IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

PREVENTING FROM ABUSING APPLICATION PERMISSIONS

[1]Govind Ganesh, [2]Dr. Manusankar C, [3]Dr. Prathibha P. H.

[1]PG Scholar, [2]HoD, [3]Assistant Professor

[1]PG Department of Computer Science,

[1]Sree Sankara Vidyapeetom College Valayanchirangara, Kerala, Ernakulam, India

Abstract: The rise of mobile apps has a significant impact on the security of the user's device and data. Even though the number of apps has increased, many of them are still not fully secure. Due to the limitation of developers' knowledge and the lack of strict development specifications, the quality of the apps cannot be guaranteed. This may lead to potential security problems, especially for the over requirements of the apps' permissions, which is called Permission Abuse Problem.Modern smartphone platforms have millions of apps, many of which request permissions to access private data and resources, like user accounts or location. 80% of the apps are requesting permissions more than what they need and actually used. Continuously, such over-privileged apps would be exposed to serious malicious behaviours. This paper discusses possible solutions to overcome this issue and suggests possible ways to select the required permissions throughout the app development process.

Index Terms - Android, Abused Permission, Privacy, Mobile Apps**.**

I. INTRODUCTION

Mobile services are playing an increasingly prominent role in our daily lives, e.g., communication, information, entertainment, business, education, etc., based on the feature rich applications. A mobile app is a software application designed to run on mobile devices such as Smartphone and tablet computers. All of these apps are created by organizations or individual developers, scanned by Google Bouncer, distributed to Google Play Store and downloaded by the end users. Due to the developers' abilities and lack of strict development specifications, the quality of the apps cannot be guaranteed, which may lead to potential security problems. As we know permission is the main security mechanism in Android operating system, which is used to enforce access control to the system APIs and applications. All applications have their own statements for permissions in their configuration file "AndroidManifest.xml". When an application is installed, the Android system notifies the user about the permissions the application is requesting and asks the user whether to install the application continually.

This is a relatively flexible security policy, which allows the users to know their personal privacy information, such as SMS, contacts, photos, location information, camera, microphone, phone states, etc., will be accessed by the installed app. Whenever an app requests more permissions than it actively needs, there is an application Permission Abuse Problem which may introduce security threat.

II. PRIVACY PROTECTION FOR END-USERS

According to Android developers all the dangerous permissions are placed into a special group called permission group which may also contain normal permissions. The user is only notified when the app requests dangerous permissions. This notification only informs the user about the type of the permission being requested and not how the app is going to use this kind of permission. In addition to permissions described by android developers as dangerous there are also some other permissions which pose a significant threat to user privacy. Examples are permission to access WIFI connection. Using this permission an app can send sensitive data to the app developer. Another example is to access accounts or to use authentication information of an email service or a social media account which puts the personal information of the user under risk. Most of the free apps depend on the ads to generate revenue and keep their business running. This situation is not encouraging as the app developers' end up requesting more permissions to gather user data.But the fact is that, all of these permissions are not required to run that specific app.

III. An Analysis of Android App Permissions

In the Android operating system, this point of contact is a trilateral relationship between the user, Google (the designer and provider of the Android operating system) and third-party app developers. Google moderates the relationship between the user and the third-party app developer using a set of "permissions" for each app a user downloads. Permissions are Google way of requiring developers to disclose how the app will be interacting with the user's device and what information the app will have access to.

IV. Google App Permissions Basic

This section of the report looks at the range of app permissions in the Google Play Store, focusing on permissions that could allow apps to collect or share users' personal information. In this dataset, there are 2.5 million apps with a total of 235 unique permissions. There are a few apps that require a lot of permissions from users, with the highest number being 127. It's not very common for apps to require this many permissions, but it does happen from time to time. Most apps only require a few permissions. The average app requests five permissions.

V. Permission Abuse Problems

Due to the nature of the apps industry, it is possible that the permissions granted to the developers can be exploited by unauthorised individuals. For example, Sound comber is a sophisticated malware with limited permissions that is able to leak sensitive high-value data based on its audible surroundings. The key idea of Soundcomber is that an app can borrow the other's permissions so that it does not have to by- pass the permission mechanism and leak private information. In this way, the app can avoid the detection of anti-virus software and leak the information no matter that it lacks the proper permissions. On the other hand, permission abuse can also unintentionally isolate the principle of least privilege, which seriously affects the code and quality specifications. By checking out the most commonly requested permissions and how they're abused, hackers can exploit them and gain personal privacy.

VI. MOSTLY ABUSED ANDROID PERMISSIONS FOUND IN THE EDUCATION APPS.

Permission Name	What it is Used For	How it can be Exploited
Network Based Location	Approximates Location of user	Location based attacks or malware
GPS Location	Gives location through GPS	Location based attacks or malware
View Wi-fi States	Give access to Wi-fi network information	Steal Wi-fi password
Retrieve Running Apps	Finding which app or process are running	Kill running application or get information about running apps
Full Internet Access	Allow internet connection	Internet access could be used for exploitation or malware
Read Phone State and identity	Allow access to information about calls, network and other identity information	Steal information from data

TABLE VI of [9]

VII. PERMISSION TRACKER

Using permission tracker we can identify whether the permission we are giving (accessing camera, audio, contacts etc.) is being misused. The Permission Tracker also allows a detailed view with information about specific permissions, Permission Tracker generates several reports to ease an analysis of applications and permissions. Reports include permissions of applications, the permissions' status at the time of access, as well as date and time of access.

User can track what happens to user allowed permissions in the application using the permission tracker tool. This paper focus on the permissions usage and its impact on the security of apps and users' data. This study is the starting point to test apps under several categories, observe the permissions requested by each category and their usage, analyze and categorize permissions to define the most abused ones by security attackers. Also, draw conclusions and put recommendations to develop more secure apps and build trusted, benign apps databases. The purpose of this paper is to help users become more security-aware, and to put recommendations in place for app developers who are developing mobile apps.

VIII. LITERATURE SURVEY

The paper titled "Analysis of Permission-based Security in Android through Policy Expert, Developer, and End User Perspectives" by Ajay Kumar Jha and Woo Jin Lee.In this paper, they analyse the permission-based security in Android through three different perspectives: policy expert, developer, and end user. We have mostly analysed and discussed the major issues or weaknesses which came across several studies of permission-based security.

The paper "Permission Tracking in Android" by Michael Kern and Johannes Sametinger In this paper they have introduced Android with its security mechanisms. The permission system is rather rigid in Android and suffers from a few drawbacks. They have developed a more flexible permission system with a permission tracker tool. The system allows users to block and monitor access to re- sources by arbitrary applications. For implementation purposes, the Android system had to be slightly extended.

The paper "Abusing Android Permissions: A Security Perspective" by Mamdouh Alenezi and Iman Almomani.In this paper uncovers the critical status of permissions usage of the available apps offered by different app stores. The proposed system in this paper has been used to deeply analyse the most rated apps in the education category. The analysis found that many of the apps are asking for permissions that are not necessary, and this could expose the app and its users to security risks.. The main aim of this study is to enhance the security awareness of the users and put recommendations to help mobile apps developers to take proper decisions regarding the required apps permissions during the development stages. Also, this study is considered as a starting point to build a truly benign datasets of mobile apps that could be used for research purposes. The researchers will be able to reproduce the study and explore more possibilities in studying apps permissions.

The paper "PACS: Permission Abuse Checking System for Android Applications based on Review Mining" by ying zheng Wu, Mutian Yang and Tianyue Luo.This paper mainly focuses on the Application Permission Abuse Problem based on review and description mining. PACS (Permission Abuse Checking System) based on data and frequent itemsets mining technique to reviews bring improvement using apps' and by the and descriptions. PACS sorts the apps by analysing the app's metadata, such as reviews and descriptions. Then, it obtains the maximum frequent item sets and constructs the permission feature database. Finally, evaluate PACS on detecting unknown applications of the abused permission. The experiment results show that 726 out of 935 applications, which account for about 77.6%, are suffering from the Permission Abuse Problem.By comparison with the previous tools, PACS has better performance.

The paper "On the Effectiveness of Application Permissions for Android Ransomware Detection" by SamahAlsoghyer and Iman AlmomaniThis paper provided a proactive mechanism to detect the ransomware before it harms the user's device.various analysis mechanisms including static, dynamic or both. In this paper, recent ransomware detection solutions were investigated and compared. Furthermore, a deep analysis of android permissions was conducted to identify significant android permissions that can discriminate against ransomware with high accuracy before harming users' devices. Consequently, based on the outcome of this analysis, a permissions-based ransomware detection system is proposed. Different classifiers were tested to build the prediction model of this detection system.

IX. CONCLUSION

Nowadays, more than 2.6 million apps are created by various organisations and individual developers. Without strict development specifications, the quality of the apps cannot be guaranteed, especially for the apps' permission requirements, which may induce Permission Abuse Problems. This problem is increasing at a higher growth rate, and may bring serious security risks. The use of an app like the Permission Tracker application is recommended for users who want to control access to their resources on a finer granularity than what is currently possible with Android. Android users who want to have advanced control over permissions granted to their apps have three options.

REFERENCES

[1] Au, K. W. Y., Zhou, Y. F., Huang, Z., Gill, P., & Lie, D. , Short paper: "a look at smart phone permission models", In Proceedings of the 1st ACM workshop on Security and privacy in smart phones and mobile devices, 2011.

[2] Iman Almomani., and Mamdouh Alenezi., "Android Application Security Scanning Process",June 11th 2019 Available: https://www.intechopen.com/books/telecommunication-systems-principles-and-applications-of-wireless-optical-technologies/android-application-security-scanning-process.,

[3] Bhaskar Sarma., Ninghui Li., Chris Gates., Rahul Potharaju., Cristina NitaRotaru., "Android Permissions: A Perspective Combining Risks and Benefits", Proceedings of the 17th ACM symposium on Access Control Models and Technologies, 2012.

[4] Veelasha Moonsamy., Jia Rong., Shaowu Liu., "Mining permission patterns for contrasting clean and malicious android applications". Future Generation Computer Systems, 2014.

[5] Ajay Kumar Jha., and Woo Jin Lee., "Analysis of Permission-based Security in Android through Policy Expert, Developer, and End User Perspectives", April 2016.

[6] A. Khatoon., and P. Corcoran., "Privacy concerns on Android devices", IEEE International Conference in Consumer Electronics (ICCE) , pp. 149-152, 2017.

[7] Gilbert, P., Chun, B. G., Cox, L. P., & Jung, J., "Vision: automated security validation of mobile apps at app markets", In Proceedings of the second international workshop on Mobile cloud computing and services, pp. 21-26, 2011,

[8] Zhang, Yuan, et al., "Rethinking Permission Enforcement Mechanism on Mobile Systems." IEEE Transactions on Information Forensics and Security11.10. 2016

[9] https://malenezi.github.io/malenezi/pdfs/Abusing%20Android%20Permissions.pdf