Effective Risk Analysis and Risk Detection for Android Apps

Deshmukh Sharad R.
PG Student,
MBES College of Engineering,
Ambajogai

Patil B. M.
PG Dept,
MBES College of Engineering,
Ambajogai

ABSTRACT
The features and functionality of the different mobile devices has made them attractive targets for malicious applications. There are different types of risks are present in recent apps. Android's permission system is intended to inform users about the risks in installing apps. Whenever a user installs an application, he or she has the chance to review the application's permission requests and cancel the installation if the permissions are imprudent or unacceptable. Basically previous research focus on reliance on users is not so effective, as most of the users don't understand the permission information. Actually in this work focused on the permission of the particular application. In this work here proposed a system to provide summary risk communication to user in friendly manner which is easy to understand. Finally in this work introduced how to relate risk permissions and risk rating with the risk analysis by using risk score and risk value.

Keywords
Apps permissions, Risk value, and Risk Analysis

1. INTRODUCTION
Now a day's smart mobile devices are use in large scale due to its popularity and functionality. Smart phones and mobile devices have become explosively popular for personal as well as business use in recent years. Our entire things related to personal or it may be business related are often stored on the devices, which contain contact lists, email messages, passwords, confidential information and access to files stored locally and in the cloud. With the advent of smart phones, users are, knowingly or not, carrying more and more private information around with them on their phones. This information ranges from the location of the device to the reading habits of the user and even his or her bank details. While attacks on mobile devices have largely focused on earning the attacker quick cash by sending text messages to or calling premium numbers, the focus has shifted towards stealing the private data contained on the devices. Possible access of such a type of information by unauthorized person puts users at risk. As the Android platform has grown to take one of the largest shares of the smart-phone market, the platform has become the prime target for criminals seeking the private data the users are carrying around with them. At the same time, the security of the platform has come under scrutiny from security professionals. Malicious software is a common problem for every software platform, and the Android platform is no exception. Since the First malicious Android application was discovered in 2010, the number of malicious applications has been consistently rising.

1.1 Objectives
- To provide solution that manage a method to assign a risk score to each installed app and display a brief of that information to users in friendly manner which is easy to understand. So that user can identify potentially risky apps.
- To provide the solution for Pileup flaws, by sending notification to users whenever the apps gets auto update and gains some extra permission without user consent. This will alert the user about apps abnormal behavior and he can accordingly decide whether to uninstall or keep the app.
- To provide the functionality to uninstall the selected app if user finds it malicious.
- To provide the functionality to block the specific permissions of the selected app if user finds it malicious or defected.

2. LITERATURE REVIEW
Android has existed publicly since 2008. A significant amount of work has been conducted on studying the Android permissions as well as security model. A lot of this work concentrates on creating theoretical formalizations of how Android security works or presents improvements to the system security.

- 2011:Felt, Greenwood and Wagner contributes evidence in support of application permission systems Out of 1000 only 15 Google Chrome extensions are used native code, which is the most dangerous as well as unprotected privileges. Approximately 30% of extension developers restrict their extensions' web access to a small set of domains. All Android applications ask for less than half of the available set of 56 Dangerous permissions, and a majority request less than 4. [2]

- 2012: Chin, Felt, and Sekary suggest that the smart phone ecosystem application vendors, application markets, and usage a pattern is relatively new as well as quite different from traditional desktop computing. They find that participants often install a huge number of applications from unfamiliar brands without reading the applications' privacy policies, which likely involves to their mistrust of applications. [2]

- 2012: Kevin et.al Juang, focused on participants remembered their passwords significantly better using the system-generated mnemonic condition compared to the other conditions. They also found that participants gave our system the highest overall usability ratings. [7]

- 2012: Kelley et.al focused on Android permissions. According to them Users do not understand Android permissions. Basically the human readable terms monitored before installing an application are at best vague, and at worst confusing. Generally users are not currently well prepared to make informed privacy and

security decisions around installing applications from the Android market. [8]

- 2013: Kelley, Cranor, and Sadeh focused on smart phones have unprecedented access to sensitive personal information. Whenever users are aware of this, basically, they do not considering privacy when they select applications to download in the application marketplace. They created a small "Privacy Facts" display, which they then tested in 20 in lab exercises and an online test of 366 participants. They found that collecting information to the user when they are making the decision and by presenting it in a clearer fashion; they can assist users in making more privacy-protecting decisions. [11]

- 2013: Varga and Muska invent the new method of informing user whether applications installed on his device are potentially harmful or not. Another concept is the evaluation of achievements reached by their methodology implemented in standalone application.

- 2013: Benton et al. confirm that as users are unaware of the implications of the requested permissions, permission requests appear to be ineffective. Low rates of user attention and comprehension suggest that significant work is needed to make the Android permission system widely usable. [12]

- 2013: The survey results of Mylonas et al. are not in line with the expectation of smart phones' security models for a reasonable security aware user. In contrast, they suggest that users are not adequately prepared to make appropriate security decisions. [13]

3. EXISING SYSTEM

In existing system, risk communication mechanism which warns the user about the permissions an app requires before the app is installed by the user, trusting that the user will make the right decision. The specific approach used in Android has been shown to be ineffective at informing users about potential risks. The majority of Android apps request multiple permissions. When a user sees what appears to be the same warning message for almost every app, warnings quickly lose any effectiveness as the users are conditioned to ignore such warnings.

3.1 Problem Statement

- Privacy Impact: The First problem addresses the issue of users not being aware of permissions and having difficulty comprehending them. The many researchers have shown [12] [13] that conclusion can be drawn from the permissions that applications request**.**

- Permission Usage: The second problem addresses the issue of in transparency. Android permissions are only displayed at install-time. After the user grants access to the requested permissions, the application has unlimited access to them, but the user has no insight into how the application is actually using those permissions.

- In the present system allows malicious or objectionable application, reports the risk in user friendly manner and warnings quickly lose any effectiveness as the users are conditioned to ignore such warnings.

3.2 Problem Definition

How to improve the usability of the Android permission system in order to increase user awareness and comprehension with respect to privacy?

4. PROPOSED SYSTEM

In order to overcome these drawbacks, the concept of risk scoring function which assigns each app a numerical score which is in the form percentage, that indicates how risky the app is. This approach presents "comparative" risk information that each app's risk is presented in a way so that it can be easily compared to other apps. However, we believe that it is better to use a risk scoring function for risk communication in the following way. Given this function, one can compute a risk ranking for each app, identifying the percentile of the app in terms of its risk score. This percentile number has a well-defined and easy-to-understand meaning. This ranking can be presented in a more user-friendly fashion, e.g., translated into categorical values such as high risk, medium risk, and low risk, an important facility of the mobile app ecosystem is that users often have choices and alternatives when choosing an android app. If the user knows that particular app is significantly more risky than another with similar functionality, then that may cause the user to select the less risky one. This approach complements well other approaches that try to identify malicious apps. After malicious or objectionable apps are removed, the remaining ones can be ranked according to their risks.

4.1 Methods and its Evaluation

- Risk Score: Basically present research focused on one possible scoring mechanism grounded in machine learning and based off permission requests. In this paper we can easily determined how to calculate risk score of particular apps using following equations.[1]

  RS=TP-SP…… … (4.1)

  Where,

  - RS=Risk Score
  - TP=Total number of permissions
  - SP=Selected permissions of particular apps.

- Risk Value: According to the calculation of Risk score we can also easily determined Risk value. This is illustrated in the following equation.

  RV=TP-RS…… (4.2)

  Where,

  -RV=Risk Value

  -TP=Total number of permissions

  -RS=Risk Score

- Risk Analysis: According to the user rating we can easily find out the risk in installed apps by using whatever user gives rating to the installed apps. Basically rating is in the form of filled stars. Here we can easily analyze more filled stars means less risky app and less filled stars means high risky app. This can be shown here following expression.

High Risk	if Selected Stars<3……..…. (4.3)
Medium Risk	if Selected stars=3
Low Risk	if selected Stars>2

4.2 Proposed System Architecture

In the following Figure.1 shows how actually flow of proposed system. Generally user installs the apps from the Google store. After that here follows the following steps according to the architecture.

4.2.1 Implementation Details

- *Getting app from Google store:*

Generally user installs the apps from the Google store. Firstly select the app which user wants to install on their android phone.

- *Getting permissions of selected app:*

Basically selected apps ask to the user for specific permissions while installing apps. User selects different permissions according to their device convenience.

- *Gives rating to installed apps:*

After installing apps on android phones then user use that app and according to their functionality user gives rating to that app.

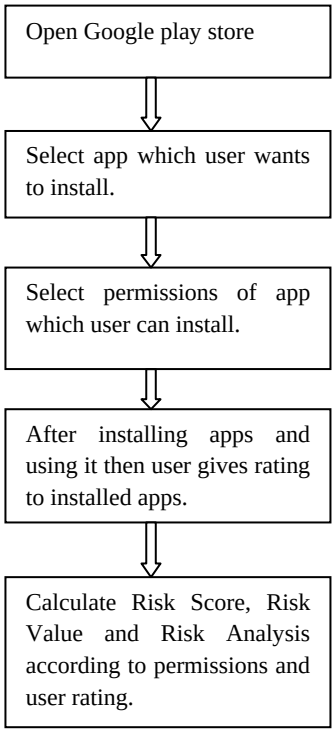

Figure 1: Proposed system architecture

- *Calculate Risk score, Risk Value and Risk Analysis:*

According to the permissions and user ratings we can easily determine risk score, risk value and risk analysis using equation 4.1, 4.2, 4.3.

5. RESULTS AND DISCUSSION

Basically here use different parameters i.e. permissions and user rating to evaluate risk score and risk value. All these parameters are analyzed and their performances are shown in Table 1.In the following Table 1.shows app name, total number of permissions, selected permissions, risk score, risk value and user rating as parameters. Here consider only five apps for analysis and five permissions. Among this five permissions user selects which they want. According to their selection here calculates risk score and risk value. After using the installed app user gives the user rating to that app. Here consider if user select 3 permission then here easily calculate risk score and risk value according to given equation 4.1 and 4.2 and analyze how much risk present in that app.

Table 1. Evaluation of Risk Analysis

App Name	Total Number of Permission	Selected Permissions	RS=TP-SP		RV=TP-RS		User Rating
			Risk Score	Risk Score in %	Risk Value	Risk Value in %	
Whats App	5	3	2	40%	3	60%	5
Twitter		1	4	80%	1	20%	4
Messenger		4	1	20%	4	80%	3
Face book		3	2	40%	3	60%	5

According to the above parameters we can easily analyze the risk analysis in the installed apps. The following graph shown how to determine risk value and risk score according to equation no 4.1 4.2 4.3.Using these formulas here illustrates following results i.e. shown in the following Figure.2 and Figure 3.

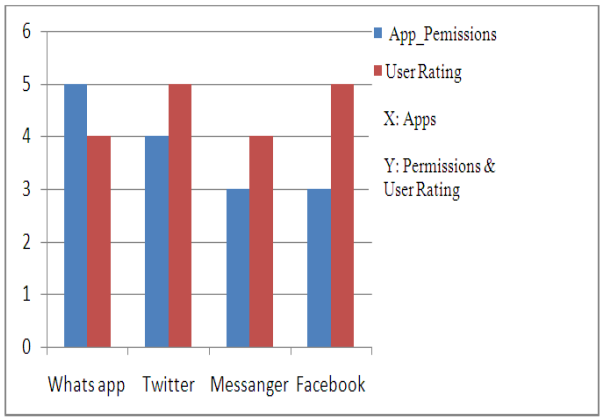

Figure 2: Low risk evaluation

Here in the above Figure 2 shows apps as x-axis and permissions as well as user ratings as y-axis. In Figure 3 shows High risk evaluation which is also be calculated using equation no 4.1 4.2 and 4.3.

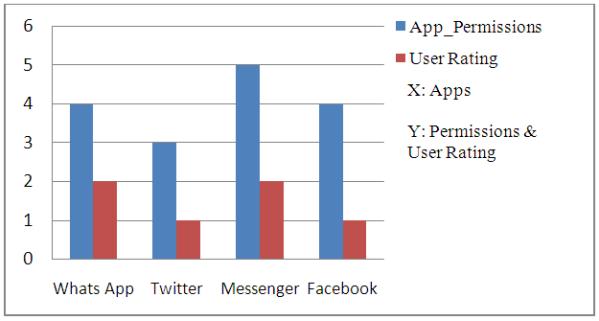

Figure 3: High risk evaluation

In the following Figure 4 illustrates how to analyze the installed apps are good or bad to our android phone. This can be achieved by using ratings and permissions which is given by different users.

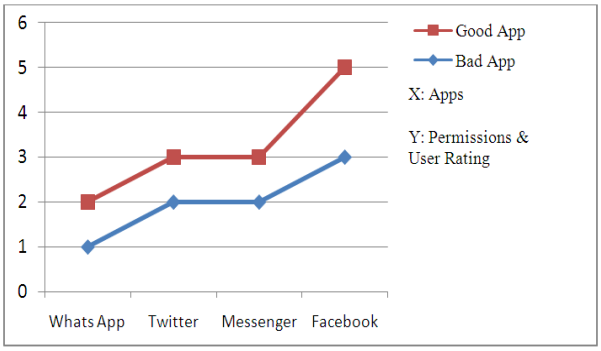

Fig 4: Risk analysis of different apps

6. CONCLUSION

Basically in this work focused on the permission of each android app. Because of number of users are ignoring the permission, but this is harmful to our mobile devices. This causes unwanted things like break the security of our mobile phones or else this can affect on our sensitive information. Whenever updated versions are installed on our mobile phones at that time we don't know the apps may be malicious or not. Actually in this work concentrate on selected permission as well as user rating and according to that factor here analyze the risk in that app. According to this analysis, we can easily detect the particular apps is malicious or not.

7. ACKNOWLEDGEMENT

I would like to sincerely express thanks to my guide Dr. B. M. Patil for his appreciable support, continuous encouragement and his invaluable suggestions. I am grateful for all the suggestions and hints provided by him. It was great moment to work with him.

8. REFERENCES

[1] Christopher S. Gates, Jing Chen, Ninghui Li, Senior Member, IEEE, and Robert W. Proctor, "Effective Risk Communication for Android Apps" IEEE Transaction on Dependable and secure computing, vol.11, no. 3, May-June 2014

[2] A.P. Felt, K. Greenwood, and D. Wagner, "The Effectiveness of Application permissions",proc. Second USENIX Conf. Web Application Development (WebApps '11), 2011.

[3] XF. Xie, M. Wang, R. Zhang, J. Li, and QY. Yu, "The Role of Emotions in Risk communication," Risk Analysis, vol. 31, no. 3, pp. 450-465, 2011.

[4] I. Rassameeroj and Y. Tanahashi,Various approaches in analyzing android applications with its permission-based security models," in Electro/Information Technology (EIT), 2011 IEEE International Conference on, pp. 1-6, 2011.

[5] E. Chin, A.P. Felt, V.Sekar, and D. Wagner, "Measuring User Confidence in Smartphone Security and Privacy," Proc. Eighth Symp. Usable Privacy and Security (SOUPS '12), pp. 1-16, 2012.

[6] A.P. Felt, E. Ha, S. Egelman, A. Haney, E. Chin, and D. Wagner, "Android Permissions: User Attention, Compréhensions, and Behavior," Proc. Eighth Symp. Usable Privacy and Security, 2012

[7] K.A. Juang, S. Ranganayakulu, and J.S. Greenstein, "Using System-Generated Mnemonics to Improve the Usability and Security of Password Authentication," Proc. Human Factor and Ergonomics Soc. Ann. Meeting, vol. 56, no. 1, pp. 506-510, 2012.

[8] P.G. Kelley, S. Consolvo, L.F. Cranor, J. Jung, N. Sadeh, and D.Wetherall, "A Conundrum of Permissions: Installing Applications on an Android Smartphone," Proc. Workshop Usable Security (USEC '12), Feb. 2012.

[9] H. Peng,C.S. Gates, B.P. Sarma, N. Li, Y. Qi, R. Potharaju, C. Nita-Rotaru, and I. Molloy,"Using Probabilistic Generative Models for Ranking Risks of Android Apps," Proc. ACM Conf. Computer and Comm. Security, pp.241-252, 2012.

[10] J. Staddon, D. Huffaker, L. Brown, and A. Sedley, "Are Privacy Concerns a Turn-Off? Engagement and Privacy in Social Networks," Proc. Eighth Symp. Usable Privacy and Security (SOUPS '12), pp. 1-13, 2012.

[11] P.G. Kelley, L.F.Cranor, and N. Sadeh,"Privacy as Part of the App Decision-Making process,"Proc. Conf. Human Factors in Computing Systems (CHI '13), pp. 3393-3402, 2013.

[12] Kevin Benton, L Jean Camp, and Vaibhav Garg. Studying the Effectiveness of Android Application Permissions Requests. In Pervasive Computing and Communications Workshops (PERCOM Workshops), 2013 IEEE International Conference on, pages 291-296. IEEE, 2013.

[13] Alexios Mylonas, Anastasia Kastania, and Dimitris Gritzalis. Delegate the Smartphone User? Security Awareness in Smartphone Platforms. Computers & Security, 34:47-66, 2013.

[14] Veelasha Moonsamy, Jia Rong, and Shaowu Liu. Mining Permission Patterns for Contrasting Clean and Malicious Android Applications. Future Generation Computer Systems, 36:122-132, 2014.