RESEARCH ON APPLICATIONS BEHAVIOR ON ANDROID

A THESIS SUBMITTED TO
THE FACULTY OF ACHITECTURE AND ENGINEERING
OF
EPOKA UNIVERSITY

BY

KLEVIS NDOKA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

JULY, 2016

Approval of the thesis:

RESEARCH ON APPLICATIONS BEHAVIOR ON ANDROID

submitted by Klevis Ndoka in partial fulfillment of the requirements for the degree of **Master of Science in Department of Computer Engineering, Epoka University** by,

Prof. Dr. ……...

_____

Dean, Faculty of Architecture and Engineering

Prof. Dr. ……..

_____

Head of Department, **Computer Engineering, EPOKA University**

Prof. Dr. ……

_____

Supervisor, **……. Dept., EPOKA University**

Prof. Dr. ……

_____

Co-Supervisor (if any), **…….. Dept., ……..University**

Examining Committee Members:

Prof. Dr. ……..

_____

……. Dept., …….. University

Prof. Dr. …….

_____

……. Dept., …….. University

Assoc. Prof. Dr. ,,,,,,,,,,,,,,,,,,,

_____

……. Dept., …….. University

Date: __/___/ 2016

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

First Name, Last Name: Klevis Ndoka

Signature:

ABSTRACT

RESEARCH ON APPLICATIONS BEHAVIOR ON ANDROID

Ndoka, Klevis
M.Sc., Department of Computer Engineering
Supervisor: Dr. Elton Domnori

In this thesis we are doing a research in two categories of applications on Android downloaded from Play Store. We will see for each application the permissions required, network packets sent and received, files written on internal storage and hardware resources used while running.

Our research will have on focus abuses by these applications and make some analysis on how many of them are abusing or having a behavior that is not normal for applications functionalities. After analyzing results, we will make a comparison between applications and categories chosen, give our conclusions and our suggestions to improve system.

Keywords: Android, Applications, Permissions, Security

ABSTRAKT

STUDIM MBI SJELLJEN E APLIKACIONEVE NË ANDROID

Ndoka, Klevis
Master Shkencor, Departamenti i Inxhinierisë Kompjuterike
Udhëheqësi: Dr. Elton Domnori

Në këtë tezë ne do të bëjmë një studim në dy kategori të aplikacioneve në Android të shkarkuara nga Play Store. Ne do të shikojmë për secilin aplikacion të drejtat (permissions) që kërkon, përmbajtjen e paketave të derguara dhe të marra nga aplikacioni, file që janë shkruar në memorjen e brëndshme të pajisjes si dhe resurset hardware që kërkon aplikacioni ndërkohe që është duke punuar.

Studimi jonë do të ketë në fokus abuzimin që mund të bëjnë këto aplikacione dhe të bëjmë një analizë se sa shpesh ndodh abuzimi me të drejtat apo sa shpesh shfaqin një sjellje që nuk është normale për natyrën e aplikacionit. Pasi të nxjerrim rezultatet, ne do të bëjmë një krahasim mes aplikacioneve dhe kategorive të zgjedhura, do të japim konkluzionet tona në lidhje me studimin si dhe të japim sugjerimet tona për përmirësimin e sistemit operativ Android.

Fjalët kyçe: Android, Aplikacione, Te drejtat, Siguria

Dedicated to the memory of my father. Thanks for your love, support and encouragement every day of my life. You couldn't be with me during this thesis but your words and your values are always with me.

TABLE OF CONTENTS

LIST OF ABBREVIATIONS

OS	Operating System
APPS	Applications
RAM	Random Access Memory
CPU	Central Processing Unit
AP	Access Point
IDE	Integrated Development Environment
ROM	A package containing an Android System Image

CHAPTER 1

INTRODUCTION

Android is today the biggest mobile operating system by powering 82.8% of smartphones [idc.com, April 2016, Smartphone OS Market Share 2015 Q2]. The power of Android cames at the applications you can install and tasks you can do on them. With applications may come also problems due to bad things that may happen on your device.

Studying applications behavior is the main focus on this thesis. We are going to see if applications abuse with permissions, communications over network, files read write and hardware resources usage. In the end we will make a comparison between tested applications, give our conclusions and suggestions on how to improve security and privacy of the user.

1.1. Background of the problem

To get a smartphone nowadays is getting easier, because prices are reduced while smartphones are getting more powerful. Smartphones helps us on everyday tasks, or helps on being connected all the time with people and so many other benefits. As everything else it has its disadvantages because in our smartphone we may have informations that no one else should have access without our permission.

Smartphones are very useful and very powerful because we can do most of the tasks we do on web or PC by using third party applications, created for smartphones operating system. In our case we will study Android.

Since third party applications are created by independent developers, some of them do not “play fair” and they get access on informations they should not. This happened before and still happens because this is how Operating Systems are created, to allow applications to have access on system resources and offer to user more functionalities.

Even if we see applications from top developers on official Android store we can see that sometimes they ask too much permissions e.g. Go Launcher, one of the top applications on Play Store with over 100 million installs requires permission to do a lot of unneeded task without user confirmation (read, send and delete SMS, place a phone call, enable disable network connections etc.). All these permissions are not necessary for a normal launcher because in its functionalities does not include sending sms. If we see another launcher from official store e.g. Google Now Launcher developed by Google, it only requires permission to add or delete shortcuts on home screen.

Because of above and some other problems on Android, we will do this research on two groups of application categories from Play Store and compare results between them. We will try to get a full view of problems and strange behaviors found after analyzing applications.

1.2. Thesis structure

In the beginning we will give some background informations that are needed to understand better what are we trying to find and the way we are going to find it. We start with some informations about Android OS, its architecture, security and privacy problems and some explaining on how permissions work on Android.

In the next chapter we will explain the conditions and the tools that will be used to analyze applications. After that we will post results for each application from tests that we have done. Next we will analyze results and make comparisons between applications and categories. Finally, we will give our conclusions and some suggestions on how to improve the privacy and the security in Android, based on our findings.

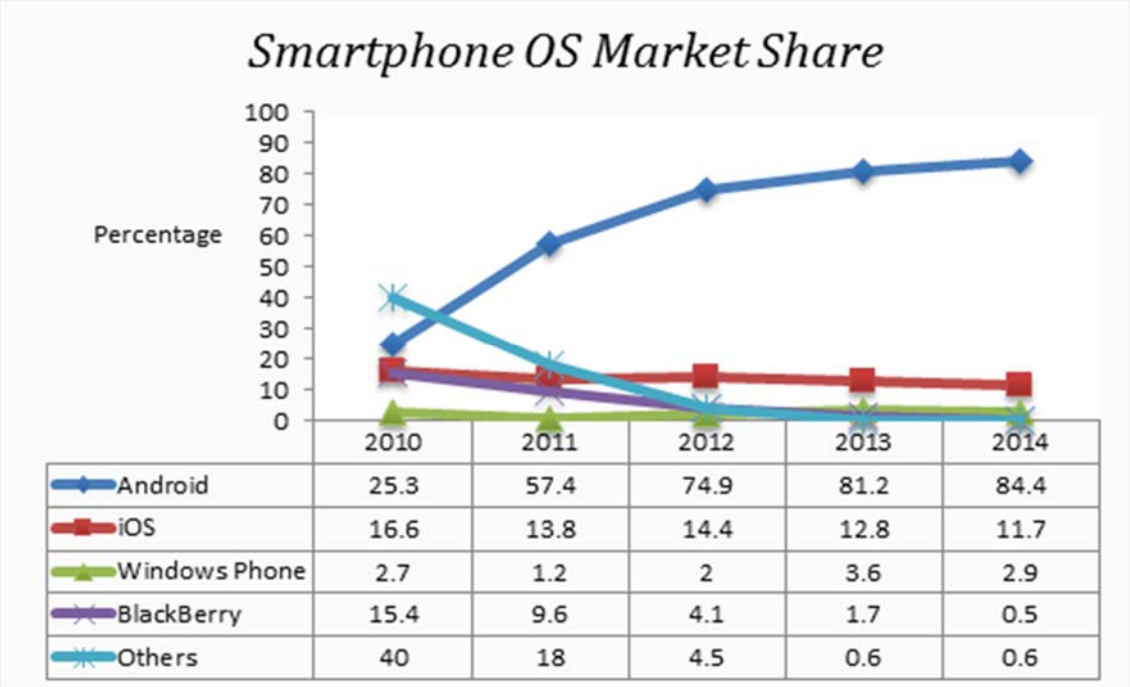
CHAPTER 2

AN OVERVIEW OF TOPICS

In this chapter we will give some background informations needed to understand how Android is designed to work. We will give a short introduction to history of Android and malwares on it. After that we will explain how permissions work for developers and for users. In the end will shortly explain some other problems with this OS.

2.1. Android OS

Android was acquired by Google since 2005 but from 2010 it started a very fast growth by powering more devices than all other mobile vendors.



	2010	2011	2012	2013	2014
Android	25.3	57.4	74.9	81.2	84.4
iOS	16.6	13.8	14.4	12.8	11.7
Windows Phone	2.7	1.2	2	3.6	2.9
BlackBerry	15.4	9.6	4.1	1.7	0.5
Others	40	18	4.5	0.6	0.6

Figure 1. (**Table 1.**) Smartphones Operating System market share by years

Android powers more than 1 billion devices and cybercriminals have started their activities on this operating system. Every day on the official Android store are

submitted more than 800 new applications and games. Although Google has implemented a system to verify applications and games if they contain malicious code before publishing them to store, Android still has a lot of problems with third party applications.

Android malwares go throw a very fast growth in last years. From 2011 to December 2014 malwares detected have increased by 300 times [quickheal.com, February 2016, The State of Android Malware in 2014]. In order to understand how fast this growth is, let's have a look at the figure below which represents malwares detected from January to December 2014.

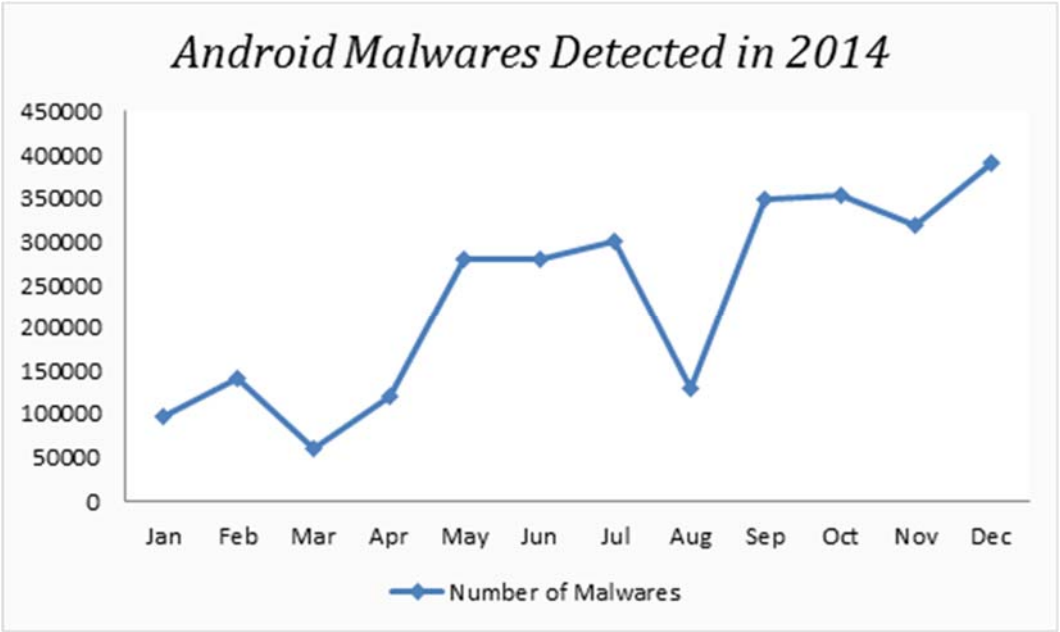


Figure 2. Android Malwares detected by month in 2014

Taking in consideration that except than Play Store, there are third party stores offering applications and games for Android this risk becomes bigger. One of the most popular strategies to distribute malicious apps from third party stores is to repack popular apps with some malicious code in it. Anyway in our study we will not take in consideration this kind of applications. We will make a study and analyze permissions and behavior of official applications from Play Store.

2.2. Application Permissions

Android architecture is created that way by default so no application can risk the privacy of user data because it cannot have access to restricted areas. For an application or game, to access restricted resources are required permissions.

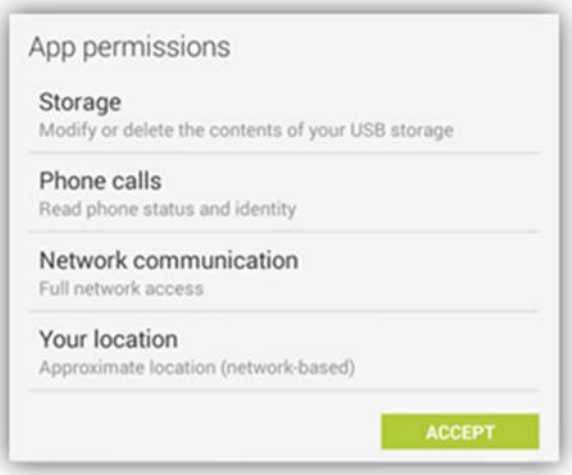


Figure 3. Screenshot from Play Store installation dialog, displaying example permissions required by an application.

Permissions are the way to restrict applications to access whatever they want in device without user approve. For an application to obtain permissions is a process with just one step from developer and one step from user. The developer just has to declare in the application manifest file, permissions needed and that's all. In the figure above we can see the warning which is shown to the user every time it installs a new application or when the application update requires a new permission.

The system is created that way so users can only accept all permissions and install the application or cancel installation. Installing application from third party stores, users may not be warned by store but android system will anyway show the list of permissions needed before installing it.

Our study will focus only on installations made by Play Store. Another thing we can notice in Play Store is the application description, some screenshots and also User Reviews and User Ratings.

2.3 Architecture and Risks

On most cases attack on Android are made using third parties applications installed on device which are abusing with permissions given by users or with packets sending/receiving over network. There are also some other problems or kind of attacks on Android. Below we will explain them.

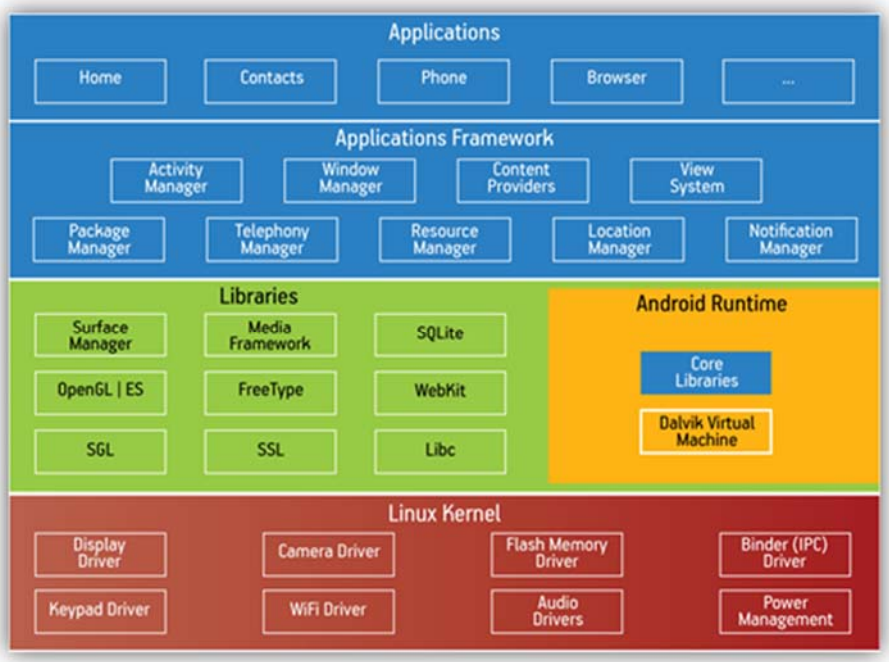


Figure 4. Android OS Architecture

As you may know Android OS is based on Linux Kernel. All input output requests received by the software are managed by Kernel. Some versions [Versions lower than Jelly Bean, Android 4.0] of Android are built with outdated version below 2.5 and some bugs became public just before these Android versions was published. So outdated kernel version increases the vulnerability of the OS.

Android like all other systems may have vulnerabilities discovered when the system is already installed on devices. When a problem is found in the system, Google has to create a patch to fix this vulnerability, merge it with the old system and send it to devices manufacturers. Manufacturers also adjust the code to be compatible with their devices, carrier, services etc. In the final step, an updated ROM [Package containing an Android System Image] is pushed to devices over the air so everyone who has an outdated version of Android, can download and install it. During the time that vulnerabilities are discovered but not fixed the user may be under attack. This problem is bigger for devices that never get updated once they are released in market.

Rooting [Granting owners administrative privileges on device] is also a security risk and the problem is the execution of native code. Native code is not executed in Dalvik VM like Java does. In this way malicious native code can download and execute binaries that can make the device very un-secure. In rooting mode native code can grant unlimited access to system resources like files and directories.

Usb debugging is a great feature for developers because it assures a higher level of access to devices but for random users is not at all. When debugging is enabled, user may get "infected" when is connected to a PC. Through Android Debug Bridge protocol, a user or a software can install applications or transfer data without a consent from device.

We have described some of the security problems in Android. There are also other problems that can leak informations from device or can affect users privacy but our study will not focus on them.

CHAPTER 3

TESTING CONDITIONS

Before starting with testing we will give an explanation of conditions and tool that will be used. Also we will explain two categories of apps we chose, device where they will be installed, and tools used for analyzing.

3.1. Categories chosen

We had to choose two categories of apps for testing and we chose Games and News & Magazines from applications. This was because they are very different not only from the functionalities but also resources needed, target of users builded for, amount of total downloads etc. Games require more hardware resources like CPU, GPU or RAM. Instead News & Magazine apps will need network to update frequently and very little amount of resources.

We will compare 5 *Top New Free games* with 5 *Top New Free apps* from News & Magazines. Difference between Top Free and Top New free is because the second has applications submitted to Play Store in last 30 days, instead 5 Top Free apps are filtered for a lot of time and they have billions of downloads, so probably will have less problems with permissions and other aspects.

3.2. Tools used

3.2.1. Network

Fiddler 4 is a tool that creates a virtual proxy server on the machine installed and all packets transmitted on that machine are captured and you can do whatever you want with them.

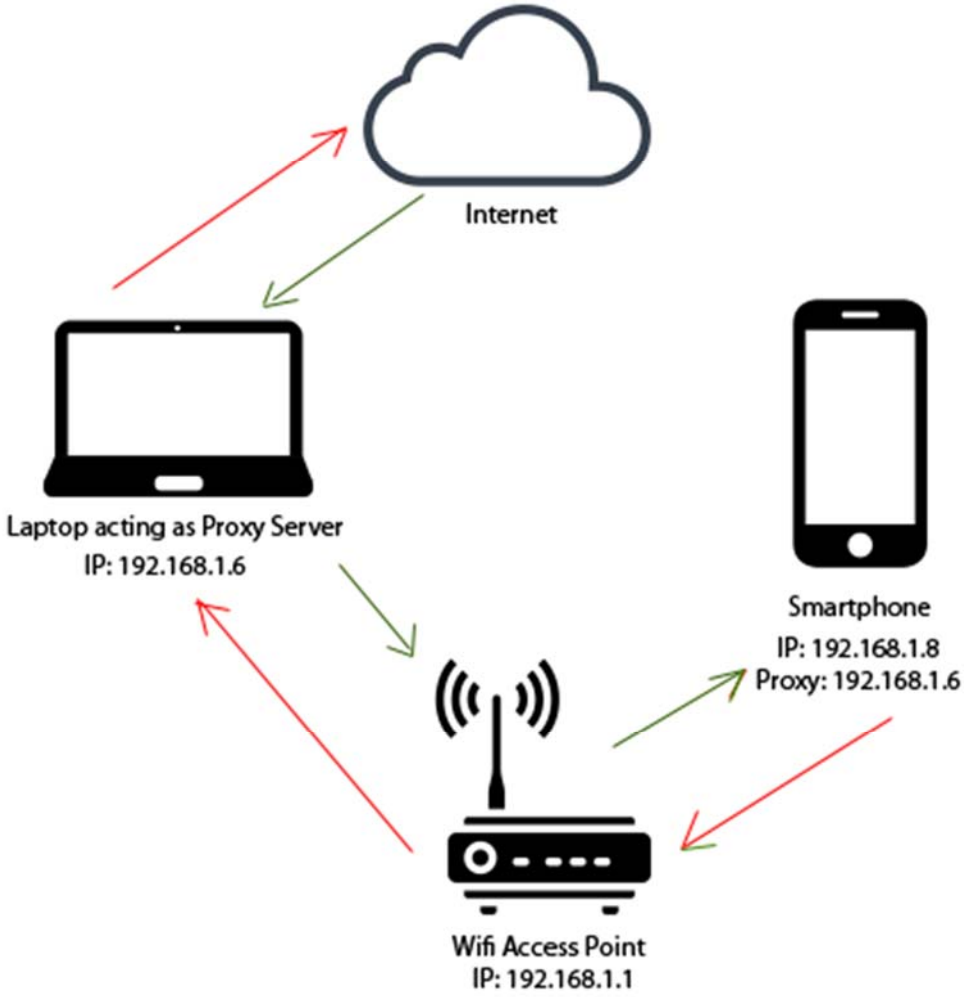


Figure 5. Connection between device and internet using machine as Proxy server

To be able to communicate with each other, device and the machine (in our case a Laptop with Windows 10) should be in the same network. In a normal situation, communication will be Smartphone -> Wi-Fi AP -> Internet. After configuring device to connect with proxy, laptop acts like a middle layer and all packets pass through him.

Last thing to do is to filter in Fiddler application, all packets sent from laptop itself so we can have only packets from device. To do this, we just need to filter all applications processes from Windows.

3.2.2. Performance

Android Monitor is tool inside Android Studio (official IDE for building apps on Android) that helps developers to debug apps and see graphs for hardware utilization. It only needs the device to be connected with PC in developer debugging mode.

In this monitor we can see utilization of hardwares like CPU, RAM or size of network packets. Android monitor also can capture screenshots or videos while using app. It also generates some detailed reports about activities, broadcast receivers, services, databases created by app, permission declared.

3.2.3 Other tools

Android Commander is another tool used as file explorer from PC to Android. When the device is connected, it can explore all directories on device and act like a dual explorer between device and PC. We used this tool to get files written on device storage directly and then analyze them on PC.

3.2.4 Device for testing

Each application is installed on a physical device for testing. In this research we used a Lenovo K3 Note powered by Android 5.1.1 (Lollipop) API 22. Phone has a CPU Octa Core processor 1.7Ghz and 2GB of RAM. Before testing each application, the phone is fully wiped and only operation system is installed, no user applications. This was to avoid mixing of packets sent over network by other applications while we are testing our application. Device is connected only with Wi-Fi because on data connection we can’t communicate with proxy server.

CHAPTER 4

TESTING RESULTS

Every application has been installed on device and tested. Below we will write results for each one with our comments about suspicious behavior. We will list only permission that are somehow suspicious for the category of application.

4.1 Top Free Games

1. Cosmic Challenge

Developer total: >5 M

Installs total: >100 K

Rating: > 4.4

Memory Average: 20 MB RAM

CPU Average: 20%-50%

PERMISSIONS:

- read phone status and identity
- ***modify or delete the contents of your USB storage***
- read the contents of your USB storage
- ***take pictures and videos***

Registered for push notifications to a third party service. (gamedonia.com)

On internal storage /data/data/com.ivanovichgames.cosmicchallenge we found a file encrypted with device informations like brand, model, time zone, location.

On network packages we didn't capture any suspicions package. Only 2 packets where sent to application backend server bestwordgames.net with json data for

synchronization. Other packets where encrypted with SSL and unable to see content inside. (unity3d.com, chartboost.com, applifier.com, zapjoy.com, facebook.com, adcolony.com)

Application tried multiple times to prompt user to login to Facebook without user request to login or share anything.

2. Blocky Soccer

Developer total: >16 M

Installs total: >500 K

Rating: > 3.9

Memory Average: 35 MB RAM

CPU Average: 25%-35%

PERMISSIONS:

- read phone status and identity

Registered for push notifications to official GCM (Google Cloud Messaging) service.

On internal storage we found only some encrypted files with an algorithm we could not decode. So we are unable to see what's inside those files.

On network packages we captured some packets containing unauthorized and unneeded informations. More than one packet contained inside, device id, device model, os version, resolution, apps store package, ***carrier id, carrier name***. Other packets where encrypted with SSL and unable to see content inside. (sponsorspay.com, facebook.com, google.com, fyber.com, flurry.com, googleapis.com)

This application contains a service which starts on boot even when game is not running.

3. My Caffe Recipes & Stories

Developer total: >5 M

Installs total: >1 M

Rating: > 4.6

Memory Average: 22 MB RAM

CPU Average: 20%-35%

PERMISSIONS:

- find accounts on the device
- modify or delete the contents of your USB storage
- read the contents of your USB storage
- run at startup

Registered for push notifications to a third party service. (parse.com)

File with device informations found in internal storage. File contains device name, model, manufacturer, device id, os version, api version.

The same informations captured on a package sent to another third party marketing platform (applovin.com) when it is not normal to send personal informations to other third parties. Other packets where encrypted with SSL and unable to see content inside. (facebook.com, sponsorspay.com, fyber.com, adjust.com, parse.com)

4. Cristiano Ronaldo Kick N Run

Developer total: >13 M

Installs total: >50 K

Rating: > 4.8

Memory Average: 35 MB RAM

CPU Average: 20%-45%

PERMISSIONS:

- find accounts on the device
- modify or delete the contents of your USB storage
- read the contents of your USB storage
- read phone status and identity

Registered for push notifications to official GCM (Google Cloud Messaging) service.

On internal memory we found a file with following content: appstore package, device, brand, os version, screen, sdk, user agent, country, ***carrier id, device currency.*** Another files have been encrypted with multiple algorithms because with base64 and url decoder, file was decoded with success but informations are still encrypted.

All packets from GET and POST requests to backend server (crazzyfrog.io) where encrypted. Also other packets where encrypted and unable to see content inside. (unity3d.com, mobileapptracking.com, facebook.com, googleapis.com, flurry.com)

5. Blodes of Brain

Developer total: >100 K

Installs total: >100 K

Rating: > 4.7

Memory Average: 25 MB RAM

CPU Average: 25%-50%

PERMISSIONS:

- modify or delete the contents of your USB storage
- run at startup

Registered for push notifications to official GCM (Google Cloud Messaging) service.

On internal memory we found a file encrypted and its content could not read inside.

Also the packets on network where encrypted and could not read content. (applifier.com, kochova.com, fyber.com, adcolony.com, googleads.com, sponsorspay.com)

4.2 Top Free News & Magazine

6. News Dog Lite

Developer total: >100 K

Installs total: >100 K

Rating: > 4.1

Memory Average: 55 MB RAM

CPU Average: 10%-40%

PERMISSIONS:

- modify or delete the contents of your USB storage

- run at startup
- record audio
- phone status and identity
- location
- get accounts on device
- create home shortcuts
- wake lock keep wake

Registered for push notifications to official GCM (Google Cloud Messaging) service.

Application didn't create any file with suspicious content. Instead on network packets we captured some of them with device content but one packet sent to newsdog.today contained: (device model, device id, ***device mac address, imei number, imsi number***). Also we captured some other packets send data encrypted and receiving encrypted data also. Unable to read their content.

7. Cosmojis

Developer total: >200 K

Installs total: >10 K

Rating: > 2.9

Memory Average: 20 MB RAM

CPU Average: 15%-25%

PERMISSIONS:

- modify or delete the contents of your USB storage
- run at startup
- phone status and identity
- find precise location
- get accounts on device
- create home shortcuts
- prevent device from sleeping
- change network connectivity
- connect and disconnect from Wi-F

Registered for push notifications to a third party service. (parse.com).

While testing this application didn't found anything strange in network packets. It has written some files to storage with configurations but only one is encrypted. The problem with this app is that he has permission to keep device awake, change network

connectivity, turn off/on and he can use this method to send receive private packets. All permissions required are a suspicious for the functionalities of application.

8. lala elhadga

Developer total: >100 K

Installs total: >10 K

Rating: > 4.6

Memory Average: 20 MB RAM

CPU Average: 15%-25%

PERMISSIONS:

- modify or delete the contents of your USB storage
- prevent device from sleeping
- find accounts on device

Registered for push notifications to official GCM (Google Cloud Messaging) service.

On internal storage there are no encrypted files. Only some files for configurations like shared preferences, SQLite database of articles and that's all.

In network packets detected one that contains some informations for device like device id, gcm id, and device model.

9. In Now News Reader

Developer total: >200 K

Installs total: >10 K

Rating: > 2.9

Memory Average: 30 MB RAM

CPU Average: 20%-30%

PERMISSIONS:

- modify or delete the contents of your USB storage
- run at startup
- phone status and identity
- find precise location
- get accounts on device
- create home shortcuts
- read sensitive log data
- change system settings
- prevent device from sleeping
- change network connectivity
- connect and disconnect from Wi-Fi

Registered for push notifications to official GCM (Google Cloud Messaging) service.

In storage we didn't find any sensitive information. All files where configuration files.

On network (particlenews.com) we captured packets sending device informations on internet. (like device model, brand, device id, mac address, imei).

10. Pakistan Jobs

Developer total: >1 K

Installs total: >1 K

Rating: > 4.6

Memory Average: 20 MB RAM

CPU Average: 10%-25%

PERMISSIONS:

- modify or delete the contents of your USB storage
- phone status and identity
- prevent device from sleeping

Registered for push notifications to official GCM (Google Cloud Messaging) service.

On internal storage only configuration files without encryption found and the database used for articles cache.

Even on network packets didn't found anything not normal, just syncing infromations.

4.3 Analyzing test

As we can see from the results above, the problem is real. Almost all apps are abusing with permissions, network packets send or received and content written on files.

In the first category we have top games and top developers with millions of installs and we see applications asking for permissions that are not needed for its functionalities.

All applications tested where using GCM push messages and this can be good for some reasons but also can be bad. If the developer writes data in a file and on a chosen moment by him, he sends a request to device to send file over network. All applications from Top Free Games used encrypted file and encrypted packets. The same situation here, it may be good because it protects user data from other people on network but it can be the developer encrypting files to receive private user informations.

We tested a game (1) that asked for permission to take photos while the game doesn't need photo or camera at all. The app was always prompting user to login to Facebook without his request. All this is not normal because apps are not used to force users to do something they don't want.

In another game we captured packets with contents like device id, device model, carrier id, carrier name and some other encrypted. Five apps (3,4,6,7,8,9) granted permission to get all account connected on device. Three apps (3,5,7) granted permission to start when phone boots. If we think that for news applications is normal because it should synchronize latest data what about the game, it should be running only when the user is playing. Six from ten apps asked permission to keep phone awake or wake whenever it wants. Two other apps that wants to record audio or to take control of network connectivity are a reason more to think that these applications are not having a normal behavior.

If we make a comparison between two categories of apps chosen, we can clearly that they are different even in the aspect of privacy and security. Top Games are apps downloaded by millions and their communication is encrypted most of the time and all data they write on files are encrypted. Permissions are not asked as much as by applications from Top News & Magazines. In difference Top News and magazines apps require permissions but the data they send are not encrypted. Content

on these packets is mostly json for applications synchronization. Also data writed on internal storage are encrypted very rarely.

CONCLUSION

Android applications with a bad behavior are becoming a real problem for the future of the OS. Google has started with some improvements on latest os update, Android M (Marshmallow) but it looks like this move will not solve the problem. In Android M permissions can be required while app is running and when the permission is needed. Anyway all developers should update their applications in order to use this feature and leaving permissions in control of the user. While apps are not updated things are the same.

OUR SUGGESTIONS

We think that there is not a single application or magical tool that will make the permanent fix. There are a lot of things that can be changed in os without losing application functionalities and by giving to user more protection.

1. If permissions will be more specific would be easier for user and for Google itself to detect bad behavior. e.g. permission “read phone status and identity” allows developer to access phone feature, place or receive calls, detect phone number and sim serial number, number of the person you are talking on phone and his contact card. If we had more specific permissions, will be easier to understand if permission asked is normal or not for application functionalities.
2. When installing the app from Play Store should be available option to install without accepting all permissions, choose only those you think are not a problem.
3. On Play Store developers should be required to describe for each permission why it is needed and what impact does it have on app.
4. Also Google has to improve the system of analyzing apps before making it available for all users.

REFERENCES

1. http://www.idc.com/prodserv/smartphone-os-market-share.jsp (April 2016) Smartphone OS Market Share, 2015 Q2
2. http://blogs.quickheal.com/state-android-malware-2014/ (May 2016) The State of Android Malware in 2014
3. Fan Yuhui, Xu Ning “The Analysis of Android Malware Behaviors”, 2015
4. Iker Burguera, Urko Zurutuza, Simin Nadjm “Behavior-Based Malware Detection System for Android”, 2012
5. Nitin Padriya, Nilay Mistry “Review of Behavior Malware Analysis for Android”, 2013