SOCIAL MEDIA FORENSICS ON MOBILE DEVICES

Yalçın ÇAKMAK
112692030

İSTANBUL BİLGİ ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ
BİLİŞİM VE TEKNOLOJİ HUKUKU YÜKSEK LİSANS PROGRAMI

Yrd. Doç. Dr. Leyla KESER BERBER

2015

SOCIAL MEDIA FORENSICS ON MOBILE DEVICES

MOBİL CİHAZLARDA SOSYAL MEDYA ADLİ ANALİZİ

Yalçın ÇAKMAK
112692030

Yrd. Doç. Dr. Leyla KESER BERBER :
Yrd. Doç. Dr. Emin İslam TATLI :
Öğr. Görv. İbrahim Halil SARUHAN :

Tezin Onaylandığı Tarih :

Toplam Sayfa Sayısı : 81

Anahtar Kelimeler (Türkçe)	Anahtar Kelimeler (İngilizce)
1) Dijital adli analiz	1) Digital forensics
2) Mobil uygulama	2) Mobile application
3) Sosyal medya	3) Social media
4) Mobil adli analiz	4) Mobile forensics
5) Kanıt	5) Evidence

ABSTRACT

Social media platforms and smartphones have been inevitably effecting our lives. Many people prefer to use mobile applications for several reasons. However, they generally do not know the artifacts they left in smartphones. Due to an increasing frequency of encountering smartphones related to various crimes, the need for new approaches in mobile forensics area is increased. Investigating the artifacts of mobile applications may help to find various evidences.

This thesis provides an overview of files and directories created by popular social media applications. Two popular smartphones with six most used social media mobile applications were taken into consideration. Mobile applications were downloaded from official stores and example scenarios were prepared for each. Both logical and physical imaging of smartphones were performed. Low level modifications like rooting and jailbreaking were accomplished for physical imaging. Both commercial and open source tools were used for imaging and analysis.

The results of the research emphasizes that mobile social media applications generally create database files, log files, xml files and plist files to store most of the private and evidentiary data. The contents of these files can be obtained easily. Malware infection or device lost may cause the malicious usage of the private data. Many ethical concerns also come into question for mobile forensic investigators. They should pay attention to the confidentiality of private information irrelevant to the case.

ÖZET

Sosyal medya platformları ve akıllı telefonlar hayatımızı kaçınılmaz olarak etkilemektedir. Birçok kişi çeşitli nedenlerle mobil uygulamaları kullanmayı tercih etmektedir. Ancak kullanıcılar, akıllı telefonlarda bıraktıkları kişisel izleri genellikle bilmemektedirler. Çeşitli suçlarla alakalı akıllı telefonlarla gittikçe daha fazla karşılaşılması, mobil adli analiz alanında yeni yaklaşımlara ihtiyacı artırmaktadır. Mobil uygulamaların kişisel izlerini inceleme çeşitli kanıtları bulmaya yardımcı olabilir.

Bu tez popüler sosyal medya uygulamaları ile oluşturulan dosyalar ve dizinler için genel bir bakış açısı sağlamaktadır. İki adet popüler akıllı telefon ve altı adet çok kullanılan sosyal medya mobil uygulaması dikkate alınmıştır. Mobil uygulamalar resmi mağazalardan indirilmiş ve her birisi için örnek senaryo hazırlanmıştır. Cep telefonlarının hem mantıksal hem de fiziksel imaj alma işlemi gerçekleştirilmiştir. Fiziksel imaj almak için rooting ve jailbreaking gibi alt seviye değişiklikler yapılmıştır. İmaj alma ve inceleme için hem ücretli hem de açık kaynak kodlu programlar kullanılmıştır.

Araştırma sonuçları, mobil sosyal medya uygulamalarının birçok özel ve kanıt olabilecek bilgiyi saklamak için genellikle veri tabanı, log, xml ve plist dosyaları oluşturduğunu vurgulamaktadır. Bu dosyaların içerikleri kolaylıkla elde edilebilmektedir. Zararlı yazılım bulaşması ve cihazın kaybolması özel verilerin kötü niyetli kullanımına neden olabilir. Mobil adli analiz araştırmacıları için birçok etik kaygı da gündeme gelmektedir. Dava ile alakasız özel bilgilerin gizliliğine dikkat etmelidirler.

TABLE OF CONTENTS

ABBREVIATIONS

ADB	:	Android Debug Bridge
ANSI	:	American National Standards Institute
ASCII	:	American Standard Code for Information Interchange
CFTT	:	Computer Forensics Tool Testing
CPU	:	Central Processing Unit
DFU	:	Device Firmware Upgrade
ESN	:	Electronic Serial Numbers
FTK	:	Forensic Toolkit
GB	:	Gigabyte
GPS	:	Global Positioning System
ICCID	:	Integrated Circuit Card Identifier
IMEI	:	International Mobile Equipment Identity
IP	:	Internet Protocol
iOS	:	iPhone Operating System
JTAG	:	Joint Test Action Group
MMS	:	Multimedia Messaging Service
NIST	:	National Institute of Standards and Technology
PIN	:	Personal Identification Number
RAM	:	Random Access Memory
ROM	:	Read Only Memory
SDK	:	Software Development Kit
SIFT	:	SANS Investigative Forensic Toolkit
SIM	:	Subscriber Identification Module
SMS	:	Short Message Service
SSH	:	Secure Shell
UFED	:	Universal Forensic Extraction Device
URL		Uniform Resource Locator
USA	:	United States of America
USB	:	Universal Serial Bus

BIBLIOGRAPHY

Appple Inc. iOS Security. White Paper .https://www.apple.com/br/privacy/docs/iOS_ Security_Guide_Oct_2014.pdf; October 2014.

Ayers R, Brothers S, Jansen W. Guidelines on Mobile Device Forensics. NIST Special Publication 800-101 Revision 1; May 2014.

Bader M, Baggili I. iPhone 3GS forensics: logical analysis using apple itunes backup utility. Small Scale Digital Device Forensics Journal; September 2010.

Barmpatsalou K, Damopoulos D, Kambourakis G, Katos V. A critical review of 7 years of Mobile Device Forensics. Digital Investigation 10 (2013) 323–349.

Bassett R, Bass L, O'Brien P. Computer Forensics: An Essential Ingredient for Cyber Security. 2006; 3(1): 22-32.

Bommisetty S, Tamma R, Mahalik H. Practical Mobile Forensics. Packt Publishing; 2014.

Brothers, S. How Cell Phone "Forensic" Tools Actually Work - Cell Phone Tool Leveling System. DoD Cybercrime Conferece. 2011. Atlanta, GA.

Casey E, Turnbull B. Digital Evidence and Computer Crime: forensic science, computers, and the Internet. Third Edition.Academic Press; 2011.

Distefano A, Me G. An overall assessment of mobile internal acquisition tool. Digit Investig 2008;5(Suppl.):S121–7.

Edwards S, Nichols P. The Current State of Digital Forensics on Mobile Devices. February 2012

Hong K. What is social media. http://seniornet.org/blog/what-is-social-media/; 2012

Hoog A. Android Forensics Investigation, Analysis,and Mobile Security for Google Android. Waltham, MA. Syngress; 2011.

Hoog A, Strzempka K. iPhone and iOS Forensics Investigation, Analysis,and Mobile Security for Apple iPhone,iPad and iOS Devices. Waltham, MA. Syngress; 2011.

Jeon S, Bang J, Byun K, Lee S. "A recovery method of deleted record for SQLite database," Personal and Ubiquitous Computing, vol. 16, no. 6, pp.707-715, 2012

Jung J, Jeong C, Byun K, Lee S. Sensitive privacy data acquisition in the iphone for digital forensic analysis. In: Secure and trust computing, data management and applicationsCCIS, vol. 186. Berlin, Heidelberg: Springer; 2011. p. 172–86.

Kent K, Chevalier S, Grance T, Dang H. Guide to Integrating Forensic Techniques into Incident Response. NIST Special Publication 800-86;August 2006.

Klaver C. Windows mobile advanced forensics. Digit Investig 2010;6:147–67. Embedded systems forensics: smart phones, GPS devices, and gaming consoles (3–4).

Lessard J, Kessler GC. Android forensics: simplifying cell phone examinations. Small Scale Digital Device Forensics Journal September 2010;4(1).

Morrissey S. iOS forensic analysis for iPhone, iPad, and iPod touch. New York: Apress; 2010.

Murphy C. Developing Process for Mobile Device Forensics;2009

Mutawa NA, Baggili I, Marrington A. Forensic analysis of social networking applications on mobile devices. Digit Investig 2012; 9(Suppl. (0)):S24–33.

National Institute of Justice. Test Results for Mobile Device Acquisition Tool: Micro Systemation XRY v6.3.1; February 2013.

National Institute of Standards and Technology. Technical Considerations for Vetting 3rd Party Mobile Applications (Draft). NIST Special Publication 800-163, August 2014.

Pooters I. Full user data acquisition from symbian smart phones. Digit Investig 2010;6:125–35. Embedded systems forensics: smart phones, GPS devices, and gaming consoles (3–4).

Satheesh Kumar S, Thomas B, Thomas K. An agent based tool for windows mobile forensics. In: Gladyshev P, Rogers M, editors. Digital forensics and cyber crime. LNICST, vol. 88. Berlin, Heidelberg: Springer; 2012.p. 77–88.

Son J. Social Network Forensics: Evidence Extraction Tool Capabilities. New Zealand; 2012.

Tso Y-C, Wang S-J, Huang C-T, Wang W-J. iPhone social networking for evidence investigations using itunes forensics. In: Proceedings of the 6th international conference on ubiquitous information management and communication. ICUIMC'12. ACM; 2012. p. 1–7 [Article 62].

ViaForensics, White Paper: appWatchdog Findings. Sensitive User Data Stored on Android and iPhone, 2011

Zdziarski J. iPhone forensics. Recovering evidence, personal data, and corporate assets. O'Reilly Media; 2008.

Zhang S, Wang L. Forensic Analysis of Social Networking Application on iOS devices. Sixth International Conference on Machine Vision (ICMV 2013), edited by Antanas Verikas, Branislav Vuksanovic, Jianhong Zhou, Proc. of SPIE Vol. 9067, 906715; 2013

LIST OF TABLES

LIST OF FIGURES

§1. INTRODUCTION

Mobile devices are used commonly in today's world. People prefer them for personal and organizational purposes. Online communication platforms like Facebook, Twitter and LinkedIn have become widespread by means of mobile devices and vice versa. These kind of websites have influence on millions of people all around the world. According to a research announced in 2014 September by We Are Social agency, 1,56 billion people have active mobile social account[1]. In other words, 22% of total world population use social media via mobile devices. According to published cases by X1 Social Discovery from 2010 and 2011 involving social media evidence, only 2% of cases refer mobile devices as a source[2]. These figures show the significance of social media forensics on mobile devices.

The digital forensic community has to follow rapid changes in mobile device and mobile application world to be able to acquire invaluable evidences. Increase in social media usage on mobile devices effects the likelihood of such devices being involved in an electronic crime like identity theft, drug dealing and fraud (Casey and Turnbull, 2011). Social networks also encourage people to share personal data such as age, location, education, job, religion and some preferences. Majority of the users of these networks are young people and the lure of social media makes them vulnerable to fraudsters, child predators and phishers. Most of the victims and criminals also do not know the artifacts they left on their digital devices. E-mails, text messages, photos, passwords, credit card numbers and internet history are types of evidences which can be retrieved by the help of mobile forensics tools. There is also a possibility to recover deleted contents even if they were deleted from devices.

Law enforcement authorities can also request information about users from social network providers. Providers release law enforcement data request guidelines which is a document describes procedures law enforcement authorities should follow to request data from them. Requested data may be basic subscriber

[1] Available at http://wearesocial.net/tag/sdmw/ , accessed on September 23, 2014.
[2] Available at http://www.x1.com/products/x1_social_discovery/case_law_2011.html, accessed on November 25, 2014

information such as an E-mail, ID number, IP logs, profile photo, connection times and a friend list. Providers may not respond to requests due to privacy policies and jurisdictional differences between countries. Undoubtedly, it is better to have a chance of comparing evidences retrieved from providers and acquired from mobile devices.

I. Aims and Objectives

The aim of this research is determination of the artifacts and potential evidences related to most popular social media applications on mobile devices. Differences between applications and operating systems are tried to be specified in terms of retrieved user artifacts. Most used social media applications in Turkey and in the world are taken into consideration. Facebook, Twitter, Google+, Instagram, WhatsApp and LinkedIn are chosen for investigations. Some worldwide known social media applications such as QQ, Qzone and WeChat are out of topic because of their local popularity and rare usage in Turkey. Mobile devices are also chosen according to mobile operating system usage statistics. Android and iOS operating systems dominate the market in 2014. Therefore, Samsung GT-i9500 Galaxy S IV and Apple iPhone 5S are popular smartphones chosen for this research.

II. Thesis Structure

This thesis organized as follows:

- Chapter 2 gives the definitions of digital forensics, social media and mobile forensics. Mobile forensics is explained in terms of operating systems, evidence extraction methods and tools, evidence types and challenges in this area. Prior work and sample cases in social media forensics on mobile devices are also discussed in this chapter.
- Chapter 3 describes the test environment and requirements. Limitations are explained and low level modification types are discussed. Applied scenarios and acquisition methods with tools are explained together with analysis methods and tools.

- Chapter 4 contains detailed implementation of Chapter 4 and presentation of obtained files and directories.
- Chapter 5 states the summary of findings and evaluation of retrieved artifacts in terms of privacy. Comparisons of applications and operating systems are also specified.
- Chapter 6 states the conclusion and the future work.

§2. LITERATURE REVIEW

I. Digital Forensics

Digital Forensics is the application of computer science to the analysis and recovery of data in various digital devices related to crime while preserving the integrity of the information (Kent et al., 2006). The aim of digital forensics is to acquire clues and information which helps to clarify criminal activity. There are some stages that experts must follow from the original incident alert through to reporting of findings (Casey and Turnbull, 2011). Acquisition, analysis and reporting stages form forensic process. Lots of digital forensic process models have been developed for specific environments but none have been accepted worldwide.

Acquisition means creating a forensic copy of the evidence. Acquiring sector level (bit-to-bit) duplicate of the electronic evidence is preferred if it is possible. In some situations, only logical copy of the electronic evidence can be retrieved and deleted data cannot be recoverable. If electronic device is still power on, some volatile data may be seizable.

Different forensic tools and techniques may be needed while analyzing the forensic copy of the evidence. EnCase, FTK, R-Studio and XRY are some special tools for examiners. Timeline analysis, data carving, string search and media analysis are common methods during analyzation phase.

Reporting is the entire reason to perform the forensic investigation and needs to answer all questions that reader may have. Report must cover the reason of analysis conducted, analyzed evidences, evidence integrity and used process. Conclusion part of the report should be as simple as possible to be understood by nontechnical readers. Finally, the findings of the examination should be listed and any conclusions supported by those findings should be clearly stated.

Digital forensics has developed diverse sub-disciplines, including computer forensics, mobile device forensics, memory forensics, network forensics and database forensics. These are different areas focused on because of electronic evidence variety.

II. Social Media and Social Networks

Social media is a form of communication that users express their ideas and share various data like photos, videos and text messages in an electronic environment. It evolves with the enhancement of smart phones and mobile applications. Social media websites can be classified as blogs and microblogs, social networking sites, virtual game worlds and content communities (Hong, 2012). Nevertheless, some social media platforms may be involved in more than one class. Social media and social network concepts will therefore be used interchangeably throughout this thesis.

According to the research announced in 2014 August by We Are Social agency as shown in Figure 1, Facebook is most popular social platform with 1320 million active accounts in the world. QQ, originally Tencent QQ has 848 million active users but most of them are from China and it is not common worldwide. QQ has simultaneous online users record with a peak of 176.4 million on 20 March 2013[3]. Qzone is also created by Tencent and 644 million users prefer it. WhatsApp has 500 million users and most of them are from Latin America, The Middle East and Africa[4]. People from China, South Korea and Japan generally do not prefer WhatsApp due to popular local social platforms.

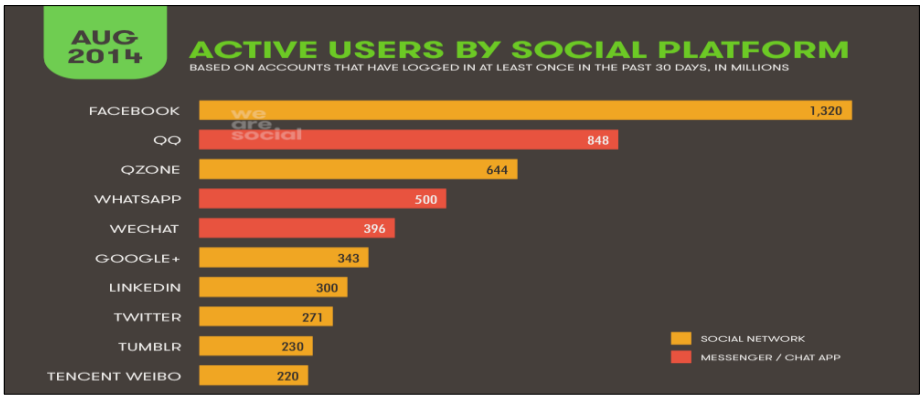


Figure 1 - Active users by social platforms in the world

[3] Available at http://www.prnewswire.com/news-releases/tencent-announces-2012-fourth-quarter-and-annual-results-199130711.html, accessed on September 24, 2014.
[4] Available at http://wearesocial.net/blog/2014/09/global-usage-whatsapp/, accessed on September 24, 2014.

WeChat is another communication service developed by Tencent in China which reaches 396 million users. Google+ is owned and operated by Google Inc. and platform's popularity is growing faster than Facebook[5]. LinkedIn is a business-oriented social networking service which is most popular in USA and India. Twitter is most popular microblogging website described as the SMS of the internet and nearly 500 million tweets sent every day. Tumblr is also a microblogging platform owned by Yahoo in May 2013 and hosts over 204,8 million blogs and more than 92,2 billion posts in total as of September 26, 2014[6]. Tencent Weibo is a microblogging platform similar to twitter with a 140 character limit.

Detailed usage statistics about Turkey is announced in 2014 July by We Are Social agency. Most used social media platforms are Facebook, Twitter, Google+, Instagram and LinkedIn. 4 of them except Instagram are also in most used 10 platforms in the world. Instagram is a mobile photo sharing platform and nearly 200 million Instagrammers capturing and sharing their lives.

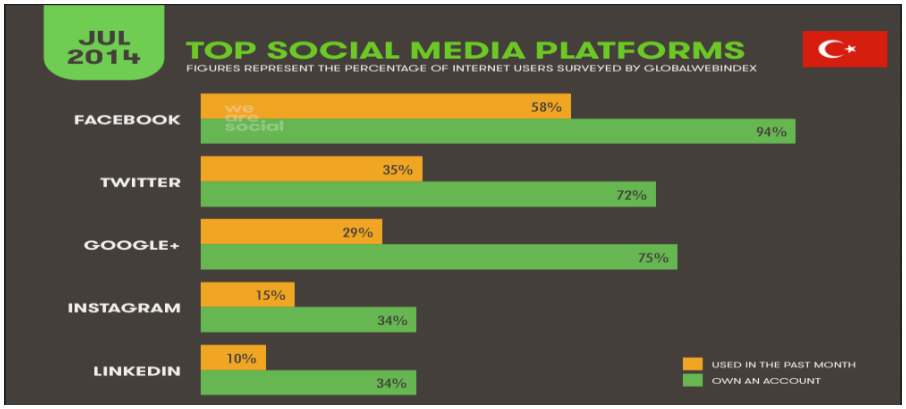


Figure 2 - Top social media platforms in Turkey

III. Mobile Forensics

Mobile Forensics is the science of recovering digital evidence or data from a mobile device under forensically sound conditions using accepted methods (Ayers et al., 2014). It is a new discipline in digital forensics area and effective methods are required due to an increasing frequency of finding smartphones at crime scenes.

[5] Available at http://wearesocial.net/?s=google%2B, accessed on September 24, 2014.
[6] Available at https://www.tumblr.com/press, accessed on September 26, 2014.

Existing computer forensic methods are not satisfactory and the nature of mobile forensics is challenging. Different CPU architectures, well-secured operating systems, limited memory resources and variety of proprietary hardware are general difficulties that mobile forensic examiners have to overcome.

Mobile forensic process can be divided into three main category; seizure, acquisition and analysis. While seizing the mobile device as a source of evidence, it must be disconnected from any network. If the device is powered off, it can be placed in a faraday bag which is a special material for isolation of device. If the device is powered on, it is better to leave the device switched on to acquire the volatile data. If the phone is encrypted or password protected, powering it off may cause inaccessibility of digital evidences. It is needed to place the device in an airplane mode and to disable all network connections to prevent remote wipe commands. Battery life problem may occur over the device and available external power source may be needed during acquisition process (Bommisetty et al., 2014).

There are many commercial and open source tools available to acquire and analyze digital evidences in mobile devices. Tool type and current state of the phone affects the amount of retrieved evidence. Multiple methods can be needed to extract all existing significant data. It can be challenging and expensive to acquire some critical data due to current unbreakable security features of mobile devices. Some acquisition methods are easy to perform with a simple software but some professional methods necessitate to desolder the internal memory chip.

The essential side of mobile forensics is avoiding any unnecessary data modification from beginning to the end of whole stages. Documenting every step of the investigation is also indispensable for examiners. Low level alterations like rooting and jailbreaking may be needed and have to be explained in detail for admissibility of the evidences in court. Due to the fact that mobile devices are generally active during investigations, possibility of data update must be taken into consideration by forensic experts. Differences may occur between forensic images of the same device. But, it is expected to get the same hash value for an individual file.

A- Operating System Perspective

Operating system type is one of the major factors in data acquisition from smartphones. Technology for telecommunication is standardized but hardware and software types are diverse. Current market shares in Table 1 shows that Android and iOS dominates the market in the second quarter of 2014. It is so clear that market share of Android is increasing while others' shares are decreasing in last four years. iOS is developed by Apple Inc. and used only by Apple mobile devices such as iPhone, iPad and Apple TV. Android is Linux-based open source operating system developed by Google Inc. and used by various vendors such as Samsung, LG and HTC. In this research, forensic analysis of these two mobile platforms will be covered and some information will be given about other platforms as needed.

Period	Android	iOS	Windows Phone	BlackBerry OS	Others
Q2 2014	84.7%	11.7%	2.5%	0.5%	0.7%
Q2 2013	79.6%	13.0%	3.4%	2.8%	1.2%
Q2 2012	69.3%	16.6%	3.1%	4.9%	6.1%
Q2 2011	36.1%	18.3%	1.2%	13.6%	30.8%

Table 1 - Market shares of mobile operating systems[7]

1. Android

Android was first released in 2007 by Open Handset Alliance which is a collaboration of mobile technology companies[8]. It is an open source platform built on the Linux kernel. Linux is portable and compiled easily on different hardware. Open source nature makes Android free and customizable. Each Android version has a special name and they are released in alphabetical order.

Mobile applications which increases the functionality of devices generally written in Java. Preinstalled applications like Contacts and Web Browser comes

[7] Available at http://www.idc.com/prodserv/smartphone-os-market-share.jsp , accessed on October 16, 2014.
[8] Available at http://www.openhandsetalliance.com/press_110507.html, accessed on October 16, 2014.

within the device as a default. User-installed applications like WhatsApp and Instagram can be downloaded from Google Play Store, Amazon Marketplace and so on. Android applications are not affected by other applications or system functions since each application runs in its own sandbox. An application cannot access the data of another one due to this security feature.

Android version 4 provides a disk encryption feature for the user partition. The encryption is based on dm-crypt which is a Linux kernel feature. Versions prior to 4.4 used a fairly easy to brute force key derivation function. Android 4.4 allows users to create a separate encryption password. It is expected for new versions of Android to come with hardware level encryption mechanisms and enabled full disk encryption by default.

The first step of Android forensics is identifying the device properly. Device vendor, device model and operating system version should be primarily determined. Then, suitable device cable and certain drivers to connect the device to the forensic workstation must be supplied. The USB cable interface of the device may be mini-USB, micro-USB, EXT-USB and so on. There is no single generic driver for all Android devices and each manufacturer writes its own driver. USB debugging option of the phone must also be enabled for accessing to the device.

Linux forensic techniques are not enough for Android forensics since different Read Only Memory (ROM) types and kernel modes. The Android Software Development Kit (SDK) is a collection of software used for application development. It is also a convenient forensic tool to access to the data in device. After enabling successful connection between workstation and device, Android Debug Bridge (ADB) in SDK enables communication between client and server. If the device is locked (by passcode, PIN code or pattern lock), it can be required to bypass it. Many techniques to bypass these lock mechanisms are available and they can be applied from easiest to hardest depending on the state of the phone. Deleting the gesture.key file, updating the settings.db file, smudge attract, Gmail account usage, recovery mode method and flashing a new recovery partition methods are common for bypassing the lock mechanisms (Hoog, 2011). The last and hardest

methods are Joint Test Action Group (JTAG) method and micro-read which will be discussed later.

The integrity of the device is crucial for admissibility in court but some low level modifications like rooting may be necessary to acquire data in a forensic manner. Rooting is simply exploiting a security bug in the device's firmware and gaining superuser access. If the device is not rooted, ADB connection will not be exactly completed. Accessing the sensitive private data in an application generally requires root access. The examiner must explain and defend the need for rooting clearly. Positive and negative consequences of the action must be stated fairly for admissibility of the evidences. By the way, the device may have been rooted before and this may be an advantage for forensic examiner.

2. iOS

iOS (previously iPhone operating system) is a mobile UNIX-based operating system unveiled in 2007 by Apple Inc. It is a closed source operating system unlike Android that means the source code is not available to the public. It is reproduced from MacOS X and used in all Apple mobile devices such as iPhone, iPad, iPod and Apple TV. According to Table 2, iOS was the most used operating system for Tablets until 2014 July worldwide.

Month	Android	iOS	Java ME	Symbian	Windows Phone	Other
November, 2013	33.89%	55.17%	4.49%	3.12%	0.67%	2.66%
December, 2013	35.41%	54.27%	3.90%	3.18%	0.55%	2.70%
January, 2014	34.60%	54.46%	4.26%	3.41%	0.56%	2.72%
February, 2014	36.14%	52.96%	4.44%	3.50%	0.45%	2.51%
March, 2014	36.58%	53.29%	3.36%	3.92%	0.69%	2.17%
April, 2014	37.75%	51.11%	4.36%	3.77%	0.83%	2.18%
May, 2014	41.58%	48.34%	3.46%	2.52%	2.10%	2.00%
June, 2014	43.75%	45.61%	3.77%	2.73%	1.99%	2.15%
July, 2014	44.62%	44.19%	4.19%	2.57%	2.49%	1.94%
August, 2014	45.01%	44.34%	3.77%	2.61%	2.69%	1.57%
September, 2014	47.06%	43.86%	2.78%	2.56%	2.38%	1.37%

Table 2 - Mobile/Tablet top operating system share trend[9]

[9] Available at https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=9&qpcustomb=1, accessed on October 17, 2014.

Each iOS versions have special names as in Android and updates are provided by Apple for new features, latest hardware and bug fixing from time to time. iPhone operating system 1.0 was the first release and new operating systems come with a numerically increasing approach like 1.1, 2.0, 2.1.1, 3.0 and 3.1.1. Mobile applications for iOS can be written with iPhone SDK and downloaded from App Store after 2.0 release. Global Positioning System (GPS) which creates great evidences for different kinds of cases is also available since this release. The first iPad comes with iPhone operating system 3.2 and Apple renamed the iPhone operating system to iOS with iOS 4.0. Apple's cloud service, namely iCloud comes with iOS 5.x and new forensics area was opened for iOS examiners.

iOS has many security features from core to surface that forensic experts must generally bypass. Passcode protection is the frequently encountered data protection mechanism. The passcode was stored simply in keychain, which is a password management system until iOS 3. Removing the record from the keychain was enough for bypassing. After iOS 4, password is not stored in the phone and data in the phone is encrypted if passcode is active. Decryption of system keybag is necessary in order to reach encrypted keychain items. Brute force attacks for four-digit passcodes many cause to lock the device or to wipe the contents. It must be performed on kernel level to prevent locking the device. Brute force attacks for determination of the passcode nearly takes 18 minutes for simple four-digit and 51 hours for four-alphanumeric passcode (Bommisetty et al., 2014).

The entire file system is encrypted in iOS devices with hardware level encryption. Secure erase mechanism of iOS removes encryption keys to protect data. There is no way to get back the encryption keys (Apple Inc., 2014). Even though data is not overwritten, it is impossible to recover encrypted contents.

Application security is provided by Apple's approval, code signing and sandboxing. Developers sign and submit the designed application to Apple for approval and if suitable, it is published on the App Store. Apple-issued certificate is required for signing the application. Unlike Android applications, App Store is the only place to download iOS applications. App Store is less vulnerable than Android application downloadable sites resulting from code signing and Apple's

approval. Sandboxing controls the limits of access to files created by the other applications like in Android.

Identification of device type and firmware version is the starting point of iOS forensics. Written information back of the device can be helpful to identify it if available. If not, internet search for shape, color and appearance may be helpful. Some information such as firmware version, capacity, serial number, IMEI and model number can be found under Settings>General>About path. Suitable cable and certain drivers for connection to the workstation must be provided afterwards. Apple uses a specific 30-pin connector for charging and file transfer. Examiner must connect the device to the workstation and find the proprietary drivers to start examination.

iOS devices have three operating modes; normal mode, recovery mode and device firmware upgrade (DFU) mode. Each mode has a specific function as a forensic perspective. Normal mode is a default mode that operating system is booted. This is the user mode for calling, sending sms and other regular activities. Recovery mode is for upgrading or downgrading the device and achievable by bypassing the booting of the operating system. DFU mode is low-level mode to upgrade firmware and to acquire a physical image of the device.

Low level modifications like jailbreaking may be necessary to acquire some evidences from device. General approach for jailbreaking is nearly the same as rooting for admissibility of the evidences. Boot ROM vulnerabilities for iPhone 4 and older devices facilitate physical acquisition. But, no such vulnerability is found for iPhone 4S and subsequent models. Jailbreaking may be inevitable for physical acquisition of recent devices.

3. Other Operating Systems

The sum of utilization rates of other operating systems is nearly %4 nowadays. Consequently, priority should be given to Android and iOS. But, it is inevitable to encounter Windows Phone, Blackberry, Symbian devices and so on for mobile forensic specialists. Windows Phone is popular in Asia, Latin America and Africa. Blackberry devices are commonly used for security and data protection

features. Symbian was one of the most popular mobile operating system until the end of 2010 (Barmpatsalou et al., 2013).

Commercial tools and Windows Phone Device Manager can be used for acquisition of Windows Phone devices. Bypassing the security features, accessing the phone and deleted data recovery are challenging aspects of these devices like other recently invented device models. Physical acquisition may be dangerous for some Windows Phone models (Klaver, 2010). Moreover, performing a standard physical acquisition method is not satisfactory due to an internal memory diversity (Satheesh Kumar et al, 2012).

Blackberry is harder to examine and Blackberry Desktop Manager is used as a forensic tool by many experts. Bypassing the security mechanisms such as passcode protection is generally troublesome. Some methods are also getting encrypted useless data. Back up method is mostly the only way for user data.

There was no security mechanism to bypass for Symbian older phone models (Distefano and Me, 2008). The tool was developed namely SMIT for bit-to-bit copy of the file system and deleted data recovery (Pooters, 2010). This was the great step in Symbian forensics world because previous tools could not retrieve deleted contents.

B- Evidence Extraction

Evidence extraction methods and tools are the fundamental side of mobile forensics. Worldwide accepted process model is not still available but there are some attempts for standardization. There is no unique tool to analyze all device models. Furthermore, there is no method to analyze some models effectively. The main reasons are too many device type and very rapid emergence of new products. Understanding the software and hardware characteristics is necessary as well as bypassing the security measures for a complete investigation.

Mobile devices comprise volatile and non-volatile memory. Most of the latest smartphones contain Random Access Memory (RAM) as a volatile memory and NAND as a non-volatile memory. When device is powered off, dynamic data

in RAM is lost. But, data is still available in NAND until secure erase or wipe operations. RAM may contain some crucial data such as passwords and credit card numbers. But, it is more difficult to acquire volatile data than non-volatile data from mobile devices.

In this research, data extraction types and tools from only non-volatile memory will be explained and performed.

1. Methods of Extraction

Data extraction process for mobile devices means taking as much information as possible within a reasonable time. This process can be divided into three main category; manual, logical and physical extraction methods. It may be needed to apply all methods one by one to check the results. Degree of difficulty of the methods are not the same. In Figure 3, extraction methods are classified (Brothers, 2011). From bottom to top, methods are getting harder and more expensive in each level. More experience and training are also required.

Figure 3 - Mobile device extraction methods

a) Manuel Extraction

The simplest meaning of manual extraction is observing the contents on the screen of the device. Manuel extraction is the easiest method but it can be time-consuming. If other methods are not applicable, this is the last chance of evidence collection. Reporting of the case may be troublesome with manual methods like taking picture or video recording of the screen. The precondition of the method is

successful booting of operating system with undamaged keyboard and screen. Experts must be cautious while navigating between menus to avoid deleting any content.

b) Logical Extraction

Logical extraction is a method applied with the help of back up and device synchronization features with a computer. The method starts with connection to the workstation generally via USB cable, RJ-45 cable or Bluetooth. Then, necessary drivers for device must be installed for detection by workstation. Existing files and directories without deleted contents in the device can be acquired with this method. In some cases, retrieved files like SQLite can contain deleted data. Logical acquisition is generally easier than physical acquisition. As a result, many mobile forensic tools can perform logical extraction. In addition, official back up and synchronization tools such as iTunes, Samsung Kies and BlackBerry Desktop Suite can retrieve many evidences as a logical extraction. The crucial side is prevention of any modification on device.

c) Physical Extraction

Physical acquisition is more likely imaging a computer. The method enables direct access to the internal flash memory and bitwise copy to the workstation. Unallocated space data which includes deleted contents can also be retrieved and many forensic methods can be applied on physical image. Even though manufacturers place some security mechanisms to protect the contents, some low level methods can dump the memory. After dumping stage, decoding is required for clarity.

Level 3 and above in Figure 3 shows physical extraction methods. More technical background and specific education is needed with particular equipment for high level methods. Experts must choose suitable and feasible one for each case based on the facilities in their lab environment.

Hex dumping is widely applied method for physical extractions. After connecting the device via cable or Wi-Fi, specific boot loader is used for dumping

the flash memory. Parsing and decoding is required since acquired data is raw and binary. JTAG method is applied through connection to the Standard Test Access port on a device and obtaining raw data from memory chips. The challenging parts of JTAG method are finding test access port and correct chips, solving memory reading protocol and applying correct voltage. It is harder than hex dumping and can be more effective for locked devices.

Chip-off method is acquisition of raw data from memory chip. Chip must be desoldered cautiously. After cleaning the chip, specific chip reader device must be used for extraction. Then, extracted binary raw data can be analyzed with various software. If data was encrypted, encryption key is necessary to decrypt the data. Variety of chip types may necessitate several hardware and software to apply this method. Detailed hardware knowledge is required for experts to avoid data destruction while heating and desoldering. Less destructive data extraction methods should be performed before chip-off if available. Damaged and inactive devices can be analyzed by this method if memory chip is in good condition.

Micro read process is reading gate status values, 0's and 1's from chips by the help of electron microscope after desoldering and cleaning chips. The creation of ASCII values from bits and forming comprehensible information is the next process. This costly and time consuming method is only applicable for high profile cases related to national security and terrorism. There is no standard methodology in this field. Moreover, no commercial tool is available for a complete investigation (Ayers et al., 2014).

2. Extraction Tools

The rapid development of mobile device market necessitates the improvement of mobile forensic tools. Preliminary investigations were applied by manually looking at the screen of devices and writing down significant data. Photographing the screen was the subsequent more reliable idea. After device synchronization with computer and back up features of mobile devices, logical acquisition methods and tools were developed. Learning internal specifications and bypassing security features of mobile devices resulted in physical extraction

methods and tools. Tools are updated frequently due to new device models and operating system version changes. It is advisable for experts to use more than one tool in their labs to compare examination results. They may also use non-forensics tools for device backup, testing and management.

An ideal mobile forensics tool should meet some requirements. It must recognize the mobile device by cable, Bluetooth or so on. It must also perform a logical acquisition and should perform a physical acquisition. It must produce a comprehensible report and should generate a log file explaining all steps. Various language support and hash calculation for each file improves the quality of the tool. The results of the tool should be useful and comprehensive. Obtained images at different times should be identical. The tool should not perform unnecessary modification of the data.

National Institute of Standards and Technology (NIST) has classified mobile forensics tools and gives some information about them[10]. The Computer Forensics Tool Testing (CFTT[11]) program is also testing the mobile forensics tools and provides a detailed information about the test result of each tool[12]. CFTT is testing acquisition and reporting of social media related data with the help of Facebook, Twitter and LinkedIn applications.

XRY, Cellebrite UFED and Oxygen are the most widely used commercial tools that will be mentioned briefly.

a) XRY

XRY is developed by Micro Systemation Company from Sweden. The company was founded in 1984 for mobile communication area, but they have focused on mobile device forensics since 2003. XRY is used by police, law enforcement, military, government intelligence agencies and forensic laboratories

[10] Available at http://www.cftt.nist.gov/tool_catalog/populated_taxonomy/index.php?ff_id=5, accessed on October 23, 2014

[11] CFTT program is a joint project of the Department of Homeland Security (DHS), the National Institute of Justice (NIJ), and the National Institute of Standards and Technology Law Enforcement Standards Office (OLES) and Information Technology Laboratory (ITL).

[12] Available at http://www.cftt.nist.gov/mobile_devices.htm, accessed on December 9, 2014.

in over 90 countries worldwide[13]. XRY 6.11 version was released in 13th October 2014 and 13048 device profiles supported with this release. The complete kit consists of various connection cables, card reader, SIM id-cloner and software with licence key.

XRY/XACT v6.10.1 was tested within CFTT program. It was tested across supported Android and iOS devices and a feature phone[14]. Social media application data reporting performance of the tool for Android and iOS devices was that the tool acquired and reported data successfully. For a future phone, the tool was unable to perform the test for social media applications.

b) Cellebrite UFED

Cellebrite tools have been developed by the Cellebrite Company since 1999. More than 300 people are working for a company. 60 countries are using Cellebrite Universal Forensic Extraction Device (UFED) all over the world. UFED performs physical, logical, file system and password extractions from various devices. UFED 4.0 version was released in September 2014 and 13685 device profiles are supported with this release[15]. The complete kit contains cables, adapters, various cases, software and son on.

UFED Physical Analyzer v3.9.6.7 was tested within CFTT program. It was tested across Android, BlackBerry, iOS, Windows mobile and various feature phones[16]. Social media application data reporting performance of the tool for Android and iOS devices was that the tool failed to return expected test results. For BlackBerry, Windows Mobile and Feature Phones, the tool was unable to perform the test for social media applications.

---

[13] Available at https://www.msab.com/company/about-us, accessed on October 23, 2014

[14]Available at https://cyberfetch.org/sites/default/files/XRY-XACT%20v6.10.1%20Test%20Report.pdf, accessed on December 9, 2014.

15 Available at http://releases.cellebrite.com/releases/ufed-release-notes-4-0.html, accessed on October 24, 2014

16 Available at https://cyberfetch.org/sites/default/files/UFED%20v3.9.6.7%20Test%20Report.pdf, accessed on December 9, 2014.

c) Oxygen

Oxygen Software Company was founded in 2000 and their tool supports more than 8400 devices as of today. "Oxygen Forensic Suite" and "Oxygen Forensic Suite 2014" are the trademarks of Oxygen Software. It is widely used in USA and Europa by federal and state agencies[17]. Examiner can import and analyze image files generated by other mobile forensics software by the help of Oxygen Forensic Suite.

d) Open Source Tools

Commercial tools are expensive and they do not support all devices. They still miss some of the data and do not parse all third party applications. Open source mobile forensics tools are generally platform-specific and concentrates mainly on smartphones. iTunes, iPhone Backup Analyzer, iExplorer, iBackupBot and Plist Editors are example tools for iOS devices. FTK Imager, Android SDK, Open Source Android Forensics Toolkit, AFLogical, dd command and Autopsy Android Module are example tools and platforms for Android devices. Katana Forensics' Lantern Lite imager, TULP2G, BitPim, WhatsApp Extract, Skype Xtractor, Foremost, SQLite Browsers, Hex Editors are also popular tools for mobile forensics.

C- Types of Evidences

Evidences related to mobile devices can be found in internal memory, volatile memory like RAM, external storage like Micro SD card and SIM card. Most of the evidences exist in devices' internal memory and tools are mostly developed to retrieve these data. Some limited researches about retrieving volatile data from mobile devices are available but this area is more challenging. SIM card readers and data acquisition software were developed by various companies for

[17] Available at http://www.oxygen-forensic.com/en/company, accessed on October 24, 2014.

stand-alone examination of SIM cards. Micro SD card investigation is similar to standard computer forensic investigation and they can be examined individually.

Following points can be potential evidences in mobile forensics area:

- İdentifiers; IMEI, ESN, ICCID and so on.
- Address Book; names, numbers, E-mails and son on.
- Call History; received, dialed and missed calls in addition to call durations.
- Calendar Information; calendar entries.
- Messages (SMS, MMS); sent and received text messages, videos, and pictures with time stamps.
- E-mails; sent, received and draft mails.
- Multimedia; photos, videos and audios (with metadata information) created by devices, downloaded from internet and sent by others.
- Web History; visited websites.
- Electronic Documents; Various documents created, downloaded or transferred by.
- Application data; data stored by installed or default applications.
- Geolocation data; GPS data, routes and maps.

Ideal mobile forensics examination should contain all potential evidences. But some points should be targeted firstly for efficiency. For example, examination sequence of drug dealing and child pornography is not the same.

D- Challenges

It is inevitable for experts to encounter difficulties in every phase of mobile forensics. From seizure to reporting, explaining technical and legal remarkable hardships briefly will help researchers.

The first challenge in this field is differences in hardware designed by sheer number of manufacturers. Lab environments must contain batteries, chargers, cables and drivers for each candidate device. If low level physical acquisition

methods are required, proprietary chip reader, data parser and decoder may also be needed.

The second challenge is operating system type and frequent version updates. After identifying the operating system type and version, proprietary investigation methods must be performed. Each operating system has different file systems and some of them are closed source. Limited investigation methods are available for some of them. The method can be effective for a device and operating system version, but it can be useless for its successors.

The next challenge is security features. Full disk encryption mechanisms, passcode locks and wipe features are most common security mechanisms encountered by examiners. Some of these can be bypassed but no vulnerabilities have been found for some of them.

Another challenge is data update. Mobile devices are generally active while acquisition. Unexpected intervention to the device may cause loss of critical data. Some running processes may change the state of the device. Unclosed cellular networks, Bluetooth, Wi-Fi and Infrared may also alter the data content. Keeping the device in faraday bag is the most common precaution. If some necessary operations cause data update, it must be stated clearly in a report.

Another challenge is choosing the right software. There are many commercial and open source tools available in this field. It may be unreasonable to buy and use all of them but necessary software which is capable of acquisition and analyzation must be chosen. However, using more than one tool gives a chance of comparing the results.

The last challenge is jurisdictional issues. There may be differences between laws related to device seizure, chain of custody, rooting and jailbreaking in different countries. Moreover, there may not exist any legislations about these issues in some countries. Forensic examiner should know the jurisdictional side of the mobile forensics science.

IV. Social Media Forensics on Mobile Devices

Many people have been working on social media and mobile forensics topics. A number of cases frequently come up about these topics. It is important to mention previous works and sample cases about social media forensics on mobile devices.

A- Related Work

The number of criminal cases related to social media are increasing all over the world. New research findings and guidelines about social network artifacts are needed. Researchers have found some investigation methods and explained obtained critical data that may help to enlighten the cases. New researches should be performed in this area due to new mobile devices and new social networks.

Bader and Baggili examined iPhone 3GS and found Facebook related database which consists some information about Facebook friends (Bader and Baggili, 2010). Lessard and Kessler examined Android phone HTC Hero and found user name, unencrypted password and various information about Twitter (Lessard and Kessler, 2010). They also found some Facebook related data in contacts database. Morrissey stated his findings about Skype, Twitter, LinkedIn, AOL AIM and Facebook applications in his book (Morrissey, 2010). He found user names, E-mails, ID numbers, friend lists, location information and time stamps. Acquisition and analysis of sensitive data from Facebook, Twitter and some Korean social networking applications with an iPhone was explained in 2011 (Jung et al., 2011). Tso et al. mentioned back up file information about Facebook, WhatsApp and so on (Tso et al., 2012).

Forensic analysis of Facebook, Twitter and MySpace applications on BlackBerry, iPhone and Android phones was performed in 2012 (Mutawa et al., 2012). The aim of the research was determination of the location, amount and significance of critical data related to widely used social networking applications on mobile devices. Similar research was performed on iOS devices for QQ,

WeChat, Sina Weibo and Skype applications which are very popular in China (Zhang and Wang, 2013).

To sum up, most of the previous researches about social network application forensics have been performed on most used applications, most used mobile phones and most used operating systems. Researches generally do not include any low level modification like rooting and jailbreaking. Logical acquisition methods have been preferred instead of physical ones. It is advisable for further researchers to use physical acquisition methods to recover deleted contents.

B- Sample Cases

In Oscar Pistorius case, WhatsApp messenger messages reveal 35,000 pages worth of messages between the couple. The evidence was extracted and shown to the court using XRY software[18].

In Ohio rape case[19], evidence on social media and on the phone enlighten the case. Text messages posted to some social networks show that the rape happened. A photo snapped by a mobile phone shows the girl naked on a floor. Digital video which is published online shows a group of students joking about the assault. In this case, no physical evidence of the rape exists. The court focused strictly on the interpretation of the media evidence.

In a teen murder case[20] in Canada, text messages, Wikipedia searches, instant messages, a confession in a World of Warcraft chat, GPS data associated with a text message sent from the scene of the murder and Google map searches for places to dump the body were found as an evidence. In total, the Tech Crimes Unit amassed the equivalent of 1.4 billion sheets of paper from computers and phones.

[18] Available at https://www.msab.com/posts/news/xry-used-in-oscar-pistorius-trial, accessed on November 20, 2014

[19] Available at http://www.minnpost.com/christian-science-monitor/2013/01/ohio-rape-case-evidence-social-media-creates-new-world-justice, accessed on November 20, 2014

[20] Available at http://www.forbes.com/sites/kashmirhill/2011/11/03/solving-a-teen-murder-by-following-a-trail-of-digital-evidence/, accessed on November 20, 2014.

In a treason case involving famous Rwandan singer Kizito Mihigo and journalist Cassien Ntamuhanga, shared messages over the phone, WhatsApp and Skype was an evidence[21].

Federal investigators in USA linked the Landry Crew to the crimes in part by mining email, cellphone records and social media accounts[22]. Some of the gang members took pictures of themselves after the robberies rolling around in cash and stolen items. Facebook, Twitter, Instagram and YouTube videos are found posted, apparently mixing cough syrup with soda, with a pistol and a bag of marijuana nearby.

[21] Available at http://www.theeastafrican.co.ke/news/Phone-evidence-used-in-terror/-/2558/2294196/-/klwpvi/-/index.html, accessed on November 20, 2014.

[22] Available at http://www.sfgate.com/crime/article/7-East-Bay-gang-suspects-indicted-after-social-5336622.php, accessed on November 20, 2014

§3. RESEARCH METHODOLOGY

The main purpose of this research is determination of user artifacts related to social media applications on mobile devices. Device acquisition methods, useful forensic tools and critical data types will be explained briefly. The methodology in this research is divided into four main phases; rooting and jailbreaking, scenarios, acquisition and analysis.

New phones including current operating systems were prepared primarily. Rooting and jailbreaking operations were achieved at first step. Then, both physical and logical acquisition of user data free devices were performed. Acquisition methods and acquisition tools were explained. Then, up to date mobile applications of most popular six social network were installed in each device. Imaginary accounts were created and some activities were prepared for each application. After that, physical and logical acquisition of devices were performed again with same methods and tools. Finally, analysis were conducted and explained in detail.

I. Test Environment and Requirements

For a complete mobile forensics investigation, forensic workstation must contain many hardware and software. Availability of both commercial and open source software provides a better quality of examination. Hardware sufficiency is also inevitable for a complete investigation.

The following is the list of used hardware and software in this research:

- Windows 7 Professional 64-bit operating system with 8 GB RAM
- Samsung GT-i9500 Galaxy S IV (16 GB capacity and Android 4.4.2)
- Apple iPhone 5S (A1457 chip, 32 GB capacity and iOS 8.1)
- Mobile applications of Facebook, Twitter, Google+, Instagram, WhatsApp and LinkedIn for iOS and Android
- XRY v6.11.1
- AccessData FTK Imager v3.0.0.1443
- Android SDK

- VMware workstation 9.0.1
- SANS Investigative Forensic Toolkit (SIFT) 2.13 Linux Workstation
- Pangu v1.2.1 for jailbreak
- Odin3 v3.09 and CF-Root Package for rooting
- Putty v0.63
- WinSCP v 5.5.6
- HFSExplorer 0.21
- WinHex 15.9
- Plist Editor Pro v2.0
- SQLite Database Browser v3.4.0
- Micro USB cable for Samsung phone
- Lightning to USB Cable for iPhone 5S
- 16 GB Micro SD card
- 2 Avea SIM Cards

In this research, main platform is Windows 7 operating system. Linux based SIFT 2.13 workstation is used in virtual machine. Android phone is rooted with Odin 3.0.9 and iOS phone is jailbreaked with Pangu software. Mobile applications are downloaded from Apple Store for iPhone and Google Play Store for Samsung. Logical acquisitions are performed with XRY v6.11.1 and physical acquisitions are performed with UNIX "dd" command[23]. Various programs are used in Windows and Linux environment for analyzation of forensic images. Specific files obtained from images are opened and analyzed with particular editors and browsers.

Mobile phones had to be unused or previously wiped in this research. They also must be unlocked to perform physical acquisitions. Rooting and Jailbreaking could not be performed for locked devices. There is still no method for iPhone 5S to bypass the passcode. Moreover, there is no tool to perform physical acquisition without jailbreaking. XRY v6.11.1 commercial software cannot perform physical

[23] dd can be used to copy from source to destination, block-by-block, regardless of file system types or operating systems. dd can also be used for imaging raw disk partitions of mobile devices.

acquisition for both phones. Low level modifications like rooting and jailbreaking for physical acquisitions seem necessary for both of them.

II. Limitations of Research

In this research, the first limitation is mobile operating systems. iOS and Android mobile operating systems were chosen for their high usage rate. iOS 8.1 and Android 4.4.2 were the last versions when the examination was started. Jailbreak and rooting methods are different for different operating system versions. Acquisition and analyzation methods are also changeable.

The second limitation is social network apps. Last versions of 6 mobile social network apps were installed and analyzed. There could be some difference between different versions of applications. Analyzation of differences between different versions of an individual application is another topic which is not mentioned in this research.

The third limitation is commercial software for forensic imaging. XRY v6.11.1 is the only commercial tool used for forensic imaging and analysis. Commercial tools generally facilitate the examination. But, it is not easy to make a payment for examiners. However, more than one open source software were used for imaging and analysis in this research.

III. Rooting and Jailbreaking

The simplest meaning of rooting is gaining root access to a device. It gives permission to replace system applications and to run special applications which requires administrator-level permissions. Jailbreaking is removing limitations on device and enables to install and use various applications like SSH. Although the main concept behind them is similar, the limitations of control over devices are different. The goal of rooting and jailbreaking is the same in this research. The main purpose of these low level modifications is acquiring physical images of devices.

There were no chance to acquire physical images without rooting and jailbreaking. These kind of modifications were performed before applying the

scenarios to prevent data loss. The main reason for choosing this method is trying to find all user artifacts in devices. Low level modifications may give a chance to detect unpredictable user remnants.

Admissibility of these mobile devices is also indispensable. Most of the countries do not have laws including rooting and jailbreaking. In some situations, data acquisition from mobile devices can be impossible without low level modifications. For example, logical image is not enough if all user data was deleted from device manually and low level alterations are necessary unless any tool supports physical imaging. In addition, widely used commercial tools may also have a support for rooting. This support also shows the necessity of modification on devices.

Different acquisition methods also may give different outcomes. All possible acquisition methods had to be applied before low level modifications. Rooting and jailbreaking can be performed as a last resort. Most of the previous researches are based on non-modified devices and logical acquisition methods. Consequently, there is a huge need for forensically sound low level modifications and physical acquisition methods.

IV. Scenarios

Preparation of specific scenarios for each application is preferred instead of real user data. The aim of the scenarios is comparing the obtained user artifacts with the scenarios. This comparison facilitate to understand which user activities were stored in the phone memory. It will be detected easily if a specific activity did not leave an artifact.

Each scenario is started from operating system installation. SIM cards were provided and both devices were connected to the internet via Wi-Fi. E-mail accounts were created and some security questions were answered while installing operating systems for each phone. After jailbreaking and rooting operations, last versions of social network applications were installed. Applied scenarios are explained in Table 3.

Application	Version		Activity
	iOS	Android	
Facebook	18.0	21.0.0.23.12	Account creation Profile update Comment posting Comment Editing Location sharing Photo upload
Facebook Messenger	15.1	16.0.0.16.15	Sending a text message Receiving a text message Deleting a message Sending a photo Deleting a photo
Twitter	6.17	5.34.0	Account creation Profile update Follow some accounts Unfollow some accounts Comment posting Private message sending to a friend Location sharing Photo upload Sample search
Google+	4.7.4	4.2.4	Account creation Profile update Follow some accounts Unfollow some accounts Comment posting Location sharing Photo upload
Instagram	6.2.0	6.10.1	Account creation Profile update Follow some accounts Unfollow some accounts Photo sharing Deleting a photo
WhatsApp Messenger	2.11.12	2.11.432	Account creation with phone number Profile update Adding a friend Sending and receiving a text message Sending and receiving a photo Location sharing Deleting any content
LınkedIn	8.1.58	3.4.3	Account creation Profile update Searching a friend

Table 3 - Activities performed for each application on each device

V. Acquisition

Acquisition phase is most challenging part of this research. Both physical and logical images of each devices were acquired. Before application installations, logical and physical images were acquired when the devices were free of user data. After installing the mobile applications, images were acquired again with same methods and tools. This procedure gives us the chance to compare the previous and next images. After comparisons, it will be easier to detect user remnants in devices.

XRY v6.11.1 is the only commercial software to acquire forensic images of mobile devices in this research. It has the ability to acquire physical images of iPhone 4 and previous iPhone devices. Unfortunately, because of hardware level security mechanisms, XRY v6.11.1 cannot acquire physical images of iPhone 4S and subsequent iPhone devices. Logical acquisitions could be performed easily with XRY v6.11.1 and open source UNIX "dd" command line tool was preferred for physical acquisition. Jailbreaking applied and Secure Shell (SSH) network protocol was used to acquire physical image of iPhone 5S. Sector level physical imaging was performed for all raw disk partitions.

XRY v6.11.1 has a 2 types of logical imaging support for Samsung GT-i9500 Galaxy S IV. Both Backup and Agent extractions were performed. Rooting was applied and Android SDK was used to acquire physical image of device. Significant raw disk partitions were tried to be acquired bit-to-bit and one by one. After the end of the image acquisition phase, analysis phase is started.

VI. Analysis

The last and detailed phase of this research is analysis phase. Many forensic tools were used to reveal user artifacts related to installed applications. Most of the tools are open source which means free of charge. Some of them are commercial and payment is required to use them with full functionalities.

The main tool is XRY v6.11.1 in this research. It helps for both acquisition and analysis. It decodes and parses the files to present in a comprehensible format.

Some of the file types could not be recognized but files could be exported by the help of XRY. Other tools were used to analyze them.

AccessData FTK Imager was used for mounting Android images and HFSExplorer was used for mounting iOS images. After mounting the images, files were able to previewed and exported with these tools. Image mounting tools were not enough for opening some specific files. It was necessary to try another tools for analyzation.

SQLite file type is used commonly in mobile devices. After image mounting and exporting the SQLite Database files, SQLite Database Browser can be used to open and analyze them. There is also a possibility to recover deleted contents from SQLite files. It depends on the deleted data management rules. The rules can be classified into three. First rule is deletion of data overwrites contents with zero(s). Recovery is impossible in first rule. The second rule is removing deleted area and it is troublesome to trace occurrence of deletion. The third rule is to set the data area as free and recovery is possible unless contents are overwritten. Web browsers generally use first rule while mobile applications generally use second and third rules (Jeon et al., 2012).

In iOS devices, property list (p-list) files are also commonly used for storing user settings and some information about applications. Plist editors can be used to open and analyze these files.

R-Studio and foremost file carving tools were also used in this research. These programs can be used for various image types and various file systems. Physical images of devices include unallocated spaces. Some deleted contents could be restored from these areas unless overwritten by other data. The main point of carving is to know headers and footers of requested files.

WinHex and some commands like “xxd” were used for low level analysis. Some data contents could not be carved by carving tools and string search might be useful. Data from scenarios would be searched to look if it was available in the images. Because of encoding and encryption mechanisms, this method may be useless in some situations.

§4. EXAMINATION AND ANALYSIS

This section covers detailed information about rooting and jailbreaking operations, scenario creations, acquisition techniques and analyzation methods with various forensic tools. Determined artifacts are also presented in detail.

I. iOS Device Forensics

iPhone 5S A1457 model with 32 GB capacity and iOS 8.1 operating system is the sample device in this research. Fundamentals of iOS forensics have been covered briefly and detailed technical analysis about each social network application will be performed in this section. Jailbreaking, scenario creation and acquisition techniques will also be explained before analysis.

A- Jailbreaking

Method of jailbreak for iOS devices may change for each device model and for each iOS version. Pangu software is the best known tool to jailbreak the iPhone 5s with iOS 8.1. Pangu version 1.2.1 also includes Cydia as well. Cydia enables to install software packages unavailable on the App Store.

Before jailbreaking, we turned off any lock screen passcode as well as Find My iPhone feature. Then, we turned on Airplane mode. We updated the iTunes version installed before. After downloading the version 1.2.1, we run the program as Administrator. While Pangu software is running, we connect the iPhone to our forensic workstation using Lightning to USB cable. After driver installation and detection the phone by Pangu, we click the “Start Jailbreak” button in the center. The device rebooted and “iPhone 5s (Global) with iOS 8.1 (12B411), Jailbroken” information appears on the screen as shown in Figure 4. This means that jailbreaking is succeeded.

Figure 4 - Pangu software screen after iOS Jailbreak

After jailbreaking, openSSH is installed to connect to device from computer via Wi-Fi. This will help to image raw disk partitions.

B- Scenarios

After jailbreaking operation and openSSH installation, the device is ready to install social network apps. Last versions of all six apps are downloaded from App Store successfully. Apple ID E-mail is used for account creation in each platform. Different passwords are chosen and profiles are updated with many personal details. Most of the personal information is chosen imaginary. After profile update, common user activities such as photo sharing, location sharing, sending a text message and adding a friend are performed.

C- Acquisition

Both logical and physical acquisitions of iOS device are performed before and after application installations. XRY v6.11.1 is the commercial software used for logical acquisition. It cannot still perform physical acquisition for our sample iOS device. After connecting the device to the forensic workstation, XRY can detect the device type and prompts to choose the right model. After choosing iPhone 5S (A1457), predesigned specific page for this device appears as shown in Figure

5. This page gives information about connection type, possible data types to acquire and some other exceptional information.

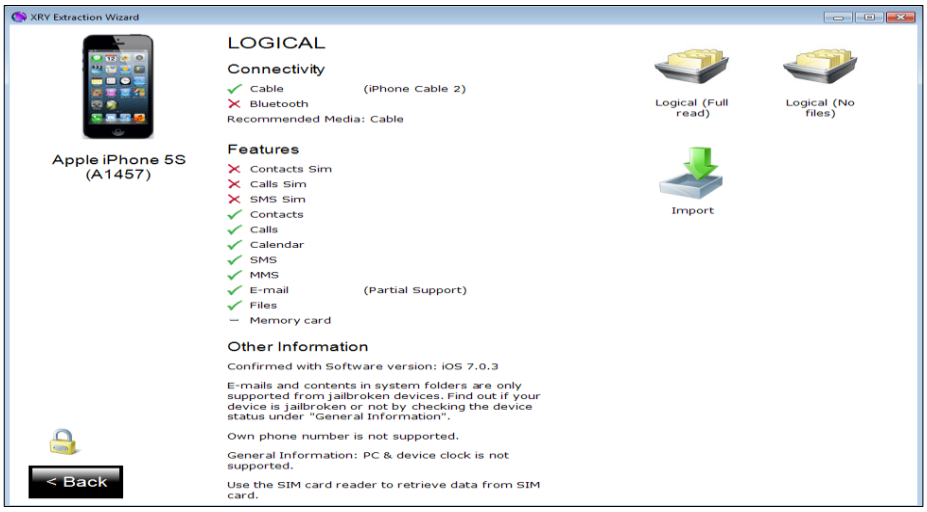


Figure 5 - XRY v6.11.1 sample screen for iPhone 5S

After clicking “Logical (Full read) button”, XRY asks for image file name and directory to save the image. Then, acquisition starts and log messages appear on the screen. “Logical extraction finished successfully!” message appears when XRY finishes the imaging. We have to click “continue” and “finish” buttons respectively for decoding the acquired data. Finally, “Image Decoding finished successfully” message appears and this shows that logical acquisition phase is finished. The image is ready to analyze now.

SIFT 2.13 Linux Workstation is used for physical imaging of the sample iOS device. After jailbreaking and openSSH installation on the phone, we changed the default SSH password. Then, the phone and forensic workstation placed on the same wireless network. “ssh root@ip_address” command is used for connection to the İOS device as a root user via SHH server. After entering password, “Joes-iPhone:~ root#” expression appears on the screen as shown in Figure 6. This expression means that SSH connection is successful. Joe is the name of the device which was given before. “ls –l /dev/rdisk*” command is used for learning raw disk partitions on the iPhone. Partitions are as follows:

- rdisk0 is the entire file system,
- rdisk0s1 is the firmware partition,

- rdisk0s1s1 is the root file system,
- rdisk0s1s2 is the user file system.

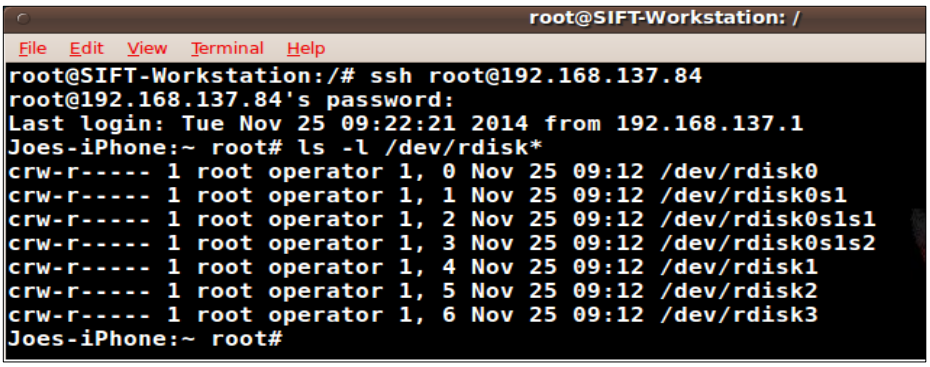


Figure 6 - Connection to the SHH Server on iOS device

File system in iOS device contains two logical disk partitions. These are system partition and user data partition. System partition has a smaller size than user data partition and it includes little information related to this research. On the contrary, user data partition includes many information about installed applications and other user activities. “mount” command shows the mounted partitions on iOS device. System partition is mounted on / (root) and user data partition is mounted on /private/var as shown in Figure 7.

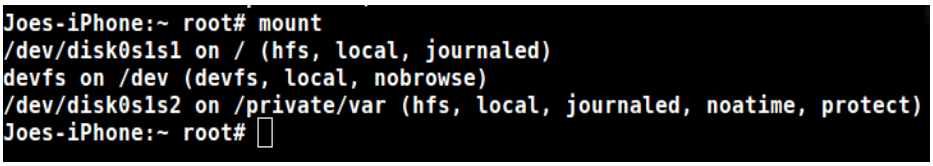


Figure 7 - Mounted partitions on İOS device

It is now clear that imaging and analyzing the user data partition (rdisk0s1s2) is the primary goal for iOS device in this research. It is also better to image and analyze system partition (rdisk0s1s1) in case of evidentiary data.

“ssh root@ip_adress “ dd if=/dev/raw_disk_partititon bs=block_size” > imageName.dmg” command helps to image raw disk partitions as shown in Figure 8. When imaging finishes, size of the image with elapsed time and imaging speed appear on the screen. Size of the user data partition is 29 GB and the system partition is 3 GB in this research. This information shows that physical imaging is

completed successfully. rdisk0s1s2.dmg and rdisk0s1s1.dmg image files are ready to analyze.

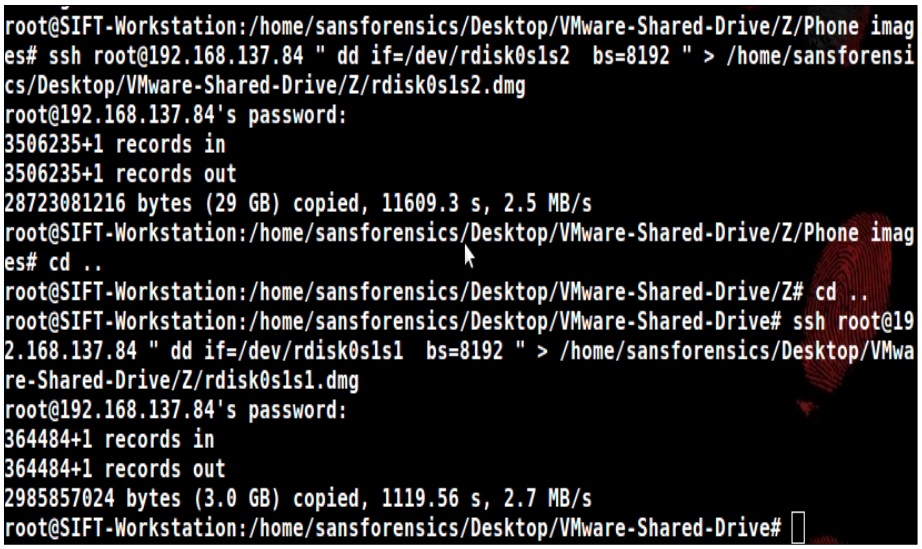


Figure 8 - Imaging raw disk partitions of iOS device

D- Analysis

After creating logical and physical images of iPhone 5S, analyzation is started. Logical image is created and analyzed with XRY v6.11.1. In documents section of XRY output, there are many xml, log and plist files related to social network applications. In databases section, many SQLite database files are also created by social network applications. These type of files are exported and opened one by one with specific file editors. File contents are compared with the scenarios and artifacts of each platform are determined.

HFSExplorer is used to open physical images of iPhone 5S. After opening the images, all files are extracted to computer as shown in Figure 9. Many SQLite, plist, xml and log files are found. File names and directories are unencrypted but file contents are encrypted in user data partition. Consequently, physically retrieved files and unallocated spaces of iOS device could not be analyzed.

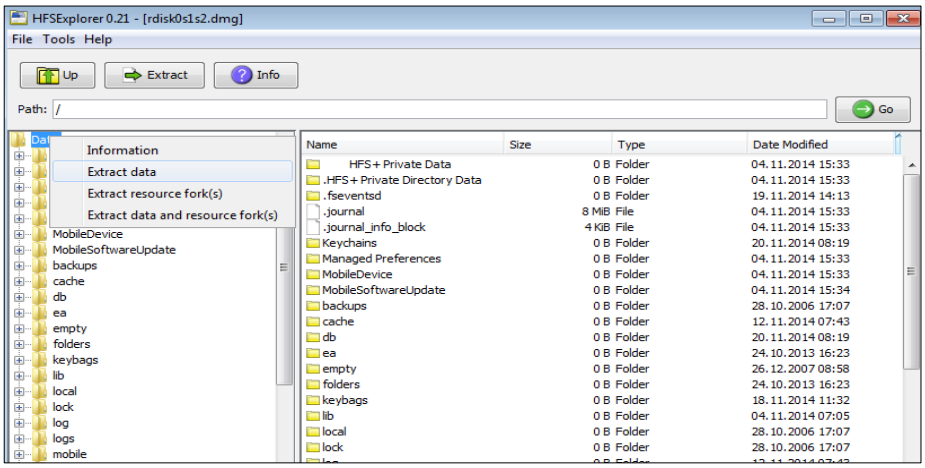


Figure 9 - HFSExplorer file extraction process from iPhone 5S images

1. Facebook Artifacts

Forensic artifacts related to Facebook application are located into Data\mobile\Containers\Data\Application\F1C60E56-6356-46AA-AFA9-9E4513A76EE8 directory in physical image of rdisk0s1s2 partition. Mainly Orca2.db, Fbsyncstore.db and 100004158494721.session.plist files store lots of information consistent with the scenarios. Some other files also contain little evidentiary data. Many directories contain image files.

a) Database Files

Orca2.db file is in 8.1_iphone_en_en_TR_messenger_messages_v1 folder. It has 9 tables and stores friends' names, Facebook IDs, chat messages with timestamps and geographic coordinates, profile picture URLs and threads as shown in Figure 10.

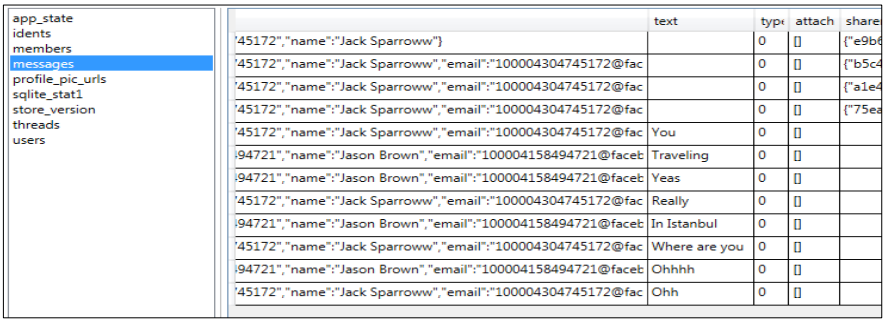


Figure 10 - Orca2.db file contains Facebook chat messages with time stamps

Fbsyncstore.db file is in 8.1_iphone_en_en_TR_messenger_contacts_v1 folder. It has 8 tables and stores Emails, phone numbers, friends' names, searched people, Facebook IDs and profile picture URLs as shown in Figure 11.

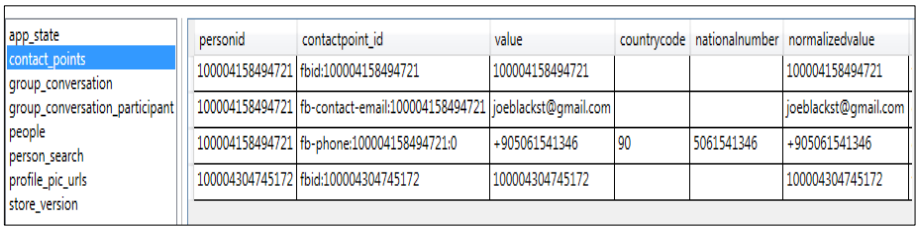


Figure 11 - Fbsyncstore.db file contains Facebook friend list

Cache.db file is in com.facebook.Facebook folder and it contains URLs generally for pictures and profile images.

There is also Store.sqlite file in 8.1_iphone_en_en_TR_default_store_ 94d4e3488c4b9ffff2844da4b3883dbe95e24e08_v1/FBStore directory and it has many tables. But evidentiary data could not be found in it.

b) Plist Files

100004158494721 is user's Facebook ID. 100004158494721.session.plist file stores profile information like high school, work, education and city as shown in Figure 12. It also stores longitude and latitude of some shared locations.

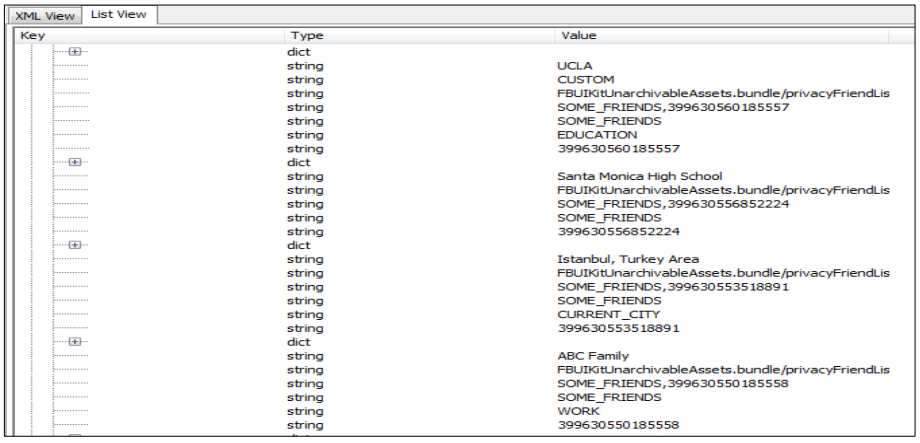


Figure 12 - 100004158494721.session.plist file contains Facebook profile information

Some more plist files also store preferences about Facebook application and some of them contain data also available in 100004158494721.session.plist file. These plist files are;

- 100004158494721.plist
- com.facebook.Facebook.plist
- com.facebook.Messenger.plist
- group.com.facebook.Messenger.plist
- group.com.facebook.Facebook.plist
- _composer_file_cache_mem_model_composition_bundle_cache.plist
- FBMapViewSnapshotCreator-98740b94d3f24a04f0c4d31a2ab755b4-v1.plist

c) Multimedia Files

Many pictures are found as a multimedia file. They are located in different directories. These directories are;

- Library\Caches_store_0961BE02-2538-431E-A7AB-B8197842AA68\8.1_iphone_en_en_TR_image_cache_v2\FBDiskCache_Image directory contains various image files.
- Library\Caches\com.facebook.Facebook\fsCachedData directory also contains image files.
- Library\Caches\ImageCache directory also contains some pictures.
- Library\Caches\Snapshots\com.facebook.Facebook\com.facebook.Facebook directory contains snapshot pictures about Facebook application

d) Deleted Artifacts

Some of the messages, posts and pictures were deleted from Facebook application. Some deleted artifacts were found in Database files. Database files are opened with WinHex program and deleted contents are searched manually.

Figure 13 shows a sample deleted text message of Facebook application. A text message with various information such as sender and receiver IDs, sender and

receiver names and time stamp are retrieved. This shows that the third deleted data management rule was applied to database file. A profile of a user can be found easily on the internet with the help of User ID. Time stamp is in Unix Epoch time format[24]. Epoch time converter can be used to determine the time value[25]. Decimal value of "14A380B0A7C" is "1418279455356" and it indicates 11 Dec 2014 06:30:55 UTC time.

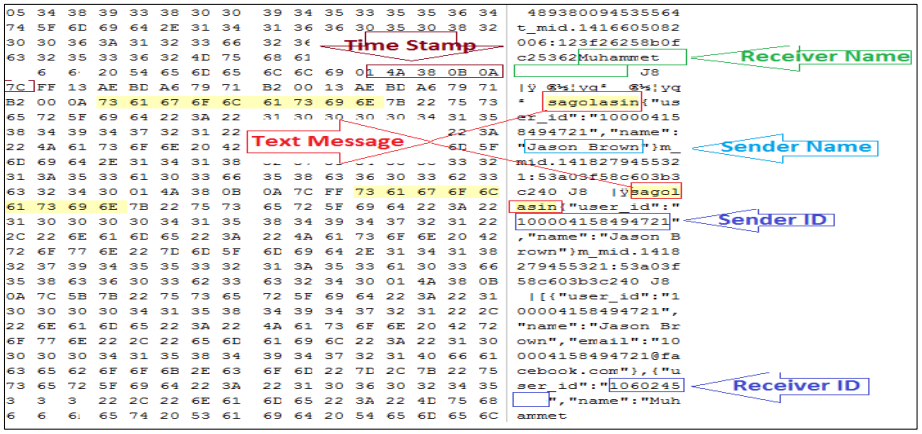


Figure 13 - Sample deleted Facebook message with metadata information retrieved from orca2.db file

Some of the messages were also containing location information. The location information of a deleted text message also retrieved as shown in Figure 14. Message status (read or unread) and message source (mobile, web and so on) are also still available in this message.

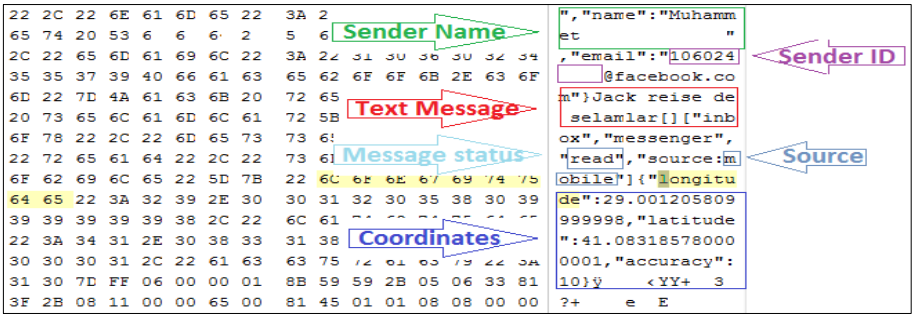


Figure 14 - Sample deleted Facebook message with location information retrieved from orca2.db file

[24] The Unix epoch (or Unix time) is the number of seconds that have elapsed since January 1, 1970. Literally speaking the epoch is Unix time 0 (midnight 1/1/1970), but 'epoch' is often used as a synonym for 'Unix time'. Many Unix systems store epoch dates as a signed 32-bit integer.

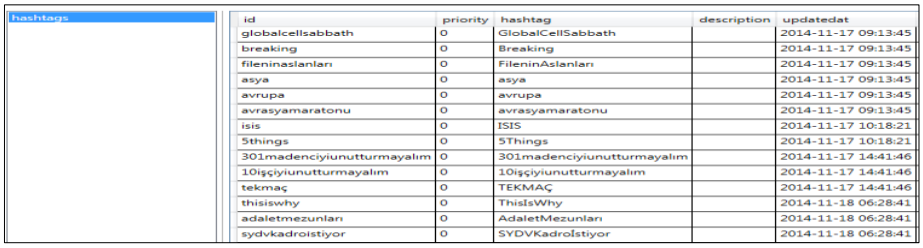
[25] Sample Epoch time converter is available at http://www.epochconverter.com/, accessed on December 13, 2014.

2. Twitter Artifacts

Twitter artifacts are located in Data\mobile\Containers\Data\Application\ 03FDB9E9-F653-42B7-A9B6-EFFE602C2CAD directory. autocomplete4.sqlite3, twitter.db and app.acct.JoeJoeblackst-437908224.detail.10.log files contain most of the evidentiary data. Some directories also contain image files.

a) Database Files

Autocomplete4.sqlite3 file contains only "hashtags" table and it contains hashtags with ID, priority, description and timestamp as shown in Figure 15.

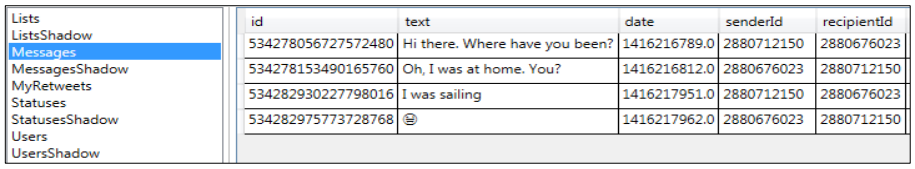


hashtags

id	priority	hashtag	description	updatedat
globalcellsabbath	0	GlobalCellSabbath		2014-11-17 09:13:45
breaking	0	Breaking		2014-11-17 09:13:45
filenınaslanları	0	FileninAslanları		2014-11-17 09:13:45
asya	0	asya		2014-11-17 09:13:45
avrupa	0	avrupa		2014-11-17 09:13:45
avrasyamaratonu	0	avrasyamaratonu		2014-11-17 09:13:45
isis	0	ISIS		2014-11-17 10:18:21
5things	0	5Things		2014-11-17 10:18:21
301madenciyiunutturmayalım	0	301madenciyiunutturmayalım		2014-11-17 14:41:46
10işçiyiunutturmayalım	0	10işçiyiunutturmayalım		2014-11-17 14:41:46
tekmaç	0	TEKMAÇ		2014-11-17 14:41:46
thisiswhy	0	ThisIsWhy		2014-11-18 06:28:41
adaletmezunları	0	AdaletMezunları		2014-11-18 06:28:41
sydvkadroistiyor	0	SYDVKadroİstiyor		2014-11-18 06:28:41

Figure 15 - Autocomplete4.sqlite3 file contains Twitter hashtags

Twitter.db file contains text messages with timestamps and user IDs, retweets, retweet counts, following accounts, status messages, location information, URLs and descriptions as shown in Figure 16.



Lists
ListsShadow
Messages
MessagesShadow
MyRetweets
Statuses
StatusesShadow
Users
UsersShadow

id	text	date	senderId	recipientId
534278056727572480	Hi there. Where have you been?	1416216789.0	2880712150	2880676023
534278153490165760	Oh, I was at home. You?	1416216812.0	2880676023	2880712150
534282930227798016	I was sailing	1416217951.0	2880712150	2880676023
534282975773728768	😀	1416217962.0	2880676023	2880712150

Figure 16 - Twitter.db file contains Twitter direct text messages

Cache.db file contains URLs for pictures and profile images.

Scribe.1.sqlite file contains user ID, time stamps and some more information about phone. Time values in scribe.1.sqlite file are in Mach Absolute Time format[26].

[26] Mac absolute time is the number of seconds that offsets the Mac epoch time, which starts on January 1, 2001. The difference between the Unix epoch time and the Mac epoch time is exactly 978,307,200 seconds. To convert the Unix epoch time to Mac absolute time, add 978,307,200 to it and calculate it as a Unix timestamp.

b) Plist Files

JoeJoeblackst is twitter user name and 437908224 is Mac Absolute Time for file creation. App.acct.JoeJoeblackst-437908224.detail.10.log file contains text messages, profile information and URLs as shown in Figure 17.

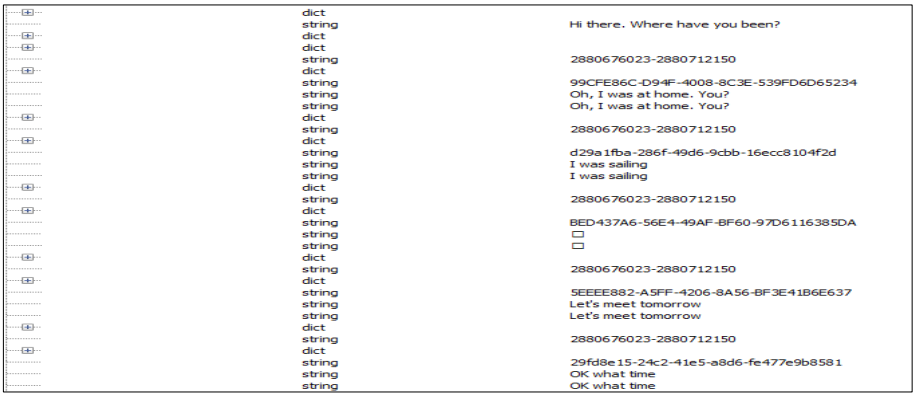


Figure 17 - App.acct.JoeJoeblackst-437908224.detail.10.log file contains Twitter profile information

Other plist files contain preferences, profile information, URLs, time stamps and some more evidentiary information. These files are;

- com.atebits.Tweetie2.plist
- group.com.atebits.Tweetie2.plist
- settings.plist

c) Multimedia Files

Twitter application creates many folders and directories for multimedia files. These are;

- Library\Caches\com.atebits.tweetie.profile-images\ directory contains profile pictures of twitter accounts.
- Library\Caches\com.atebits.Tweetie2\fsCachedData directory contains various image files
- Library\Caches\Snapshots\com.atebits.Tweetie2\com.atebits.Tweetie2 directory contains snapshot pictures about twitter application.
- Tmp file contains some more pictures related to twitter application.

d) Deleted Artifacts

Some of the direct messages and twitter posts were deleted and tried to recover. Deleted direct messages are searched in database file but they cannot be retrieved. The file size is also reduced. It is seen that the second deleted data management rule was applied to twitter.db SQLite file. The vacuum[27] command might be used for secure deletion.

Deleted twitter posts with various information can be retrieved from logical phone image. Deleted twitter posts are still available more than one location in an image. This shows that the third deleted data management rule was applied to database file.

Figure 18 shows the sample deleted tweets. Three twitter posts were deleted but they are still available. Time stamp shows the creation time of the tweet and Unix Epoch Time format is used like Facebook application. Decimal value of "5469C6C5" is "1416218309" and it indicates 17 Nov 2014 09:58:29 UTC time. User ID is "ABB3A0B7" and its decimal value is 2880676023. Shared location name and related coordinate information are also available.

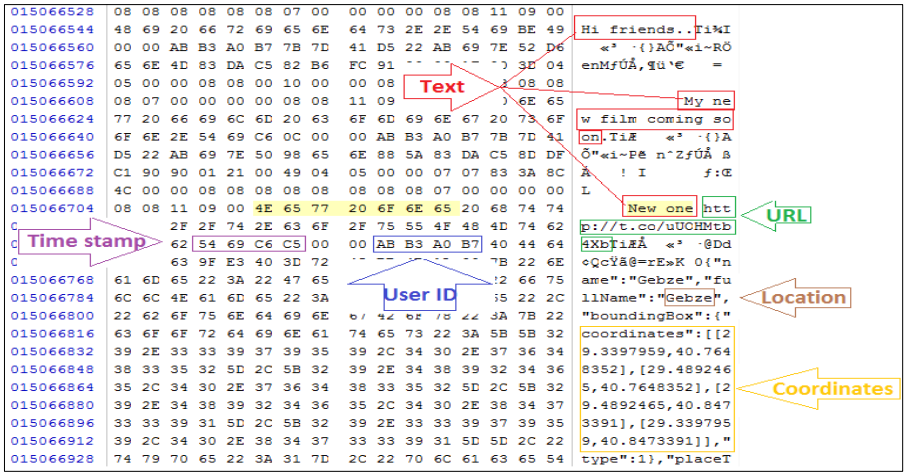


Figure 18 - Sample posted tweet format with metadata information for Twitter application

Many twitter accounts were followed and some of them removed from the list. Many twitter posts were found related to these accounts.

[27] Vacuum command rebuilds the entire database, removing the space in the free-list and shrinking the database.

3. Google+ Artifacts

Google+ artifacts are stored in Data\mobile\Containers\Data\Application\ B1D02A66-54B3-43C2-8A52-D0EABCB11CBC directory. Profile.plist and com.google.PlusCore.PersonCacheCollection.111613886622229336085.plist files contain most of the critical data. Multimedia files are stored in specific directories.

a) Plist Files

Profile.plist file contains profile information such as name, E-mail, gender, picture URL and unique ID as shown in Figure 19.

Figure 19 - Profile.plist file contains Google+ profile information

111613886622229336085 is the unique user ID number for this user. Com.google.PlusCore.PersonCacheCollection.111613886622229336085.plist file contains various profile information about user and user's friends as shown in Figure 20.

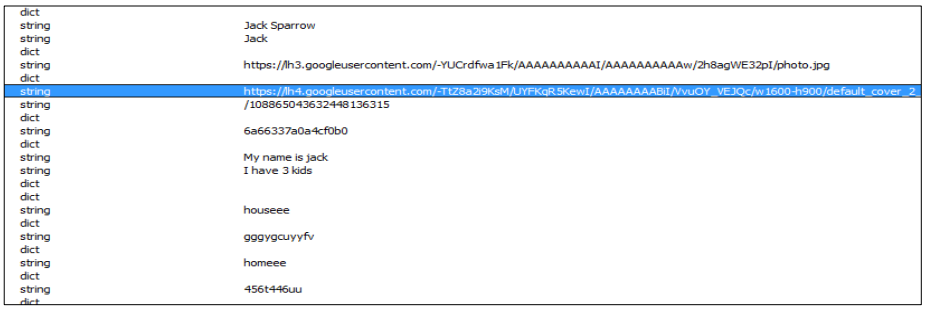


Figure 20 - Com.google.PlusCore.PersonCacheCollection.111613886622229336085.plist file contains Google+ profile information about user and friends

b) Multimedia Files

Library\Caches\com.google.GooglePlus\fsCachedData directory contains various pictures.

Library\Caches\Snapshots\com.google.GooglePlus\com.google.GooglePlus directory contains snapshot pictures of Google+ application.

c) Deleted Artifacts

Profile update and deletion of some posts were performed to test the deleted contents situation. Some of the old profile entries such as E-mail can be found but some of them cannot. Moreover, deleted posts cannot be found in logical image. Most of the previous research also show that deleted content recovery is limited for logical images. If data was written in more than one location in memory, delete operation would not delete all of them. There is a greater possibility to recover deleted contents in this situation.

4. Instagram Artifacts

Instagram artifacts are stored in Data\mobile\Containers\ Data\Application\56AFDA11-30FF-4841-BCE2-189A9B0EE69F directory. Lastentries.coded.log and recent-users.coded.log files store critical data.

a) Plist Files

Some plist files do not have a .plist extension. Some of them have different extensions. Log files for Instagram application are in plist file format.

Lastentries.coded.log file contains profile information and incoming posts with picture urls as shown in Figure 21.

string	https://igcdn-photos-e-a.akamaihd.net/hphotos-ak-xap1/10809669_730694583678804_387207101_a.jp
string	joeblackst
string	Joe Joe Black
string	I am hero
dict	
string	http://www.jhvgfscgeghv.com
dict	
integer	1416231572000000
real	1416231567

Figure 21 - Lastentries.coded.log file contains Instagram profile information and incoming posts

Recent-users.coded.log file contains various profile information and following people as shown in Figure 22.

Figure 22 - Recent-users.coded.log file contains Instagram profile information of friends

b) Multimedia Files

Library\Caches\com.burbn.instagram.IGCache directory contains many binary plist files each of them include URLs for pictures.

Library\Caches\Snapshots\com.burbn.instagram\com.burbn.instagram directory contains snapshot pictures of Instagram application.

c) Deleted Artifacts

Instagram application do not use SQLite database files in iOS. Log, plist and xml files store evidentiary contents. These files do not contain deleted data. Consequently, deleted artifacts and old profile information cannot be found in logical image of the phone.

5. WhatsApp Artifacts

WhatsApp artifacts are stored in Data\mobile\Containers\Data\Application\ 2A7CA593-D6CF-44E4-8F8B-A24D13893C42 directory. Documents and Library folders mainly contain evidentiary data.

a) Database Files

Database files are in Documents folder. ChatStorage.sqlite file contains user names, phone numbers, text messages, time stamps, unique IDs and used words list as shown in Figure 23.

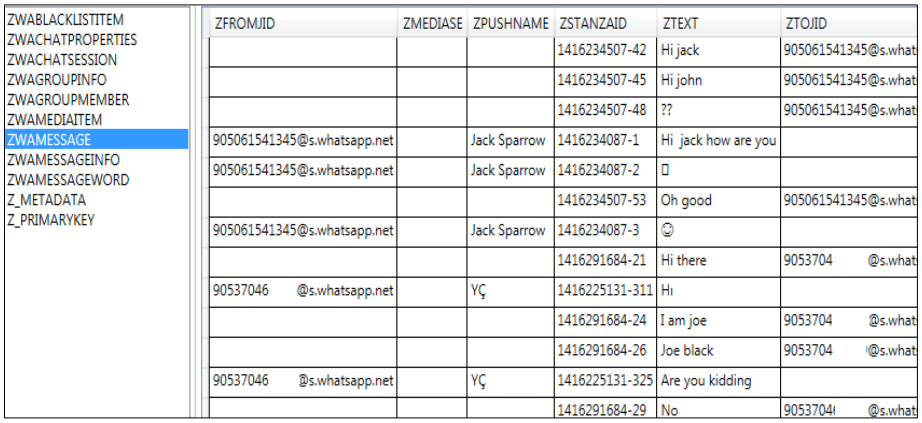


Figure 23 - ChatStorage.sqlite file contains WhatsApp text messages

Contacts.sqlite file contains user names, phone numbers, time stamps and status messages as shown in Figure 24.

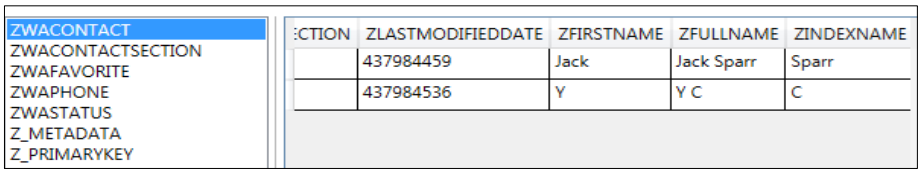


Figure 24 - Contacts.sqlite file contains WhatsApp contact list

ChatSearch.sqlite file contains chat messages, phone numbers and time stamps as shown in Figure 25.

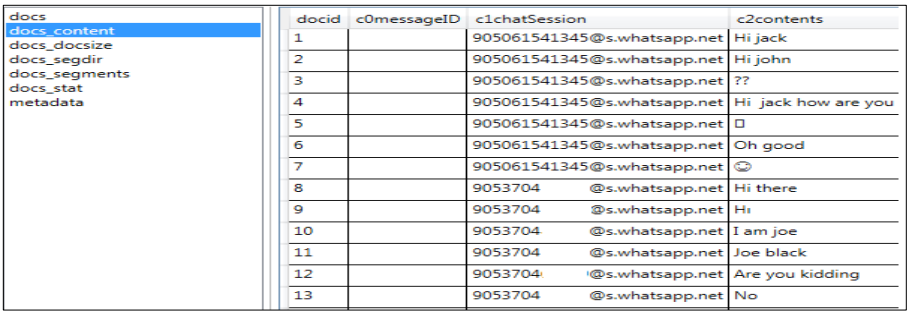


Figure 25 - ChatSearch.sqlite file contains WhatsApp text messages

Cache.db file is in Library\Caches\net.whatsapp.WhatsApp directory and it contains time stamps, logs and some more evidentiary information.

b) Plist Files

Net.whatsapp.WhatsApp.plist file contains user name, status message, login count, payment information and number of received and sent messages as shown in Figure 26.

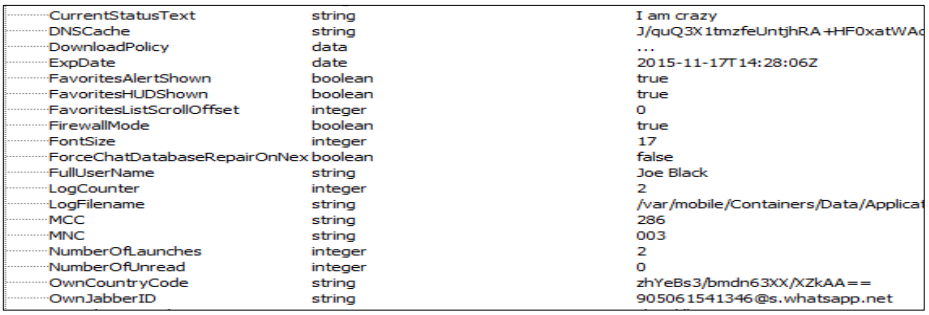


Figure 26 - Net.whatsapp.WhatsApp.plist file contains WhatsApp profile information

StatusList.plist, SyncHistory.plist, net.whatsapp.WhatsApp.plist and calls.log files also contain some evidentiary data.

c) Log Files

Library\Logs directory contains two log files; whatsapp-2014-11-17-16-27-09.763.1.log file and whatsapp-2014-11-18-08-21-22.763.2.log file. They are not in plist file format. They contain time stamps for messages, phone numbers, IP addresses, device information, carrier name and some more data.

d) Multimedia Files

Following directories include many pictures related to WhatsApp application;

- Library\Caches
- Library\Caches\Chat
- Library\Caches\net.whatsapp.WhatsApp\fsCachedData
- Library\Caches\ProfilePictures

- Library\Caches\Snapshots\net.whatsapp.WhatsApp\net.whatsapp.WhatsApp
- Library\Media

e) Deleted Artifacts

Some of the contents related to WhatsApp application were deleted and tried to retrieve with low level methods. ChatStorage.sqlite and ChatSeach.sqlite files still contain deleted contents such as chat messages and location sharing messages. This shows that the third deleted data management rule was applied to database files.

In Figure 27, ChatSeach.sqlite file is opened with WinHex and its content is compared with the WhatsApp screenshot of the phone. Text messages and phone numbers are still available in the file. In some situations, time stamps of the text messages may be available near the messages. The hexadecimal value of the time of the text message may be recoverable.

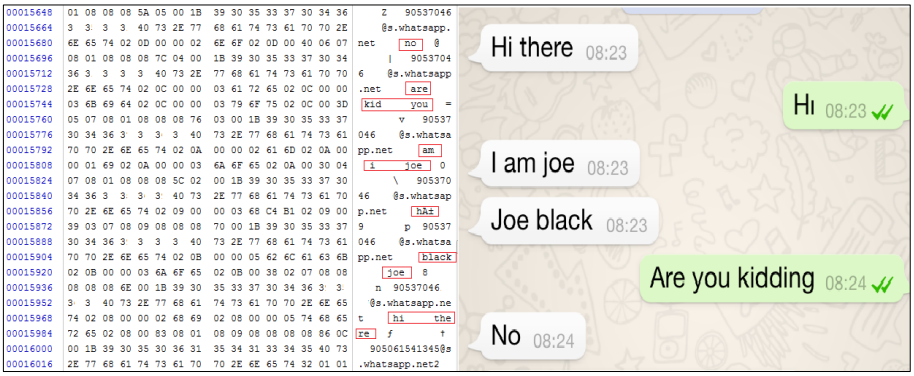


Figure 27 - Comparison of WhatsApp deleted messages in ChatSeach.sqlite file

Some of the words of the text messages also found more than one time in SQLite database files. They have no order and both deleted and undeleted words are together as shown in Figure 28.

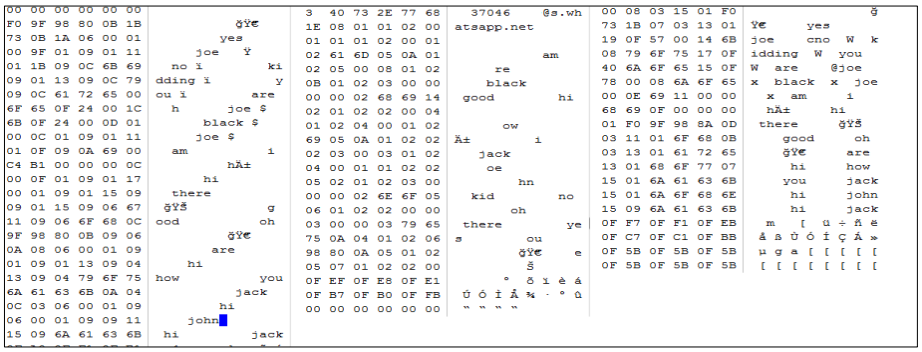


Figure 28 - Found WhatsApp message words in database files

In Figure 29, ChatStorage.sqlite file is opened with WinHex and compared with the WhatsApp screenshot of the deleted location. Explanation of shared location is still available in the file.

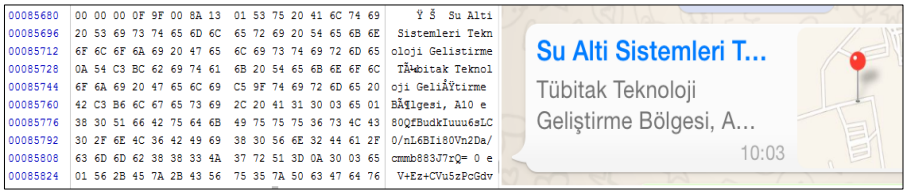


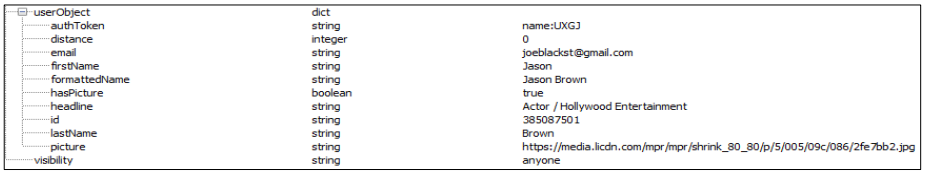
Figure 29 - Comparison of deleted WhatsApp location information in ChatStorage.sqlite file

6. LinkedIn Artifacts

LinkedIn artifacts are stored in Data\mobile\Containers\Data\Application\54065111-DA1E-49D8-93D3-D622C366E47F directory. cacheInfo.plist, liv2profile385087501.xml, notificaions_data_center_key.xml and com.linkedin.LinkedIn.plist files include critical data.

a) Plist Files

Com.linkedin.LinkedIn.plist file contains basic profile information such as name, E-mail, headline, picture URL and user ID number as shown in Figure 30.



userObject	dict	
authToken	string	name:UXGJ
distance	integer	0
email	string	joeblackst@gmail.com
firstName	string	Jason
formattedName	string	Jason Brown
hasPicture	boolean	true
headline	string	Actor / Hollywood Entertainment
id	string	385087501
lastName	string	Brown
picture	string	https://media.licdn.com/mpr/mpr/shrink_80_80/p/5/005/09c/086/2fe7bb2.jpg
visibility	string	anyone

Figure 30 - Com.linkedin.LinkedIn.plist file contains LinkedIn profile information

CacheInfo.plist file contains file names that contain accessed URLs, file sizes and time stamps as shown in Figure 31.

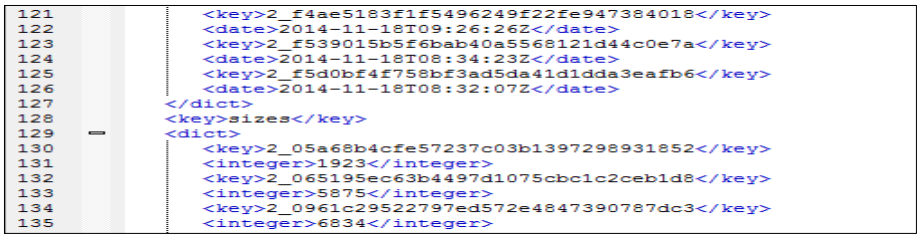


Figure 31 - CacheInfo.plist file contains LinkedIn file names associated with URLs

b) Xml Files

Liv2profile385087501.xml file contains user ID and profile information such as name, location, E-mail, phone number, address, education, job, website and picture URL as shown in Figure 32.

Figure 32 - Liv2profile385087501.xml file contains LinkedIn profile information

Notificaions_data_center_key.xml file contains information about who viewed the profile as shown in Figure 33.

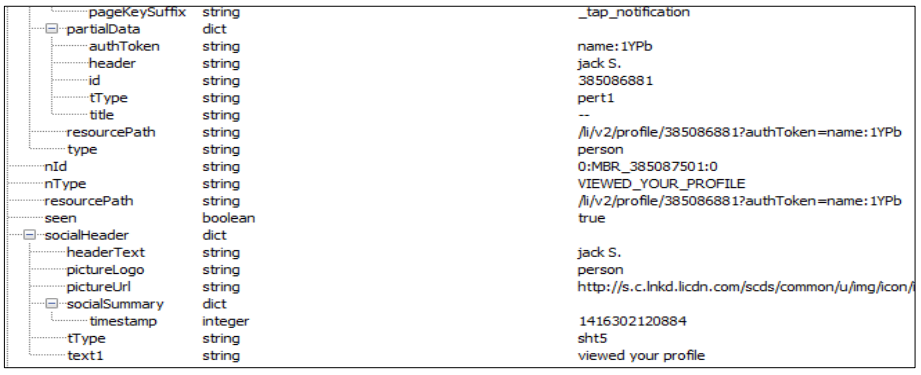


Figure 33 - Notificaions_data_center_key.xml file contains who viewed the LinkedIn profile

c) Deleted Artifacts

LinkedIn profile update was performed. Some information such as phone number, address, university, occupation and profile picture were changed and tried to recover old information. Only university and occupation can be found in the logical image. Some of the old profile information cannot be found.

7. Artifacts after Uninstallation of Applications

Applications are uninstalled from device and changes are observed in terms of user artifacts. Logical image of iOS device is analyzed solely after uninstallation of applications because of the fact that physical image of data partition is encrypted. It is determined that all files produced by installed applications are deleted and cannot be recovered. Some of the contents of applications were also duplicated in different locations for other applications in the device. Some pictures, sms messages internet history and account information related to deleted applications are retrieved from device. Actually, these are related to deleted applications but possessed of other preinstalled applications. In conclusion, it can be said that social media applications only leave artifacts which are duplicated and associated with other applications in iOS device.

II. Android Device Forensics

Samsung GT-i9500 Galaxy S IV with 16 GB capacity and Android 4.4.2 operating system is the sample device in this research. Fundamentals of Android forensics have been covered briefly and detailed technical analysis about social network artifacts will be performed in this section. Rooting, scenario creation and acquisition techniques will also be explained before analysis.

A- Rooting

Method of Rooting for Android devices may change for each device and each Android version. Odin3 v3.09 and CF-Root Package are used to root sample Android device in this research. At first step, "USB Debugging" option must be enabled under "Settings > Developer Options". Device is connected to the workstation and necessary drivers are installed. Odin3 v3.09 and CF-Root Package are downloaded from internet and unzipped. Next, the device is turned off and booted into Download Mode by pressing and holding the Volume Down + Home + Power buttons at the same time for a few seconds and released after a warning message on screen. Then, odin3 v3.09.exe is opened as an Administrator. When Odin detects the device, ID:COM box turns to blue and "Added!!" message appears as shown in Figure 34. After that, AP button is pressed and CF-Auto-Root-ja3g-ja3gxx-gti9500.tar.md5 file is browsed. Finally, Start button is pressed to begin installation. A couple of minutes later, the device rebooted and a pass message appears with green background. This shows that Rooting is succeeded.

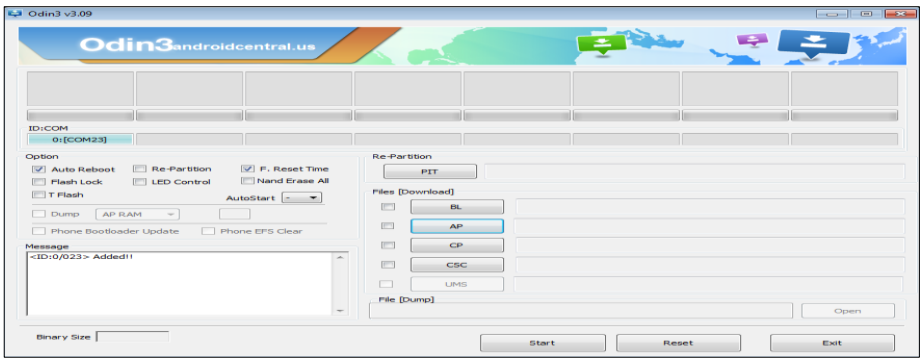


Figure 34 - Rooting Android device with Odin3 v3.09

B- Scenarios

After rooting operation, Android device is ready to install social network apps. Last versions of all six apps are downloaded from Google Play Store successfully. The same E-mail is used for account creation in each platform. Different passwords are chosen and profiles are updated with many personal details. Most of the personal information is chosen imaginary. After profile update, common user activities such as photo sharing, location sharing, sending a text message and adding a friend are performed.

C- Acquisition

Both logical and physical acquisitions of Android device are performed before and after application installations. XRY v6.11.1 is the commercial software used for logical acquisition. It cannot still perform physical acquisition for our sample Android device. After connecting the device to the forensic workstation, XRY can detect the device. After choosing Generic ADB option instead of Generic AT, predesigned specific page for this device appears as shown in Figure 35. This page gives information about connection type, possible data types to acquire and some other exceptional information.

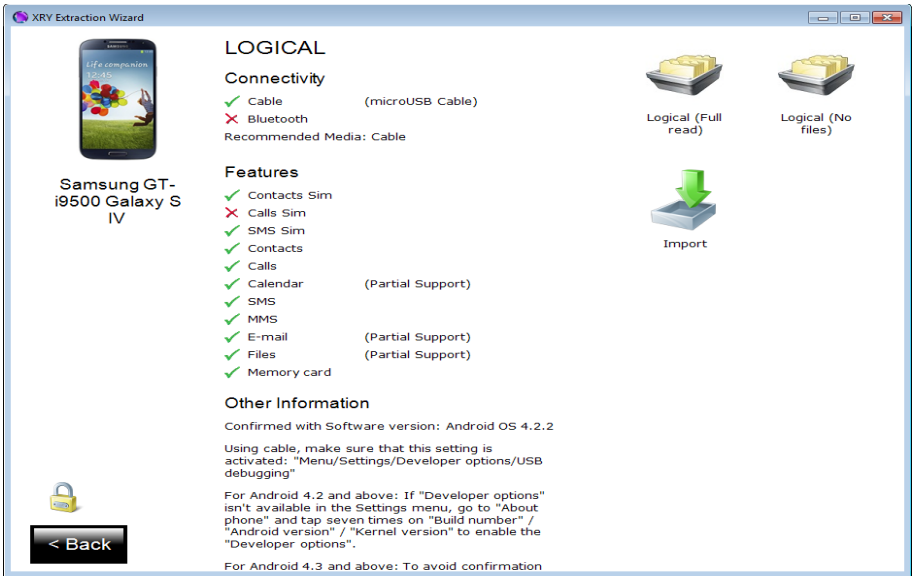


Figure 35 - XRY v6.11.1 Sample screen for Samsung GT-i9500 Galaxy S IV

After clicking “Logical (Full read) button”, XRY asks for image file name and directory to save the image. Then, a message screen appears to choose “Backup” or “Agent” extraction methods. After clicking “Backup” option, a message appears on the phone. After clicking “Back up my data” on the phone, acquisition starts a couple of seconds later and log messages appear on the screen of forensic computer. “Logical extraction finished successfully!” message appears when XRY finishes the imaging. We have to click “continue” and “finish” buttons respectively for decoding the acquired data. Finally, “Image Decoding finished successfully” message appears and this shows that logical acquisition phase is finished.

The same steps are applied for Agent extraction method. Backup method is for retrieving data from third party apps and agent method is for getting system app data. Two logical images are ready to analyze now.

Platform tools in Android SDK is used for physical imaging of the sample Android device. After rooting the device, “adb.exe devices” command is used for looking the list of attached devices. Then,”adb.exe shell” and “su” commands are used for root access to the phone as shown in Figure 36.

“cat /system/build.prop” command can be used to see some information about device such as device brand, model, manufacturer and Android version as shown in Figure 36.

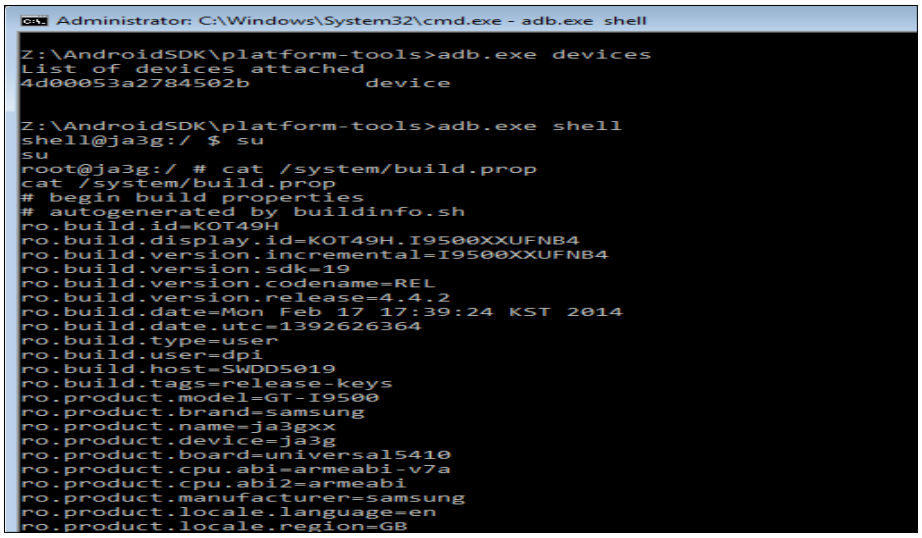


Figure 36 - ADB connection to the Android device

"mount" command is used to see mounted partitions in device as shown in Figure 37. Imaging and analyzing the following partitions is enough for investigating application artifacts:

- "/system" partition which is mounted on block mmcblk0p20 includes system related files,
- "/data" partition which is mounted on block mmcblk0p21 includes user data related to applications,
- "/cache" partition which is mounted on block mmcblk0p19 includes frequently accessed data,
- "/efs" partition which is mounted on block mmcblk0p3 includes some sensitive data.

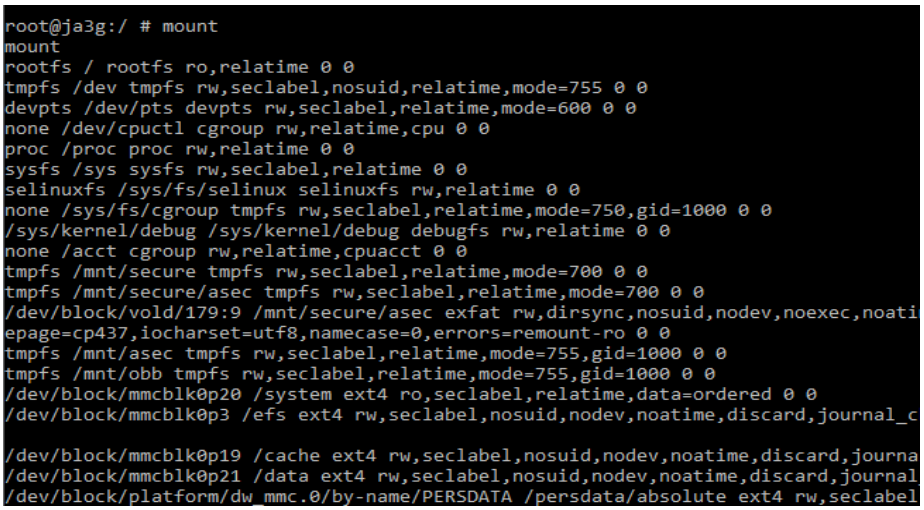


Figure 37 - Mounted partitions in Android device

Raw image of each partition is stored in external Micro SD Card in device and "adb pull" command is used to transfer images to the forensics workstation. "dd if=/dev/block/mmcblk0p21 of=/storage/extSdCard/data_part_image.dd" command is used to image data partition on block mmcblk0p21 as shown in Figure 38. "adb.exe pull /storage/extSdCard/data_part_image" command is used to transfer image file to the computer. After each process, size of the partition with elapsed time and imaging speed appear on the screen. This is an indication of successful operation. Physical imaging of system, cache and efs partitions are also performed with same methods and tools.

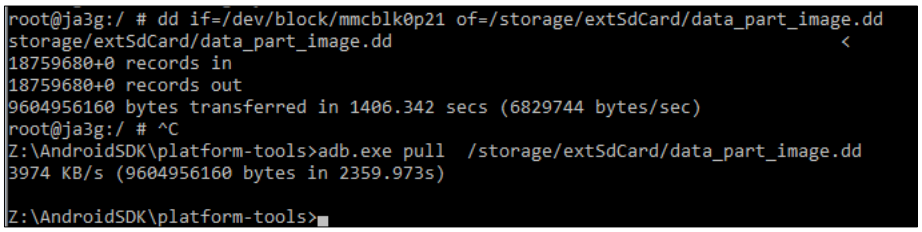


Figure 38 - Imaging raw disk partitions of Android device

D- Analysis

After creating logical and physical images of Samsung GT-i9500 Galaxy S IV, analyzation is started. Logical images are created and analyzed with XRY v6.11.1. In documents and databases sections of XRY output, there are many files with various extensions related to social network applications. All files are exported and tried to open one by one with specific file editors. File contents are compared with the scenarios and artifacts of each platform are determined.

FTK Imager is used to open physical images of Samsung GT-i9500 Galaxy S IV. After opening the images, all files are extracted to computer as shown in Figure 39. Many xml, database, log and multimedia files are found. Each file is opened manually and file contents are compared with the scenarios. File contents are not encrypted. Unallocated spaces and physically retrieved files can be analyzed.

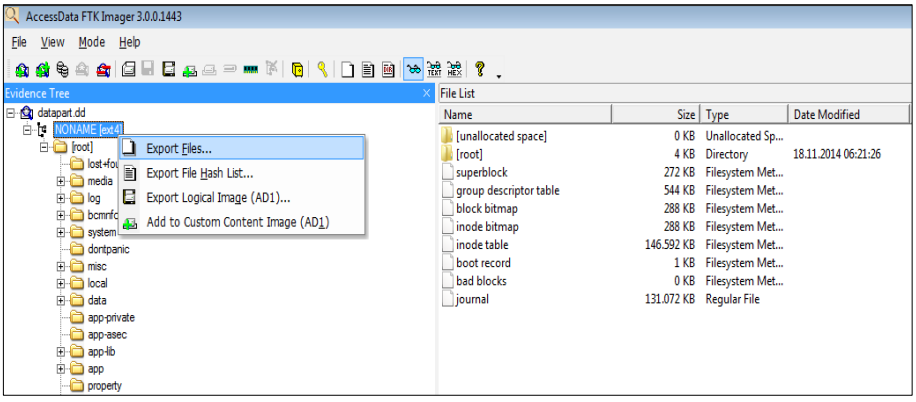


Figure 39 - FTK Imager file extraction process for Android images

Files are opened with file editors and their contents are analyzed. Manuel string search method was used for deleted contents. Strings are searched in files and

raw image of data partition. R-studio and foremost tools are used to recover photos with different extensions.

1. Facebook Artifacts

User artifacts related to Facebook application are stored in data\com.facebook.katana\ and data\com.facebook.orca\ directories. Each directory contains many subdirectories. Important subdirectories are databases, cache and files.

Databases folder contains 66 database files. Most important files are contacts_db2 and threads_db2. Cookies.db and Notifications_db files also contain some evidentiary data. Cache folder contains audio and image folders. Files folder contains video-cache folder which includes video files without any extension.

a) Database Files

Contacts_db2 file contains number of contacts, contact IDs, phone numbers, names and surnames, picture URLs and some more information about Facebook contacts as shown in Figure 40.

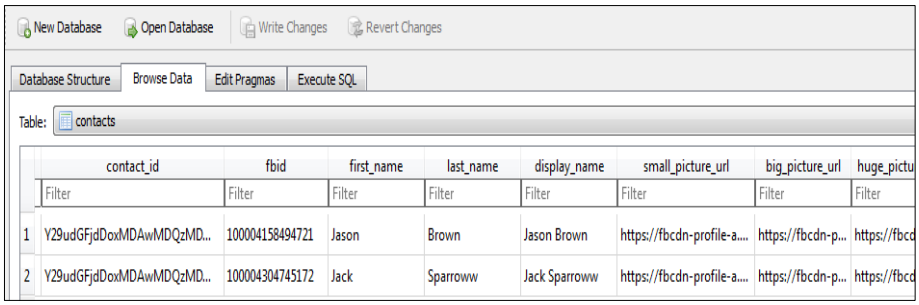
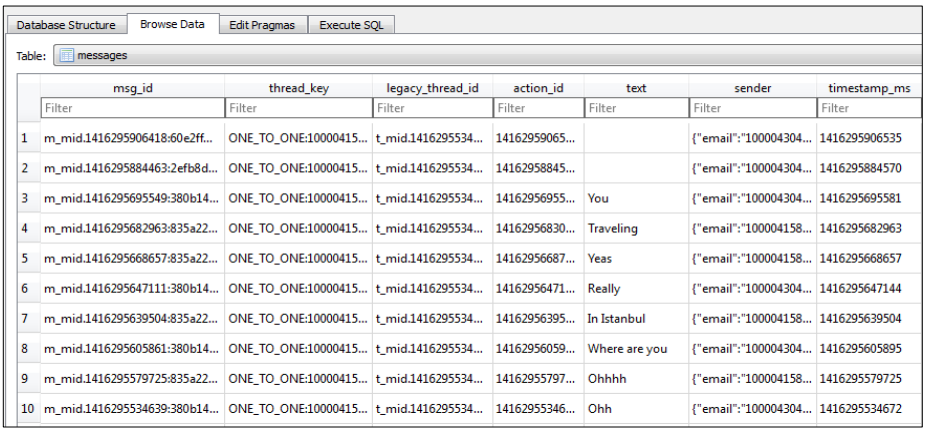


Figure 40 - Contacts_db2 file contains Facebook contacts

Threads_db2 file contains chat messages with time stamps, group conversations, various unique IDs, phone numbers, coordinates and last seen time as shown in Figure 41.



Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: messages

	msg_id	thread_key	legacy_thread_id	action_id	text	sender	timestamp_ms
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	m_mid.1416295906418:60e2ff...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162959065...		{"email":"100004304...	1416295906535
2	m_mid.1416295884463:2efb8d...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162958845...		{"email":"100004304...	1416295884570
3	m_mid.1416295695549:380b14...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162956955...	You	{"email":"100004304...	1416295695581
4	m_mid.1416295682963:835a22...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162956830...	Traveling	{"email":"100004158...	1416295682963
5	m_mid.1416295668657:835a22...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162956687...	Yeas	{"email":"100004158...	1416295668657
6	m_mid.1416295647111:380b14...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162956471...	Really	{"email":"100004304...	1416295647144
7	m_mid.1416295639504:835a22...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162956395...	In Istanbul	{"email":"100004158...	1416295639504
8	m_mid.1416295605861:380b14...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162956059...	Where are you	{"email":"100004304...	1416295605895
9	m_mid.1416295579725:835a22...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162955797...	Ohhhh	{"email":"100004158...	1416295579725
10	m_mid.1416295534639:380b14...	ONE_TO_ONE:10000415...	t_mid.1416295534...	14162955346...	Ohh	{"email":"100004304...	1416295534672

Figure 41 - Threads_db2 file contains Facebook text messages

Every user, message, sticker, thread and action has a unique ID in threads_db2 file. Last seen time, messenger install time, sent and received message time, thread time and last fetch time for each threads are examples of timestamps. Coordinates include latitude, longitude and accuracy.

There is also some more database files such as cookies.db and notifications_db. Cookies.db file contains cookie name, creation time, its value, expiration time and last access time. Notifications_db file contains notification ID, recipient ID, cache ID, notification message and profile picture URLs. Another database files contain less evidentiary data.

b) Multimedia Files

Image files are in data\com.facebook.katana\cache\image\v2.ols100.1 and data\com.facebook.orca\cache\image\v2.ols100.1 directories. Files have .cnt extension and their names consist of random letters and numbers.

Videos are stored in data\com.facebook.katana\files\video-cache\ directory and they do not have extension. An additional unique file exists for each video to store size of the videos.

c) Deleted Artifacts

Some of the messages, posts and pictures were deleted from Facebook application. Some deleted messages can be found in both raw image of data

partition and SQLite Database files. This shows that the third deleted data management rule was applied to database files. But some deleted contents are retrieved from only raw image.

Figure 42 shows the sample deleted text message of Facebook application. It is so clear that a text message with various information such as user IDs, time stamp and coordinates are retrieved. User IDs of both receiver and sender are available. Profile of the user could be found easily on the internet with the help of User ID. Time stamp is in Unix Epoch time format. Epoch time converter can be used to determine the time value. Decimal value of "149C1CBE687" is "1416295605895" and it indicates 18 November 2014 07:26:45 UTC time. Latitude, longitude and accuracy are available as a coordinate information. Exact location can be found on Google maps with coordinate values.

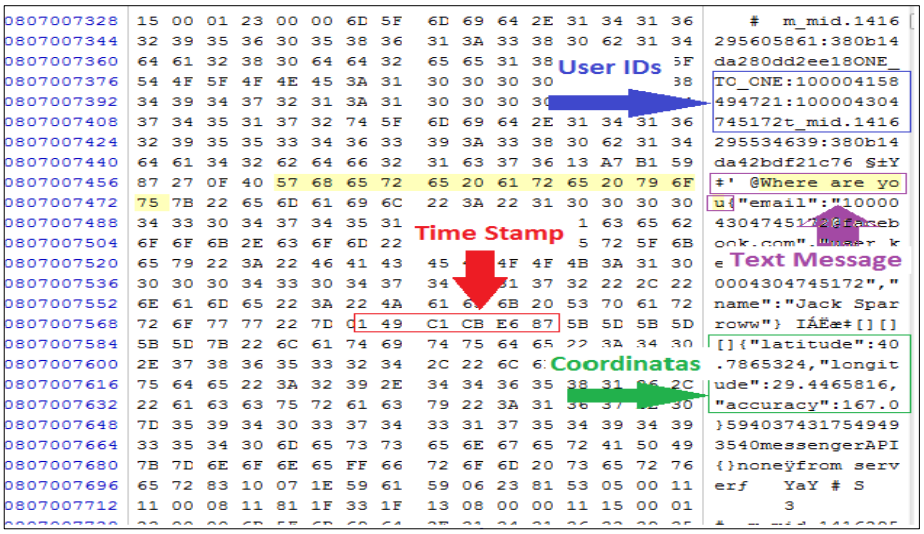


Figure 42 - Sample message format with metadata information for Facebook application

Deleted posts related to location sharing are searched in the raw image of data partition and many location information can be found easily. While performing a check in, Facebook suggests some near locations. All of these locations can be found as well as checked locations. User ID, user name, coordinates, creation time stamp and check in count of the location values are still available as shown in Figure 43.

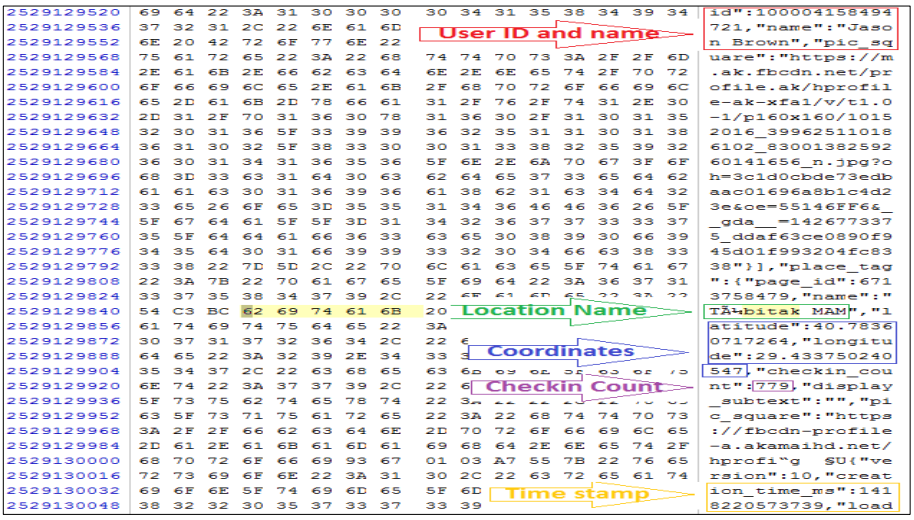


Figure 43 - Sample location sharing post format with metadata information for Facebook application

2. Twitter Artifacts

Twitter artifacts are located in data\com.twitter.android\ and media\0\Android\data\com.twitter.android\ directories. data\com.twitter.android\ directory contains 4 subdirectories; cache, databases, files and shared_prefs. Databases and shared_prefs subdirectories mainly contain evidentiary data.

a) Database Files

Database files are stored in databases folder. 2880712150-17.db file contains twitter conversations with time stamps, unique IDs, URLs, searches, hashtags and followers as shown in Figure 44. File name contains user ID 2880712150.

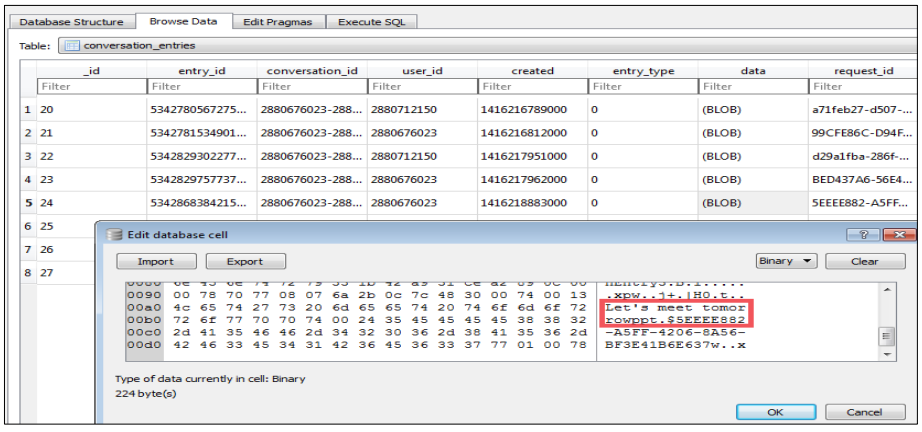


Figure 44 - 2880712150-17.db file contains Twitter direct text messages

Global.db file contains account name, user ID, tweet and mention count. 0-scribe.db file contains logs and IDs in scribe table. 0-17.db, 69079351-17.db and 2880712150-drafts.db files contain many tables but no evidentiary data could be found in them.

b) Xml files

21 xml files are stored in shared_prefs folder. Jacksparrowtm.xml and com.twitter.android_preferences.xml files are most important files. They contain some preferences and settings about twitter application. Other files contain less evidentiary data.

c) Multimedia Files

Multimedia files are located in media\0\Android\data\com.twitter.android\ cache\ directory. Gallery, users and photos folders contain various pictures that their names consist of random numbers and letters.

d) Deleted Artifacts

Some of the direct messages and twitter posts were deleted and tried to recover. Deleted direct messages are searched in both database files and raw image of data partition but they cannot be retrieved. It is noticed that the name of the 2880712150-17.db file changed to 2880712150-19.db. The file size is also reduced. All of these detections show that the second deleted data management rule were applied to database file.

Deleted twitter posts with various information can be retrieved from both SQLite database files and raw image of data partition. Deleted twitter posts are still available more than one location in database file. This shows that the third deleted data management rule were applied to database file.

Figure 45 shows the sample deleted tweet retrieved from both 2880712150-19.db SQLite database file and raw image of data partition. User ID is “ABB42DD6” and its decimal value is 2880712150. Posted text message is

"Waiting new movie". Source fragment shows that the source of the tweet is an Android. Example sources may be iPhone, Web Client, TweetDeck application and so on. Time stamp shows the creation time of the tweet and Unix Epoch time format is used like Facebook application. Decimal value of "149BD378220" is "1416218772000" and it indicates 17 Nov 2014 10:06:12 UTC time. "40.7891341" is the latitude and "29.4414867" is the longitude as a coordinate information.

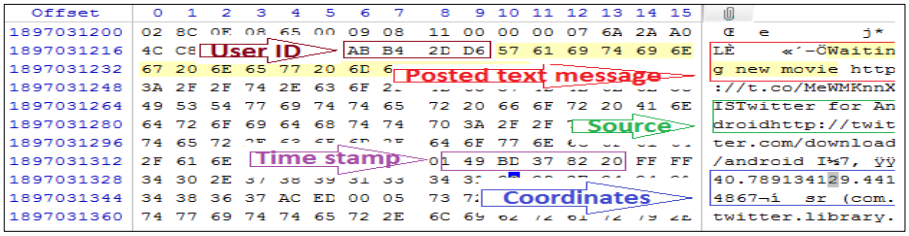


Figure 45 - Sample posted tweet format with metadata information for Twitter application

Many twitter accounts were followed and some of them are removed from the list. Many twitter posts are found related to old accounts in raw image of data partition.

3. Google+ Artifacts

Google+ artifacts are stored in data\com.google.android.apps.plus\ directory. Database and xml files contain evidentiary data.

a) Database Files

Es0.db, es1.db and es2.db database files have the same structure and they contain contacts, user names, unique IDs, time stamps, URLs, activities, comments, searches and locations.

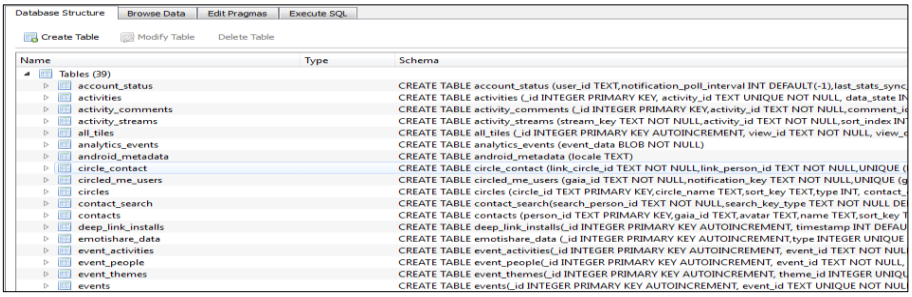


Figure 46 - Es2.db file contains Google+ contacts and comments

b) Xml Files

Accounts.xml file contains account names, emails, unique IDs, URLs and so on. Iu_settings.xml file contains emails, time stamps and some more settings about Google+ application.

c) Deleted Artifacts

Profile update and deletion of some posts were performed to test the deleted contents situation. Some of the deleted posts and old profile entries are found. This shows that the third deleted data management rule was applied to database file. Some of the profile information and posted messages contain unique strings. These unique strings facilitate the analysis.

Figure 47 shows the example posted message format. It was deleted but it is still available in es1.db file with various information. User ID, display info (public or private), text message, URL, creation and modification times are determined.

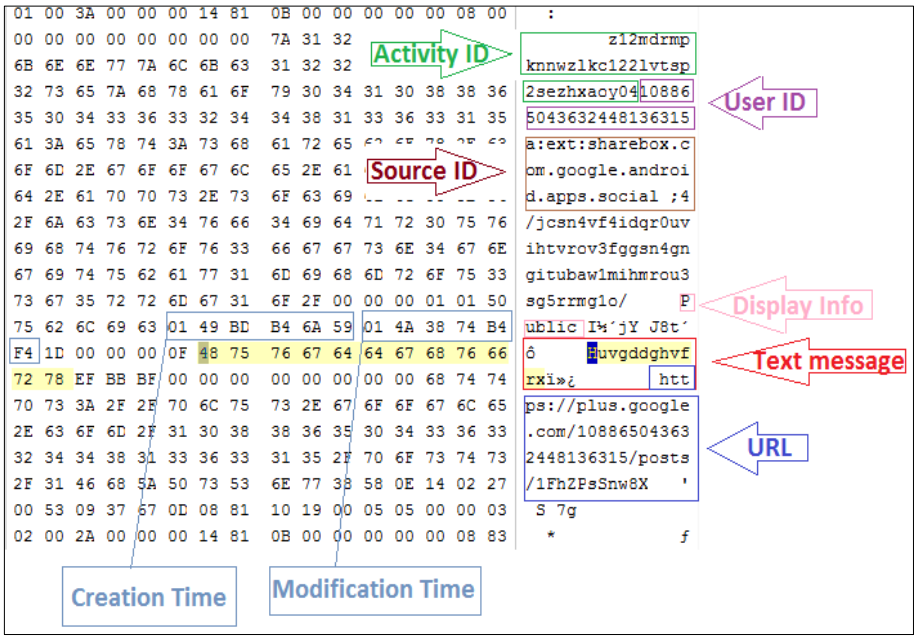


Figure 47 - Sample posted message format with metadata information for Google+ application

Old profile information is searched in raw image of data partition. All of the old information such as phone number, E-mail and occupation are found. Some

accounts were followed and two of them removed from the list. Many old posts are found related to old accounts in raw image of data partition.

4. Instagram Artifacts

Instagram artifacts are located in data\com.instagram.android\, media\0\Android\data\com.instagram.android\ and media\0\Pictures\Instagram\ directories. Instagram application do not use SQLite database files in Android. Xml files, pictures and videos may be evidentiary data.

Data\com.instagram.android\shared_prefs directory contains many xml files that contain some preferences, profile information, time stamps and unique IDs. 1564603320_USER_PREFERENCES.xml file contains geotag enabled or disabled, recent user searches, contacts count, inbox new share count and so on as shown in Figure 48. 1564603320 is Instagram user ID.

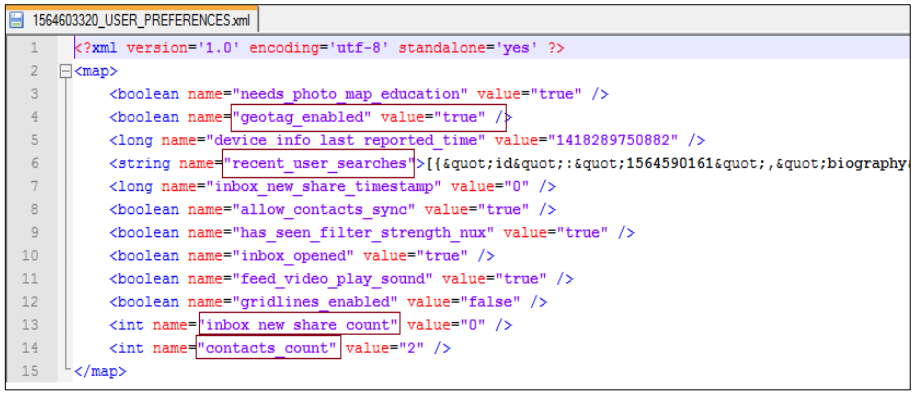


Figure 48 - 1564603320_USER_PREFERENCES.xml file contains Instagram user preferences

1564603320_video_view.xml file contains the watched videos and watching times. Time values are in Unix Epoch time format.

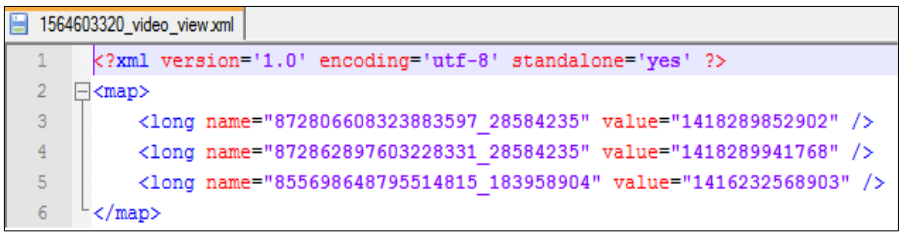


Figure 49 - 1564603320_video_view.xml file contains Instagram watched videos

a) Multimedia Files

Data\com.instagram.android\cache\images\ directory contains various image files related to Instagram application.

Data\com.instagram.android\cache\ directory contains pictures shared via Instagram.

Data\com.instagram.android\files directory contains pending pictures via Instagram.

Media\0\Pictures\Instagram directory contains pictures shared via Instagram.

Media\0\Android\data\com.instagram.android\cache\video\ directory contains videos related to Instagram application.

b) Deleted Artifacts

Instagram application do not use SQLite database files. Profile information is stored in xml files. Old profile information cannot be found in both xml files and raw image of data partition.

5. WhatsApp Artifacts

WhatsApp artifacts are located in data\com.whatsapp\ and media\0\WhatsApp\Media\ directories. Databases, files and shared_prefs subdirectories in data\com.whatsapp\ directory mainly contain evidentiary data.

a) Database Files

Database files are in databases folder. Msgstore.db file contains chat messages with time stamps, phone numbers, pictures as raw data and user IDs as shown in Figure 50.

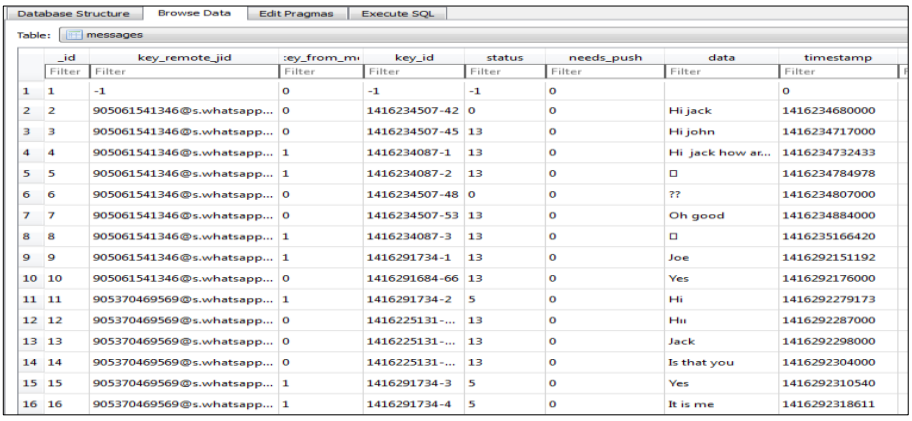

Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: messages

	_id	key_remote_jid	:ey_from_m	key_id	status	needs_push	data	timestamp
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	-1	0	-1	-1	0		0
2	2	905061541346@s.whatsapp...	0	1416234507-42	0	0	Hi jack	1416234680000
3	3	905061541346@s.whatsapp...	0	1416234507-45	13	0	Hi john	1416234717000
4	4	905061541346@s.whatsapp...	1	1416234087-1	13	0	Hi jack how ar...	1416234732433
5	5	905061541346@s.whatsapp...	1	1416234087-2	13	0	□	1416234784978
6	6	905061541346@s.whatsapp...	0	1416234507-48	0	0	??	1416234807000
7	7	905061541346@s.whatsapp...	0	1416234507-53	13	0	Oh good	1416234884000
8	8	905061541346@s.whatsapp...	1	1416234087-3	13	0	□	1416235166420
9	9	905061541346@s.whatsapp...	1	1416291734-1	13	0	Joe	1416292151192
10	10	905061541346@s.whatsapp...	0	1416291684-66	13	0	Yes	1416292176000
11	11	905370469569@s.whatsapp...	1	1416291734-2	5	0	Hi	1416292279173
12	12	905370469569@s.whatsapp...	0	1416225131-...	13	0	Hıı	1416292287000
13	13	905370469569@s.whatsapp...	0	1416225131-...	13	0	Jack	1416292298000
14	14	905370469569@s.whatsapp...	0	1416225131-...	13	0	Is that you	1416292304000
15	15	905370469569@s.whatsapp...	1	1416291734-3	5	0	Yes	1416292310540
16	16	905370469569@s.whatsapp...	1	1416291734-4	5	0	It is me	1416292318611

Figure 50 - Msgstore.db file contains WhatsApp text messages

Wa.db file contains contact people with names, phone numbers, status messages and time stamps as shown in Figure 51.

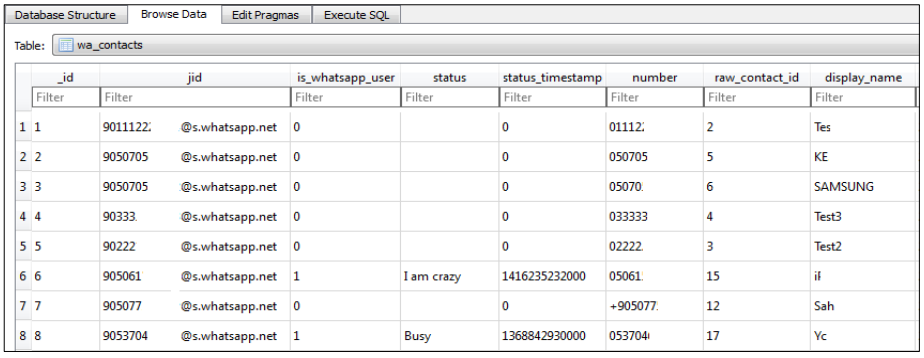

Database Structure | Browse Data | Edit Pragmas | Execute SQL

Table: wa_contacts

	_id	jid	is_whatsapp_user	status	status_timestamp	number	raw_contact_id	display_name
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	9011122 @s.whatsapp.net	0		0	01112	2	Tes
2	2	9050705 @s.whatsapp.net	0		0	050705	5	KE
3	3	9050705 @s.whatsapp.net	0		0	05070	6	SAMSUNG
4	4	90333 @s.whatsapp.net	0		0	033333	4	Test3
5	5	90222 @s.whatsapp.net	0		0	02222	3	Test2
6	6	905061 @s.whatsapp.net	1	I am crazy	1416235232000	05061	15	if
7	7	905077 @s.whatsapp.net	0		0	+905077	12	Sah
8	8	9053704 @s.whatsapp.net	1	Busy	1368842930000	053704	17	Yc

Figure 51 - Wa.db file contains WhatsApp contacts

Axolotl.db file contains public key, private key, some more keys and records in an encrypted format as shown in Figure 52.

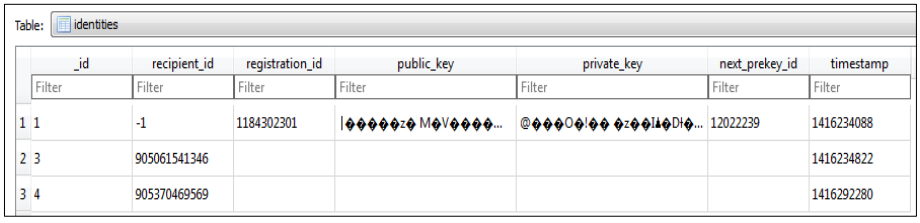


Table: identities

	_id	recipient_id	registration_id	public_key	private_key	next_prekey_id	timestamp
	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	-1	1184302301	|z����z M V���...	@���O ! ��z ��I D ...	12022239	1416234088
2	3	905061541346					1416234822
3	4	905370469569					1416292280

Figure 52 - Axolotl.db file contains encrypted keys and records

b) Xml Files

Xml files are stored in shared_prefs folder. Com.whatsapp_preferences.xml file contains phone number, time stamps, program version and some more preferences. RegisterPhone.xml file contains phone number and registration info. VerifySms.xml file contains sms verification state and time.

c) Log Files

Whatsapp.log file contains various log information about WhatsApp application. Whatsapp-2014-11-19.1.log.gz file is a compressed file and whatsapp-2014-11-19.1.log file also contains various log information.

d) Multimedia Files

Media\0\WhatsApp\Media\WhatsApp Images directory contains sent and received pictures via WhatsApp app. data\com.whatsapp\files\Avatars directory contains profile picture for WhatsApp application.

e) Deleted Artifacts

Some of the text messages, pictures, and location sharing messages related to WhatsApp application were deleted and tried to retrieve with low level methods. Msgstore.db file still contains deleted messages with phone numbers and various time stamps. The deleted location sharing message also retrieved from msgstore.db file. These show that the third deleted data management rule was applied to database file.

Figure 53 shows the sample deleted text message retrieved from msgstore.db file. The message is found only one times. The fragment contains phone number, text message and 4 time stamps. Time stamps are in Unix Epoch time format. First one is a timestamp of a message sent, second one is received timestamp, third one is receipt server timestamp and fourth one is receipt device timestamp. First time stamp value is “149C199BD93” and it indicates 18 Nov 2014 06:31:58 UTC time. There are only very little differences between time stamps.

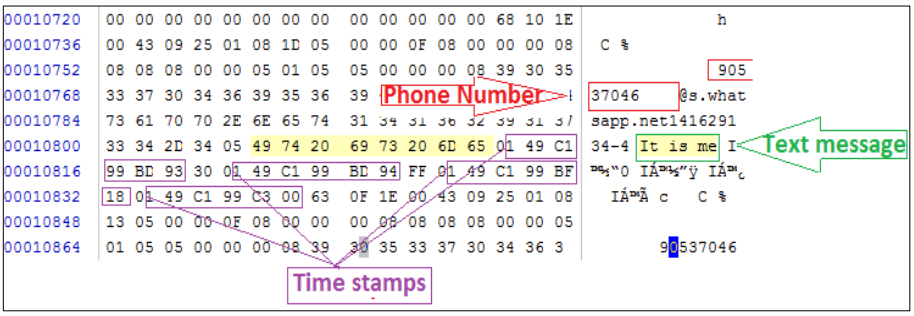


Figure 53 - Sample message format with metadata information for WhatsApp application

Figure 54 shows the comparison of deleted location sharing message retrieved form msgstore.db file with WhatsApp screenshot. The phone number and location information are still available in database file.

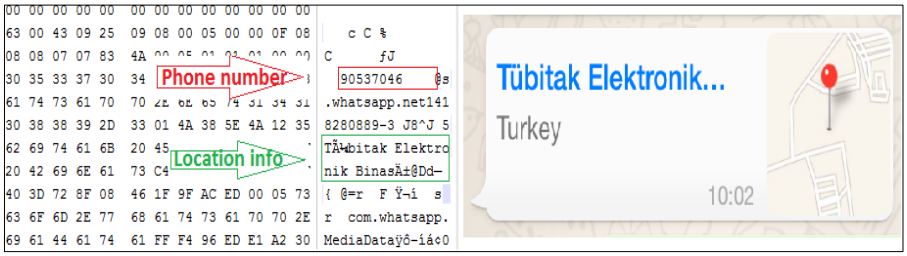


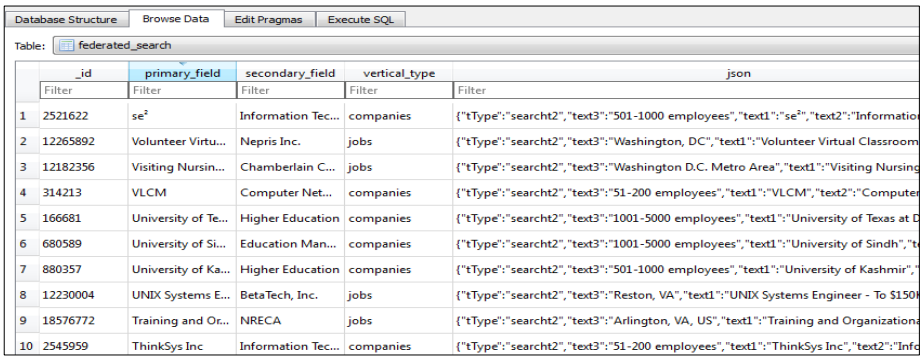
Figure 54 - Sample WhatsApp location sharing message comparison with WhatsApp screenshot

6. LinkedIn Artifacts

LinkedIn artifacts are located in data\com.linkedin.android\ and media\0\Android\data\com.linkedin.android\cache\li_images directories.

a) Database Files

Linkedin.db file contains profile information, time stamps, search results, notifications, companies and recommended people as shown in Figure 55.



	_id	primary_field	secondary_field	vertical_type	json
	Filter	Filter	Filter	Filter	Filter
1	2521622	se²	Information Tec...	companies	{"tType":"searcht2","text3":"501-1000 employees","text1":"se²","text2":"Informatio
2	12265892	Volunteer Virtu...	Nepris Inc.	jobs	{"tType":"searcht2","text3":"Washington, DC","text1":"Volunteer Virtual Classroom
3	12182356	Visiting Nursin...	Chamberlain C...	jobs	{"tType":"searcht2","text3":"Washington D.C. Metro Area","text1":"Visiting Nursing
4	314213	VLCM	Computer Net...	companies	{"tType":"searcht2","text3":"51-200 employees","text1":"VLCM","text2":"Computer
5	166681	University of Te...	Higher Education	companies	{"tType":"searcht2","text3":"1001-5000 employees","text1":"University of Texas at D
6	680589	University of Si...	Education Man...	companies	{"tType":"searcht2","text3":"1001-5000 employees","text1":"University of Sindh","t
7	880357	University of Ka...	Higher Education	companies	{"tType":"searcht2","text3":"501-1000 employees","text1":"University of Kashmir","
8	12230004	UNIX Systems E...	BetaTech, Inc.	jobs	{"tType":"searcht2","text3":"Reston, VA","text1":"UNIX Systems Engineer - To $150
9	18576772	Training and Or...	NRECA	jobs	{"tType":"searcht2","text3":"Arlington, VA, US","text1":"Training and Organizationa
10	2545959	ThinkSys Inc	Information Tec...	companies	{"tType":"searcht2","text3":"51-200 employees","text1":"ThinkSys Inc","text2":"Info

Figure 55 - Linkedin.db file contains LinkedIn profile information

b) Xml Files

Auth_library_prefs.xml file contains user name, member ID and E-mail. LinkedInPrefs.xml file contains user name, E-mail, member ID and profile picture URL.

c) Multimedia Files

Media\0\Android\data\com.linkedin.android\cache\li_images directory contains various image files related to LinkedIn application.

d) Deleted Artifacts

LinkedIn profile update was performed. Some information such as profile picture, university, department, personal website, mobile phone number and address were changed and tried to recover old information. All of them are recovered from linkedin.db file. This shows that the third deleted data management rule was applied to database file. Figure 56 shows the retrieved User ID, phone number, industry, website, last name, time stamp and job looking information.

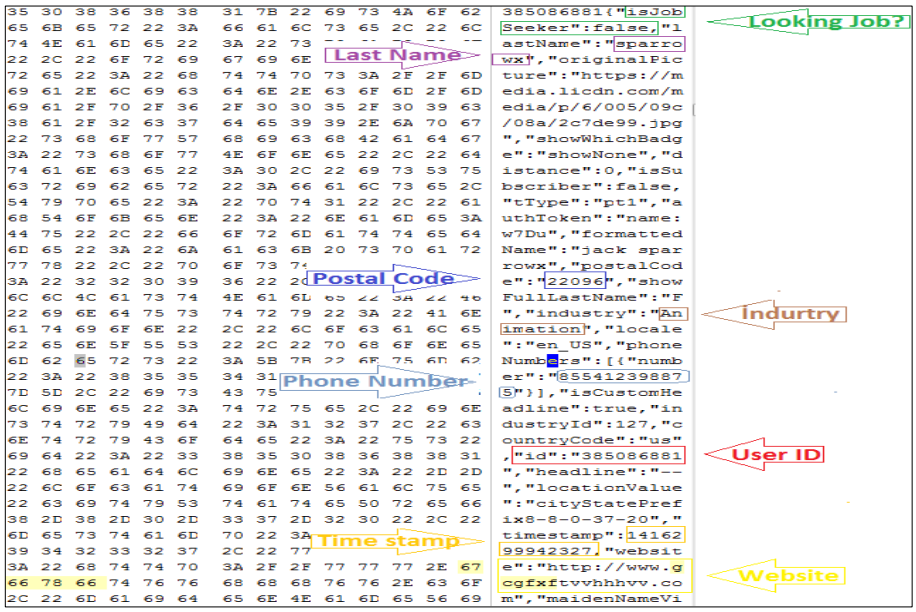


Figure 56 - Deleted LinkedIn profile information retrieved from linkedin.db file

7. Artifacts after Uninstallation of Applications

Applications are uninstalled from device and changes are observed in terms of user artifacts. Physical images of Android device are analyzed to investigate unallocated spaces. It is determined that all files produced by installed applications are deleted but they left some evidentiary data behind.

Facebook messages, posts and user ID's cannot be found in physical image of user data partition. Only some pictures can be recovered from unallocated spaces. Twitter account information such as screen name, profile image URL, description, location, creation time and friend count can be retrieved from xml file remnants. Direct messages and posts cannot be found in physical image. Some of the pictures can also be recovered. Google+ user ID and some profile information such as E-mail and phone number can be retrieved but posts cannot. Instagram user ID and preferences cannot be retrieved but some of the posted pictures can be recovered. WhatsApp phone numbers and account names can be found but chat messages cannot be found in physical image. It was determined that "s.whatsapp.net" string appears before every chat message for WhatsApp application. This string still appears 170 times in physical image but messages near string are overwritten with

garbage values. LinkedIn profile information is also searched but cannot be found in physical image.

Some of the contents of applications were also duplicated in different locations for other applications in the device. Some pictures, E-mails, internet history and account information related to deleted applications are retrieved from device. Actually, these are related to deleted applications but possessed of other preinstalled applications.

In conclusion, it can be said that social media applications generally leave pictures and profile information behind and it is difficult to retrieve chat messages and posts after uninstallation. Duplication of artifacts increases the chance of recovery.

E- More Data Recovery Techniques

Some more data recovery methods and tools; "R-Studio", "foremost" and "strings" command are used for Android device.

R-studio is a commercial powerful data recovery software. It can be used for raw file recovery even for unknown file systems. It can function on damaged, formatted or deleted partitions. It is used for recovery of deleted photos, videos and database files in this research[28]. Figure 57 shows file carving process of Android user data partition image with R-Studio program.

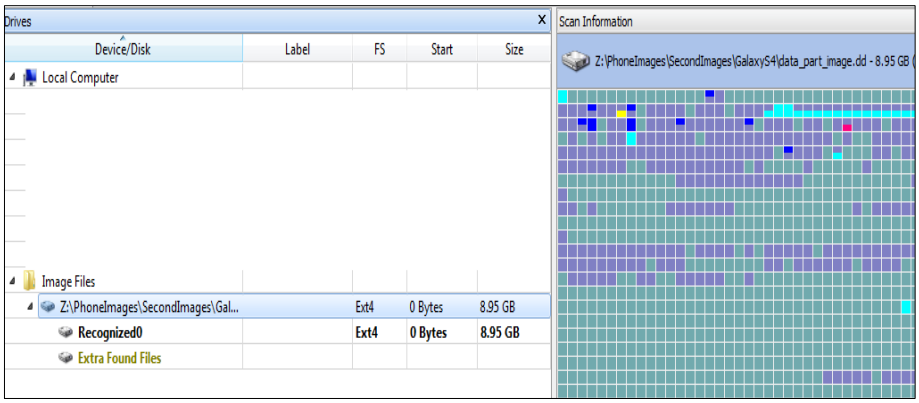


Figure 57 - File carving process of Android image with R-Studio

[28] Available at http://www.data-recovery-software.net/, accessed on December 10,2014

Figure 58 shows the retrieved files after file carving process.

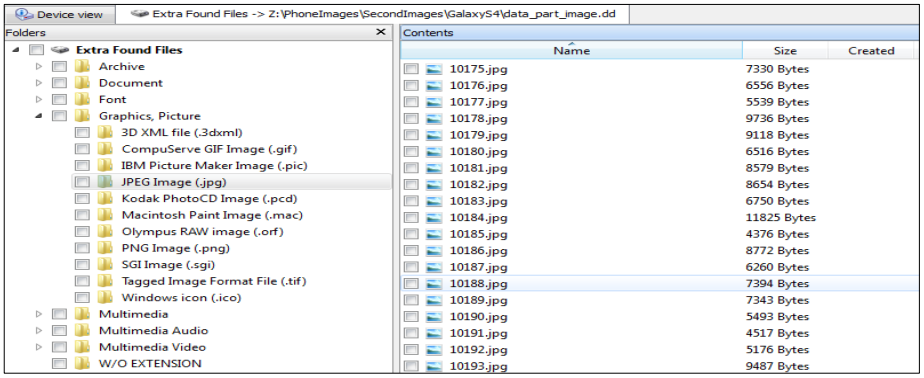


Figure 58 - Results of file carving from Android image with R-Studio

Foremost is a data recovery program in Linux. It recovers files using their headers, footers, and data structures. Foremost is designed to ignore the type of underlying file system. It can recover specific file types, including jpg, gif, png, bmp, avi, exe, mpg, wav, riff, wmv, mov, pdf, ole, doc, zip, rar, htm, and cpp[29].

"# foremost data_part_image.dd" command creates files and directories as shown in Figure 59.

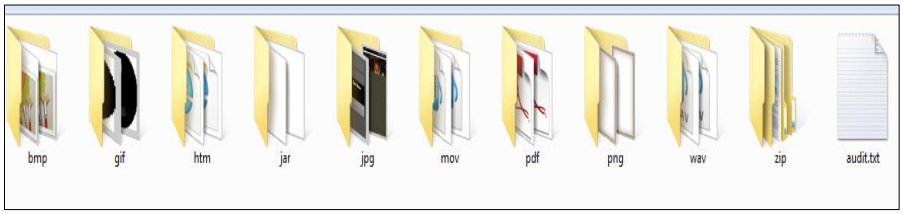


Figure 59 - Foremost output files and directories for Android image

Strings command from Sysinternals can be used to search ANSI and UNICODE strings. Command creates a text file which can be used for regular expression searches.

"#strings data_part_image.dd > out_file.txt" command finds all possible strings and writes them in "out_file.txt" file. After running the command, the size of the "out_file.txt" file became 223 MB.

[29] Available at http://en.wikipedia.org/wiki/Foremost_(software), accessed on December 10, 2014

§5. RESEARCH FINDINGS AND EVALUATION

I. Summary of Research Findings

In this research, the goal of finding the artifacts related to social media applications installed on mobile devices was achieved. The examination process aimed to identify what crucial data is stored, where it is stores and how it can be analyzed. Hundreds of files in proprietary formats were explored and examined manually. User names and user IDs, contacts, chat messages, pictures, time stamps, location coordinates and some more private information were retrieved. If some of them were deleted, there would be a possibility to recover them for most of the applications. Most of these are found in plaintext format in user data partition of the mobile devices. Some of the contents such as keys and special records are found encrypted. It is also checked if retrieved artifacts are consistent with the scenarios. All of these may help law enforcements to reconstruct and clarify a criminal case.

Social media postings also contain metadata which is embedded to visible data or file and generally invisible to the users. Unique IDs, time stamps, coordinates and URLs may be included in metadata category.

Mobile devices use proprietary file formats to store information. Different applications may store the same kind of information in different formats at varied locations. File editors such as SQLite and plist viewers can be used to analyze the stored data. But, deleted entries may not be visible by using file editors. Deleting a record generally just removes the reference to it, but leaves the actual data in the file. There is a possibility to retrieve them by reverse engineering the storage format with hexadecimal viewer. Recovery of data is impossible for some secure deletion methods. If there is a data protection mechanism, the file key is lost after file deletion. The content may remain encrypted in unallocated space and they cannot be retrieved by most powerful carving tools. Secure deletion of the contents can also be achieved by file wipe tools. Some special tools can erase the files and unallocated spaces securely.

Most of the mobile forensic tools do not automatically parse the third party applications. The average forensic tool supports nearly 30 applications out of more than a million iOS and Android apps. They generally choose to focus on most downloaded applications. A relatively small proportion of the entire user data area is examined by typical mobile forensics tools. These kind of examinations are the most time-consuming because of manual methods. Lack of time and training opportunities may also result in missing mobile application data.

II. Evaluation of Artifacts in Terms of Privacy

Social media is the part of the modern world. Collecting and viewing this kind of information by digital forensic investigators brings a number of ethical concerns. For example, an investigator may find irrelevant evidence of adultery or some other sort of inappropriate material (Basset et al., 2006). These kind of irrelevant data need to be ignored and they should remain confidential.

The popularity of social media platforms shows the power of user-created content. People tend to share their location, religion, state of health, family life, pleasure and many opinions about various matters. All of them should be stored securely. In the event of a lost device or malware infection, data stored insecurely can be compromised. Criminal and malicious usage of the contents is getting increasingly widespread.

As a matter of fact, people prefer to share some social media contents publicly. It is possible to achieve these shared contents easily via the internet. The significance of privacy concerns about this public information is questionable. On the other hand, some unshared contents such as chat messages and private pictures must remain confidential.

The best practice for mobile application developer community is that critical data should not be stored on the device. Experts specify that it is possible to develop more secure mobile applications and to reduce storage of sensitive data. If it is needed to store, it must be encrypted. Unfortunately, some of the applications store even passwords and bank account information on the device’s nonvolatile memory

in plain text format (ViaForensics, 2011). These sensitive data are readable as textual data without much processing. If a cybercriminal is able to steal these kind of valuable data by malware infection, users may experience various difficulties. None of the examined applications in this research store passwords in plain text format.

According to a guideline announced by NIST, mobile apps were tested in terms of some requirements (NIST, 2014). Protecting sensitive data and preserving privacy are the first two requirements. It is stated that mobile apps should protect sensitive data at rest and in transit by using cryptographic security services provided by the underlying platform. The app must ask for permission to use personal information and using it only for authorized purposes. This includes location services, geotagging photographs, accessing the camera and so on.

The main reason of developing unsecure mobile apps is that developers generally focus on technical features and they ignore security issues. They also may be unaware of the underlying platform. Users are also social engineered easily. All of these cause information security weakness.

III. Comparison of Applications

User artifacts of six social media mobile applications were observed. Most of the information was found easily and a little information was found a bit harder. This research shows that a forensic examiner can gain nearly a complete access to private messages, public or private posts, time stamps, shared locations, friend lists and profile information of all six social media platforms. Platforms have similarities and differences about stored data contents, time stamp formats, picture formats, unique IDs and logs.

All platforms have a friend lists or following people lists. Detailed information about these people or these accounts was retrieved for all platforms. Their ID numbers, names, surnames, phone numbers, profile pictures and E-mails are examples of their profile information. They were not encrypted and found easily.

Chat messages were found in clear text format easily in both iOS and Android. People prefer Facebook messenger and WhatsApp messenger more than other platforms for chatting. Actually, it is hard to read chat messages in WhatsApp without rooting in Android. They are encrypted in “msgstore.db.crypt” file. There is no need to jailbreak the iOS device to reach the WhatsApp database files in iOS. Backup and synchronization features are enough to see the chat message contents. Twitter has also direct messaging option. Messages were found easily in iOS and Android. But, deleted direct messages could not be found in both devices. There is not any deleted direct message remnant of Twitter even in physical image of user data partition in Android. However, many deleted messages of Facebook and WhatsApp with metadata information could be found in both devices.

Phone numbers are available as a profile information or a part of a user ID. WhatsApp messenger uses phone number as a part of the user ID. So, it is easier to reach a phone number in WhatsApp messenger. If user has shared the phone number as a profile information, it can also be retrieved in other platforms.

Miscellaneous unique IDs were found for users, messages and posts. Facebook stores user (sender and recipient) ID, message ID, action ID, offline threading ID and thread ID. Some of these IDs are associated with time. Twitter stores sender and recipient ID for direct messages, hashtag ID, entry ID, and status ID. Google+ stores user ID, activity ID, comment ID, author ID, view ID, photo ID, owner ID, circle ID, event ID, theme ID and some more unique IDs in Android. Instagram stores some unique values in xml and plist files. They are not obvious, but they may be user IDs and post IDs. WhatsApp stores user IDs like “phone_number@s.whatsapp.net”. Each message has a unique ID which is associated with time. For example, timestamp is 1416234884000 and unique ID is 1416234507-53 for “Oh good” message. WhatsApp also stores picture ID. LinkedIn stores member (user) ID, company ID, notification ID, group ID, message ID, Job ID, server ID and campaign ID in linkedin.db database file in Android platform. There is no database file in iOS platform for LinkedIn, Instagram and Google+. These unique IDs may be stored in plist and xml files in iOS but they are not obvious.

Multifarious time and date information were found during the investigation of user artifacts. Applications store time values in three different formats: Unix Epoch time, Mach Absolute time and clear text time formats. Facebook stores messenger install time, message time, last visible action time, snippet times for threads and last synchronization time in Unix Epoch time format. Twitter stores profile create time, friendship time, search time, direct message time, status message (posted tweet) time and status message update time in Unix Epoch time format. It stores hashtag update time in clear text format. On the other hand, time values in scribe.1.sqlite file is in Mach Absolute time format. Google+ stores circle synchronization time, people synchronization time, event synchronization times, last analytics synchronization time, last settings synchronization time, last squares synchronization time, last emotishare synchronization time, last notification time, activity creation and modification time, comment creation time, profile update time, contact update time in Unix Epoch time format. Instagram stores last device log time in Mach Absolute Time format in iOS. It stores app install date, some registration and expiration dates in clear text format. It stores device info last reported time, expiration dates, picture and video view time, last registration time, first run time, posted message time, some expiration times in Unix Epoch time format. WhatsApp stores chat message date, status message date and last modification date for contacts in Mach Absolute time format in iOS. It stores these times and some other times for messages such as message receive time in Unix Epoch time format in Android. LinkedIn stores notification time and cache time in Unix Epoch time format. It stores profile metadata time in clear text format.

Location information generally consists of longitude, latitude and place name for both iOS and Android platforms. Facebook location information contains longitude, latitude and accuracy in both İOS and Android. Twitter states longitude, latitude, place ID and place name in İOS and Android. Google+ states longitude, latitude and place name in Android. Instagram specifies geotag option enabled or disabled in Android. WhatsApp location information contains longitude, latitude and place name in Android. Only location name of a deleted WhatsApp location sharing message could be found in iOS.

Various URL information was found for all platforms. They may indicate profiles, profile pictures, shared locations and shared posts. URLs may also contain user IDs. Some of the URLs can be used to look the contents on the internet. But, some of them are only usable with users' account credentials.

IV. Comparison of Operating Systems

The main difference between iOS and Android is encryption mechanisms. Every iOS device has a dedicated AES 256 crypto engine which comprises the device's unique ID (UID) and a device group ID (GID). The UID is unique to each device and the GID is common to all processors in a class of devices No software or firmware can read them directly (Apple Inc., 2014). The entire file system is encrypted in iOS devices with hardware level encryption. User data partition could be imaged but file contents could not be seen. Any vulnerability has not been found yet for physical imaging of iPhone 5S without jailbreaking. Imaging the user data partition is enough for examining the files in Android. So, iOS seems more secure than Android in this respect. If new versions of Android come with hardware level encryption mechanisms, Android may compete with iOS to become more secure.

3 of the 6 social media mobile apps, Facebook, Twitter and WhatsApp use SQLite database files to store entire database in both iOS and Android. Google+ and LinkedIn use SQLite files only in Android. Consequently, more evidentiary data were found for them in Android. Instagram do not use SQLite files in both devices. Only xml, log and plist files were analyzed for Instagram. To avoid using SQLite database system makes Instagram more secure than others. Actually, xml and plist files of Instagram are also readable but it is more difficult to analyze them than straightforward SQLite files.

Jailbreaking and Rooting methods are different. The main concept behind them is similar. But, the limitations of control over devices are different. Jailbreaking and Rooting methods are changing since new operating system versions come with new security features.

Connection with forensic workstation methods were chosen different in this research. iOS devices is connected via wireless network. Android device is connected via USB cable.

Logical imaging with XRY is nearly the same for both iOS and Android. Backup method is used for iOS. However, both backup and agent extraction methods are available for Android.

Data carving from unallocated spaces is easier in Android. After bit-to-bit imaging of user data partition, carving tools and reverse engineering methods were used. So much deleted content could be found if they were not overwritten. Data recovery for some older iOS versions (for iPhone 4 and older models) is possible. But, there is no known method for new (for iPhone 4S and newer) devices.

§6. CONCLUSION

Social media and mobile device are two major appreciated concepts in digital forensics area. Social media forensics on mobile devices is the topic of the future world. Users generally do not know the artifacts they left behind. Insecure data storage mechanisms, social engineering threats and the spread of mobile malwares are increasing the necessity of social media mobile forensics.

Two popular smartphones with six most used social media mobile applications were investigated. Smartphones and applications were chosen based on statistical usages in Turkey and in the world. Sample scenarios were prepared for each application. Forensically sound current imaging and analyzation methods were used.

The results of the research shows that mobile social media applications generally create database files, log files, xml files and plist files to store most of the private and evidentiary data. These files can be retrieved and examined easily with commercial and open source forensic tools. User names and user IDs, phone numbers, friend lists, E-mails, chat messages, pictures, time stamps, location data and profile information were retrieved as a potential evidence even after they have been deleted. iOS platform seems more secure and harder to analyze than Android in application forensics perspective.

I. Future Work

This research is an overview of user artifacts related to current most popular social media applications. Several future researches are needed inevitably in social media forensics on mobile devices. Renewed operating system versions and new social media applications will require new researches. Better forensically sound methods can be performed for physical imaging of the nonvolatile memory of mobile devices. Further research in this area may be performed on acquiring and analyzing the volatile memory (RAM). Encryption and secure erasing of private social media data may be accomplished in any future research.