with user's private data and for this purpose it is highly recommended and required that users acknowledge and understand the meaning of each permission prompted by the application.

2.5 Android Permissions: User Attention, Comprehension, and Behavior

Another research from University of California in 2012, it indicates that measurable room for improvement is needed to make Google's Android permission model more effective. They performed two usability studies; an internet survey of 308 Android users and a laboratory study of 25 Android users. They used AdMob advertisements for Internet respondents and Craigslist advertisements for laboratory study participants.

Their primary findings were

(1) *Attention:* In both internet survey and laboratory study only 17% of respondents paid attention to permissions during installation, whereas 42% of the laboratory participants were unaware of the existence of permissions.
(2) *Comprehension:* Only 3% of internet survey respondents could correctly answer the comprehension questions. Other way 24% of laboratory study participants could tell about comprehension.
(3) *Behavior:* Majority of internet survey participants told that they decided not to install applications because of their permissions at least once. Whereas, 20% of the laboratory study participants told that permissions caused them to uninstall the application at times [5].

Table 1 shows the study of attention to permission while installing applications and it clearly shows the percentage of users who looked at the permissions at time of installation, i.e. 17% [5].

Table 1. Attention in permissions

Attention to permissions	Number of users		95% of CI
Looked at the permissions	4	17%	5% to 37%
Did not look but aware	10	42%	22% to 63%
Is unaware of the permissions	10	42%	22% to 63%

2.6 Messing with Android's Permission Model

Andre' et al. of RWTH Aachen University in 2012, detailed the permission model of Google's Android Platform and presented a selection of attacks that can compromise the user's device by demanding suspicious permissions. They showed how those attacks can silently root the targeted device. Moreover they further discussed the four permission protection levels of Android system.

- Level-zero

These permissions are so called normal permissions. They pose a low-risk factor and can only affect the application's scope.

- Level-one

These permissions are called dangerous permissions. They possess higher-risk and allow costly access such as sensitive user data.

- Level-two

These are called signature permissions and they are only granted if the application being installed is signed with the private key.

- Level-three

This highest category permission can be granted to applications that have signed the same certificate as the system image.

They further provided the list of susceptibility related to log permissions, formatted SD Cards, WebKit and other browser engines. Lastly, they presented their novel attacks against the permission model. They showed how an attacker silently root a user's device and can mount various attacks [6].

The attack path they proposed is composed of many smaller attacks based on different vulnerabilities of the permission model. That is how it becomes obvious that the permission model has failed to sufficiently secure a user against malicious applications. All in all, this paper highlights the fact that how inconspicuously looking applications demanding non-suspicious permissions can silently root your Android device and take-over the UI which is a high threat and can compromise the user's privacy.

2.7 Short Paper: A Look at Smartphone Permission Models

One of the various problems with the application model is the over-declaration of permissions. A research by Kathy et al., discusses about the issue of over-declaration. They conducted a survey of different permission models. Table 2 shows the summary of smartphone permission models [7].

Table 2. Summary of smart phone permission model

OS	Initial release date	# of permissions	Control	Information	Interactivity
Android	2008/09/23	75[a]	Medium	High	Low
Windows Phone 7	2010/10/11	15	Medium	Medium	Low
Apple iOS	2007/06/29	1	Low	Low	Low
WebOS	2009/06/09	1	Low	Low	Low
Blackberry OS	2006 Q3[b]	24	High	High	High
Maemo	2005/11/-	0	None	None	None

The control column represents how much control the permission system gives the user over applications. The information column represents what permissions application developer assumes the app will use and what permissions are actually accessed. The interactivity column represents how much interaction is required to use the system [7].

Android demands the highest number of permissions among the various OS. With the new versions of Android, number of permissions has also changed.

Due to the increase in permissions, there exists lack of documentation which greatly increases the effort to determine what permissions the application actually needs. To stop over-declaration, the balance between the cost of correctly declaring the permissions and the benefits of doing so must be upturned. The solution proposed by

Kathy et al., is to create an application that automatically defines the permissions required by an application. To accomplish this, mapping between API calls will be required.

With the rapid increase in the applications availability on the app market, the number of permissions has also increased. The more number of permissions mean more access to user data and privacy. Along with this there exists a gap between the permissions actually used by an application and number of permissions asked by the application. In such a situation, it becomes vital for the user to properly acknowledge and understand the permissions being granted to the applications in order to protect the privacy of the user. The current application permission model, unfortunately, fails to serve this service i.e. making the user aware of the permissions being granted. Thus, there arises a need for modification in the application permission model.

3 Proposed Model

Users do not pay attention to the permissions being asked at the time of installation or using an application for the first time. Due to which user's privacy and security is at risk. In order to improve and bring the user's attention towards permissions, our proposed model makes use of visualization. Visualization refers to displaying user's actual data instead of the textual representation. User's attention would be grabbed when they would see their personal data. For instance, the current model displays the permissions in the textual form which reads like "This app would access your photos". In our proposed model, in addition to this text, some random photos from user's gallery would be displayed as well. In this way, users would better understand them and at least think twice before allowing or denying a particular permission. Hence, the goal, i.e. user security will be achieved.

4 Experimental Design

Users were presented with our proposed app store named as "Exclusive Visual App Store". Silent Observation i.e. no communication took place during the experiment and Retrospective Testing i.e. respondent and researcher looked at the video recording together and the respondent discussed about his views and opinions, Questionnaire on Android Application Model and Exclusive Visual App Store in combination with System Usability Scale [8] were used. These four evaluation techniques were used for evaluating Exclusive Visual app store under the light of human computer interaction in comparison to the current Google Play Store.

The experiment was qualitative in nature and thus it involved 35 participants. 16 males and 19 females were selected for this experiment. The respondents were taken from the age group 18–34. This age group was selected because research illustrates that Millennials (ages 18–34) are more likely than older generations to use smart phones worldwide [9].

Exclusive Visual App Store is built with design using HCI techniques. Instead of textual representation of the permission, the visual representation is displayed. User's data is shown instead of textual representation of the permissions. The platform used for making the app store is Android Studio.

Before beginning with the experiment, the system was tested with few users to check if the system was working fine. Pilot testing included:

- Guarantee of privacy by presenting the user with a signed document.
- Assurance that the system is being tested not the respondents.
- Installing of Exclusive Visual App Store.
- Questionnaire on Android Application Permission Model.
- Random Selection of apps to download from Exclusive Visual App Store.
- Questionnaire on Exclusive Visual App Store.

The testing was conducted in the sequence of the following steps:

(1) Pilot testing
(2) Privacy document signing
(3) Questionnaire on Android Application Permission Model
(4) Installing of random apps from Exclusive Visual App Store
(5) Review on the user reaction by the user and researcher
(6) Questionnaire on Exclusive Visual Permission Model
(7) System usability scale.

5 Prototypes

For designing the layout of Exclusive Visual App Store and making sure that it matches the criteria of user friendly layout, paper prototypes were created. Another purpose was to select the design of permission model which the users find most appropriate and comfortable while using. The purpose of paper prototype is to take suggestions and feedback. Paper is used so that the user freely gives feedback and comments out the odd things of the design as he has in mind that not much effort has been put into making those designs. Initially three different designs of permission model were put out onto the paper. This was so, so that the user can select any one design which he finds the most appropriate. The designs of the first iteration of prototypes are shown in Figs. 1, 2 and 3.

These three designs were presented to 10 users. 5 out of 10 people selected Full Screen Display of permissions, 2 people selected display of permissions on button click and 3 people selected drop down menu. Upon these stats, Full Screen Display of Permissions and Drop down menu was selected for 2nd iteration of paper prototype. Changes were made and the design was enhanced.

The designs of 2nd iteration of paper prototype are shown in Figs. 4 and 5.

The first design of 2nd Iteration as shown in Fig. 4 shows an enlarged drop down menu for displaying the details of the permissions. The second design of 2nd Iteration as shown in Fig. 5 shows half screen dedicated for permission details.

These two designs of permission model were again presented to ten users and they were asked which design they would like to use on the basis of user friendliness and understandability. Seven out of ten people voted for permission display that covered

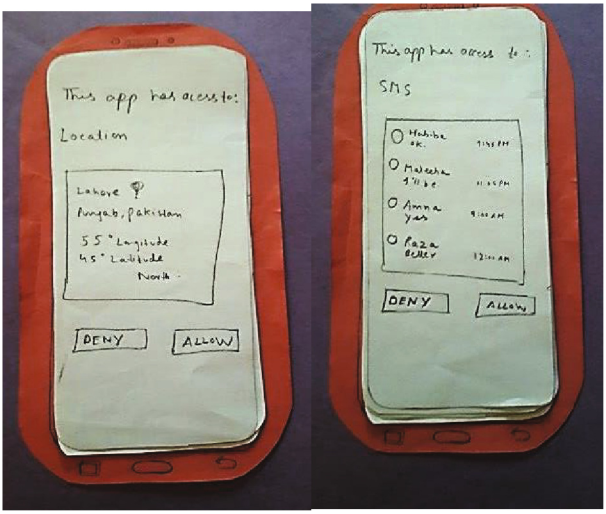


Fig. 1. Paper prototype – 1st iteration - design 1.

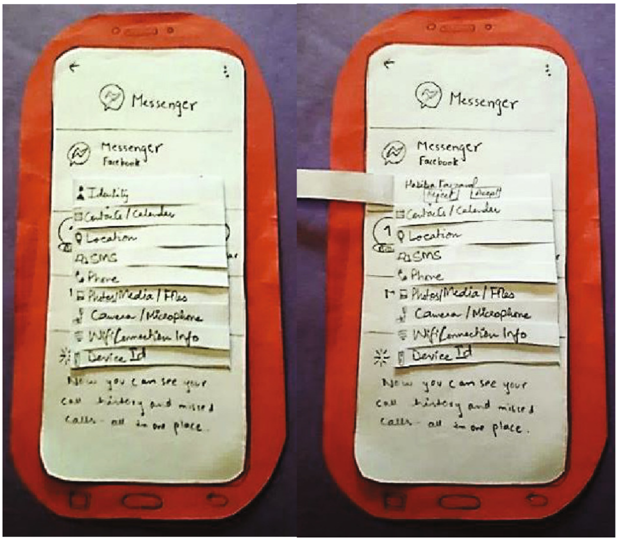


Fig. 2. Paper prototype – 1st iteration -design 2.

half of the screen (design#2) and three people voted for drop down menu (design#1). Using democratic decision model, design#2 was chosen for software prototype.

Software prototype was the third iteration for the design of permission model. For software prototype, Justinmind Prototyper was used. A few screenshots are shown in Fig. 6.

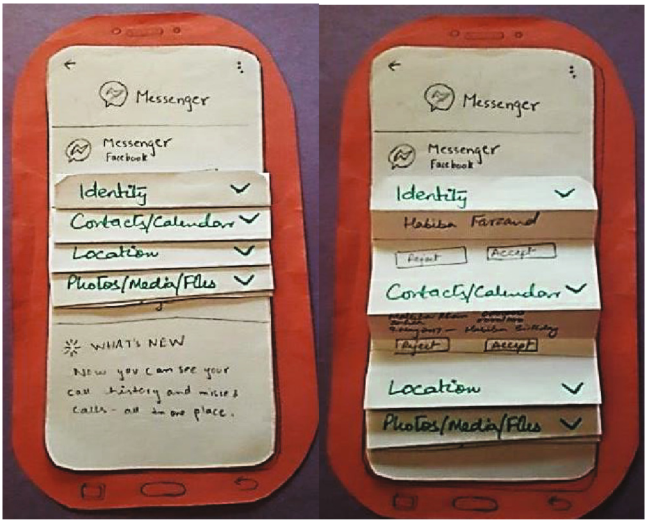


Fig. 3. Paper prototype –1st iteration- design 3.

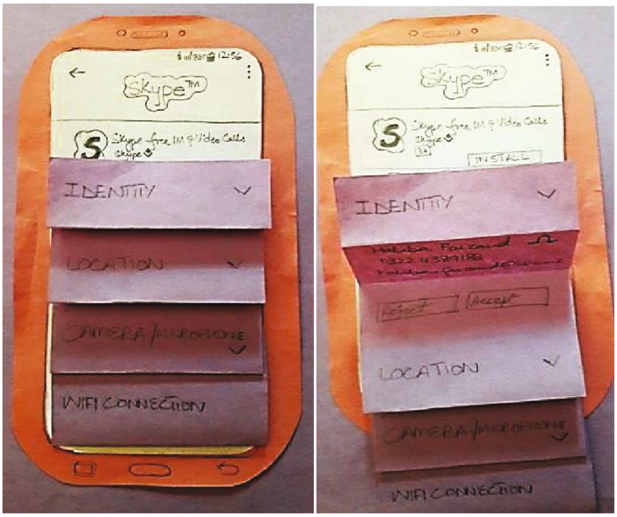


Fig. 4. Paper prototype-2nd iteration-1st design.

This software prototype was again presented to ten users and was asked for improvements in the design. Users suggested some minor changes like back button and cancel installation button. All of the changes were incorporated in the second iteration of software prototype. The second prototype was presented to 10 users again. In the second iteration of software prototype, no more changes were suggested by the users. Thus, this design was finalized for the permission model.

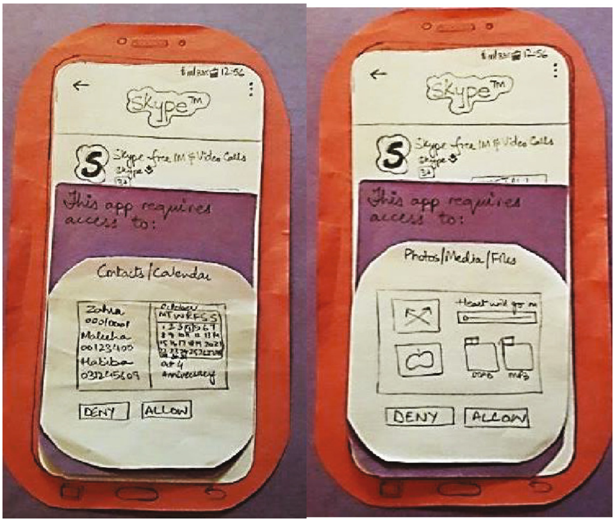
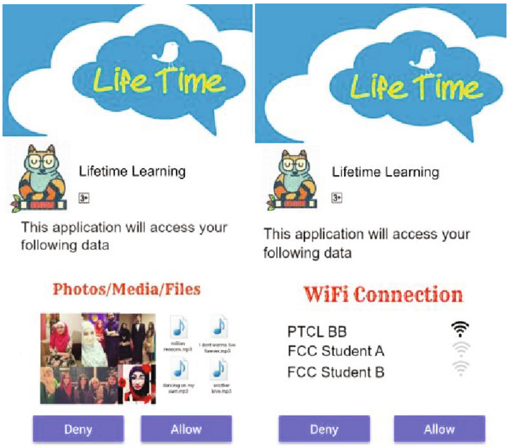


Fig. 5. Paper prototype – 2nd iteration – 2nd design.



This application will access your following data

This application will access your following data

Fig. 6. Software prototype.

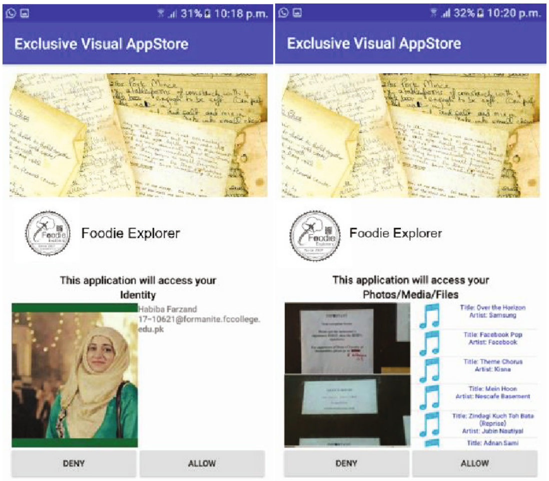
6 Implementation of Exclusive Visual App Store

Android Studio was selected as platform for the development of Exclusive Visual App Store. All the permissions were listed in the manifest file. A check for android version was placed as versions older then Marshmallow displayed permissions before downloading of an app in a list view and versions from Marshmallow onwards displayed permissions one by one after downloading the app. This was done so that Exclusive Visual App Store is workable on all android versions.

When Exclusive Visual App Store is launched for the very first time, permissions are displayed in the traditional android manner, i.e. textual representation. After the user has accepted the permissions, only then Exclusive Visual App Store is able to display the visualization of permissions. Keeping in mind the main purpose of this research, after installation of Exclusive Visual App Store, it was launched and all permissions were accepted which were displayed at the first launch. It was then handed over to the user. This was done so that the user does not go through the experience of textual representation of permissions (i.e. android's traditional way of displaying permissions) and is only exposed to visualization of permissions available at Exclusive Visual app store's applications.

Permissions were displayed visually one by one. The android default back button on each permission was disabled.

The user can either allow or deny the permissions displayed. If the user chooses to deny one or more permissions, the application will be downloaded anyway but the access to the denied permissions would not be granted. Some screenshots from Exclusive Visual App Store are shown in Fig. 7.



This application will access your Identity

Habiba Farzand
17-10621@formanite.fccollege.edu.pk

This application will access your Photos/Media/Files

Title: Over the Horizon
Artist: Samsung

Title: Facebook Pop
Artist: Facebook

Title: Theme Chorus
Artist: Kisna

Title: Mein Hoon
Artist: Nescafe Basement

Title: Zindagi Kuch Toh Bata (Reprise)
Artist: Jubin Nautiyal

Title: Adnan Sami

Fig. 7. Exclusive Visual App Store.

7 Testing

The testing was conducted with 35 users. Users belonged to the age group of 18–34. Exclusive Visual App Store was installed into the user's phone through connecting their phone with android studio using USB data cable. This was done so that the user experience his/her own data while going through the process of installation of an app from Exclusive Visual App Store and is best able to narrate the experience in terms of awareness, understandability and attention grabbing. The experience of the users was recorded in videos and questionnaires (pre-testing and pro-testing). The testing began with a brief discussion on what the research was all about. The testing proceeded with signing of a privacy document i.e. consent form. Users were then given their phones and were asked to select any app available at Exclusive Visual App Store and follow the steps of downloading.

8 Results

For this research, 35 users were selected out of which 54.29% females and 45.71% males were chosen. The results of questionnaire of Android Application Permission Model are summarized below.

Understanding of the permission is necessary in order to make the right decision of allowing or denying that particular permission. 63.15% females responded that they understood only some permissions of android application permission model. Whereas only 26.31% responded that they understood permissions. On the other hand, only 50% males responded that they understood permissions and 37.5% understood some of them. 12.5% males and 10.5% females responded that they did not understand the permissions. This shows that the users find it hard and are unable to understand all permissions and hence not in a situation to make the correct decision of allowing or denying a permission.

62.5% males and 31.5% females responded that they are aware of the permissions while 36.8% females and 12.5% males responded that they are not aware of the permissions. 31.5% females said they are totally unaware of the permissions. 31.5% females and 25% males said they are aware of some of the permissions. If users are not aware of the permissions, then this means that there is no logical reasoning behind allowing or denying a permission which leads to user privacy and security concerns.

Only 21.05% female respondents and 18.75% male respondents answered that current Android Application Permission Model grab their attention towards the permission being prompted, whereas 31.25% males and 36.84% females responded with a "no". Moreover, 37.5% and 36.84% answered respectively that "sometimes it does but not all the times". These stats shows how unaware are the users while allowing permissions to a particular application.

Additional findings about Android Application Permission Model include:

- Only 31.25% males responded that they read permissions while no female responded that they read permissions. On the other hand, 47.36% females responded that they do not read permissions. 18.75% males responded that they do not read permissions. 37.5% males and 36.84% females responded that they read the permissions sometimes. This shows that the current android application permission model is unable to grab user attention towards the permissions prompted.
- 25% males and 15.78% females responded that they rejected the idea of installation because there were too many permissions. 25% males and 36.84% females responded that they rejected the idea of installing an app because they did not like the permissions. One possible reason for not liking the permissions is that they were unable to understand the permissions and thus it resulted in not installing the app.

- 31% of males and 36% of females answered that the current application permission model is ineffective at conveying security information. 31% of males and 47% of females answered that it was effective only sometimes. Only 5.26% females and 6.25% males responded that the current application permission model is effective. The ineffectiveness of android application permission model refers to little or no understanding and attention of user towards the permissions granted to applications being installed on user's phone.
- Only 21.05% female respondents and 18.75% male respondents answered that current Android Application Permission Model grab their attention towards the permission being prompted. Whereas 31.25% males and 36.84% females responded with a "no". Moreover, 37.5% and 36.84% answered respectively that sometimes it does but not all the times. These stats shows how unaware are the users while allowing permissions to a particular application.
- Statistics show how user friendly is the current Android Permission Model, 37% males and 16% females answered in yes it is easy to use. But on contrary, 37% and 21% male and females responded with no that it is not easy to use. Whereas 18% males and 52% females rated it as normal to use. Probably the reason is the ambiguity in the model that it is not clear to use.
- In response to the questions from respondents, when asked about do they think textual representation of permissions is an effective way of displaying permissions, 44% males and 47% females responded that "no, textual representation is not effective". 50% males and 31.57% females responded that they are not sure about it. On the contrary only 6.25% males and 21.05% females responded with a "yes" i.e. textual representation is an effective way. (Before Exclusive Visual App Store Testing)
- The response for asking about either visualization of permission statements will help in grabbing user attention towards the permissions was high with a "yes". 75% males and 57.89% female respondents responded with "yes that visualization will help in grabbing user's attention towards the permissions being demanded". While 12.5% males and 36.84% females responded with "maybe". It shows that respondents were interested in visualization of the permission statements. (Before Exclusive Visual App Store testing).
- The response for the question on either they prefer textual or visual representation of permissions were that 44% males and 53% females responded that they will prefer visual representation of permissions over textual, while only 10.52% females and no male answered that they prefer textual representation. Moreover 43.75% and 47.36% respondents responded that they prefer both textual and visual representation respectively. (Before Exclusive Visual App Store testing).

The results for Exclusive Visual App Store are shown in Fig. 8.

94.73% females voted that Exclusive Visual App Store was successful in grabbing their attention towards the permissions prompted by the application. On the other hand, 75% males voted that their attention was taken towards the permissions. This shows that the visualization of the permissions was an effective method to grab the user's attention towards the permission. Once the attention is grabbed, the user is then conscious about letting an app access his data.

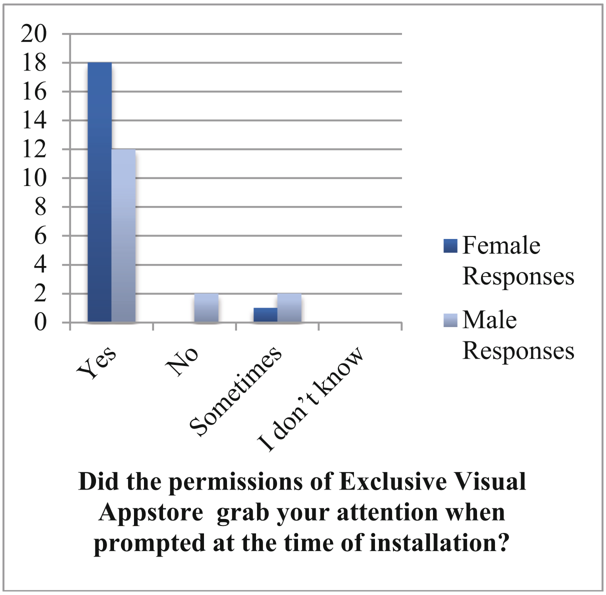


Fig. 8. Attention level of Exclusive Visual App Store.

Figure 9 shows the understanding of permissions of Exclusive Visual App Store. 100% success was achieved in making the users understand the permissions. 100% females and 100% males responded that they understood the permissions. The user knew what the permission is about and what the app will access if he allows the app to access it.

Figure 10 shows the awareness level about permissions of Exclusive Visual App Store. Exclusive Visual App Store was successful in making the users aware about the permissions. 100% males responded that they were aware of the permissions whereas 89.74% females responded that they were aware about the permissions. Only 10.52% females responded that they were aware of some of the permissions. No user responded for null awareness about permissions prompted.

Additional findings about Exclusive Visual App Store include:

- 43.7% of males and 57.8% of females did not reject the idea of installing an app due to permissions. This shows that they understood the permissions and they were clear about the meaning of each permission and thus had no issue in allowing the app to access their data.
- After testing and discussion, it has been recorded that Exclusive Visual permission Model was way effective in conveying security information. As per the statistics, 94% males and 79% females agreed and emphasized that Exclusive Permission Model was effective than Android Permission Model for conveying and making the user understand about security concerns.
- The stats for how user friendly the proposed model they experienced was of high number. 75% males and 100% females answered that the proposed Exclusive Model was easy to use and user friendly. There was not any ambiguity or misunderstanding. Permissions were visually understandable and clear to the respondents.

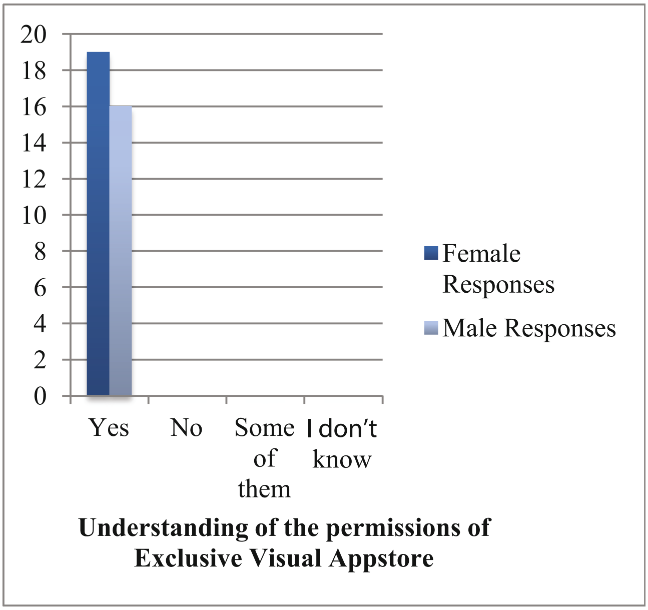


Fig. 9. Understanding of permissions of Exclusive Visual App Store.

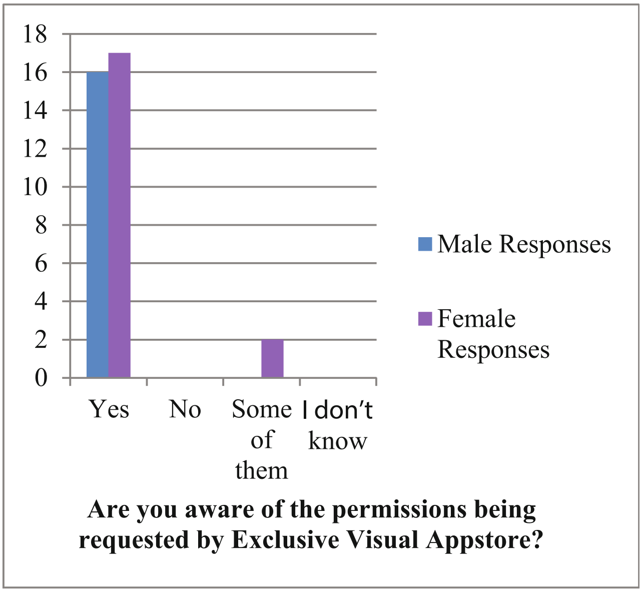


Fig. 10. Awareness of permissions in Exclusive Visual App Store.

- After using the proposed model the respondents were asked about is visual representation the effective way of displaying permissions. Stats recorded shows that 81% males and 100% females responded with yes. They agreed and respond back after testing the Exclusive system that visualization is an effective way to display permissions instead of text only.
- Statistics says that when asked from respondents, what do they prefer textual or visual representation for alert/messages. In context to that 50% males and 53% females voted for visual representation of alerts. While 34% and 36% females responded that they prefer both textual and visual representation for messages (after Exclusive Visual App Store testing).

System Usability Scale is an effective tool to measure the usability of software. This tool was taken under use to evaluate usability of Exclusive Visual App Store. The average score of System Usability Scale (Females) is 81.6 and the average score of System Usability Scale (Males) is 81.84. This shows that the design of Exclusive Visual App Store is user-friendly and not much training time is required for learning how to use the system.

9 Discussion and Analysis

The survey included 35 participants. There were 45.71% males' respondents and 54.29% female respondents.

The acknowledgement of the permissions from user side while downloading an app from app store is a very important aspect. The current android application permission model is ineffective in the accomplishment of this task. The current android application permission model is unable to make the user read, understand and then decide whether to allow or disallow an application to access user data. The evidence lies in the results of the questionnaire of Android Application Permission Model.

From derived statistics it has been recorded that there were only 6% respondents who responded that they read permissions at first and even further 12% respondents who answered that if they even read they do not get or understand the permissions prompted by a particular application. Users were not even aware of the permissions being requested at the time of installing or opening an application. As statistics indicate that 26% of the respondents said that they were not aware while 29% users voted that they were only aware of some of them and not all because of the current model's ineffectiveness. This situation here is alarming because it has been noticed and further security related concerns have been raised widely.

Let us suppose that if an alarm clock application requires or demands access to your call logs or device information, it makes no sense. Like this any Trojan application can mislead a user into its true intent. There were 30% respondents who agreed that they have not installed an application because of the permissions the application demanded and they did not like the permissions. While 20% of the respondents said that there were too many permission as a whole and it seemed irrelevant so they rejected the idea of installing an application.

Security concerns has always been a serious matter for tech savvy people and through survey forms in this research it has been clearly noticed that only 5.7% respondents said that the current Android Permission Model was effective in conveying security information. While on the other hand, rest of the respondents said no it is not effective and some said sometimes but not all the times as it should be that effective in order to convey security concerns because raising awareness about security is critical for users. It has been recorded, there were not satisfied answers that current permission model doesn't grab user's attention towards the attention of permissions being asked by the application. 71% respondents recorded their answers with no and sometimes. This shows that current permission model has failed to make users grab their attention in order to make them understand and raise awareness about permissions.

Table 3 summarizes the most significant findings of the research, i.e. usability satisfaction, awareness level, understandability and efficiency at conveying security information of android application permission model and of Exclusive Visual App Store.

Table 3. Android vs Exclusive Visual App Store

	Usability satisfaction	Awareness	Understandability	Attention	Efficient at conveying security information
Android	25.71%	26%	12%	20%	5.7%
Exclusive Visual App Store	88.57%	94.28%	100%	85.71%	85.71%

Further stats depicts that only 25.7% respondents answered that they are satisfied with the current permission model for how easy it is to use. While rest said that they are not satisfied or it is normal that it is not able to grab the attention. It has been further recorded that only 14% respondents answered and agreed that textual form of representation is an effective way displaying permissions while rest of the respondents considered it otherwise. When asked in survey what do respondents thinks that visual representation of permissions will help in grabbing user's attention towards the permission or not, the response was overwhelming 66% respondents answered that they think visualization of permission model will be effective and improve user's attention towards the permissions.

At the end when respondents were asked about what do they prefer textual or visual representation so only 5% respondents said that they prefer textual representation. While on the contrary, 49% answered that they prefer visual. Moreover, there were 46% respondents who actually thought that textual as well as visual representation will have a better impact on the user and it will fulfill the awareness need among users towards the permissions.

Exclusive Visual App Store not only was successful in grabbing the user attention towards the permissions but it also helped the users in understanding the users about the permissions prompted by the application. 85.71% users responded that their attention was grabbed towards the permissions while 100% users responded that they understood the permissions. Not just this, users were more aware about the permissions then before. 94.28% users responded that they were more aware about the permissions than while using Android Application Permission Model.

The idea of rejecting an application due to the dislike of permissions which was due to non-understanding of permissions was reduced to a large extent. 51.42% respondents did not choose to deny the idea of installing an application due to disliking permissions. This was so because now they were very much aware of the permissions and they understood what each permission meant.

Visualization of permissions proved to be quite successful in conveying the security information linked with application on app stores. 85.71% respondents answered that the Exclusive Visual App Store was successful in conveying the security information. This leads to another achievement i.e. lowering the chances of malicious apps to access unwanted user's data.

Usability is another key factor for the success of any software. Exclusive Visual App Store was designed through the process of iterations which involved paper prototypes and software prototypes. The carefully designed interface of Exclusive Visual app Store resulted in good usability. 88.57% users were satisfied with the system with respect to how easy it is to use. For further detailed usability study, System Usability Scale was used. The average score of System Usability Score by females was 81.6% and the average score of System Usability Score by males was 81.8%.

User's opinion after testing is another best way to examine and discover more about the system being tested. Users were very much satisfied with visualization of the permissions. They said that it was a good way in making the user know about the permissions and giving clear meaning to words. Text alone did not make much impact as compared to visualization. 91.42% respondents agreed to the statement that visualization is an effective way of displaying the permissions. When asked about their preference among textual and visual representation; 51.42% people voted for visual representation whereas 37.14% people voted for visual and textual representation of permissions.

10 Conclusion

In this paper we examined the efficiency and effectiveness of textual and visual display of permissions in Application Permission Model. 11.4% users responded that they understood the permissions of Android Permission Model whereas 100% users responded that they understood each permission of Exclusive Visual App Store. While using Android Application permission model, 25.7% participants said that they were aware of the permissions. On the other hand, 94.2% participants answered that they became aware of permissions by using Exclusive Visual App Store. Android Application Permission Model could only gather the attention of 34.2% participants whereas Exclusive Visual App Store was able to grab the attention of 86% participants. According to the data collected from the experimentation, it has been concluded that Android Permission Model is successful in conveying security information to 5.7%

respondents only. On the contrary, 85.7% participants found Exclusive Visual App Store effective at conveying security information.

Based on the high feedback of participants admitting they do not even read permissions being requested by the application, hence it can be concluded that the current model does not influence and grab user's attention. Therefore they are not even aware of the permissions and security concerns hence textual permission requests appear to be ineffective.

To improve and enhance the permissions prompted to be more effective and raise awareness, we introduced our proposed model named Exclusive Visual App Store. It had statistically significant effect when tested. Participants were more satisfied and agreed upon that visualization is the best and effective way of displaying permissions as well as conveying security information.

In light of these results, it can be concluded that visualization is an effective tool in order to safeguard user's private data.

Acknowledgment. We would like to use this opportunity to express our gratitude to Dr. Nazim Ashraf for his support in the development of the Exclusive Visual App Store. We would also like to thank all the participants who took part in the study.

References

1. Liu, B., et al.: Follow my recommendations: a personalized privacy assistant for mobile app permissions. In: Twelfth Symposium on Usable Privacy and Security (SOUPS 2016), June 2016
2. Johnson, R., Wang, Z., Gagnon, C., Stavrou, A.: Analysis of Android applications' permissions. In: Proceedings of the 2012 IEEE 6th International Conference on Software Security and Reliability Companion (2012)
3. Benton, K., Camp, L., Garg, V.: Studying the effectiveness of Android application permissions requests. In: 2013 IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshops) (2013)
4. Muska, P., Varga, J.: Presenting risks introduced by Android application permissions in a user-friendly way. Tatra Mt. Math. Publ. **60**, 85–100 (2014)
5. Porter, A., Ha, F., Egelman, S., Haney, A., Chin, E., Wagner, D.: Android permissions: user attention, comprehension, and behavior. In: Proceedings of the Eighth Symposium on Usable Privacy and Security (2012)
6. Egners, A., Marschollek, B., Meyer, U.: Messing with Android's permission model. In: 2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications, Apr 2012
7. Au, K.W.Y., Zhou, Y.F., Huang, Z., Gill, P., Lie, D.: Short paper: a look at smartphone permission models. In: Proceedings of the 1st ACM Workshop on Security and Privacy in Smartphones and Mobile Devices (2008)
8. Brooke, J.: SUS—a quick and dirty usability scale. In: Jordan, P.W., Thomas, B., Weerdmeester, B.A., McClelland, A.L. (eds.) Usability Evaluation in Industry, vol. 189, pp. 4–7. Taylor and Francis, London (1996)
9. Pouchter, J.: Smartphone Ownership and Internet Usage Continues to Climb in Emerging Economies. Pew Research Center, Washington (2016)