Forensics analysis of wi-fi communication traces in mobile devices

Amundsen, Anja; Ovens, Kenneth

Published in:
International Workshop on Big Data Analytic for Cyber Crime Investigation and Prevention

DOI:
10.1109/BigData.2017.8258357

Publication date:
2018

Document Version
Peer reviewed version

Forensics Analysis of Wi-Fi Communication Traces in Mobile Devices

Anja Evelyn Amundsen
Criminal Investigation Department
NCIS Norway
Oslo, Norway
Anja.Evelyn.Amundsen@politiet.no

Kenneth M. Ovens
Computer, Communications, and Interactive Systems
Glasgow Caledonian University
Glasgow, United Kingdom
Kenneth.Ovens@gcu.ac.uk

***Abstract*—Smart mobile devices have become an essential part of our lives, both professionally and privately. As we expand our digital presence and usage of these devices, we also increase the amount of evidence left behind. Locard's principle states that every contact leaves a trace, a principle applicable not only to traditional crime but also the cyber realm. This project addresses an area in which this principle is applied, in terms of wireless networks and their access points that have been identified through the wireless scanning process. Both access points and smart devices use different, yet similar, network frames to advertise their existence before a connection can be established between two devices. These frames contain the data that this project aims to locate, MAC addresses, which are uniquely identifiable data, that can be used to identify any device that holds a wireless interface. By analysing the mobile device of a user after a short, city centre walk, the MAC addresses of nearby wireless access points were identified and used to retrace the route travelled by the user. However, as the information was only found in volatile memory, there was a limited timeframe to capture this data.**

Keywords-Digital Forensics; Android; Volatile Memory; Wireless Communication; MAC Address

I. INTRODUCTION

Technology has become an inescapable part of our lives. In particular, mobile devices have seen an exponential increase over the past decade, exceeding a regular computer in terms of the user specific information they contain. With this development, law enforcement has recognised the potential evidentiary value that mobile devices possess, but are struggling to match the pace in which mobile devices are being developed. This includes the development of forensic tools that are available for practitioners.

Mobile devices require an operating system (OS) to function. Google´s open-source Android OS currently holds 87.6% of the total mobile device market share [1]. To benefit from all the features of a modern mobile device, users need to be connected to the Internet.

Inherently, with the growing number of mobile devices, the need to access wireless networks is growing. Individuals often want their device to connect to the Internet through known access points (AP) as soon as they are within reach. This results in the wireless scanners on mobile devices always, or regularly, scanning for APs, regardless of their location.

From a digital forensics perspective, data can be extracted and analysed from mobile devices, helping to create a timeline of events and actions, such that it can be used as evidence in a court of law [2]. Mobile forensics and digital forensics as a whole, are constantly playing catch up with the ever evolving and multiplying collective noun of technology. Possibly one of the cases in recent times with the greatest media coverage for mobile forensics was the locked Apple device belonging to one of the suspects of the attack in San Bernardino in December 2015 [3].

Edmond Locard, a pioneer within forensic science, stated that every contact leaves a trace [4]. This applies not only in the physical world, but also in the digital world where evidence is represented in bits and bytes [5].

Applying Locard's principle, this study aims to identify, extract and analyse artefacts left over from the frequent communications between mobile devices and nearby wireless APs. This information will then be used to reconstruct a timeline of events that could be used to aid an investigation.

The remainder of this paper is divided into the following sections: Section II addresses and presents the Android OS, both its inner workings and how it communicates with wireless networks and related work. Section III elaborates on the experiments created for the project, including the devices utilised. Section IV presents the findings from the applied methodology, with a discussion regarding their presence and their evidentiary value in a criminal investigation. Lastly, Section V concludes the project, with an elaboration on further research derived from this project.

II. BACKGROUND AND RELATED WORK

A. Wireless Networks

Wireless communication has, for the past two decades, been under the 802.11 standard. APs and routers are two terms that often are used interchangeably, regardless of their distinct differences [6]. A router can route traffic it is sent and be an AP, whilst an AP only extends the range of the wireless networks. In this paper, the term AP includes both routers and APs.

As with all digital devices that communicate through the 802.11 standard, they are required to have a unique Media Address Control (MAC) which they can be identified by [7]. It is a 48-bit sequence of hexadecimal characters in the following format ff:ff:ff:aa:aa:aa. The three first hexadecimal sets are used to identify the manufacturer and the remaining three are used to identify the specific device.

Wireless APs need methods of either informing about their presence or finding other devices. This is done through what is known as *active* and *passive* discovery [8], by advertisement of their existence, which includes sharing their MAC addresses. Active discovery consists of the mobile device searching for APs, sending out probe request with the Service Set Identifier (SSID) of previously connected APs and hence wishes to connect [8]. Should an AP match the SSID to which the mobile device wishes to connect to, it would reply with a probe response. In passive discovery, it is the AP that sends out frames called beacons containing information about itself [9].

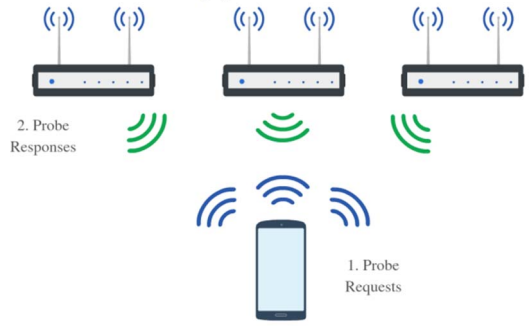

Figure 1. Active discovery [10]

B. Android OS

The Android OS facilitates a freely open and accessible OS, which in turn manufacturers and users can make their own. When it was developed by the Open Handset Alliance in 2007, there were three functional requirements that the software needed to provide; methods of connecting to the Internet, methods of accessing and downloading applications and methods of storing data on the device itself [11].

There are two types of memory one will often find in smart devices, volatile and non-volatile, both of which can have value in a criminal investigation. Through this research it became evident that non-volatile data has been subject to a greater deal of research than its sibling, volatile data, in terms of both recovery and its application in criminal investigations. This may be due to the nature of the data having little to no application in a court of law, as it has often been lost or obsolete before an investigator could access it.

Android mobile devices have an Application Programming Interface (API) identified as *android.net.wifi*, which is the location where the rules for how the wireless network entity functions [12]. This also addresses the time intervals in which a devices searches for wireless networks, as per the active discovery method. Specifically, for the device used in this research, the time interval between scans was 15 seconds.

An Android device can be rooted by a user to allow for administrative privileges, which enables the instalment of applications not supported by Google Play. This is not the only result of the rooting process, as it also allows an investigator to access data that they would otherwise have difficulty obtaining, such as volatile memory. There are a variety of tools available for this process such as King Root Android and One-Click Root, which manipulates a manufacture specific software weakness in the OS [13]. An issue arises when a device that is not rooted is obtained at a crime scene, as the act of rooting a device could directly affect the device and the data within. As there are no methods of providing integrity controls pre- and post-rooting.

C. Related Work

Mobile devices can contain a plethora of evidence that may be of pertinent importance in an investigation. There is a range of tools available to source predetermined information from a mobile device. However, from the AP evidentiary aspect of mobile devices, the only research located was by Andriotis et al. [14] and Minnaard´s contribution in terms of AP memory [15], reversing the process and analysing APs.

Andriotis et al. [14] produced findings that confirmed that there were traces of communication between devices, both when using 802.11 and Bluetooth technologies. Locations were specified within the filesystems of the devices they performed experiments on, however they early identified that some of the locations were device specific to manufacturers. In particular for this research a file named *cache.wifi* in the */data/com.google.android.location/files* was of interest, as it contained details of AP MAC addresses of and timestamps.

The study performed by Minnaard [15] addressed the opposing angle to this project; traces left on APs. As with Andriotis' study, the data was located in buffers (volatile memory), hence subject to potentially frequent change, depending on the device at hand. In addition to this, the application of Locard's principle can be derived from Minnaard's research, as any devices within a crime scene that supports, may store traces to help determine who, where and when, without consideration of the role of the device.

In terms of Locards principle, it has been subject to discussion whether the term 'contact' is fitting in the physical world [16]. This is due to there may not be any contact between the receiver and the source. A term they coined and believed to be better was 'transference', as there need not be contact between two entities for one of them to leave or receive a trace. This is also applicable in this research, as the smart device in question does not connect with APs, but rather there is a transference of unique identifiable data, in which they can later use to connect.

To determine the location of an AP there are a range of freely available network mapper tools that can be used to establish their location, using either SSID or MAC. The most prominent one found in research was *WiGle* [14, 17] which holds at the time of writing, information on 374,956,858 APs. A less known tool is the *LocationAPI* provided by Unwired Labs. At the time of writing it holds 1.5 billion entries [18] which is far greater than WiGle. Increasing the

possibility that a particular AP is stored in the LocationAPI tool with 356% over WiGle.

III. METHODOLOGY

The work conducted in this project employed an experimental approach, with two independent experiments; one laboratory and one field-based experiment. The experiments where inherently similar, with the difference of the device being exposed to the real world, in the field experiment, hence subject to many unknown factors. This is the advantage of such an experiment, its application in the real world, however creating a fundamental idea of how technology works in a strict, controlled laboratory environment is also important.

When addressing digital evidence and technology, there are certain guidelines one may choose to follow or must adhere to; in the United Kingdom, there are four principles known as The Association of Chief Police Officers (ACPO) guidelines [19]. These principles are simplified as, no data should be changed that can be relied upon in court. The examiner needs to be competent when accessing original data, to justify decisions. An audit trail of all actions needs to be documented and lastly there has to be a lead investigator with overall responsibility for adhering to both law and guidelines. These were applied and followed during the experiments conducted in this research.

As with any forensics investigation it is key that an investigator is methodical and thorough when addressing a mobile device. To adhere to these two aspects, specific methods and methodologies are selected. Within the forensics environment there are a range of methodologies, ranging from being very open-ended with phases in the process whilst some may give the investigator specific tasks all the way through. It is, however, important to acknowledge that they all cover the same core topics [20].

In this research, the general digital forensics model is used. The advantage of applying the general forensic process model is simplicity [21]. By making it an uncomplicated process, in which each phase builds upon the previous, without the deeply detailed coverage, which would differ from device to device, each aspect of the investigation is addressed. There are four steps to this methodology: collection of media, examination of data, analysis of information and presentation of results.

Many tools exist to help complete digital investigations. These can be both open source tools and commercially available tools. However, these are often based on finding artefacts that one knows exists [22]. Resulting in a magnitude of evidence, both exculpatory and incriminating, being missed and possibly leading to an incorrect conviction [23]. As a defence for any legal practitioner, there are not enough hours in the day to manually sift through the ever-growing amount of data that one may be given in a case, hence why one will utilise tools that may not portray all essential data.

The device used as a digital forensics workstation in the experiments was an Asus laptop running the OS version 64-bit Kali Linux 2012.2. This OS allowed the device to be forensically clean upon conducting the collection, extraction and analysis of each experiment.

Android Debugger Tool (ADB) facilitates communicate with an Android device through a shell, which accommodates the use of certain commands that can be used to create images of mobile devices, either fully or partially [24]. In this research it was used to acquire the state of the memory.

The Android mobile device used for this research is presented in Table I. As the tool selected for the collection of media is an Android specific tool, namely ADB, this methodology can only be applied to a device running the Android operating system. Hence excluding Windows and Apple devices from this research. This device was not connected to a mobile network. In both the experiments there were tasks that needed to be completed prior to the generation of data, followed by the general digital forensics methodology.

1) The Android mobile device was reset to factory settings.
2) Connected to the Internet to allow the device to perform updates.
3) Developer Options in settings was enabled and the setting USB debugging was permitted [25].
4) The device was disconnected from the network and the network was forgotten, in such that it would not connect to it again, during the experiment.

TABLE I. DEVICE SPECIFICATION

Feature	Motorola moto E
Operating System	Android 4.4.4
Memory	1GB
MAC Address	80:6c:1b:72:c4:9c

A. Laboratory-based Data Generation

Being able to create an understanding for where potential data regarding APs were stored, the length in which the data was stored and what data was stored, was the at the pinnacle of this research project. To enable such an understanding, a laboratory-based experiment with limited outside interference was conducted. This included the usage of a Linksys router in which the MAC address was known.

In this scenario, the mobile device was allowed to search for wireless networks in time intervals of 10, 20 and 30 minutes. As previously identified, the mobile device searches every 15 seconds for APs. This would further mean that for ten minutes it would have completed 40 scans; for 20 minutes, it would have completed 80 scans; and for 30 minutes, 120 scans would have been completed. Thus, rendering a sufficient amount of data generated for scanning and the potential length of time it is stored there.

Additionally, the *Aeroplane mode* was identified to potentially have an effect on the data stored on the device, as enabling it would stop further searches of wireless networks, and potentially therefore avoid the deletion of prior scans as the memory of the device was limited. Hence a scan of 15 minutes was completed, in which the Aeroplane mode was

TABLE II. STRINGS SEARCHED

Strings	Reasoning
ssid	Often used as the prefix before presenting an APs name.
wifi	Identifies any occurrence of the Android device enabling or using wireless features.
mac	Often used as the prefix before presenting an APs unique identity.
([0-9A-Fa-f]{2}[:-]){5}([0-9A-Fa-f]{2})	Regular expression that matches the pattern of a MAC address.

enabled and after 5 minutes the device was subject to the same methodology as in the collection, extraction and analysis section.

B. Field-based Data Generation

To compare the results from the laboratory experiments with the field-based experiments, a route was determined in a highly populated area, both in terms of people and businesses. At certain points in the route the time of passing was manually documented in a logbook. This was to enable the comparison of timestamps potentially found on the device, with those manually confirmed via the log.

Two scenarios were established and completed for data generation for the field experiment. One where the mobile device remained in the pocket of the researcher and the other where the mobile device was used to record the time of passing at specific points on a predetermined route. In both scenarios, the wireless search function was enabled at the start of the route. This was based on the fact that many individuals use their mobile device whilst walking, either for communication, entertainment or simply direction guidance, and that this may have had an impact on data generated. The predetermined route took approximately 15 minutes to walk, the distance being 1.4 kilometres, from start to finish, from Glasgow Central station to the corner of Sauchiehall Street and West Nile Street.

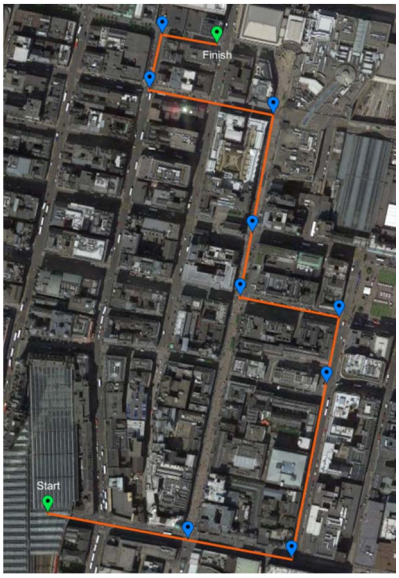

Figure 2. Established route

C. Collection, Extraction and Analysis

Although the methods of data generation differed, the process of media collection, data extraction and information analysis were similar. The files that were extracted from the mobile device, were labelled with the date and time of which the collection and specific task was initiated.

1) The mobile device and forensic workstation were connected through a USB cable to allow communication and the following command was used to confirm connection,
```
adb devices
```
2) The memory data of the device was collected,
```
adb shell dumpstate >
location/filename.txt
```
3) To allow for evidence continuity and integrity a SHA256 hash value was calculated for each memory extractions,
```
sha256sum location/filename.txt >>
hash_values.txt
```
4) Having completed the media collection, the phase of data examination was reached and points of potential interest extracted. This was completed using the *strings* command, piping it into a *grep* search, as identified in Table II. The command syntax used for the strings search was:
```
strings filename.txt | grep –i
‘searchstring’ >
filename_searchstring.txt
```
5) For the regular expression search the following syntax was used,
```
strings filename.txt | grep –E ‘RegEx’
> filename_searchstring.txt
```
6) These search files were also subjected to a SHA256 hash value calculation for evidence continuity.

Having sifted the information, it was then manually analysed to establish if the different experiments were rendered fruitful in terms of the evidence found.

IV. RESULTS

The two experiments that were conducted were independent of each other, however, combined they gave an understanding and potential application in a real-world scenario.

A. Laboratory Results

Data containing information about APs was discovered in the volatile memory of the mobile device. However, no traces of any Wi-Fi communication were located in non-volatile memory.

The search results found scans entitled ‘*WIFI_SCAN_RESULTS’* that included the date and time in which the scan started. Following this, each AP that the mobile device had communicated with were listed, including the SSID and MAC address. Additionally, it was found that

these entries appeared every 15 seconds, which confirmed that the scans did indeed occur in the specified time interval in the laboratory environment.

The information of greatest value was found in the *mac* entry for each of the identified APs in the memory of the device. Upon initial inspection, it was apparent that if this was the MAC address of an AP, it had been obfuscated as it was not presented in its normal hexadecimal format. The string found within the memory was determined to be decimal, as when converted to hexadecimal it represented the MAC address format. This was confirmed by comparing a known MAC address to its converted decimal to hexadecimal value.

A further issue that was observed upon conversion was the absence of a hexadecimal value in the string, as there was only five, not the required six values. When the value is stored in memory and the value is converted to decimal, if there are two leading zeros in the MAC address, these two are suppressed. This only occurred when they were leading, and not when there were two zeros in the middle of the address. An example would be the MAC address A2:EE:A6:9C:F1, which is missing a hexadecimal value, when converted to its decimal value 699788598513. But by adding the two leading zeros the issue was mitigated, and converting the value 00:A2:EE:A6:9C:F1 gave the same decimal value as in the memory.

The regular expression search yielded results that the other string searches did not find; an entry in the memory which stored SSIDs and MAC addresses in clear text and original format. This was found under an identifying process number, which varied from collection to collection. A theory is that this is the last scan which is presented to the users when they choose which network they wish to connect to, however, this could not be confirmed.

All the data located was of the volatile type, which is sensitive to change, through the effects of time or device usage. This included the scan entries in the memory, as after 10 to 11 minutes the last scan was overwritten and lost. Additionally, the experiment determined, as expected, that data was lost when the device was powered down.

The aeroplane mode had no effect on the length in which the data was stored in the memory of the device. Even though all communication channels were disabled, there are still processes running on the device that require memory and resulted in the entries being lost.

During one of the experiments, the mobile device had not been disconnected from the Internet, having concluded the preparatory phase. This raised the question if there would potentially be more or other pieces of evidence regarding location, having been connected during active discovery. Therefore, the media was extracted and analysed per the aforementioned methodology.

Within the memory dump, two entries named, *wifiResult=WifiLocatorResult* were found within the results of two AP searches. The time difference between the two did not establish any identifiable regularity in which they would appear. These two entries included all the APs seen in the scan and included what appeared to be latitude and longitude coordinates. The application of these two values into Google Maps displayed that the coordinates were within close proximity to where the experiments were taking place. It is unknown how the coordinates were determined, other than the device itself using its Global Positioning System (GPS) to establish these.

B. *Field Results*

There were two different scenarios created for this experiment. One where the phone was left in a pocket and not interacted with during the route, and the other where the device was used to determine the time of passing specific locations. These two scenarios yielded different results.

For the *no-interaction* scenario, no significant data was discovered from the device when it was analysed upon completion of the route. This revealed that the device required some form of interaction to initiate the scans.

When the device was utilised to determine the time of passing, there was evidence of wireless network scans in the memory. By accessing the screen to determine the time of passing, the device appeared to initiate a scan that was equal to the time of passing. These were not split by 15 second intervals as found in the laboratory experiment, but rather the time in which the screen was activated.

Part of this research was to determine if the data gathered could be used to track the movements of a device, through the collected MAC addresses. This was done through the LocationAPI tool, where the MAC addresses were plotted on a map and was correlated with the predetermined route map. Through this, one could easily determine the route the device had taken, although the first 5 minutes of the route was missing from the map, as these scans had been lost due to the 10 – 11 minutes storage time of scans in the memory.

Figure 3. APs seen by the mobile device

V. CONCLUSION AND FUTURE WORK

From this research, we have established that under certain circumstances, evidence of Wi-Fi interactions can be located in the memory of Android mobile devices.

As identified by Minnaard [15], the inner workings of a device can inform an investigator about how it interacts with wireless networks, but that does not infer that evidence of communications are always present. This is clear in the experiments that were conducted, as data was present when certain variables were applied and absent when not. In addition, the length of time which it was stored comes under Minnaard's contribution, as it was only contained in memory for ten minutes until the first entry was overwritten, or the device was powered down. This is a very limited time frame for any investigator.

To access the memory of this Android device, it had to be rooted and this was done prior to the experiments. Rooting the device would result in the device being powered down and memory data would be lost. An investigator would therefore have to refer to the ACPO guidelines, providing justifications for their actions and acknowledging the potential consequences.

The evidence gathered from these experiments showed that there are alternative methods of establishing where a device has been, in contrast to traditional location data portrayed through GPS. An extended use of this data could be in conjunction with GPS data to corroborate the location of the device at a specific point in time.

In terms of utilisation within law enforcement, the findings of this research are limited due to the length of time the information was stored. An area for future research is to determine if other applications on the device utilise the Wi-Fi information and store it in non-volatile memory. This would allow investigators more time to capture this information and recreate a timeline of user movements.

Further future research should encompass other devices and other OSs, such as iOS and Windows, in such a way that a further understanding and potential application of techniques to access data for use in law enforcement can be determined.

REFERENCES

[1] International Data Corporation, 2016. *Smartphone OS Market Share 2016*. International Data Corporation.

[2] Kohn, M., Eloff, M. & Eloff, J., 2013. Integrated digital forensic process model. *Computers & Security.* **38**, pp. 103-115.

[3] Cahyani, N. D. W., Rahman, N. H. A., Xu, Z., Glisson, W. B., Choo, K.-K. R., 2016. The role of mobile forensics in terrorism investigations involving the use of cloud apps. Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications. Xi'an, 2016. Brussels: Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering, pp. 199-204.

[4] Burkey, C. R., Bensel, T. t. & Walker, J. T., 2014. Forensic Investigation of Sex Crimes and Sexual Offenders. New York: Elsevier Science

[5] Bouhaidar, R., 2005. Forensic webwatch: Forensic computing. Journal of Clinical Forensic Medicine. **12**(1), pp.47-49.

[6] Wirelesshack, 2016. *Wireless Access Point VS Router*. WirelessHack.

[7] Abedi, N., Bhaskar, A. & Chung, E., 2013. Bluetooth and Wi-Fi MAC address based crowd data collection and monitoring: benefits, challenges and enhancement. *Australasian Transport Research Forum 2013 Proceedings.* Brisbane, 2013. Brisbane: Queensland University of Technology, pp. 1-17.

[8] Freudiger, J., 2015. How talkative is your mobile device? An experimental study of Wi-Fi probe requests. *Proceedings of the 8th ACM Conference on Security & Privacy in Wireless and Mobile Networks*. New York, 2015.

[9] Geier, J., 2002. *Understanding 802.11 Frame Types.* Wi-Fi Planet.

[10] Amundsen, A., 2017. *Active Discovery*.

[11] Hoog, A., 2011. *Android Forensics - Investigation, Analysis and Mobile Security for Google Android.* 1st ed. Waltham: Syngress.

[12] Shi, J., 2017. *How Android Wifi State Machine Works*. Pearls in Life.

[13] Vidas, T., Zhang, C. & Christin, N., 2011. Toward a general collection methodology for Android. *Digital Investigation*. **8**(Supplement), pp. S14-S24.

[14] Andriotis, P., Oikonomou, G. & Tryfonas, T., 2012. Forensic analysis of wireless networking evidence of android smartphones. *IEEE International Workshop on Information Forensics and Security.* Tenerife, 2012. Piscataway: IEEE, pp. 109-114.

[15] Minnaard, W., 2014. Out of sight, but not out of mind: Traces of nearby devices' wireless transmissions in volatile memory. *Digital Investigation*. **11**(Supplement), pp. S104- S111.

[16] Horswell, J. & Fowler, C., 2004. Associative evidence - the Locard exchange principle. In: *The Practice of Crime Scene Investigation.* Boca Raton: CRC Press, pp. 45-55.

[17] Fu, X., Zhang, N., Pingley, A., Yu, W., Wang, J., Zhao, W., 2012. The Digital Marauder's Map: A WiFi Forensic Positioning Tool. *IEEE Transactions on Mobile Computing*. **11**(3), pp. 377-389.

[18] Unwired Labs, 2017. *Cell Tower & WiFi Coverage.* Unwired Labs.

[19] Williams, J., 2012. *ACPO Good Practice Guide for Digital Evidence*. Metropolitan Police Service.

[20] Sammons, J., 2015. *The Basics of Digital Forensics.* 2nd ed. Waltham: Syngress.

[21] Reith, M., Carr, C. & Gunsch, G., 2002. An Examination of Digital Forensic Models. *International Journal of Digital Evidence* . **1**(3), pp. 1-12.

[22] Garfinkel, S.L., 2010. Digital forensics research: The next 10 years. *Digital Investigation*. **7**(Supplement), pp. S64-S73.

[23] Goodison, S.E., Davis, R.C. & Jackson, B.A., 2015. *Digital Evidence and the U.S. Criminal Justice System: Identifying Technology and Other Needs to More Effectively Acquire and Utilize Digital Evidence*. Santa Monica, CA, USA: RAND Corporation.

[24] Android Developer, 2014. *Android Debug Bridge*. Android Studio.

[25] Thomas, D., 2015. *How to Enable Developer Options & USB Debugging*. WonderHowTo.