Design and Implementation of an Integrated Testing Environment for Mobile Devices by Live DVD/USB

Wen-Han Chuang [1] Chung-Huang Yang [2]
Graduate Institute of Information and Computer Education
National Kaohsiung Normal University, Kaohsiung, Taiwan [1]
Graduate Institute of Information and Computer Education
National Kaohsiung Normal University, Kaohsiung, Taiwan [2]
jlfmy@hotmail.com.tw [1]
chyang@nknu.edu.tw [2]

Abstract

Nowadays, security issues of mobile devices draw growing attention. Although there are many security analysis tools for mobile devices on the Internet, but they seldom collected for developers and security researchers, so it causes developers and researchers have to consume a lot of time to install what the tools are related to mobile device security.

The goal of this research is to integrate useful security tools for mobile device; we divide the proposed tool set into six separate groups as follows: Development tool, Forensics tool, Penetration testing tool, Reversing tool, Wireless analysis tool, and Malware analysis tool. And we set up the whole tool set into a Live DVD/USB based on Ubuntu-14.04 LTS GNU/Linux. Additionally, we update the newest version of tools and change English GUI interface to Chinese GUI for the users. In summary, we propose an environment with friendly user interface for developers and researchers to evaluate mobile security.

Keywords ： Security tools; Android security; live DVD/Live USB; mobile device.

1. INTROCDUCTION

According to report from eMarketer [1], smartphone users worldwide will total 1.75 billion and 4.55 billion people to use mobile phone in 2014, this means the smartphone and mobile devices are rapidly becoming indispensable devices for many users. Unfortunately, they also become fertile grounds for hackers to deploy attack technology, many researchers in order to react to the rapid demand in the domain of mobile security, they develop a variety security tool of mobile device on the internet, but these tools have not been collected to an integrate system, so when the user face to the mobile security problems or they are interested in mobile security. First, they need to understand which tool can help them to solve the security problem. Second, they have to find where can download the tools. Third, they need to learn how to install and use them. This is a very time-consuming process.

Based on the above reasons, we propose an integrated environment for analyzing mobile Security, this environment integrate several famous and useful mobile device security tools on the live DVD/USB, it can reduce the time it takes the user to install and modify the English user interface into Chinese for easily understood purpose.

2. MOBILE SECURITY

This section describes the security issues of mobile devices as a basis for collecting mobile device security tools. According to the literature review, we divide it into four separate sections as follows: Operating system, security, application security, data storage security, data transmission security.

2.1 Operationg system security

Android and IOS platform for today's two largest mobile device operating systems, the operating system security problem can't be changed by application developers, users or researchers, it depend on the google or apple to upgrade the system or release the patch to fix the security problem, these problems like the following: Attacker found the method which could bypass the IOS screen lock to directly browse photos when IOS7 just released [2]; In 2013, Android vulnerabilities are be disclosure by Bluebox Security, it allows a hacker to modify APK code without breaking an application's cryptographic signature, to turn any legitimate application into a Trojan [3]. In the operating system's perspective, android is more complex than IOS, because it have several manufacturers which modify the android system to their own version, it cause the different problems in different manufacturer's mobile devices and the other problem is that android updating slower than IOS [4].

2.2 Application permission security

Application permission issues related to both developers and malicious applications, in the Adrienne Porter Felt's research, Due to developer's

＊ This work was supported by National Science Council grant (NSC 102-2221-E-017-003-MY3).

immature development skill, they ofthen lead to declared too many permission in the application, and these permissions never be used, it will reducing the security of mobile devices [5-6]; In malware respect, Due to the android application installation mechanism, the application have to display permissions to all user see, then ask user whether they agree to install, but in the Suarez-Tangil's research, users rarely read the permission detailed or they don't understand what is the permission meaning, lead to user install the malicious application, and malicious application which can do something other work they didn't inform the user [4]. And this kind of permission issues can come to understand the actual behavior of program rights issues and program execution through the Drozer security assessment Tool or DroidBox dynamic analysis tools.

2.3 Data storage security

In a series optimum safety measure of mobile devices by Tae Oh, one of the suggestion is that data are required to confirm whether the stored data is safe and be protected [7]. Ahmad divides Data storage security into two parts, external storage and built in storage. External storage's data are stored on SD card, however, the data on the SD card is a very dangerous storage method, once application stores the personal information data on the SD card without encrypting, the other application can read the data on the SD card just declare the normal permission, so it is easy be stolen [8].

The built in storage usually be used for storing private data, it is a more secure way than external storage, in the default mode, the data only can be accessed by the data owner, but in the other situation, malware can use vulnerable to get the root permission, and then the data also could be accessed.

Recent research has found that volatile memory in the mobile phone isn't safely, researchers could use volatile memory forensic tool to fetch the user data which in the volatile memory, studies have shown that the most of user password can be fetched by the tools [9].

2.4 Data transmission security

We divided data transmission security into two parts, one for the data transmission via the internet, and the second is data transmission between the applications or between the components. First, comes to the network transmission of data security, the traditional cyber security vulnerabilities are also occurring on mobile devices such as information leakage, social engineering attacks, Denial of Service attacks and spam bot engines [10]. Mobile devices translate the data to the server by wireless network, so it is more dangerous than traditional network. In [13] it is pointed that some Android applications are most certainly vulnerable to an invisible proxy MITM attack. The attacker could be at any public access point, have someone on the same network. The other serious problem is the transfer application data in plain text in the way of transmission, simply use the Internet Sniffer tool or Proxy tool in this case, it can be can capture the user's data by performing the packet capture tool.

The data transmission between the applications, there are some problems such as Broadcast Theft, Activity Hijacking, Service Hijacking, Broadcast Injection, and the high proportion of application has these security problems [11]. Data transmission between the components is mean that android using intent to translate data between service, activity and Broadcast receiver, however, there are two types of the intent: explicit intents and implicit intents, the main difference is whether specified the component when communicating, if it didn't specify component, then the malicious application can Intercept the data when the component translating the data, this situation sometimes be happening by immature developer.

3. INTEGRATED ENVIRONMENT FOR TESTING MOBILE SECURITY

We considerate all the security issues in the previous section, we assume that the developer or mobile device researcher, when they face the any kind of security problems in the previous section, what kind of tools he would need. For example, a user wants to analysis a malicious application, he will need the dynamic analysis tools, static analysis tools, wireless analysis tool, reversing tool, so based on this ideal and then we propose an integrated environment for testing mobile security.

We divide the tools into six separate groups as follows: Development tool, Forensics tool, Penetration testing tool, reversing tool, Wireless analysis tool, Malware analysis tool. In the next section we will illustrate above toolkit.

3.1 Development tool

In the Development toolkit, in order to build a based development environment for researchers and developers, we collected development tool which is only relevant Android development environment, because Apple's restrictions for the development of the environment, lead to the development toolkit exclude the apple's integrated development environment, "Xcode", so the development toolkit only including the Android development tools. The tools are as follows:

1) Java Development Kit (JDK): JDK was developed by Sun Microsystems, is used to develop Android, Java programming

environment essential.

2) Android SDK manager: *The Android SDK separate tools, platforms, and other components into packages let users download using the SDK Manager.*
3) Android NDK: *NDK is a toolset that allows users to implement parts of the user app using native-code languages such as C and C++.*
4) Eclipse: *Eclipse is an integrated development environment (IDE). It contains a base workspace and an extensible plug-in system for customizing the environment.*

3.2 Forensics tool

The forensic tools are collected due mainly to the data storage security issue. Nowadays, a lot of sensitive information exists in the mobile device such as: email account and password, community website account and password, bank account information, GPS information, personal health information, and these files are stored in the mobile device's internal storage, external storage or volatile memory [8]. Once the data was not encrypted or adopt other protection, it is possible to fetch the sensitive personal information out through the forensic tools. We collect the tools as follows:

1) Aflogical-ose: AFLogical OSE was released in December 2011 and is now hosted on GitHub. The app provides a basic framework for extracting data from Android devices using Content Providers and then saves the data to the SD Card.
2) Libimobiledevice: *Libimobiledevice is a cross-platform software library that talks the protocols to support iPod, iPhone, Touch, iPad and Apple TV devices.*
3) Scalpel: *The Scalpel is a file carving and indexing application that runs on Linux and Windows.*
4) Ipba2: *The IPhone Backup Analyzer is a utility designed to easily browse through the backup folder of an iPhone (or any other IOS device). Read configuration files, browse archives, lurk into databases.*
5) SQLiteSPY: *SQLiteSpy is a fast and compact GUI database manager for SQLite. Its graphical user interface makes it very easy to explore, analyze, and manipulate databases.*

3.3 Penetration testing tool

The main purpose of penetration testing is hoped that through internal or external penetration testing, thereby to assess the safety of the operating system or applications, and as a developer it is necessary to understand the risk of their software. On the other hand, researchers can understand the security problem through penetration testing on mobile devices. The purpose of collecting penetration testing tools in our study, so we want to collect the penetration testing tools which associated with mobile devices. We collect the tools as follows:

1) Nmap: Nmap is a free and open source utility for network discovery and security auditing.
2) SQLmap: *SQLmap is an open source penetration testing tool that automates the process of detecting and exploiting SQL injection flaws and taking over of database servers.*
3) SSLstrip: *SSLstrip is a MITM tool that implements Moxie Marlinspike's SSL stripping attacks.*
4) W3af: *w3af is a Web Application Attack and Audit Framework. The goal is to help the user secure their web.*
5) ZAP: *ZAP is an easy to use integrated penetration testing tool for finding vulnerabilities in web applications.*
6) Ettercap: *Ettercap is a comprehensive suite for man in the middle attacks. It features sniffing of live connections, content filtering on the fly and many other interesting tricks.*

3.4 Reversing tool

Decompile applications could let researchers and developer to know the actual behavior of the application, whether the researchers or developers are need the reversing tool to analysis application or malware; Therefor, we have collected a number of commonly used reversing tools in our toolkit, all tools are as follows:

1) Apkinspector: The goal of apkinspector is to aid analysts and reverse engineers to visualize compiled Android packages and their corresponding DEX code.
2) Apktool: *Apktool is for reverse engineering 3rd party, closed, binary Android apps. It can decode resources to nearly original form and rebuild them after making some modifications.*
3) AntiLVL: *The AntiLVL's purpose is to subvert standard license protection methods such as the Android License Verification Library, Amazon App store DRM.*
4) JD-GUI: *JD-GUI is a standalone graphical utility that displays Java source code of ".class" files.*
5) Dex2jar: *Dex2jar is a tool for transforming ".dex" into ".jar".*
6) Radare2: *Radare2 is a rewrite from scratch of radare in order to provide a set of libraries and tools to work with binary files.*

3.5 Wireless analysis tool

Part of the data transmission security is the data transmission via the wireless, It relates to the packet encryption and decryption on the internet or Secure communication protocol, so capture the packet on the internet and then further analysis it, that is must be done if developers or researchers want to check the security about the applications via the network, so we provide multiple wireless network analysis tool for the user. All tools are as follows:

1) Aircrack: Aircrack-ng is an 802.11 WEP and WPA-PSK keys cracking program that can recover keys once enough data packets have been captured.
2) Kismet: *Kismet is an 802.11 layer2 wireless network detector, sniffer, and intrusion detection system.*
3) Wifite: *Wifite is a tool for attacking multiple WEP, WPA, and WPS encrypted networks in a row. This tool is customizable to be automated with only a few arguments.*
4) Mitmproxy: *An interactive console program that allows traffic flows to be intercepted, inspected, modified and replayed.*

3.6 Malware analysis tool

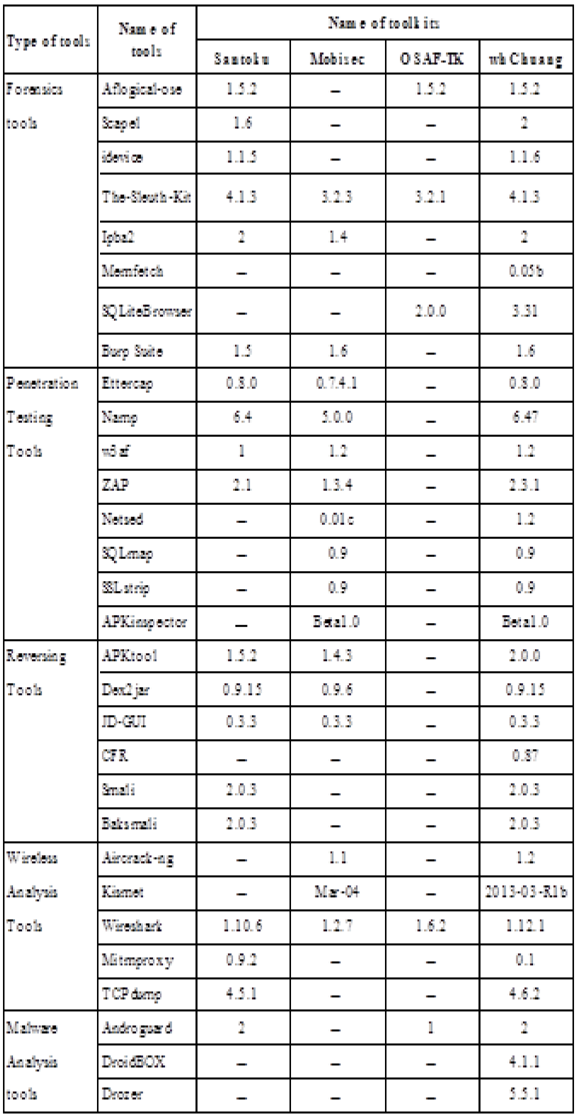
Mobile malware has been growing in scale and complexity as mobile device usage continues to rise. In [12] it is shown that 90% of the samples were compromised and turn into the bot; 45.3% try to misuse SMS or call services to obtain financial profit; and 51.1% gain user information. There are many different types of malware such as premium service abuser, adware, information theft, malicious Downloader, remote control, these malware will let users lost the money and privacy was being stolen, so we offer the number of tools to analyze the malware and malware samples. All tools and samples we are collected as follows:

1) Androguard: Reverse engineering, Malware and good ware analysis of Android applications.
2) DroidBox: *DroidBox is developed to offer dynamic analysis of Android applications.*
3) Drozer: *Drozer is a comprehensive security audit and attack framework for Android.*
4) Droidkungfu: *It is a malware sample, the main behavior was leaking information, IMEI, IMSI and location.*

4. VERSUIB OF THE TOOLS

Mobile Version of the tools device security related tools not only have the many kinds of tool and the updated version is also releasing very fast. Therefore, we provided the tools which collected to the latest version, in our testing mobile security environment, existing tools comparison chart in the table I.

TABLE I. Comparisons between other tools with ours



Type of tools	Name of tools	Name of toolkits			
		Santoku	Mobisec	OSAF-TK	wh Chuang
Forensics tools	Aflogical-ose	1.5.2	–	1.5.2	1.5.2
	Scapel	1.6	–	–	2
	idevice	1.1.5	–	–	1.1.6
	The-Sleuth-Kit	4.1.3	3.2.3	3.2.1	4.1.3
	Ipba2	2	1.4	–	2
	Memfetch	–	–	–	0.05b
	SQLiteBrowser	–	–	2.0.0	3.31
	Burp Suite	1.5	1.6	–	1.6
Penetration Testing Tools	Ettercap	0.8.0	0.7.4.1	–	0.8.0
	Namp	6.4	5.0.0	–	6.47
	w3af	1	1.2	–	1.2
	ZAP	2.1	1.3.4	–	2.3.1
	Netsed	–	0.01c	–	1.2
	SQLmap	–	0.9	–	0.9
	SSLstrip	–	0.9	–	0.9
	APKinspector	–	Beta1.0	–	Beta1.0
Reversing Tools	APKtool	1.5.2	1.4.3	–	2.0.0
	Dex2jar	0.9.15	0.9.6	–	0.9.15
	JD-GUI	0.3.3	0.3.3	–	0.3.3
	CFR	–	–	–	0.87
	Smali	2.0.3	–	–	2.0.3
	Baksmali	2.0.3	–	–	2.0.3
Wireless Analysis Tools	Aircrack-ng	–	1.1	–	1.2
	Kismet	–	Mar-04	–	2013-03-R1b
	Wireshark	1.10.6	1.2.7	1.6.2	1.12.1
	Mitmproxy	0.9.2	–	–	0.1
	TCPdump	4.5.1	–	–	4.6.2
Malware Analysis tools	Androguard	2	–	1	2
	DroidBOX	–	–	–	4.1.1
	Drozer	–	–	–	5.5.1

5. LIVE DVD/USB

A live DVD/USB is a complete, bootable computer installation including operating system that can be booted, live DVD/USB are closely related to live CD, but sometimes have the ability to persistently save settings and permanently install software packages back onto the USB device. There are a variety of different applications on the Internet such as, CAINE, it offers a complete forensic environment that is organized to integrate existing software tools as software modules and to provide a friendly graphical interface; Backtrack, that focused on security based on the Ubuntu Linux distribution aimed at digital forensics and penetration testing use. Such Live DVD / USB's purpose is looking to integrate domain-specific tool kit for the user to quickly configure the necessary environment.

6. DESIGN AND IMPLEMENTATION

6.1 System architecture

In this research, we developed the system based on the operating system of Ubuntu 14.4 LTS, Ubuntu 14.4 is the latest version of the Ubuntu, which is more powerful and smooth, and we choose it as our operating system, shown in Figure 1.

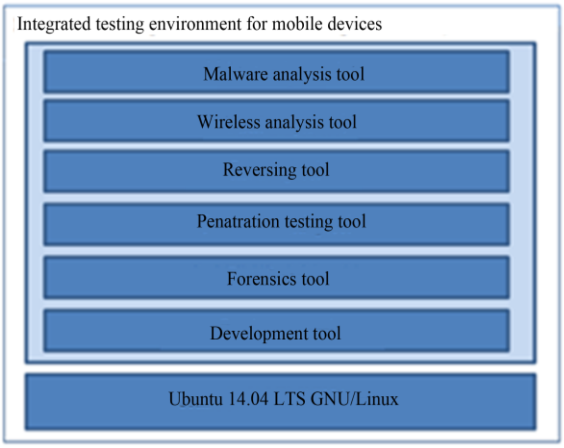
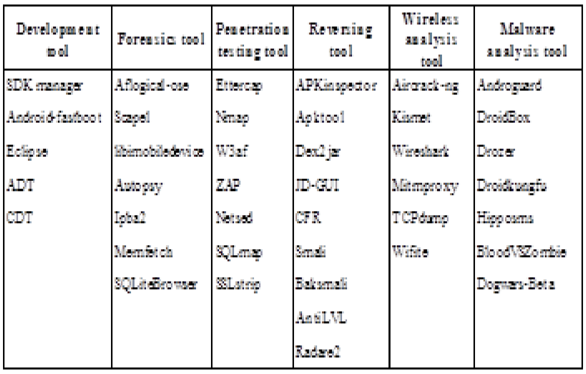


Fig. 1. System architecture.

In addition, as mentioned previously, we divide the tools into six separate groups, and which tool we collected was shown in Table II.

TABLE II. Security tools kit



Development tool	Forensics tool	Penetration testing tool	Reversing tool	Wireless analysis tool	Malware analysis tool
SDK manager	Aflogical-ose	Ettercap	APKinspector	Aircrack-ng	Androguard
Android-fastboot	Scapel	Nmap	Apktool	Kismet	DroidBox
Eclipse	libimobiledevice	W3af	Dex2jar	Wireshark	Drozer
ADT	Autopsy	ZAP	JD-GUI	Mitmproxy	Droidkungfu
CDT	Ipba2	Netsed	CFR	TCPdump	Hippoerns
	Memfetch	SQLmap	Smali	Wifite	BloodVSZombie
	SQLiteBrowser	SSLstrip	Baksmali		Dogwars-Beta
			AntiLVL		
			Radare2		

6.2 System flow

In this study, we want to offer a system which is convenient and easy to configure for developers or researchers, as shown in Figure 2. Fist, users have to download our system, second, set up it by virtual machine or create it to the bootable DVD/USB, and you will figure out that all related tools are installed in the system, what the user need to be done is turn on the computer by live DVD/USB, then all the settings will be set up, then users could just use it.

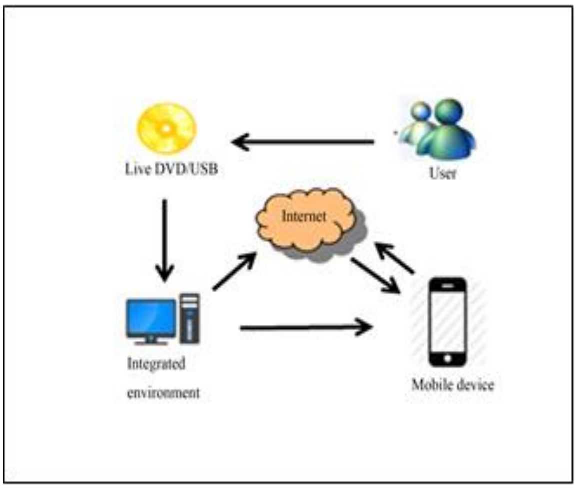


Fig. 2. System architecture.

6.3 Chinese user interface

In addition, the other thing we have done, we translate the system's user interface into Chinese, because we not only hope to make it unnecessary for users to collect tools and installing tools, but also let the system be felt friendly when user using it.

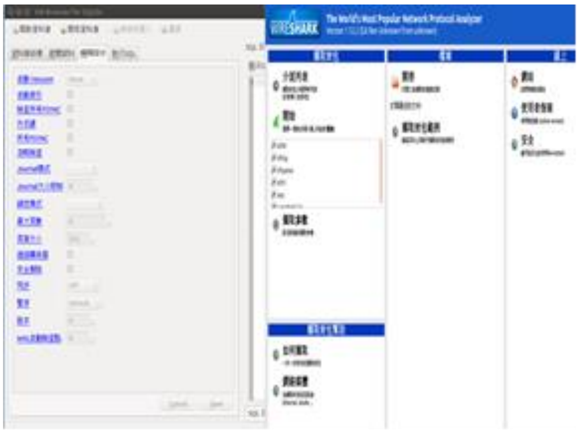


Fig. 3. Chinese user interface

7. CONCLUSIONS

In this study. First, we have reviewed the literature based on the related terms needed in the research; Second, according to the literature review we have divided the tools into six separate groups as follows: Development tool, Forensics tool, Penetration testing tool, Reversing tool, Wireless analysis tool, Malware analysis tool; Third, we have collected a lot of well-known mobile security tools, not only installed them, but also transformed the English user interface into Chinese; Finally, we have setup the whole system into Live DVD or Live USB based on Ubuntu 14.04 LTS to become an integrated environment for testing mobile security.

The results of the implementation, we offer the developer or researcher an integrated environment, which make it unnecessary for users to collect tools and install tools, so it reduces time-consuming when user configure the system environment, and provide a user-friendly and easy to use implementation of testing mobile security.

ACKNOWLEDGEMENT

This work was supported by National Science Council grant (NSC 102-2221-E-017-003-MY3).

REFERENCES

[1] eMarketer, Smartphone users worldwide will total 1.75 billion in 2014. [Online]. Available: http://www.emarketer.com/Article/Smartphone-Users-Worldwide-Will-Total-175-Billion-2014/1010536

[2] G. Prabhu., IOS 7.0.2 bug lets anyone bypass lock screen passcode to access Phone app. [Online]. Available: http://www.iphonehacks.com/2013/09/ios-7-0-2-bug-bypass-lock-screen-passcode-access-phone-app.html

[3] J. Forristal, Android Master Key Exploit – Uncovering Android Master Key That Makes 99% of Devices Vulnerable. [Online]. Available: https://bluebox.com/technical/uncovering-android-master-key-that-makes-99-of-devices-vulnerable/

[4] G. Suarez-Tangil, J.E. Tapiador, P. Peris-Lopez, and Ribagorda, Arturo. "Evolution, detection and analysis of malware for smart devices," IEEE Commun. Surveys & Tutorials, vol. 16, pp. 961 – 987, November 2013.

[5] A.P. Felt, E. Chin, S. Hanna, D. Song, and D. Wagner, "Android permissions demystified," in Proc.18th ACM Conf. on Computer and communications security. New York, NY, USA: ACM, October 2011, pp. 627–638.

[6] A.P. Felt, K. Greenwood, and D. Wagner, "The effectiveness of application permissions," in WebApps'11 Proceedings of the 2nd USENIX Conf. on Web application development, Portland, OR, USA, June 2011, pp. 7–7.

[7] T. Oh, B. Stackpole, E. Cummins, C. Gonzalez, R. Ramachandran, and Shinyoung Lim, "Best security practices for android, blackberry, and iOS," in Proc. First IEEE workshop on Enabling technologies for smartphone and internet of things, Seoul, June 2012, pp. 42- 47.

[8] M.S. Ahmad, N.E. Musa, R. Nadarajah, R. Hassan, and N.E. Othman, "Comparison between android and iOS operating system in terms of security," in 8th Int. Con. on Information technology in Asia, Kota Samarahan, July 2013, pp. 1-4.

[9] C. Ntantogian, D. Apostolopoulos, G. Marinakis, and C. Xenakis, "Evaluating the privacy of Android mobile applications under forensic analysis," Computer & Security, vol. 42, pp. 66-76, January 2014,.

[10] R. Hunt, "Security testing in android networks – a practical case study," in 19th IEEE Int. Conf. on Networks, Singapore, December 2013, pp.1-6.

[11] E. Chin, AP. Felt, K. Greenwood, and D. Wagner, "Analyzing inter-application communication in android," in Proc. 9th Int. Conf. on Mobile systems, applications, and services. New York, NY, USA: ACM, June 2011, pp. 239–252.

[12] Y. Zhou and X. Jiang, "Dissecting android malware: characterization and evolution," in Proc. 33rd IEEE Symp. Security and Privacy, San Francisco, CA, May 2012, pp. 95-109.

[13] J. Hubbard, K. Weimer, and Yu Chen, "A study of SSL attacks on Android and iOS mobile Application," in Proc. IEEE Int. Conf. on Consumer communications and networking Conf., Las Vegas, January 2014, pp.86-91.

[14] A. Gupta, Learning Pentesting for Android Devices, Packt Publishing-ebooks Account, 2014.

[15] J. Six, Application Security for the Android Platform, O'reilly & Associates Inc., 2011.

[16] Google. Open Source database of Android Malware. [Online]. Available: https://code.google.com/p/androguard/wiki/DatabaseAndroidMalwares

[17] TrendLabs, The invisible becomes visible: Trend micro security predictions for 2015 and beyond. [Online]. Available: http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/reports/rpt-the-invisible-becomes-visible.pdf

[18] F. Stahl, and J. Ströher, Security Testing Guidelines for mobile Apps. [Online]. Available: https://www.owasp.org/images/0/04/Security_Testing_Guidelines_for_mobile_Apps_-_Florian_Stahl%2BJohannes_Stroeher.pdf