UNIVERSITY OF CENTRAL OKLAHOMA

Edmond, Oklahoma

Jackson College of Graduate Studies

MOBILE FORENSICS: ANALYSIS OF THE MESSAGING APPLICATION SIGNAL

A THESIS

SUBMITTED TO THE GRADUATE FACULTY

In partial fulfillment of the requirements

For the degree of

MASTER OF SCIENCE IN FORENSIC SCIENCE

By

Samantha M. Judge

Edmond, Oklahoma

2017

MOBILE FORENSICS: ANALYSIS OF THE MESSAGING APPLICATION SIGNAL

By: Samantha Michelle Judge

A THESIS

APPROVED FOR THE W. ROGER WEBB FORENSIC SCIENCE INSTITUTE

November 2017

By ______________________________

Dr. Mark R. McCoy Committee Chair

____

Dr. John P. Mabry Committee Member

____

Ms. Rachael Elliott Committee Member

Table of Contents

Abstract

This study reviewed if there are ways to recover messages, image, videos, and call logs within the mobile application Signal, developed by Open Whisper Systems. The purpose of this study was to research the data recovery as fact or fiction, while providing which tools and extraction methods produced more accurate results. Further research was needed to explore data recovered from an Android mobile device compared to an iOS mobile device. The forensic tools used to conduct this research included UFED 4PC (Universal Forensic Extraction Device), version 6.3.1.477 with an internal build version 4.7.1.477 and UFED Physical Analyzer version 6.3.11.36, developed by Cellebrite. The study also compared the results using Cellebrite to three different open source tools, iPhone Analyzer, iExplorer, and Autopsy. The meaning of open source can be a tool or program that is designed for specific tasks, yet the source code is openly published to the public. These tools or programs are free of charge unless the user opts to pay for the expanded versions.

Overall, the results were dependent on the make and model of the mobile devices. Out of four different types of mobile devices, only one device produced viable results when it came to the Signal Application. The physical extraction from UFED 4PC and Physical Analyzer on the Android ZTE Z993 device was able to recover an abundant amount of data. The other three devices produced minimal results only showing the installation of the application, but no real message data using the UFED 4PC version 6.3.1.477 and UFED Physical Analyzer version 6.3.11.36 software. The three open source software, iPhone Analyzer, iExplorer, and Autopsy also produced minimal results with the exception of the Android ZTE Z993 device. Autopsy free version was able to parse the data missed by the Cellebrite commercial tools and recover some of the missing images within messages sent inside of the Signal Application.

Mobile Forensics: Analysis of the Messaging Application Signal

Chapter I

Introduction

Before the dawn of the digital age, crime scene technicians didn't require the same steps to preserve evidence as they do now. Mobile device forensics is directly connected to digital forensics and can be defined as being the recovery of digital information or data which is often used for criminal evidence (Bommisetty et al., 2014). Mobile Device Forensics by definition applies only to mobile devices, e.g. tablets, cell phones etc., but the term also includes any portable digital device that has both internal memory and communication abilities such as PDA devices and GPS devices (Bommisetty et al., 2014). With all the recent advancements in technology, there is an increasing amount of digital evidence that could be considered vital to a case. Many computers, phones, and tablets hold the key to unlocking a mystery, solving a case, or ensuring the guilt or innocence of a person of interest. Mobile devices hold a magnitude of information, such as call logs, text messages, pictures, videos, GPS locations, internet browsing history, and an array of applications to uncover and dissect. Unfortunately, with those same advancements, come security and privacy concerns. Looking at mobile devices, those concerns are now becoming helpful to the user, but harmful to an investigation.

Security and privacy not only create obstacles for crime scene technicians, but also, investigations can stall while attempting to obtain vital evidence from an encrypted device or an encrypted messaging application. There are multiple applications that enable a mobile device to send and receive instant messaging, video calls, images, and text messages. These applications were once not as secure and did not enable the user to change or hide what was sent or received, those times have changed. Currently, Android applications use SQLite database files to store data. SQLite is an implementation of a Structured Query Language database. SQLite is an open source, in process library that contains a no configuration database engine. This type of database is often popular due to its size and reliability. The database can be recognized by the file extensions *.db* and *.sqlitedb*, or have no extension at all (Kelel 2013). In order to retrieve this database, it would have to be acquired from the devices using a validated method of extraction.

In the last eight to ten years, open source code has been shared in order to enable developers to encrypt the databases of these devices. As previously stated, the meaning of open source can be a tool or program that is designed for specific tasks, yet the source code is openly published to the public. These tools or programs are free of charge unless the user opts to pay for the expanded versions. While this is great for the user in terms of privacy and secure conversations, this could be damaging if the encryption was unable to be decrypted. The database stores all of the messaging capability within some applications available both commercially and open source. Commercial tools are best explained as the software or tools that are paid for, while opens source tools are available through the internet. Open source tools are completely free, while having the option of paying for certain aspects or extra tools within the software. With all of these changes, it has become difficult for the technicians and investigators to keep up with the evolving technology. It seems like every day there are more and more ways that a criminal can hide a digital footprint that was once left behind for a forensic examiner to find. If criminals are able to download free applications in order to commit crimes and hide the evidence such as timestamped messages, images, and even “ghosting” GPS locations, then the investigator or examiner could end up with nothing or unreliable results.

This study began as a conversation between work colleagues, as well as, current media coverage of the government using the Signal application for messages during the recent

Presidential election. There seemed to be a common misconception that since government agencies, such as the Central Intelligence Agency, could not break the Signal encryption that forensically, the messages could never be retrieved (Barrett 2017). This revealed an array of questions. What if? What if they couldn't be retrieved? What if this changes everything we can do when it comes to mobile forensics?

Background

As previously stated, there are an abundance of applications in different operating systems like Android and iOS. One application is Signal. Signal is end-to-end encrypted, meaning that no one but the device and conversational partner's device can read the messages sent. The team behind the software, Open Whisper Systems, is a privacy centered not-for-profit organization, and relies on grants and donations. This company is very small with only around 10-12 employees. Perhaps most importantly, Signal is open source, meaning that the code is publicly viewable. It can be examined for potential security holes, and according to many different reviews, it has stood up to auditing. The tool is peer reviewed, meaning it's reviewed by not only the author, but other colleagues, users, and testers, to determine if there are any technical errors or issues with the code of the software. In this case, the code is open source so that the public can also review, modify, and even correct any errors found. The developers claim that no matter what, the data could not be recovered by a third party and is not stored on any servers within Whisper Systems (Marlinspike 2017). The information collected is saved long enough to send and receive the messages and then it would be gone.

There are plenty of articles and recent news worthy uses of signal, but what does this mean for the forensic community? Well, in short, that is why this research needed to be done.

Many articles claim that the security of the application renders hackers unable to access your message, images, or videos. The claim is dependent on if you are in a secure connection with another person. It has been reviewed over and over yet nobody speaks about what could happen if that same phone is in someone else's hand.

Signal started out as a voice calling application called RedPhone with an encrypted texting program called TextSecure in 2010. The company then switched to Whisper Systems and released firewalls, as well as, other forms of encrypting data. At the time, all of their applications were only for the Android platform. In 2011, they were acquired by Twitter who then released these two applications under the GPL3v (General Public License) making it an open source software. In 2014, Whisper Systems changed the name of the protocol to merge RedPhone and TextSecure thus creating the Signal Application. Since then, Signal has been consistently updated with more and more features (Wikipedia 2017).

Currently, Signal allows voice and video calls, group or single text messages, pictures, and video messages on iOS and Android devices. Signal uses a Wi-Fi or data connection, and uses encryption keys to verify each of the end-to-end encryption. The keys that are used for the encryption are stored on the users' mobile device, not with the developer. In order to authenticate between connections, users will either compare key fingerprints or scan QR codes. Signal will also notify users of key changes. Not all communication within Signal has to be end-to-end encrypted. The user can opt out of the encryption and allow unencrypted communication to be sent and received. In order for any of the encryption to work, both users must have the Signal application installed (Shelton 2017). Recently, Whisper Systems released another update to the application. Users can now set timers for their messages to be deleted once the intended recipient has read the message (Lund 2017). The timers can be set from seconds to up to a week long. As of right now, unencrypted messages, to include those on an iOS cloud backup, are not able to use this feature.

Signal seems to have created some sense of security; it has also created and overcome many limitations. One of the limitations is the set-up. Signal requires users to have a phone number for verification that can only be used on one device. This phone number does not have to be the user's number located on the device, but can be a VoIP or even a landline telephone number. The phone number just needs to be able to receive the verification code for the set-up to be completed. Users can also set-up a Google Voice account in order to receive the verification. Signal was once only specific to the Android platform, but since has branched out to the iOS platform.

While Signal stores messages, keys, and passphrases on the user device, overall it still needs servers to relay these messages and locate contacts who also have the Signal application installed. With video and voice calls, the exchange is peer-to-peer. If the caller is not within the contacts of the Signal user, the call is then routed through the servers in order to conceal the users IP address (Marlinspike 2017). One major limitation still exists for Signal. The application uses the servers to locate other contacts that are registered, thus not having any preservation for privacy. Whisper Systems claims that the numbers are only stored long enough to connect the calls, but what if the servers were hacked, would all the registered phone numbers be uncovered? The phone number may not be able to show who the direct user is, but even though it's easy to use another phone number, many users still communicate using the phone number original to the mobile device. Overall, it's unclear if the user's anonymity would remain intact.

Whisper Systems has released the complete source code so that users and developers can examine the code and report back to the creators that it is functioning as it should. This also enables developers to make their own copies and versions of applications using the same encryption code. Everything from the encryption to the servers is available as open source code. Whisper Systems will provide support with their own applications and servers, but will not provide support for those users or developers who host their own servers (Lund 2017). The Signal application is available and distributed through Google Play, Apple, and the Chrome Web Store. Even with the encryption and server available to the public, many features within Signal have caught the attention of some government officials, as well as, Edward Snowden, former CIA employee and NSA contractor. Snowden is also responsible for the leak of multiple agency surveillance programs. He has deemed this application and anything created by Whisper Systems a secure and reliable application to use.

In 2012, The National Security Agency deemed one of the original parts to Signal, RedPhone, a major threat to the ability to track and reveal communications between enemies of our country. As previously stated, in 2015, the American Civil Liberties Union urged officials at the U.S. Capitol to have the staff begin to use the Signal application as a form of secure communication. It had been rumored that during the recent presidential election, candidates and staff used Signal in order to exchange communications about the opposing candidate. In 2017, the U.S Senate approved the use of Signal within government organizations. Senator Ron Wyden stated “I have long argued that strong, backdoor-free encryption is an important cybersecurity technology that the government should be embracing, not seeking to regulate or outlaw. My own Senate website, which has used HTTPS by default since 2015, was the first Senate website to do so. With the transition to default HTTPS for all of the other Senate websites and the recent announcement by your office that the end-to-end encrypted messaging app Signal is approved for Senate staff use, I am happy to see that you too recognize the important defensive cybersecurity role that encryption can play”. (Hardwick 2017). HTTPS is the HyperText Protocol Secure. HTTP defines how messages are formatted and transmitted over the internet, but in cases with HTTPS, there is a secure socket layer (SSL) for security purposes. The types of websites that normally use this are e-commerce, banking, and investments. An example of an e-commerce site would be Target. Target uses the secure socket layer to protect user’s credit card or banking information while making purchases from their website.

Overall, the application is becoming more and more popular due to the security and privacy it offers the user. In an article written by Thorin Klosowki, he summed up the easiest way to understand what signal is and how it works. Klosowki stated “Pretty much any article you read about security, from Snowden to Russia, includes a mention of Signal. That’s because every message that’s sent over Signal supports end-to-end encryption. This security measure means that if someone intercepted your messages, or found them on a server somewhere, they would see gibberish, not the actual text of a conversation. Signal is also open-source, peer-reviewed, and routinely audited, which means it’s pretty much always up to date from a security standpoint.” While the company may have a small amount of employees, around ten or so, the application is taking off like a wildfire, and is sure to grow more and more interest over the coming years.

Chapter II

Research Questions

During this research there were four questions that were to be examined. Can you retrieve data from the Signal Application on Android and iOS platforms? Signal was originally for the Android platform and then expanded to the iOS platforms and in order to give the research some depth, both platforms needed to be utilized. The extraction methods were dependent on the model of the mobile devices and not all extraction methods were available for each device. This led to the next question in the research. Which methods of extraction are the most useful? Depending on the model of the mobile devices, the extraction method was either physical, logical, or the file system. The types of extractions also give different amounts of data.

In order to properly do the extractions, multiple tools needed to be utilized. Which software or tools were able to provide the most data? The tools used were from Cellebrite, iPhone Analyzer, Autopsy, and iExplorer. Not all tools were used on all of the devices since some of the tools were only made to operate with either Android or iOS. Finally, the question remained if more or the same amount of data would be recovered between the Android and iOS platforms?

Methodology

This study reviewed if there was a way to forensically recover messages, images, videos, and call logs within the Signal Application developed by Open Whisper Systems. Currently, there is not enough data or research on whether this is a viable option. If the data is not able to be recovered using the current tools in mobile forensics then this could pose serious issues to investigations and the forensic community. Not being able to retrieve this data could give a suspect the ability to hide a vast amount of criminal activity. The purpose of the study was to research the data recovery abilities as fact or fiction, while providing which tools and extraction methods were the most useful. Further research was conducted to explore if the data was more easily recovered from an Android device or an iOS device. The accessibility to the data was the key to the completion of this study.

Due to limited commercial forensic tools, the extractions were performed with UFED 4PC (Universal Forensic Extraction Device), version 6.3.1.477 with an internal build version 4.7.1.477 and UFED Physical Analyzer version 6.3.11.36. UFED 4PC enables multiple acquisitions and updates frequently to support the current mobile operating systems. The type of forensic acquisition method was dependent on the make and model of phone and operating system. The Android ZTE Z993, allowed for a physical, logical, and filesystem extraction, while the Android LG, iPhone 4S, and the iPhone 7, devices only allowed the filesystem or logical extractions. Due to the security on these devices, recent versions of iOS platforms are now only allowing the extraction via logical or filesystem. A physical extraction is the extraction that would produce the most data. It extracts the bit-by-bit binary image of the mobile device flash memory. This type of memory contains the file system, user data, hidden files, unallocated space, and may even contain passwords. The Filesystem extraction can produce the user data, file system, mobile application data, and, dependent on the make and model of a device, the hidden or protected files. The logical extraction contains the user data such as, SMS, call logs, pictures, videos, audio files, contact lists, and some application data. The logical extraction does not recover deleted data.

In the event that an acquisition could not be completed on a specific Android device, rooting the device could possibly allow access to the file system extraction. Rooting allows the user to obtain privileged control or root access. It acts as an administrative permission to overcome limitations that either the carrier or hardware manufacturers set so that the user can alter or replace system applications, change settings, or run special applications. Rooting is similar to Jailbreaking for an iOS device but Jailbreaking bypasses multiple types of prohibitions that Apple has placed on the device. Jailbreaking allows the user to change the operating system and install non-approved applications to an iOS device. For this study, none of the mobile devices were rooted or jailbroken. Once the acquisition is performed using UFED 4 PC version 6.3.1.477, Cellebrite Physical Analyzer version 6.3.11.36 enables the user to run physical and file system extractions on an iOS device depending on the make and model of the phone. Physical Analyzer version 6.3.11.36 also allowed an advanced logical extraction to be performed on the iPhone 4S. Physical Analyzer version 6.3.11.36 can also decode Android physical extractions. Currently, Cellebrite, the creator of UFED tools, also has a feature called UFED Cloud Analyzer. On both the iPhone 4S and the iPhone 7, iCloud had a listed account, but the backup feature was turned off. This feature was not useful during the course of this study. This tool is only available in Physical Analyzer version 6.3.11.36 as an additional feature.

Two Android devices and two iOS Apple mobile devices will be used to extract data from the Signal application and forensically examine which data was available and the amount of data available. The first device used was a ZTE Z993 Prelude, Android 4.1.1 with a kernel version of 3.4.0 (*Figure 1*). The extractions were performed using a Cellebrite Black Tip T-100 with cable adapter A.   The second device was a LGUS375, Android 6.0 with a kernel version of 3.10.49 (*Figure 2*). The extractions were performed also using a Cellebrite Black Tip T-100 with cable adapter A. The Apple devices being utilized were an iPhone 4S IOS 9.3.5 (*Figure 3*) and an iPhone 7 IOS 11.0.3 (*Figure 4*). The iPhone 4S extractions were with a Black Tip T-110 with cable adapter A, while the iPhone 7 used a Cellebrite 210 cable.

Each mobile device was chosen with separate operating systems due to the fact that in some cases, the systems will store data differently. A factory reset was done on each device prior to the install of the Signal Application with the exception of the iPhone7. The iPhone 7 did not have a factory reset completed to show exactly how much information would be on a suspects phone, as well as, if there were any differences using a newer device vs the older versions. All available extractions for each of the four devices were completed using UFED 4PC version 6.3.1.477 to show what the basic software available would obtain from the Signal Application. In the following paragraphs, step-by-step instructions are detailed on how to properly install and activate the Signal Application using a Google Voice obtained mobile number. The hardware that was used was provided by the University of Central Oklahoma. The computer used was a Forensic Recovery of Evidence Device or FRED for short. All extractions, reports, screenshots, and other images used in the study were saved on a DiamondMax 8S SATA150 HDD, 40 GB external hard drive.

Figure 1 ZTE Z993

Figure 2 LGUS375

Figure 3 iPhone 4S

Figure 4 iPhone 7

Signal requires a telephone number in order to install and register the application. The research done with this study was completed using WIFI only and the mobile devices did not have the cellular option enabled. Because of this, a google voice account was set for each device. The first step to verifying Signal was to set up each phone with a Gmail account. Once the email account was active, a google voice account was attached to each email. The google voice works just as if it had been a mobile device. The user can make calls, send messages, and even have a voicemail set up. Within Signal, the main requirement is to verify the application with a mobile or landline phone number. The google voice number was inputted and a code was sent to the google voice account in order to register that number to that Signal account. Signal only allows for one phone number to be registered at a time so there were four separate accounts set up for each one of the mobile devices used. This requirement, while bothersome at times to users, also gives users the availability to register a phone number that is not tied to their personal mobile device or home phone numbers. Signal locates contacts that are already registered users stored in the mobile device. Other users can be located by typing in the Signal number that the user has set up and registered. See *Figures 5-12* listed below.

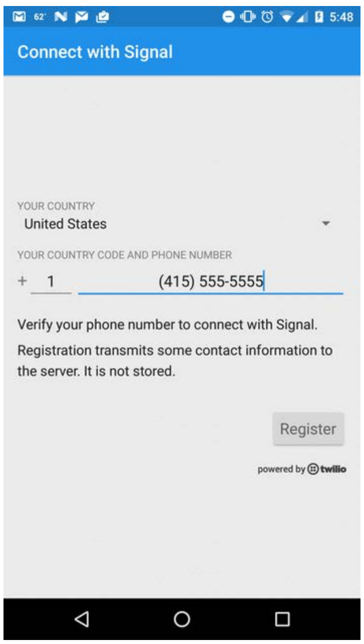


Figure 5 Installation Steps Collazo (2017)

The first step is to enter either the actual mobile device number or the one created using Google Voice. In this case, the phone numbers were generated from Google Voice.

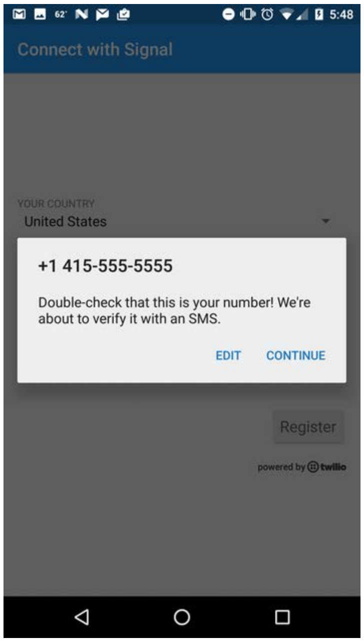


Figure 6 Installation Step 2 Collazo (2017)

The next step will inform the user the mobile or landline number is about to be verified. Hit the continue button.

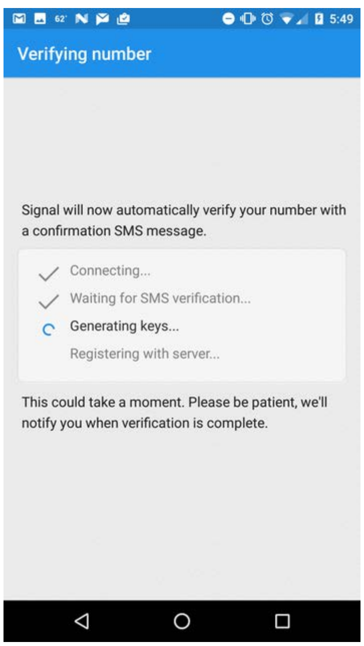


Figure 7 Installation Step 3 Collazo (2017)

Once the process begins, the six digit code will be sent via text message. After the code is entered, Signal will register the phone number and the application can be used.

In order to provide a variety of data to extract on the devices, calls, text, images, and videos were exchanged. *Table 1* shows an example of the variation of message types were delivered and received between all four mobile devices. In each set of messages at least one image and basic text messages were exchanged. There were also a videos sent between the four devices. In order to use all the functions of Signal, multiple messages were set to disappear and some were also deleted by hand. Signal enables the sender to set messages to disappear between five seconds and up to one week from the time the message is read (*Figure 8*). Open Whisper Systems prides itself on being a secure way to communicate. In order to provide that security, they have given users the ability to check what they call safety numbers to ensure that the person you are communicating with is also secure and verified. *Figures 9 through 12* show users how these safety numbers look within the Signal Application. The safety numbers can be verified between parties by either matching the numbers listed or if you are in the same area as the person, you can scan the QR code and it will either show a green check mark or a red X if the numbers do not match.

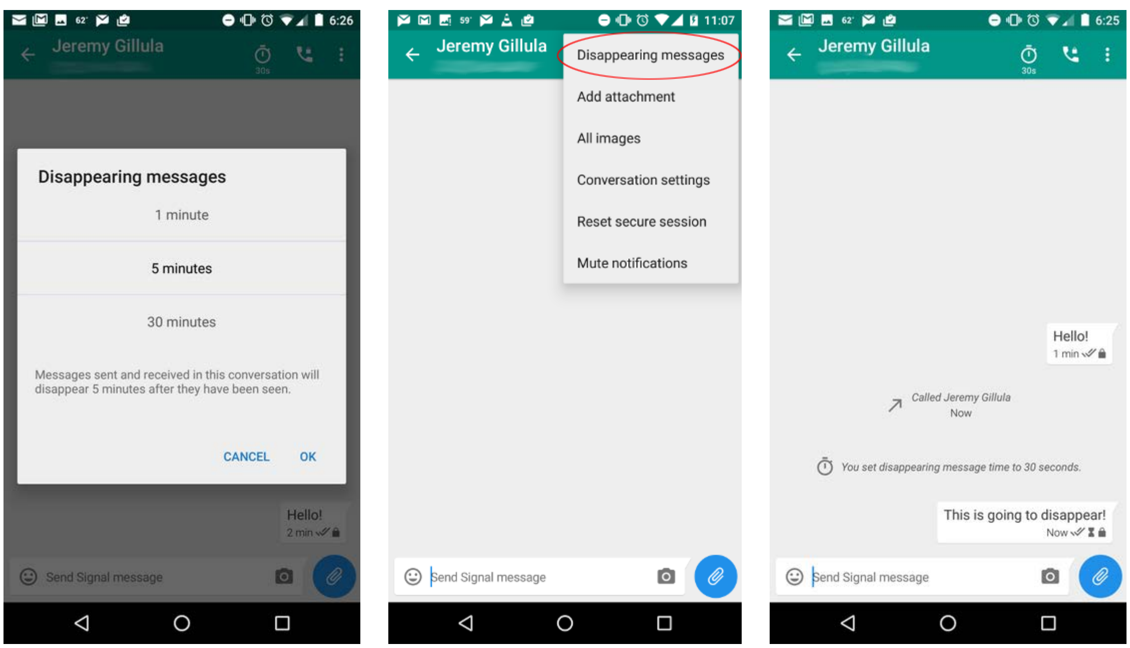


Figure 8 Disappearing Message Settings Collazo (2017)

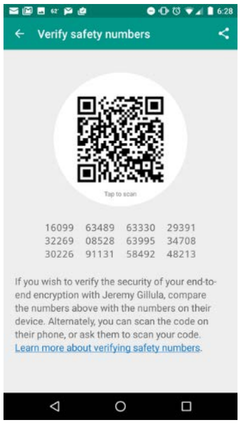


Figure 9 Safety Numbers Collazo (2017)

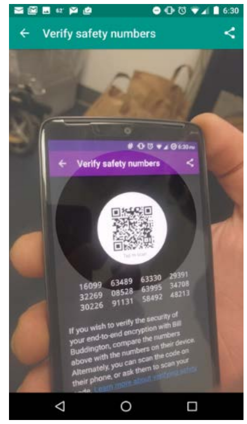


Figure 10 Safety Numbers Collazo (2017)

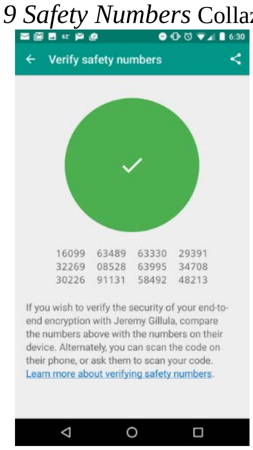


Figure 11 Safety Accepted Collazo (2017)

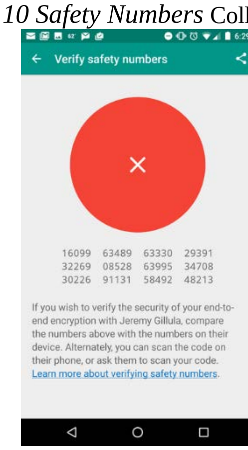


Figure 12 Safety Denied Collazo (2017)

ZTE Z993									
Date	Time (UTC-5)	Type of Message	Sent/Received	Phone	Date	Time (UTC-5)	Type of Message	Sent/Received	Phone
9/4	6:42p	Text Message	Sent	LGK8	9/4	06:48p	Text Message	Received	LGK8
9/4	6:45p	Text Message	Sent	LGK8	9/4	06:53p	Text Message	Received	LGK8
9/4	6:46p	Text Message	Sent	LGK8	9/4	07:20p	Text Message	Received	LGK8
9/4	6:55p	Image	Sent	LGK8	9/4	07:46p	Image (set to disappear/5min)	Received	LGK8
9/11	11:03p	Text Message	Sent	LGK8	9/4	06:53p	Text Message	Received	LGK8
9/12	6:48p	Text Message plus article link	Sent	LGK8	9/12	10:19p	Video	Received	LGK8
9/12	10:14p	Video	Sent	LGK8	9/4	6:52p	Text Message	Received	iPhone 4S
9/4	7:52p	Text Message	Sent	iPhone 4S	9/12	8:54p	Text Message	Received	iPhone 4S
9/4	7:57p	Phone Call out	Sent	iPhone 4S	9/12	8:58p	Image	Received	iPhone 4S
9/11	10:04p	Text Message	Sent	iPhone 4S	9/12	9:02p	Text Message	Received	iPhone 4S
9/12	8:55p	Text Message	Sent	iPhone 4S	9/20	07:46p	Image (set to disappear/5min)	Received	iPhone 7
9/12	8:59p	Text Message	Sent	iPhone 4S	9/20	06:53p	Text Message	Received	iPhone 7
9/12	10:16p	Text Message	Sent	iPhone 4S	10/19	10:19p	Video	Received	iPhone 7
9/11	06:08p	Text Message	Sent	iPhone 7	10/20	07:20p	Text Message	Received	iPhone 7
9/12	09:02p	Text Message	Sent	iPhone 7	9/12	8:58p	Image	Received	LGK8
9/12	09:30p	Text Message	Sent	iPhone 7	9/12	9:02p	Text Message	Received	LGK8
9/14	04:30p	Video	Sent	iPhone 7	9/20	07:46p	Image (set to disappear/5min)	Received	LGK8
9/14	06:30p	Text Message	Sent	iPhone 7	9/20	06:53p	Text Message	Received	LGK8

Table 1 ZTE Z993 Log of messages

In order to properly examine the data that was extracted from all four devices, open source tools were also used to achieve the best possible results. There are multiple types of open source mobile tools. The open source tools being used were Autopsy and iPhone Analyzer. The Autopsy tool supports parsing commonly missed items from Android devices, while giving faster access to the File System directory. iPhone Analyzer extracts backups, photos, SMS messages, and GPS information from iOS devices. Another open source tool being used was iExplorer. iExplorer enables the user to examine the contents of their own device from a back-up. This also gives the user to be able to save voicemails, messages, or even call logs. Signal gives an option to merge the messaging with the mobile device messaging, if user choses this option, the iExplorer can recover the chat logs. A limitation that may be encountered when using open source tools is that they are not fully tested or validated and may miss some of the data that could be useful in the examination. The purpose was to see which of these tools will still acquire data, if any.

Results

The results of this study vary by the make and model of the device. Some of them were surprising, while others left the examination with more questions. Out of all four devices, the Z993 Prelude (ZTE Android) phone was the only one that allowed a physical extraction. When the physical extraction was completed on the ZTE device, it showed almost everything within Signal sent and received by the other devices (*Figure 13)*. One of the surprising factors was that a few of the messages came up labeled as TextSecure instead of Signal (*Figure 19)* within the filesystem extraction results. TextSecure was the original name prior to the merger to create Signal.

Another raised questions was on the deleted message, it gives a timestamp, showed a read status, who sent and who received, yet shows it unsent on the information panel on the right side of UFED 4 PC version 6.3.1.477. Also, on the deleted message it shows that it was delivered and read, yet on the main screen of UFED 4 PC version 6.3.1.477 it shows unread as shown in *Figure 17*. While the ZTE Android did provide more information, getting the extractions to complete required many more steps than working with an iPhone. Some of the extractions on the Android devices required the mobile device to have specific settings unlocked or locked in order to obtain the extractions. For example, on the ZTE, the USB debugging option, developer tools, stay awake mode must be turned on, while, the screen lock mode needs to be turned off. Cellebrite tools gave exact directions on which device had to have these types of options on or off depending on which version of Android you are using. The menu has listings for Android 4.2 and higher and Android 4.0 – 4.1.2.

The Android ZTE Z993, did produce some results with the open source version of Autopsy. Autopsy recovered data concluding the application does exist on the device, the phone numbers accessed and messaged, and one of the images that was sent within a Signal message was uncovered (*Figure 21)*. It also showed, an event occurred assigning the safety numbers, but did not reveal the actual numbers used from Signal to the other user (*Figure 22)*. Autopsy has a commercial version as well and much more data could have been retrieved having used the commercial version. Overall, Autopsy was the open source tool that recovered the most data from the phones used in this study. Autopsy parsed the data and found missed items that Cellebrite did not locate. For example, there was an image sent through a Signal message that showed up in UFED 4PC version 6.3.1.477, but did not show the actual image. In Autopsy, that image was revealed with a timestamp, yet not message information (*Figure 20)*. The iPhone Analyzer and iExplorer are not compatible with Android devices to do any types of extractions.

In *Tables 2 through 9*, the results of the findings are broken down by the type of extraction completed, the make of the device and the commercial or open source tool used for that extraction. The findings listed are only the ones associated with the Signal Application. The extractions provided an abundance of information, but since it was not pertinent to the research of the Signal application, it was not included in these results.

ZTE Z993				Recovered			
Date	Time (UTC-5)	Type of Message	Sent/Received	Cellebrite	Autopsy	iExplorer	iPhone Analyzer
9/4	6:42pm	Text	Sent	✓		N/A	N/A
9/4	6:44pm	Image (deleted)	Sent	Missing Image but shows deleted		N/A	N/A
9/4	6:46pm	Text	Sent	✓		N/A	N/A
9/4	6:48pm	Text	Received	✓		N/A	N/A
9/4	6:52pm	Text	Received	✓		N/A	N/A
9/4	6:53pm	Text	Received	✓		N/A	N/A
9/4	6:55pm	Image	Sent	Image did not show only timestamp	Shows missing image only	N/A	N/A
9/4	7:20pm	Text	Received	✓		N/A	N/A
9/4	7:46pm	Image (set to delete)	Received			N/A	N/A
9/4	7:52pm	Text	Sent	✓		N/A	N/A
9/4	7:57pm	Phone Call	Sent	✓		N/A	N/A
9/11	6:08pm	Text	Sent	✓	Shows phone number only no message data	N/A	N/A
9/11	10:04pm	Text	Sent	✓	Shows phone number only no message data	N/A	N/A
9/11	11:03pm	Text	Sent	✓	Shows phone number only no message data	N/A	N/A

Table 2 ZTE Z993 Results

ZTE Z993				Recovered			
Date	Time (UTC-5)	Type of Message	Sent/Received	Cellebrite	Autopsy	iExplorer	iPhone Analyzer
9/12	6:48pm	Text (article link)	Sent	✓		N/A	N/A
9/12	7:32pm	Image	Sent	✓		N/A	N/A
9/12	8:54pm	Text	Received	✓		N/A	N/A
9/12	8:55pm	Text	Sent	✓		N/A	N/A
9/12	8:58pm	Image	Received	✓		N/A	N/A
9/12	8:59pm	Text	Sent	✓		N/A	N/A
9/12	9:02pm	Text	Sent	✓		N/A	N/A
9/12	9:03pm	Text	Received	✓		N/A	N/A
9/13	9:30pm	Text	Sent	✓		N/A	N/A
9/13	10:14pm	Video	Sent	✓		N/A	N/A
9/13	10:16pm	Text	Sent	✓		N/A	N/A
9/13	10:19pm	Video	Received	✓		N/A	N/A
9/20	6:55pm	Text	Received	✓		N/A	N/A
9/20	7:46pm	Image & Text (set to delete)	Received			N/A	N/A
10/19	10:19pm	Video	Received	✓		N/A	N/A
10/20	7:20pm	Text	Received	✓		N/A	N/A

Table 3 ZTE Z993 Results

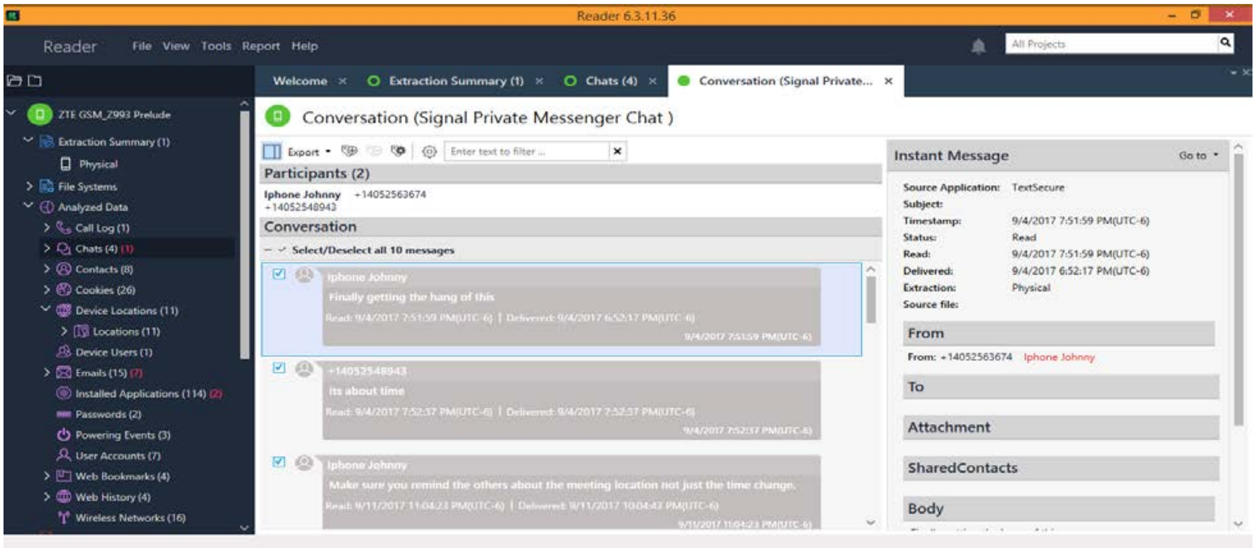


Figure 13 Physical Extraction Results ZTE Z993 Prelude

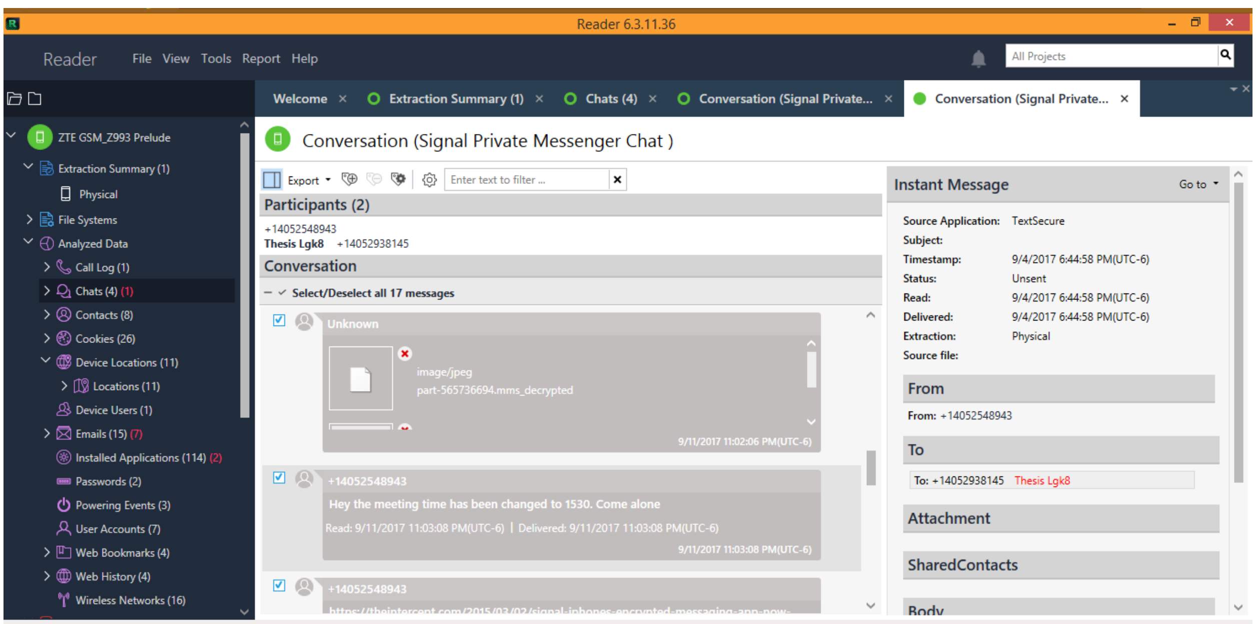


Figure 14 Deleted message shown without the actual message ZTE Z993 Prelude

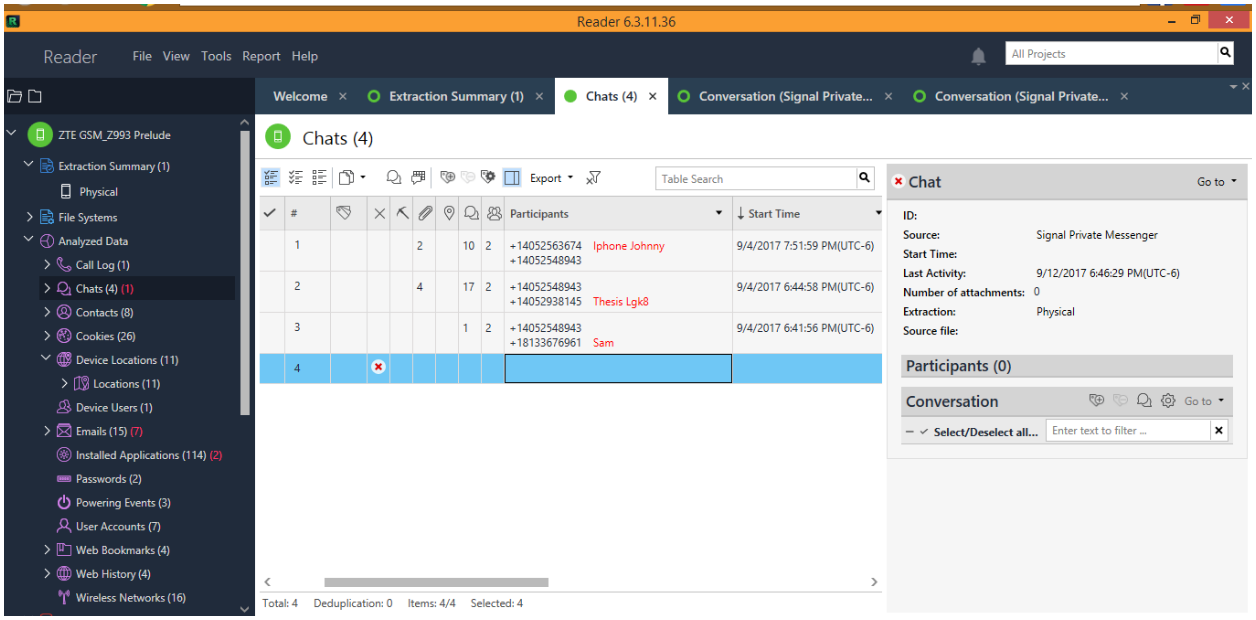


Figure 15 Deleted chat but no message uncovered ZTE Z993 Prelude

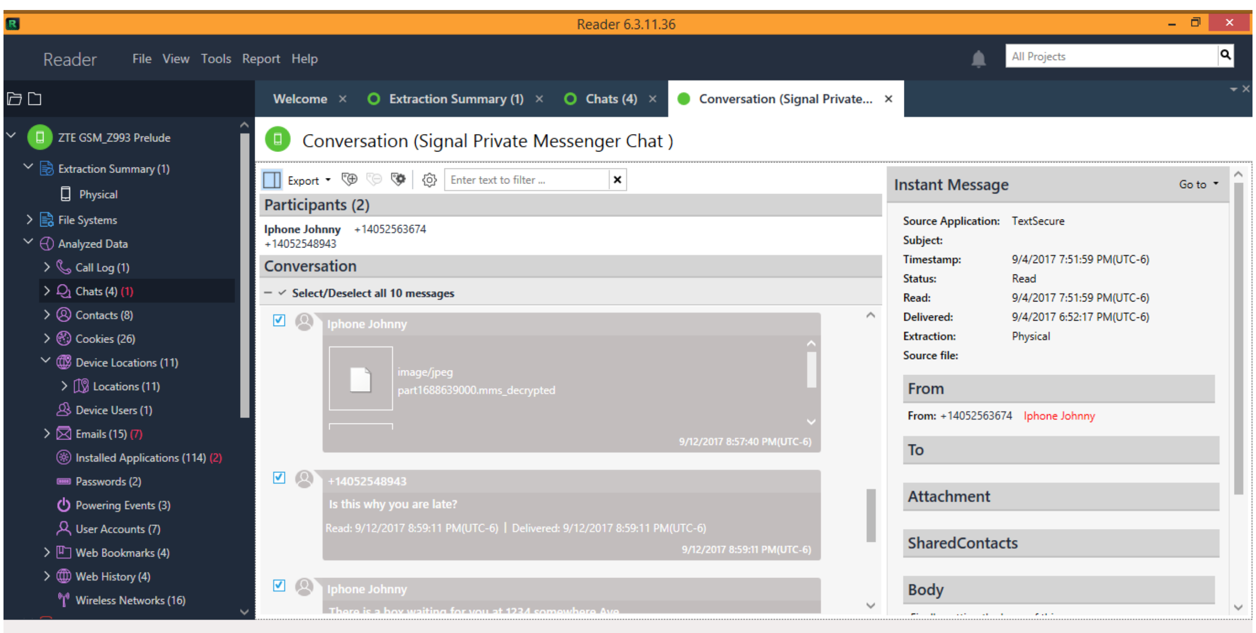


Figure 16 Signal message missing the image ZTE Z993 Prelude

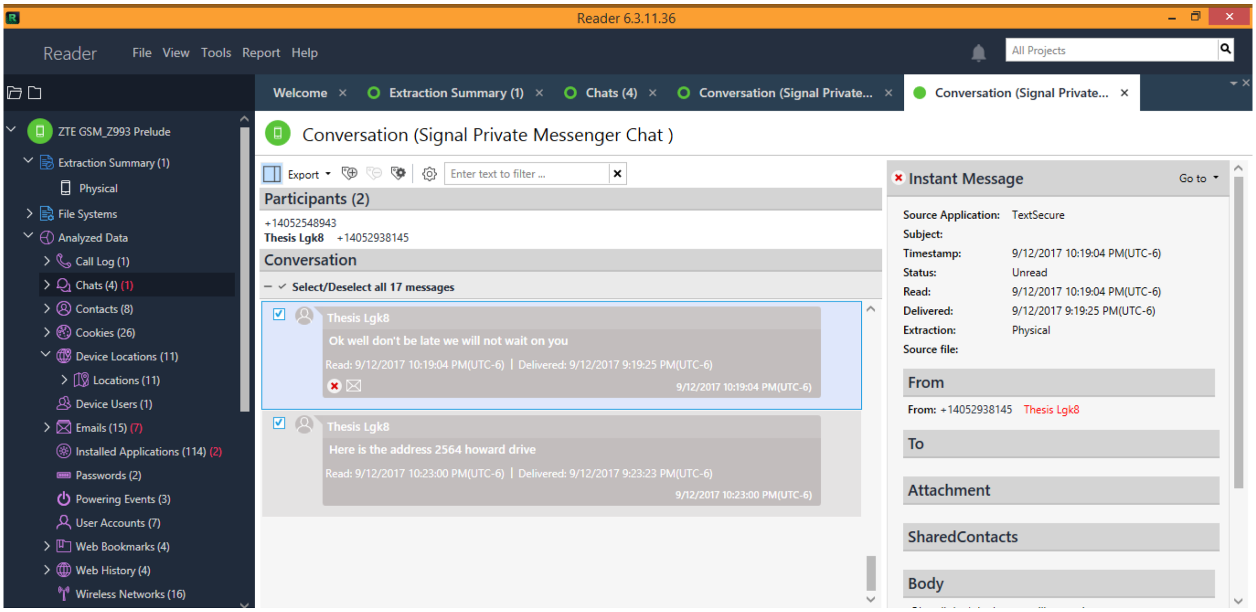


Figure 17 Shows deleted message delivered and read but unread on the right ZTE Z993 Prelude

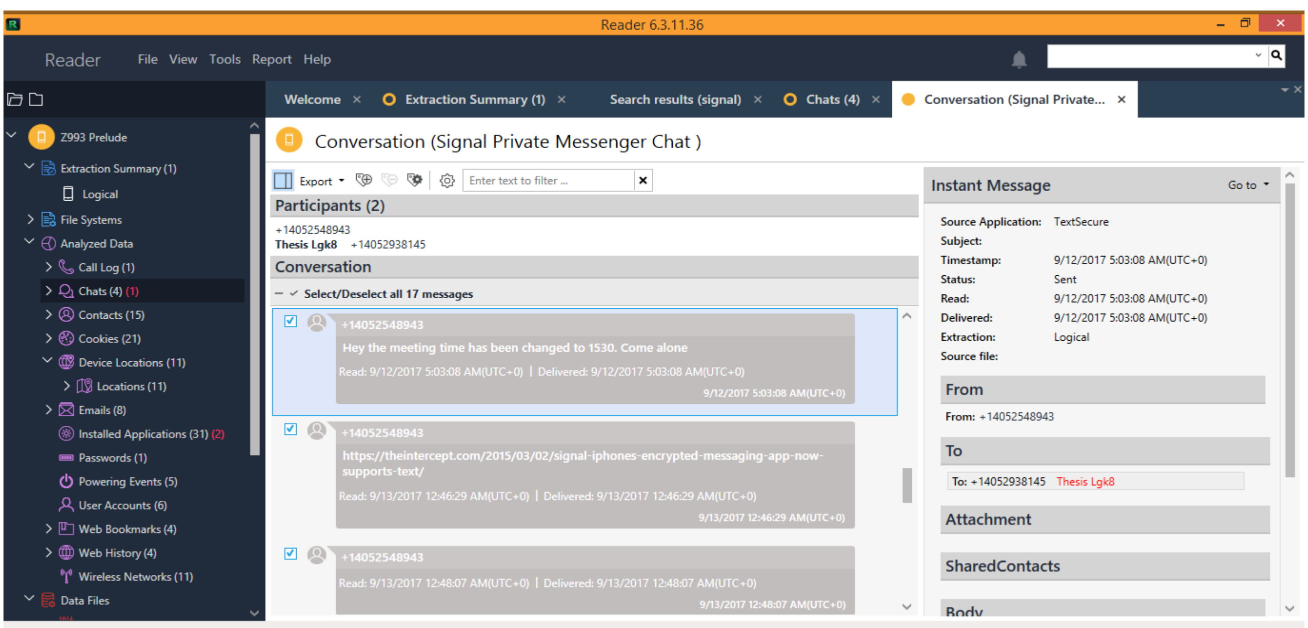


Figure 18 Logical Extraction Results ZTE Z993 Prelude

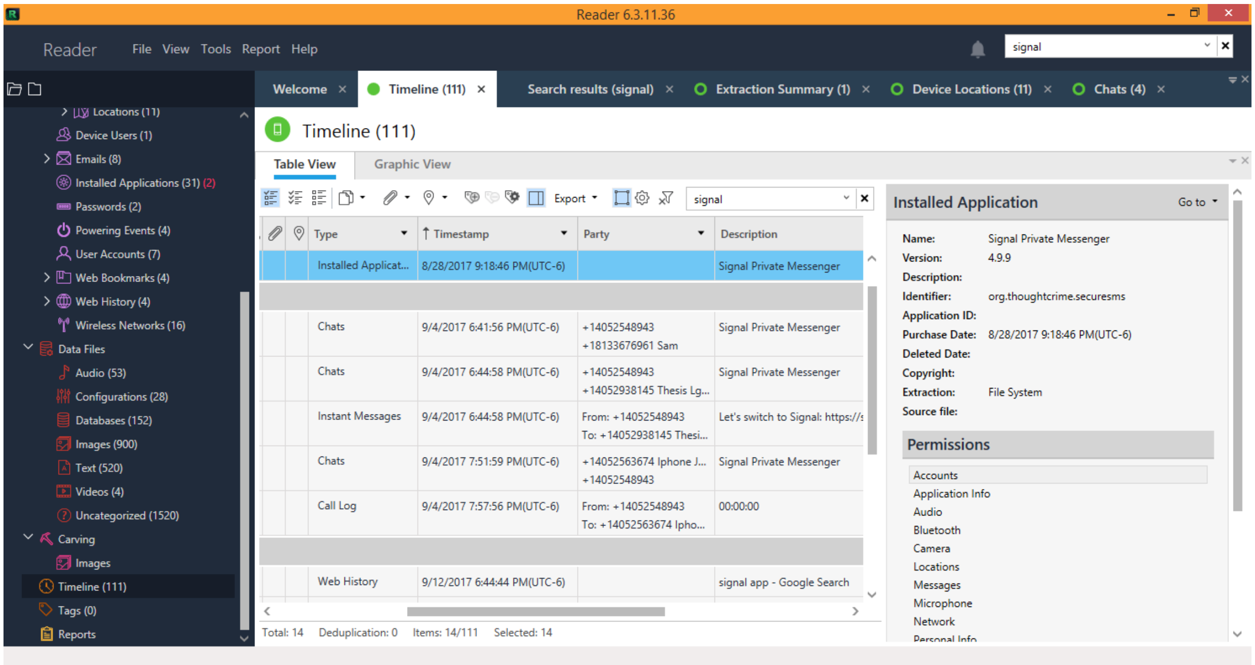


Figure 19 Filesystem Extraction Results Chats and call inside of Signal App on timeline ZTE Z993

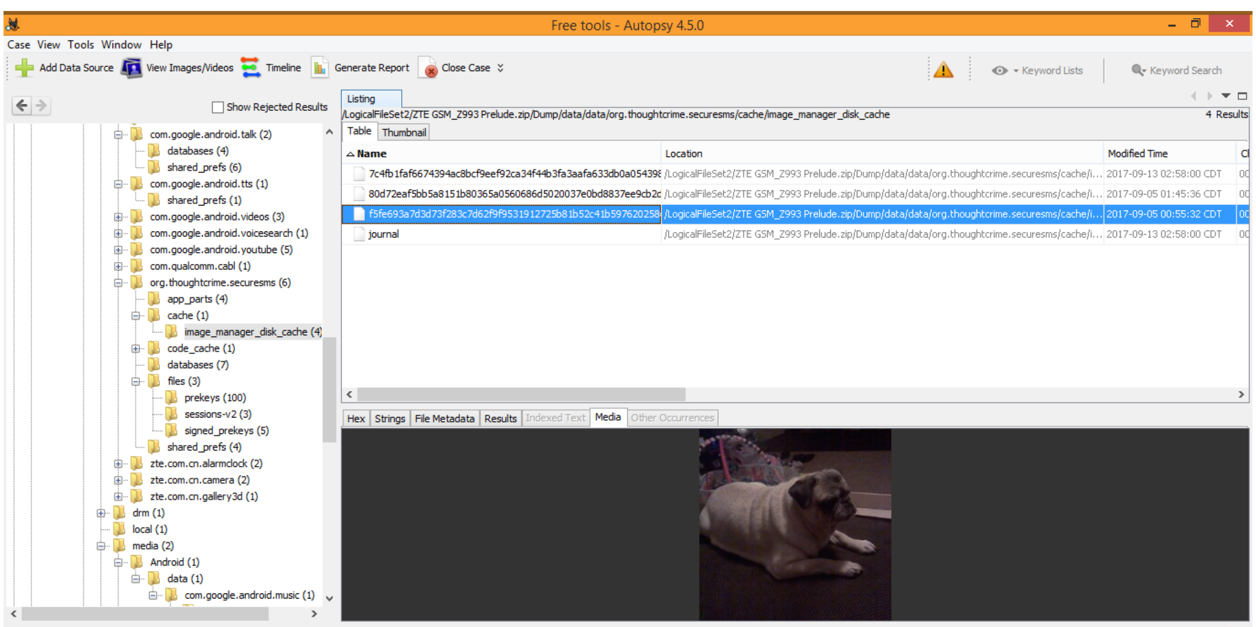


Figure 20 Autopsy showing picture data inside Signal Application with ZTE Z993

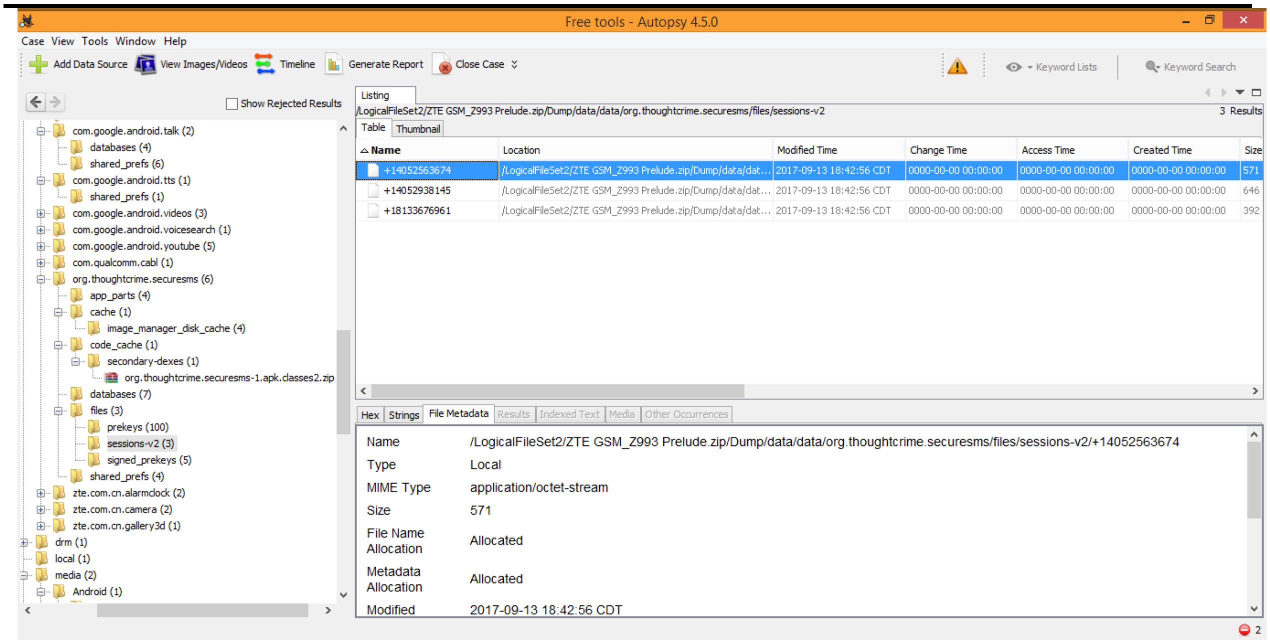


Figure 21 Autopsy showing phone numbers but not messages inside Signal Application with ZTE Z993

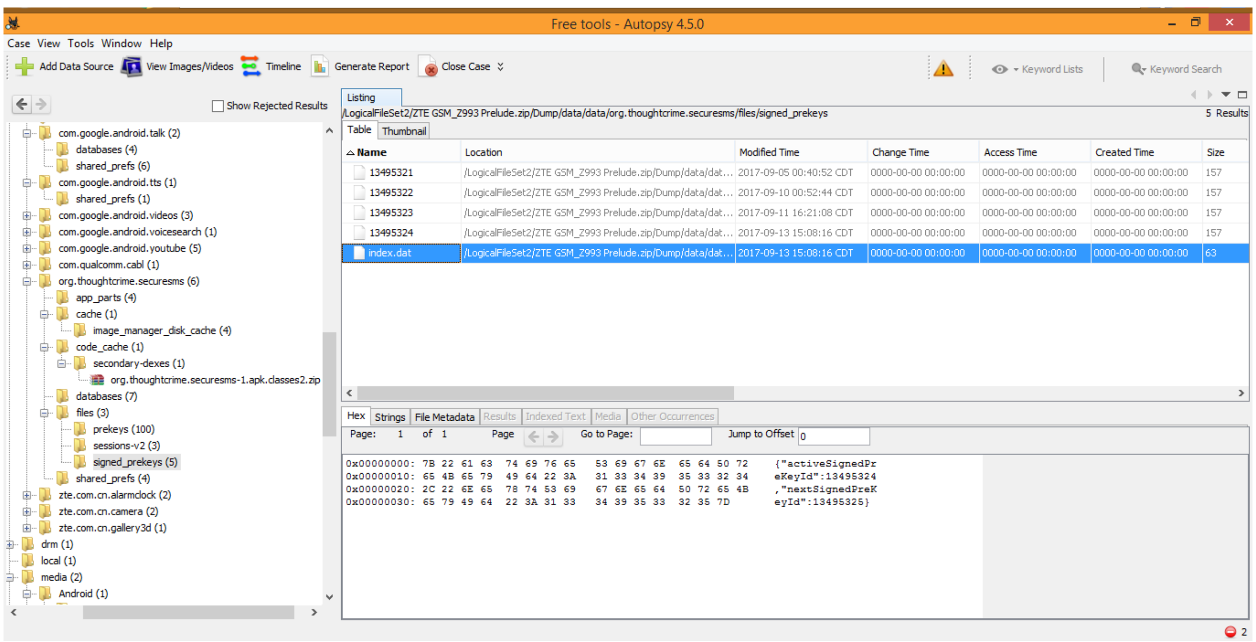


Figure 22 Autopsy showing assignment of pre key but not the key inside Signal Application with ZTE Z993

In *Table 4 and 5,* the data recovered from the logical and filesystem extractions are listed. Physical extractions were not supported for the LGUS375 Android device. In *Figure 23*, the extraction shows that the application was installed, but contained no further data regarding the messages sent or received. Once again, if a physical extraction would have been available then an exact copy of the mobile device could have yielded better results. *Figure 24*, the filesystem extraction recovered and equal amount of data as in the logical extraction. The only information shown was that the application was installed. Both iPhone and Android users have the option for storage and backups. Google Drive is the storage option used for Android devices. The access to this drive would only be available to examiners with the additional Cloud Analyzer feature which was not available for this study.

LG US375 K8				Recovered			
Date	Time (UTC-5)	Type of Message	Sent/Received	Cellebrite	Autopsy	iExplorer	iPhone Analyzer
9/4	6:42pm	Text	Sent			N/A	N/A
9/4	6:45pm	Text	Sent			N/A	N/A
9/4	6:46pm	Text	Sent			N/A	N/A
9/4	6:48pm	Text	Received			N/A	N/A
9/4	6:50	Image	Sent			N/A	N/A
9/4	6:53pm	Text	Received			N/A	N/A
9/4	6:55pm	Image (set to delete)	Sent			N/A	N/A
9/4	7:20pm	Image	Received			N/A	N/A
9/4	7:52pm	Text	Sent			N/A	N/A
9/4	7:57pm	Image (set to delete)	Received			N/A	N/A
9/11	6:18pm	Text	Received			N/A	N/A
9/11	10:34pm	Text	Sent			N/A	N/A
9/12	7:27pm	Image	Received			N/A	N/A
9/12	7:45pm	Text	Received			N/A	N/A
9/12	11:03pm	Text	Sent			N/A	N/A
9/12	11:16pm	Image	Sent			N/A	N/A

Table 4 LG US375 K8 Results

LG US375 K8				Recovered			
Date	Time (UTC-5)	Type of Message	Sent/Received	Cellebrite	Autopsy	iExplorer	iPhone Analyzer
9/12	11:22pm	Video	Received			N/A	N/A
9/13	9:05am	Text	Sent			N/A	N/A
9/13	9:13am	Text	Received			N/A	N/A
9/13	9:30am	Text	Received			N/A	N/A
9/13	7:30pm	Text	Sent			N/A	N/A
9/13	10:16pm	Image & Text (set to delete)	Sent			N/A	N/A
9/13	10:19pm	Video	Sent			N/A	N/A
9/20	6:55pm	Text	Received			N/A	N/A
9/20	7:46pm	Image & Text (set to delete)	Received			N/A	N/A
9/20	8:24pm	Phone Call	Sent			N/A	N/A
9/20	8:43pm	Text	Received			N/A	N/A
10/19	10:19pm	Video	Received			N/A	N/A
10/20	7:20pm	Text	Received			N/A	N/A

Table 5 LG US375 K8 Results

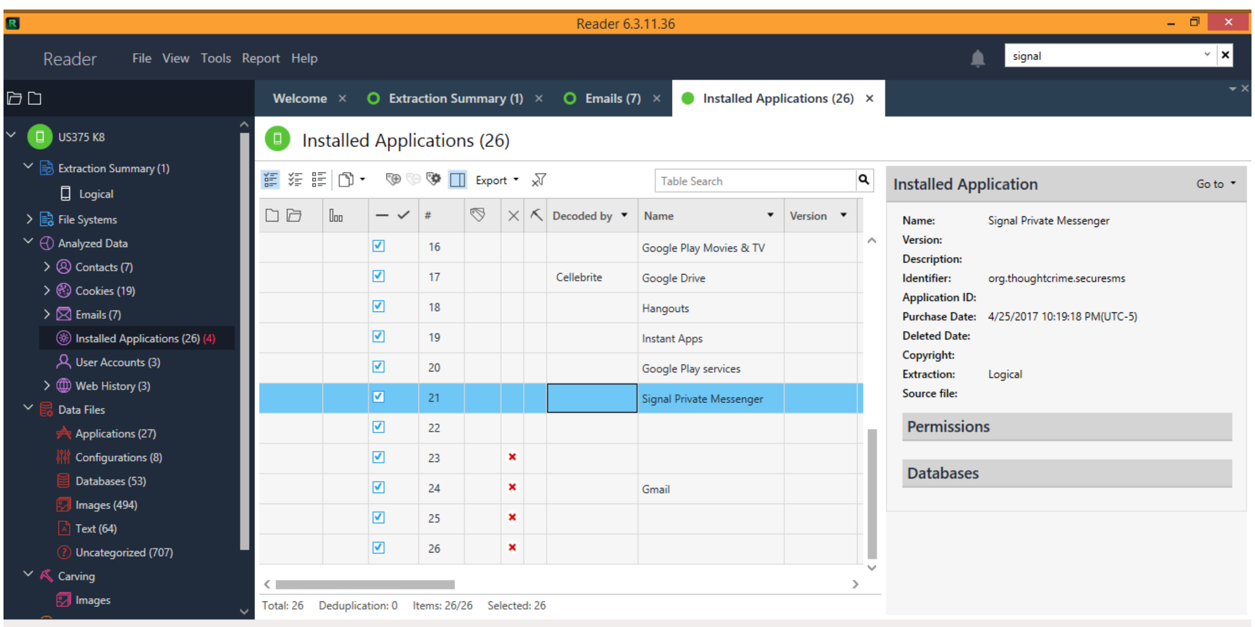


Figure 23 LG Logical

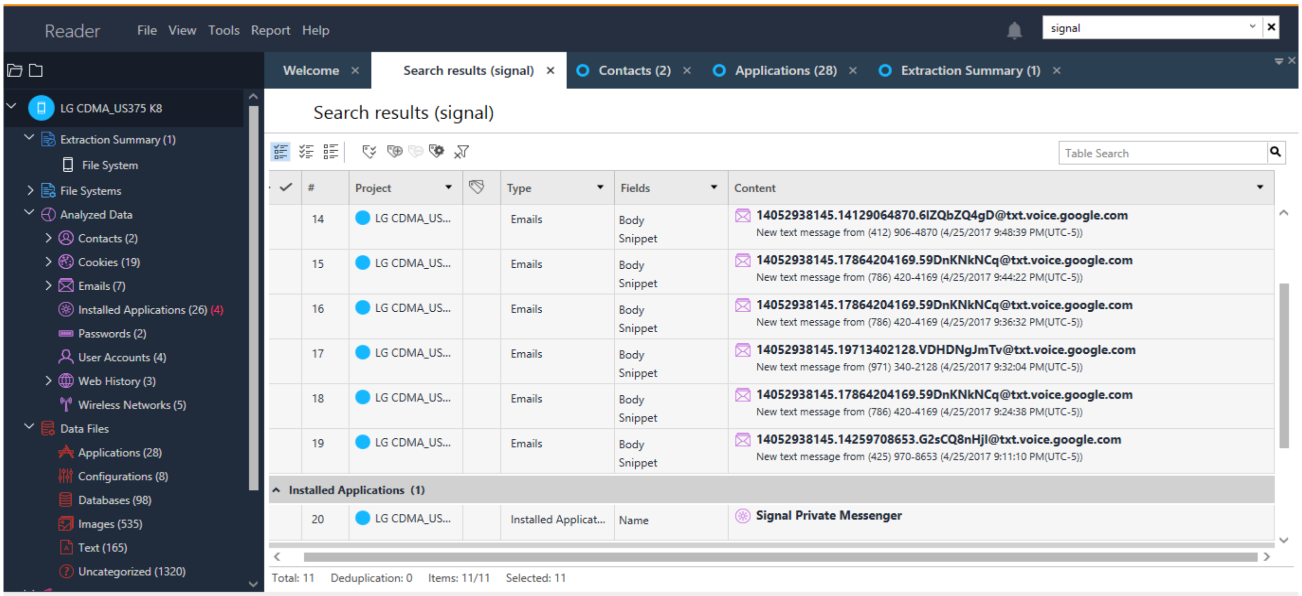


Figure 24 LG File System

The physical extractions were not supported for neither the iPhone 4S or the iPhone 7. Since a physical extraction by definition acquires more data from the device, it could be assumed, especially with the iPhones, that there could have been more relevant data recovered. The iPhone 4S and iPhone 7 data did not produce the results that were expected. Neither the commercial nor the open source tools were helpful in retrieving the messages, images, and video, sent between the devices. In all available extractions for the iPhone 4S and iPhone 7, the application was shown as installed, but the actual data within the application was not visible. IPhone’s can automatically back up the data from Signal to iCloud, but this is also a feature that can be manually turned off, as in this case.

The iCloud is iPhone’s version of a storage drive. Although, Signal can back up data to the user’s iCloud account, that doesn’t mean it is retrievable with current commercial forensic tools. Currently, the UFED Cloud Analyzer that is available does not support gathering data from the Signal application. The interesting part of the supported devices, is that the WhatsApp is supported which was developed using the Signal encryption (Cellebrite, Cloud Analyzer 2017). *Tables 6 through 9* show the breakdown of what was found using both the commercial tools and open source tools.

iPhone 4S				Recovered			
Date	**Time (UTC-5)**	**Type of Message**	**Sent/Received**	**Cellebrite**	**Autopsy**	**iExplorer**	**iPhone Analyzer**
9/4	5:30pm	Text	Received				
9/4	6:02pm	Image (deleted)	Sent				
9/4	6:16pm	Text	Sent				
9/4	6:25pm	Text	Received				
9/4	6:52pm	Text	Sent				
9/11	3:45pm	Text	Received				
9/11	3:50pm	Image	Received				
9/11	6:22pm	Text	Sent				
9/11	6:54pm	Image (set to delete)	Received				
9/11	7:02pm	Text	Sent				
9/11	7:37pm	Image	Sent				
9/11	8:20pm	Text	Received				
9/11	8:28pm	Text	Sent				
9/11	9:15pm	Text	Received				

Table 6 iPhone 4S Results

iPhone 4S				Recovered			
Date	**Time (UTC-5)**	**Type of Message**	**Sent/Received**	**Cellebrite**	**Autopsy**	**iExplorer**	**iPhone Analyzer**
9/12	6:48pm	Image	Sent				
9/12	7:32pm	Image	Sent				
9/12	8:54pm	Text	Received				
9/12	8:55pm	Text	Sent				
9/12	8:58pm	Image	Received				
9/12	8:59pm	Text	Sent				
9/12	9:12pm	Text	Sent				
9/12	9:23pm	Text	Received				
9/20	9:40pm	Text	Sent				
9/20	10:14pm	Video	Sent				
9/20	10:16pm	Text	Sent				
9/20	10:19pm	Video	Received				
9/20	6:55pm	Text	Received				
9/20	7:46pm	Image & Text (set to delete)	Sent				
10/19	10:19pm	Video	Received				
10/20	7:20pm	Text	Received				

Table 7 iPhone 4S Results

iPhone 7				Recovered			
Date	Time (UTC-5)	Type of Message	Sent/Received	Cellebrite	Autopsy	iExplorer	iPhone Analyzer
9/11	3:30pm	Text	Received				
9/11	4:02pm	Image (deleted)	Sent				
9/11	4:18pm	Text	Sent				
9/11	4:22pm	Image	Received				
9/11	4:35pm	Text	Sent				
9/11	5:02pm	Text	Received				
9/11	5:10pm	Image	Received				
9/11	5:30pm	Text	Sent				
9/11	5:52pm	Image	Received				
9/11	6:03pm	Text (set to delete)	Sent				
9/11	6:31pm	Text	Sent				
9/12	6:40pm	Text	Received				
9/12	6:42pm	Text	Sent				
9/12	7:00pm	Text	Received				

Table 8 iPhone 7 Results

iPhone 7				Recovered			
Date	Time (UTC-5)	Type of Message	Sent/Received	Cellebrite	Autopsy	iExplorer	iPhone Analyzer
9/12	6:48pm	Image	Sent				
9/12	7:32pm	Image	Sent				
9/12	8:54pm	Text	Received				
9/12	8:55pm	Text	Sent				
9/13	9:30pm	Text	Received				
9/13	10:14pm	Video	Received				
9/13	10:16pm	Text	Received				
9/13	10:19pm	Video	Sent				
9/20	9:40pm	Text	Sent				
9/20	10:14pm	Video	Sent				
9/20	10:16pm	Text	Sent				
9/20	10:19pm	Video	Received				
9/20	6:55pm	Text	Received				
9/20	7:46pm	Image & Text	Sent				
10/19	10:19pm	Video	Received				
10/20	7:20pm	Text (set to delete)	Sent				

Table 9 iPhone 7 Results

With the iPhone, there were no features that had to be turned on or off in order to complete the extraction. The examiner only had to allow trust between the device and the computer. Two advanced logical extractions were completed with Physical Analyzer version 6.3.11.36 on the iPhone 4S, but the iPhone 7 was not supported. Once all the extractions were completed with the available commercial tools, UFED 4 PC version 6.3.1.477 and UFED Physical Analyzer version 6.3.11.36, the open source tools were utilized. The open source tools used in this study were iPhone Analyzer, iExplorer and Autopsy. There was no surprise by the small amounts of data that was recovered using the open source tools. Some of these tools also have a pay option which could expand the results, but not definitive to the Signal Application.

With iPhone Analyzer, iPhone 4S only shows the Signal application files and libraries on the device, yet messages, emails, contact logs, contact lists, or images were included. The iTunes backup of the iPhone 4S and iPhone 7 were the files used to analyze and recover with iPhone Analyzer. The only view was that it was installed onto the device (*Figures 29*). The factory applications were also revealed. The iPhone 7 produced similar results, but also had SMS messages from the original messaging source, not Signal. Signal gives the option to link the SMS messaging already on the mobile device with the Signal Application. If this occurs, then it could be expected to see all of the messaging due to the results showing all other messages located on the device.

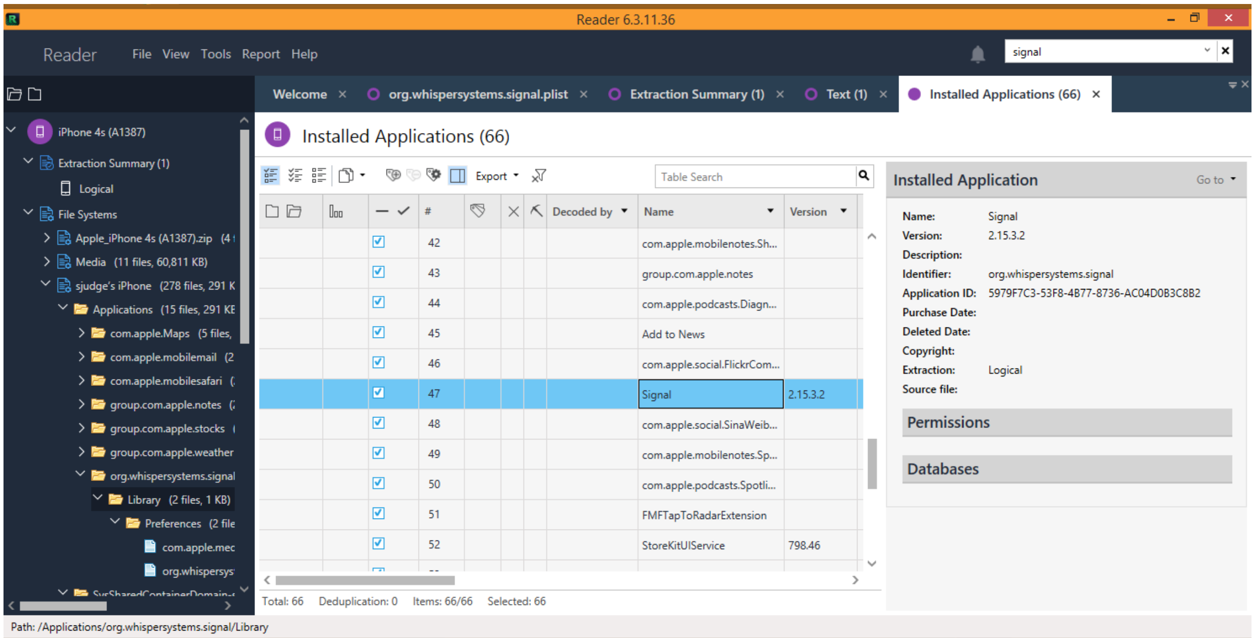


Figure 25 iPhone 4S Logical Extraction showing installed Signal Application

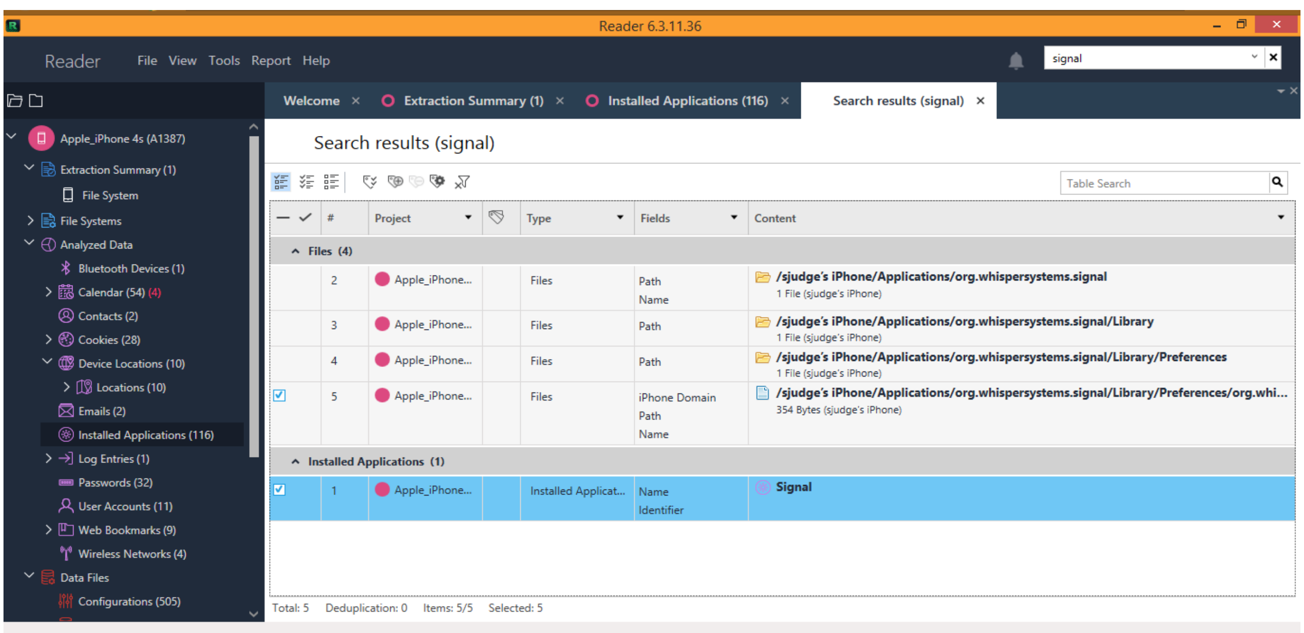


Figure 26 iPhone 4S Filesystem Extraction showing installed Signal Application

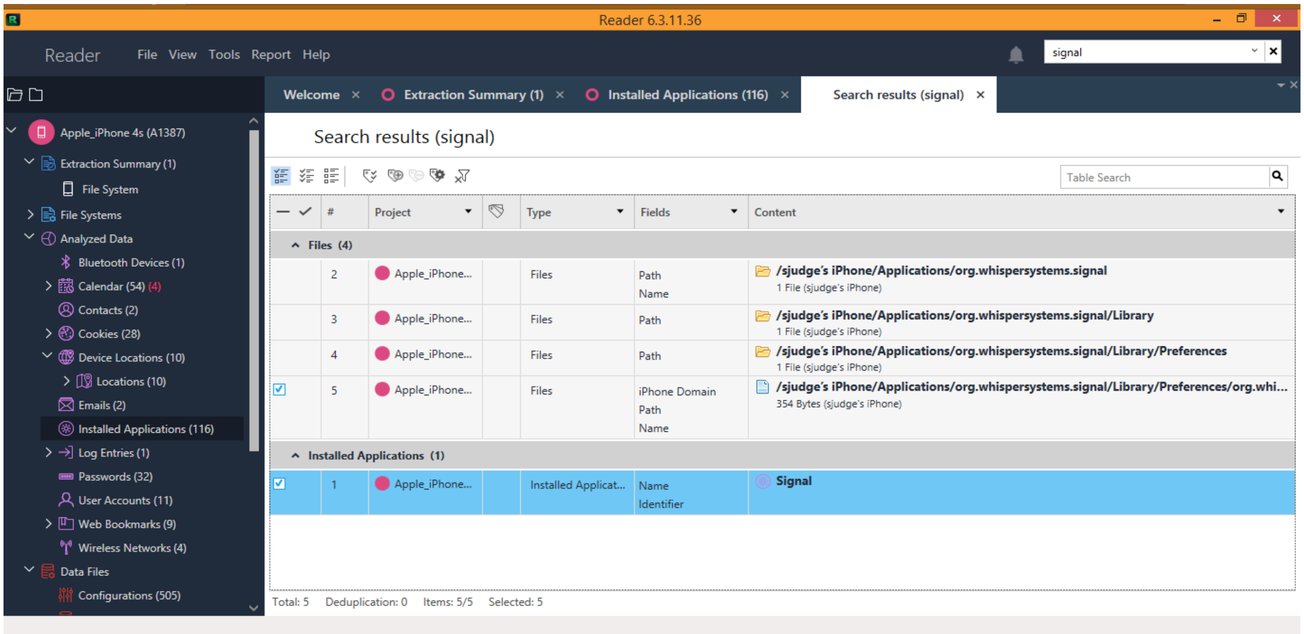


Figure 27 iPhone 4S Advanced Logical Extraction showing app installed

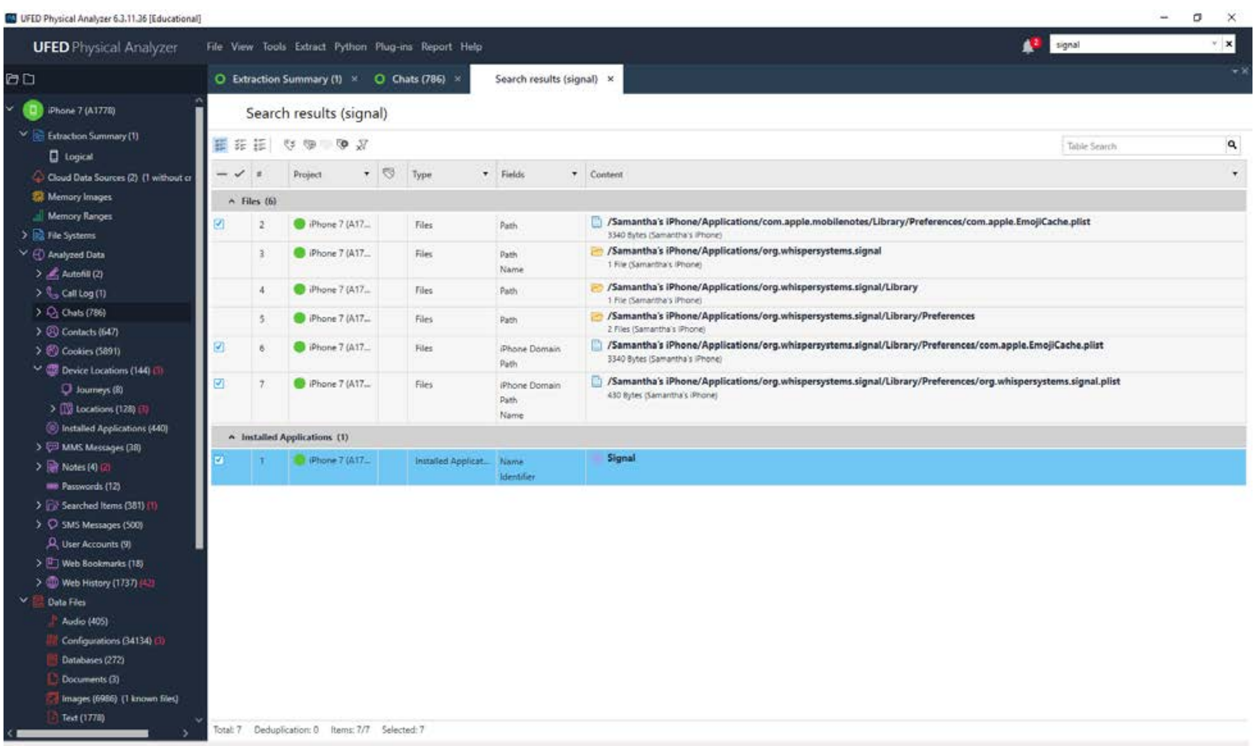


Figure 27 iPhone 7 Logical Extraction showing installed Signal Application

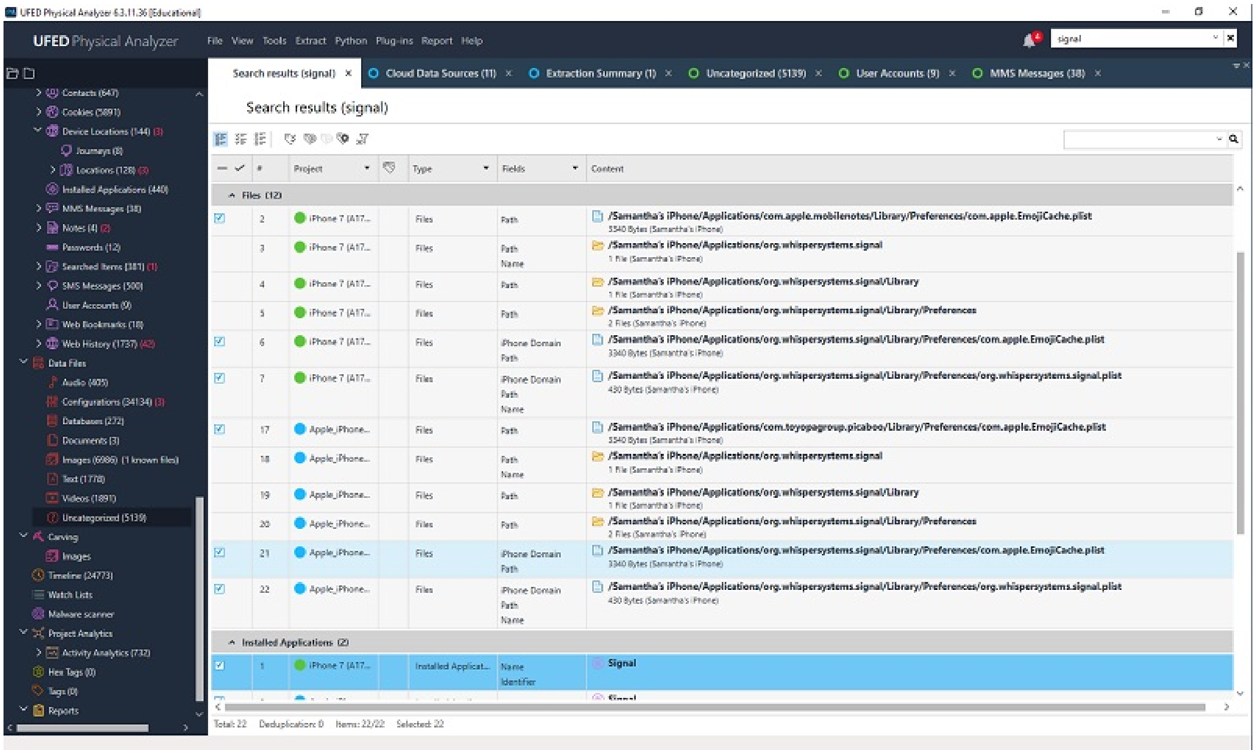


Figure 28 iPhone 7 Filesystem Extraction showing installed Signal Application

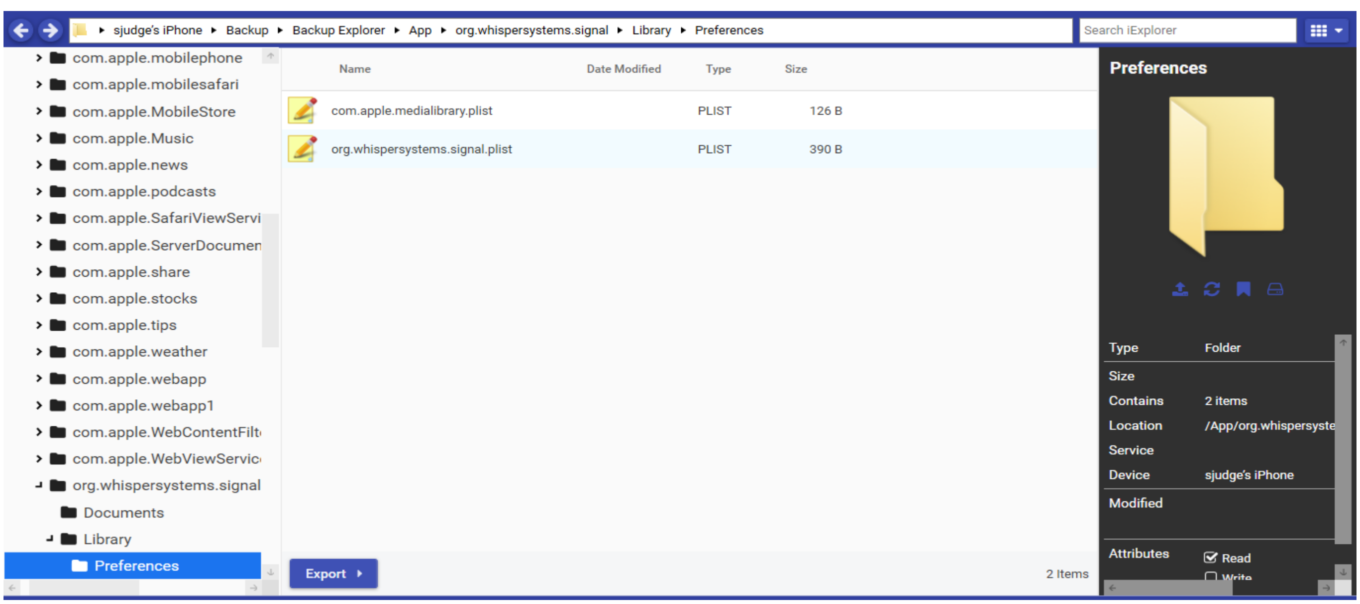


Figure 29 iPhone Explorer iPhone 4S and iPhone 7

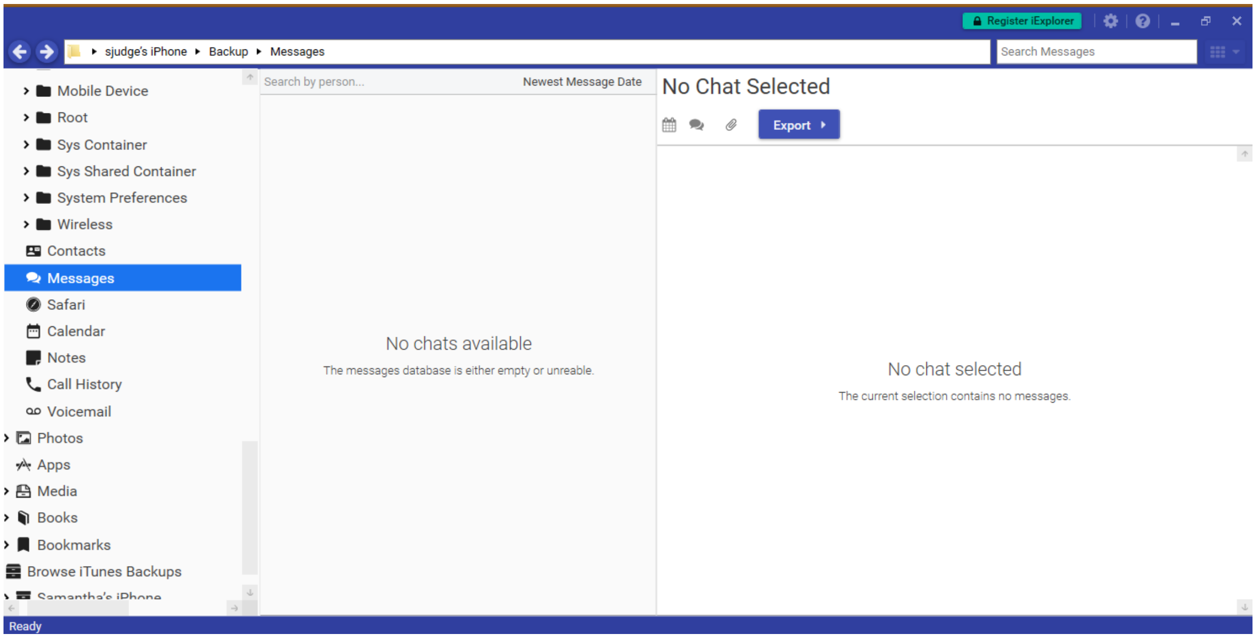


Figure 30 iExplorer showing no messages or chats with iPhone4S and iPhone7

Overall, the results were less than promising. Thus far, the amount of information that was received from the device were absolutely dependent on the type of device and the type of extraction that is available. The physical extraction yielded the most results compared to the file system and logical extractions. When it came down to which operating system would provide more information, in this study, it was the Android. More research is needed to determine if there would have better results extracting the information from the backups located on the iCloud accounts. Currently the UFED Cloud Analyzer is not compatible with the Signal data source. Future research could be completed with updates to the Cellebrite software and available data sources. The only phone that displayed almost everything from the installation, to all of the conversations between the four devices was the ZTE Z993 Prelude with the Cellebrite tools. This was the only phone that supported a physical extraction, which could answer why it had the most results recovered. The ZTE Z993 was one of the older versions out of all four devices, yet it compiled higher results than the three other phones. This is a clear example of the types of phones supported by Cellebrite, and what the software, security, and model will allow the examiner to recover.

The point of this study was to see the volume of data, if any, could be extracted. The other reasons were to find out which brand of phone had more discoverable data, in this case it was the Android phone, ZTE Z993 Prelude that produced the most information. Although, with the ZTE Z993, the images and actual text were missing that were set to disappear or had been manually deleted, the timestamp and date were still recoverable along with many other messages in that particular chat session within the Cellebrite tools. One of the questionable results that were obtained was why the text messages that did show, come up as the prior name for the company? Cellebrite also showed some of the messages were delivered and sent, yet on the information panel they show unsent or unread in some cases. This occurred on both of the Android devices, and only on the commercial software from Cellebrite. For this study, only three open source tools were selected, but all had the potential to become commercial tools with more features and options if the user decided to pay additional fees. If the commercial versions of these tools were used then the results may have been different. In the future, the commercial version of Cellebrite could offer physical extractions regardless of brand or version of the iPhone which would help tremendously with any future extractions. The physical extraction will always give the majority of the data that is needed when reviewing data within applications.

Chapter III

Literature Review

Over the years, more and more open source messaging applications have become readily available. One of the articles that was reviewed was by Shubham Sahu (2014). The purpose of the article was focused on extracting useful data from the WhatsApp along with similar applications installed on an Android platform device. The depth of the study covered the extraction of the data and the tool available in order to decrypt and organize SQLite database files. The article also reviewed both an older version of the application and the most recent upgrade.

One of the main reasons many people move to these new messaging applications is the fact that there are no restrictions on the length of the message or any fees involved. Originally the WhatsApp messages were being stored in the *msgstor.db* file. This posed a serious security risk due to the messages, including deleted ones, in its entirety could be retrieved. Currently, according to WhatsApp officials, the database has a custom AES encryption algorithm with above a 192 bit encryption key mainly used in the WhatsApp Android platform. WhatsApp data is stored in the internal memory of the mobile device. Once the application is installed, it synchronizes with the users contacts, and shows other users who have the application installed. When the mobile device is turned on after installation, the *com.whatsapp* sends a signal to start the *ExternalMediaManage* and *MessageService* services, which run in the background.

The accelerated increase in open source messaging applications over the last five to eight years has enabled users to communicate in ways far more advanced than we could have ever imagined. The next article reviewed was by Aditya Mahajan, M.S. Dahiya, and H.P. Sanghvi (2013), focused on two messaging applications. The first application is the WhatsApp and the second is the Viber. “People are constantly exchanging information like images, videos, activities, and events” (Mahajan et al., 2013). With all the advancements in technology, the security of the same people consistently chatting and exchanging information is becoming more and more vulnerable. Criminals are also catching onto the fact that fast deleting applications are now a haven for criminal activity and anonymity. In 2013, when this article was published, WhatsApp had already had over a million downloads on Google Play. Viber was sitting at one hundred and forty million.

The overall focus of this article was to show the forensic examination of data and information stored by these two applications, as well as, the data extraction tools and techniques used. WhatsApp focuses more on the exchanges of text, video, and audio messages. They also enable the user to have group communication. The Viber application is mainly for free calling and free texting only. Some of the limitations we face with Android include the difficulty in accessing and extracting the data needed to do a forensic examination. This can become more difficult if the information is encrypted or has been deleted from the device. Often the device is connected to the internet as well so it can be remotely wiped if the owner wishes.

Sahu extracted the database file from an Android device. The file msgstor.db.crypt was retrieved in the acquisition yet a problem arose because of its encryption. In order to properly decrypt the file, Sahu used a tool created by Francesco Picasso called the WhatsApp Xtract. “A python script uses this same key to decrypt the encrypted db file and presents the result in a well-organized HTML page”. (Sahu, 2014) This decryption is completed on the WhatsApp 2.11.186, at the time, the most recent version. An alternative to this method was to read the database files entirely through the SQLite browser. The location examined for an Android platform is */sdcard/WhatsApp/Databases/msgstore.db.crypt* and for an iOS platform it is located at *Application/net.whatsapp.WhatsApp/Documents/ChatStorage.sqlite*. Sahu included how to properly install and run the WhatsApp Xtract tool. Listed below are the steps included:

How to use: (Sahu, 2014)

Step 1:Download WhatsApp Xtract package on your computer and extract it.

Step 2: Download and install Python programming language environment on your computer.

Step 3: Open the folder where you downloaded the WhatsApp Xtract archive. Find a file with name *!install pyCrypto.bat*, right click on it and click run as administrator. This bat file will execute the following Python command, *pypm install pycrypto*. This common automatically installs the pycrypto library on your computer, which will be used to decrypt the WhatsApp backup data.

Step 4: In the same folder, run either *whatsapp_xtract_iphone.bat*, *whatsapp_xtract_android_crypted.bat or whatsapp_xtract_android.bat*

Depending upon the backup file you used. To run any of these files, simply right click on it and click run as administrator, just like above.

You can also run *whatsapp_xtract_console.bat* to specify the WhatsApp backup file manually.

/* For Android DB: */

python whatsapp_xtract.py -i msgstore.db -w wa.db

/* If wa.db is unavailable */

python whatsapp_xtract.py -i msgstore.db

/*For crypted DB*/

python whatsapp_xtract.py -i msgstore.db.crypt

/*For iPhone DB*/

```
python whatsapp_xtract.py -i ChatStorage.sqlite
```

Reviewing the methods for the article by Aditya Mahajan, M.S. Dahiya, and H.P. Sanghvi (2013), the study was much more in depth than Sahu's. "Five Android phones were analyzed covering three different versions of Android OS: Froyo 2.2, GingerBread 2.3x, and Ice Cream Sandwich 4.0x" (Mahajan et., al). The types of mobile devices included Sony Xperia STI5i mini, Sony Xperia Neo V (MT11i), LG P698, Samsung GSM GT-S5830, and HTC A8181 Desire. The acquisitions were done with both rooted and non-rooted Android devices. The purpose of this analysis was to determine what data and information could be located on the device's internal memory. The main focus was to explore the data within instant messenger, chat logs, images, or video.

A File System extraction was conducted on each device using UFED (Universal Forensic Extraction Device), Classic Ultimate version 1.8.0.0. Prior to the extractions being conducted, the USB Debugging option was enabled within the setting menu. In this type of extraction, UFED extracts the various files such as database and configuration files. It also extracts data of each application installed on the device and places it into separate folders. In order to properly review this data once extracted, UFED Physical Analyzer, no version numbers were listed within this research, was used. The examiner in these cases need to know where to look for the files and folders and what type of data they are examining because UFED Physical Analyzer can misinterpret the data or completely skip the data all together. The following tables explain where each type of data can be found (Mahajan et., al).

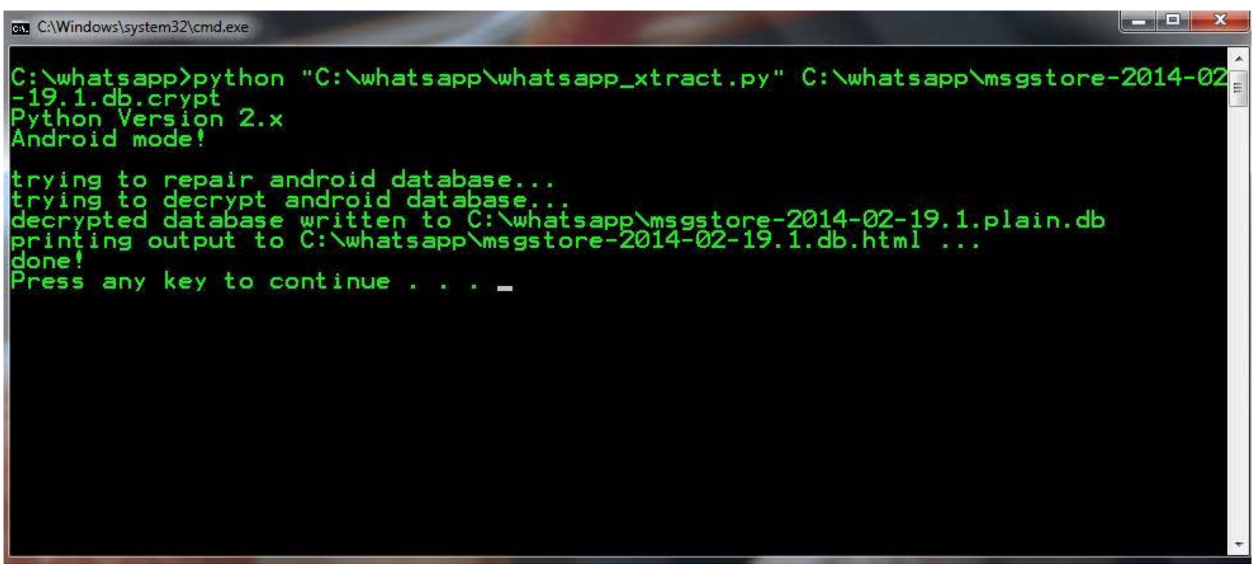
WhatsApp	***Msgstore.db***	***Messages, chat_ list***
	Wa.db	*Wa.contacts, sqlite_sequence*
Viber	*Viber_call_log.db*	*Viber_call_log*

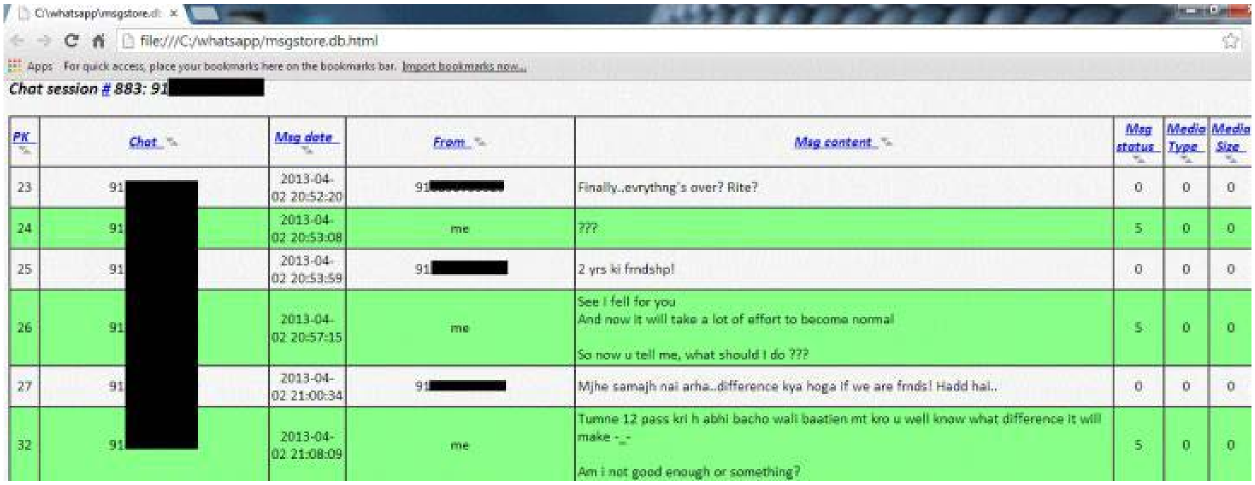
	Viber data (database)	*Android_metadata*
		Phonebook raw contact
		Phonebook contact
		Phonebook data
		Viber numbers
		Calls
	Viber messages (a database)	*Android_metadata*
		Messages
		Sqlite_sequence
		Threads
		participants

The file system of the devices examined contained these files on the internal memory of the phone, yet the images, video, and audio files of the WhatsApp are stored on the external memory card. All of these images, videos, and audio files are stored within a folder called media on the memory card. The methods used in this research were conducted with forensically sound equipment and followed the stringent rules of digital evidence collection and analysis. Hash values were also generated for each device examined. Users of the devices were asked to complete a set of pre-determined activities such as sending and receiving messages, video, audio, and images. The users continuously used both applications for a period of three months. The study was conducted on both devices with the WhatsApp and Viber applications installed, and those devices that were manually installed just for this research. The same tasks were completed on all devices. The *.ufd* file was loaded into UFED Physical Analyzer, as well as, examined manually. The following images are the devices used in this study.

Results

Ultimately, Sahu was able to recover the data requested in its decrypted state. The WhatsApp Xtract tool displays the information in the default browser on the user's computer as shown in the figures 2-4 below.

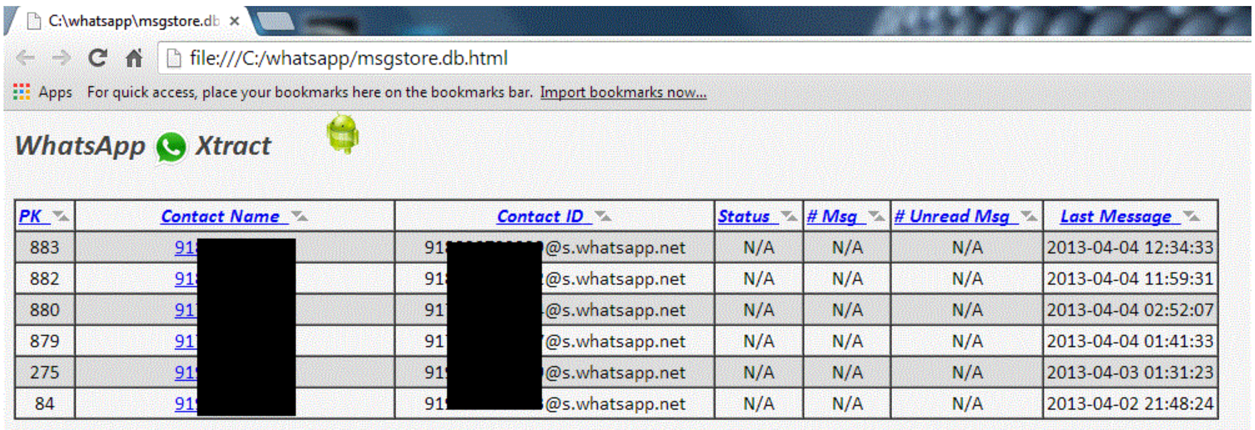




C:\whatsapp\msgstore.db ×
file:///C:/whatsapp/msgstore.db.html
Apps For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

Chat session # 883: 91[redacted]

PK	Chat	Msg date	From	Msg content	Msg status	Media Type	Media Size
23	91[redacted]	2013-04-02 20:52:20	91[redacted]	Finally..evrythng's over? Rite?	0	0	0
24	91[redacted]	2013-04-02 20:53:08	me	???	5	0	0
25	91[redacted]	2013-04-02 20:53:59	91[redacted]	2 yrs ki frndshp!	0	0	0
26	91[redacted]	2013-04-02 20:57:15	me	See I fell for you And now it will take a lot of effort to become normal So now u tell me, what should I do ???	5	0	0
27	91[redacted]	2013-04-02 21:00:34	91[redacted]	Mjhe samajh nai arha..difference kya hoga if we are frnds! Hadd hai..	0	0	0
32	91[redacted]	2013-04-02 21:08:09	me	Tumne 12 pass kri h abhi bacho wali baatien mt kro u well know what difference it will make -_- Am i not good enough or something?	5	0	0



C:\whatsapp\msgstore.db ×
file:///C:/whatsapp/msgstore.db.html
Apps For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...

WhatsApp Xtract

PK	Contact Name	Contact ID	Status	# Msg	# Unread Msg	Last Message
883	91[redacted]	91[redacted]@s.whatsapp.net	N/A	N/A	N/A	2013-04-04 12:34:33
882	91[redacted]	91[redacted]@s.whatsapp.net	N/A	N/A	N/A	2013-04-04 11:59:31
880	91[redacted]	91[redacted]@s.whatsapp.net	N/A	N/A	N/A	2013-04-04 02:52:07
879	91[redacted]	91[redacted]@s.whatsapp.net	N/A	N/A	N/A	2013-04-04 01:41:33
275	91[redacted]	91[redacted]@s.whatsapp.net	N/A	N/A	N/A	2013-04-03 01:31:23
84	91[redacted]	91[redacted]@s.whatsapp.net	N/A	N/A	N/A	2013-04-02 21:48:24

Overall, the data that was retrieved included contacts and messaging. Although the article states that you can also retrieve video, images, GPS, and even audio, Sahu did not supply images to reveal those results. These results were achieved on both iOS (*ChatStorage.sqlite*) and Android (*msgstore.db* & *wa.db*) databases.

The results for Mahajan were much more plentiful. The amount of data retrieved showed that investigators can still gain drones of evidence from Android devices. After conducting the file system extractions on all five devices, it was found that while most of the data was stored on the database file, some was also located on the memory card of the device. There were chat logs that were stored in the internal memory of the phone and the external memory card, yet on the external card, the logs were encrypted. Within the Viber extraction, five database files were located, yet only three of the databases contained forensically useful information.

The extractions and completed with UFED, Classic Ultimate version 1.8.0.0 and UFED Physical Analyzer yielded an abundance of results for the WhatsApp, but not as in depth for the Viber application. Table 1.1 shows the results for the analysis performed with UFED Physical Analyzer.

Artifacts Found	Artifacts Not Found
Sent chat messages	Contact list
Received chat messages	Profile pictures of users or contacts
Time stamps of each chat session	Locations of downloaded images or videos within WhatsApp

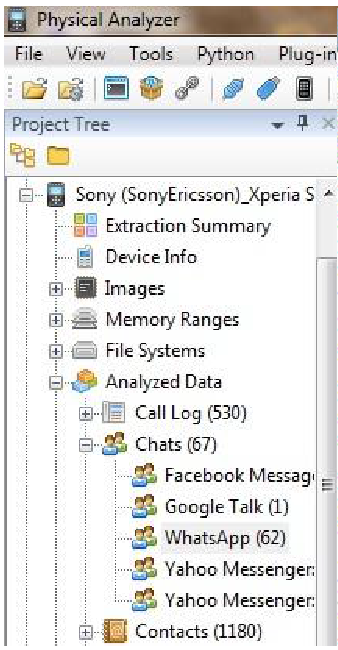
Table 1.1

Table 1.2 shows the information and data collected by doing a manual extraction.

Activities Performed	Similar Data Forensic Examinations Found/Not Found	Artifact related to the WhatsApp account with which data was shared Found/Not Found
Login phone number of user	Not found	N/A
Received chat messages	Found with timestamp	Found with timestamp
Outgoing messages	Found with timestamp	Found with timestamp
Sent Images	Found with timestamp. Sent image file name was found and the location of the image was on the memory card	Found with timestamp
Received Images	Found with timestamp. Received image file name was found and the location of the image was on the memory card	Found with timestamp
Sent/received videos	Found with timestamp. Sent/received video file name was found and the location of the image was on the memory card	Found with timestamp

Table 1.2

The results for the application Viber were less than useful when examining with UFED, Classic Ultimate version 1.8.0.0 and UFED Physical Analyzer. During the examination, there were no signs of anything pertaining to Viber until a manual extraction was completed. On the manual extraction, phone numbers, text messages, dates, times, and duration of calls was located. The data also revealed all of the messaging information with dates, times, and who the phone number to the person on the other end of the communication. Figures 5 and 6 reveal the results in UFED, Classic Ultimate version 1.8.0.0, without Viber being listed and the WhatsApp database examination with SQLite Database Browser.



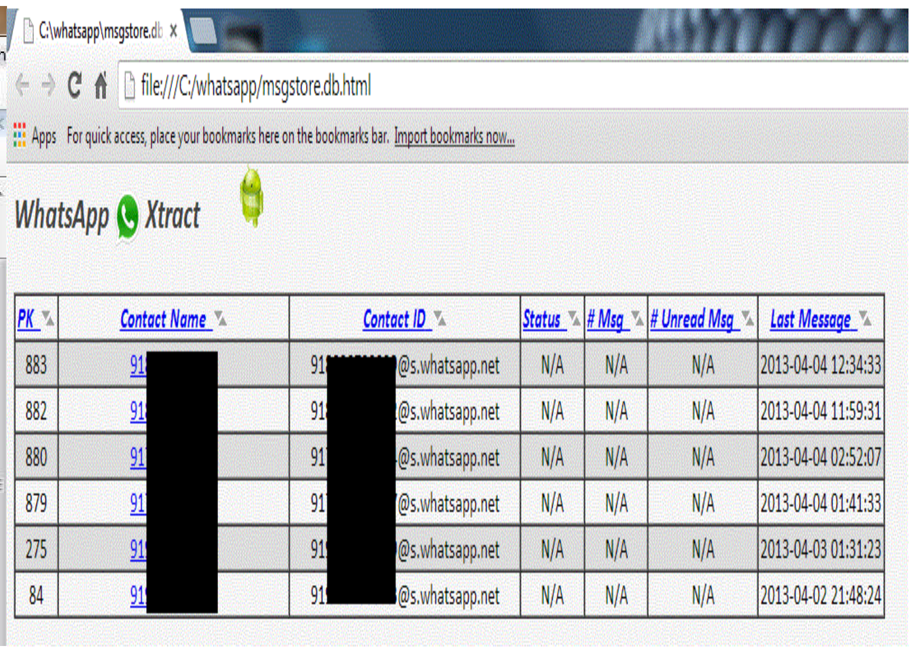


Figure 5 *Figure 6*

While technology continuously advances, the ability for users to hide both criminal activity and obtain more anonymity increases the risk of investigators unable to locate information. In the coming years, it may become more difficult to obtain pertinent information regarding criminal cases using mobile forensics. With the articles that were reviewed, the information was found relatively easily in some cases. Unfortunately, UFED, Classic Ultimate version 1.8.0.0 and UFED Physical Analyzer could not locate information on the application Viber. This application is an older talk and texting application and the information within these articles are also over three years old. More research would need to be conducted to see if newer versions of UFED, Classic Ultimate version 1.8.0.0, would be able to locate this information.

The WhatsApp created a wealth of information with both commercial and open source tools. It is unclear why the creator of this application would encrypt their databases, then allow the creation of a tool that could easily decrypt the same information it was protecting. At the time of this article, the WhatsApp was using a version of the same encryption protocol as Signal. In 2016, the WhatsApp has completely integrated with the Signal protocol creating a secure end-to-end encryption. More research is needed to see if the same information could still be acquired from the WhatsApp using the same tools and extractions techniques listed.

Chapter IV

Discussion/Conclusion

The evolution of technology is upon us. Every day, mobile device users are finding ways to cover their tracks, develop new applications, and ensure their own security instead of relying on companies. These self-made companies such as Open Whisper Systems are popping up all over the globe. These companies make the promise of safety, security, and anonymity when it comes to the users' online presence and safeguarding all of their means of communication. One of the concerns within the forensic community is that the software available can be helpful or hinder an investigation depending on the device. With every advancement in technology, companies offer more commercial services to law enforcement so that they can extract data from a user's device. Often, there are limitations to these commercial products. Even with product licensing, some features may still not be available. For example, in this study, access to the iCloud was not available using Cellebrite's Cloud Analyzer.

Another limitation to this study, every phone is different and especially with Android devices, the data is stored differently with different operating systems. What you could discover on one device may not be located in the same location in the memory on another Android. While storage of data is a limitation, the backup data is as well. Androids can back up to Google Drive, while iPhone has the iCloud, both not accessible to law enforcement without additional warrants. While the UFED Cloud Analyzer support retrieved data from Google Play, WhatsApp, and other messaging applications, it does not support received data from the Signal Application.

Androids and iPhones also have device security that could limit what type and how much data is retrieved. If any of the devices used would have had passwords, finger print, or even facial recognition to unlock the phone, then it could have hindered the data, if any, retrieved. In this study, there were no devices with security. Cellebrite does offer tools that will unlock some types of passwords etc., but it was not required for this study. With any phone an examiner could run into the problem of having extraction options limited. This was the case in this study. Three out of the four phones did not offer the physical extraction which limited the amount of data that was recovered. On three of the devices, almost no data was retrieved. Any examiner can run into this issue on a daily basis depending on the type of device. The Cellebrite tools do not support every type of phone for a physical extraction.

The overall study was to discover which devices had more data uncovered, as well as, which products and open source tools were able to extract that data. Based on the results, the data extracted was entirely dependent on which type of mobile device was used and which forensic tool or software was able to extract data from that type of device. A physical extraction was only achieved on one of the devices. This device, the ZTE Z993, recovered more data than the other mobile devices in this study. While the other Android device was not much newer than this one, it yielded less impressive extraction results. Unfortunately, there were not nearly the same amount of results from the extractions on the iPhone 4S or the iPhone 7. These two devices did not get physical extractions, only logical and the filesystem extractions. Since the physical extraction always extracts the most data, if the physical extractions were supported for the Android LG, iPhone 4S or the iPhone 7, there could have potentially been more data recovered.

Research previously conducted on a similar topic was with the WhatsApp. While this application uses a similar version of the signal encryption, the results just with a tool called the WhatsApp extractor was able to pull viable results from user mobile devices. This research was dated, circa 2012-2013, which created even more reason to give this topic updated results with the Signal Application that started it all. As previously stated, the results here, while not in abundance, shows the limitations of the software, and the differences in devices.

The future potential for studies such as this are exponential. There are updates to mobile devices, software, and the development of applications on a daily basis. One of the markets on the rise is the development of not only messaging applications, but those with security features that are necessary for the user to maintain anonymity. Users sometimes forget that even if they don’t want to, their lives are stored in some way on their phones. As stated, the government were even taking advantage of the newer applications such as Signal to maintain privacy and security of communications. The type of research that can be further explored can include different types of software, paid options of open source tools, and analyzing data from the cloud or back up storage with both Android and Apple devices. It could have uncovered different results using different deleted recovery methods or different types of mobile devices. Since this study was only meant to be completed using the basic forensic commercial and open source tools, another examiner could potentially use this data and expand on software and hardware capabilities.

In conclusion, the forensic community will continue to evolve as long as there is a need for the services. In this case, mobile devices are just becoming more and more prominent in criminal, civil, and private litigations. These results are only based on the devices that were used in this study, but with different devices, operating systems, and future forensic tools, the results could be completely different. The results that may or may not be retrieved can create many more questions or uncover more or less data. These devices store the user’s whole life on them, and no matter how many security measures taken, there will always pieces left behind.

References

Sahu, M. S. (2014). An Analysis of WhatsApp Forensics in Android Smartphones. *International Journal of Engineering Research, 3*(5), 349-350. doi:10.17950/ijer/v3s5/514

Mahajan, A., Dahiya, M. S., & Sanghvi, H. P. (2013). Forensic Analysis of Instant Messenger Applications on Android Devices. *International Journal of Computer Applications, 68*(8), 38-44. doi:10.5120/11602-6965

Bommisetty, S., Tamma, R., & Mahalik, H. (2014). *Practical mobile forensics*. Birmingham: Packt Publishing.

Kilel, B. (2013). *Digital forensics: crime on digital devices*. Frederick, MD: Zapphire Pub.

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

Marlinspike, M. (2013-2017) *Open Whisper Systems Overview*. Retrieved March 6, 2017, from https://signal.org

Mobile Forensic Tools. (2017) Retrieved March 8, 2017, from https://www.concise-courses.com/security/mobile-forensics-tools/

Signal Overview. (2017) Retrieved March 5, 2017, from https://en.wikipedia.org/wiki/Signal_(software)

Signal protocol. (2017). Retrieved March 8, 2017, from https://github.com/whispersystems/libsignal-protocol-java

Smiley, L (2017) *Signal Secure Messaging*. Retrieved March 7, 2017, from http://www.theverge.com/2017/1/12/14244634/signal-app-secure-messaging-trump-surveillance-encryption

Cellbrite. (2017) Retrieved March 7, 2017, from http://www.cellebrite.com/

Collazo, J. (2017) *How to use signal for iOS*. Retrieved March 2, 2017, from https://ssd.eff.org/en/module/how-use-signal-ios

Collazo, J. (2017) *How to use signal for Android*. Retrieved March 2, 2017, from https://ssd.eff.org/en/module/how-use-signal-android

Mahliak, H. (2017) *Open Source Forensic Tools*. Retrieved September 6, 2017, from http://smarterforensics.com/category/open-source/

Cellbrite Cloud (2017) *UFED Cloud Analyzer Supported Data Sources*. Retrieved November 6, 2017 from https://media.cellebrite.com/wp-content/uploads/2017/08/UFED_CloudAnalyzerSupportedDevices.pdf

Klosowski, T. (2017) *Secure Messaging Showdown*. Retrieved May 6, 2017, from https://lifehacker.com/secure-messaging-app-showdown-whatsapp-vs-signal-1794684943

Shelton, M. (2017) *Signal Beginners*. Retrieved September 16, 2017 from https://freedom.press/news/signal-beginners/

Barrett, B. (2017) *WikiLeaks CIA Hack Signal*. Retrieved September 17, 2017 from https://www.wired.com/2017/03/wikileaks-cia-hack-signal-encrypted-chat-apps/

Hardwick, Tim (2017) *Encrypted Messaging App 'Signal' Approved for Use by U.S. Senate.* Retrieved June 25, 2017 from https://www.macrumors.com/2017/05/17/encrypted-app-signal-approved-for-use-us-senate/