# Permission Based Risk Communication For Android Apps

**Viral Mandaliya, Bhavesh Jain, Sanket Palekar, Kunal Patel**

*Students, Dept. of Information Technology Engineering, MET'S BKC IOE Nasik, Maharashtra, India*

---***---

**Abstract** - *The popularity and advancement in the functionality of mobile devices has made them targets for malicious and intrusive applications (apps). Although there are strong security measures provided for most mobile systems, the area where these systems fall short are their reliance on the users to make decisions that affect the security of the devices. As our prime example, Android depends on users to understand the permissions that an app is requesting and base the installation decision on the list of permissions. Research has shown that this reliance on users is ineffective, as most users do not understand or consider the permissions important. We propose a solution that leverages a method to assign a risk score to each app and display a summary of that information to users.*

***Key Words***: **Malicious, Intrusive, applications, security, permissions.**

## 1. INTRODUCTION

The present research focuses on the Android platform, because of its openness, its popularity, and the way in which Android handles access to sensitive resources.

An android app requests for a specific permission to access a particular resource. Before installing android app user is warned by android about the permissions required by a particular app, the user is expected to make a proper decision. The strength of defense mechanism largely depends on users' preferences. The declaration of an app whether it is secure or not, depends on users' choices. Therefore, it is important characteristic of security to notify the user, the risk involved in installing an app and to influence user to make a good decision to install a particular app. The summary risk rating for each app is added in the proposed system.

A summary risk rating provides easy risk comparisons for apps that have same functions. The reason behind the ignorance of critical information like permission system from user is that the information is provided in standalone way which requires a lot of technical knowledge. There is always similar app available and users always have alternate choice while choosing app of similar type in mobile ecosystem. If user finds app riskier then user can always choose alternative app of similar kind having less. Then the developers will ask for only the necessary permissions required by the app and developers will follow least privilege principle. One possible method for generating a principled metric to rank an apps risk based on the set of permissions it requests. Any android app on Google Play can be ranked by this method. The risk rank can be categorized like low, very low, medium and high, to provide a summary risk rating.

Our hypothesis is that when a summary risk rating is presented in a user friendly fashion, it will encourage users to choose apps with lower risk. In this work, we tested the hypothesis experimentally. Additionally, we explored how to communicate this risk information to an average user effectively and efficiently.

Android's recommendation system is a part where the recommendation for the user are generated. However, these are generated depending on the apps being used by the neighbors. We look into different techniques for computing item to item.

Similarities and different techniques for obtaining recommendation from them. Recommender system apply knowledge based discovery technique for making recommendation. It utilizes entire user item database to generate a prediction. It uses statistical technique to 2nd set of user (neighbor). Once neighborhood is formed these systems use different algorithm to combine preference in neighbor to produce a prediction for active user. This technique is more popular and widely used in practice.

## 2. RELATED WORK

### 2.1 Security

Information security and privacy are issues for users of all types of electronic devices. With regard to smart phones, privacy of mobile phone is the first priority of users than that of computers, and they especially worry about the threat of malicious apps. However, although people are shown the permissions an app requests before it is installed, they do not understand them well. The addition of new security indicators not only may decrease the frequency of risky user behaviors, but it may also facilitate the use of smart phones for online transactions by more individuals. People will not use security features properly if they fail to understand the purpose of the features or the information on which their decisions should be based. The security features also will not be used if the users find the features intrusive or too difficult to master. Therefore, interactions between users and the systems need to be simple and user friendly. Despite this need, studies of various security and privacy measures have shown their usability is typically deficient, which often leads to user resistance. Studies have also demonstrated that usability can be improved by systematically studying the human information-processing requirements associated with effective use of the measures and incorporating the resulting knowledge into the designs.

## 2.2 Usability

Usability of security mechanisms has been studied in contexts other than mobile platforms. When the privacy indicators were presented alongside the search results, participants who chose to visit only a single website paid more money for a higher level of privacy. However, when this information was provided after a website had been selected, participants did not alter their initial decision to purchase from a cheaper website with lower level of privacy.

## 3. PERMISSIONS

| *Permission Group* | *Permission* |
|---|---|
| CALENDAR | READ_CALENDAR<br>WRITE_CALENDAR |
| *CAMERA* | *CAMERA* |
| *CONTACTS* | READ_CONTACTS<br>WRITE_CONTACTS<br>GET_ACCOUNTS |
| *LOCATION* | ACCESS_FINE_LOCATION<br>ACCESS_COARSE_LOCATION |
| *MICRO-PHONE* | RECORD_AUDIO |
| *PHONE* | READ_PHONE_STATE<br>CALL_PHONE<br>READ_CALL_LOG<br>WRITE_CALL_LOG<br>ADD_VOICEMAIL<br>USE_SIP<br>PROCESS_OUTGOING_CALLS |
| *SMS* | SEND_SMS<br>RECEIVE_SMS<br>READ_SMS<br>RECEIVE_WAP_PUSH<br>RECEIVE_MMS |
| *STORAGE* | READ_EXTERNAL_STORAGE<br>WRITE_EXTERNAL_STORAGE |

## 4. MODULES

## 4.1 Risk Communication

Probabilistic generative risk scoring models are being introduced and proposed. Probabilistic generative models have been used rigorously in many applications in machine learning and computational theises, to model complex data structure. The main strength is to model features in a large number of unmarked data.

With the help of these models, we assume that some randomized parameteric process generates the metadata of app and learns the parameters of model based on the metadata of app. Then each apps probability is computed by the model. The risk score function probability is inversely related, so that lower probability translates into a higher score. We define categories of permissions namely safe, normal, average and critical (dangerous).

The ultimate aim is to make an app ask less critical permission so the risk of app is less and stop the app from asking more critical permission. Permission categories defined in the list is used for identification of permissions. An app with high risk score can reduce the risk score by not asking rare permissions and those which are not needed critically which carries out harmful activities. Permissions are categorized on the frequency of its use.

E.g. Full network access, Location access, camera, device information permissions etc. can be defined as critical permissions. Then the automated function calculates the number of critical, average, normal and safe permissions. It then generates the risk score by calculating which permissions belongs to which category. The risk score is presented in percent of risk and even in graphical way.

## 4.2 Recommendation

Here we analyze different item-based recommendation generation algorithm. We considered different approaches for similarities between item to item for computation and different techniques for obtaining recommendation from them. Recommender system apply knowledge based discovery technique for making recommendation. Two technique can be proposed

### 4.2.1 Memory based (User-based).

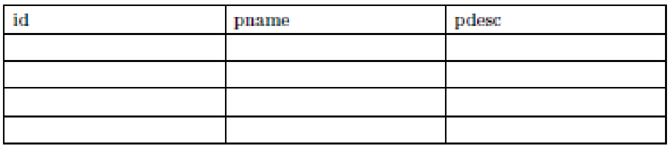
It utilizes entire user item database to generate a prediction. It uses statistical technique to find set of user (neighbor). Once neighborhood is formed these systems use different algorithm to combine preference in neighbor to produce a prediction for active user

### 4.2.2 Model based (Item-based)

It provides item recommendation by developing model of user rating. Algorithm takes probability approach and envision the collaborative filtering process as computing the expected value of user prediction. It can be done by machine learning algorithm such as Bayesian network, Clustering and rule based approach.

## 5. FIGURES AND TABLES

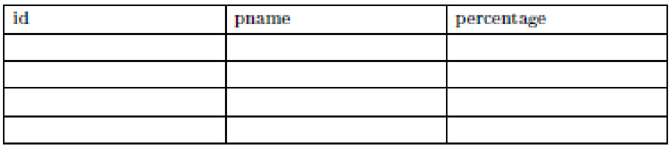
1) In this table, permissions from app gets inserted in the table, it has given particular id no and its description. Name of the apps and its descriptions will be taken in this table.



| id | pname | pdesc |
|---|---|---|
| | | |
| | | |
| | | |
| | | |

Fig. 1. Table apps

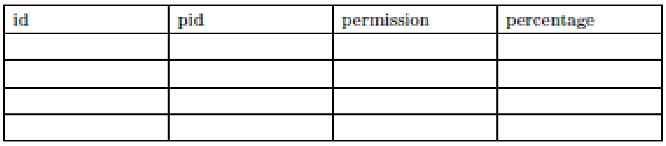
2) In the basis permission table, every permission has given some criterion and risk percentage is given. Here each and every permission created by Google will be mentioned and assessed on the basis of their misuse to user's privacy.



| id | pname | percentage |
|---|---|---|
| | | |
| | | |
| | | |
| | | |

Fig. 2. Table basis permission

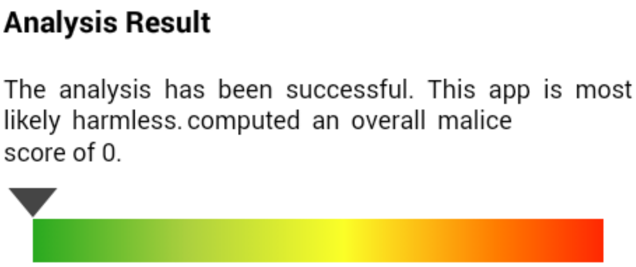
3) In permission table, pid is foreign key of apps table and percentage of basis permission table will be given. This table will be shown to user. Every permission asked by app is mentioned and from permission table, the risk percentage is given along with name of the permission.



| id | pid | permission | percentage |
|---|---|---|---|
| | | | |
| | | | |
| | | | |
| | | | |

Fig. 3. Table permissions

4) Graphical representation of the risk score.



Analysis Result

The analysis has been successful. This app is most likely harmless. computed an overall malice score of 0.

Fig. 4. Graphical representation of the risk score.

## 6. CONCLUSION

When generated risk is presented in an easy way E.g. the app is represented category wise and presented in the selection process, this will result in choosing of apps with lower risk by the user. The majority of people wanted to have such a risk metric in Google Play Store. We expect that adding a summary risk metric would affect in positive changes in the app ecosystem. It will lead the developers to develop the apps with less privileges as the user will demand an app with lower risks. The introduction of risk score opens up the possibility of more users will pay for low risk apps preferably. Thus, this limits the developer for developing a particular app with only the required permissions. This will help in skipping out the invasive needless permissions. Our studies are not the last word on the question of how to best present risk information. For example, we have also not examined how the risk score interacts with other factors to affect a user's choice, such as user ratings in the natural setting and whether an app is free or not. Also of interest is how users behave when choosing among a list of search results (as opposed to choosing between two options). These topics are important ones for future research.

## ACKNOWLEDGEMENT

We sincerely express our deep sense of gratitude towards our respected Prof. Namita Kale. For her valuable guidance, profound advice, persistent encouragement and help during the completion of this work. Her time to time helpful suggestion boosted us to complete this task successfully. She has helped me in all possible ways right from gathering the materials to report preparation.
We express our thanks to our Project Coordinator Prof. Kalpana Metre for her kind cooperation. We extend our sincere thanks to Our Head of Department Prof. Namita Kale for providing all kinds of cooperation during the course.
Finally we are thankful to the supporting staff of Information Technology department and all those who directly or indirectly contributed to complete this work.

## REFERENCES

[1] H. Lockheimer, \Android and security", GoogleMobile Blog, http://googlemobile.blogspot.com/2012/02/android-and-security.html.

[2] A. P. Felt, M. Finifter, E. Chin, S. Hanna, and D. Wagner, \A survey of mobile malware in the wild," in Proc. 1st ACM SPSMWorkshop, 2011, pp. 314.

[3] Y. Zhou, Z. Wang, W. Zhou, and X. Jiang, \Hey, you, get out of my market: De-tecting malicious Apps in social and alternative Android markets", in Proc. of 19th NDSS, 2012.

[4] H. Peng, C. Gates, B. Sarma, N. Li, Y. Qi, R. Potharaju, C. Nita-Rotaru, and I. Molloy, \Using probabilistic generative models for ranking risks of Android Apps", in Proc. of 19th ACM CCS, 2012, pp. 241252.

[5] S. Chakradeo, B. Reaves, P. Traynor, and W. Enck, \MAST: Triage for market-scale mobile malware analysis", in Proc. 6th of ACM WiSec, 2013, pp.1324.

[6] Christopher S. Gates, Jing Chen, Ninghui Li, Robert W. Proctor, \"Effective Risk Communication for Android Apps", IEEE Transactions on dependable and secure computing, VOL. 11, NO. 3, MAY-JUNE 2014.