COMPARATIVE FILE STRUCTURE ANALYSIS OF VIDEO FILES SENT AND RECEIVED VIA WHATSAPP

by

SIMON JEROME VICTOR

B.F.A. University of the West Indies, 2016

A thesis submitted to the

Faculty of the Graduate School of the

University of Colorado in partial fulfillment

of the requirements for the degree of

Master of Science

Recording Arts Program

2020

This thesis for the Master of Science degree by

Simon Jerome Victor

has been approved for the

Recording Arts Program

by

Catalin Grigoras, Chair

Cole Whitecotton

Jeffrey Smith

Date: December 12, 2020

Victor, Simon Jerome (M.S., Recording Arts Program)

Comparative File Structure Analysis of Video Files Sent and Received via WhatsApp

Thesis directed by Associate Professor Catalin Grigoras

ABSTRACT

WhatsApp is a popular cross-platform messaging service that is used by a sizable fraction of the world's population for passing along all manner of information. This paper compares the differences introduced in video files sent, and then received via WhatsApp. It utilizes videos created in native video recording applications, and where possible, files recorded in the WhatsApp application using the recording feature. The information gathered in this research project will be useful in understanding changes to the file structure of video files sent through WhatsApp, which aids in the process of video authentication for forensic purposes. It will also explore the differences in available acquisition methods of the files sent through WhatsApp to preserve as much data as possible for forensic purposes. To achieve this the study will look at metadata, binary data, file structure, and any other relevant observations.

The form and content of this abstract are approved. I recommend its publication.

Approved: Catalin Grigoras

DEDICATION

I dedicate this thesis to my family who has supported me unconditionally throughout my studies. To my children Amaziah and Arielle and my partner Kelly who dealt directly with my extensive absence. For my children who I hope to inspire to be their best even when it is not the easiest path. To Kelly who has willing listened to me rehearsing concepts after classes and assisted me in debugging MATLAB scripts - for this her name should be added to my diploma. To my siblings Sharon and Justin who have supported me through my endeavors over the years. To Lisa and Marva who encouraged me to step out in faith when I had no money to start this program. To Ronald who introduced me to Audio Forensics in the early days. To my close friends who encouraged me to keep going even when challenges came. Thank you all.

ACKNOWLEDGEMENTS

I wish to acknowledge the staff of the NCMF for their guidance throughout my degree program, going beyond the call of duty to ensure that I was accommodated. Their guidance has been instrumental in my success to this point and I hope to pay this forward. I wish to acknowledge the Organization of American States for their sponsorship which was instrumental in my being able to begin this program. The East Caribbean Conference of Seventh-day Adventists for allowing me the time I needed to travel for my studies.

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

TABLE

LIST OF FIGURES

FIGURE

LIST OF ABBREVIATIONS

AVC – Advanced Video Coding

CODEC – Coder-decoder

FAQ – Frequently asked questions

HD – High Definition

MB – Megabyte

IM – Instant Messaging

MMS – Multimedia Messaging Service

OS – Operating System

SMS – Simple Messaging Service

CHAPTER I

INTRODUCTION

Humans are social creatures, drawn to communicate with each other. Technology developed by mankind to facilitate communication seeks to fill the basic need for connection between people, and while the tools evolve over time the fundamental purpose remains. Modern instant messaging platforms share underlying similarities with the telegraph, which was designed to send short messages over long distances as quickly as possible. From the first telegram in 1844 until present times humans continue to send short messages - sentences at a time - to people who are far away, albeit with the added ability to send images, videos, and audio recordings. This functionality is equally useful for non- personal and formal purposes. Businesses, celebrities, and a plethora of entities use this technology daily, just as do persons with ill-intent. Disputes arise from messages sent, offenses are committed, and sometimes evidence of wrongdoing is transmitted through messaging services. Thus, it is important for investigators to have knowledge of how these services work to aid in examination for forensic purposes.

Evolution of Messaging

Text messaging is defined as alphanumeric communication sent from one electronic device to another, utilizing a cellular network[1]. SMS (Short Message Service) messaging is a form of text messaging originally designed for mobile phone use with a typical length of 160 characters and provides convenient low-cost way to send information quickly between users[2]. MMS (Multimedia Messaging Service) builds on SMS by adding images, audio, and video files[3]. SMS and MMS are popular services for the ability to communicate at low cost, with 5 billion persons utilizing these services the world over[4].

Instant Messaging (IM) builds on the foundations of MMS but works over internet only services, whereas SMS and MMS travel over cellular networks. IM also takes the concept further by introducing a dedicated application which allows users to create a list of 'contacts' to communicate with and includes a plethora of features that MMS cannot accommodate because of the technical limitations imposed by mobile networks. There can be a cost advantage to using Instant Messaging since it is available over the internet and allows for unlimited messaging at no extra charge, compared to text messaging which may incur costs depending on the destination of the message[5]. Rich Communication Service (RCS) is similar to WhatsApp in general functionality, facilitating instant messaging, sending of media files, and read receipts, among other features on mobile devices as an alternative to SMS and MMS[6]. Facilitation of RCS messaging can be handled by the mobile data network where available and is otherwise facilitated by Google servers. RCS is currently controlled by Google, and works on Android smartphones with no current scope for them working on Apple iPhones, as RCS is seen as a rival service to Apple's iMessage[7]. Other than unavailability on iPhone, the adoption of RCS is also limited as it is available only in the United Kingdom, Germany, Mexico, and the United States at the time or writing[8].

Description of WhatsApp

WhatsApp is a cross-platform communication service that employs end-to-end encryption for all communication. It is at the core an instant messaging application, expanded to offer additional features. WhatsApp is owned by Facebook as one of their instant messaging platforms and has worldwide reach unlike most other instant messaging platforms. Over 2 billion persons in more than 180 countries use WhatsApp, with 1 billion daily active users sending 65 billion messages per day[9].

To access the platform, WhatsApp provides native applications for Android, iOS, Macintosh, Windows, and Chrome OS. A user simply needs to install the application on a mobile/cellular device and register the phone number of that device with WhatsApp - once registered it uses the 'phonebook' of the device to build a contact list, and thus it requires no further actions to connect individuals. WhatsApp offers instant messaging, Voice-Over-IP calls, video calls and group chats. Like MMS services, it allows for the sending of still images, audio recordings, and video recordings.[10]

With such ease of access to free encrypted communication, WhatsApp can be used for illegal activity, and so it is important to understand what to expect when one examines data extracted which may have been suspected to be evidence of an offense.

The purpose of this research paper is to glean useful information about how a video changes as it passes through WhatsApp. It will examine the changes made to files at a structural level and compare different methods of downloading video files sent through WhatsApp to preserve as much information as possible.

Previous Research

There has been much research into related areas on which this paper builds. Risemberg details in the paper "File Structure Analysis of Media Files Sent and Received Over WhatsApp[11]" an analysis of the effects of WhatsApp on a variety of media sent through the service. While it examines in detail images and audio recordings it doesn't investigate videos transmitted.

As far back as 2011 there was a need for analysis of the files sent through social media platforms. "A Forensic Analysis of Images on Online Social Networks" by Castiglione, et al, looks specifically at this, urging at the time that the state of technology may change soon and so the work needed to be updated[12]. This is not a novel idea as numerous other researchers have targeted this area of study. Pippen saw the need to examine Facebook specifically to gain insight into changes made to files when passed through this platform[13]. Orr and Castro examined WhatsApp for Android specifically to establish chronology of contacts and messages for investigative purposes[14]. Anglano researched the reconstruction of a messaging history with the assistance of artifacts left behind by WhatsApp[15]. Based on the widespread use of such social media services, it is integral and beneficial to the forensic community to have this research available. It must also be considered that technology is changing constantly and so the need for research is on-going.

Fundamentally, this paper uses the general idea that there is a need for an understanding of how a variety of social networks change files, and it is the author's intention to contribute existing the body of knowledge.

CHAPTER II

MATERIALS

The aim of this research was to determine the kinds of changes that occur when a video file is sent over WhatsApp. Since WhatsApp is a cross-platform service, the experiment was carried out using multiple Operating Systems (OS) and devices to simulate a wide variety of transfers. The platforms used for testing were Android, Chrome OS, Macintosh OS, and Windows 10.

Table 1. Devices used for sending and receiving of files

Device Make	Device Model	Operating System	OS version	WhatsApp version	Function
Nokia	6/TA-1000	Android	7.1.1	2.20.197.20/2.20.200.22	Sending only
Xiaomi	Redmi Note 5 (unlocked bootloader)	Android	9 (MIUI Global 11.0.3.0)	2.20.197.20/2.20.200.22	Receiving only
Hewlett Packard	Chromebook x2	Chrome OS	85.0.4183.133	2.2035.14	Sending & receiving
Apple	MacBook Pro 13inch, mid 2012	Macintosh	Mojave 10.14.6	2.2027.10	Sending only
Apple	iMac, 27 inch, Late 2013	Macintosh	Mojave 10.14.6	2.2035.15	Receiving only
Acer	Aspire E5-575/N16Q2	Windows	10 Version 1903, Build 18362.1082	2.2035.14	Sending & receiving

Video Creation

Each device utilized for sending was used to first create its own videos, and then those videos were sent over WhatsApp. The default settings were used for video recordings in all cases except those which required a change in resolution to test the platform's handling of different resolutions. Video recording settings were only changed for a subset of files that were sent from the Android Device.

File Sending and Receiving

Two separate WhatsApp accounts were used for the experiment; the sending account A and the receiving account B were always used for their respective purposes. All files created were sent from account A to B, and then all files sent and received were transferred to the iMac computer used for analysis. No files were forwarded for the use of this experiment.

Data Transfer

For the two Android devices used, transfers were done via USB cable, utilizing Android File Transfer[16]. Files created and received via Chromebook were transferred by zipping them and uploading them to Google Drive. The file created on the MacBook Pro was collected via flash drive, whereas files downloaded on the iMac using account B were kept on that system as this was where analysis was carried out. The file sent and those received from the Windows computer was transferred via USB flash drive – first by zipping and then being copied onto external storage for transfer to the iMac computer.

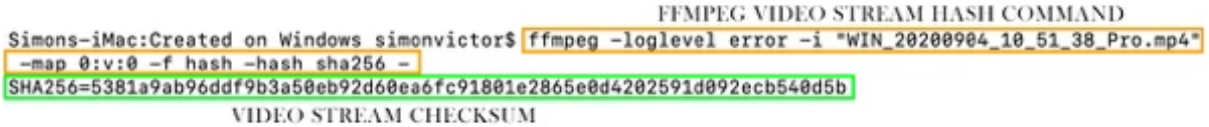
Data Integrity

Where possible all transferred files were checked for their integrity via hashing. File hashing is the process of summarizing a file into a fixed-length string of characters by passing the contents of the file through a hashing algorithm. This resultant string of characters known as a checksum is unique to each file based on its contents, and if any data in the file changes a different checksum will be returned[17]. Likewise, once the same data is passed through the same hashing algorithm it will return the same checksum.

The SHA-256 hashing algorithm was used to verify the integrity of the video files as they were transferred between the various devices. For the Android devices there was no known way to hash the files created. On the Chromebook, Macintosh and Windows computers, SHA256 hashing was utilized to ensure that the files remained unchanged throughout the transfer process.

Stream Hashing

While file hashing is useful to ensure data integrity in file transfers it can also be useful to check for changes made to specific components of a file. A video file is made up of metadata, a video stream and an audio stream wrapped in a file container[18]. When a file is modified any changes to each of the four components will return a new checksum when hashed. And so to help us to understand the changes to a file in a more detailed way, we can run the video and audio components of the file as 2 separate components to be hashed[19]. This is referred to as stream hashing and was used throughout the experiment.



```
Simons-iMac:Created on Windows simonvictor$ ffmpeg -loglevel error -i "WIN_20200904_10_51_38_Pro.mp4"
 -map 0:v:0 -f hash -hash sha256 -
SHA256=5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b
```

Figure 1. FFMPEG video stream hashing command and checksum

Stream hashing of the video and audio streams was carried out to gain insight into what changes occurred at different stages of processing within the uploading and transferring of files through WhatsApp. Figure 1 on the previous page shows an example of a video stream hash in FFMPEG.

Stream hashing is also useful to determine that while the metadata contents in the container of a file may have changed the media content remains the same.

Data

The videos were transferred utilizing sending account A and receiving account B. In some instances, a video was sent and received on the same device utilizing different accounts, resulting in 2 separate files - the video created, and the video received from WhatsApp. Figure 2 on the next page shows the devices used to create files and send them from account A, along with the corresponding devices used to collect those files using account B.

Files were first created on each device and then sent via WhatsApp. On the Android phone files were recorded via the native Camera app, Open Camera[20], and in WhatsApp. On Macintosh and Windows files were only recorded via native applications provided with the operating system for recording images. On the Chromebook the files recorded using the native imaging application weren't playing when opened on other devices, and so the application Open Camera was used to create the video sent from the Chromebook. Figure 2 on the following page illustrates the distribution methods utilized.

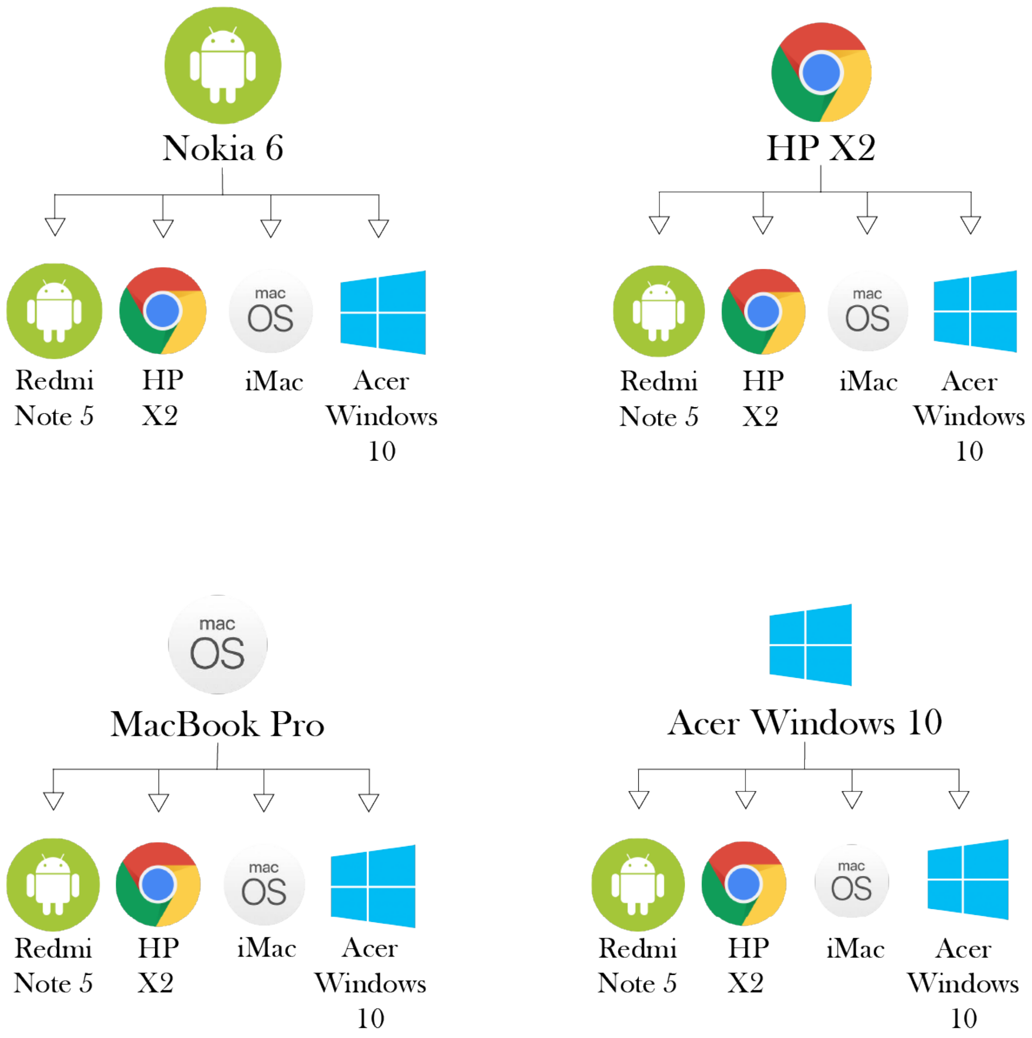


Figure 2. Devices used for sending and receiving video files

Resolution

The resolution of the files transmitted were seen to be a factor which determined the treatment in processing when sending files from the Android device.

Resolution refers to the number of pixels that are used to reproduce a digital image. All digital images which are displayed on electronic displays are created by an array of pixels held in a specified arrangement. Each pixel has a specific color and brightness value, and when all pixels are displayed as specified it gives the illusion of an image being displayed[21]. Video resolution is generally specified in a rectangular shape with the resolution being defined as number of pixels wide by number of pixels high, or Width x Height[22].

High Definition resolution was first adopted by National Television System Committee as a standard and is defined as 2 different sizes – 1280x720 (HD) and 1920x1080 (Full HD)[23]. Based on the observations from preliminary tests, differences in resolution seemed to bring about varying outcomes relative to whether a file's video and audio streams would be re-encoded went sent via WhatsApp. It was thus determined that video recordings needed to be collected at HD, Full HD, and a lower resolution in order to track changes made to files due to their resolution. Table 2 shows all files collected with emphasis on the original resolution, compared to the resolution after being sent through WhatsApp.

Table 2. Changes to resolution of files sent through WhatsApp

Sending Platform	Recording Application	Resolution & File size	Original Resolution	Resolution after Download
Android	Native Camera app	1080HD > 64MB	1920 x 1080	640 x 352
Android	WhatsApp	720 HD	1280 x 720	1280 x 720
Android	Native Camera app	1080HD < 64MB	1920 x 1080	640 x 352
Android	WhatsApp	Max length recording	1280 x 720	640 x 352
Android	Open Camera	< HD < 64MB	1024 x 768	1024 x 768
Android	Open Camera	< HD > 64MB	1024 x 768	640 x 480
Android	Open Camera	720HD < 64MB	1280 x 720	640 x 352
Chrome	Open Camera	< 64MB	1920 x 1080	1920 x 1080
macOS	Photo Booth	< 64MB	1080 x 720	1080 x 720
Windows 10	Native Camera app	< 64MB	1280 x 720	1280 x 720

File Format

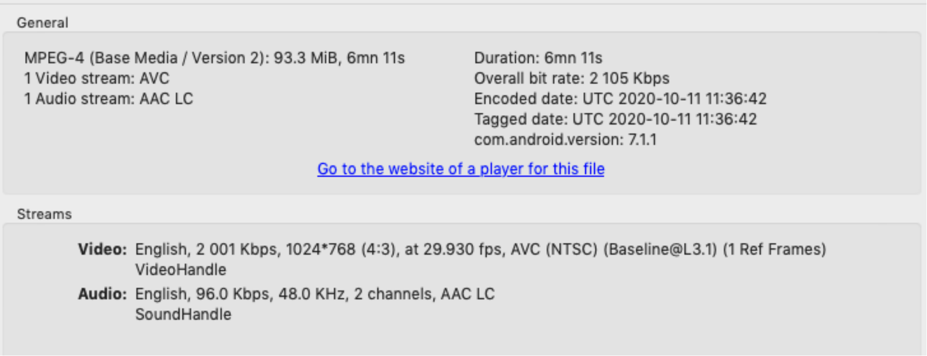
The format of a file determines how the data is stored and encoded[24]. The format of a file is specified by the container[25], as this ‘contains’ the data that we access. The way that the container organizes the data stored is referred to as the structure. In some cases, the same data can be transferred to different containers, as different containers are useful in different instances. The container type is typically easily identified by the file extension, designated by a period and a string of characters, for example .mp4, .wmv, .avi.

File Structure

The structure of a file refers to the way that the subsets of data within the file are arranged[24]. Much like a well-ordered physical environment, there is an order to how the information is stored within the file, based on the rules stipulated by the file container. In a video file one would expect to find general metadata, video, and audio as distinct components with the ability to access and manipulate or analyze each part independently. In a video file we examine the structure based on the arrangement of the separate components, along with the internal arrangement and information found in those components.

Metadata

Metadata is referred to as data which describes data. It holds basic information about the file and its attributes, which facilitate the handling and categorization of the files[26]. This also simplifies the processes of finding and working with files on a system in which they are stored. Metadata in a video file would be expected to provide at the very least the resolution, codec, and duration of a file.



General

MPEG-4 (Base Media / Version 2): 93.3 MiB, 6mn 11s
1 Video stream: AVC
1 Audio stream: AAC LC

Duration: 6mn 11s
Overall bit rate: 2 105 Kbps
Encoded date: UTC 2020-10-11 11:36:42
Tagged date: UTC 2020-10-11 11:36:42
com.android.version: 7.1.1

Go to the website of a player for this file

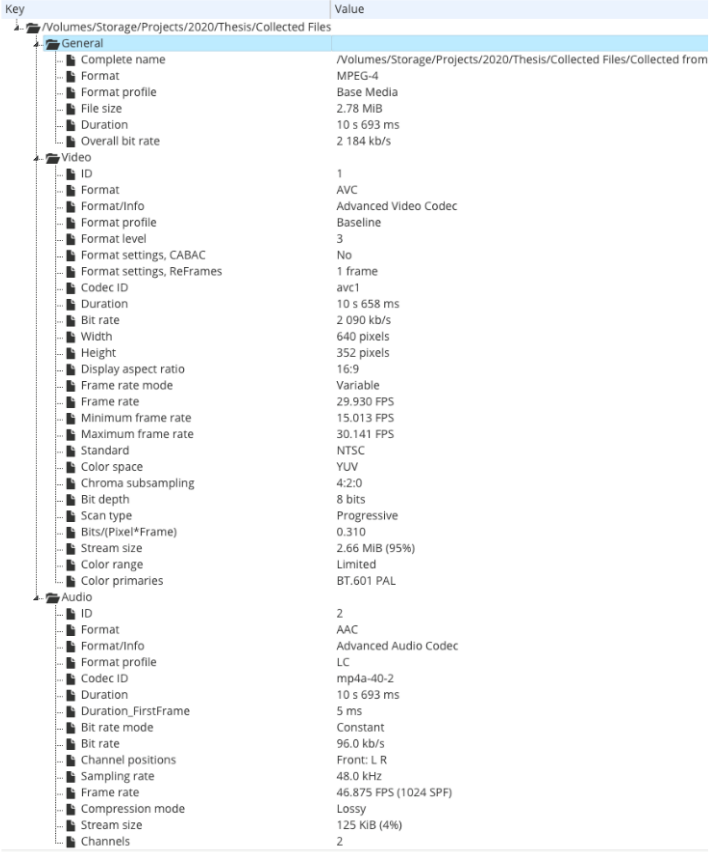
Streams

Video: English, 2 001 Kbps, 1024*768 (4:3), at 29.930 fps, AVC (NTSC) (Baseline@L3.1) (1 Ref Frames)
VideoHandle

Audio: English, 96.0 Kbps, 48.0 KHz, 2 channels, AAC LC
SoundHandle

Figure 3. 'Simple view' of metadata in MediaInfo

Typically, much more information is stored in metadata and the ability to access and examine this information was important in the analysis of files in this experiment.



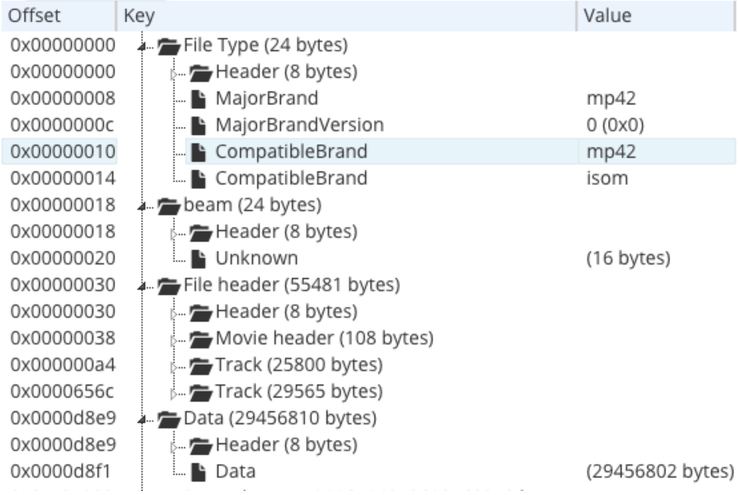
```
Key                                         Value
/Volumes/Storage/Projects/2020/Thesis/Collected Files
  General
    Complete name                           /Volumes/Storage/Projects/2020/Thesis/Collected Files/Collected from
    Format                                  MPEG-4
    Format profile                          Base Media
    File size                               2.78 MiB
    Duration                                10 s 693 ms
    Overall bit rate                        2 184 kb/s
  Video
    ID                                      1
    Format                                  AVC
    Format/Info                             Advanced Video Codec
    Format profile                          Baseline
    Format level                            3
    Format settings, CABAC                  No
    Format settings, ReFrames               1 frame
    Codec ID                                avc1
    Duration                                10 s 658 ms
    Bit rate                                2 090 kb/s
    Width                                   640 pixels
    Height                                  352 pixels
    Display aspect ratio                    16:9
    Frame rate mode                         Variable
    Frame rate                              29.930 FPS
    Minimum frame rate                      15.013 FPS
    Maximum frame rate                      30.141 FPS
    Standard                                NTSC
    Color space                             YUV
    Chroma subsampling                      4:2:0
    Bit depth                               8 bits
    Scan type                               Progressive
    Bits/(Pixel*Frame)                      0.310
    Stream size                             2.66 MiB (95%)
    Color range                             Limited
    Color primaries                         BT.601 PAL
  Audio
    ID                                      2
    Format                                  AAC
    Format/Info                             Advanced Audio Codec
    Format profile                          LC
    Codec ID                                mp4a-40-2
    Duration                                10 s 693 ms
    Duration_FirstFrame                     5 ms
    Bit rate mode                           Constant
    Bit rate                                96.0 kb/s
    Channel positions                       Front: L R
    Sampling rate                           48.0 kHz
    Frame rate                              46.875 FPS (1024 SPF)
    Compression mode                        Lossy
    Stream size                             125 KiB (4%)
    Channels                                2
```

Figure 4. Metadata showing specifications and information on a video file

The metadata of a file is typically stored separately and distinctly from the audio and video data of the file, generally at the beginning or the end of the data.

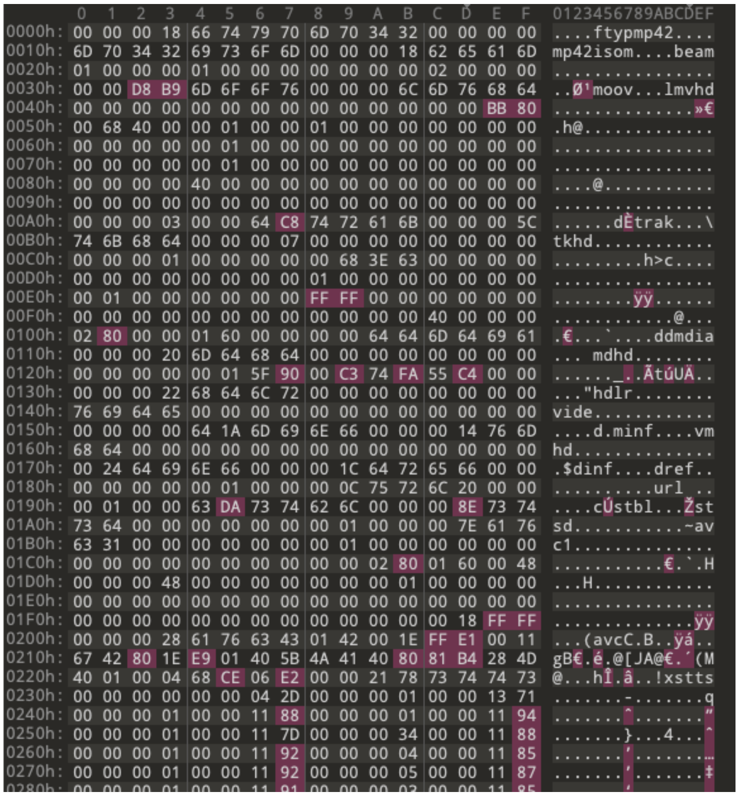
The metadata stored at the beginning of a file is known as the file header[27]. Most pieces of software leave a marker with information or make drastic changes to a file's header. It is not uncommon to see the name of software stored in the metadata of a video that would have been used to edit or transcode the file. For this purpose, forensic examiners often search for keywords within the metadata of the file to look for traces of software that may have been used to manipulate it in some manner.

While some metadata is human-readable as ASCII text, much of the metadata of a file needs to be decoded with a hexadecimal editor. Figure 5 below shows an example of metadata interpreted while Figure 6 on the following page shows the same metadata as seen through a hexadecimal reader.



Offset	Key	Value
0x00000000	File Type (24 bytes)	
0x00000000	Header (8 bytes)	
0x00000008	MajorBrand	mp42
0x0000000c	MajorBrandVersion	0 (0x0)
0x00000010	CompatibleBrand	mp42
0x00000014	CompatibleBrand	isom
0x00000018	beam (24 bytes)	
0x00000018	Header (8 bytes)	
0x00000020	Unknown	(16 bytes)
0x00000030	File header (55481 bytes)	
0x00000030	Header (8 bytes)	
0x00000038	Movie header (108 bytes)	
0x000000a4	Track (25800 bytes)	
0x0000656c	Track (29565 bytes)	
0x0000d8e9	Data (29456810 bytes)	
0x0000d8e9	Header (8 bytes)	
0x0000d8f1	Data	(29456802 bytes)

Figure 5. Metadata showing the basic organization of file structure



```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F  0123456789ABCDEF
0000h: 00 00 00 18 66 74 79 70 6D 70 34 32 00 00 00 00  ....ftypmp42....
0010h: 6D 70 34 32 69 73 6F 6D 00 00 00 18 62 65 61 6D  mp42isom....beam
0020h: 01 00 00 00 01 00 00 00 00 00 00 00 02 00 00 00  ................
0030h: 00 00 D8 B9 6D 6F 6F 76 00 00 00 6C 6D 76 68 64  ..Ø¹moov...lmvhd
0040h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 BB 80  ..............»€
0050h: 00 68 40 00 00 01 00 00 01 00 00 00 00 00 00 00  .h@.............
0060h: 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00  ................
0070h: 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00  ................
0080h: 00 00 00 00 40 00 00 00 00 00 00 00 00 00 00 00  ....@...........
0090h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
00A0h: 00 00 00 03 00 00 64 C8 74 72 61 6B 00 00 00 5C  ......dÈtrak...\
00B0h: 74 6B 68 64 00 00 00 07 00 00 00 00 00 00 00 00  tkhd............
00C0h: 00 00 00 01 00 00 00 00 00 68 3E 63 00 00 00 00  .........h>c....
00D0h: 00 00 00 00 00 00 00 00 01 00 00 00 00 00 00 00  ................
00E0h: 00 01 00 00 00 00 00 00 FF FF 00 00 00 00 00 00  ........ÿÿ......
00F0h: 00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 00  ............@...
0100h: 02 80 00 00 01 60 00 00 00 00 64 64 6D 64 69 61  .€...`....ddmdia
0110h: 00 00 00 20 6D 64 68 64 00 00 00 00 00 00 00 00  ... mdhd........
0120h: 00 00 00 00 00 01 5F 90 00 C3 74 FA 55 C4 00 00  ......_..ÃtúUÄ..
0130h: 00 00 00 22 68 64 6C 72 00 00 00 00 00 00 00 00  ..."hdlr........
0140h: 76 69 64 65 00 00 00 00 00 00 00 00 00 00 00 00  vide............
0150h: 00 00 00 00 64 1A 6D 69 6E 66 00 00 00 14 76 6D  ....d.minf....vm
0160h: 68 64 00 00 00 00 00 00 00 00 00 00 00 00 00 00  hd..............
0170h: 00 24 64 69 6E 66 00 00 00 1C 64 72 65 66 00 00  .$dinf....dref..
0180h: 00 00 00 00 00 01 00 00 00 0C 75 72 6C 20 00 00  ..........url ..
0190h: 00 01 00 00 63 DA 73 74 62 6C 00 00 00 8E 73 74  ....cÚstbl...Žst
01A0h: 73 64 00 00 00 00 00 00 00 01 00 00 00 7E 61 76  sd...........~av
01B0h: 63 31 00 00 00 00 00 00 00 01 00 00 00 00 00 00  c1..............
01C0h: 00 00 00 00 00 00 00 00 00 00 02 80 01 60 00 48  ...........€.`.H
01D0h: 00 00 00 48 00 00 00 00 00 00 00 01 00 00 00 00  ...H............
01E0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
01F0h: 00 00 00 00 00 00 00 00 00 00 00 00 00 18 FF FF  ..............ÿÿ
0200h: 00 00 00 28 61 76 63 43 01 42 00 1E FF E1 00 11  ...(avcC.B..ÿá..
0210h: 67 42 80 1E E9 01 40 5B 4A 41 40 80 81 B4 28 4D  gB€.é.@[JA@€.´(M
0220h: 40 01 00 04 68 CE 06 E2 00 00 21 78 73 74 74 73  @...hÎ.â..!xstts
0230h: 00 00 00 00 00 00 04 2D 00 00 00 01 00 00 13 71  .......-.......q
0240h: 00 00 00 01 00 00 11 88 00 00 00 01 00 00 11 94  .......ˆ.......”
0250h: 00 00 00 01 00 00 11 7D 00 00 00 34 00 00 11 88  .......}...4...ˆ
0260h: 00 00 00 01 00 00 11 92 00 00 00 04 00 00 11 85  .......’.......…
0270h: 00 00 00 01 00 00 11 92 00 00 00 05 00 00 11 87  .......’.......‡
0280h: 00 00 00 01 00 00 11 91 00 00 00 03 00 00 11 85  .......‘.......…
```

Figure 6. Metadata displayed in a hexadecimal editor

Video and Audio Streams

A bit-stream is defined as a series of bits in the transmission of data[28]. In a file the video and audio are both stored separately in the container and the information stored in each bit-stream can be observed and handled separately, thus allowing for analysis integral to examination of the structure of a file.

Codecs

A CODEC is a coder and decoder algorithm used to encode and decode a data stream[25]. For use in a computer system data must represented as bits which must be encoded by an algorithm to facilitate the storage and handling of the data. In order to access the stored data, it must then be decoded using the inverse algorithm. In media files, many codecs exploit limits of the human auditory and visual systems to remove unnecessary data to make them optimized for efficient storage and transmission within a system.

Dataset

The full dataset used in this experiment was as follows: 10 files recorded with various devices and then sent over WhatsApp to be downloaded by 4 devices, yielding 40 files downloaded.

The dataset contains a total 50 files. *Table 3* below shows the list of files sent and number of downloads on each device.

Table 3. List of files recorded (sent) and received (downloaded)

Recording Platform	**Resolution & File size**	**File format of original video**	**File format of received video**				
Android	1080HD > 64MB	mp4	mp4	Each file downloaded on **Android phone** Total - 10 files	Each file downloaded on **Chromebook** Total - 10 files	Each file downloaded on **MacBook Pro** Total - 10 files	Each file downloaded on **Windows 10** Total - 10 files
Android	720 HD (WhatsApp recording)	mp4	mp4				
Android	1080HD < 64MB	mp4	mp4				
Android	Max length recording in WhatsApp	mp4	mp4				
Android	< HD < 64MB	mp4	mp4				
Android	< HD > 64MB	mp4	mp4				
Android	720HD < 64MB	mp4	mp4				
Chrome	< 64MB	mp4	mp4				
macOS	< 64MB	mov	mp4				
Windows 10	< 64MB	mp4	mp4				

CHAPTER III

METHODOLOGY

Video files sent over WhatsApp are modified in some form, and this paper seeks to determine how and to what extent.

Based on research, social media applications do not simply pass a file through its platform unchanged[13]. There are considerations for file standardization, concerns about privacy, issues of bandwidth at scale, and optimization of the network. And so, we know that the files will be changed in a way that best suits the objectives of the platform while remaining apparently the same to casual users. The important information that is being sought in this research is to determine how video files change through transmission so that we can know what to expect when examining video files purported to be sent through WhatsApp. This comparison highlights the differences in file handling to determine a pattern that may be useful for forensic purposes.

The methods of analysis chosen in this experiment were selected to best reflect the stated purpose; changes made to the video file's structure and bit streams. To this end special attention was paid to the file container, the metadata stored, the hierarchy of metadata, and the video and audio streams.

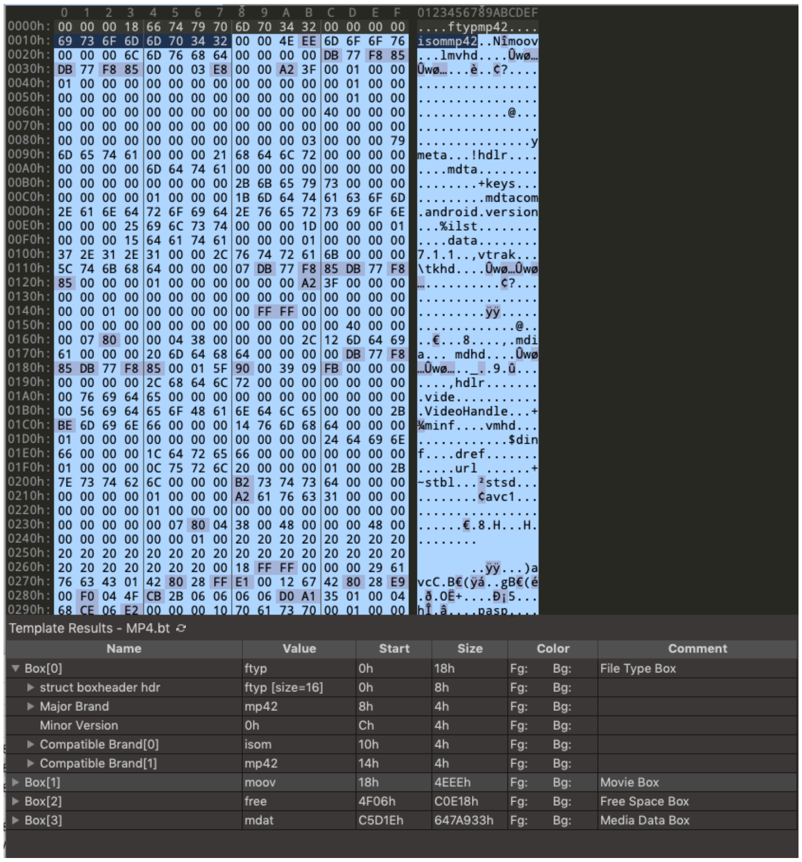
Software Used

WhatsApp Native and Web Applications

WhatsApp offers native standalone apps to access the service on Android, Macintosh & Windows operating systems. On Chrome OS there was no native application at the time of writing and so the web application was used to access the service. Neither the web application, Macintosh, or Windows applications offer video recording via WhatsApp, and so Open Camera was used to capture video on these three devices.

Software Used for Analysis

For analysis purposes 010 Editor[29] was used to examine the hexadecimal information along with the file structure.



```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
0000h: 00 00 00 18 66 74 79 70 6D 70 34 32 00 00 00 00  ....ftypmp42....
0010h: 69 73 6F 6D 6D 70 34 32 00 00 4E EE 6D 6F 6F 76  isommp42..Nîmoov
0020h: 00 00 00 6C 6D 76 68 64 00 00 00 00 DB 77 F8 85  ...lmvhd....Ûwø…
0030h: DB 77 F8 85 00 00 03 E8 00 00 A2 3F 00 01 00 00  Ûwø…...è..¢?....
0040h: 01 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00  ................
0050h: 00 00 00 00 00 00 00 00 00 00 00 00 00 01 00 00  ................
0060h: 00 00 00 00 00 00 00 00 00 00 00 00 40 00 00 00  ............@...
0070h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0080h: 00 00 00 00 00 00 00 00 00 00 00 03 00 00 00 79  ...............y
0090h: 6D 65 74 61 00 00 00 21 68 64 6C 72 00 00 00 00  meta...!hdlr....
00A0h: 00 00 00 00 6D 64 74 61 00 00 00 00 00 00 00 00  ....mdta........
00B0h: 00 00 00 00 00 00 00 00 2B 6B 65 79 73 00 00 00  ........+keys...
00C0h: 00 00 00 00 01 00 00 00 1B 6D 64 74 61 63 6F 6D  .........mdtacom
00D0h: 2E 61 6E 64 72 6F 69 64 2E 76 65 72 73 69 6F 6E  .android.version
00E0h: 00 00 00 25 69 6C 73 74 00 00 00 1D 00 00 00 01  ...%ilst........
00F0h: 00 00 00 15 64 61 74 61 00 00 00 01 00 00 00 00  ....data........
0100h: 37 2E 31 2E 31 00 00 2C 76 74 72 61 6B 00 00 00  7.1.1..,vtrak...
0110h: 5C 74 6B 68 64 00 00 00 07 DB 77 F8 85 DB 77 F8  \tkhd....Ûwø…Ûwø
0120h: 85 00 00 00 01 00 00 00 00 00 00 A2 3F 00 00 00  …..........¢?...
0130h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0140h: 00 00 01 00 00 00 00 00 00 FF FF 00 00 00 00 00  .........ÿÿ.....
0150h: 00 00 00 00 00 00 00 00 00 00 00 00 00 40 00 00  .............@..
0160h: 00 07 80 00 00 04 38 00 00 00 00 2C 12 6D 64 69  ..€...8....,.mdi
0170h: 61 00 00 00 20 6D 64 68 64 00 00 00 00 DB 77 F8  a... mdhd....Ûwø
0180h: 85 DB 77 F8 85 00 01 5F 90 00 39 09 FB 00 00 00  …Ûwø….._..9.û...
0190h: 00 00 00 00 2C 68 64 6C 72 00 00 00 00 00 00 00  ....,hdlr.......
01A0h: 00 76 69 64 65 00 00 00 00 00 00 00 00 00 00 00  .vide...........
01B0h: 00 56 69 64 65 6F 48 61 6E 64 6C 65 00 00 00 2B  .VideoHandle...+
01C0h: BE 6D 69 6E 66 00 00 00 14 76 6D 68 64 00 00 00  ¾minf....vmhd...
01D0h: 01 00 00 00 00 00 00 00 00 00 00 00 24 64 69 6E  ............$din
01E0h: 66 00 00 00 1C 64 72 65 66 00 00 00 00 00 00 00  f....dref.......
01F0h: 01 00 00 00 0C 75 72 6C 20 00 00 00 01 00 00 2B  .....url ......+
0200h: 7E 73 74 62 6C 00 00 00 B2 73 74 73 64 00 00 00  ~stbl...²stsd...
0210h: 00 00 00 00 01 00 00 00 A2 61 76 63 31 00 00 00  ........¢avc1...
0220h: 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00 00  ................
0230h: 00 00 00 00 00 07 80 04 38 00 48 00 00 00 48 00  ......€.8.H...H.
0240h: 00 00 00 00 00 00 01 00 20 20 20 20 20 20 20 20  ........        
0250h: 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20                  
0260h: 20 20 20 20 20 20 20 00 18 FF FF 00 00 00 29 61         ..ÿÿ...)a
0270h: 76 63 43 01 42 80 28 FF E1 00 12 67 42 80 28 E9  vcC.B€(ÿá..gB€(é
0280h: 00 F0 04 4F CB 2B 06 06 06 06 D0 A1 35 01 00 04  .ð.OË+....Ð¡5...
0290h: 68 CE 06 E2 00 00 00 10 70 61 73 70 00 01 00 00  hÎ.â....pasp....
```

Template Results - MP4.bt

Name	Value	Start	Size	Color	Comment
▼ Box[0]	ftyp	0h	18h	Fg: Bg:	File Type Box
▶ struct boxheader hdr	ftyp [size=16]	0h	8h	Fg: Bg:	
▶ Major Brand	mp42	8h	4h	Fg: Bg:	
Minor Version	0h	Ch	4h	Fg: Bg:	
▶ Compatible Brand[0]	isom	10h	4h	Fg: Bg:	
▶ Compatible Brand[1]	mp42	14h	4h	Fg: Bg:	
▶ Box[1]	moov	18h	4EEEh	Fg: Bg:	Movie Box
▶ Box[2]	free	4F06h	C0E18h	Fg: Bg:	Free Space Box
▶ Box[3]	mdat	C5D1Eh	647A933h	Fg: Bg:	Media Data Box

Figure 7. Metadata examination in 010 Editor

Media Conch was used to specifically check the hierarchy of the file structure. MediaInfo[30] was used to look at the specifications of the file's video and audio data. FFMPEG[31] was used to hash the video and audio streams. MATLAB[32] was used for keyword searching in the metadata.

Files Used

Based on preliminary tests done with data, it was determined that the dataset should be made up of several different files. The preliminary tests showed that certain video files that had a resolution of over 1280 x 720 pixels would be re-encoded by the Android application but in other instances this was not the case. Therefore, several files were recorded on Android to test the conditions under which re-encoding happens.

WhatsApp applications on Macintosh and Windows and the web application run on Chromebook only allowed for the uploading of video files smaller than 64MB. This contrasts with the information on WhatsApp FAQ page that states that WhatsApp only allows file sizes of under 16MB to be sent[33]. These applications also did not allow video recording, although they did allow for a photograph to be taken. Thus, any files sent through these applications had to be recorded in another application and must have been smaller than 64MB. This narrow criterion meant that it was only necessary to send one file from these applications.

The preliminary tests showed that the Android application is not so narrowly restricted. Based on the tests carried out the Android application allowed for files larger than 64MB to be uploaded from the phone's internal storage. It also allowed video files to be recorded in the application up to a limit of 64MB file size, with the restriction being enforced by stopping the recording when the limit is reached. Based on this, several files were sent from the Android device to test various conditions. The following conditions were tested: natively recorded over 64MB, natively recorded under 64MB, WhatsApp recorded up to 64MB, WhatsApp recorded under 64MB, natively recorded over 64MB smaller than HD, natively recorded under 64MB larger than HD, natively recorded under 64MB smaller than HD. Table 4 on the next page gives details about the video files recorded along with the criteria met.

Table 4. Files recorded in Android, along with criteria

Sending Platform	Recording Application	Resolution & File size
Android	Native Camera app	1080HD > 64MB
Android	WhatsApp	720 HD (WhatsApp recording)
Android	Native Camera app	1080HD < 64MB
Android	WhatsApp	Max length recording in WhatsApp
Android	Open Camera	< HD < 64MB
Android	Open Camera	< HD > 64MB
Android	Open Camera	720HD < 64MB

While general file hashing was useful for verifying the integrity of files transferred, stream hashing was used to verify the integrity of the video and audio payloads. If a file is transferred through WhatsApp the file's hash may change because of a change relating to the structure of the file, but the audio and video data may be identical to what was sent. To detect such differences, hashing of individual audio and video streams was employed.

Keywords were searched for in the metadata of the files. The MATLAB keyword search function is designed to examine the metadata of a file to find words that may have been left by software that may have processed the file.

Analysis

Applications Used

The files were analyzed for changes in container, structure, video and audio bits-streams. Hexadecimal examination was carried out using 010 Editor for its ability to employ user templates to assist in identifying different parts of the data chunks of a file. It is also useful in describing different parts of a file when examining the structure, highlighting the area under examination and showing user readable information in a clear manner. The files were cross-checked on another hexadecimal editor Hex Fiend[34] to ensure that the same information was shown.

Table 5. Software used for analysis

Application	Software Version
010 Editor	11.0
Android File Transfer	1.0.12 (1.0.507.1136)
MediaConch	18.03.2
MediaInfo	19.09
FFMPEG	4.3.1
Exiftool	12.07
Hex Fiend	2.8
MATLAB	R2019a update 5 (9.6.0.1174912)

Android File Transfer is a light-weight application made by Google to facilitate access to the file system through a desktop computer. To transfer files from the Android phones, these devices were connected to the computer via USB cable.

MediaConch[35] is a tool designed to check media files for compliance to standards. It was used for checking and displaying the structure of the files, showing the size of chucks and the hierarchy of metadata. This was cross-checked with information gathered from 010 Editor.

MediaInfo was chosen to gather metadata on the files. This software is a popular free tool that shows in depth information about files with the ability to arrange the views to best suit the needs of the user. Metadata about the file and the data streams was gathered with this software. Exiftool[36] was used to crosscheck the results of MediaInfo, however this was not as easily readable in comparison to and so was used to verify results. Additionally, it also showed more information which was beyond the scope of this experiment and thus was not used for heavy data gathering. To minimize the possibility of errors in this process MediaInfo was used for the ease of following the graphical user interface to gather information.

FFMPEG is a cross platform software package used for media recording, conversion and analysis. It is an open-source command line application that is designed to handle a wide array of media files.

MATLAB is a cross platform programing environment and language used for scientific processing and analysis. It was used in this experiment to search for keywords stored within the metadata. Any instance of the following text in *Table 6* will appear as being found if it exists within the examined files:

Table 6. Keywords used to search metadata

ACD	Ducky	imagenomic	noiseware	picnik	windows
ACDSee	Elements	iPhone	Paint	Picnik	XnView
Adobe	FastStone	Imaging	PaintShop	Prescan	Android
adobe	gd-jpeg	ImageIO.framework	paintshop	quicktime	android
Apple	GIMP	Irfan	photomapper	Quicktime	ANDROID
AppleMark	Gimp	IrfanView	Photomatix	QuickTime	Chrome
ashampoo	gimp	Jurnalul	photoscape	ScannerID	CHROME
bibble	http://ns.adobe.com	kipi	Photoscape	Snapseed	Chrome
Barcroft	http://ns.microsoft.com/photo	LEAD	PhotoScape	snapseed	Mac
capture	http://www.iec.ch	microsoft	Photoshop	Standard	mac
coachware	http://purl.org	MicrosoftPhoto	photoshop	Systems	MAC
commander	http://www.w3.org	MicrosoftPhoto:DateAcquired	PhotoSnap	Technologies	WhatsApp
Corel	imageready	MicrosoftPhoto:LastKeyword	photowatermark	ViewNX	whatsapp
corel	idimager	NeatImage	picasa	watermark	
digikam	idImager	Nero	Picasa	Windows	

Computer Used for Analysis

The computer used for analysis was an Apple iMac 27 inch (late 2013) running mac OS version 10.14.6.

Areas of Testing

Based on the hypothesis and the intended areas of analysis the following areas were tested and compared.

- File Structure
- Metadata chunk categories and sizes
- File hash checksum
- Audio/Video hash checksum
- File sizes and bitrates
- Resolution
- Codec
- Container
- Framerate
- Duration
- Color-space, chroma sub-sampling, bit-depth

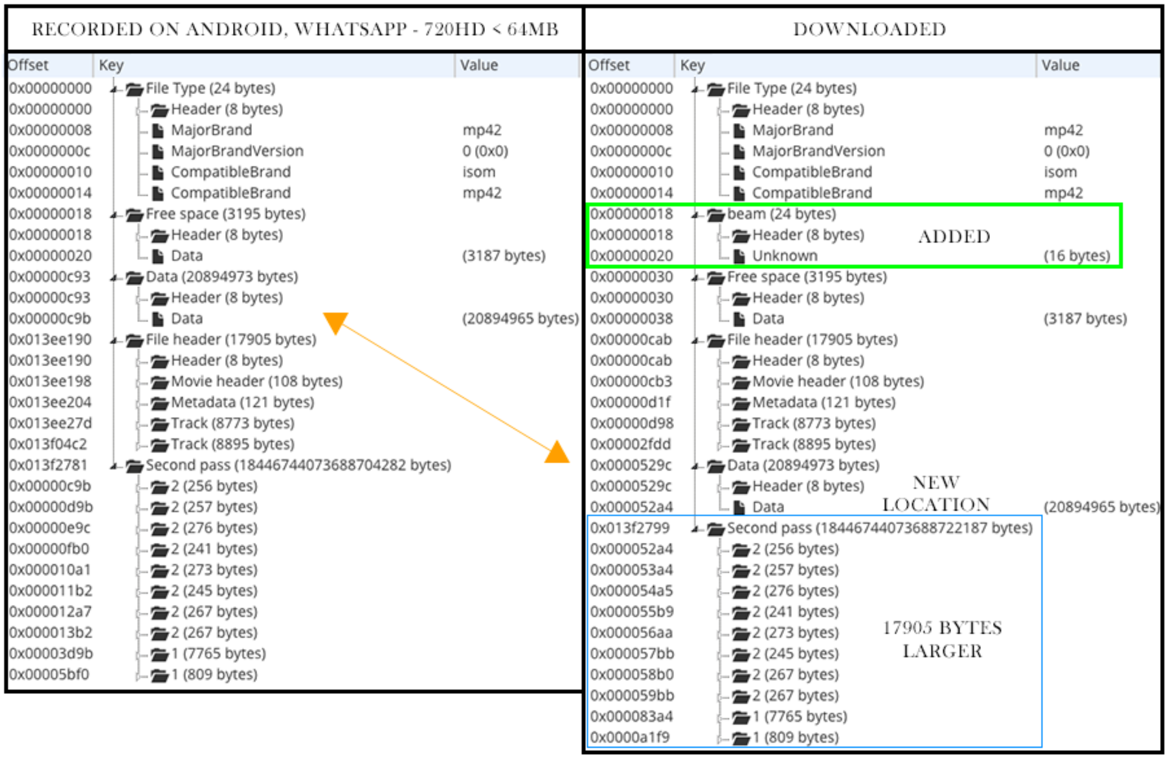
These areas of interest allow the files to be examined to gain understanding about the changes made to through transmission, and ultimately whether these changes are significant for the purpose of examination during investigation.

CHAPTER IV

RESULTS

Structure

The structure of all of the files changed once they were sent through WhatsApp, albeit to varying degrees based on the device used to send it.



RECORDED ON ANDROID, WHATSAPP - 720HD < 64MB

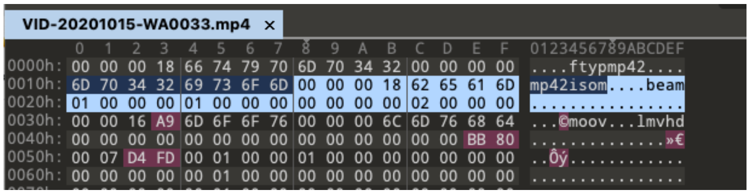
Offset	Key	Value
0x00000000	File Type (24 bytes)	
0x00000000	Header (8 bytes)	
0x00000008	MajorBrand	mp42
0x0000000c	MajorBrandVersion	0 (0x0)
0x00000010	CompatibleBrand	isom
0x00000014	CompatibleBrand	mp42
0x00000018	Free space (3195 bytes)	
0x00000018	Header (8 bytes)	
0x00000020	Data	(3187 bytes)
0x00000c93	Data (20894973 bytes)	
0x00000c93	Header (8 bytes)	
0x00000c9b	Data	(20894965 bytes)
0x013ee190	File header (17905 bytes)	
0x013ee190	Header (8 bytes)	
0x013ee198	Movie header (108 bytes)	
0x013ee204	Metadata (121 bytes)	
0x013ee27d	Track (8773 bytes)	
0x013f04c2	Track (8895 bytes)	
0x013f2781	Second pass (18446744073688704282 bytes)	
0x00000c9b	2 (256 bytes)	
0x00000d9b	2 (257 bytes)	
0x00000e9c	2 (276 bytes)	
0x00000fb0	2 (241 bytes)	
0x000010a1	2 (273 bytes)	
0x000011b2	2 (245 bytes)	
0x000012a7	2 (267 bytes)	
0x000013b2	2 (267 bytes)	
0x00003d9b	1 (7765 bytes)	
0x00005bf0	1 (809 bytes)	

DOWNLOADED

Offset	Key	Value
0x00000000	File Type (24 bytes)	
0x00000000	Header (8 bytes)	
0x00000008	MajorBrand	mp42
0x0000000c	MajorBrandVersion	0 (0x0)
0x00000010	CompatibleBrand	isom
0x00000014	CompatibleBrand	mp42
0x00000018	beam (24 bytes)	
0x00000018	Header (8 bytes)	
0x00000020	Unknown	(16 bytes)
0x00000030	Free space (3195 bytes)	
0x00000030	Header (8 bytes)	
0x00000038	Data	(3187 bytes)
0x00000cab	File header (17905 bytes)	
0x00000cab	Header (8 bytes)	
0x00000cb3	Movie header (108 bytes)	
0x00000d1f	Metadata (121 bytes)	
0x00000d98	Track (8773 bytes)	
0x00002fdd	Track (8895 bytes)	
0x0000529c	Data (20894973 bytes)	
0x0000529c	Header (8 bytes)	
0x000052a4	Data	(20894965 bytes)
0x013f2799	Second pass (18446744073688722187 bytes)	
0x000052a4	2 (256 bytes)	
0x000053a4	2 (257 bytes)	
0x000054a5	2 (276 bytes)	
0x000055b9	2 (241 bytes)	
0x000056aa	2 (273 bytes)	
0x000057bb	2 (245 bytes)	
0x000058b0	2 (267 bytes)	
0x000059bb	2 (267 bytes)	
0x000083a4	1 (7765 bytes)	
0x0000a1f9	1 (809 bytes)	

Figure 8. File structure analysis. Minimal changes made to the structure and data

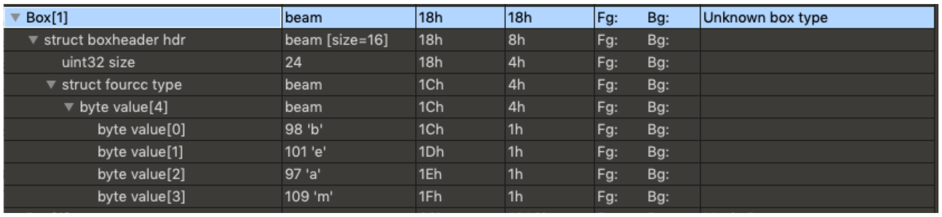
For files sent via Chrome web, and Windows 10 applications there was minimal change as shown in Figure 8. In each of these instances a **BEAM** atom was added to the metadata as in Figure 9 on the following page.



VID-20201015-WA0033.mp4 ×

```
        0  1  2  3  4  5  6  7  8  9  A  B  C  D  E  F   0123456789ABCDEF
0000h: 00 00 00 18 66 74 79 70 6D 70 34 32 00 00 00 00   ....ftypmp42....
0010h: 6D 70 34 32 69 73 6F 6D 00 00 00 18 62 65 61 6D   mp42isom....beam
0020h: 01 00 00 00 01 00 00 00 00 00 00 00 02 00 00 00   ................
0030h: 00 00 16 A9 6D 6F 6F 76 00 00 00 6C 6D 76 68 64   ...©moov...lmvhd
0040h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 BB 80   ..............»€
0050h: 00 07 D4 FD 00 01 00 00 01 00 00 00 00 00 00 00   ..Ôý............
0060h: 00 00 00 00 00 01 00 00 00 00 00 00 00 00 00 00   ................
```

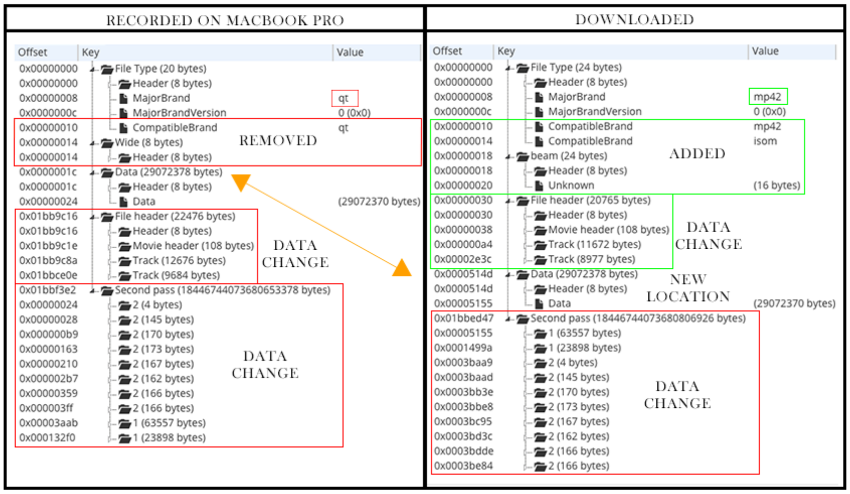
Figure 9. BEAM atom added in a downloaded file (010 Editor)



▼ Box[1]	beam	18h	18h	Fg:	Bg:	Unknown box type
▼ struct boxheader hdr	beam [size=16]	18h	8h	Fg:	Bg:	
uint32 size	24	18h	4h	Fg:	Bg:	
▼ struct fourcc type	beam	1Ch	4h	Fg:	Bg:	
▼ byte value[4]	beam	1Ch	4h	Fg:	Bg:	
byte value[0]	98 'b'	1Ch	1h	Fg:	Bg:	
byte value[1]	101 'e'	1Dh	1h	Fg:	Bg:	
byte value[2]	97 'a'	1Eh	1h	Fg:	Bg:	
byte value[3]	109 'm'	1Fh	1h	Fg:	Bg:	

Figure 10. Structure of the BEAM atom with details (010 Editor)

The BEAM atoms found seem to hold information of some significance, but without the ability to decode it there is no way of knowing what it means. Figure 9 and *Figure 10* show the same BEAM information from 2 different views.



RECORDED ON MACBOOK PRO

```
Offset       Key                                              Value
0x00000000   File Type (20 bytes)
0x00000000     Header (8 bytes)
0x00000008     MajorBrand                                     qt
0x0000000c     MajorBrandVersion                              0 (0x0)
0x00000010     CompatibleBrand                                qt
0x00000014   Wide (8 bytes)
0x00000014     Header (8 bytes)
0x0000001c   Data (29072378 bytes)
0x0000001c     Header (8 bytes)
0x00000024     Data                                           (29072370 bytes)
0x01bb9c16   File header (22476 bytes)
0x01bb9c16     Header (8 bytes)
0x01bb9c1e     Movie header (108 bytes)
0x01bb9c8a     Track (12676 bytes)
0x01bbce0e     Track (9684 bytes)
0x01bbf3e2   Second pass (18446744073680653378 bytes)
0x00000024     2 (4 bytes)
0x00000028     2 (145 bytes)
0x000000b9     2 (170 bytes)
0x00000163     2 (173 bytes)
0x00000210     2 (167 bytes)
0x000002b7     2 (162 bytes)
0x00000359     2 (166 bytes)
0x000003ff     2 (166 bytes)
0x00003aab     1 (63557 bytes)
0x000132f0     1 (23898 bytes)
```

DOWNLOADED

```
Offset       Key                                              Value
0x00000000   File Type (24 bytes)
0x00000000     Header (8 bytes)
0x00000008     MajorBrand                                     mp42
0x0000000c     MajorBrandVersion                              0 (0x0)
0x00000010     CompatibleBrand                                mp42
0x00000014     CompatibleBrand                                isom
0x00000018   beam (24 bytes)
0x00000018     Header (8 bytes)
0x00000020     Unknown                                        (16 bytes)
0x00000030   File header (20765 bytes)
0x00000030     Header (8 bytes)
0x00000038     Movie header (108 bytes)
0x000000a4     Track (11672 bytes)
0x00002e3c     Track (8977 bytes)
0x0000514d   Data (29072378 bytes)
0x0000514d     Header (8 bytes)
0x00005155     Data                                           (29072370 bytes)
0x01bbed47   Second pass (18446744073680806926 bytes)
0x00005155     1 (63557 bytes)
0x0001499a     1 (23898 bytes)
0x0003baa9     2 (4 bytes)
0x0003baad     2 (145 bytes)
0x0003bb3e     2 (170 bytes)
0x0003bbe8     2 (173 bytes)
0x0003bc95     2 (167 bytes)
0x0003bd3c     2 (162 bytes)
0x0003bdde     2 (166 bytes)
0x0003be84     2 (166 bytes)
```

Figure 11. Structural changes made to video recorded on MacBook Pro

Figure 11 shows more significant changes made to the video recorded on the MacBook Pro in Photo Booth. Some metadata atoms were removed, some replaced, and others discarded.

Neither the checksum of the file nor the audio stream of the file in *Figure 11* match the original file. However, the video stream remained the same, and so did all related video stream metadata.

When files were sent via the Android application there were different results based on the file size, resolution, and method of recording. Table 7 shows the changes which occurred to the videos created and sent from the Android platform. All files re-encoded had a change in video bitrate, resolution, and none of the file hashes matched the original file.

Table 7. List of all files recorded on Android phone with reasons for re-encoding

Sending Platform	**Recording Application**	**Resolution & File size**	**Notes**	**Re-encoded**
Android	Native Camera app	1080HD > 64MB	Re-encoded because it was brought in from internal storage, full HD resolution, and exceeded 64MB	yes
Android	WhatsApp	720 HD (WhatsApp recording)	Was not re-encoded because it was created in WhatsApp and did not exceed 64MB	no
Android	Native Camera app	1080HD < 64MB	Re-encoded because the resolution was full HD and came from internal memory	yes
Android	WhatsApp	Max length recording in WhatsApp	Recording exceeded 64MB and thus was re-encoded	yes
Android	Open Camera	< HD < 64MB	File was passed through with only a small change to metadata	no
Android	Open Camera	< HD > 64MB	File was re-encoded because file size exceeded 64MB	yes
Android	Open Camera	720HD < 64MB	File was re-encoded because resolution is HD and came from an internal memory	yes

File Size

File size was a critical factor in determining the processing that each file was subjected to while being sent. The file size of videos uploaded was limited to 64MB for media sent from Chrome web, Macintosh, or Windows 10 applications. Videos sent from Android that were larger than 64MB were all re-encoded, regardless of other factors.

Resolution

The resolution of files sent determined the treatment upon uploading in some instances. On Chrome web, Macintosh, and Windows 10 the resolution was of no consequence in this experiment (the only limit on uploading was file size). For the Android phone the resolution only affected the condition of re-encoding if the resolution was HD or larger and the video was uploaded from internal storage (as opposed to being recorded in WhatsApp).

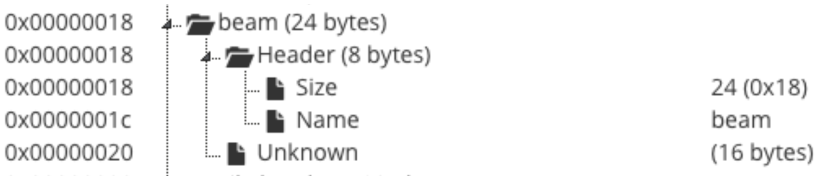
Method of Recording

The method of recording affected files on the Android platform only. For the videos recorded in WhatsApp there would be re-encoding only if the recording reached the file size limit. The limit of the file size is 64MB. Once the limit was not reached all videos recorded in WhatsApp were not re-encoded, but simply re-wrapped.

For files that were created on the Android phone via 'Camera' application and Open Camera there was no observed effect on the file related to the method of recording.

Metadata

The metadata of all files sent through WhatsApp was changed through the upload process, albeit in varying degrees. The files which saw the least changes were those created on the Android devices – there was the removal of the "FREE SPACE" atom, along with minimal rearrangement of metadata chunks, and an additional entry marked "BEAM".



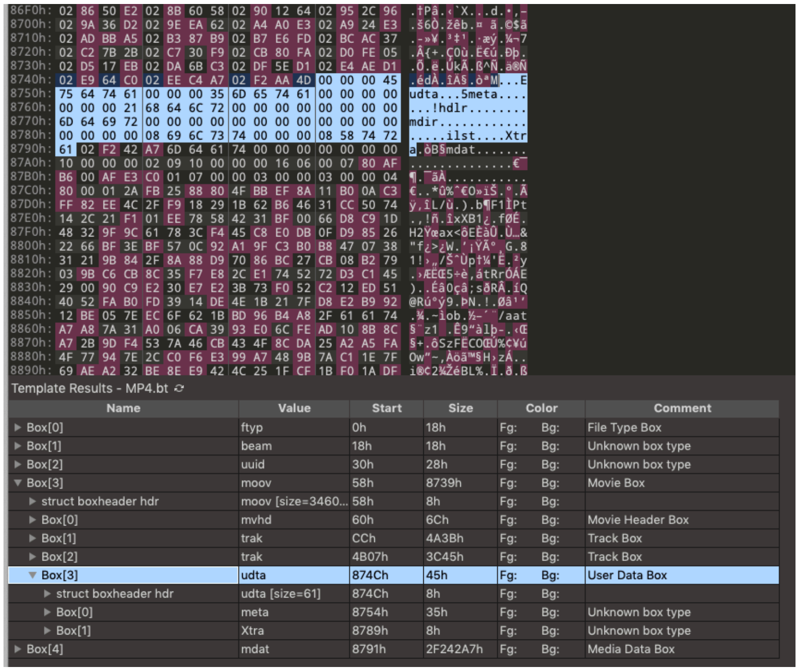
```
0x00000018   beam (24 bytes)
0x00000018     Header (8 bytes)
0x00000018       Size                                24 (0x18)
0x0000001c       Name                                beam
0x00000020     Unknown                               (16 bytes)
```

Figure 12. BEAM atom and its contents shown as a directory tree (MediaConch)

All files which were uploaded and sent inserted BEAM into the header, a 24-byte atom that holds information which was unable to be decoded in this experiment. The origin of the BEAM atom could not be fully verified, although some research pointed to the possibility that it may be from the Beamr[37] encoding tools which are utilized by some companies for media file optimization to make files smaller. The Beamr video encoder is not freely available, and a request to the company for an FFMPEG plugin was not responded to up until the time of publishing this paper. The BEAM atom is consistently present in all of the downloaded files examined.

Some instances of files had more metadata removed overall than others. The video created on the MacBook Pro went through numerous changes in file structure, as seen in *Figure 11.*

In the Windows file a "USER DATA" atom was observed. The information could not be decoded to determine whether the information identifies the sender of the file. This is shown on the next page in Figure 13.

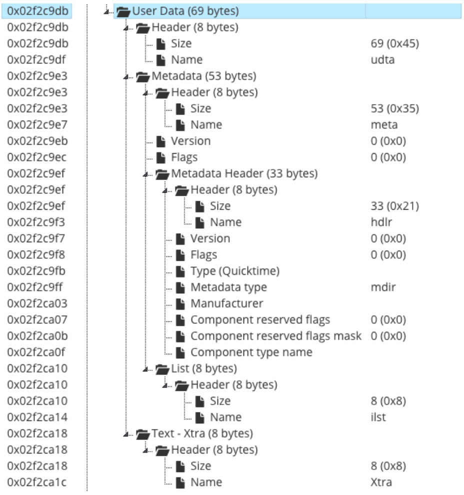


```
86F0h: 02 86 50 E2 02 8B 60 58 02 90 12 64 02 95 2C 96  .†Pâ.‹`X...d.•,–
8700h: 02 9A 36 D2 02 9E EA 62 02 A4 A0 E3 02 A9 24 E3  .š6Ò.žêb.¤ ã.©$ã
8710h: 02 AD BB A5 02 B3 87 B9 02 B7 E6 FD 02 BC AC 37  .»¥.³‡¹.·æý.¼¬7
8720h: 02 C2 7B 2B 02 C7 30 F9 02 CB 80 FA 02 D0 FE 05  .Â{+.Ç0ù.Ë€ú.Ðþ.
8730h: 02 D5 17 EB 02 DA 6B C3 02 DF 5E D1 02 E4 AE D1  .Õ.ë.ÚkÃ.ß^Ñ.ä®Ñ
8740h: 02 E9 64 C0 02 EE C4 A7 02 F2 AA 4D 00 00 00 45  .édÀ.îÄ§.òªM...E
8750h: 75 64 74 61 00 00 00 35 6D 65 74 61 00 00 00 00  udta...5meta....
8760h: 00 00 00 21 68 64 6C 72 00 00 00 00 00 00 00 00  ...!hdlr........
8770h: 6D 64 69 72 00 00 00 00 00 00 00 00 00 00 00 00  mdir............
8780h: 00 00 00 00 08 69 6C 73 74 00 00 00 08 58 74 72  .....ilst....Xtr
8790h: 61 02 F2 42 A7 6D 64 61 74 00 00 00 00 00 00 00  a.òB§mdat.......
87A0h: 10 00 00 00 02 09 10 00 00 00 16 06 00 07 80 AF  ..............€¯
87B0h: B6 00 AF E3 C0 01 07 00 00 03 00 00 03 00 00 04  ¶.¯ãÀ...........
87C0h: 80 00 01 2A FB 25 88 80 4F BB EF 8A 11 B0 0A C3  €..*û%ˆ€O»ïŠ.°.Ã
87D0h: FF 82 EE 4C 2F F9 18 29 1B 62 B6 46 31 CC 50 74  ÿ‚îL/ù.).b¶F1ÌPt
87E0h: 14 2C 21 F1 01 EE 78 58 42 31 BF 00 66 D8 C9 1D  .,!ñ.îxXB1¿.fØÉ.
87F0h: 48 32 9F 9C 61 78 3C F4 45 C8 E0 DB 0F D9 85 26  H2Ÿœax<ôEÈàÛ.Ù…&
8800h: 22 66 BF 3E BF 57 0C 92 A1 9F C3 B0 B8 47 07 38  "f¿>¿W.’¡ŸÃ°¸G.8
8810h: 31 21 9B 84 2F 8A 88 D9 70 86 BC 27 CB 08 B2 79  1!›„/ŠˆÙp†¼'Ë.²y
8820h: 03 9B C6 CB 8C 35 F7 E8 2C E1 74 52 72 D3 C1 45  .›ÆËŒ5÷è,átRrÓÁE
8830h: 29 00 90 C9 E2 30 E7 E2 3B 73 F0 52 C2 12 ED 51  )..Éâ0çâ;sðRÂ.íQ
8840h: 40 52 FA B0 FD 39 14 DE 4E 1B 21 7F D8 E2 B9 92  @Rú°ý9.ÞN.!.Øâ¹’
8850h: 12 BE 05 7E EC 6F 62 1B BD 96 B4 A8 2F 61 61 74  .¾.~ìob.½–´¨/aat
8860h: A7 A8 7A 31 A0 06 CA 39 93 E0 6C FE AD 10 8B 8C  §¨z1 .Ê9“àlþ.‹Œ
8870h: A7 2B 9D F4 53 7A 46 CB 43 4F 8C DA 25 A2 A5 FA  §+.ôSzFËCOŒÚ%¢¥ú
8880h: 4F 77 94 7E 2C C0 F6 E3 99 A7 48 9B 7A C1 1E 7F  Ow”~,Àöã™§H›zÁ..
8890h: 69 AE A2 32 BE 8E E9 42 4C 25 1F CF 1B F0 1A DF  i®¢2¾ŽéBL%.Ï.ð.ß
```

Template Results - MP4.bt

Name	Value	Start	Size	Color		Comment
▶ Box[0]	ftyp	0h	18h	Fg:	Bg:	File Type Box
▶ Box[1]	beam	18h	18h	Fg:	Bg:	Unknown box type
▶ Box[2]	uuid	30h	28h	Fg:	Bg:	Unknown box type
▼ Box[3]	moov	58h	8739h	Fg:	Bg:	Movie Box
▶ struct boxheader hdr	moov [size=3460...	58h	8h	Fg:	Bg:	
▶ Box[0]	mvhd	60h	6Ch	Fg:	Bg:	Movie Header Box
▶ Box[1]	trak	CCh	4A3Bh	Fg:	Bg:	Track Box
▶ Box[2]	trak	4B07h	3C45h	Fg:	Bg:	Track Box
▼ Box[3]	udta	874Ch	45h	Fg:	Bg:	User Data Box
▶ struct boxheader hdr	udta [size=61]	874Ch	8h	Fg:	Bg:	
▶ Box[0]	meta	8754h	35h	Fg:	Bg:	Unknown box type
▶ Box[1]	Xtra	8789h	8h	Fg:	Bg:	Unknown box type
▶ Box[4]	mdat	8791h	2F242A7h	Fg:	Bg:	Media Data Box

Figure 13. “User Data” located in the file created on the Windows computer

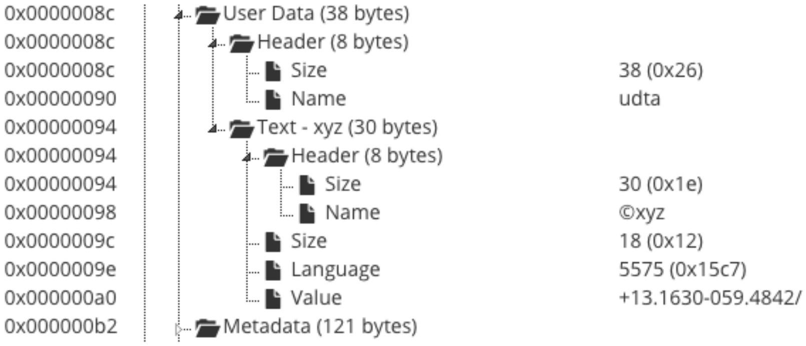


```
0x02f2c9db   User Data (69 bytes)
0x02f2c9db     Header (8 bytes)
0x02f2c9db       Size                              69 (0x45)
0x02f2c9df       Name                              udta
0x02f2c9e3     Metadata (53 bytes)
0x02f2c9e3       Header (8 bytes)
0x02f2c9e3         Size                            53 (0x35)
0x02f2c9e7         Name                            meta
0x02f2c9eb       Version                           0 (0x0)
0x02f2c9ec       Flags                             0 (0x0)
0x02f2c9ef       Metadata Header (33 bytes)
0x02f2c9ef         Header (8 bytes)
0x02f2c9ef           Size                          33 (0x21)
0x02f2c9f3           Name                          hdlr
0x02f2c9f7         Version                         0 (0x0)
0x02f2c9f8         Flags                           0 (0x0)
0x02f2c9fb         Type (Quicktime)
0x02f2c9ff         Metadata type                   mdir
0x02f2ca03         Manufacturer
0x02f2ca07         Component reserved flags        0 (0x0)
0x02f2ca0b         Component reserved flags mask   0 (0x0)
0x02f2ca0f         Component type name
0x02f2ca10       List (8 bytes)
0x02f2ca10         Header (8 bytes)
0x02f2ca10           Size                          8 (0x8)
0x02f2ca14           Name                          ilst
0x02f2ca18     Text - Xtra (8 bytes)
0x02f2ca18       Header (8 bytes)
0x02f2ca18         Size                            8 (0x8)
0x02f2ca1c         Name                            Xtra
```

Figure 14. "User Data" atom structure of file created on Windows computer

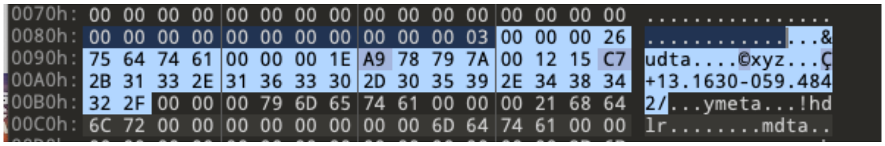
In Figure 14 the “USER DATA” atom is shown with the data in a ‘tree’ structure, showing the category and hierarchy of information. This information remained intact through the transfer of the file through WhatsApp. There was no observed data that identified the source of the file, or any other user specific information in this atom.

In an Android file a "USER DATA" atom was also observed with noticeably different information. Figure 15 below shows less data than the Windows file shown in Figure 14 on the previous page, but it seems to be more specific than the information in the Windows file's "USER DATA" atom. However, the information in the Android file is not human readable would need to be decoded to ascertain what it represents. Figure 16 below shows the same information in a hexadecimal editor.



```
0x0000008c    User Data (38 bytes)
0x0000008c      Header (8 bytes)
0x0000008c        Size                          38 (0x26)
0x00000090        Name                          udta
0x00000094      Text - xyz (30 bytes)
0x00000094        Header (8 bytes)
0x00000094          Size                        30 (0x1e)
0x00000098          Name                        ©xyz
0x0000009c        Size                          18 (0x12)
0x0000009e        Language                      5575 (0x15c7)
0x000000a0        Value                         +13.1630-059.4842/
0x000000b2    Metadata (121 bytes)
```

Figure 15. "USER DATA" in Android file



```
0070h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00  ................
0080h: 00 00 00 00 00 00 00 00 00 00 00 03 00 00 00 26  ...............&
0090h: 75 64 74 61 00 00 00 1E A9 78 79 7A 00 12 15 C7  udta....©xyz...Ç
00A0h: 2B 31 33 2E 31 36 33 30 2D 30 35 39 2E 34 38 34  +13.1630-059.484
00B0h: 32 2F 00 00 00 79 6D 65 74 61 00 00 00 21 68 64  2/...ymeta...!hd
00C0h: 6C 72 00 00 00 00 00 00 00 00 6D 64 74 61 00 00  lr........mdta..
```

Figure 16. Android "USER DATA" atom in hexadecimal form

A "USER DATA" atom was not found in all Android files, and there was no noticed pattern as to when this atom would be generated. Of note is the fact that the information was not passed on when the file was sent. The "USER DATA" atom was not found in any of the downloaded files.

File Hash

In all instances once a video file was sent through WhatsApp it experienced some level of change, and so the hash value changed regardless of the method of sending or the device used to send. However, the file hash of all files downloaded remained the same regardless of the devices used to download them. Table 8 shows the checksums of some original files compared to the downloaded versions of those files.

Table 8. Checksums of original files compared to the same videos downloaded

	Created on Chromebook	**Created on macOS**	**Created on Windows**
Original	cce6f00366dc9805e9ff54a3c049919d091a0e241551df4ad82bc852842c30cb	3aea128ceb0c62ba62d61f688db1caffa8704251fe9c63c93126470ee5ea45ae	05223ca2bb30dc1ffc489c28ce76adee13f33efe8af9f404b21ba7db5daca8bf
Downloaded on Chromebook	f45cdf9c1a86682562a7ec9f1a924978ed8953443a9d1d66b6bf26a000e89bb0	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202
Downloaded on Macintosh	f45cdf9c1a86682562a7ec9f1a924978ed8953443a9d1d66b6bf26a000e89bb0	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202
Downloaded on Windows	f45cdf9c1a86682562a7ec9f1a924978ed8953443a9d1d66b6bf26a000e89bb0	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202

Stream Hash

Video and audio stream hashes changed depending on the device of uploading along with the file size and resolution of the video – if conditions were met for re-encoding then the video/audio payload would be modified. When either of the stream checksums changed (suggesting re-encoding) there would also be a reflected change in resolution and bitrate, which resulted in a smaller file size. This demonstrated that a method of knowing whether a file was re-encoded was to simply compare the video stream checksum with the original. Table 9 on the next page provides the video stream checksum values of the dataset, comparing the original files with the downloaded files values.

Table 9. Correlation of video stream checksums and re-encoded status (yellow for re-encoded, green for not re-encoded)

Sending Platform	Recording Application	Resolution & Filesize	Original video stream checksum	Downloaded video stream checksum	Re-encoded	File codec & format of original video	File codec & format of received video
Android	Native Camera app	1080HD > 64MB	6e982f60dba3b0b7fa6d794e8d85b309ec67ce67687b6c513d75fa72776133b7	7d453e55a43c132237fde9a9354f5cb458dfb35323261543564bd97f9004d987	yes	AVC, .mp4	AVC, .mp4
Android	WhatsApp	720 HD (WhatsApp recording)	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b	no	AVC, .mp4	AVC, .mp4
Android	Native Camera app	1080HD < 64MB	f14bfd0e383b8bcafe761b3590900d35edffe6b2892abb88ca6ecc47ee4b56b4	24a2211641f3359fc545668336b2f36da3080f6bf59a8c87b74224832f64379c	yes	AVC, .mp4	AVC, .mp4
Android	WhatsApp	Max length recording in WhatsApp	c81b4ecc36144b22e9676e5f0310c023b96fd933d4fbeabd0324c02651055a9c	952a4028eaf7c09dc82777fab321c51e832662435daefaf8495a4ad6812695c7	yes	AVC, .mp4	AVC, .mp4
Android	Open Camera	< HD < 64MB	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0	no	AVC, .mp4	AVC, .mp4
Android	Open Camera	< HD > 64MB	ae98ccdc82435315eb9c8f16f8f1121e9a259d049e85214f0162ea47a7536785	f17f895db0b24a23566c407807e5d3fb91e150ccd561d6e26c5d13b6cf880c70	yes	AVC, .mp4	AVC, .mp4
Android	Open Camera	720HD < 64MB	9b3764d132bd48e66b3309c0b4322f66eb5536f26ecf1bc2d272fcf9b2fff5df	4d81ca2b399f186d4861b4f7be685f230303253b2e509816127175001492054c	yes	AVC, .mp4	AVC, .mp4
Chrome	Open Camera	< 64MB	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592	no	AVC, .mp4	AVC, .mp4
macOS	Photo Booth	< 64MB	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f	no	AVC, .mov	AVC, .mp4
Windows 10	Native Camera app	< 64MB	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b	no	AVC, .mp4	AVC, .mp4

File Size and Bitrates

Files sent via Chrome web, Macintosh, or Windows 10 application retained constant file sizes and bitrates. The file size and bitrate changed only in files that were re-encoded; this occurred in some of the files upload via Android. *Table 10* shows the files in which there was a reduction in file size and bitrate.

Table 10. Rate of loss of data shown by file size and video bitrate

Attributes		Original file size (MB)	File size after download (MB)	% of data lost	Original bitrate (Mbps)	Bitrate after download (Mbps)	% difference between new & old bitrates
Android	1080HD > 64MB	101	10	90.10%	20.4	2.025	90.07%
Android	720 HD	19.9	19.9	0.00%	3.8	3.8	0.00%
Android	1080HD < 64MB	25.3	2.57	89.84%	20.8	2.188	89.48%
Android	Max length recording	64.3	28.1	56.30%	3.789	1.659	56.22%
Android	< HD < 64MB	2.94	2.94	0.00%	2.479	2.479	0.00%
Android	< HD > 64MB	93.3	52.4	43.84%	2.105	1.183	43.80%
Android	720HD < 64MB	19.2	2.78	85.52%	14.9	2.184	85.34%
Chrome	< 64MB	42.1	42.1	0.00%	17.3	17.3	0.00%
macOS	< 64MB	27.7	27.7	0.00%	5.134	5.134	0.00%
Windows 10	< 64MB	47.2	47.2	0.00%	8.016	8.016	0.00%

Resolution

Resolution was key in determining the effect of re-encoding on the files sent. All files sent from the Chrome web, Macintosh, or Windows 10 applications saw no changes to resolution. From the Android application it was observed that resolutions higher than 720HD were re-encoded regardless of other any other factors if uploaded from internal memory. Files with a resolution lower than 720HD retained that resolution, once the file size was below 64MB. Where the criteria were met for re-encoding, in each case the resolution changed to retain the original aspect ratio, with the larger dimension of the video being limited to 640 pixels.

Table 11. Resolution change resulting from re-encoding after upload

Attributes		Original Resolution	Resolution after download	Re-encoded
Android	**1080HD > 64MB**	1920 x 1080	640 x 352	yes
Android	**720 HD**	1280 x 720	1280 x 720	no
Android	**1080HD < 64MB**	1920 x 1080	640 x 352	yes
Android	**Max length recording**	1280 x 720	640 x 352	yes
Android	**< HD < 64MB**	1024 x 768	1024 x 768	no
Android	**< HD > 64MB**	1024 x 768	640 x 480	yes
Android	**720HD < 64MB**	1280 x 720	640 x 352	yes
Chrome	**< 64MB**	1920 x 1080	1920 x 1080	no
macOS	**< 64MB**	1080 x 720	1080 x 720	no
Windows 10	**< 64MB**	1280 x 720	1280 x 720	no

Codec

The codec of all files sent was MPEG 4 part 10, Advanced Video Coding (AVC), otherwise known as H.264, and remained the same when downloaded.

File Format

The original file format of most of the files was MPEG4-part 12: ISO Base Media File Format, denoted by the extension MP4. The exception was the file recorded on the MacBook Pro, utilizing an MOV container. This MOV file was rewrapped as an MP4 file as illustrated in *Figure 11.*

Framerate

Original files that had a 'constant' framerate mode were changed to 'variable' once sent. Files that had a constant bitrate were those created on the Chromebook and Windows devices. The actual framerate of each file generally remained the same with a few slight differences. These differences are shown in Table 12 as being a variance of 0.001 frames.

Table 12. Framerate changes to files

Sending Platform	Recording Application	Resolution & File size	Original framerate (fps)	Downloaded framerate (fps)	Difference (fps)
Android	Native Camera app	1080HD > 64MB	30	30	0
Android	WhatsApp	720 HD (WhatsApp recording)	29.97	29.97	0
Android	Native Camera app	1080HD < 64MB	20.053	20.052	0.001
Android	WhatsApp	Max length recording in WhatsApp	25.885	25.884	0.001
Android	Open Camera	< HD < 64MB	29.923	29.923	0
Android	Open Camera	< HD > 64MB	29.93	29.93	0
Android	Open Camera	720HD < 64MB	29.931	29.93	0.001
Chrome	Open Camera	< 64MB	30	30	0
macOS	Photo Booth	< 64MB	20	20	0
Windows 10	Native Camera app	< 64MB	29.435	29.435	0

Duration

The length of the videos changed slightly in some cases, up to approximately 350 milliseconds. This didn't seem to be dependent on the status of re-encoding of video/audio streams, framerate changes, or any other measured factors. These changes seem negligible but are changes nonetheless which may be of significance in some other area. The information on length can be found in the appendix.

Color-space, Chroma Sub-sampling, Bit-depth

The color-space, chroma sub-sampling and bit-depth remained constant for files through upload and download. In all cases color-space was YUV, chroma sub-sampling 4:2:0, and bit-depth was 8 bits.

Keywords

The metadata of the files was analyzed with a 'keyword' search done through a script in MATLAB. The first 10,000 characters were examined by the software. Table 13 on the following page shows the keywords found, along with their respective offsets.

Table 13. Keywords found in metadata

Sending Platform	Recording Application	Resolution & File size	Keywords in Original	Keywords in Downloaded
Android	Native Camera app	1080HD > 64MB	Offset: D1 -> android	none found
Android	WhatsApp	720 HD (WhatsApp recording)	none found	Offset: D64 -> android
Android	Native Camera app	1080HD < 64MB	Offset: F7 -> android	none found
Android	WhatsApp	Max length recording in WhatsApp	none found	none found
Android	Open Camera	< HD < 64MB	Offset: D1 -> android	Offset: E9 -> android
Android	Open Camera	< HD > 64MB	Offset: D1 -> android	none found
Android	Open Camera	720HD < 64MB	Offset: D1 -> android	none found
Chrome	Open Camera	< 64MB	Offset: D1 -> android	Offset: E9 -> android
macOS	Photo Booth	< 64MB	none found	none found
Windows 10	Native Camera app	< 64MB	none found	none found

Results

Structural changes have been shown to occur with every video file sent through WhatsApp. The degree of change depends on the conditions under which the file is uploaded.

Changes observed ranged from minor structural changes with a re-wrapped video/audio streams, to a re-encoding of the video and audio payloads.

Chromebook, Macintosh and Windows Uploads

- Files uploaded could not exceed 64MB. (pg 18)
- Only minor structural changes were observed after download. (pg 23)
- Video stream checksum of downloaded files matched original file. (pg 51, 52, 53)
- Audio stream checksum of downloaded files matched original file. (pg 51, 52, 53)
- No observed resolution restrictions. (*Table 1*)

Android Upload

Files sent from internal storage which were smaller than 64MB and lower resolution than 720HD underwent minor structural changes along with a re-wrapping of the original video and audio streams.

Files recorded in WhatsApp and not exceeding 64MB underwent minor structural changes along with a re-wrapping of the original video and audio streams.

The following conditions introduced file re-encoding in addition to structural changes. (Table 7)

- Files uploaded from internal storage with 720HD resolution or greater.
- Files uploaded from internal storage larger than 64MB (regardless of resolution).
- Files recorded in WhatsApp and exceeding 64MB.

In all cases, all downloads of the same video produced identical files across devices. All file hashes matched regardless of download device, platform or time.

All files framerate mode was converted to variable if the originally uploaded file had a constant framerate.

Color space, chroma sub-sampling and pixel bit-depth all remained the same between the original and downloaded files.

Audio bitrate, sample rate, and channel count remained constant between the original and downloaded files.

Observations and Recommendations

Based on the experiments the following recommendations can be made:

- Any device used to download a file from WhatsApp will yield an identical file compared to another device downloading that same file and so the downloading method/device doesn’t matter.
- The file structure is consistent across downloaded files and may be useful as a starting point in case of any file authentication efforts where one is trying to determine a file's provenance or source in blind.
- All downloaded files contained a “BEAM” atom which was not present in the original file. This appears to be an atom added to the header through file optimization that seems to be built-in to WhatsApp.

All collected metadata can be viewed in the Appendix with comparison between the original file and the downloaded versions of it.

CHAPTER V

CONCLUSIONS

Based on the experiments carried out, the evidence shows that every video file that has been sent via WhatsApp undergoes changes. The degree to which these changes happen are based on the method of transmission and can be broken into 2 basic categories: pass-through and full re-encode.

Pass-through occurs where the file's video and audio streams remain unchanged, and the structure of the file is modified to include a chunk of metadata named "BEAM". There is also a stripping of some metadata from the file in some instances and a reordering of the metadata atoms depending on the original file. This means that the video and audio streams themselves are not changed. They are the same as the original video on the original device. It is the data "around" the streams that is changed.

Full re-encode occurs when the Android WhatsApp application changes the structure of the video file and re-encodes the video and audio streams. The video resolution and bitrate are changed, and the video/audio stream checksums are different in comparison to the original file. Structural changes observed depend on the original uploaded file's structure and can range from a simple re-ordering of metadata atoms to deletion of some metadata. In all instances a "BEAM" atom was added to the metadata.

When files were downloaded from all platforms used, the results were the same – file hashes matched regardless of the method of download, suggesting that no changes are made to the file by the downloading device. This also implies that the sending device makes the changes to video files - whether it be pass-through or full re-encode – and the downloading devices simply receives a copy of the already encoded file. For forensic purposes this signifies that any device can be used to acquire a file, once it is the same file sent.

Future Research

While a 'signature' could not be established from this research, it is the opinion of this researcher that there may be some potential for establishing a pattern of changes to all video files sent through WhatsApp as there are some common factors observed.

The "BEAM" atom added to all files sent holds some data albeit a small amount, but this may prove to be helpful in establishing a pattern.

All files created on the devices used in this experiment utilized the same media codecs, and there was only one MOV container while all others utilized MP4 containers. Various containers are common on mobile devices but were not covered due to the original scope of this research. It may be of some significance to examine other frequently used codecs and containers to establish a pattern of changes.

Forwarding video files and sending them to multiple accounts was beyond the scope of this research but is a common practice among users. This may be an area where other changes occur to a video file.

WhatsApp Statuses is a similar feature to Snapchat Stories – it allows users to post a piece of media for their contacts to see, not exceeding 30 seconds, and the media 'disappears' from the public view after 24 hours. Video files uploaded through this feature may be modified and can be an area research.

REFERENCES

[1] "What is text messaging (texting or wireless messaging)? - Definition from WhatIs.com," *SearchMobileComputing*. https://searchmobilecomputing.techtarget.com/definition/texting (accessed Oct. 06, 2020).

[2] "SMS (Short Message Service) Definition." https://techterms.com/definition/sms (accessed Oct. 21, 2020).

[3] "MMS (Multimedia Messaging Service) Definition." https://techterms.com/definition/mms (accessed Oct. 21, 2020).

[4] "44 Mind-Blowing SMS Marketing and Texting Statistics [UPDATED]," *SlickText*, Nov. 17, 2018. https://www.slicktext.com/blog/2018/11/44-mind-blowing-sms-marketing-and-texting-statistics/ (accessed Oct. 05, 2020).

[5] "Text Messaging, Premium Text Messaging, Text Alerts and Instant Messaging - Terms & Conditions." https://www.verizon.com/support/text-messaging-legal/ (accessed Nov. 11, 2020).

[6] "Every Android Phone Now Has RCS Messaging. Here's What You Should Know," *Fortune*. https://fortune.com/2019/12/16/androids-what-is-rcs-messaging/ (accessed Nov. 11, 2020).

[7] "What is RCS messaging, and why is it important to Android?," *Android Central*, Nov. 14, 2019. https://www.androidcentral.com/what-rcs-and-why-it-important-android (accessed Nov. 11, 2020).

[8] "What is RCS Messaging, and Exactly How Does It Work?," *Digital Trends*, Oct. 28, 2020. https://www.digitaltrends.com/?p=1211292 (accessed Nov. 11, 2020).

[9] "WhatsApp Revenue and Usage Statistics (2020)," *Business of Apps*, Aug. 08, 2017. https://www.businessofapps.com/data/whatsapp-statistics/ (accessed Oct. 05, 2020).

[10] "Features," *WhatsApp.com*. https://www.whatsapp.com/features/?lang=en (accessed Oct. 21, 2020).

[11] H. L. Risemberg, "FILE STRUCTURE ANALYSIS OF MEDIA FILES SENT AND RECEIVED OVER WHATSAPP," p. 59.

[12] A. Castiglione, G. Cattaneo, and A. D. Santis, "A Forensic Analysis of Images on Online Social Networks," in *2011 Third International Conference on Intelligent Networking and Collaborative Systems*, Nov. 2011, pp. 679–684, doi: 10.1109/INCoS.2011.17.

[13] M. Pippin, "DIGITAL IMAGE RECOMPRESSION ANALYSIS: FACEBOOK," p. 46.

[14] "Whatsapp Messenger on the Android Platform: A Forensic Examination of a Physical Device," *FSMT*, vol. 1, no. 1, pp. 04–19, Sep. 2018, doi: 10.33513/FSMT/1801-05.

[15] C. Anglano, “Forensic Analysis of WhatsApp Messenger on Android Smartphones,” *Digital Investigation*, vol. 11, no. 3, pp. 201–213, Sep. 2014, doi: 10.1016/j.diin.2014.04.003.

[16] “Android File Transfer,” *Android*. https://www.android.com/filetransfer/ (accessed Nov. 18, 2020).

[17] “Checksum Definition.” https://techterms.com/definition/checksum (accessed Oct. 22, 2020).

[18] “Ultimate Guide to Encoding: Fun with Container Formats - Part 1,” *Bitmovin*, Jun. 18, 2019. https://bitmovin.com/container-formats-fun-1/ (accessed Oct. 22, 2020).

[19] C. M. Whitecotton, “YOUTUBE: RECOMPRESSION EFFECTS,” p. 37.

[20] “Open Camera.” https://opencamera.org.uk/ (accessed Nov. 18, 2020).

[21] “Pixel Definition.” https://techterms.com/definition/pixel (accessed Oct. 22, 2020).

[22] “Resolution Definition.” https://techterms.com/definition/resolution (accessed Oct. 22, 2020).

[23] “High-definition television - New World Encyclopedia.” https://www.newworldencyclopedia.org/entry/High-definition_television#Modern_systems (accessed Oct. 22, 2020).

[24] “File Format Definition.” https://techterms.com/definition/file_format (accessed Oct. 22, 2020).

[25] “Video File Formats, Codecs, and Containers Explained | TechSmith,” *Welcome to the TechSmith Blog*, Jan. 09, 2018. https://www.techsmith.com/blog/video-file-formats/ (accessed Oct. 22, 2020).

[26] “What is metadata,” *WhatIs.com*. https://whatis.techtarget.com/definition/metadata (accessed Oct. 18, 2020).

[27] “Header.” https://www.fileformat.info/mirror/egff/ch03_02.htm (accessed Oct. 22, 2020).

[28] “Definition of bitstream,” *PCMAG*. https://www.pcmag.com/encyclopedia/term/bitstream (accessed Oct. 18, 2020).

[29] “010 Editor - Pro Text/Hex Editor | Edit 160+ Formats | Fast & Powerful.” https://www.sweetscape.com/010editor/ (accessed Nov. 18, 2020).

[30] “MediaInfo.” https://mediaarea.net/en/MediaInfo (accessed Nov. 18, 2020).

[31] “FFmpeg.” https://ffmpeg.org/ (accessed Nov. 18, 2020).

[32] “MathWorks - Makers of MATLAB and Simulink.” https://www.mathworks.com/ (accessed Nov. 18, 2020).

[33] "WhatsApp FAQ - I get a message that my video is too long and it won't send," *WhatsApp.com*. https://faq.whatsapp.com/general/i-get-a-message-that-my-video-is-too-long-and-it-wont-send/?lang=en (accessed Oct. 04, 2020).

[34] "Hex Fiend, a fast and clever hex editor for Mac OS X." https://ridiculousfish.com/hexfiend/ (accessed Nov. 18, 2020).

[35] "MediaConch." https://mediaarea.net/MediaConch (accessed Nov. 18, 2020).

[36] "ExifTool by Phil Harvey." https://exiftool.org/ (accessed Nov. 18, 2020).

[37] B. Video, "Beamr - Media Optimization for Better Monetization," *Beamr*. http://beamr.com (accessed Nov. 22, 2020).

APPENDIX

Data collected on file – Android, 1080HD > 64MB

File 1					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Android	Android	Android	Android	Android
Recording Application	Camera app	Camera app	Camera app	Camera app	Camera app
Attributes	1080HD > 64MB	1080HD > 64MB	1080HD > 64MB	1080HD > 64MB	1080HD > 64MB
File name	VID_20200904_095746	VID-20200904-WA0024	WhatsApp Video 2020-09-04 at 10.01.45	WhatsApp Video 2020-09-04 at 10.01.45	WhatsApp Video 2020-09-04 at 10.01.45
Keywords found	Offset: D1 -> android	none found	none found	none found	none found
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	101	10	10	10	10
Duration (s.ms)	41.535	41.508	41.508	41.508	41.508
Overall Bitrate - Mbps	20.4	2.025	2.025	2.025	2.025
File hash (SHA 256)	89c775c683d00cfdd3959724905716f103be12d40955728b04d9e049855f6bcc	d8ebac020c42c68a417a37cd5019b5f35cda82ece5b14a96267c392898e897f0	d8ebac020c42c68a417a37cd5019b5f35cda82ece5b14a96267c392898e897f0	d8ebac020c42c68a417a37cd5019b5f35cda82ece5b14a96267c392898e897f0	d8ebac020c42c68a417a37cd5019b5f35cda82ece5b14a96267c392898e897f0
Video stream hash (SHA 256)	6e982f60dba3b0b7fa6d794e8d85b309ec67ce67687b6c513d75fa72776133b7	7d453e55a43c132237fde9a9354f5cb458dfb35323261543564bd97f9004d987	7d453e55a43c132237fde9a9354f5cb458dfb35323261543564bd97f9004d987	7d453e55a43c132237fde9a9354f5cb458dfb35323261543564bd97f9004d987	7d453e55a43c132237fde9a9354f5cb458dfb35323261543564bd97f9004d987
Audio stream hash (SHA 256)	b8238f76aa9b968564dd1b76843c4aaa19f5884210d7bd5559b9bd8811e07e43	5f36a6dc1063a0d38c8f0a7fa0cb017a8892be8bb6a7a3fe0ce6bf5a04ce25a4	5f36a6dc1063a0d38c8f0a7fa0cb017a8892be8bb6a7a3fe0ce6bf5a04ce25a4	5f36a6dc1063a0d38c8f0a7fa0cb017a8892be8bb6a7a3fe0ce6bf5a04ce25a4	5f36a6dc1063a0d38c8f0a7fa0cb017a8892be8bb6a7a3fe0ce6bf5a04ce25a4
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1920 x 1080	640 x 352	640 x 352	640 x 352	640 x 352
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	20.2	1.925	1.925	1.925	1.925
Framerate	30	30	30	30	30
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Android, 720 HD < 64MB (WhatsApp recording)

File 2					
Source of file	**Original file**	**Downloaded on Android**	**Downloaded on Chromebook**	**Downloaded on Macintosh**	**Downloaded on Windows**
Original Operating System	Android	Android	Android	Android	Android
Recording Application	WhatsApp	WhatsApp	WhatsApp	WhatsApp	WhatsApp
Attributes	720 HD	720 HD	720 HD	720 HD	720 HD
File name	VID-20200904-WA0010	VID-20200904-WA0025	WhatsApp Video 2020-09-04 at 10.00.37	WhatsApp Video 2020-09-04 at 10.00.37	WhatsApp Video 2020-09-04 at 10.00.37
Keywords found	none found	Offset: D64 -> android	Offset: D64 -> android	Offset: D64 -> android	Offset: D64 -> android
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	19.9	19.9	19.9	19.9	19.9
Duration (s.ms)	44.031	44.031	44.031	44.031	44.031
Overall Bitrate - Mbps	3.8	3.8	3.8	3.8	3.8
File hash (SHA 256)	1a2e94e54f1f0a07d9dfe409e3f884067970af167d365dbb31d1f77e9c6ee6fc	d33641db83276453a44175150162cb16622404a125a953e15213ea6bc60418a0	d33641db83276453a44175150162cb16622404a125a953e15213ea6bc60418a0	d33641db83276453a44175150162cb16622404a125a953e15213ea6bc60418a0	d33641db83276453a44175150162cb16622404a125a953e15213ea6bc60418a0
Video stream hash (SHA 256)	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b	20c174164b9f9dc8fb6367da35e4262e96548b1f6abad8117ab206aff542431b
Audio stream hash (SHA 256)	98e1607d80ca601abb0be9c32ad96038c02f5099276 9ac9180100e0e85003349	98e1607d80ca601abb0be9c32ad96038c02f50992769ac9180100e0e85003349	98e1607d80ca601abb0be9c32ad96038c02f50992769ac9180100e0e85003349	98e1607d80ca601abb0be9c32ad96038c02f50992769ac9180100e0e85003349	98e1607d80ca601abb0be9c32ad96038c02f50992769ac9180100e0e85003349
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1280 x 720	1280 x 720	1280 x 720	1280 x 720	1280 x 720
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	3.711	3.711	3.711	3.711	3.711
Framerate	29.97	29.97	29.97	29.97	29.97
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Android, 1080HD > 64MB

File 3					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Android	Android	Android	Android	Android
Recording Application	Camera app	Camera app	Camera app	Camera app	Camera app
Attributes	1080HD < 64MB	1080HD < 64MB	1080HD < 64MB	1080HD < 64MB	1080HD < 64MB
File name	VID_20200929_230111	VID-20200930-WA0001	WhatsApp Video 2020-09-29 at 23.12.15 (1)	WhatsApp Video 2020-09-29 at 23.12.15 (1)	WhatsApp Video 2020-09-29 at 23.12.15 (1)
Keywords found	Offset: F7 -> android	none found	none found	none found	none found
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	25.3	2.57	2.57	2.57	2.57
Duration (s.ms)	10.176	9.841	9.841	9.841	9.841
Overall Bitrate - Mbps	20.8	2.188	2.188	2.188	2.188
File hash (SHA 256)	8f174058783427c916c5a9b8b4e4eb1315512e9b04a9d87de9bf8e5583860d0b	827f20771e3fb5421e41e6569d40c307f52fa8d392660 3d4b36472a03410a9c7	827f20771e3fb5421e41e6569d40c307f52fa8d3926603d4b36472a03410a9c7	827f20771e3fb5421e41e6569d40c307f52fa8d3926603d4b36472a03410a9c7	827f20771e3fb5421e41e6569d40c307f52fa8d3926603d4b36472a03410a9c7
Video stream hash (SHA 256)	f14bfd0e383b8bcafe761b3590900d35edffe6b2892abb88ca6ecc47ee4b56b4	24a2211641f3359fc545668336b2f36da3080f6bf59a8c87b74224832f64379c	24a2211641f3359fc545668336b2f36da3080f6bf59a8c87b74224832f64379c	24a2211641f3359fc545668336b2f36da3080f6bf59a8c87b74224832f64379c	24a2211641f3359fc545668336b2f36da3080f6bf59a8c87b74224832f64379c
Audio stream hash (SHA 256)	b3c7d09bf489c662f1e70330d1292472a856ece9319004da88b1227ba9970a9e	57700f8fc6c3cc328239100f5769678d18cb08f2fcb0332fd07578bb1ce3f2f7	57700f8fc6c3cc328239100f5769678d18cb08f2fcb0332fd07578bb1ce3f2f7	57700f8fc6c3cc328239100f5769678d18cb08f2fcb0332fd07578bb1ce3f2f7	57700f8fc6c3cc328239100f5769678d18cb08f2fcb0332fd07578bb1ce3f2f7
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1920 x 1080	640 x 352	640 x 352	640 x 352	640 x 352
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	20.7	2.102	2.102	2.102	2.102
Framerate	20.053	20.052	20.052	20.052	20.052
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Android, max length recording (WhatsApp)

File 4					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Android	Android	Android	Android	Android
Recording Application	WhatsApp	WhatsApp	WhatsApp	WhatsApp	WhatsApp
Attributes	Max length recording	Max length recording	Max length recording	Max length recording	Max length recording
File name	VID-20200929-WA0050	VID-20200930-WA0002	WhatsApp Video 2020-09-29 at 23.12.15	WhatsApp Video 2020-09-29 at 23.12.15	WhatsApp Video 2020-09-29 at 23.12.15
Keywords found	none found	none found	none found	none found	none found
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	64.3	28.1	28.1	28.1	28.1
Duration (m:s)	2:22	2:22	2:22	2:22	2:22
Overall Bitrate - Mbps	3.789	1.659	1.659	1.659	1.659
File hash (SHA 256)	bae856637bd3ec95b94b02a0ccdf3e1847dabbd8bc2555a8da6c9acdce96dff9	d1c027792a0ae7ff6a0442dc692dbdffec2e08e25397cd63f5a08e905d70d762	d1c027792a0ae7ff6a0442dc692dbdffec2e08e25397cd63f5a08e905d70d762	d1c027792a0ae7ff6a0442dc692dbdffec2e08e25397cd63f5a08e905d70d762	d1c027792a0ae7ff6a0442dc692dbdffec2e08e25397cd63f5a08e905d70d762
Video stream hash (SHA 256)	c81b4ecc36144b22e9676e5f0310c023b96fd933d4fbeabd0324c02651055a9c	952a4028eaf7c09dc82777fab321c51e832662435daefaf8495a4ad6812695c7	952a4028eaf7c09dc82777fab321c51e832662435daefaf8495a4ad6812695c7	952a4028eaf7c09dc82777fab321c51e832662435daefaf8495a4ad6812695c7	952a4028eaf7c09dc82777fab321c51e832662435daefaf8495a4ad6812695c7
Audio stream hash (SHA 256)	7c3cc1446bfbadf3b974a3b4547b119e00e4558e245e8832e7760608da05030d	c484c98e749d47c2eb1e25a59705e21963e7ec5454d4985c01aa66660be886d0	c484c98e749d47c2eb1e25a59705e21963e7ec5454d4985c01aa66660be886d0	c484c98e749d47c2eb1e25a59705e21963e7ec5454d4985c01aa66660be886d0	c484c98e749d47c2eb1e25a59705e21963e7ec5454d4985c01aa66660be886d0
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1280 x 720	640 x 352	640 x 352	640 x 352	640 x 352
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	3.692	1.56	1.56	1.56	1.56
Framerate	25.885	25.884	25.884	25.884	25.884
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Android, < HD < 64MB

File 5					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Android	Android	Android	Android	Android
Recording Application	Open Camera	Open Camera	Open Camera	Open Camera	Open Camera
Attributes	< HD < 64MB	< HD < 64MB	< HD < 64MB	< HD < 64MB	< HD < 64MB
File name	VID_20201011_072349	VID-20201011-WA0002	WhatsApp Video 2020-10-11 at 07.25.11	WhatsApp Video 2020-10-11 at 07.25.11	WhatsApp Video 2020-10-11 at 07.25.11
Keywords found	Offset: D1 -> android	Offset: E9 -> android	Offset: E9 -> android	Offset: E9 -> android	Offset: E9 -> android
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	2.94	2.94	2.94	2.94	2.94
Duration (s.ms)	9.941	9.941	9.941	9.941	9.941
Overall Bitrate - Mbps	2.479	2.479	2.479	2.479	2.479
File hash (SHA 256)	27ef4c1f501234651c2d69b9589197odc3e071ec89ba0e2ff39727dc6f2a384a	d5ceca1aaf79367ebf50a5614b0cc3c16c681ca7a02c36176c95597e739d78f0	d5ceca1aaf79367ebf50a5614b0cc3c16c681ca7a02c36176c95597e739d78f0	d5ceca1aaf79367ebf50a5614b0cc3c16c681ca7a02c36176c95597e739d78f0	d5ceca1aaf79367ebf50a5614b0cc3c16c681ca7a02c36176c95597e739d78f0
Video stream hash (SHA 256)	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0	a7b9b6976e69ae7923ab99e01ece8b51978143c3b54bd8069f795032366ec6a0
Audio stream hash (SHA 256)	f55761b3381a6570eba062f5fba1df1c2398f2384185c45671ff4f093efa62b7	f55761b3381a6570eba062f5fba1df1c2398f2384185c45671ff4f093efa62b7	f55761b3381a6570eba062f5fba1df1c2398f2384185c45671ff4f093efa62b7	f55761b3381a6570eba062f5fba1df1c2398f2384185c45671ff4f093efa62b7	f55761b3381a6570eba062f5fba1df1c2398f2384185c45671ff4f093efa62b7
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1024 x 768	1024 x 768	1024 x 768	1024 x 768	1024 x 768
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	2.075	2.075	2.075	2.075	2.075
Framerate	29.923	29.923	29.923	29.923	29.923
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Android, < HD > 64MB

File 6					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Android	Android	Android	Android	Android
Recording Application	Open Camera	Open Camera	Open Camera	Open Camera	Open Camera
Attributes	< HD > 64MB	< HD > 64MB	< HD > 64MB	< HD > 64MB	< HD > 64MB
File name	VID_20201011_073028	VID-20201011-WA0003	WhatsApp Video 2020-10-11 at 07.44.14	WhatsApp Video 2020-10-11 at 07.44.14	WhatsApp Video 2020-10-11 at 07.44.14
Keywords found	Offset: D1 -> android	none found	none found	none found	none found
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	93.3	52.4	52.4	52.4	52.4
Duration (m:s)	6m11s	6m11s	6m11s	6m11s	6m11s
Overall Bitrate - Mbps	2.105	1.183	1.183	1.183	1.183
File hash (SHA 256)	ac56bb78fb1120d857257af9ef6dfae2a519938764ce37fa9a3038a078df6fea	cae67e68828978c36224bba88c1462f4fcc78dfb5c09f93c163faf3b37f673d5	cae67e68828978c36224bba88c1462f4fcc78dfb5c09f93c163faf3b37f673d5	cae67e68828978c36224bba88c1462f4fcc78dfb5c09f93c163faf3b37f673d5	cae67e68828978c36224bba88c1462f4fcc78dfb5c09f93c163faf3b37f673d5
Video stream hash (SHA 256)	ae98ccdc82435315eb9c8f16f8f1121e9a259d049e85214f0162ea47a7536785	f17f895db0b24a23566c407807e5d3fb91e150ccd561d6e26c5d13b6cf880c70	f17f895db0b24a23566c407807e5d3fb91e150ccd561d6e26c5d13b6cf880c70	f17f895db0b24a23566c407807e5d3fb91e150ccd561d6e26c5d13b6cf880c70	f17f895db0b24a23566c407807e5d3fb91e150ccd561d6e26c5d13b6cf880c70
Audio stream hash (SHA 256)	dd0d580813e5133a867d3c288a4998a7fda6a9f1d55f5b0961d024c2e8678cf3	bfb90ba2dc62404342df038f8b7d0c095d03cd11f58107d70ebd930371e4a76d	bfb90ba2dc62404342df038f8b7d0c095d03cd11f58107d70ebd930371e4a76d	bfb90ba2dc62404342df038f8b7d0c095d03cd11f58107d70ebd930371e4a76d	bfb90ba2dc62404342df038f8b7d0c095d03cd11f58107d70ebd930371e4a76d
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1024 x 768	640 x 480	640 x 480	640 x 480	640 x 480
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	2.001	1.084	1.084	1.084	1.084
Framerate	29.93	29.93	29.93	29.93	29.93
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Android, 720 HD < 64MB

File 7					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Android	Android	Android	Android	Android
Recording Application	Open Camera	Open Camera	Open Camera	Open Camera	Open Camera
Attributes	720HD < 64MB	720HD < 64MB	720HD < 64MB	720HD < 64MB	720HD < 64MB
File name	VID_20201013_073641	VID-20201015-WA0033	WhatsApp Video 2020-10-13 at 07.37.38	WhatsApp Video 2020-10-13 at 07.37.38	WhatsApp Video 2020-10-13 at 07.37.38
Keywords found	Offset: D1 -> android	none found	none found	none found	none found
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	19.2	2.78	2.78	2.78	2.78
Duration (s.ms)	10.816	10.693	10.693	10.693	10.693
Overall Bitrate - Mbps	14.9	2.184	2.184	2.184	2.184
File hash (SHA 256)	147556 99cd88f44c54985 2422a10df72cd299d776ca808f958925377986758f3	b712c6fff5573b412613071afd9602af07f1f96682bef032247365of6e0b6df4	b712c6fff5573b412613071afd9602af07f1f96682bef032247365of6e0b6df4	b712c6fff5573b412613071afd9602af07f1f96682bef032247365of6e0b6df4	b712c6fff5573b412613071afd9602af07f1f96682bef032247365of6e0b6df4
Video stream hash (SHA 256)	9b3764d132bd48e66b3309c0b4322f66eb5536f26ecf1bc2d272fcf9b2fff5df	4d81ca2b399f186d4861b4f7be685f230303253b2e509816127175001492054c	4d81ca2b399f186d4861b4f7be685f230303253b2e509816127175001492054c	4d81ca2b399f186d4861b4f7be685f230303253b2e509816127175001492054c	4d81ca2b399f186d4861b4f7be685f230303253b2e509816127175001492054c
Audio stream hash (SHA 256)	b1c6baf11c03417237bd7a60fa42c5a6754a3cab0350bdf1c79b154983fc4488	2d186b22c369f18231e780c7218e057bc1b01ebe7424705fcb0a2dc4398b9601	2d186b22c369f18231e780c7218e057bc1b01ebe7424705fcb0a2dc4398b9601	2d186b22c369f18231e780c7218e057bc1b01ebe7424705fcb0a2dc4398b9601	2d186b22c369f18231e780c7218e057bc1b01ebe7424705fcb0a2dc4398b9601
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1280 x 720	640 x 352	640 x 352	640 x 352	640 x 352
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	14.6	2.09	2.09	2.102	2.102
Framerate	29.931	29.93	29.93	20.052	20.052
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Data collected on file – Chromebook, < 64MB

File 8					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Chrome	Chrome	Chrome	Chrome	Chrome
Recording Application	Open Camera	Open Camera	Open Camera	Open Camera	Open Camera
Attributes	< 64MB	< 64MB	< 64MB	< 64MB	< 64MB
File name	VID_20200908_190919	VID-20200910-WA0013	WhatsApp Video 2020-09-10 at 10.12.35	WhatsApp Video 2020-09-10 at 10.12.35	WhatsApp Video 2020-09-08 at 19.10.23
Keywords found	Offset: D1 -> android	Offset: E9 -> android	Offset: E9 -> android	Offset: E9 -> android	Offset: E9 -> android
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	42.1	42.1	42.1	42.1	42.1
Duration (s.ms)	20.405	20.405	20.405	20.405	20.405
Overall Bitrate - Mbps	17.3	17.3	17.3	17.3	17.3
File hash (SHA 256)	cce6f00366dc9805e9ff54a3c049919d091a0e241551df4ad82bc852842c30cb	f45cdf9c1a86682562a7ec9f1a924978ed895344 3a9d1d66b6bf26a000e89bb0	f45cdf9c1a86682562a7ec9f1a924978ed8953443a9d1d66b6bf26a000e89bb0	f45cdf9c1a86682562a7ec9f1a924978ed8953443a9d1d66b6bf26a000e89bb0	f45cdf9c1a86682562a7ec9f1a924978ed8953443a9d1d66b6bf26a000e89bb0
Video stream hash (SHA 256)	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592	de342475dcbfdc234797ca4abb1d6387c7a302c83da0d5ca0163631e37144592
Audio stream hash (SHA 256)	d05bd263c1401196d3a71e17c9340f045a1dea03b20a29c4bf4ee2e48a0e580c	d05bd263c1401196d3a71e17c9340f045a1dea03b20a29c4bf4ee2e48a0e580c	d05bd263c1401196d3a71e17c9340f045a1dea03b20a29c4bf4ee2e48a0e580c	d05bd263c1401196d3a71e17c9340f045a1dea03b20a29c4bf4ee2e48a0e580c	d05bd263c1401196d3a71e17c9340f045a1dea03b20a29c4bf4ee2e48a0e580c
Framerate mode (video)	Constant	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1920 x 1080	1920 x 1080	1920 x 1080	1920 x 1080	1920 x 1080
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	17	17	17	17	17
Framerate	30	30	30	30	30
Audio bitrate (Kbps)	96	96	96	96	96
Audio Sampling rate(KHz)	44.1	44.1	44.1	44.1	44.1
Audio Channels	1	1	1	1	1

Data collected on file – macOS, < 64MB

File 9					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	macOS	macOS	macOS	macOS	macOS
Recording Application	Photo Booth	Photo Booth	Photo Booth	Photo Booth	Photo Booth
Attributes	< 64MB	< 64MB	< 64MB	< 64MB	< 64MB
File name	Movie on 9-4-20 at 11.36 AM	VID-20200904-WA0019	WhatsApp Video 2020-09-04 at 11.53.15	WhatsApp Video 2020-09-04 at 11.53.15	WhatsApp Video 2020-09-04 at 11.53.15
Keywords found	none found	none found	none found	none found	none found
Container	MOV	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	27.7	27.7	27.7	27.7	27.7
Duration (s.ms)	44.762	44.768	44.768	44.768	44.768
Overall Bitrate - Mbps	5.2	5.199	5.199	5.199	5.199
File hash (SHA 256)	3aea128ceb0c62ba62d61f688db1caffa8704251fe9c63c93126470ee5ea45ae	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa	118d27968de89f8dfa848136053a7cb7440533db9793cfc3fa5c83eb9830aefa
Video stream hash (SHA 256)	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f	40ea64e8cc29f60f01fab905c664f4f330f6c868558f44b71917e0f087d4628f
Audio stream hash (SHA 256)	3e07ac1c1a0f5cbb2d2567c7e08a0a27643fe679c6fb661b02cdd6c78754d45e	bc3d30b28db13006a09721ca5c74034f08660978f07ca209b04a818b2c6a8c7b	bc3d30b28db13006a09721ca5c74034f08660978f07ca209b04a818b2c6a8c7b	bc3d30b28db13006a09721ca5c74034f08660978f07ca209b04a818b2c6a8c7b	bc3d30b28db13006a09721ca5c74034f08660978f07ca209b04a818b2c6a8c7b
Framerate mode (video)	Variable	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1080 x 720	1080 x 720	1080 x 720	1080 x 720	1080 x 720
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	5.134	5.134	5.134	5.134	5.134
Framerate	20	20	20	20	20
Audio bitrate (Kbps)	64	64	64	64	64
Audio Sampling rate(KHz)	44.1	44.1	44.1	44.1	44.1
Audio Channels	1	1	1	1	1

Data collected on file – Windows 10, < 64MB

File 10					
Source of file	Original file	Downloaded on Android	Downloaded on Chromebook	Downloaded on Macintosh	Downloaded on Windows
Original Operating System	Windows 10	Windows 10	Windows 10	Windows 10	Windows 10
Recording Application	Camera app	Camera app	Camera app	Camera app	Camera app
Attributes	< 64MB	< 64MB	< 64MB	< 64MB	< 64MB
File name	WIN_20200904_10_51_38_Pro	VID-20200904-WA0026	WhatsApp Video 2020-09-04 at 13.20.19	WhatsApp Video 2020-09-04 at 13.20.19	WhatsApp Video 2020-09-04 at 13.20.19
Keywords found	none found	none found	none found	none found	none found
Container	MPEG 4	MPEG 4	MPEG 4	MPEG 4	MPEG 4
File size - MB	47.2	47.2	47.2	47.2	47.2
Duration (s.ms)	49.367	49.367	49.367	49.367	49.367
Overall Bitrate - Mbps	8.016	8.016	8.016	8.016	8.016
File hash (SHA 256)	05223ca2bb30dc1ffc489c28ce76adee13f33efe8af9f404b21ba7db5daca8bf	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202	3324232408e8eeda7a4ce90e885fa00a63ccba2d52b6631bf2795a2d3fa2e202
Video stream hash (SHA 256)	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b	5381a9ab96ddf9b3a50eb92d60ea6fc91801e2865e0d4202591d092ecb540d5b
Audio stream hash (SHA 256)	3a244e368cce000de8bf1e4be3d1263558081e5ecb32fdf0353cb698b15a9505	3a244e368cce000de8bf1e4be3d1263558081e5ecb32fdf0353cb698b15a9505	3a244e368cce000de8bf1e4be3d1263558081e5ecb32fdf0353cb698b15a9505	3a244e368cce000de8bf1e4be3d1263558081e5ecb32fdf0353cb698b15a9505	3a244e368cce000de8bf1e4be3d1263558081e5ecb32fdf0353cb698b15a9505
Framerate mode (video)	Constant	Variable	Variable	Variable	Variable
Video Codec	AVC	AVC	AVC	AVC	AVC
Resolution	1280 x 720	1280 x 720	1280 x 720	1280 x 720	1280 x 720
Colorspace	YUV	YUV	YUV	YUV	YUV
Chroma subsampling	4:2:0	4:2:0	4:2:0	4:2:0	4:2:0
Video Bit depth (bits)	8	8	8	8	8
Video bitrate (Mbps)	7.899	7.899	7.899	7.899	7.899
Framerate	29.435	29.435	29.435	29.435	29.435
Audio bitrate (Kbps)	179	179	179	179	179
Audio Sampling rate(KHz)	48	48	48	48	48
Audio Channels	2	2	2	2	2

Details on devices used

Device Make	Device Model	Operating System	OS version	WhatsApp version	Function	IMEI	Serial Number
Nokia	6/TA-1000	Android	7.1.1	2.20.197.20/2.20.200.22	Sending only	355457087308345/355457087308337	D1CGAPE741500312
Xiaomi	Redmi Note 5 (unlocked bootloader)	Android	9 (MIUI Global 11.0.3.0)	2.20.197.20/2.20.200.22	Recieving only	869794036561627/869794036561629	e4d56a81
Hewlett Packard	Chromebook x2	Chrome OS	85.0.4183.133	2.2035.14	Sending & recieving	n/a	5CD8379QZJ
Apple	MacBook Pro 13inch, mid 2012	Macintosh	Mojave 10.14.6	2.2027.10	Sending only	n/a	C02J3DS3DTY4
Apple	iMac, 27 inch, Late 2013	Macintosh	Mojave 10.14.6	2.2035.15	Recieving only	n/a	DGKLL0PYF8JC
Acer	Aspire E5-575/N16Q2	Windows	10	2.2035.14	Sending & recieving	n/a	NXGG5AAOO57342OCEE76OO

Details on camera applications used

Device Make	Device Model	Operating System	OS version	Camera App used	Camera app version
Nokia	6/TA-1000	Android	7.1.1	Camera app (native)	7.0110.24
Nokia	6/TA-1000	Android	7.1.1	Camera App	1.48.2
Hewlett Packard	Chromebook x2	Chrome OS	85.0.4183.133	Open Camera	1.48.2
Apple	MacBook Pro 13inch, mid 2012	Macintosh	Mojave 10.14.6	Photo Booth	10.0(1009)
Acer	Aspire E5-575/N16Q2	Windows	10	Camera app (native)	N/A