Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 2, Issue. 9, September 2013, pg.247 – 255

RESEARCH ARTICLE

Performance Base Static Analysis of Malware on Android

Khyati Rami[1], Vinod Desai[2]

[1]Ph.D Research Scholar, Mewar University, Rajasthan, India

[2]Department of Computer Application, Kadi Sarva Vishwa Vidyalaya, Gujarat, India

Abstract— According to a Gartner study (Gartner, 11/2010), Android is now the No. 2 worldwide mobile operating system and will challenge Symbian for No.1 position by 2014. In addition to Android's large market share, the number of Android applications is growing at a fast rate. There are currently more than 100,000 Android applications available (Techeye, 26/11/2010). With the increasing numbers of applications available for Android; spyware is becoming a real concern. Several malicious applications, ranging from fake banking applications to an SMS Trojan embedded into a fake media player, have already been discovered on the Android Market since the beginning of this year. However, there are other forms of malware that may also emerge. What about hiding spyware in the background of a well-known application? For example, imagine an application claiming to be the latest version of a famous Twitter client, which actually runs spyware in the background and uploads all private data to the attacker. The purpose of this paper will be to explore a study of static analysis on Android and provide real case malware attack scenarios. Reverse engineering will be used, because most users do not check the permissions of the applications loaded onto their mobile device.

Keywords: - Malware; Static analysis; translator; Reverse Engineering

I. INTRODUCTION

Android is an open-source mobile operating system, based upon a modified version of the Linux kernel, initially developed by Android Inc., a firm purchased by Google in 2005.A Gartner study released on November 2010 outlined that Android has become the second-most popular OS in the world (Gartner, 11/2010). The growth of Android has exceeded their previous study, released last year, in which they had predicted that Android will be the No.2 worldwide mobile operating system in 2012 (The H, 08/10/2009). According to another Gartner study (Gartner, 08/2010)., there will be only a slight difference between Symbian and Android market share in 2014: 30.2% for Symbian against 29.6% for Android.

All Smartphones are preferred targets of attacks. These attacks exploit weaknesses related to smartphones that can come from means of communication like short messaging systems (SMS), and multimedia messaging systems (MMS)., Wi-Fi networks, and global system for mobile communications (GSM). There are also attacks that exploit software vulnerabilities from both the web browser and operating system. Finally, there are forms of malicious software that rely on the weak knowledge of average users.

In August 2010, the first SMS Trojan targeting the Android platform appeared. This Trojan named Trojan-SMS.AndroidOS.FakePlayer.a penetrates smartphones running Android in the guise of a harmless media player application. Once installed on the phone, the Trojan uses the system to begin sending SMSs to premium rate numbers without the owner's knowledge or consent, resulting in money passing from a user's account to that of

© 2013, IJCSMC All Rights Reserved

the cybercriminals (Kaspersky, 08/2010). A complete analysis of this Trojan can be found on Jon Oberheide blog (Oberheide, 10/08/2010). Since the first release of this SMS Trojan, several variants have appeared in the wild under the name of pornplayer.apk masquerading also as a media player (Kaspersky, 09/2010).

II. MOBILE SECURITY STACK

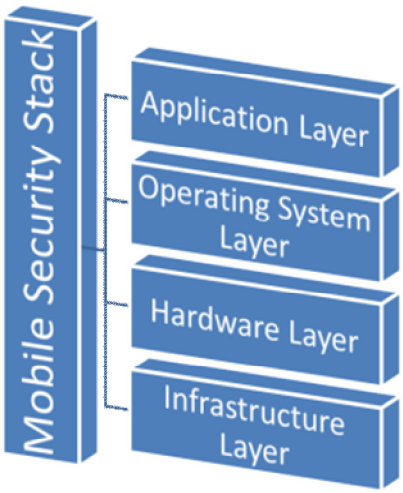


Fig.1 Mobile Security Stack

The mobile security stack can be broken up into four distinct layers. The lowest layer of the stack is the infrastructure layer, followed upward by the hardware, operating system and application layers. These security stack layers each define a separate section of the security model of a smartphone or mobile device.

Each layer of the model is responsible for the security of its defined components and nothing more. The upper layers of the stack rely on all lower layers to ensure that their components are appropriately safe. This abstraction based model allows the design of a particular security mechanism to focus on a single specific area of concern without expending the resources required to analyse all layers that support its current location within the stack. Websites, swap YouTube videos and many more features to come.

Direct attack:

- Direct attack is called when an intruder enters into the device and makes deliberate actions to make some damage. One way to perform this attack is social engineering, through simple observation of the target. It suffices to identify the device when it has asset infringing services. With the aforementioned, the hacker determines the best exploit to use against that device. One of the methods used to locate devices is through the signals they emit such as Bluetooth which will be used to exemplify this type of attack. After locating the device and accessing it, the hacker can perform various actions that can break the stability of the device, such as delete, edit, upload data, change settings or use the device in an unauthorized manner.

III. MALWARE LIFE CYCLE

Malware consists of the following stages:

Birth:

In this stage, bringing the mobile malware to life, malware writer (the person who wrote the malware) designs the malware and then creates it using a programming language.

Release:

In this stage the malware writer sends it out to the wild (the cyberspace, the virtual mobile world) via wifi, Bluetooth, or through any other networks.

Propagation:

Malware replicate by nature. The malware target in this stage is to replicate and infect as many victims as possible without drawing any attention. A well-designed malware will replicate for a long time before it activates, which allows it plenty of time to spread.

© 2013, IJCSMC All Rights Reserved

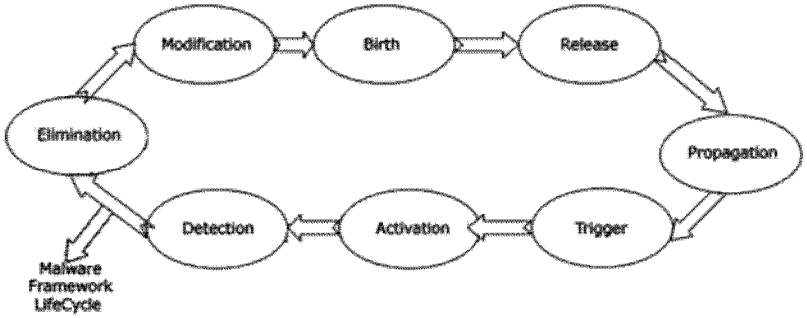


Fig.2 Malware Life Cycle

Trigger:

In this stage the malware becomes alive when the trigger is reached. The malware writer usually determines the trigger; it could be a specific date, a certain task, or anything else depending on the writer's choice.

Activation:

In this stage the virus has the ability to run its destructive routine. The effect of this could vary from erasing phone memory content to making limited damage. Malware that have damage routines will activate when certain conditions are met, for example, on a certain date or when a particular action is taken by the user. Malware without damage routines don't activate, instead causing damage by stealing storage space.

Detection:

This could happen at any stage of the malware lifecycle, detecting the malware in the early stages makes it easier to remove it without causing any damage.

Elimination:

The ability to eliminate the effect of malware varies from one type to another, and also depends on the available tools. The solution could be simple and inexpensive e.g., deleting the malware or complicated and expensive e.g., reformatting and restoring the phone memory or buying a new one.

Modification:

In this stage the malware lifecycle may be repeated with an improved version, this could be done by the original malware writer or someone else.

IV. MALWARE FRAMEWORK

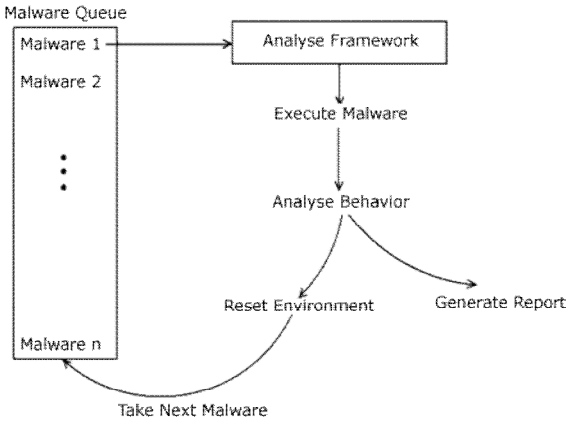


Fig.3 Malware Framework Life Cycle

© 2013, IJCSMC All Rights Reserved

All the malwares which were detected are sent to the mobile security company by submitting through mobile antivirus program. The mobile security company collect all the malware sent by the users and then analyse them one by one. They take one malware at a time for analysis and start executing the code and analyse its behaviour and generate report. If any solution is available then they reset the environment and take next malware for analysis. The generated reports are sent back to user as an antivirus software update.

V. MALWARE DETECTION TECHNIQUE

The techniques available for detecting mobile malware and other security vulnerabilities have varying strengths and weaknesses.

Static analysis

Static analysis is a quick, inexpensive approach to finding malicious characteristics or bad code segments in an application without executing them. The techniques illustrated in Fig.15 are widely used in a preliminary analysis, when suspicious applications are first evaluated to detect any obvious security threats.

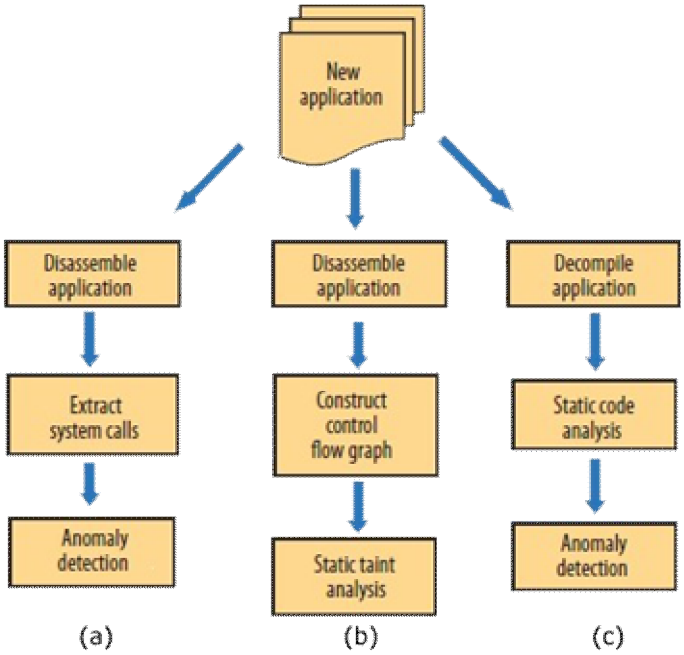


Fig.4 Static Analysis Techniques based on System Calls (b) Taint(c) Source Code

Fig.4(c) shows a malware detection technique proposed for Android.

VI. IMPLEMENTATION OF STATIC ANALYSIS OF MALWARE ON ANDROID

Reverse engineering is a process of analysing the possible functionalities of an application. In order to analyse a malware there are two methods, namely, static analysis and dynamic analysis. The tools used for the same are discussed below.

Tools:

Static analysis android applications can be performed using various tools. In this project we use the method which involves the utilization of *APKTOOL* and an editor such as *Notepad++* to analyse the malware code.

⇒ ApkTool:

The ApkTool is a 3rd party tool that is used to analyse closed Android application binaries. Its features include: This tool is capable of disassembling applications to practically original form and repackaging them after certain modifications It also enables the user to debug the smali code. It allows the user to work with the applications in an easier way as it provides automation of some recurring tasks and project-like file structures.

© 2013, IJCSMC All Rights Reserved

⇒ Android SDK:

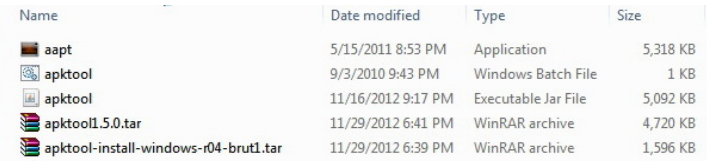
The Android Software Development Kit (SDK) is a collection of development tools that are used to create applications. The following components/tools are included in the SDK:

- ✓ Debugger
- ✓ An emulator
- ✓ Necessary libraries
- ✓ Sample source codes
- ✓ Appropriate documents for Android APIs (Application Program Interfaces)

Windows, Linux and Mac OS X are platforms that are compatible with Android SDK. The components of the SDK can be downloaded independently or 3rd party additions are also obtainable for downloads. Android SDK can be used to develop applications by writing programs in the command prompt, however, it is more feasible and easy to use an Integrated Development Environment (IDE). The most preferred IDE is Eclipse with ADT (Android Development Tools) plug-in.

Procedure:

The first step is to download the pre-packaged ApkTool and extract it using WinRAR to a directory of your choice.



Name	Date modified	Type	Size
aapt	5/15/2011 8:53 PM	Application	5,318 KB
apktool	9/3/2010 9:43 PM	Windows Batch File	1 KB
apktool	11/16/2012 9:17 PM	Executable Jar File	5,092 KB
apktool1.5.0.tar	11/29/2012 6:41 PM	WinRAR archive	4,720 KB
apktool-install-windows-r04-brut1.tar	11/29/2012 6:39 PM	WinRAR archive	1,596 KB

ESET Antivirus Virus Notification

The ApkTool is available for download at "http://code.google.com/p/android-apktool". The malware, "Flash-Player_install", is then downloaded to the same directory as the ApkTool. It is important to note that just before downloading the malware,

Anti-Virus on the system needs to be turned off.Open the command prompt window and navigate to the root directory of ApkTool and type the following command:

apktool if "Flash-Player_install.apk"

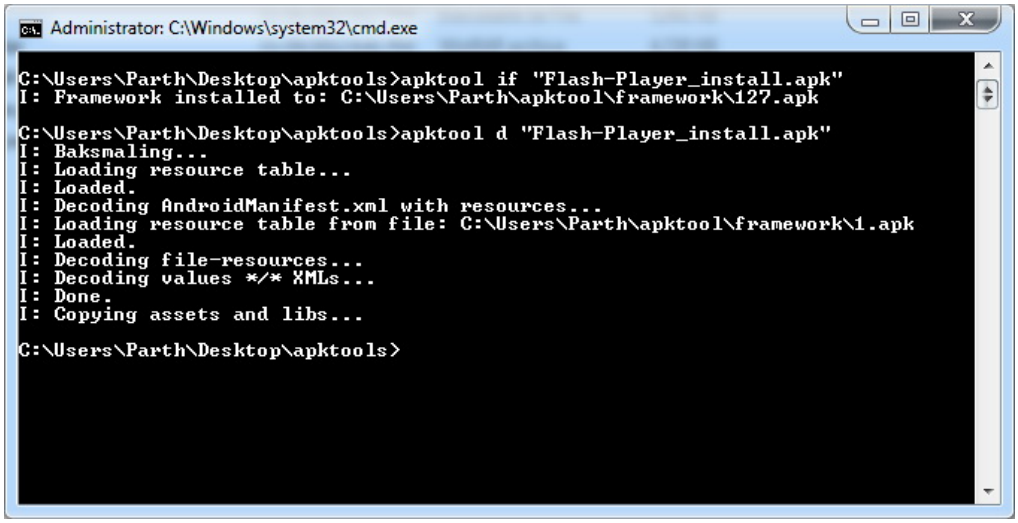
The output of this command is:

I: Framework installed to: C:\Users\Parth\Desktop\apktools\framework\127.apk

To decompile the apk file type the following command:

apktool d "Flash-Player_install.apk"

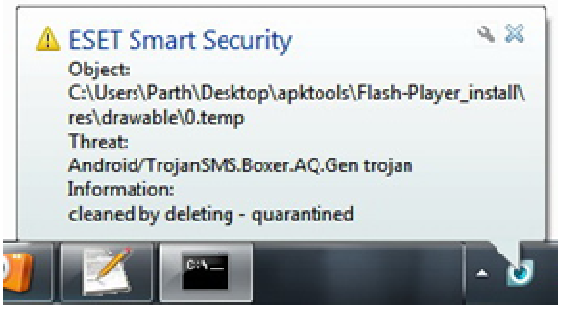
The output of this command is:



APK Tool : Decompilation

As we told before, the antivirus software should be disabled. We got a notification by antivirus program ESET Smart Security that "threat detected"

© 2013, IJCSMC All Rights Reserved



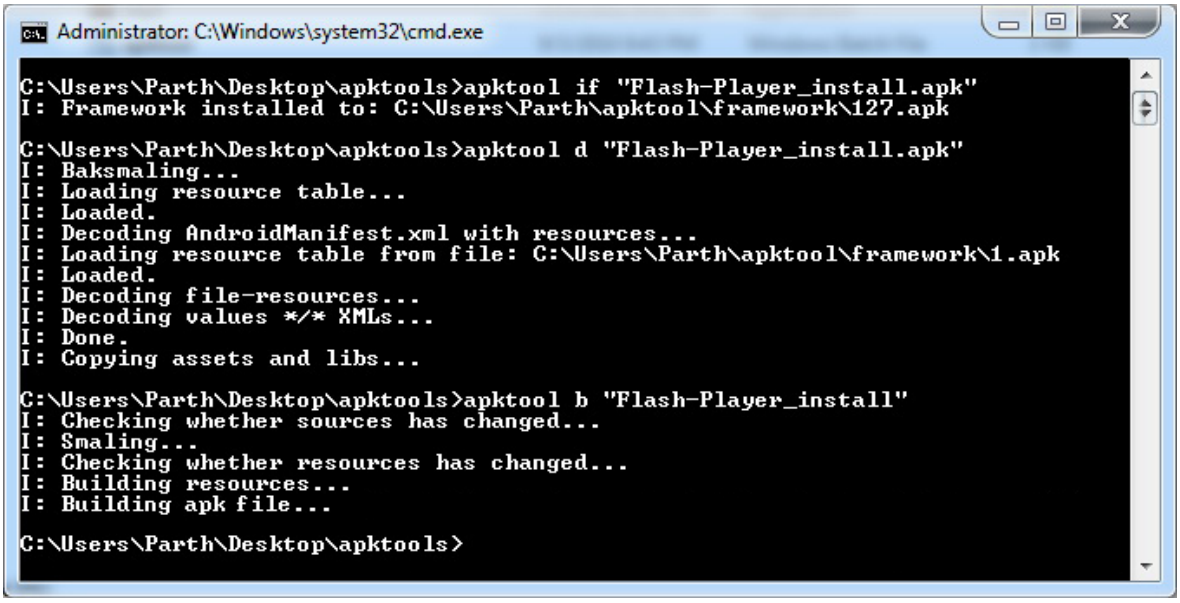
Once this command is executed, without any errors, a folder with the application name or in our case the malware name will be created in the same directory.

- ✓ Res folder: This folder contains XMLs defining the layout, drawables, attributes etc.
- ✓ Android Manifest File: It is one of the most important XML file which contains information about the permissions that the application needs or accesses.
- ✓ Smali folder: It contains the source code of the application in .smali format.

Once the code is analyzed and the malicious code is detected and explored we can make the necessary modification and repackage the application.

To repackage the application type this command:

apktool b "Flash-Player_install"

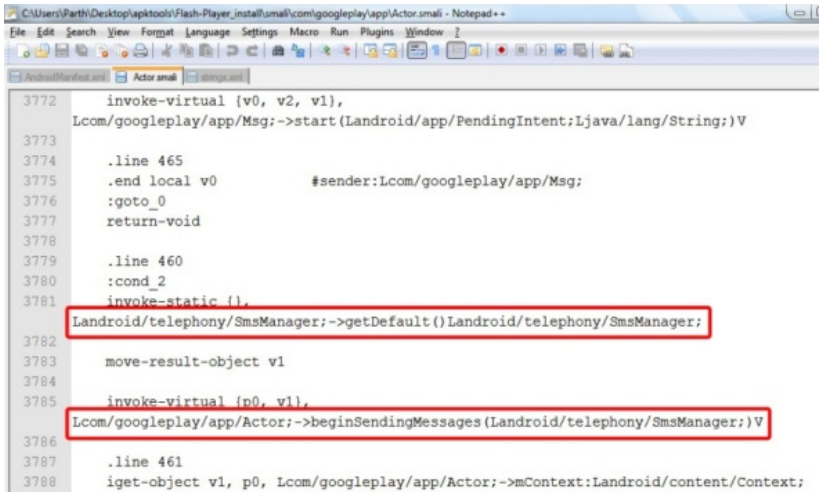


VII. RESULT

The android malware that is analysed in this project is called “Flash-Player install” which gets downloaded automatically when we install “com.rockastar.gtavicety_1” game application. This application “Flash-Player install” is a malware that apparently sends SMS to a different premium numbers without the user’s consent.

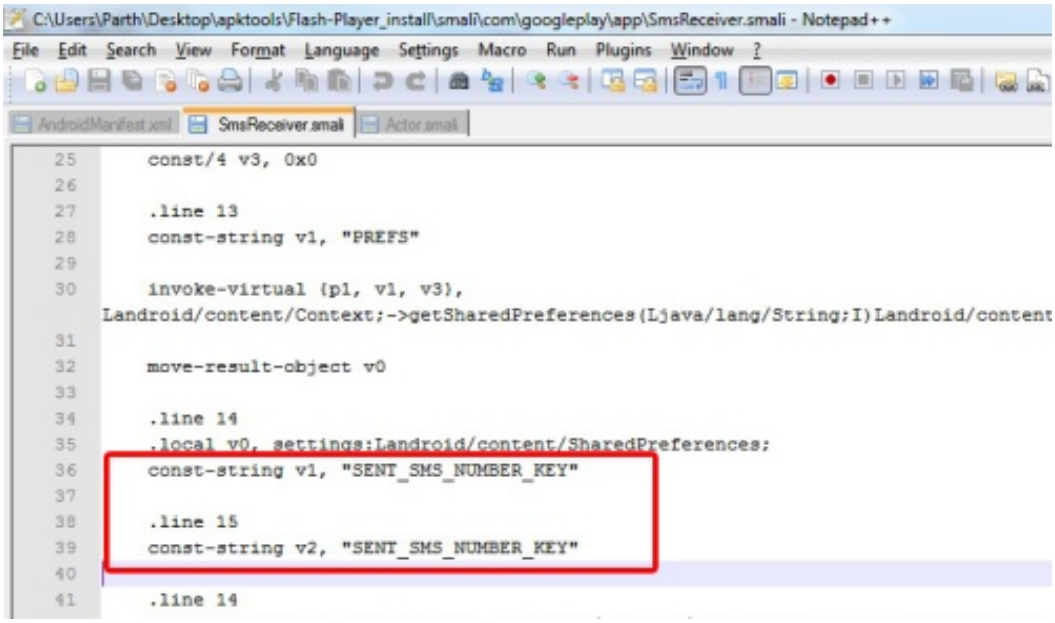
From the analysis made in above, we can confirm that this is a malicious application that aims to subscribe the user to monthly offers without his/her knowledge.

© 2013, IJCSMC All Rights Reserved



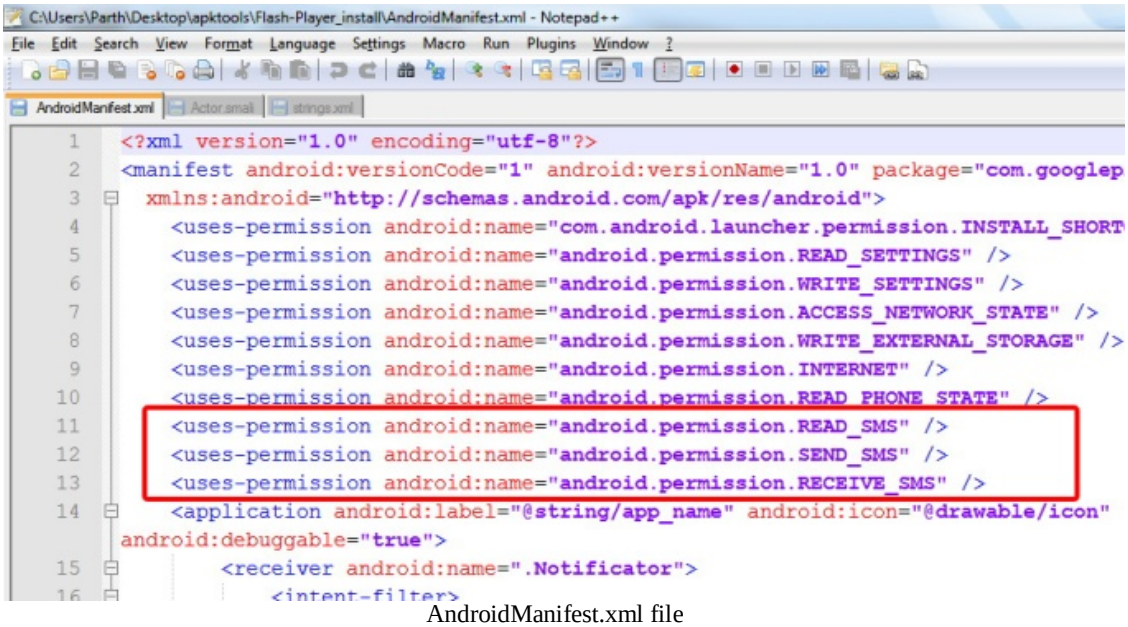
Actor.smali file

Let us first take a look at the smali files. It is evident that two files, namely, Actor.smali and SmsReceiver.smali, are the only files to contain the malicious code.



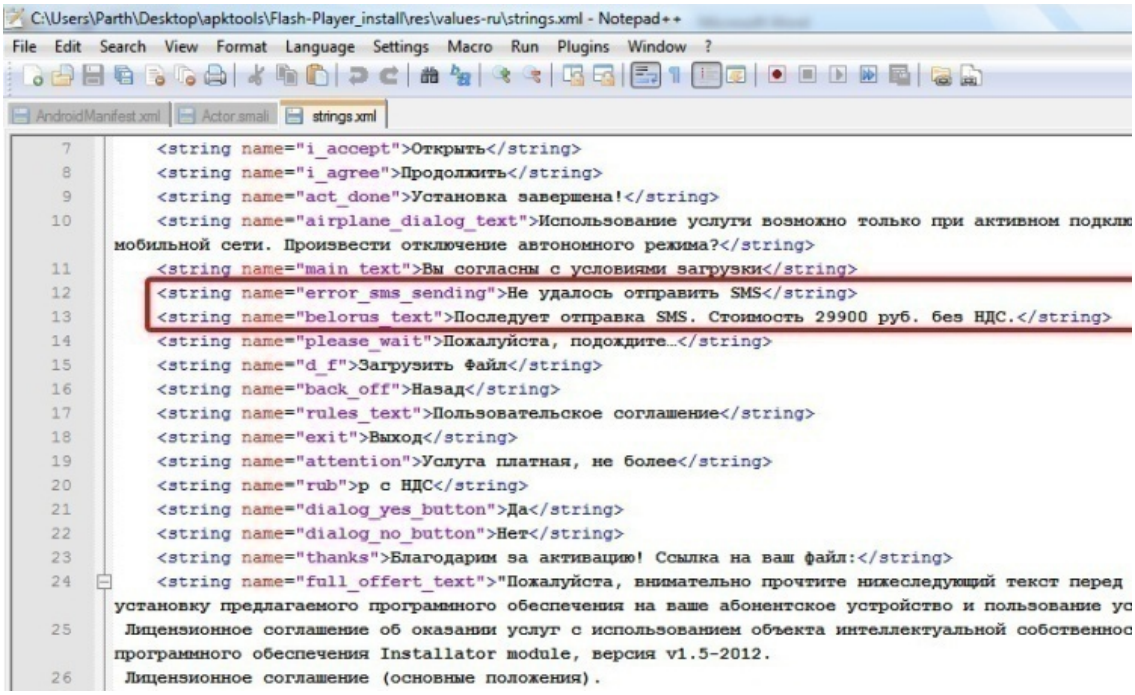
Let us also take a look at the Android Manifest File in order to see the permissions that the malware "Flash-Player_install" can access.

© 2013, IJCSMC All Rights Reserved



AndroidManifest.xml file

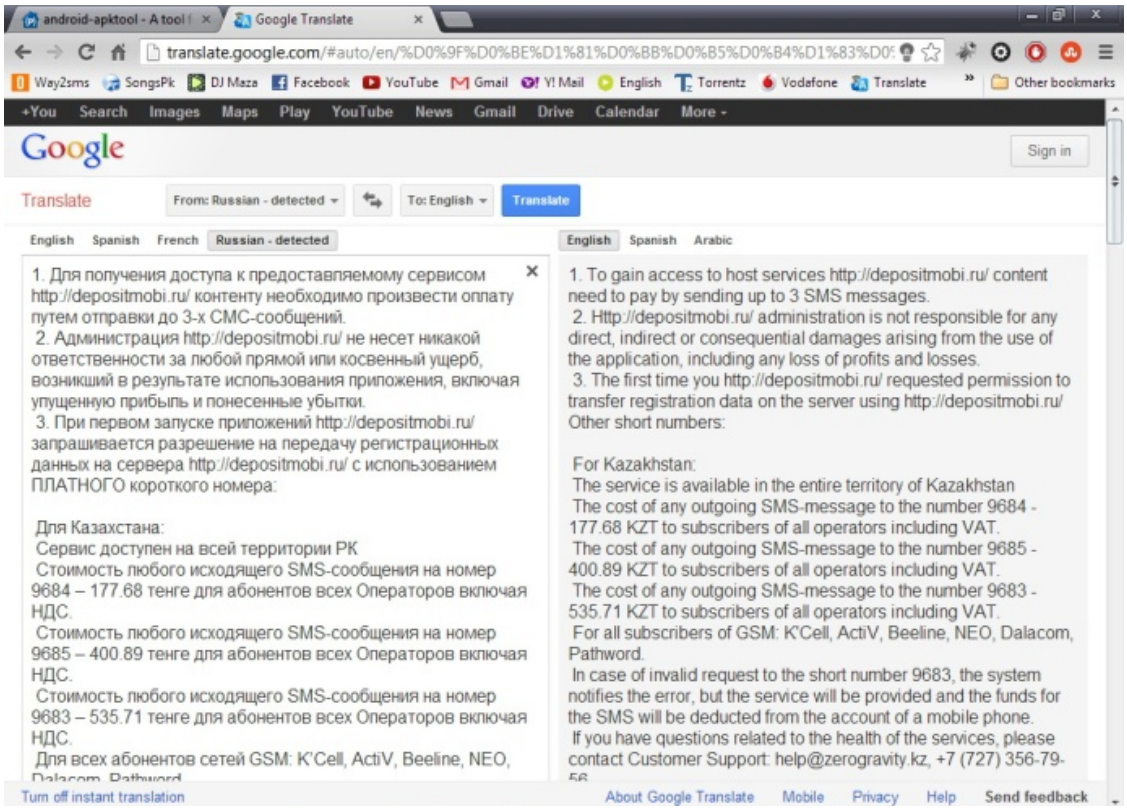
String.xml file located in "\res\values-ru" directory is a file which contains information written in russian language.



String.xml file

But when we copied some code of "string.xml" file and translated to english using "http://translate.google.com", we found some premium numbers along with their country names and call charges.

© 2013, IJCSMC All Rights Reserved



Therefore it becomes very crucial for every user to check the permissions that any application he/she is downloading really requires access to them or not. One of the major disadvantages of Android applications is that without agreeing to grant access to all the permissions, an application cannot be installed on the device.

The application "Flash-Player install" is only supposed to install flash plugins for the android device and do nothing more. Hence it is obvious that it does not require permission to send messages or receive messages etc.

REFERENCES

[1] Gartner. (11/2010) "Competitive Landscape: Mobile Devices, Worldwide, 3Q10." Retrieved from http://www.gartner.com/it/page.jsp?id=1466313
[2] Techeye. (26/11/2010) "More than 100000 Android applications are available". Retrieved fromhttp://www.techeye.net/mobile/more-than-100000-ndroidapplications-are-available
[3] The H. (08/10/2009) "Gartner: Android to be No. 2 smartphone by 2012". Retrieved from http://www.h-online.com/open/news/item/Gartner-Android-to-be-No-2-smartphone-by-2012-819890.html
[4] Kaspersky. (08/2010) "First SMS Trojan Detected for Smartphones running Android" Retrieved from http://www.kaspersky.com/news?id=207576156
[5] Oberheide, Jon. (10/08/2010) "Dexcode Teardown of the Android SMS Trojan" Retrieved from http://jon.oberheide.org/blog/2010/08/10/dexcode-teardown-ofthe-android-sms-trojan/
[6] Kaspersky. (09/2010) "Popular Porn Sites Distribute a New Trojan Targeting Android Smartphones". Retrieved from http://www.kaspersky.com/news?id=207576175
[7] Asaf Shabtai, Uri Kanonov, Yuval Elovici, Chanan Glezer, Yael Weiss "Andromaly": A Behavioral Malware Detection Framework For Android Devices", 2011, 163-170;
[8] Laurent Weber , "Dynamic Analysis of Malware", McAfee 2011, 3-7.
[9] Manuel Egele and Christopher Kruegel, "A Survey on Automated Dynamic Malware Analysis Techniques and Tools", 2011, 4-11.
[10] Raymond Roberts, "Malware Development Life Cycle", Virus Bulletin '08, 180-187.
[11] Vibha Manjunath, "Reverse Engineering of Malware on Android", Sans Inst.'11, 25-45.
[12] Joany Boutet, "Malicious Android Applications: Risks and Exploitation", Sans Inst.'10, 23-29.

© 2013, IJCSMC All Rights Reserved