International Journal of Engineering & Technology, 7 (2.6) (2018) 301-305

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

An Investigation on android background services for controlling the unauthorized accesses using android LOG system

Jay Kotecha[1*], Prabu P.[2]

[1]Dept. of Computer Science, Christ University, Bangalore.
[2]Asst. Professor Dept. of Computer Science, Christ University, Bangalore.
**Email: kotecha.jagdishbhai@mca.christuniversity.in*

Abstract

In the current world of technology, people are updating to smart devices day by day. Users often keep personal information on their smartphones, but it increases the threat to the privacy of their data. There are many applications which require the user data to provide specific services such as WhatsApp, Instagram, Paytm, PhonePe, etc., There are special permissions which can be misused by some applications, like *READ_EXTERNAL_STORAGE, READ_SMS, SEND_SMS, WRITE_EXTERNAL_STORAGE, READ_CONTACTS,* etc., Many applications take users data to their servers without the knowledge of user by running in the background. In this paper, it is shown that how an application runs in the background by running various services, and performs the background activities like notifications, displaying ads, etc., A proposed algorithm is described that how every activity of the background services can be monitored using Android Log and user can be alerted by showing which data is being accessed by the particular application.

Keywords: *Android Security, User Data Privacy, Background Activity, User Interface Attacks.*

1. Introduction

In 2008, Android mobile operating system reached a huge achievement. The number of Android users have been increased largely since few years by then. The usage of smartphones has increased and has become a necessity for the users. The quality of mobile applications has been grown immensely due to the expectations of the users. Since, most of the expectations are met the need for privacy and security have become a top priority.

As the smartphone users have increased, the amount of sensitive personal information stored on the mobile phones is comparatively more when compared to laptops and desktops or any other computing devices. There are many malicious applications which track down the private data such as contacts, messages, device location, etc. of the user stored in the device, without the knowledge of the user. Therefore, this highly raises the question regarding the privacy of user data.

Many user interface attacks to the device have also occurred several times. These attacks are executed by some malicious applications or some background services which takes advantage of the user permissions which is granted to the particular applications and uses the data of the user irrelevantly. Sometimes the application which user is installing on their device, ask for more unnecessary permissions to leads this kind of attacks on the privacy of the user data.

If any user is playing a game on an Android smartphone, he/she might not be aware of the application's background services. It might be possible that the service which is running in the background takes the user's external SD card data to their server by uploading it if the device is connected, or when it connects to the Internet.

There is also another possibility that any application on the user's smartphone is malicious. And when any user is making an online payment or feeding any sensitive information on the device, it pops up between, and the user can get confused by the original screen and the dummy screen which it appeared to use the sensitive information. At this point, the user is entirely unaware of the background process running on its smartphone. It is hard to detect the background running service, which can cause this privacy breach to the user device.

And this doesn't get over here, using dynamic applications or web applications creates the same risk of privacy. Apps with WebView has the same functionality as a Web Browser.The only difference is that the user is not able to insert the URL. Instead, it is precoded that the browser requests to the server with HTTP protocol and provides the data to the user which is on the web. In this process, an API is used to perform some features on the internet, which is called addJavascriptInterface. It allows the JavaScript code to invoke the app's Java code, which allows the webpage to control the content of the device such as files, user location, contacts, messages, etc. With this technique, also there are possibilities for breach of user's data privacy.

A malware named Ransomware was very much popular these days. It was initially targeted on laptop and desktop users, but by the second half of 2016, the number of targeted users on Android

smartphones was very huge. It encrypts all the data in the user device and asks for a ransom to decrypt the data back to the user. There were different sources found where this malware attacks the phone. Mostly it was from the unverified apps downloaded by the user, but many others like fake antivirus, battery saver apps, phone boost apps, etc. There were cases where this malware disables the phone, and it does not unlock the phone until the user pays the amount. As the security tips, the users were asked to check the apps reviews before installing and to take the backup of their data.

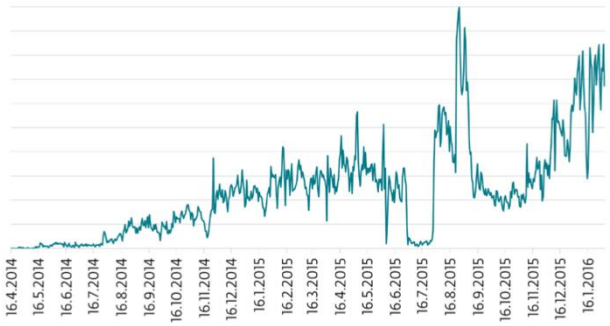

Fig. 1: Android Ransomware Detection [9]

The above diagram shows the increase in the number of Ransomware detected in the Android devices around the globe during the period of mid-2014, till 2016.

The goal of this research is to improve the security of the Android smartphones regarding privacy of the user data. Every application which is using the resources of the device should be strongly detected and should alert the user. In this section, an introduction is given about the privacy is being breached by many sources. In the next section, the different views and methods regarding user data privacy are discussed. After that, the proposed work is explained in detail with a solution to reduce this problem to the minimal level. And in the last section, the conclusion is given which will summarize the whole research.

2. Literature Review

[1] Tongobo Luo, HaoHao, Wenliang Du and Heng Yin described attacks on and from WebView in their paper, "Attacks on WebView in the Android System." They represented different possible attacks which are being occurred through WebView to the users' smartphone. The malicious JavaScript code can invoke the applications' Java code and can use all the resources of the device such as contacts, files, databases, messages, etc. An API called *addJavascriptInterface* allows to remotely executes the code and can control over as the user has permitted specific permissions to the app. Also, the authors have described the different threat models like attacks from malicious apps, *frame confusion*, *JavaScript Injection*, *Event Sniffing and Hijacking*, etc.

[2] JaapVermeij, in this paper, "Alerting Users on Android: The Effect of an Alert During a GUI Confusion Attack", describes the GUI confusing attacks which are taking place alternatively to many Android users. These attacks, according to the author are not much famous but are very much effective and dangerous regarding user data privacy. The author has described mainly about *Phishing*, with many specific reasons that how it can be harmful to the private information of the user. Also, the author has shown an experiment on *Bol.com* application, which is mainly on how the GUI attack is being measured and has alerted the user with the notification of the possible attack on the device.

[3] Bhavani A B, in this paper, "Cross-site Scripting Attacks on Android WebView" presented different attacks based on the *HttpClient*, users' sessions and the WebView API. This cross-site scripting attack is performed on the WebView, which contains the user's previous sessions and cookies. It can be dangerous as the browser stores the credential information of the user. Also, Session Hijacking is also focused in this paper, as it can access and stole private data like contacts and messages. The demonstration of the attack with the help of HTTP and PHP method *$_POST*. When the data from the device is accessed, it is sent to the attacker's servers without the knowledge of the user.

[4] In this paper,"Labyrinth: Visually Configurable Data-Leakage Detection in Mobile Applications," Marco Pistoia, Omer Tripp, PaolinaCentonze and Joseph W. Ligman has presented a self-data-leakage system, to detect and reduce the data leakage of the users' private data on the device. It identifies every module of the application and also alerts the user about the usage of the module to the resources of the device, such as reading contacts, location, etc. Also, the result and success rate of this system has been presented in detail with the maximum possibilities of detection of data leakage.

[5] WenruiDiao, Xiangyu Liu, Zhou Li and Kehuan Zhang, in this paper, "No Pardon for the Interruption: New Interface Attacks on Android Through Interrupt Timing Analysis" described the interrupt statistical information */proc/interrupts* in detail. It can leak the data and also the attackers are exploited. Furthermore, an analysis has provided with the real-time data which can deal with the lock patterns of the user. It can even breach into the other data of the phone too; hence it has been taken care of. There are multiple experiments which are described in the details. The attack prototype has also been taken and evaluated with the real-time data.

[6] Arun Sharma and Harmeet Malhotra in their paper, "Vulnerable Android: A Study on UI Inference Attacks and Malware Attacks" discussed different attacks on the Android Smartphone. The authors have described a lot of attackers' programs to use it to breach the data privacy of the user. Methods and malware like *Activity Hijacking* and *Rootkits*. There are many vulnerabilities described in this paper. Many state UI reference attacks do not require any special permission to breach the privacy is focused in this paper. Strategies discussed were improvising trusted applications, vendor-specific protection, and inbuilt detection tool.

[7] Marcus Niemietz and JorgSchwenk presented about different types of clickjacking attack in their paper, "UI Redressing Attacks on Android Devices." Different *clickjacking* like *cursorjacking*, *likejacking*, *eventjacking*, *tabmabbing*, *double clickjacking*, etc. are the attacks which are explained in is paper. Here, the authors create a notifying system which alerts the user during these above mentioned any attacks occurred in the Android smartphone. Even by basic operations like drag and drop, content extraction, etc. possibilities of attacks are there. And though the authors have created an identifying system, which is been described in this paper.

[8] EarlenceFernandes, Justin Paupore, Georg Essl, J. Alex Halderman, Qi Alfred Chen, Z. Morley Mao and Atul Prakash have represented about the User Interface Deception Attacks in their paper, "Android UI Deception Revisited: Attacks and Defenses." Like in the previous papers, this paper also has the attacks discussed in it such as *Phishing*, *Activity Hijacking*, *Clickjacking*, etc. Here, the defences are more focused than the attacks. The authors have shown the possibilities of the result against these attacks.

[9] Robert Lipovsky, Lukas Stefanko, and Gabriel Branisa well-presented about the Ransomware in their report on "The Rise of Android Ransomware." In Android smartphones, ransomware is categorized into two, Lock-screen Ransomware and Crypto-ransomware. The whole report shows about the increasing of the android device attacks by the malware. Globally 72% is the United States who has been targeted by the Ransomware. Different results and outcomes are given in this paper about how to handle the ransomware if infected any of the phones with it. Also, the

report of the FBI says that this malware is too dangerous that it can make user data inaccessible.

[10] William Enck, Damien Octeau, Patrick McDaniel, and Swarat Chaudhuri in their research work, "A Study of Android Application Security" have presented a decompiler for the applications which identifies dangerous functionality and also the vulnerabilities. A framework also has created to observe the actions of the applications. The primary intention of this paper is to analyse the application certification given by the play store can be revised and make it strict to the applications which were getting the clearance certification very quickly. Their work is based on application installation, malicious attack detection, and phishing.

[11] Saranya T., Shalini A.P., and Kanchana A. presenting a counter system for detection and prevention of the malicious attacks in the Android smartphones, in their research work, "Detection and Prevention for Malicious Attacks for Anonymous Apps." In this paper, more of the focus is on the third-party applications which are anonymously using the user's private data along with the malicious attacks by the external sources. The counter system which they have presented in this paper works on the user permissions allowed by the user. This method prevents the access of malicious source to the mobile's resources. It also detects the number of permissions unnecessarily provided to the applications which are not related to the application's functionality.

[12] Poornima Mahesh, Ashwini Jayawant, and Geetanjali Kale have presented their research, “Smartphone Security: Review of Attacks, Detection, and Prevention" on the detection and prevention of the attacks on Android smartphones. This paper major focuses on security issues such as permissions, data leak, etc. It shows the different technique to detect and prevent the malware and make the security of Android at a minimum risk.

[13] Bahman Rashidi and Carol Fung have presented a survey on the Android threats and its defenses since 2010 till the current time. They have stated that 70% of the applications are taking the user data inadequately and using it in a wrong way. The behavior of the apps are noted, and they concluded that there are specific points where security can be easily breached, and the data can be harmfully used. Even there are specific points where they have stated that there are no modifications needed for the security. As they have proposed work and not executed any practical work, they are defining the whole survey based on the other research works which have been referred.

[14] Sanggeun Song, Bongjoon Kim, and Sangjun Lee have presented a research work on the Ransomware prevention and to monitor the activities so that it can be taken care that device's resources are not used or affected in the wrong way. In this paper, "The Effective Ransomware Prevention Technique Using Process Monitoring on Android Platform" a defensive method is proposed which monitors every activity such as input-output events, processor status, etc. It is shown that this technique can reduce the risk of ransomware at such extinct that the data can be prevented from this malware. It also ensures that the damage due to this attack can be minimized at a significant level.

[15] Qi Alfred Chen, Zhiyun Qian, and Z. Morley Mao presented this research work, "Peeking into Your App without Actually Seeing It: UI State Inference and Novel Android Attacks." In this paper, the focus is on UI state inference and Novel attacks. The implementation of this paper is done by design and execution of the attack which is built by the authors for getting the output that what maximum these attacks can affect the device. These attacks have shown a robust building block which makes these attacks more dangerous. Also, a security concept is discussed to reduce these attacks.

3. Research Problem

3.1. User Data Privacy

In the Android OS, since the beginning, as it was open source, there was no such security factor in the initial years of the release. After some years, as people were becoming more familiar with the Android, it became necessary to make the data secure on the phone. People started sharing more private and confidential information on their phones, and so it was the attraction to the attackers, which cause a panic in the technology world [13]. There were many attacks on the Android mobile OS which causes many data privacy breach, misuse of the confidential data, etc. As the updates were released to upgrade the OS, the risk factor of the data breach was reduced with each release and a new feature regarding security. But still, it wasn't sufficient as there were new methods to breach the user data privacy [12]. And it is very much essential that these loopholes are to be corrected in the Android OS itself. There are possibilities where the application which is allowed by some specific permissions such as read contacts, read/write storage, can be taking user data to their servers without even user's knowledge [14].

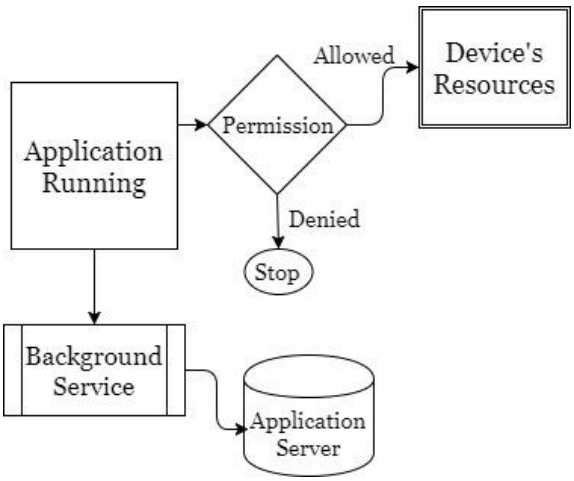

Fig. 1: Application running in Background taking device resources to their servers

After the permissions are allowed to the user, it is not notified to the user about what information from their device is being used or taking it to the application servers. It creates a severe issue which requires a focus, and though this research has considered this risk factor.

3.2. Malicious Attacks

Concerning security, Android OS has come a far way better since its first version was released. Initially, there were many issues regarding the protection of the data, but with the updates and latest versions, it was also taken into the concern with a higher priority[13]. Since it is an open source mobile operating system, it always has the risk factor for malicious attacks. In the solution to this issue, there were a lot of developers came up with antivirus for this operating system. Trusted and verified antivirus became very much helpful to the users. But the users with lack of detailed knowledge about the antiviruses were installing any random antivirus which having a familiar logo on the app[6]. It leads to risk their private data stored in the device. The antivirus requires all the system permissions which can be sensitive to allow. But in an illusion of taking the protection from the virus[9], users are granting all the required permissions to the app. It can leads to data privacy breach as the app is not trusted.

4. Proposed Work

Regarding the data privacy breach, the proposed technique is designed, which can help the user to know about the apps which are taking the device data into their servers without the knowledge of the user. The user is alerted when there is any process of accessing or reading the device data through the applications' background process. Although the user has already allowed the permissions required to install the app, it is necessary to know that what data it is using. And for that reason, this technique is designed. It can be an Android OS inbuilt feature so that there is no need for any external application or system to handle the issues when the priority is set for the privacy of the data.

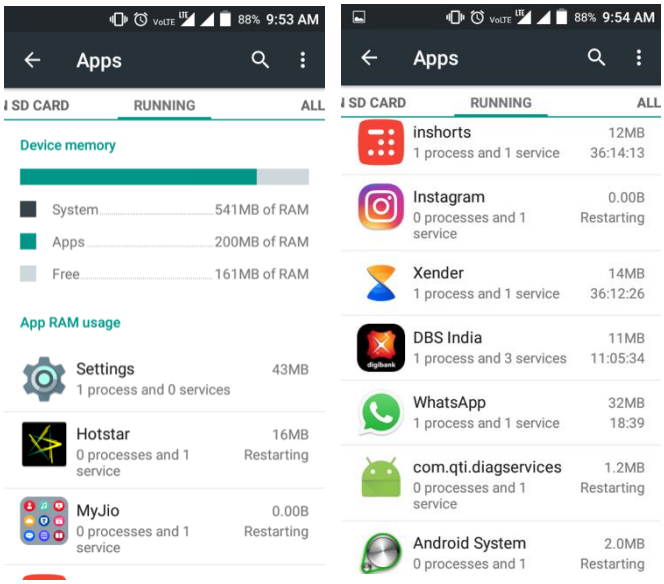

Fig. 1: Applications' background services running in the mobile device

The above figure shows the background services running on the device. It is visible that for a long time these processes are running. These services mentioned in Figure 3, are of trusted applications, which are not taking any data without the knowledge of the user[7]. But like these services, there are possibilities that the services running in the background can access the data of which the permission is allowed. To reduce the risk of data breach, or data leakage, the following technique is designed, which notifies the user every time, when the applications use the sensitive data. It can help the user to know what data is being used and is it required by the applications to use it or not. The user can entirely allow or deny that action. The actions will be detected by the Android Log, which in the system will be monitored. On any unusual activity related to the users' private data, it will pause that action and will notify the user[4]. The users will decide to allow or to deny that action, as it will show a brief alert that what data was being used and by which application. And the choice of the permission to allow or deny by the user will be remembered for that particular application.

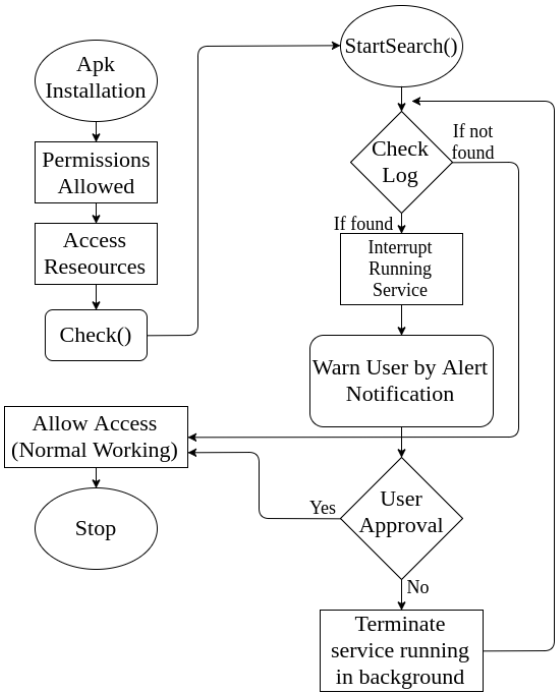

Fig. 2: Background Process Monitoring through Android Log

When the application is installed on the mobile, it requires a specific set of permissions to the user. Permissions like*ACCESS_NETWORK_STATE*, *ACCESS_WIFI_STATE*, *READ_SYNC_STATS*, *REQUEST_INSTALL_PACKAGES*, *USE_FINGERPRINT*, etc. are considered as standard permissions. These permissions are harmless and just uses the functionality of the user device to service the application with different features [12]. But, on the other hand, permissions like *SEND_SMS*, *READ_SMS*, *ACCESS_FINE_LOCATION*, *READ_CONTACTS*, *READ_EXTERNAL_STORAGE*, *WRITE_CONTACTS*, *WRITE_EXTERNAL_STORAGE*, *READ_CALL_LOG*, *WRITE_CALL_LOG*, etc. are considered as the dangerous permissions. These are system permissions which are allowed by the user to some of the applications to execute specific tasks[15].

When the user grants the permissions to the application, it can access all the resources which the permissions are granted. It can access, modify or take a backup of the data to their server. And therefore, it can be risky regarding the data privacy of the user [10].

This proposed technique is designed to reduce the risk of data privacy. In this method, when the user allows the permissions it checks the list of permissions granted to the application. After the installation when the application accesses the resources of the device while it is on, this method checks the Log of Android continuously to detect the irrelevant usage of the data [8]. It monitors the whole process state of the application. And when if the data is accessed irrelevantly, it interrupts the process and notifies the user about the specific process is using the data with the details of data which is being obtained [13].

If the user allows the process, then it will resume, and the monitoring will be continued for the next detection. And if the user denies the process, then it will be terminated and will continue monitoring the log.

This method helps the user to know what data is being used by which application. In most of the cases, the user is not aware of these situations and possibilities. This technique will help the user to know about the possible risks and has a choice to avoid it. This

process should be included in the system process itself as there should be no need for an external app to involve with the system processes. It can be riskier if any external app is granted such sensitive permissions.

5. Conclusion

In this paper, initially, the possible risk factors were taken into consideration, then the current scenario about the Android Security was discussed at a glance. Later, the registered attacks and vulnerabilities were disclosed with the possible data privacy breach. In the last part, the method is proposed to reduce the risk factor regarding the data privacy, and with a benefit of user awareness about the data processing in their devices by the background processes. This method should be added to the Android system functionalities, to reduce the privacy breach of the user data.

References

[1] T. Luo, H. Hao, W. Du, Y. Wang, and H. Yin, "Attacks on WebView in the Android system," in 27th Annual Computer Security Applications Conference, ACSAC 2011, December 5, 2011 - December 9, 2011, Orlando, FL, United states, 2011, pp. 343-352.

[2] JaapVermeij, "Alerting Users on Android: The Effect of an Alert During a GUI Confusion Attack", University of Twente, P.O. Box 217, 7500AE Enschede, The Netherlands.

[3] Bhavani A B, "Cross - site Scripting Attacks on Android WebView", IJCSN International Journal of Computer Science and Network, Vol 2, Issue 2, April 2013, ISSN (Online) : 2277-5420.

[4] Marco Pistoia, Omer Tripp, PaolinaCentonze, and Joseph W. Ligman, "Labyrinth: Visually Configurable Data-leakage Detection in Mobile Applications", 2015 16th IEEE International Conference on Mobile Data Management.

[5] WenruiDiao, Xiangyu Liu, Zhou Li, and Kehuan Zhang, "No Pardon for the Interruption: New Inference Attacks on Android Through Interrupt Timing Analysis", Department of Information Engineering, The Chinese University of Hong Kong, 2016 IEEE Symposium on Security and Privacy.

[6] Arun Sharma, and Harmeet Malhotra, "Vulnerable Android: A Study on UI Inference Attacks and Malware Attacks", International Journal of All Research Education and Scientific Methods (IJARESM), ISSN: 2455-6211, Volume 3, Issue 4, April 2015

[7] Marcus Niemietz, JörgSchwenk, "UI Redressing Attacks on Android Devices", Horst Görtz Institute for IT-Security, Ruhr-University Bochum, Germany.

[8] EarlenceFernandes, Qi Alfred Chen, Justin Paupore, Georg Essl, J. Alex Halderman, Z. Morley Mao, and Atul Prakash, "Android UI Deception Revisited: Attacks and Defenses", University of Michigan, Ann Arbor.

[9] Robert Lipovský, LukášŠtefanko, and Gabriel Braniša, "The Rise of Android Ransomware", Report on Ransomware ESET Antivirus 2016.

[10] William Enck, Damien Octeau, Patrick McDaniel, and Swarat Chaudhuri, "A Study of Android Application Security", Systems and Internet Infrastructure Security Laboratory, Department of Computer Science and Engineering, The Pennsylvania State University.

[11] Saranya .T, Shalini .A.P., and Kanchana .A, "Detection and Prevention for Malicious Attacks for Anonymous Apps", International Journal of Innovative Research in Computer and Communication Engineering, Vol. 2, Issue 3, March 2014.

[12] Poornima Mahesh, Ashwini Jayawant, and Geetanjali Kale, "International Journal of Advanced Research in Computer Science and Software Engineering", ISSN: 2277 128X, Volume 5, Issue 3, March 2015.

[13] Bahman Rashidi, and Carol Fung, "A Survey of Android Security Threats and Defenses", Virginia Commonwealth University, Richmond, Virginia, USA, 2015.

[14] Sanggeun Song, Bongjoon Kim, and Sangjun Lee, "The Effective Ransomware Prevention Technique Using Process Monitoring on Android Platform", School of Computing, Soongsil University, Sangdo-ro, Dongjak-gu, Seoul 06978, Republic of Korea, Accepted 10 March 2016.

[15] Qi Alfred Chen, Zhiyun Qian, and Z. Morley Mao, "Peeking into Your App without Actually Seeing it: UI State Inference and Novel Android Attacks", 23rd USENIX Security Symposium 2014, ISBN 978-1-931971-15-7.