OPPORTUNITY COST ANALYSIS OF ANDROID SMARTPHONES' PERMISSIONS

BY SWAPNIL SARODE

A thesis submitted to the

Graduate School—New Brunswick

Rutgers, The State University of New Jersey

in partial fulfillment of the requirements

for the degree of

Master of Science

Graduate Program in Electrical and Computer Engineering

Written under the direction of

Prof. Janne Lindqvist

and approved by

_____

_____

_____

_____

New Brunswick, New Jersey

October, 2014

ABSTRACT OF THE THESIS

Opportunity Cost Analysis of Android Smartphones' Permissions

by Swapnil Sarode

Thesis Director: Prof. Janne Lindqvist

This thesis provides the opportunity cost for reading androids permission model. We investigate the opportunity cost for users and an example nation (United States), if people would actually read these permission screens during installation time.While the Federal Trade Commission and reserachers try to protect users' privacy and to improve their comfort level with mobile applications, users still remain unaware of these changes. Users are given a choice to overview the permissions an app would use and have to make an on the spot decision to accept these and move forward with the installation. In this research we project the time required by an average user if they were to read the permissions and compute the monetary value of that time in different situations. An average user may spend half an hour in overviewing permission screens bearing maximum opportunity cost of $23 and a minimum of $3 based on whether it was read at work or leisure. Other than this, if the users decide to read the details of these permissions as well, they will spend more time and hence bear more cost. Reading permissions with details would require users to spend two and half hours annually with a maximum cost of $106 and minimum of $13. An entire nation (United States) would have to invest a minimum of $174 million and a maximum of $6 billion, in reading permissions.

Acknowledgements

First and foremost, I would like to thank my advisor Prof. Janne Lindqvist for giving me this research opportunity. I sincerely thank him for his support and guidance. I would also like to thank my parents and family for always believing in me and encouraging me. I would like to thank the review committee for their useful feedback. Lastly I would also like to thank my HCI labmates for their suggestions and timely feedbacks. This material is based upon work supported by the National Science Foundation under Grant Number 1223977. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

Table of Contents

List of Tables

List of Figures

0.1 Introduction

The majority (61%) of US mobile subscribers are smartphone users [1]. Today, smartphone functionality can be augmented with mobile apps that can have access to several privacy-sensitive capabilities of the smartphone. These apps are very popular, the largest app markets today are Google Play with over 1,000,000 apps and over 50 billion downloads, and Apple's App Store with over 1,000,000 apps and over 60 billion downloads.

The mobile app stores aim to protect people's privacy by two main approaches: Apple by vetting applications before allowing them to the App Store and Google by introducing a user-centric security model, which requires the app developers to explicitly declare access to capabilities that Google has deemed sensitive. Currently, there are 130 such capabilities that need explicit permission [2].

During the installation of an app, the Android system shows to the user the capabilities requested by the app. An example of this is shown in Figure 1. This app called "Despicable Me" requests access to several capabilities, including "Modify or delete the contents of your USB storage", "Full network access", "Read phone status and identity", and "view Wi-Fi connections". The users' explicit task here is to either install or not install the app. We note that in Apple's model, the apps have similar capabilities, but the users are prompted only about allowing location-access prior to installing an app, nothing else is revealed to the users. This has lead to controversies with "social apps" such as Facebook and Twitter which were deemed to be "harvesting contacts" [3].

The Federal Trade Commission (FTC) looks into the matters of privacy for mobile phone users. This involves setting the guidelines for app developers, rules and regulations to be followed and privacy policy disclosure. With rapid development in the mobile technology, some app developing companies have started accessing consumers' private data without their knowledge. The FTC, while making attempts to resolve the privacy issues is still striving to provide a precise set of guidelines to the developers.

Due to these and other controversies, the Federal Trade Commission (FTC) in

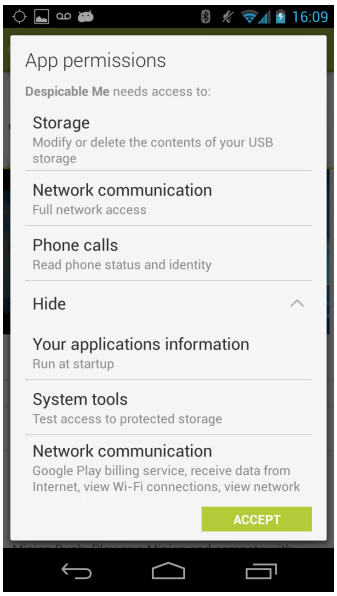

Figure 1: Example Android app permission screen for "Despicable Me" – a game app based on a movie. The app is requesting access to "Modify or delete the contents of your USB storage", "Full network access", "Read phone status and identity", "view Wi-Fi connections", among others.

the United States has been investigating the privacy issues with mobile phones. In 2013, a FTC report provided recommendations and guidelines to be followed by app developers. "FTC staff strongly encourages companies in the mobile ecosystem to work expeditiously to implement the recommendations in this report. Doing so likely will result in enhancing the consumer trust that is so vital to companies operating in the mobile environment. Moving forward, as the mobile landscape evolves, the FTC will continue to closely monitor developments in this space and consider additional ways it can help businesses effectively provide privacy information to consumers," the report states [4].

Another United States government organization, The National Telecommunications and Information Administration (NTIA), has recently outlined that, for example, user permission screens should make it easier for users to understand the resources accessed by an application [5].

In this paper, we investigate the opportunity cost for Android smartphone users *if they would actually read all the user permission screens*. We acknowledge that past work has shown that few people read End-User License Agreements (EULAs) [6] or

web privacy policies [7], because (a) there is an overriding desire to install the app or use the web site, (b) reading these policies is not part of the users main task (which is to use the app or web site), (c) the complexity of reading these policies, and (d) a clear cost (i.e. time) with unclear benefit. Thus, we desire to evaluate what would the actual cost be in terms of time and money.

0.1.1 Economic Thinking

The FTC sets guidelines to protect the consumers' privacy rights. These set of rules or guidelines can be made strict enough to protect the consumers' privacy but that might have a negative impact on the app development progress, considering additional limitations the developers will have to face.

Whether or not to invest time in reading privacy policy is an individual choice which is majorly influenced by the factor, 'advantage'. Consumers would rather avoid reading privacy policies, if they do not see the benefits of their actions clearly [8]. Here, time to read privacy policies has a subsequent equivalent cost, a cost that in economic world has great importance [9].This cost affects the consumers' rationale of reading or skipping privacy policies and keep the consumers' expense to optimum [10].

The contributions of the paper are as follows. We present the time spent annually by a user in reading permission screens and cost to read these permissions. Further, we also show the time spent by an entire nation (United States) in reading permissions and the cost involved in reading. Additionally we provide time and cost estimates for reading permissions of pre-installed apps and developers' privacy policies. We also present the actual reading time spent by users through a lab study.

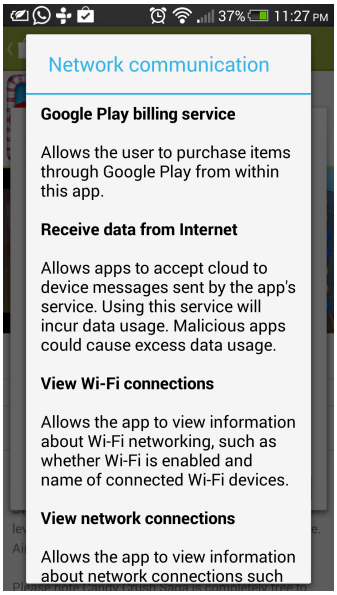

Figure 2: The figure shows details of one of the permissions, Network Communication. The detailed version has extra explanation for each of the permission variables Google Play billing service, Receive data from internet, View Wi-Fi connections and View network connections.

0.2 Background and Related Work

In this chapter, we discuss the implementation of the user-centric mobile app security and privacy model on the Android platform ("Android permission screen"), and the related work.

0.2.1 Android Permission Screens

Androids' permission model aides the users to get knowledge on what resources the application will be accessing. It serves to help the user to make a decision as to whether or not a particular application is trustworthy of the resources and private data that it would be accessing. In the past, consumers had no control over the choice of resources that were accessed by the application and their only option was uninstalling the application from their phones. This action saw sudden drop in the usage of some applications as more and more users became aware of their private data being accessed, which was out of their comfort zone. Without privacy policies, companies have all the information about their practices and consumers have none, leading to an information

asymmetry [11]. Information asymmetries are one potential cause of market failure [8].

The idea behind the permission screen is to show to the users capabilities of the app that can potentially access sensitive information about the user or affect the functionality of the phone. These potentially sensitive permissions are defined by Google, and currently there are 130 of them available [2].

There are several ways that the users of Android smartphones can interact with the permission screens. Also users may have multiple devices requiring them to read and understand permissions for different platforms. For example, a tablet device running Android may have different permissions for the same app than the permissions that app uses on an Android smartphone. Readability also depends on the users prior knowledge of permissions, this includes but is not limited to the number of apps installed, familiarity with the permissions and number of devices owned by the user. Users may encounter permissions in various ways.

During Standard Installation

1. The user connects to the Google Play app market.
2. Finds the application that suits his or her needs and clicks Install.
3. Brief version of permission screen is displayed to the user (Figure 1.
4. If the user needs more information and taps on the permission a more detailed explanation is displayed (Figure 2).
5. The user decides to install the app and the download and installation begins. Alternatively, the user can decide not to install the app.

After Standard Installation

1. The user has to navigate to the Apps details by following, Settings → Apps.
2. The user then looks for the app, which permissions he or she is interested in.
3. After selecting the app, the user can scroll down to find the permissions used by that app.

Installing from Other Sources

1. The user downloads the app from a source such as Amazon App store or directly from a third-party website.
2. Androids' package launcher asks the user if the source is trusted before installing the app.
3. Steps here are the same as during standard installation.

Installing Updates

1. The user finds the app he or she desires to be updated.
2. After choosing to update, a permission screen is shown. This can only be delta to the previous installation.

0.2.2 Related Work

Becher et al. [12] give an overview of mobile phone security history and developments, and identifies further research domains related to mobile security. Anderson et al. have studied different application markets and installation mechanisms [13], and several authors have written position papers about application markets [14–17]. Considerable effort has been spent on understanding e.g. Android security and permissions [18–24], and hardening Android security model [25–28]. Recently, taint analysis has been used to analyze leaks from Android [29] and iPhone [30]. AppFence [31], MockDroid [32], and TISSA block data leaking to network and by faking the capabilities of the phone so that potential sensitive data cannot be retrieved.

Wei et al. showed that some applications may request permission which are never used, and hence increase the vulnerabilities in the app and raise concern of security risks [24]. Such apps were tagged as over privileged apps. Although some developers may need extra permissions to make sure that their

Table 1: Categories of applications considered for experiment.

Categories
Games
Books and References
Business
Comics
Communication
Education

updated versions of apps in the future work without any problem, some frown upon this choice and consider it as a bad practice. Permission system of Android is complicated with new permissions being added to the system as the technology progresses. Another study proposes an analysis tool that can be used to extract an accurate permission specification to improve Androids' permission system [23]. TaintDroid was another such attempt to track the sources of private data being used for reasons other than what it was expected for [29]. This tracking tool helped users in identifying the applications' misuse of permitted resources.

On the user interface side, Lin et al. [33] used crowd sourcing to capture users expectations of what sensitive resources mobile apps use, and designed a new privacy summary interface that prioritizes and highlights places where mobile apps break peoples expectations. Kelley et al. studied the users' decisions to select apps based on permission requirements [34]. They conclude that the position of privacy disclosure in the application selection process is partially responsible for users ignoring them. They experimented by putting the permission requirements on the main screen in simple words and found that, this implementation prompts the user to make a decision between similar applications based on their permission requirements. Kelley et al. also studied into what causes the users' to ignore the permission screens, and one of the reasons were difficulties in understanding the permissions [35].

Issues with user-centric mobile app security and privacy models are relatively new,

however, researchers have been studying web privacy policies for much longer. In the past, studies have been done about web privacy policies and the reasons behind why consumers would rather avoid reading privacy policies and just enjoy the web [8, 11]. Finally, the most related from web privacy policies is the work of McDonald and Cranor [8], who studied the opportunity costs of reading web privacy policies. Since web privacy policies are considerably longer, they estimated reading web privacy policies would cost in time approximately 201 hours a year, which would be worth about $3,534 annually per United States web user. The user centric permission model of mobile platforms is a research domain that is still being explored and the following research is a step in the same direction.

0.3 Method

In this section, we present how we estimated the time required to read the permission screen shown to users before installing an app on their smartphones. We calculate the time taken by each consumer and the time estimate for an example nation (United States).

We estimate annual time to read for individual consumer as,

$$T_R(individual) = n \times W \div R$$

where, n is the number of applications downloaded in a year, W is the average number of words in app permissions, and R is the average national reading rate.

With this equation we can estimate the time taken to read the permission screens by an individual user over a year. To obtain the time taken for an entire nation we extend the same equation, with the only difference being that the number of applications downloaded (n), is the nations' share of applications downloaded globally. The time was later used for computing opportunity cost. Opportunity cost is the monetary value of the time spent in performing some task, in this case reading permissions. A similar method was later followed for obtaining opportunity cost for reading permissions of pre-installed applications. This method was contrasted with a lab study which is explained in Chapter 5 to find actual reading time spent by users while reading permissions.

0.3.1 Procedure

For our analysis on permission screens we focused on the Android platform. We considered the top ten applications in each category seen on the Google Play Store, the default app market for Android smartphone users. The categories considered for this analysis are listed in the table 1. We looked at the top ten apps in each of the categories to be able to estimate the common lengths for permission screens that users can encounter when installing apps from the app market.

The length of a permission screen was determined using the number of words on the screen as a measurement metric. Every application has a set of permission requirements.

Permissions
Storage
Phone Calls
System Tools
Network Communication
Your Location
Your Personal Information
Your Messages
Services that Cost You Money
Your Accounts
Affects Battery
Development Tools

Table 2: Common permissions types required by applications. Note that a permission type can include several kind of permissions, e.g. "Your Location" includes both "fine-grained" and "coarse-grained" location permissions.

We categorized these permissions which made it easier to get the number of words for all the considered applications. All the different permissions encountered with this analysis are listed in table 2. Additionally description of some permission may be hidden from the users by default. These default hidden permissions can be found under the "Hide" subsection of the permissions screen. There may be more information available on a particular permission under this subsection. The permission screen gives brief explanation of different permissions required by the application, which may not be enough for everyone to understand what is implied. Hence, each of the brief descriptions has a detailed version that is displayed if user taps on a particular permission, which acts as an extension for explaining that permission. We calculated the total number of words involved in reading the entire list of permissions as the sum of words in the brief description and words in the detailed explanation to the permission. We also used only the word count of the brief version of permissions for getting minimum amount of time a user spends on the permission screen. For computing the reading time in each case a reading rate of 180 words per minutes was used, with respective word count for that part. Since the computed time is based on assumptions, to get a scalable solution the values were computed with upper and lower bounds for each case.

Figure 1 is an example of the permission screen as seen by the user. In some cases all of the content is seen on a single screen, whereas in other cases some factors are hidden

or the list is long and hence requires scrolling up and down to read entire content. All this also depends on the screen resolution and the orientation while reading. Figure 2 shows a snapshot of permission screen with the further details of one of the permission.

Additionally, we also collected data on whether or not there is a link to the complete privacy policy provided on the app portal. The complete privacy policy describes every permission in detail and some even inform users of specific details about what a particular permission is used for, inside the app. The link of privacy policy being included as a part of app description is just an FTC guideline, and hence it is not binding on the developer to show additional privacy policy details.

0.4 Results

The time to read a permission screen depends on two factors: the reading rate and the number of words. The reading rate also varies depending on the medium that carries the text, for example, reading rate for reading text on a paper is different than reading rate on computer screens. The rate also varies with screen sizes and since smartphones are of varied screen sizes we performed the experiment with 4.7 inch screen smartphone. The reading rate for an average human is 180 words per minutes (WPM) or 3 words per seconds (WPS) for reading material on smartphone screens [36, 37]. Next, we will discuss how many permission screens the user visits in a year, followed by estimation of how much time would be spent reading them.

0.4.1 Apps on a Phone

Table 3 shows the number of applications a user downloads over a year, which also reflects the permission screens he or she has to go through in a year. Nielsen research estimated that a smartphone user have up to 41 applications on their mobile phones [38]. We use this number as our base value to calculate the number of applications downloaded in a year by an average smartphone user. Reports state that an average smartphone user downloads *two* to *nine* applications every month. Research2guidance is an analysis firm that reports the number of installs per month to be two [39], and Admob, a mobile advertising company, reports the value as nine [40]. We used these values as lower and upper bound respectively. The number for app downloads shown in the table 3 will be used later to find annual opportunity cost to an average user.

Estimates	Downloads/month	Downloads/year
Lower Bound	2 apps/month	65 apps/year
Point Estimate	5 apps/month	101 apps/year
Upper Bound	9 apps/month	149 apps/year

Table 3: Permission screens visited in a year. This counts the number of permission screens encountered by the users.

0.4.2 Time for reading permission screen with brief information

When users are about to install an app they are first presented with permissions as shown in figure 1. This view just shows the overview of permissions. Table 4 shows the reading time required by an average user to read the permission screens of various lengths based on their word count. The word count represented here is obtained as quartiles of total word count from results of the analysis described in Chapter 3 (see also appendix A). This time represents the time required to read only the brief description of the permissions and hence we call it the partial reading time. With the help of values obtained in table 3, we have the word count and also the annual time that would be spent behind reading the brief descriptions of permissions for all the permission screens seen in a year, as shown in table 5. Thus on an average user would be required to spend 32 minutes reading these brief permissions over a year.

Policy Size	Word Count	Reading Rate	Time to Read
Short	41 words	3 WPS	14 sec
Medium	58 words	3 WPS	20 sec
Long	75 words	3 WPS	25 sec

Table 4: Time spent on one app to read the brief permission screens. The reading rate is represented in Words Per Second.

Estimates	Word count	Time to read
Lower Bound	2665 words	15 minutes
Point Estimate	5858 words	32 minutes
Upper Bound	11175 words	62 minutes

Table 5: Time spent annually to read all the brief permission screens.

0.4.3 Time for Reading Brief Permission Screen and Detailed Descriptions of Each Permission

As mentioned earlier, every permission has a detailed explanation, giving more information about that permission. If users were to read the permissions in detail it would obviously require more time. Table 6 shows the time required to read the entire permission screen with detailed explanation for every permission, for one app. Similar to the partial reading time, we use the quartiles from the experiment described in Chapter 3 for the word count. Table 7 has the estimates for annual time required to read

the permissions with their details obtained using the values in table 3 . Although the value of annual time does seem small, studies have shown that smartphone users spend *thirty-nine* minutes out of their daily time using the smartphone apps [38]. Thus if users were expected to read permissions for one fourth of their time spent on smartphone, they would either ignore permissions or otherwise look for other options such as privacy protecting apps.

Policy Size	Word Count	Reading Rate	Time to Read
Short	180 words	180 WPM	1 minute
Medium	256 words	180 WPM	1.4 minutes
Long	347 words	180 WPM	1.9 minutes

Table 6: Time to read all permission screens of an app along with the details for every permission. The reading rate is represented in Words Per Minute.

Estimates	Word count	Time to read
Lower Bound	11700 words	1 hour
Point Estimate	25856 words	2.4 hours
Upper Bound	51703 words	4.8 hours

Table 7: Time spent annually to read all permission screens of all installed apps along with their details.

The opportunity cost is the worth of the alternative option that could replace the activity under consideration. In our case, this represents worth of other leisure activities

or work done, instead of reading the permissions required by an app. We estimate opportunity cost for leisure and work in terms of mean hourly wage. In the United States, overhead at work is estimated as twice the rate of take home pay [41]. This means that the overhead of reading the permission screens at work would cost twice the wage. Studies show that, people estimate their leisure time at one quarter of their take home pay [42]. Based on this, we estimate cost to read permission screen at leisure as one quarter of the wage.

The opportunity cost of the time was calculated with the help of average daily wage a person gets. Most of the time spent on smartphone comes under leisure activities. However, due to lack of research that confirms the usage of smartphone only during leisure, we calculated the opportunity cost for both, usage during leisure and usage during work time. The United States Department of Labor reports the average hourly wage to be $22.01 [43]. Thus the hourly worth of leisure time would be $5.5 and for time at work it would be $44.02. Based on these values, we provide the opportunity cost for reading permission screens at leisure and work in table 8. Table 9 shows estimates of opportunity cost to an average consumer for reading the permission screens along with their details. If permissions are only read during leisure, reading brief permissions would cost $2.93 for a year ,while reading permissions only read at work would cost $23.47. Similarly reading permissions with details would cost $13.2 annually if read during leisure and $105.7 if read at work.

Estimates	Time to read	Read at Leisure	Read at work
Lower Bound	15 min	$1.375	$11
Point Esti-mate	32 min	$2.93	$23.47
Upper Bound	62 min	$5.68	$45.48

Table 8: Opportunity cost for reading brief permission screens. The value represents the annual cost for reading all the brief permission screens the user comes across.

Estimates	Time to read	Read at Leisure	Read at work
Lower Bound	1 hour	$5.50	$44.02
Point Esti-mate	2.4 hours	$13.2	$105.7
Upper Bound	4.8 hours	$26.4	$211.3

Table 9: Opportunity cost for reading detailed permission screens. The value represents the annual cost for reading all the permission screens the user comes across along with the permission details.

Not all of the time spent on smartphone can be considered as leisure usage, most of the applications are also used at work such as, Daily Schedule Manager, Google Docs, Dropbox, and many more. The results seen in table 9, provide us individual values for

leisure usage and work usage. However the actual value of opportunity cost would lie in between these two, so considering the point estimate, we would have the opportunity cost of reading permissions ranging from $2.93 to $23.47 per year for one smartphone user.

0.4.4 Cost to the Nation

The opportunity cost to the entire nation (United States) was calculated from percentage market share of the nation out of the global share. The Google Store generally reports their downloads as total number of installs from the market. By June, 2012 there were 20 billion applications installed [44], and the number rose to 48 billion by May, 2013 [45]. The total downloads from the Google Store were 28 billion and the United States contributes to 21 percent, according to App Annie report of November 2012 [46]. App Annie reported the monthly download share of United States for October, 2012, however, due to lack of other reports that would provide annual share, we extended the same value for annual share to get us the app downloads in the United States. Thus, the opportunity cost for nation was computed using 5.88 billion annual app downloads in United States. Table 10 , shows the time required by the entire nation to overview the permissions and the opportunity cost for that. As mentioned earlier, the brief versions may not be enough for all users and hence we also present estimates for time required by the nation to read the entire permissions list with details about every permission and the opportunity cost involved in doing so in table 11. The actual opportunity cost could range between $173 million to $6.14 billion.

Estimates	Word Count (words)	Time to read (hours)	Cost to read at Leisure	Cost to read at Work
Lower Bound	241.08 B	22.32 M	$122.76 M	$982.5 M
Point Estimate	341.04 B	31.58 M	$173.68 M	$1.39 B
Upper Bound	441 B	40.8 M	$224.4 M	$1.79 B

Table 10: Opportunity cost to the nation over a year considering reading of brief permission screens.

Estimates	Word Count (words)	Time to read (hours)	Cost to read at Leisure	Cost to read at Work
Lower Bound	1058 B	98 M	$539 M	$4.3 B
Point Estimate	1505 B	139.4 M	$766.7 M	$6.14 B
Upper Bound	2040 B	189 M	$1.04 B	$8.32 B

Table 11: Opportunity cost to the nation over a year, considering time to read detailed permission screens.

0.4.5 Analysis of Pre-installed Applications

Pre-installed applications are the apps that already exist in the phone as system apps. Some of these apps may be third party apps, but since they were embedded by the manufacturer they are included under system apps. To realize how much time would be required in case the user has to read the permission screens, we did an analysis on the pre-installed apps. The pre-installed apps are considered as system apps, which are tagged by a flag called FLAG_SYSTEM under the Android class ApplicationInfo. We found pre-installed apps on nine devices programmatically, by checking for this flag. After examining these devices, we found the average number of pre-installed apps on a device to be 205.

Estimates	Number of pre-installed apps	Word Count (words)	Time to read (hours)	Cost to read at Leisure	Cost to read at Work
Lower Bound	189	34020	3.15	$17.32	$138.6
Point Esti-mate	205	52480	4.9	$27	$215.7
Upper Bound	220	76340	7	$38.5	$308.1

Table 12: Opportunity cost for reading permissions of pre-installed apps, considering time to read detailed permission screens for an individual.

Estimates	Word Count (words)	Time to read (hours)	Cost to read at Leisure	Cost to read at Work
Lower Bound	13.86 T	1.28 B	$7.04 B	$56.3 B
Point Esti-mate	21.5 T	2 B	$11 B	$88 B
Upper Bound	31.23 T	2.89 B	$15.89 B	$127.2 B

Table 13: Opportunity cost to the nation over a year, for reading permissions with details of per-installed apps.

Table 12 shows the results for time required to read the permission screens and opportunity cost for the same. Reports on smartphone research suggest that globally there have been 787.2 million android smartphone shipments and the United States is responsible for around 52 percent of the Androids' global smartphone share [47]. Thus there are 409 million Android smartphones in United States which will have pre-installed apps. Table 13 shows the opportunity cost evaluation for the entire nation for reading the permissions of pre-installed apps.

0.4.6 Cost to Read Additional Mobile App Privacy Policies

As mentioned earlier, some developers provide a link to companys privacy policy. We also examined how much time it would require to read these privacy policies. The privacy policy gives all the details of every permission their app is using and even informs the user about the data usage policy of the company. Data usage includes, but is not limited to, whom the company shares data with, how long does it retain users' data, which accounts are accessed, and so on. These privacy policies are lengthy and

may take significant amount of users time. Linking companys privacy policy to the app is just a part of FTC guideline and not mandatory on the developers. Also the company decides what information of privacy policy should be shown or kept from users, hence the lengths of privacy policies vary largely.

Estimates	Word Count (words)	Time to read (hours)	Cost to read at Leisure	Cost to read at Work
Lower Bound	59475	5.5	$30.25	$242.1
Point Esti-mate	145440	13.5	$74.3	$594.3
Upper Bound	360580	33.4	$183.7	$1470.3

Table 14: Opportunity cost to an average user for reading the privacy policy, considering time to read entire privacy policy with all details.

Estimates	Word Count (words)	Time to read (hours)	Cost to read at Leisure	Cost to read at Work
Lower Bound	349 T	32.3 B	$177.65 B	$1.4 T
Point Esti-mate	855 T	79.6 B	$437.8 B	$3.5 T
Upper Bound	2120 T	196.29 B	$1.07 T	$8.64 T

Table 15: Opportunity cost to the nation over a year, considering time to read privacy policies with all details.

Table 14 shows the cost for reading the privacy policies on the company websites. The word count in the table is the annual word count, obtained using apps downloaded in a year from table 3 and the quartiles of word count of privacy policies. Thus, cost to an average user to just read these privacy policies would be between $74 to $594. Table 15, provides the annual opportunity cost projections to the nation for reading privacy policies provided by developers. The nation bears an opportunity cost ranging between $437.8 billion to $3.5 trillion on an annual basis.

0.4.7 Privacy Toolkit

In keeping with the code of conduct of NTIA for privacy policies, Lookout [48] came up with an open sourced privacy policy making toolkit, Private Parts. Private Parts focuses on making privacy disclosures clear and transparent by efficiently organizing different parts of privacy policy into a simple permission interface.

The toolkit allows developers to keep the permission interface simple while still providing users sufficient information about the resources accessed by the app. The interface would also have a separate section on shared data, which informs users, with

whom the data is shared. Additional information is still available for curious users who need to know exactly what their app is capable of doing. The permission interface generated using this toolkit categorizes the permissions and identifies these categories with easily recognizable identifiers. This makes it easy to read the permissions and additionally it would be helpful for users recognition ability, in that, users would have to spend less time when they encounter similar permissions again and therefore less opportunity cost.

0.4.8 Summary

We estimated the time spent by users to go through the permission screens and also the cost involved in doing so. The time spent on a permission screen is a function of reading rate and the number of words seen on the screen. We started by evaluating the word count of permission screens for regularly downloaded apps from the Google Play Store and the time required for reading. We provide estimate for both the time required for reading brief descriptions and the time required for reading detailed explanations of permissions as well. We also find the opportunity cost for reading the permission screens with and without the details of the permissions. Additionally, we also analyzed the cost involved in reading the permission screens of pre-installed apps and the complete privacy policy provided by the developers on their website. All these factors together provide a total cost estimate for reading permission screens of all apps the users have on their smartphones. We also present our findings on opportunity cost for reading permission screens for an example nation (United States). Finally to check the impact of users' ability to learn permissions we measure actual reading times which are then contrasted with the computed times and same is done for the opportunity costs.

Categories for permissions
Hardware
Internet/Network
Application Functionality
Calls
Messages
I do not know

Table 16: Users were presented with these categories. The categories were simple enough to identify the permissions.

0.5 Preliminary Lab Study

It is a fact that reading same text again and again causes readers to skim through the content due to their ability of recollection. Ritter et al. [49] have studied the effect of learning curve on the tasks performed. They found that ability of users performing tasks depends on their cognitive model governed by the power law of practice. The same applies while reading the permissions as well. The users develop their cognitive model with more repetitions of trials, these trials being reading permissions of apps. To find how the users make use of their ability of recollection we designed a lab study. The main motive of this study was to find out if the actual reading time is different than the time computed using average reading rate. The study would show if the time to read would be affected by users ability to remember the permissions from previous encounters with similar permissions.

0.5.1 Method

As the users come across the permissions more and more frequently they tend to skim through the permissions. To test this behavior and the users' ability of remembering the permissions we performed a study that would require users to read permissions of applications assuming back to back installations of apps. The study was meant to find out the time required by the users to read the permissions when they already have read similar permissions earlier. As opposed to time computed using method in Chapter 3, this time would reflect the users' ability to learn the permissions and skim through them.

Each participant was presented with an apps' permissions. The participant was then asked to categorize these permissions into categories shown in table 16. Application Functionality is category which would contain permissions that users think are required for special functionality in that app. Rest of the categories were simple enough to indentify the permissions. The participants also were given the freedom to add a new category while they were categorizing the permissions. The participants had to categorize the permissions based on their knowledge and assumptions of the permissions. Since the test was to verify the users cognitive model, users putting permissions under an unrelated category did not make any difference.

Although users had freedom while making decisions on the permissions, there were certain rules followed for standardization. All the participants in the study were made to perform the study on 4.7 inch screen sized smartphone. All the apps used for study were messaging services. Users were only presented permissions of apps they had not used before. To ensure that length of the permission does not affect the time taken to read, all the apps presented were selected such that they had similar permission lengths. Users were also made to identify the permissions by simply indexing the permissions by numerals, so that writing speeds do not affect the recorded times.

The study was divided into 2 phases. The first phase required the participant to categorize the permissions of any one of the apps shown in table 17. During this phase the participant could ask questions regarding any category or permissions. The first phase was to make the participant familiar with the procedure. After this exercise in second phase, they were asked to repeat the procedure for three other applications and these three iterations were timed. The exercise was repeated three times since the upper bound on app downloads is 9 per month, table 3, and hence about 3 apps per week. The actual installation rate may vary depending on the type of user.

0.5.2 Results

From the study we found the average time taken for reading the permissions when the participants already had some prior knowledge on the permissions. The average of the reading times taken by the participants was found to be 1.43 minutes. The first

Applications
Facebook Messenger
Whatsapp
myPeople Mesenger
Go SMS PRO
BBM
Viber
Tango Messenger
WeChat

Table 17: Applications used for the study. All the applications are messaging services.

iteration of the exercise produced an average reading time of 1.81 minutes. The second iteration average was 1.39 minutes and the third iteration average was 1.10 minutes. Out of the 11 participants, some of the more knowledgeable participants created new categories such as App Priority, Settings, Information, Application Description and Operating System, while some were surprised to find the number of permissions used by a messaging service application. Table 18 shows the annual opportunity cost based on the reading times obtained from the lab study. The average difference between the first and second lap times is 0.42 minutes and the average time difference between second and third laps is 0.29 minutes. This shows that there is gradual decrease in the time taken to read the permissions in quick succession. The results also confirm that participants spent less time reading permissions that they had already encountered, thus reducing the overall time spent on reading permissions. Out of 11 participants 6 were regular Android users, 4 were iphone users who also owned Android devices and hence were familiar with installation process, and 1 new Android user who had never read Android permissions before. Thus the data obtained is well distributed amongst different types of users. Table 19 shows the opportunity cost for the entire nation(United States). The values in this table are computed using 5.88 billion app downloads in a year in United States.

Estimates	Time to read	Read at Leisure	Read at work
Lower Bound	1.19 hours	$6.54	$52.38
Point Estimate	2.4 hours	$13.2	$105.64
Upper Bound	4.47 hours	$24.58	$196.76

Table 18: Opportunity cost for reading permission screens in one year. The value of cost represented accounts for users ability of recollection.

Estimates	Time to read(hours)	Read at Leisure	Read at work
Lower Bound	107.8 M	$592.9 M	$4.74 B
Point Estimate	140.1 M	$770.5 M	$6.16 B
Upper Bound	177.3 M	$975.1 M	$7.8 B

Table 19: Opportunity cost to nation for reading permission screens. The value of cost represented accounts for users ability of recollection.

0.5.3 Inference

Table 6, shows that time to read the permissions word by word would require 1.4 minutes for one app with an upper bound of 1.9 minutes. And table 9 shows the opportunity cost for reading the permissions in detail. Comparing these tables with the reading time obtained from the study and the opportunity cost seen in table 18, we can see that in both cases the point estimate ranges between $13 to $105, hence we conclude that the users ability of remembering previously read permissions is not

significant to the time taken in reading the permissions. This means that time required for reading the permissions with prior knowledge is just as same as reading the complete permissions. This shows that, the users spend the same amount of time reading the permissions when they have some knowledge or assumptions about permissions, as they would when they just read the permissions word by word. The iterations in the study were successive assuming that users would install 3 apps in a week on the same day. However, the actual gap in between two installations can vary between 1 to 3 days. Thus the actual upper bound in the reading time would be higher than the one shown in table 18, owing to the introduction of forgetting curve in the users cognitive model.

0.6 Discussion

Our results in this paper depend on Google Play app market values and reports from analysis firms that date from 2012 onwards. The Google Play Store has seen a surge of downloads since 2012, owing to the improved sales of Android devices. The results may be higher if calculated as per the current values. The United States spends 31.58 million hours or value equivalent to $173.68 million. This value is for reading the brief description of permissions of only the permissions of applications that the user downloads. Most of the users will not understand the permissions in a single read and hence may have to read more than just once, which means higher cost. The lab study that was performed to understand users learning curve and find the time taken to read is based on their knowledge of the permissions. A gap of few days in between installations may change users' knowledge about the permissions. Also the participants were presented with similar permissions since all the apps were messaging services. This may be the reason for faster reading or skimming through the permissions. Androids' permissions and their descriptions are subject to change, this may also be a factor affecting the time to read. In reality someone with better understanding of permissions may spend much lesser time than others who have limited technical background or are new Android users. A symbolical representation of permissions in the form of indicators would be much more effective for the users cognitive model. Since remembering indicators is an easier task than remembering text. A combination of simple text with effective indicators would be a great way to convey users' about the permissions an app is using while simultaneously reducing the time taken to read permissions and thereby the opportunity cost as well.

0.6.1 Other Platforms and Privacy Disclosures

The readability of permission screens also depends on the privacy disclosure technique. Different platforms handle this differently. Android displays the permission screen once before installing an app, but in iOS, there are only some specific permissions (such as location) requested during first time using an app. Blackberry also shows permission

screen before installing but similar to Android shows brief descriptions of permissions, however, it allows the user to disable a particular permission. If users are allowed to choose from required permissions it may affect their choice of reading the permissions, since they have complete control over the permissions that app can use. Another thing that might affect the decision to read a particular permission is the time of disclosure, as seen in the case of iOS. The rationale behind this is that, if the users can associate a particular permission with an apps behavior they will not feel the need to read it every time. There might also be some side loaded apps on Android devices. Side loaded apps are the ones that are installed on the phone either via Amazons' app store or downloaded manually from the web. For such apps there is no detailed explanation of permissions but just the brief description of permissions depending on the Android version the device is running, so if the user desires to get more information about the permissions he has to manually search for them over the web adding to the opportunity cost.

0.6.2 Alternatives to reading

In the recent past, NTIA has attempted to create innovative user interface that would make it easier for users to understand the resources accessed by an application [5]. These user interfaces can inform the user, what data is collected and with whom is the data shared. This simplifies the process of users interaction with the permissions' model, and we think it will affect the opportunity cost for reading the permissions as well. If the explanation of these permissions are made simple enough for users with limited technical background it might lessen the required time and hence reduce the opportunity cost as well.

So far our discussion has been about the time people would invest to secure their privacy on mobile devices by reading the permissions, but again there are those who would rather pay for apps that can protect their privacy, for example, there are apps that can act as anti-virus agents and also protect users' private data. Users have the option to just ignore reading all the permissions by trusting such applications.

We acknowledge that users cannot be expected to read the permission screen of

each and every application, since that is not the primary objective, but installing and using that app is the users' motive. We think that the user needs to be hinted to read the permissions only in certain cases. We propose certain modifications to the interface of the Play Store. The users should be able to set a profile of their expected privacy settings when they setup their accounts. Now when the users searches for an app and finds one, a green flag should be shown if the app uses permissions in tune with the users expectations and a red flag otherwise. There should be an icon that informs user what permission caused the flag to go red, thus helping him make a decision. We also suggest that there should be a data sharing icon informing the user who the data is shared with, thus, in case the red flag is for data sharing the user can still make an informed decision. A profile based permission setting option for the play store interface could help in reducing the expected time to be spent on the permissions, since users would only need to read in case the permission is a new permission or it is not in tune with their expectations from the app. Previous attempts in designing effective indicators for easy identification of permissions still require users to read permissions, unlike our proposed solution in which permissions would be flashed only if it is new.

0.6.3 Opportunity cost and mobile ad revenue

Google's contribution to mobile ad revenue in United States is approximately $1 billion per year or higher [50, 51]. According to our findings reading permission screens in a year has opportunity cost ranging from $766 million to $6.14 billion. Most of the users will not understand the permissions in a single read owing to its technical contents and hence would read it more than once, thus bearing more cost. The opportunity cost is therefore, slightly higher or on par with the revenue of industry that exploits privacy the most. Average opportunity cost of reading the permissions is $6.14 billion as seen in table 11. Thus, to achieve a mobile ad revenue of $1 billion, users in United States have to bear opportunity cost of $6.14 billion. The current situation shows that the nation is bearing a cost six times the revenue generated by the industry relying on the android apps. A profitable scenario would be a lower opportunity cost to users or in other words lower time spent in reading permissions. This will be achieved by creating

concise permissions, but care needs to be taken to keep the permissions simple and informative as well.

0.7 Conclusion and Future Work

A person's mobile device holds his private data and might even hold private data of people they know in the form of messages, mails, pictures or videos. There are different ways in which this data can be accessed. Privacy related permissions allow access to digital data as well as hardware controls, like camera access, access to device sensors, etc. However to know these permissions the user has to go through the privacy policy of that application, which involves spending significant amount of time as our results show. Not only that but our findings show that this time spent on the policies has a significant amount of opportunity cost based on whether it is read during leisure or at work. Whether or not to bear this opportunity cost in order to protect self privacy on mobile devices is an individuals' choice. For someone who is only concerned about an application getting access to his audio settings or display settings it may not be worth investing in reading privacy policy. But knowing that a permission as simple as allowing complete Wi-fi access can make a mobile device vulnerable to privacy threats might make the user rethink his decision to ignore the permission screen. In the long run it is always better to know what your app is capable of doing with your device, and hence spending some time on reading privacy policy is recommended. According to the point estimate we found a user would spend approximately one minute to read the permissions behind every application, which is a reasonable bargain.

Users are presented with permission screen before installing an Android app and have to make an informed decision after reading these permissions. We evaluate the time spent by users if they have to read all the permission screens of all apps on their smartphones. Our findings show that annually users might end up spending a minimum of half an hour and maximum of two and half hours in reading, which has a minimum cost of $3 and a maximum of $106. The cost varies in the specified range based on, where the user read the permission screen and whether the permission details were read. Similar to the opportunity cost to an average smartphone user the annual cost to nation was found to vary over a wide range with a minimum of $174 million and a maximum of $6 billion. We also provide estimates for reading permission screens of pre-installed

apps. Users will bear a one time cost between $27 to $216 if they have to read the permissions for these apps with the permission details. We estimate the opportunity cost for reading developers' privacy policies, which ranges from $74 to $594 annually. From our lab study we also conclude that the users ability to remember the previously read permissions does not impact the annual time spent on reading permissions.

0.7.1 Future Work

The lab study conducted for getting the actual reading times was on a rather smaller scale with 11 participants. A similar study on larger scale would give a better understanding of the users learning curve regarding knowledge of permissions. This would also require a study of general categories to present to the participants so that they can easily categorize the permissions. Also to compute annual cost a similar study on other Android devices of varying screen sizes needs to be done. Such an extensive study in this field would help developers to come up with better permission models and improve the permission interface as well. Permissions are a way of conveying the capability of the app to the users. Users also generally use more than one device and hence would be required to read permissions on other devices as well. Similar extensive study on other platforms would reveal a more accurate overall opportunity cost not only to an average user but to a nation as well.

References

[1] Nielsen. Mobile majority: U.S. smartphone ownership tops 60%, June 2013. http://www.nielsen.com/us/en/newswire/2013/mobile-majority–u-s–smartphone-ownership-tops-60-.html.

[2] Android Developers. Manifest.permission. http://developer.android.com/reference/android/Manifest.permission.html.

[3] BBC News. Social apps 'harvest smartphone contacts', February 2012. http://www.bbc.co.uk/news/technology-17051910.

[4] Federal Trade Commission. Mobile privacy disclosures: Building trust through transparency. `http://www.ftc.gov/os/2013/02/130201mobileprivacyreport.pdf`, 2013.

[5] National Telecommunications and Information Administration. Mobile app transparency. `http://www.ntia.doc.gov/other-publication/2013/privacy-multistakeholder-process-mobile-application-transparency`, 2013.

[6] Nathaniel Good, Rachna Dhamija, Jens Grossklags, David Thaw, Steven Aronowitz, Deirdre Mulligan, and Joseph Konstan. Stopping spyware at the gate: a user study of privacy, notice and spyware. In *Proc. SOUPS '05*, pages 43–52, New York, NY, USA, 2005. ACM.

[7] Carlos Jensen and Colin Potts. Privacy policies as decision-making tools: an evaluation of online privacy notices. In *Proc. CHI '04*, pages 471–478, New York, NY, USA, 2004. ACM.

[8] Aleecia M McDonald and Lorrie Faith Cranor. Cost of reading privacy policies. *ISJLP*, 4:543, 2008.

[9] Gary S Becker. A theory of the allocation of time. *The economic journal*, 75(299):493–517, 1965.

[10] Alessandro Acquisti and Jens Grossklags. Privacy and rationality in individual decision making. *Security & Privacy, IEEE*, 3(1):26–33, 2005.

[11] Tony Vila, Rachel Greenstadt, and David Molnar. Why we can't be bothered to read privacy policies models of privacy economics as a lemons market. In *Proc. ICEC '03*, pages 403–407. ACM, 2003.

[12] Michael Becher, Felix C. Freiling, Johannes Hoffmann, Thorsten Holz, Sebastian Uellenbeck, and Christopher Wolf. Mobile security catching up? revealing the nuts and bolts of the security of mobile devices. In *Proc. SP '11*, pages 96–111, Washington, DC, USA, 2011. IEEE Computer Society.

[13] Jonathan Anderson, Joseph Bonneau, and Frank Stajano. Inglourious installers: Security in the application marketplace. In *Proc. WEIS '10*, June 2010.

[14] Patrick McDaniel and William Enck. Not so great expectations: Why application markets haven't failed security. *IEEE Security and Privacy*, 8:76–78, September 2010.

[15] Peter Gilbert, Byung-Gon Chun, Landon P. Cox, and Jaeyeon Jung. Vision: automated security validation of mobile apps at app markets. In *Proc. MCS '11*, pages 21–26, New York, NY, USA, 2011. ACM.

[16] David Barrera and Paul Van Oorschot. Secure software installation on smartphones. *IEEE Security and Privacy*, 9:42–48, May 2011.

[17] D. Wetherall, D. Choffnes, B. Greenstein, S. Han, P. Hornyack, J. Jung, S. Schechter, and X. Wang. Privacy revelations for web and mobile apps. In *Proc. HotOS '13*, pages 21–21, Berkeley, CA, USA, 2011. USENIX Association.

[18] Erika Chin, Adrienne Porter Felt, Kate Greenwood, and David Wagner. Analyzing inter-application communication in android. In *Proc. MobiSys '11*, pages 239–252, New York, NY, USA, 2011. ACM.

[19] William Enck, Damien Octeau, Patrick McDaniel, and Swarat Chaudhuri. A study of android application security. In *Proc. SEC '11*, pages 21–21, Berkeley, CA, USA, 2011. USENIX Association.

[20] Adrienne Porter Felt, Kate Greenwood, and David Wagner. The effectiveness of application permissions. In *Proc. WebApps '11*, pages 7–7, Berkeley, CA, USA, 2011. USENIX Association.

[21] Adrienne Porter Felt, Erika Chin, Steve Hanna, Dawn Song, and David Wagner. Android permissions demystified. In *Proc. CCS'11*, pages 627–638, New York, NY, USA, 2011. ACM.

[22] Adrienne Porter Felt, Helen J. Wang, Alex Moshchuk, Steven Hanna, and Erika Chin. Permission re-delegation: Attacks and defenses. In *Proc. SEC '11*, August 2011.

[23] Kathy Wain Yee Au, Yi Fan Zhou, Zhen Huang, and David Lie. Pscout: analyzing the android permission specification. In *Proc. CCS '12*, pages 217–228. ACM, 2012.

[24] Xuetao Wei, Lorenzo Gomez, Iulian Neamtiu, and Michalis Faloutsos. Permission evolution in the android ecosystem. In *Proc. ACSAC '12*, pages 31–40. ACM, 2012.

[25] Machigar Ongtang, Stephen McLaughlin, William Enck, and Patrick McDaniel. Semantically rich application-centric security in android. In *Proc. ACSAC '09*, pages 340–349, Washington, DC, USA, 2009. IEEE Computer Society.

[26] Machigar Ongtang, Kevin Butler, and Patrick McDaniel. Porscha: policy oriented secure content handling in android. In *Proc. ACSAC '10*, pages 221–230, New York, NY, USA, 2010. ACM.

[27] Asaf Shabtai, Yuval Fledel, and Yuval Elovici. Securing android-powered mobile devices using selinux. *IEEE Security Privacy Magazine*, 8(3):36–44, 2010.

[28] William Enck, Machigar Ongtang, and Patrick McDaniel. On lightweight mobile phone application certification. In *Proc. CCS '09*, pages 235–245, New York, NY, USA, 2009. ACM.

[29] William Enck, Peter Gilbert, Byung-Gon Chun, Landon P. Cox, Jaeyeon Jung, Patrick McDaniel, and Anmol N. Sheth. Taintdroid: an information-flow tracking system for realtime privacy monitoring on smartphones. In *Proc. OSDI '10*, pages 1–6, Berkeley, CA, USA, 2010. USENIX Association.

[30] Manuel Egele, Christopher Kruegel, Engin Kirda, and Giovanni Vigna. Pios: Detecting privacy leaks in ios applications. In *Proc. NDSS '11*, February 2011.

[31] Peter Hornyack, Seungyeop Han, Jaeyeon Jung, Stuart Schechter, and David Wetherall. These aren't the droids you're looking for: retrofitting android to protect data from imperious applications. In *Proc. CCS '11*, pages 639–652, New York, NY, USA, 2011. ACM.

[32] Alastair R. Beresford, Andrew Rice, Nicholas Skehin, and Ripduman Sohan. Mockdroid: trading privacy for application functionality on smartphones. In *Proc.HotMobile '11*, March 2011.

[33] Jialiu Lin, Shahriyar Amini, Jason I. Hong, Norman Sadeh, Janne Lindqvist, and Joy Zhang. Expectation and purpose: Understanding users mental models of mobile app privacy through crowdsourcing. In *Proc. UbiComp'12*, September 2012.

[34] Patrick Gage Kelley, Lorrie Faith Cranor, and Norman Sadeh. Privacy as part of the app decision-making process. In *Proc. CHI '13*, pages 3393–3402. ACM, 2013.

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

[36] Martina Ziefle. Effects of display resolution on visual performance. *Human Factors: The Journal of the Human Factors and Ergonomics Society*, 40(4):554–568, 1998.

[37] Ralf Biedert, Andreas Dengel, Georg Buscher, and Arman Vartan. Reading and estimating gaze on smart phones. In *Proc. ETRA '12*, pages 385–388. ACM, 2012.

[38] Nielsen. State of the appnation, a year of change and growth in u.s. smartphones. `http://www.nielsen.com/us/en/newswire/2012/state-of-the-appnation-a-year-of-change-and-growth-in-u-s-smartphones.html`, 2012.

[39] Research2guidance. Android market insights, volume 7, 2011.

[40] Admob. Admob mobile metrics report, 2010.

[41] Ronald Eugene Kmetovicz. *New product development: design and analysis*. Wiley-Interscience, 1992.

[42] Timothy Leunig. Time is money: a re-assessment of the passenger social savings from victorian british railways. *Journal of Economic History*, 66(3):635, 2006.

[43] Bureau of Labor Statistics. Occupational employment statistics. `http://www.bls.gov/oes/current/oes_nat.htm`, 2013.

[44] Engadget. Google play hits 600,000 apps, 20 billion total installs. `http://www.engadget.com/2012/06/27/google-play-hits-600000-apps/`, 2012.

[45] Androidauthority. Google: 900 million android activations, 48 billion app installs. `http://www.androidauthority.com/google-io-android-activations-210036/`, 2013.

[46] App Annie. App annie index: Japan overtakes u.s. for google play revenues. `http://blog.appannie.com/app-annie-index-november-2012/`, 2012.

[47] IDC. Idc reports smartphone shipments. `http://www.idc.com/getdoc.jsp?containerId=prUS24676414`, 2013.

[48] Lookout. `https://blog.lookout.com/blog/2014/03/12/open-source-privacy-policy/`, 2014.

[49] Frank E Ritter and Lael J Schooler. The learning curve. *International encyclopedia of the social and behavioral sciences*, 13:8602–8605, 2001.

[50] Pew Research Center. Digital: As mobile grows rapidly, the pressures on news intensify. `http://stateofthemedia.org/2013/digital-as-mobile-grows-rapidly-the-pressures-on-news-intensify/`, 2013.

[51] Emarketer. Facebook to see three in 10 mobile display dollars this year. `http://www.emarketer.com/Article/Facebook-See-Three-10-Mobile-Display-Dollars-This-Year/1009782`, 2013.

Appendix A

Quartiles of results

Point estimate is the best guess in an event of uncertainty however we were looking for scalable solution and hence used quartiles from the available data. Quartiles split the data set at different percentiles. The first quartile is used as lower bound, the second as the point estimate and the third as the upper bound in all the cases. We select the interquartile range which provides a robust scalable range. While getting the tabular results each quartile was multiplied with respective quartile from other table to implement standardization.

Appendix B

Calculations for tables

The calculations to obtain values as listed in the tables above are as follows.

- Table 3, shows a column of applications downloaded over a year. The value is obtained as existing apps plus the number of new apps downloaded per year.
- Table 5 and table 7, presents the word count, which is number of applications downloaded times the word count of a single permission screen and time to read for the same. Table 5 uses the word count for reading brief descriptions of permission screen as given in table 4. Table 7, uses the word count for reading the entire list of permissions along with their details as given in table 6.
- Table 8 and 9, show the values of cost of reading if read during leisure or if read at work. Cost to read during leisure is the time required to read times the leisure value of hourly wage and cost to read at work is the time required to read times the work value of hourly wage.
- Table 10 and 11, show cost to the nation(United States) for reading permission screens. The word count for these tables were obtained as annual app downloads times the word count of permission screens as given in tables 4 and 6, respectively.
- Table 12 displays values of opportunity cost for reading permission screens of pre-installed apps to an average user.
- Table 14 has the opportunity cost for reading privacy policies of companies that develop apps. The word count in this table is obtained as app downloads in year from table 3, times the of word count in privacy policies. Table 15 gives the

annual cost to the nation for reading privacy policies, using 5.88 billion annual downloads as the number of apps downloaded in United States in a year.

- Table 18 shows the opportunity cost calculated using the actual reading times from the study mentioned in Chapter 5. It is calculated using the actual times from the study and the annual app downloads in table 3. Table 19 shows the opportunity cost for the nation, computed using 5.88 billion annual downloads.