

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ

Σχολή Μηχανικών

Τμήμα Μηχανικών Πληροφορικής και Υπολογιστών

Πρόγραμμα ΜεταπτυχΙακών Σπουδών (ΠΜΣ) Κυβερνοασφάλεια (Cybersecurity)

Πρώτες ενέργειες κατά την αντιμετώπιση συμβάντος ασφαλείας σχετιζόμενο με συσκευές Android

Συγγραφείς

&

Δονάτος Δόσης AM: 2010

Μιχαήλ Κωτσής

AM: 2015

Επιβλέπων:

Παναγιώτης Ριζομυλιώτης

Αθήνα, Φεβρουάριος 2022



UNIVERSITY OF WEST ATTICA **Faculty of Engineering** Department of Informatics and Computer Engineering Cybersecurity

First steps in incident response related to Android Based Devices

Students:

&

Δονάτος Δόσης

Μιχαήλ Κωτσής **R.N.: 2015**

R.N.: 2010

Supervisor:

Panagiotis Rizomiliotis

Athens, February 2022

Μέλη Εξεταστικής Επιτροπής συμπεριλαμβανομένου και του Εισηγητή

A/A	ΟΝΟΜΑ- ΕΠΩΝΥΜΟ	ҮПОГРАФН
1	ΠΑΝΑΓΙΩΤΗΣ ΡΙΖΟΜΥΛΙΩΤΗΣ	
2	ΠΑΝΑΓΙΩΤΗΣ ΓΙΑΝΝΑΚΟΠΟΥΛΟΣ	
3	ΣΤΕΦΑΝΟΣ ΓΚΡΙΤΖΑΛΗΣ	

Η μεταπτυχιακή διπλωματική εργασία εξετάστηκε επιτυχώς από την κάτωθι Εξεταστική Επιτροπή:

ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΩΝ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΕΡΓΑΣΙΑΣ

Οι κάτωθι υπογεγραμμένοι 1. **Δονάτος Δόσης** του Φωτίου με αριθμό μητρώου **2010** και 2. **Μιχαήλ Κωτσής** του Κωνσταντόινου, με αριθμό μητρώου **2015** φοιτητές του Προγράμματος Μεταπτυχιακών Σπουδών «Cybersecurity» («Κυβερνοασφλάλεια») του Τμήματος Μηχανικών Πληροφορικής και Υπολογιστών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνω ότι:

«Είμαστε συγγραφείς αυτής της μεταπτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχαμε για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες κάναμε χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνουμε ότι αυτή η εργασία έχει συγγραφεί από εμάς αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μας, όσο και του Ιδρύματος.

Παράβαση της ανωτέρω ακαδημαϊκής μας ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μας».

Επιθυμώ την απαγόρευση πρόσβασης στο πλήρες κείμενο της εργασίας μου μέχρικαι έπειτα από αίτηση μου στη Βιβλιοθήκη και έγκριση του επιβλέποντα καθηγητή.

Οι Δηλούντες

Table of Contents

Περίλ	ληψη		V
Ackn	owledg	ements	vii
Discl	aimer		viii
Table	es List		.ix
Figu	res List.		х
•			
Chap	oter 1. I	ntroduction in DFIR for Android Devices	. 1
		Response	
	1.1.1.	An Incident Response Plan	
	1.1.1.1	Preparation	
		Identification	
		Containment	
		Eradication	
		Recovery	
		Lessons Learned	
		IR Handling Report	
	1.1.2.	Incident Response Team	
	1.1.2.1	IR Team Model	
		Members of the IR Team	
		IR Team Preparing	
		IR Team Roles	
1.2.		Forensics	
1.2.	1.2.1.	Data acquisition	
	1.2.2.	A forensic image	
	1.2.2.1	Physical image (full file system)	
	1.2.2.1		
	1.2.2.2		
	1.2.3.	Partition	
	1.2.3.	Android File System	
	1.2.4.	Android Debug Bridge	
	1.2.5.	Android Boot Loader	
	1.2.0.	Known IR Practices on acquiring Android devices	
	1.2.7.		
		Device is powered off	
	1.2.7.2		
4.0		Device is on and locked	
1.3.		red Query Language (SQL) and SQLite Data	
	1.3.1.	SQLite Database Structure	
		Header	
		SQLite Pragma Statements	
		B-Tree Pages	
	1.3.1.4	Freelist Pages	13
		Atomic Commit in SQLite	
		Rollback Journal	
		Write Ahead Log (WAL) and Shared Memory (shm)	
1.4.		cs tools solutions	15
	1.4.1.	Android Mobile Devices Acquisition and Examination	
	1.4.2.	SQLite Databases Examination, Recovery and Analysis	17

2.1. Getting Root access 19 2.1.1. Getting root access on LG G6 19 2.1.2. Getting root access on Samsung A50 20 2.1.3. Wiping the device 20 2.1.3. Wiping the data via software 21 2.3. Extracting the data via software 24 2.3.1. Speed of imaging and transferring data 32 2.4. Examination of the acquired data. 33 Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.2. Experiments Goals 40 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. Whethodology 42 3.5.1. Viber 44 3.5.1. Viber 44 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3.1 General Information 46		pter 2. Preparation and Examination of an Android Device	
2.1.2. Getting root access on Samsung A50. 20 2.1.3. Wiping the device 20 2.2. Extracting the data manually. 21 2.3. Extracting the data via software 24 2.3. Stratacting the data via software 24 2.3. Speed of imaging and transferring data 32 2.4. Examination of the acquired data. 33 Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.1. The scenario 40 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber Viber 44 3.5.1. Viber communication Encryption 45 3.5.2. WhatsApp 3.5.2. WhatsApp 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3.1 General Information 46 3.5.3.3. Telegram's Concerns 47 <td>2.1.</td> <td></td> <td></td>	2.1.		
2.1.3. Wiping the device 20 2.2. Extracting the data manually 21 2.3. Extracting the data via software 24 2.3. Speed of imaging and transferring data 32 2.4. Examination of the acquired data. 33 Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber' s Communication Encryption 44 3.5.2. WhatsApp 45 3.5.2.1 General Information 46 3.5.2. WhatsApp 46 3.5.3. Telegram 46 3.5.3. Telegram's Communication Encryption 46 3.5.3. Telegram's Communication Encryption 46 3.5.3. Telegram's Communication Encryption 47 3.6. Applications' Cloud Servers 47 3.6. Applications' Cloud Servers 47 3.6. Applications' Cloud Servers 47 3.6. Application Seenario 50 4.1. Android 9 vs Android 11 Comparison		0	
2.2. Extracting the data manually. 21 2.3. Extracting the data via software 24 2.3.1. Speed of imaging and transferring data 32 2.4. Examination of the acquired data. 33 Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology. 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber's Communication Encryption 45 3.5.2. WhatsApp. 45 3.5.2.1. General Information 46 3.5.2.2. WhatsApp 46 3.5.3.1 General Information 46 3.5.3.2.1 General Information 46 3.5.3.3. Telegram 46 3.5.3.3. Telegram's Communication Encryption 46 3.5.3.3. Telegram 's Privacy Concerns. 47 3.6. Application Scenario 50 4.1. Viber's Examination 50 4.1.1. Android 9 vs Android 11 Comparison. 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of th			
2.3. Extracting the data via software. 24 2.3.1. Speed of imaging and transferring data 32 2.4. Examination of the acquired data. 33 Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1.2 Viber's Communication Encryption 45 3.5.2.2 WhatsApp. 45 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Applications' Cloud Servers 47 3.6. Applications' Cloud Servers 50 4.1.4. Volatility of the "viber_messages.db" 51 4.1.3. Analysis of the "viber_messages.db" 57 4.1.4. Volatility of t			
2.3.1. Špeed of imaging and transferring data 32 2.4. Examination of the acquired data. 33 Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp S Communication Encryption 46 3.5.2. WhatsApp S Communication Encryption 46 3.5.3.1 General Information 45 3.5.3.2 Telegram 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.4 General Information 46 3.5.3.5.3.7 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3	2.2.	Extracting the data manually	21
2.4. Examination of the acquired data	2.3.		
Chapter 3. Android Incident Response Scenario 40 3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber's Communication Encryption 45 3.5.2. WhatsApp. 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3. Telegram's Communication Encryption 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messag		2.3.1. Speed of imaging and transferring data	32
3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. General Information 44 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Application Scenario 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in questio	2.4.	Examination of the acquired data	33
3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. General Information 44 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Application Scenario 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in questio			
3.1. The scenario 40 3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. General Information 44 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Application Scenario 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in questio			
3.2. Experiments Goals 40 3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.1.1 General Information 44 3.5.1.2 Viber's Communication Encryption 45 3.5.2.2 WhatsApp 45 3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.4.4 Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 50 4.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db"			
3.3. Messaging Application Choice 41 3.4. Followed Methodology 42 3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1. Viber 44 3.5.1.1 General Information 44 3.5.2. WhatsApp 45 3.5.2. WhatsApp 45 3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 50 4.1. Viber's Examination 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 57 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature			
3.4. Followed Methodology. 42 3.5. General information about Viber, WhatsApp & Telegram. 44 3.5.1. Viber. 44 3.5.1. General Information 44 3.5.1. Viber's Communication Encryption 45 3.5.2. WhatsApp. 45 3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3.3 Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 50 4.1.1 Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 57 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature. 60 4.2. Viber's Media Files Attachments 56 4.2.1. LG device's media artifacts 67 <td>3.2.</td> <td></td> <td></td>	3.2.		
3.5. General information about Viber, WhatsApp & Telegram 44 3.5.1. Viber 44 3.5.2. Viber's Communication Encryption 45 3.5.2. WhatsApp. 45 3.5.2. WhatsApp 45 3.5.2. WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 WhatsApp's Communication Encryption 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 41.1 Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature 60 4.2. Viber's Media Files Attachments 65 4.2.1. LG device's media artifacts 66 4.2.1. LG device's media artifa	3.3.	Messaging Application Choice	41
3.5.1. Viber 44 3.5.1.1 General Information 44 3.5.1.2 Viber's Communication Encryption 45 3.5.2. WhatsApp 45 3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3.3 Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Application Scenario 50 4.1. Viber application Scenario 50 4.1.2 General Information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 54 4.1.4 Volatility of the "viber_messages.db" 57 4.1.5 Retrieving the messages in question and the timebomb feature 60 4.2.1 LG device's media artifacts 66 4.2.1 LG device's media artifacts 67	3.4.	Followed Methodology	42
3.5.1.1 General Information 44 3.5.1.2 Viber's Communication Encryption 45 3.5.2. WhatsApp 45 3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Application Scenario 50 4.1.1 Android 11 Comparison 50 4.1.2 General Information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 54 4.1.4 Volatility of the "viber_messages.db" 57 4.1.5 Retrieving the messages in question and the timebomb feature 60 4.2.1 LG device's media artifacts 66 4.2.1 Samsung device's media artifacts 67 <td>3.5.</td> <td></td> <td></td>	3.5.		
3.5.1.2 Viber's Communication Encryption .45 3.5.2. WhatsApp .45 3.5.2.1 General Information .45 3.5.2.2 WhatsApp's Communication Encryption .46 3.5.3. Telegram .46 3.5.3.1 General Information .46 3.5.3.2 Telegram's Communication Encryption .46 3.5.3.3 Telegram's Communication Encryption .46 3.5.3.3 Telegram's Concerns .47 3.6. Applications' Cloud Servers .47 3.6. Application Scenario .50 4.1. Viber application Scenario .50 4.1.1. Android 11 Comparison .50 4.1.2. General information about the under-examination database .51 4.1.3. Analysis of the "viber_messages.db" .54 4.1.4. Volatility of the "viber_messages.db" .57 4.1.5. Retrieving the messages in question and the timebomb feature .60 4.2. Viber's Media Files Attachments .65 4.2.1. LG device's media artifacts .66 4.2.1.		3.5.1. Viber	44
3.5.2. WhatsApp		3.5.1.1 General Information	44
3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Application Scenario 50 4.1. Viber application Scenario 50 4.1.1 Android 9 vs Android 11 Comparison 50 4.1.2 General information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 57 4.1.4 Volatility of the "viber_messages.db" 57 4.1.5 Retrieving the messages in question and the timebomb feature 60 4.2.1 LG device's media artifacts 66 4.2.1 LG device's media artifacts 67		3.5.1.2 Viber's Communication Encryption	45
3.5.2.1 General Information 45 3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 3.6. Application Scenario 50 4.1. Viber application Scenario 50 4.1.1 Android 9 vs Android 11 Comparison 50 4.1.2 General information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 57 4.1.4 Volatility of the "viber_messages.db" 57 4.1.5 Retrieving the messages in question and the timebomb feature 60 4.2.1 LG device's media artifacts 66 4.2.1 LG device's media artifacts 67		3.5.2. WhatsApp	45
3.5.2.2 WhatsApp's Communication Encryption 46 3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 A.1. Viber application Scenario 50 4.1.1 Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature 60 4.2.1 LG device's media artifacts 66 4.2.1. LG device's media artifacts 66			
3.5.3. Telegram 46 3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 50 4.1. Viber's Examination 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature 60 4.2. Viber's Media Files Attachments 65 4.2.1. LG device's media artifacts 66 4.2.1. Samsung device's media artifacts 67			
3.5.3.1 General Information 46 3.5.3.2 Telegram's Communication Encryption 46 3.5.3.3 Telegram's Privacy Concerns 47 3.6. Applications' Cloud Servers 47 Chapter 4. Viber application Scenario 47 Scenario 41.1 Android 9 vs Android 11 Comparison 41.2 General information about the under-examination database 41.3 Analysis of the "viber_messages.db" 41.4 Volatility of the "viber_messages.db" 41.5 Retrieving the messages in question and the timebomb feature 42. Viber's Media Files Attachments 42.1 LG device's media artifacts <td></td> <td></td> <td></td>			
3.5.3.2 Telegram's Communication Encryption. 46 3.5.3.3 Telegram's Privacy Concerns. 47 3.6. Applications' Cloud Servers. 47 Chapter 4. Viber application Scenario 47 Android 11 Comparison			
3.5.3.3 Telegram's Privacy Concerns. 47 3.6. Applications' Cloud Servers. 47 Chapter 4. Viber application Scenario 47 Chapter 4. Viber's Examination 41.1 Android 9 vs Android 11 Comparison 50 4.1.2 General information about the under-examination database 51 4.1.3 Analysis of the "viber_messages.db" 54 4.1.4 Volatility of the "viber_messages.db" 57 4.1.5 Retrieving the messages in question and the timebomb feature 60 4.2.1 LG device's media artifacts 66 4.2.1 Samsung device's media artifacts 67 <td></td> <td></td> <td></td>			
3.6. Applications' Cloud Servers			
Chapter 4. Viber application Scenario 50 4.1. Viber's Examination 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature 60 4.2. Viber's Media Files Attachments 65 4.2.1. LG device's media artifacts 66 4.2.1. Samsung device's media artifacts 67			
4.1. Viber's Examination 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature 60 4.2. Viber's Media Files Attachments 65 4.2.1. LG device's media artifacts 66 4.2.1. Samsung device's media artifacts 67	36	Applications' Cloud Servers	Δ7
4.1. Viber's Examination 50 4.1.1. Android 9 vs Android 11 Comparison 50 4.1.2. General information about the under-examination database 51 4.1.3. Analysis of the "viber_messages.db" 54 4.1.4. Volatility of the "viber_messages.db" 57 4.1.5. Retrieving the messages in question and the timebomb feature 60 4.2. Viber's Media Files Attachments 65 4.2.1. LG device's media artifacts 66 4.2.1. Samsung device's media artifacts 67	3.6.	Applications' Cloud Servers	47
4.1.1. Android 9 vs Android 11 Comparison	3.6.	Applications' Cloud Servers	47
4.1.2. General information about the under-examination database			
4.1.2. General information about the under-examination database	Cha	pter 4. Viber application Scenario	50
4.1.3.Analysis of the "viber_messages.db"	Cha	pter 4. Viber application Scenario	50 50
4.1.4.Volatility of the "viber_messages.db"	Cha	pter 4. Viber application Scenario Viber' s Examination 4.1.1. Android 9 vs Android 11 Comparison	50 50 50
 4.1.5. Retrieving the messages in question and the timebomb feature60 4.2. Viber's Media Files Attachments	Cha	pter 4. Viber application ScenarioViber' s Examination4.1.1.Android 9 vs Android 11 Comparison4.1.2.General information about the under-examination database	50 50 50 51
4.2. Viber's Media Files Attachments	Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db"	50 50 51 54
4.2.1.LG device's media artifacts664.2.1.Samsung device's media artifacts67	Cha	pter 4. Viber application Scenario Viber' s Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db"	50 50 51 54 57
4.2.1. Samsung device's media artifacts67	Cha 4.1.	pter 4. Viber application Scenario Viber' s Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature	50 50 51 54 57 60
	Cha 4.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments	50 50 51 54 57 60 65
	Cha 4.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments 4.2.1. LG device's media artifacts	50 50 51 54 60 65 66
	Cha 4.1. 4.2.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts	50 50 51 54 60 65 66 67
	Cha 4.1. 4.2.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts	50 50 51 54 60 65 66 67
	Cha 4.1. 4.2.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts	50 50 51 54 60 65 66 67
Chapter 5. WhatsApp application Scenario	Cha 4.1. 4.2. 4.3.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber	50 50 51 54 60 65 66 67 68
	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber	50 50 51 54 60 65 66 67 68
5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination	50 50 51 54 65 65 66 67 68 68
5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison. 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments. 4.2.1. LG device's media artifacts. 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison.	50 50 51 54 60 65 66 67 68 69 69 69
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison 5.1.2. General information about the under-examination database	50 50 51 54 60 65 66 67 68 69 69 69 69
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db"	50 50 51 54 60 65 66 67 68 69 69 69 69 69 69
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db"	50 50 51 54 65 65 66 67 68 69 69 69 69 69 69 70
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments. 4.2.1. LG device's media artifacts. 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber pter 5. WhatsApp application Scenario. WhatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison. 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db". 5.1.4. Volatility of the "msgstore.db"	50 50 51 54 60 65 66 67 68 69 69 69 69 69 70 73 74
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber pter 5. WhatsApp application Scenario WhatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db" 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup	50 50 51 54 57 60 65 65 66 67 68 69 69 69 69 70 74 74 80
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80 5.2. WhatsApp's Media Files Attachments 80	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison. 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber whatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db". 5.1.4. Volatility of the "msgstore.db" and the "wa.db". 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup. WhatsApp's Media Files Attachments.	50 50 51 54 60 65 66 67 68 69 69 69 69 69 69 70 73 74 80 80
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80 5.2. WhatsApp's Media Files Attachments 80 5.2.1. Media persistence on WhatsApp Server 81	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison. 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments. 4.2.1. LG device's media artifacts. 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber pter 5. WhatsApp application Scenario. WhatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db" and the "wa.db" 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup WhatsApp's Media Files Attachments 5.2.1. Media persistence on WhatsApp Server	50 50 51 54 65 65 66 67 68 69 69 69 69 69 69 70 73 74 80 81
Chapter 5. WhatsApp application Scenario	Cha 4.1. 4.2. 4.3.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber	50 50 51 54 60 65 66 67 68
	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber	50 50 51 54 60 65 66 67 68
5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination	50 50 51 54 65 65 66 67 68 68
5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison. 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments. 4.2.1. LG device's media artifacts. 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison.	50 50 51 54 60 65 66 67 68 69 69 69
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison 5.1.2. General information about the under-examination database	50 50 51 54 60 65 66 67 68 69 69 69 69
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db"	50 50 51 54 60 65 66 67 68 69 69 69 69 69 69
 5.1. WhatsApp's Examination	Cha 4.1. 4.2. 4.3. Cha	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber whatsApp's Examination 5.1.1. Android 9 vs Android 11 Comparison 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db"	50 50 51 54 65 65 66 67 68 69 69 69 69 69 69 70
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber pter 5. WhatsApp application Scenario WhatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db" 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup	50 50 51 54 57 60 65 65 66 67 68 69 69 69 69 70 74 74 80
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts 4.2.1. Samsung device's media artifacts Final Conclusions on Viber pter 5. WhatsApp application Scenario WhatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db" 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup	50 50 51 54 57 60 65 65 66 67 68 69 69 69 69 70 74 74 80
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80 5.2. WhatsApp's Media Files Attachments 80	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison. 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments 4.2.1. LG device's media artifacts 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber whatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db". 5.1.4. Volatility of the "msgstore.db" and the "wa.db". 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup. WhatsApp's Media Files Attachments.	50 50 51 54 60 65 66 67 68 69 69 69 69 69 69 70 73 74 80 80
5.1. WhatsApp's Examination 69 5.1.1. Android 9 vs Android 11 Comparison 69 5.1.2. General information about the under-examination database 69 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 70 5.1.4. Volatility of the "msgstore.db" 73 5.1.5. Retrieving the messages in question 74 5.1.6. WhatsApp Backup 80 5.2. WhatsApp's Media Files Attachments 80	Cha 4.1. 4.2. 4.3. Cha 5.1.	pter 4. Viber application Scenario Viber's Examination 4.1.1. Android 9 vs Android 11 Comparison. 4.1.2. General information about the under-examination database 4.1.3. Analysis of the "viber_messages.db" 4.1.4. Volatility of the "viber_messages.db" 4.1.5. Retrieving the messages in question and the timebomb feature. Viber's Media Files Attachments. 4.2.1. LG device's media artifacts. 4.2.1. Samsung device's media artifacts. Final Conclusions on Viber pter 5. WhatsApp application Scenario. WhatsApp's Examination 5.1.2. General information about the under-examination database 5.1.3. Analysis of the "msgstore.db" and the "wa.db" 5.1.4. Volatility of the "msgstore.db" and the "wa.db" 5.1.5. Retrieving the messages in question 5.1.6. WhatsApp Backup WhatsApp's Media Files Attachments 5.2.1. Media persistence on WhatsApp Server	50 50 51 54 65 65 66 67 68 69 69 69 69 69 69 70 73 74 80 81

	5.2.3.	Samsung device's media artifacts	82
5.3.	Final C	onclusions on WhatsApp	83

Chapter 6. Telegram application Scenario	84
6.1. Telegram's Examination	84
6.1.1. Android 9 vs Android 11 Comparison	84
6.1.2. General information about the under-examination database	84
6.1.3. Analysis of the "cache4.db"	85
6.1.3.1 Analysis of TDSs	
6.1.4. Volatility of the "cache4.db"	92
6.1.5. Retrieving the messages in question and the timebomb feat	ure92
6.1.6. Wiped Databases	
6.2. Telegram's Media Files Attachments	
6.2.1. Normal chat media transmission	96
6.2.2. Secret chat media transmission	
6.2.3. Final Conclusions on Telegram	
Chapter 7. Dual Applications Feature	100
Chapter 8. Applications Comparison	
Conclusions, Challenges & Future Work	103
References	105
Appendix	112
Appendix	

Abstract

Digital Forensics as a science is the process of acquiring, preserving, analyzing, and reporting of raw data residing in electronic or digital devices using scientific methods that are demonstrably reliable, verifiable and repeatable, such that they may be used in judicial and other format proceedings (SWGDE, 2014). One of the most particular circumstances that digital forensics must be applied is during the handling of a security incident where the operations of the first responders are extremely important and critical and a mistake is usually irreversible.

Mobile devices are in fact a computer in small form, equipped with high performance processor, huge storage, and enhanced functionality. They have become an essential part of the owner's live since they have drastically revolutionized the way that everyday activities (e.g., connecting with other people, e-banking) are achieved. As a result, a mobile device is now a huge repository that holds sensitive and personal information about its owner. A huge percentage of these personal and sensitive information is originated from social media and instant messaging applications like Viber, WhatsApp, and Telegram. In 2021, the number of mobile devices operating worldwide stood at almost 15 billion and their number is expected to reach 18.22 billion by 2025 (O' Dea S., 2021).

The huge proliferation of mobile devices and the social media and instant messaging applications have also shaped modern crime. Almost in every form of crime, bad actors make use of these technologies for planning and operating. This has, in turn, led to the development of a branch of digital forensics solely dedicated to mobile devices. Mobile forensics have unique characteristics and peculiarities since some of the core principles of other digital forensics do not apply like data preservation. Data inside a mobile device is continuously modified and the standard write protection techniques cannot be applied. For that reason, the actions of a first responder / examiner have even greater impact in the event of handling a mobile device as evidence.

The content of this Thesis is divided into four distinct parts. The first part introduces the concepts of digital criminology when dealing with a security incident, as well as basic concepts related to the forensic examination of a mobile phone device with Android operating system installed. The structure and basic elements of the SQLite database, on which most of the applications on such a device are based, are then analyzed. In addition, well-known and widespread forensic and non-forensic tools and software used to examine both Android devices and databases (SQLite) are presented. The second part presents suggested methods for obtaining administrative rights on two devices of different manufacturer (LG, Samsung) and different operating system version (Android 9 and Android 11), to create forensic copies (physical image) and examine them more effectively. In the third part, a description of a hypothetical scenario based on realistic incidents, the structure of the experiments and the followed methodology are given. Furthermore, in this part is explained the selection of the messaging applications Viber, WhatsApp and Telegram used for the experiments. The fourth part presents an analysis of the aforementioned messaging applications' operation, their examination and finally, the results of the experiments and a comparative analysis among them.

Key words: DFIR, First Responders, Mobile forensics, Android Applications, Viber, WhatsApp, Telegram.

Περίληψη

Η επιστήμη της ψηφιακής εγκληματολογίας (Digital Forensics) είναι η διαδικασία της συλλογής, διατήρησης, ανάλυσης και ερμηνείας δεδομένων που αποθηκεύονται σε ψηφιακή μορφή με επιστημονικό τρόπο, ώστε να μπορούν να αξιοποιηθούν σε νομικές διαδικασίες (SWGDE, 2014). Κύριος στόχος κατά την εξέταση μίας ψηφιακής συσκευής είναι η εξαγωγή και η ανάκτηση όλων των δεδομένων και πληροφοριών που βρίσκονται σε αυτή χωρίς όμως να επηρεαστεί η ίδια η συσκευή και τα δεδομένα που φέρει. Μία από τις πιο ιδιάζουσες εφαρμογές της ψηφιακής εγκληματολογίας είναι η αντιμετώπιση συμβάντων ασφαλείας (Incident Response), κατά την οποία οι ενέργειες των πρώτων ανταποκριτών είναι ιδιαίτερα σημαντικές και κάθε λάθος κίνηση συνήθως μη αναστρέψιμη.

Αναμφίβολα, στη σύγχρονη κοινωνία η πιο διαδεδομένη ψηφιακή συσκευή είναι το κινητό τηλέφωνο, το οποίο έχει μετατραπεί σε ψηφιακό συνεργάτη του ανθρώπου. Μία συσκευή κινητής τηλεφωνίας αποτελεί σήμερα ένα αποθετήριο με τεράστιο όγκο πληροφοριών, προερχόμενες σε μεγάλο ποσοστό από την αλληλεπίδραση του χρήστη μέσω εφαρμογών κοινωνικής δικτύωσης και ανταλλαγής μηνυμάτων, όπως για παράδειγμα το Viber, το WhatsApp και το Telegram. Ο τεράστιος όγκος πληροφοριών αυτών δικαιολογείται από το γεγονός ότι βρίσκεται σε συνεχή λειτουργία και διεπαφή με το χρήστη. Σήμερα, χρησιμοποιούνται σχεδόν δεκαπέντε (15) δισεκατομμύρια συσκευές κινητής τηλεφωνίας σε παγκόσμιο επίπεδο, ενώ εκτιμήσεις κάνουν λόγω για αύξηση του αριθμού πάνω από τα δεκαοκτώ (18) δισεκατομμύρια έως το 2025 (Ο' Dea S., 2021).

Η τεράστια διάδοση του κινητού τηλεφώνου και των εφαρμογών κοινωνικής δικτύωσης και ανταλλαγής μηνυμάτων έχουν διαμορφώσει και την σύγχρονη εγκληματικότητα. Σχεδόν σε όλα τα εγκλήματα η βασική μορφή επικοινωνίας μεταξύ των υπόπτων/δραστών πραγματοποιείται διαμέσου εφαρμογών ανταλλαγής μηνυμάτων που βρίσκονται εγκατεστημένες σε συσκευές κινητής τηλεφωνίας. Αυτή η ευρέα διάδοση των κινητών τηλεφώνων ως μέσο διευκόλυνσης ή ακόμη και διάπραξης εγκληματιαν οδήγησε στην ανάπτυξη ενός ξεχωριστού κλάδου ψηφιακής εγκληματολογίας, που εστιάζει αποκλειστικά σε αυτές (Mobile Forensics). Ο εν λόγω κλάδος έχει αρκετές ιδιαιτερότητες, καθώς βασικές αρχές αναιρούνται ή παρακάμπτονται, καθόσον τα δεδομένα της συσκευής συνεχώς μεταβάλλονται. Ως απόρροια αυτού, σε αντίθεση με τους γενικούς κανόνες της ψηφιακής εγκληματολογίας, τα δεδομένα είναι αδύνατο να παραμείνουν άθικτα σε κάθε πράξη. Έτσι, η κάθε επέμβαση από μεριάς των πρώτων ανταποκριτών και

Το περιεχόμενο της παρούσας εργασίας διαχωρίζεται σε τέσσερα διακριτά μέρη. Στο πρώτο μέρος εισάγονται οι έννοιες της ψηφιακής εγκληματολογίας κατά την αντιμετώπιση ενός περιστατικού ασφαλείας, καθώς επίσης αναφέρονται και βασικές έννοιες που σχετίζονται με την εγκληματολογική εξέταση μίας συσκευής κινητού τηλεφώνου με εγκατεστημένο λειτουργικό Android. Στη συνέχεια αναλύονται η δομή και τα βασικά στοιχεία της βάσης δεδομένων τύπου SQLite, στην οποία βασίζεται η πλειονότητα των εφαρμογών σε μία τέτοια συσκευή. Επιπλέον παρουσιάζονται γνωστά και ευρέως διαδεδομένα εγκληματολογικά και μη εργαλεία και λογισμικά που χρησιμοποιούνται για την εξέταση τόσο των Android συσκευών όσο και των βάσεων δεδομένων (SQLite). Στο δεύτερο μέρος παρουσιάζονται προτεινόμενοι μέθοδοι για απόκτηση διαχειριστικών δικαιωμάτων σε δύο συσκευές διαφορετικού κατασκευαστή (LG, Samsung) και διαφορετικής έκδοσης λειτουργικού συστήματος (Android 9 και Android 11), με απώτερο σκοπό την δημιουργία εγκληματολογικών αντιγράφων (physical image) και την αποτελεσματικότερη εξέτασή τους. Ακολούθως, στο τρίτο μέρος περιγράφονται ένα υποθετικό

σενάριο βασισμένο σε ρεαλιστικά περιστατικά, πάνω στο οποίο θα βασιστούν τα πειράματα που θα πραγματοποιηθούν στην παρούσα εργασία και η μεθοδολογία που ακολουθήθηκε τόσο για την πειραματική διαδικασία, όσο και για την επιλογή των εφαρμογών ανταλλαγής μηνυμάτων Viber, WhatsApp και Telegram. Στο τέταρτο μέρος παρουσιάζεται η ανάλυση, η εξέταση και τα αποτελέσματα των πειραμάτων για την εκάστοτε προαναφερθείσα εφαρμογή, καθώς επίσης παρουσιάζεται και η συγκριτική τους ανάλυση.

Λέξεις κλειδιά: DFIR, First Responders, Mobile forensics, Android Applications, ενέργειες πρώτου προσαράξαντα σε περιστατικό ασφαλείας, εγκληματολογική εξέταση κινητών, εφαρμογές Android, Viber, WhatsApp, Telegram.

Acknowledgements

We would like to express our sincere thankfulness to our beloved wives Helen and Spyridoula for supporting us to our educational journey and give us all the needed time and space.

Of course, want to say a big thanks to our professor P. Rizomiliotis who put his trust on our idea and support us till the end of this Thesis.

Last but not least comes our Chief Commander of our digital forensics' lab, A. Vasilaras without his will for everyday progress and excel of our tasks nothing of this would have been done.

Disclaimer

This Thesis has been for educational purposes and there is no intention to inspire persons acting malicious or illegal.

The methods are presented made it much easier to make conclusions and is not advised to use if there are not specific and reasoned purposes.

Unlocking and rooting a device can easily set the device unstable or even nonfunctioning. Moreover, the daily usage of these devices is defined as insecure since all security fail safes have been disabled.

Tables List

Table 1. Some examples of the evolution of Androids' partition tables. 8
Table 2. Cellebrite UFED 4PC user interface. Acquisition methods while the Samsung A50 device is connected
Table 3. OXYGEN Forensic Extractor user interface. Acquiring methods while the Samsung A50 device is connected 29
Table 4. MSAB XRY user interface. Acquiring methods while the LG G6 device is connected
Table 5. Chat applications' total downloads (Source: Google Play Store)41
Table 6. Chat application databases' characteristics (sort alphabetically)
Table 7. Chat application databases' main characteristics (sort alphabetically)
Table 8. Total conducted experiments' number
Table 9. WhatsApp, Telegram and Viber applications' cloud servers where the devices connected (tool used: ntopng with geoip plugin, device examined: Samsung A50)
Table 10. FBI' ability legally content access of Viber, WhatsApp and Telegram. Content of FBI's training document released on Jun. 2021 (Cimpanu C., 2021)
Table 11. Comparing settings of "viber_messages.db" on different versions of the application (tool used: SQL Expert)
Table 12. TDSs (Telegram Data Structure) objects in Normal Chat thread
Table 13. TDSs (Telegram Data Structure) objects in Secret Chat thread
Table 14. Media file TDS (Telegram Data Structure) objects in Secret Chat thread 90
Table 15. How to decode participant's "uid" from a Telegram's Secret Chat
Table 16. Comparison of how UFED PA (above) and Oxygen Forensics (below) were ableto decode different kind of TDS (Telegram Data Structure)
Table 17. Records of a Secret Chat in "cache4" database's WAL file
Table 18. The state of "cache4.db" and its WAL file in different conditions
Table 19. Progress of experiments' results on "cache4.db" and its WAL file. 95
Table 20. Correlation of a sent image file's record in "cache4.db" with "imgcache.sqlite" file (tool used: DB Browser for SQLite, Examined Device: LG G6)
Table 21. Detection of a sent image file in "cache4.db" from a Secret chat thread
Table 22. Retrieving artifacts from the "graph.db" for media transfers of Telegram's Normal Chat thread (tool used: DB Browser for SQLite, Examined Device: Samsung A50)
Table 23. The volatility of each application in different conditions

Figures List

Figure 1. Denial of access on Samsung A50 device's filesystem as simple user	8
Figure 2. Flow of Boot Sequence on Android Devices (González J., 2012)	9
Figure 3. Viber-messages SQLite database header parsed via X-Ways forensics Softwin hex mode.	
Figure 4. Viber-messages SQLite database header parsed via FQLite	12
Figure 5. SQLite Pragma Statements from DB Browser for SQLite interface	12
Figure 6. SQL journal transaction process flow	14
Figure 7. SQL -wal transaction process flow	15
Figure 8. SQL -shm transaction process flow	15
Figure 9. TWRP settings parameter "Use rm -rf instead of formatting"	20
Figure 10. Partition table view of LG G6 device via ADB command	21
Figure 11. Partition table symlinks view of LG G6 device via ADB command	22
Figure 12. Magnet Acquire user interface while LG G6 is connected	24
Figure 13. Speed monitoring while acquiring LG G6 device with MAGNET Acquire	32
Figure 14. Speed monitoring while acquiring Samsung A50 device with MAGNET Acq	
Figure 15. aLEAPP user interface	33
Figure 16. DB Browser for SQLite user interface (database: "viber_messages.db")	34
Figure 17. SQlite Expert user interface (database: "viber_messages.db")	34
Figure 18. Oxygen forensic SQlite Viewer user interface (database: "viber_messages.c	,
Figure 19. Magnet Axiom Forensics user interface (database: "viber_messages.db")	35
Figure 20. UFED Physical Analyzer user interface (database: "viber_messages.db")	36
Figure 21. Autopsy Suite user interface (database: "viber_messages.db")	36
Figure 22. Paraben E3 user interface (database: "viber_messages.db")	36
Figure 23. Belkasoft X user interface (database: "viber_messages.db")	37
Figure 24. X-ways Forensics user interface (database: "viber_messages.db")	38
Figure 25. SQLite Forensic Explorer user interface (database: "viber_messages.db")	38
Figure 26. FQLite user interface (database: "viber_messages.db")	38
Figure 27. Andriller user interface (default decoders for databases)	39
Figure 28. Forensic Toolkit for SQLite user interface.	39
Figure 29. Workflow of the applications' choice.	42
Figure 30. Workflow of methodology	44
Figure 31. Comparison of the "viber_messages-journal" file size while increasing the of main database with transaction between two (2) active users. The jo file is always 381 kb (Examined Device: LG Device).	urnal

Figure 32.	Comparison of "viber_messages.db" journal file's size while increasing the size of main database with transaction among four (4) active users on the Samsung A50 device
Figure 33.	Comparison of "viber_messages.db"-journal file's size while increasing the size of main database with transaction among four (4) active users on the LG G6 device
Figure 34.	Comparison of "viber_messages" database in Samsung A50 (left) and in LG G6 (right) (tool used: SQLite Expert)
Figure 35.	SQLite Pragma Statements of "viber_messages.db" in edit mode (tool used: DB Browser for SQLite)
Figure 36.	Depiction of the tables tree of "viber_messages.db"54
Figure 37.	Some of the columns that are consisted in "messages" table
Figure 38.	SQL query combining some of the columns and tables56
Figure 39.	Adding another case statement in the previous SQL query, it can be viewed if the message has been edited or not
Figure 40.	The device's "viber_messages.db" was not modified after 20 hours, even changing its state several times
Figure 41.	How "viber_messages.db" and its journal file are altered after one message was sent (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50) 58
Figure 42.	How "viber_messages.db" journal file was altered after one message was sent (tool used: FQLite, Examined Device: Samsung A50)
Figure 43.	How "viber_messages.db" journal file was altered after two messages were sent (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50)58
Figure 44.	How "viber_messages.db" journal file was altered after three messages were sent (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50)
Figure 45.	How "viber_messages.db" journal file was altered after three messages were opened by the receiver. By changing the "read_message_time" flag, all records which were previously unread are moved to the journal (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50)59
Figure 4	 Before removing Viber Notification: 18 messages existed in "viber_messages.db" and 16 messages existed in journal (tool used: Belkasoft-X, Examined Device: Samsung A50)
Figure 47	7. After having Viber Notification removed: 18 messages existed in "viber_messages.db" and 33 messages existed in it's journal file. Removing messages notification can potentially push something important out from the journal file by replacing it (tool used: Belkasoft-X, Examined Device: Samsung A50)60
Figure 48.	Use of timebombs and how they are depicted in the "viber_messages.db" (tool used: DB Browser for SQLite, Examined Device: Samsung A50)60
Figure 4	I9. Searching timebombs messages content using "user_id" in "viber_messages.db" (tool used: X-Ways Forensics, Examined Device: Samsung A50)61

- **Figure 50**. How "viber_messages" journal file was modified after messages with timebomb were deleted (tool used: FQLite Examined Device: Samsung A50) 62
- **Figure 51.** How "viber_messages" journal file was modified after timebomb messages were deleted (tool used: Belkasoft-X, Examined Device: Samsung A50) 62
- **Figure 52.** How "viber_messages" file was modified after timebomb messages were deleted (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50) 62

- Figure 57. How "viber_messages" journal file was modified five minutes after the timebomb messages were deleted (tool used: UFED PA, Examined Device: LG G6)64

- **Figure 60.** How "viber_messages" journal file is modified after the notification, of the previously received messages, is removed from the device. The data are further corrupted introducing false positive records (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50)......65
- Figure 66. Parsing "imagecache0.sqlite" (tool used: DB Browser for SQLite, Examined Device: LG G6)66

Figure 68.	The "MediaAttribute" table of Samsung "graph.db" (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50)68
Figure 69.	The "files" table of Samsung "core2.db" (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50)68
-	Analysis of "msgstore.db" properties (tools used: SQLite Expert (left) & DB Browser for SQLite (right))
Figure 71.	Depiction of the tables tree of "msgstore.db"
Figure 72.	Depiction of the tables tree of "wa.db"
Figure 73.	SQL Diagram of the joined tables (tool used: Forensic Browser for SQLite)72
Figure 74.	SQL query combining several columns from three different tables in two different databases
Figure 75.	Differences in the "msgstore.db" after the receipt of one message (tool used: KS DB Merge Tools)
-	Differences in the "msgstore.db" after sending one media file (tool used: KS DB Merge Tools)
Figure 77.	Parsing messages' content in "msgstore.db". The tool did not succeed to parse any of the records in question (tool used: FQLite, Examined Device: Samsung A50)
Figure 78.	Parsing messages' content in "msgstore.db". The tool managed to retrieve/parse some of the records in question (tool used: Magnet Axiom, Examined Device: Samsung A50)
Figure 79.	Parsing messages' content in "msgstore.db". The tool was able to retrieve all of the records in question (tool used: Belkasoft-X, Examined Device: Samsung A50)
Figure 77.	Searching for messages content using "key_remote_jid" in "msgstore.db" (tool used: X-Ways Forensics, Examined Device: Samsung A50)75
Figure 81.	The undeleted records after transmission using SQL queries in a reconstructed database made from "msgstore.db" and its WAL file (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50)
Figure 82.	How "msgstore.db" and its WAL file were modified after 2 hours with no activity by the user (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50)
Figure 83.	The "msgstore.db" and its corresponding WAL file were modified after several messages received/sent/read/deleted (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50) 78
Figure 84.	How "msgstore.db" and its WAL file are modified after a second deletion of a chat thread. The in-question messages were still retrievable. The messages 83 and 84 were removed (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50)
Figure 85.	How "msgstore.db" and its WAL file alters after a second deletion of a chat thread. The messages of interest (between 70 and 81) were removed. The deleted messages 83 and 84 were still retrievable (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: LG G6).79

Figure 86. Decrypting a crypt14 backup with free tool «WhatsApp Crypt14 Database Decrypter» (EIDavoo D., 2021)
Figure 87. The path that the WhatsApp transmitted files were stored (tool used: X-WAYS Forensic, Examined Device: LG G6)80
Figure 88. The path that the WhatsApp transmitted files' hashes were stored (tool used: X- WAYS Forensic, Examined Device: LG G6)81
Figure 89. The WhatsApp media files trash folder (tool used: X-WAYS Forensic, Examined Device: LG G6)
Figure 90. Download transmitted files via WhatsApp server
Figure 91. Media urls and decryption keys were stored on "message_media" table of "msgstore.db"
Figure 92. Decrypting downloaded from WhatsApp server media files, by using python script named «whatsapp-media-decrypt»
Figure 93. Viewing the "imgcache.sqlite" file (BLOB) (tool used: DB Browser for SQLite, Examined Device: LG G6)82
Figure 94.Exploring "MediaAttribute" table of Samsung "graph.db" (tool used: Oxygen Forensic SQlite, Examined Device: Samsung A50).83
Figure 95. Properties of "cache4.db" (tools used: SQLite Expert (left) & DB Browser for SQLite (right))85
Figure 96. Depiction of the tables tree of "cache4_db"
Figure 97. SQL query combining several columns from different tables from "cache4.db"
Figure 98. Use of dual Viber application. (Device used: Samsung A50)100
Figure 99. Depiction of second Viber account's separate databases (path: «/user/95/com.viber.voip/databases», Examined Device: Samsung A50) 100
Figure 100. The second Viber user (95) has a separate folder structure for media located in «/media/95/Adroid/data/com.viber.voip/file/User photos». The original user of the device is located in folder (0) (Examined Device: Samsung A50).

Chapter 1.

Introduction in DFIR for Android Devices

The acronym DFIR stands for Digital Forensics Incident Response and as its name states, is about forensics in digital evidence (hardware devices or software) during an incident response. In this chapter basic concepts about an incident response (IR) plan and team are presented. Moreover, there is an introduction to Android forensics basics, such as data acquisition methods, filesystems, partitions, connection methods (ADB) and known practices on how to prepare and acquire an android device during a security incident. Since the vast majority of android applications and operating systems utilizes the SQLite database format, a thorough forensic analysis of its structure and features is included. Finally, a brief reference of several forensic software and toolkits solutions for Android in general and SQLite in specific, is made.

1.1. Incident Response

An easy way to determine what the term "incident response" means exactly, is to think a how-to guide for a potential cyberattack in an organization. Is a plan in which are referred as many threats and scenario attacks as possible, with detailed plan on how to identify, restrain, prioritize, and eliminate each and every one of them (Cynet, 2021). Throughout the incident response plan, an organization make its employees be aware of security incidents in order to prevent severe attacks and be ready to spot as soon as possible an attacker and stop his acts. The purpose is to minimize the caused damage and the arisen loss.

1.1.1. An Incident Response Plan

In order to make an incident response plan an organization must determine what is the main reason of doing so. To be more specific if the main reason is to avoid data loss the procedures in the response plan should include a way to always keep secure a backup of the essential files. Of course, as everything in security, the accepted budget-to-invest is for sure something to bear in mind. Especially if the hole plan is undertaken from outside of organization experts. Below is presented a summary of common use strategy of making an incident response plan (Cynet, 2021).

1.1.1.1 Preparation

Firstly, there must be an assessment of risks and specification of critical incidents based on sensitive assets in order to build a team to handle all these incidents. This team is known also as "Computer Security Incident Response Team (CSIRT)". The preparation step should include:

• Making a strict and clear policy with rules, principles, and practices to help security processes. For example, there would be a displayed banner informing the users that every activity is being monitored and any unauthorized act or violation will be punished.

- A response strategy on how to prioritize and handle incidents according to its impact on organization. There is quite a difference between a login credentials hack of a simple user from an admin.
- A who-to-contact-with policy. In any instance that may occur there should be specific persons from organization's IR team or/and law enforcement.
- Creating documentation on who, what, when, where, why and how any procedure in organization works. If something is against this documentation, there may be an intruder's act.
- Making the IR Team a good one. By choosing members from different expertise (security, IT, human resources, public relations etc.), keep them up to date with training and access to professional tools, an organization make it more feasible to be protected from the known threats.

1.1.1.2 Identification

In this step the IR Team must detect any deviations from normal organization's operations and determine whether represents a security incident or not, and how important is. An identification procedure may include:

- A monitoring set up for all sensitive infrastructure.
- Setting up a platform to analyze and report, if possible, any security incident which occurred from several sources, such as log files, alerts, error/fault messages, any other tools.
- Establish communication/support center for notifying IR Team.
- Documentation in detail of who, what, where, why and how incident responders must act.
- Making threat detection and prevention capabilities.

1.1.1.3 Containment

In order to limit the caused damage and prevent any spreading while keep untouched any evidence needed there are a few measures that an organization should take. These measures could be:

- A short-term limitation of the damage before gets worse, by isolating specific network spaces for example, preventing the propagation.
- Taking a forensic image of the affected hard disks before wipe and rewrite on them. Thus, there will be evidence for further investigation or accusing the intruder in court.
- A long-term limitation of the damage by addressing the root cause, fixing any back doors, and deleting any accounts or authentication mechanism used by attacker.

1.1.1.4 Eradication

In this step any identified malwares or malicious artifacts are removed, and the affected systems are restored back to normal. An eradication process may include:

- A complete wipe and re-image of all affected hard disks in order to find out if all malicious content has gone successfully.
- Making sure that the root cause is not able to damage/hack any system in the future.
- Making sure that any system is up to date, any software is upgraded and there is no unnecessary service running on them.
- Using anti-malware and anti-virus software for detecting any remaining malicious packages.

1.1.1.5 Recovery

In this step is ensured that all systems are cleaned and fully operational again. In recovery process:

- Based on IR Team findings and prompts will be decided the exact date and time of system services restoring.
- After restoring, tests and verification is needed to ensure that everything works as it should.
- Observe any abnormal behaviors by monitoring for a while after the incident.
- Applying anything could be helping to prevent the same attack.

1.1.1.6 Lessons Learned

After every security incident, the IR Team should gather all the important information acquired due to handling and the whole process, for extracting any lessons that could be used in future incident response activity. So, this step could include:

- Document a full picture of the incident until it's end.
- Publish a detailed report answering once again who, what, where, why, and how.
- Notice what didn't go as good as it could or gone bad.
- Share and discuss learned lessons and findings with others working in the same field to immediately implement preventing measures.

1.1.1.7 IR Handling Report

A well-presented IR handling report has to include:

- The time and date when the incident was first detected.
- How it detected and by whom.
- The extend of the impact.

- What took to contain and eradicate it.
- Acts during recovery.
- The best practices used by IR Team. What went as it should.
- What requires improvement.

1.1.2. Incident Response Team

1.1.2.1 IR Team Model

An IR Team could take one of the following three models, based on NIST framework (Cynet, 2021):

- 1. Central, where the whole organization's incidents are managed from a centralized body.
- 2. Distributed, where there are several teams coordinate each other if needed. Each team is responsible for a specific part of the IT infrastructure, physical location, or department.
- 3. Coordinated, where there is command or knowledge center consists of a central team, which monitors systems and sound the alarm for distributed teams as required, providing the necessary assistance.

There is no restriction about the general form of IR Teams, but based on their tasks they are usually divided in three main types:

- A team which is responsible for preventing, detecting, and responding to incident response cyber security events or incidents, known as Computer Security Incident Response Team (CSIRT).
- A team with similar task as the previous one but with prioritizing the development on threat intelligence and best practices based on security responses. This team focuses on partnerships with government, law enforcement, academia, and industry and it is known as Computer Emergency Response Team (CERT).
- A team which is responsible for monitoring and directing incident response and defending systems. Its tasks include controls configuration and overseeing general operations. This team is well known as Security Operations Center (SOC) and in general includes a CSIRT or CERT teams with a broader scope of cyber security.

1.1.2.2 Members of the IR Team

To achieve the effectiveness of an IR Team, must include a variety of professions and expertise. As it is true, there are some prerequisites to bear in mind while building an IR Team, such as:

- Availability. There is a need for 24/7 assistance and critical incident handling.
- If human resources are limited a virtual or on-call members may be required for dealing with problems when they occur, no matter the time or the place.

- A senior-level executive, like a chief information security officer (CISO) is a good asset in a team. This is the person who can communicate the importance of the incident to high level personnel and also find the required funding when needed.
- Technical diversity between the members to more easily identify and handle threats and find solutions.
- All persons should be moral and have a team spirit in order to create strong relationships which will lead to great collaboration and communication through the stressful situations that definitely will arise.

1.1.2.3 IR Team Preparing

No matter how good prepared the members of a team as individuals are, always should train together. Some wise principles to train an IR Team are: i). The supplementation of cyber security management tools to gain a more secure profile of what an incident is. For instance, the members of the team could monitor carefully to spot local network (intranet) traffic's misuse or abuse. Not only find anomalies or suspicious acts in accesses, but could also watch for abnormal consumptions, like when there is a dropdown in performance due to big demands for exporting data. In these situations, the team could note what a normal user/system behavior is look like and what is not. ii). The centralization of all members efforts when something occurs. In an organization there are so many different systems and subsystems with different software and logs files. So, in order to be always aware of what is happening there must be a central system of parsing, monitoring and alerting when needed. A way to achieve that is by implementing a system information and event management (SIEM). In addition, there are also security orchestration, automation, and response (SOAR) solutions, making easier to get alerted when something goes wrong and automate protective procedures, standardizing responding eco-systems. iii). Learn to act base investigating findings and not by gut feelings. All members must be aware of when you underestimate an attack and you think that you got everything write because of your experience is likely to miss something that will cause your organization's revenue, brand fame or worse. So, is better to make an accurate conclusion during investigations, than based on faulty assumptions. Once an IR Team gets a clear view of the whole incident affection, could response effectively and efficiently remediate the damage.

1.1.2.4 IR Team Roles

As a good point to start is to divide IR Team into following roles:

- A team leader, who will be responsible for coordinating activities and reporting to upper-level management.
- A communications guy who has the responsibility for managing communications not only throughout the team and organization, but also throughout stakeholders, customers, and public authorities so that everyone is properly informed about any incidents.
- An investigation's leader, who will perform as a first investigation commander, giving guidance in analysis and evaluating the primary results.

- More than one member, if possible, who perform as analysts and researchers. They are about to provide threat intelligence and context for an incident. These members usually conduct the first responders' activities.
- Lastly, there must be a legal representant, who acts like a lawyer and make that every act is compliant with law enforcement, and standards of integrity for forensic evidence.

1.2. Android Forensics

To cut long story short, digital forensics is a science or a scientific procedure that like common forensics looking for evidence in a digital world. A recovered deleted message from a mobile phone, is like a fingerprint from the burglar on the banker's desk. Like in all the forensics, when something happened and there is a need for digital investigation, the "crime scene" must remain intact, in order to keep evidence unaltered. In this thesis we would try to clarify how the interfering with digital evidence, and to be more specific android-based devices, alter the integrity of data and the course of investigations (Gogolin, 2021). At the next chapters there will be focus on android forensics and the DFIR including such devices, but before that we must define the main terms that will be used.

1.2.1. Data acquisition

In digital forensics data acquisition, is the process of copying the data without tampering and make them available for presenting, examining, analyzing, and storing purposes. The data are located or stored to be more accurate, in storages like hard disks, memories (i.e., RAM), usb, cd-rom, on motherboard memory chips (mobile cell phones), etc.

1.2.2. A forensic image

In order to acquire the data from a storage space of a device a clone must be created, and thus not modifying them. A forensic image is just that, an unaltered copy of electronic data which are stored in a device. The proper way to take a forensic image is by using forensic tools or/and write-block hardware or software. There are multiple methods of data imaging, but not each one includes every single bit of data from the device they are extracted from. The choice of the method or type of imagining is up to device unique specifications, encryption, and security restrictions. The source storage capacity is also a factor to keep in mind, because if there is no need of keeping everything inside it, the best to do is to make a selective acquiring. Once the image is taken, the forensics examinations are good to go without the fear of tampering and alter the physical device's stored data. Thus, we also gain the integrity of acquired data.

1.2.2.1 Physical image (full file system)

A physical image is a full clone of a data storage source. With term "clone" it means a bitby-bit copy including all partitions, unallocated and unused space of a storage. As always is written, a physical image includes all the source's zeros and ones (Vasilaras A., 2016). This type of imaging gives access throughout the examination process by forensic tools in the deleted and occasionally formatted files. A physical image could be a raw image file (i.e. .dd) or a compressed one (i.e. .E01) without all the zeros (no stored data) of the storage source. Not all the image's formats are compatible with any forensic tool.

1.2.2.2 Logical image (partial logical or partial file system)

In contrast with physical, a logical image includes only the data that is visible to the file system. Is a sound forensic method to acquire data from a storage source, but there are missing much data fragments. For instance, if a mobile application has been deleted by the user, the examination couldn't be able to show that the app it was installed in first place (Tahiri S., 2016).

1.2.2.3 Selective or Targeted image

In addition to above mentioned methods, there is one more costless in terms of resources and time and quicker most of the times. This method requires to know exactly what you want to extract from the storage source for examining and analyzing. Needless to say, that this method is like logical and so no unused or unallocated space or file fragments is feasible to be restored (Precise Law, 2017).

1.2.3. Partition

Partitions are logical divisions of a storage memory (i.e., hard disk) which is being recognized by the operating system as a separate unit (Rahmanpour, 2019). When partitioning is applied, there is a more convenience way to store and organize files and also works as a protection for sensitive data. In an android device for example, there are partitions that user data are stored and partitions that user can tamper with, in which sensitive data for the operating device are included. Hence, no matter how user interact, the device can always be formatted and restored in order to operate without problems.

1.2.4. Android File System

State of the art android devices tend to have larger internal storage than their ancestors did, and their operating systems use their internal storage as a hard drive (i.e., sda) and not as an SD card (i.e., mmcblk0). The table below depicts the structure of the partition table in different android versions.

🔐 X-Ways Investigator*: GT-i5500	/sbin # cat /proc/partitions major minor #blocks name	l lucye:/#cat proc/partitions major minor #blocks name	a50:/ # cat /proc/partitions major minor #blocks name
File Edit Search View To Case Root (2.135) blk0_bm10_c (1) blk12_st112 (988) app (184) bin (165) case (146) ec (60) fonts (9) framework (43) eff (6) ec (146) eff (6) eff (72) eff (73) eff (74)	179 0 30539776 mmcblk0 179 1 12288 mmcblk0p1 179 2 8192 mmcblk0p2 179 4 453632 mmcblk0p4 179 6 10240 mmcblk0p6 179 6 10240 mmcblk0p6 179 7 5120 mmcblk0p5 179 9 29375488 mmcblk0p0 179 32 2048 mmcblk0p0t1 179 16 2048 mmcblk0bot0 /sbin #	2541 0 967712 1:2:seff 2581 16 37768 56712 1:2:seff 8 16 37768 56712 1:2:seff 8 18 3072 sdb1 8 18 3072 sdb1 8 19 4496 5633 8 20 2444 sdb5 8 21 1224 sdb1 8 14 1522 sda1 8 5 61444 sda5 8 6 37768 sda13 8 16 5722 sda14 8 12 sda13 37768 sda14 8 12 sda13	1 0 8192 ram0 1 1 8192 ram1 1 2 8192 ram3 1 4 8192 ram3 1 4 8192 ram3 1 4 8192 ram3 1 4 8192 ram3 1 6 8192 ram6 1 7 8192 ram7 1 8 8192 ram7 1 8 8192 ram10 1 1 8192 ram11 1 1 8192 ram11 1 1 8192 ram11 1 1 8192 ram13 1 14 8192 ram14 1 15 8192 ram15 254 0 2097152 ram0 8 0 124944384 sda 8 1 4608 sda1 8 8 192 sda4 8 8 8192 sda4 8 8 8192 sda4 8 8 8192 sda1
Android 2.2.2. Froyo (Samsung GT-i5500) (created by: X-WAYS	Android 5.1.1 Lollipop (Nexus 7) (created by: adb shell	Android 9 Pie (LG G6 [H870])	Android 11 (Samsung A50)
Investigator)	command line view)	(created by: adb shell command line view)	(created by: adb shell command line view)

Table 1. Examples of the evolution of Androids' partition tables.

The fundamental principle that lies in every device is that the largest partition contains the user data. However, several changes in the partition structure of the OS are observed, depending not only on the variations of the android versions, but also on the proprietary vendor's firmware. From forensic view, this is the partition which contains the vast majority of valuable evidence data. Examiners should be aware that only super user (su) has access to all of the filesystem (**Figure 2**).

a50:	/ \$ cd /data/data
a50:	/data/data \$ ls
ls:	.: Permission denied
1 a5	0:/data/data \$

Figure 1. Denial of access on Samsung A50 device's filesystem as simple user.

1.2.5. Android Debug Bridge

Creating a forensic image of an android device is impossible without setting up a connection bridge first. The way to achieve that is by a command-line tool known as ADB or Android Debug Bridge. This communication method is based on Linux kernel and works like command-line shell. In the official android development site, they refer to ADB capabilities as «a variety of device actions, such as installing and debugging apps, and it provides access to a Unix shell that you can use to run a variety of commands on a device» (SalvationDATA, 2020).

1.2.6. Android Boot Loader

Bootloader known also as boot manager or bootstrap loader is a vendor-proprietary image responsible for boot the kernel on an android device. It monitors the device state and is responsible for initializing the T.E.E (Trusted Execution Environment). The main property is to verify the boot and recovery partitions' integrity, before continuing to the execution of kernel. When something is out of order displays the corresponding boot state warnings. When is unlocked it makes it possible to install a custom firmware or/and giving full access privileges. Hence, modifications and more advanced access on the system are allowed (Bair J., 2017). In general bootloader on android devices is locked and is responsible for what programs are loading and starts to run when the device is turned on.

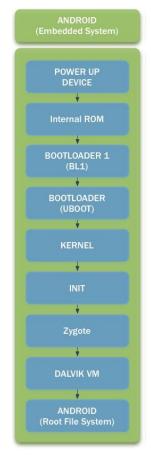


Figure 2. Flow of Boot Sequence on Android Devices (González J., 2012)

1.2.7. Known IR Practices on acquiring Android devices

Every time an android device is examined for a security incident the extraction of data is crucial. For this reason, the acquisition of a device's physical image is important \mathbf{f} , in order to have extensive access to its stored data. But as it is already mentioned not every incident complies with examiner's or incident responder's convenience. As Mahalik et al. (2021) points out, before proceeding in data acquisition there are several steps to walk through in order to prepare the android device for the extraction. The power status, of most android devices, is a crucial factor relating to the procedures that have to be followed.

1.2.7.1 Device is powered off

If the device is powered off an attempt to get a physical acquisition is a sound choice, but if the extraction fails then the device must be turned on and continue as shown in the next two occasions.

1.2.7.2 Device is powered on and unlocked

The steps that are followed on this case are:

- <u>Network connections must be disabled</u> such as Wi-Fi, Hotspots, Bluetooth or mobile data and the device must be set on airplane mode while removing any sim cards.
- <u>Preparation of the device</u> for data extraction by removing any passcodes and enabling the "USB Debugging" and "stay awake" option from the settings and disabling any screen lock features.
- <u>Choosing the acquiring method</u>. First of all, a physical acquisition must be attempted and if this method is not supported, then other extractions (logical, file system, backup) have to be attempted.
- <u>Imaging extra media and sim cards</u>. The examination of any sim cards or/and external memory cards included in source device have also to be added to the evidence collection procedure.

1.2.7.3 Device is on and locked

There are at least two kind of lock protections on android devices. The first one is before android system is fully loaded which is a passphrase with digits and/or letters and the second one is screen lock (pattern, pin code, password, biometrics). If a physical image acquisition can be achieved without tampering device settings, then is highly recommended to proceed with the extraction without any further delay. Network isolation step is still the first thing to do when you want to examine a device. Some forensic tools have their custom bootloaders and will try to use them for gaining access to the device's data, by bypassing any lock, but if this fails then there are not many options left. The searches must be focused on how to get the passcode from the owner or other sources (use known pin codes like birthdays or examine associated media or hard disks). After having completed the above procedure and the device is now unlocked, the described in previous paragraph steps remain the same.

1.3. Structured Query Language (SQL) and SQLite Data

SQLite is lightweight, self-contained database type that fulfills the ACID (Atomicity-Consistency-Isolation-Durability) properties for reliable transactions. The database can be easily integrated into any application without the necessity of any server software (Pawlaszczyk & Hummert, 2021). The efficiency of an android application's operation depends on both the android and SQL programming skills of the developer.

1.3.1. SQLite Database Structure

1.3.1.1 Header

The filename extension of an SQLite database is irrelevant as it can be found as db2, db3, SQLite or none. The proper way to examine the properties of a database is by inspecting the header of a file. The header of an SQLite database is comprised of it's first one hundred (100) bytes divided into several fields. (SQLite, 2021)

□ Name▲								Des	cript	ion			1	т Ту	pe	T :	Size	ΥC	reat	ed		YN
🗆 📄 cdr.db] cdr.db						e	existing					5	sqlite	db	4.	O KB	202	/09/	02d21	:22:32	2021,
🗆 📄 cdr.db-shr	cdr.db-shm							existing					(db-sh	m	32.	0 KB	202	/09/	02d21	:22:32	2021,
🗆 📄 cdr.db-wa	cdr.db-wal								existing					db-wa	al	60.4	4 KB	202	/09/	02d21	:22:32	2021,
🗆 📄 com.goog	com.google.android.datatransport.eve								existing					sqlite	db	4.	0 KB	202	/09/	02d21	:22:32	2021,
🗆 📄 com.goog	com.google.android.datatransport.eve								existing					db-sh	m	32.	0 KB	202	/09/	02d21	:22:32	2021,
🗆 📄 com.goog	com.google.android.datatransport.eve								existing					db-wal 109 KB			2021/09/02d21:22:32 2021,					
🗆 📄 exoplayer.	exoplayer.db							existing					5	sqlite	db	4.	0 KB	202	/09/	02d21	:22:32	2021,
🗆 📄 exoplayer.	exoplayer.db-shm						e	existing					0	db-shm 32.0 KB			0 KB	2021/09/02d21:22:32 2021,				
🗆 📄 exoplayer.	exoplayer.db-wal						e	existing					(db-wa	-wal 64.4 KB			2021/09/02d21:22:32 202			2021,	
🗆 📄 google_ap	google_app_measurement_local.db					e	existing					5	sqlite	db	16.	0 KB	202	/09/	02d21	:22:32	2021,	
🗆 📄 mixpanel	📄 mixpanel						e	existing					5	sqlite	db	71	6 KB	202	/09/	02d21	:22:32	2021,
	viber_data						e	existing					2	sqlite	db	12	0 KB	202	/09/	02d21	:22:32	2021,
	viber_data-journal						e	existing									7 KB	3 2021/09/02d21:22:32 2021,				
viber_mes	viber_messages						e	existing, already viewed					5	sqlite	db				2021/09/02d21:22:32 2021,			
	🛛 📄 viber_messages-journal						e	existing						381 KB 2021/09/02d21:22:32								
	viber_prefs							existing					5					B 2021/09/02d21:22:32 2021,				
viber_pref	s-jou	rnal					e	existing							52.	6 KB	202	/09/	02d21	:22:32	2021,	
Se Volume		ື Fil	e		Prev	iew	5	Deta	ils		Galle	ry		Caler	ndar		Lege	end	Ĵ		Syr	nc
Offset	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	-		AN	SI A	SCII	1
00000000	53	51	4C	69	74	65	20	66	6F	72	6D	61	74	20	33	00	SQ	Lit	e f	orma	t 3	
00000016	10	00	01	01	00	40	20	20	00	00	01	56	00	00	00	62			6	v	b	
00000032	00	00	00	00	00	00	00	00	00	00	00	48	00	00	00	04				Н		
00000048	00	00	00	00	00	00	00	49	00	00	00	01	00	00	00	CF			I		Ï	
00000064	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00						
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	56					v	
00000096	00	2E	3B	Fl	05	00	00	00	05	0F	E7	00	00	00	00	5F		;ñ		ç	_	
00000112	0F	FB	0F	F6	0F	Fl	0F	EC	0F	E7	00	00	00	00	00	00	û	ö	ñì	ç		

Figure 3. Viber-messages SQLite database header parsed via X-Ways forensics Software in hex mode.

Nowadays, several forensic software products provide a way to interpreter those one hundred (100) header bytes by parsing them.

e Info													
e Into													
d & / 6 4 III													
data bases		Filter:											
viber_messages viber_messages-journal	File info Header Fields SQL-Schema												
	Offset	Property	Value										
	0	The header string	SQLite format 3 (0x53514c69746520666f726d6174203300)										
	16	The database page size in bytes	4096										
	18	File format write version	1										
	19	File format read version	1										
	20	Unused reserved space at the end of each page	0										
	21	Maximum embedded payload fraction. Must be 64.	64										
	22	Minimum embedded payload fraction. Must be 32.	32										
	23	Leaf payload fraction. Must be 32.	32										
	24	File change counter.	63										
	28	Size of the database file in pages.	95										
	32	Page number of the first freelist trunk page.	0										
	36	Total number of freelist pages.	0										
	40	The schema cookie.	72										
	44	The schema format number. Supported schema formats are 1, 2, 3,	4										
	48	Default page cache size.	0										
	52	The page number of the largest root b-tree page when in auto-vacu	73 (true)										
	56	The database text encoding.	UTF-8										
	60	The "user version"	207										
	64	True (non-zero) for incremental-vacuum mode. False (zero) otherwi	0 (false)										
	92	The version-valid-for number.	63										
	96	SOLITE VERSION NUMBER	3030001										

Figure 4. Viber-messages SQLite database header parsed via FQLite.

In terms of forensics, some interesting points in the header, are the number of freelist pages (bytes 36 to 40), the auto vacuum feature (bytes 52 to 56) and the incremental vacuuming (bytes 64 to 68). The above information can determine whether deleted records can be recovered. (Nemetz et al., 2018)

1.3.1.2 SQLite Pragma Statements

As it is defined by the official site of SQLite (2021), PRAGMA statements are SQL extensions for SQLite and are used for controlling various environmental variables and state flags within the SQLite database. There is a vital difference between PRAGMA statements and SQLite commands although they are using the same interface. There are specific PRAGMA statements that can be removed or added in an updated future release of an SQLite database, with no guarantee of backwards compatibility. Below is depicted some of the "Pragmas" that can be viewed or edited by a database tool.

Database Structure	Browse Data	Edit Pragmas
Auto Vacuum 🙆	None	\sim
Automatic Index 😣		
Case Sensitive Like 🔮		
Checkpoint Full FSYNC	0	
Foreign Keys 😡		
Full FSYNC 😣		
Ignore Check Constrai	ints 🙆 🗌	
Journal Mode 🙆		\sim
Journal Size Limit 😣	0	**************************************
Locking Mode 😣		\sim
Max Page Count 🙆	0	
Page Size 😣		\sim
Recursive Triggers 🙆		
Secure Delete 🥹		
Synchronous 🙆	Off	\sim
Temp Store 😟	Default	t ~
User Version 😡	0	- <u>+</u>
WAL Auto Checkpoint	0	*

Figure 5. SQLite Pragma Statements from DB Browser for SQLite interface.

The most interesting parameters, as shown at the above picture, for database forensics, from DFIR spectre are the following:

- <u>Auto Vacuum</u>: this parameter indicates how database freelist pages are being handled when a data deletion commit occurs. When it is enabled (registered value "Full" or "1"), the freelist pages are moved to the end of the database file and are removed after the end of every transaction commitment. On the other hand, if it has a value of "None" or "0" that means the auto-vacuum PRAGMA statement is disabled and no matter what insertions or deletions are committed, the database's file size is retained, as unused data are added to freelist pages and reused for subsequent inserts.
- <u>Journal Mode</u>: this parameter defines how the journal file interacts with the "main" database every time a transaction is committed. If value of "DELETE" is selected (default value), the rollback journal is invalidated at the completion of each transaction. The other used method for journaling mode is "WAL" that uses the write-ahead log instead of the rollback journal.
- <u>Journal Size Limit</u>: this parameter may be used for size limitation of rollback-journal and WAL files. By default, the size parameter takes the value of "-1", for unlimited journal size, but as it will be proved during in the experiments held in this thesis the android application's database have a size limitation of their rollback journal files.
- <u>Secure Delete</u>: this is a boolean parameter and holds values "on" or "off". When the value is 'on' SQLite overwrites deleted content with zeros thus wiping the deleted data. "Secure_delete" is quite a CPU and disk intensive process with great impact in battery life. For this reason, the vast majority of applications on android devices has the "secure_delete" option disabled.

1.3.1.3 B-Tree Pages

The B-tree pages store the active data from the tables and indexes divided in four main areas: the page header, the cell pointer array, the unallocated space, and the cell content area. The latter is comprised by the active records, the free blocks, and the fragmented bytes. In some cases, it is possible to recover data from the unallocated space and the freeblocks, but it is nearly impossible from the fragmented bytes, as their size is too small (less than 4 bytes).

1.3.1.4 Freelist Pages

Freelist pages are the ones, which are not currently used by a database. Their records were deleted, and they are flagged for re-use at a later time. By default, SQLite does not wipe their contents and so there is a possibility to recover deleted records. This is not possible, if the pragma statement of "secure_delete" is enabled meaning that SQLite overwrites deleted content with zeros (SQLite, 2021). Other page types are the overflow, the pointer map and the lockbyte pages but since, they have not any forensic interest will not be further examined.

1.3.1.5 Atomic Commit in SQLite

As it is already mentioned, SQLite is ACID compliant. For this reason, its atomic commit feature is very important. Atomic commit is achieved when all database changes occur within a

single transaction (SQLite, 2021). This is implemented through the rollback journal (legacy) or, since SQLite 3.7, with the Write Ahead Log (WAL). These two methods utilize very different approaches for the atomic commit and rollback features of a database. The method is used can be determined by bytes 18 and 19 of the file header. If the value, in the header, is set to "01 01" the rollback journal is used and if the value is set to "02 02" then the WAL method is used.

1.3.1.6 Rollback Journal

SQL .journal file holds a copy of the page prior to committing a transaction and if the transaction is committed, it is invalidated, but if the transaction fails, the page is copied back to the main database. The reasons why there may be a failed transaction vary. Some examples are file overwriting, file locking issues or failure in synchronization with the storage media. There are five Rollback Journal modes: Delete, Truncate, Persist, Memory and Off. The Delete mode is the default one. On the contrary, Memory and Off modes are rarely encountered. The selected mode cannot be determined by the file header but can be found into PRAGMA statements (SQLite, 2021). The Rollback Journal could have an invalid state, which is potentially a previous version of the state of the main database or a valid/hot one which indicates that the last transaction failed leaving the main database unmodified. There are cases, especially after a fresh installation of an application in mobile applications in which the journal/wal file and the main database is not present in the device's folder structure.

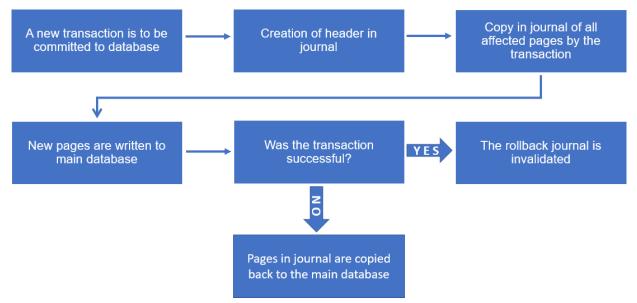


Figure 6. SQL journal transaction process flow.

1.3.1.7 Write Ahead Log (WAL) and Shared Memory (shm)

The WAL process, which was implemented after SQLite 3.7, takes a different approach by leaving the main database intact and committing the changes to a separate file. The pages of the main database are left unmodified until a WAL checkpoint is reached. Usually this happens after one thousand (1.000) commits, when the application forces a commit back to the main database. The examination of the WAL file provides information about the most recent state of the database. For instance, recently deleted data cannot be found in the main database but might be located in the database's WAL file.

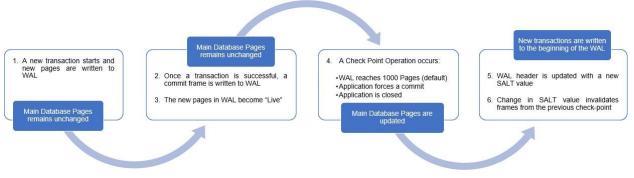
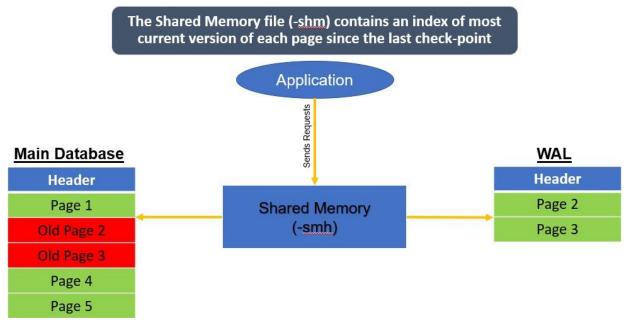
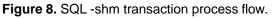


Figure 7. SQL -wal transaction process flow.

The Shared Memory file contains an index of the most current version of each page since the last checkpoint. Thus, an application can decide wherefrom the most recent data exists in order to pull them off.





1.4. Forensics tools solutions

1.4.1. Android Mobile Devices Acquisition and Examination

There are a lot of information about forensic tools supporting android forensics. In fact there are plenty of sources that describe them and compare their capabilities and features (Easttom, C., 2021, Easttom, C., 2021, Raji et al., 2018, Vasilaras A., 2016). In the section below, the major capabilities of commercial and non-commercial well-known software and tools used for android forensic purposes are outlined. Some of the mentioned tools are used both for extracting and analyzing data, while others are used only for examinations. Most of the tools' user interfaces will be included in the next chapter.

- Autopsy. A free of cost software suite developed by Brian Carrier and his team. There is an Android Analyzer module for android devices which works with physical images and file system dumps. In newer versions (4.18 and above) the aLEAPP parser has been implemented in the Android Analyzer module. It supports images created from other tools such as Cellebrite and MSAB.
- aLEAPP. A noncommercial github tool written in Python by a law enforcement officer Alexis Brignoni. This tool is a log events and protobul parser and works also with physical or file system dumps files (.zip, .tar) and creates a report including all valuable data.
- Andriller. A freeware utility including a collection of tools for android based devices, which can perform both forensic analysis and acquisition.
- Cellebrite UFED. A commercial tool that includes UFED touch or UFED 4PC and UFED physical Analyzer. The first two are utilized for extraction purposes providing a user-friendly interface. The last one is utilized for analyzing the device's extraction.
- Magnet Forensics. Magnet provides both an acquisition (Magnet Acquire) and an examination and analysis tool (Magnet Axiom). Also, it provides a very convenient report for viewing the evidence findings. It also supports various type of images including the ones created from Cellebrite.
- Oxygen forensics. Oxygen Extractor supports a variety of different chipsets as well as the abovementioned tools. The analyzing tool (Oxygen Detective) is capable of building connections and correlations among multiple acquired evidence devices as well as visualizing them in a user-friendly manner. It also supports images created from other tools such as Cellebrite and MSAB.
- Belkasoft. Belkasoft X is another useful tool combining extractions from android devices and supports their examination and analysis.
- MSAB. XRY digital data is able to extract data from mobile devices (both physical and logical) and XAMN is used to analyze and examine them.
- MOBILedit. Through MOBILedit Forensic Express tool, android physical and/or logical extractions as well as analysis of the extracted data are feasible. It supports Cellebrite and Oxygen exported images.
- Access Data. Since Forensic ToolKit (FTK) v7.4.104 there is an addon for mobile data analysis and chat application parsing tool It also provides a QView viewer for more convenient visualization of examined data. It supports Cellebrite and MSAB exports.
- Hancom. MD-Series tools provide several different solutions for acquiring and examining android devices. Among other extraction methods, JTag (via MD-box) is included.
- Paraben E3 Forensic Platform. A platform which supports all kind of acquisitions and forensic examinations.
- X-Ways. Since the android filesystem is similar/same to UNIX/LINUX, X-ways can provide an accurate and fast hex interpreter parser and file system viewer.

Most of the referred tools have the capabilities of bypassing, removing or brute forcing the pin code, pattern, or password security measures of android devices. In addition of the previous mentioned features, the tool should include a decryption method of the device's data. More and more applications implement cypher methods in order to encrypt their databases but not all of the tools are capable of decrypting and parsing every single one of them. Bear in mind that some versions of the abovementioned commercial tools are only available for law enforcement agencies and not for enterprises. Specifically, the law enforcement versions include extra features and artifacts for both extraction and analysis. Be aware that there is not a tool that supports every

single android device or chipset. For this reason, the examiners should wisely select the tools that will help them in their work.

1.4.2. SQLite Databases Examination, Recovery and Analysis

The forensic analysis of an SQLite Database's live records is relatively easy compared to deleted ones. The SQLite Carving for deleted records is a difficult and tedious task for various reasons. Firstly, the number of forensically interesting applications, which use SQLite databases, and the speed these are being updated is too fast for any tool to keep up with. Secondly, the pragma statements (e.g., secure_delete, autovacuum etc.), which are implemented by the application's developers and the way unallocated space and free blocks (containing random fragments of data), are located in these pages, makes the carving of any useful information is challenging to say at least. Some options for recovering and analyzing data from SQLite Databases are:

- aLEAPP (Free). The integrated python scripts which are implemented in the tool can create a useful report of android artifacts, including records from applications' databases. In addition, the SQL queries inside the scripts could be used separately for browsing SQLite databases in other software such as DB Browser for SQLite. One disadvantage is that it cannot retrieve deleted records.
- DB Browser for SQLite (Free). One of the most known and useful tools is DB Browser for SQLite. It does not have the automated ability to parse journal or deleted records. It is not considered as forensic software as the user can modify the examined database's data.
- SQlite Expert (Free and Propietary). SQlite Expert can visualize and modify BLOB fields and present multiple databases in one tree. It is also not considered as forensic software as the user can modify examined database's data.
- Oxygen forensic SQlite Viewer (Propietary). Part of Oxygen forensic Suite which can recover deleted records and parse journal and wal files.
- Magnet Axiom Forensics (Propietary). Magnet Axiom suite integrates an SQLite Viewer which has the automated ability to parse and combine the main database with its corresponding WAL/Journal file.
- UFED Physical Analyzer (Proprietary). Ufed includes the most popular mobile application's database schemas and via App Genie and SQlite Wizard features, can potentially recover/parse deleted records.
- Autopsy Suite (Free). Autopsy, is the best known free forensic suite and with the usage of python plugins can parse SQLite Databases and their deleted records.
- Paraben E3 (Free and Proprietary). Paraben E3 free edition can parse SQLite Databases but not Journal / Wal files.
- Belkasoft X (Proprietary). Belkasoft X can parse records from SQLite Database and their corresponding Journal / wal files.
- X-ways Forensics (Proprietary). X-ways, by creating virtual files, can analyze SQLite databases.
- acquire Forensics (Shareware). Acquire SQLite Forensic Explorer has the ability to parse and recover records from an SQLite database (including records residing in journal/wal file) even if they are corrupted or deleted.
- FQlite (Free). FQLite is a tool to find and restore deleted records in SQLite databases. It therefore examines the database for entries marked as deleted. Those entries can be recovered and displayed.

- Andriller (Free). Has a variety of tools for parsing and decrypting databases but it cannot retrieve deleted records.
- KS DB Merge Tools for SQLite (Free & Proprietary). A tool for comparing the same database in different states based on the schema, the tables, and the data inside them. The tool uses a GUI version of the "sqldiff" executable.
- Forensic Toolkit for SQLite. This is a toolkit which includes three separate tools: "The Forensic Browser", "Forensic Recovery" and "SQLite Forensic Explorer" for SQLite. Each of these tools has different usage. For example, "Forensic Browser" has the ability to reconstruct a database including the records from its wal/journal file.

Some other known command line tools which could be used are undark (Nemetz et al., 2018) and bring2lite (Meng & Baier, 2019).

Chapter 2.

Preparation and Examination of an Android Device

In this chapter, the preparation required for gaining admin privileges (root) and examining of two different Android OS devices is presented. An LG G6 (H870) with Android 9 and a Samsung A50 (A505FN/DS) device with Android 11 were utilized and granted root access by unlocking bootloader, installing custom recovery menu and software. The important folders and files of the filesystem are referenced. The commands (CLI) which were used to navigate in the filesystem and extract the desired data are also analyzed. In addition, user interfaces of several software and tools for extracting and examining data are presented.

2.1. Getting Root access

In order to gain a root access, the devices' bootloaders have to be unlocked and a custom recovery image known as Team Win Recovery Project (TWRP) has to be installed. This has as a result, third-party firmware can be installed, and full system backups can be created. The unlocking of the device's bootloader requires the enabling of USB-Debugging and OEM unlocking options. The above-mentioned settings can be found under Developer Options. After that, the device can be put in bootloader mode via Command Prompt or Terminal command with the command:

adb reboot bootloader

After this step the bootloader unlocking procedure varies depending on the devicevendor's requirements. For instance, LG requires to send the unique device id and IMEI through their developer portal in order to receive back a binary unlock file (unlock.bin), whereas Xiaomi uses the Mi Unlock application for this purpose (Xiaomi MIUI, 2021). In any case the userdata partition is wiped when the bootloader unlocks. After having unlocked the devices' bootloader, the "Magisk" solution (Magisk Manager, 2021) was chosen for getting root access. As the official developer, John Wu, refers «Magisk is a systemless rooting», which means that the core code of the mobile's operating system is not altered.

2.1.1. Getting root access on LG G6

The examined LG G6 device had the following specifications:

- Android Kernel version is 3.18.120.
- Android 9 with security patch dated in May of 2019 (Latest available).
- The official rom firmware which was installed is the latest available (v30b-EUR-XX).

The "3.2.3-Nebula-Alpha_20180813 TWRP" from XDA-developers forum (zefie, 2017) was used as a recovery image, and the Magisk version 23.0 was installed for granting root access

privileges to the system. For further convenience, an extra Magisk module was installed, named "Busybox for android NDK 1.33.1", which offers additional Unix commands (i.e., netcat).

2.1.2. Getting root access on Samsung A50

The examined Samsung device had the following specifications:

- Android Kernel version is 4.14.113.
- Android 11 with security patch dated in July of 2021.
- The official rom firmware which was installed has the built number "a505fxxs9cuf1".

The "3.5.0_9_0 (UNOFFICIAL recovery)"¹ from XDA-Developers forum (redymedan, 2021) was used as a recovery image and the unofficial Magisk version 22.1 was installed in order to get administrating (root) permissions. For further convenience, the Magisk module, "Busybox for android NDK 1.33.1" was also installed. In addition, on the custom recovery image (twrp), the "multidisabler-samsung 3.1" from Macdonald, I. (2020) addon was installed, in order to disable the auto-reflash of stock recovery, which is implemented from Samsung as a countermeasure, against the unauthorized installation of third-party firmware.

2.1.3. Wiping the device

By enabling the parameter "Use rm -rf instead of formatting" in the Settings menu of TWRP, a force deletion of every file/folder in the userdata directory was succeeded.

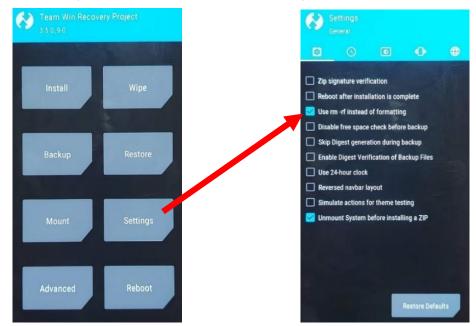


Figure 9. TWRP settings parameter "Use rm -rf instead of formatting".

¹ This is an unofficial TWRP. At the time there is not an official TWRP supporting android 11 for Samsung A50.

2.2. Extracting the data manually

For the acquisition of the data with ADB commands (Tamma et el., 2020), both Windows and Linux environment systems were utilized. In the Windows environment (Microsoft Windows 10) the SDK platform tools v. 31.03 were utilized. Furthermore, the "ncat" binary was used instead of the netcat executable that is detected as a malicious one from the majority of antiviruses. Additionally, all the executables were added to the environmental path of the operating system for convenience. In the Linux environment (ubuntu 20.04) the same version of the SDK platform tools was installed by copying them to the usr/bin folder instead of installing the obsolete ones from the repository of the Ubuntu operating system. The following commands were used:

> Directing commands to the only attached USB device (mobile phone) ("-d") or ("-s")²

\$ adb -d shell

Getting root access (super user)

\$ su

Previewing partition table³

cat /proc/partitions

-				
			/proc/parti	
	major	minor #1	olocks nam	ne
	254	Θ	967212	zramΘ
	254	1	967212	zram1
	8	Θ	30912512	sda
	8	1	49152	sda1
	8	2	49152	sda2
	8	3	32768	sda3
	8	4	10240	sda4
	8	5	6144	sda5
	8	6	32768	sda6
	8	7	67584	sda7
	8	8	512	sda8
	8	9	512	sda9
	8	10	512	sda10
	8	11	512	sda11
	8	12	512	sda12
	8	13	32768	sda13
	8	14	5726208	sda14
	8	15	524288	sda15
	259	Θ	352256	sda16
	259	1	24018944	sda17
	259	2	7636	sda18

Figure 10. Partition table view of LG G6 device via ADB command.

Previewing the symlinks of the partitions table by navigating to the path «dev/block/platform/soc/624000.ufshc/by-name» and listing all the files.

cd / dev/block/platform/soc/624000.ufshc/by-name

² if multible devices were connected.

³ Notice: the largest partitions usually is the whole block.

ls -l

1	27 Lucve:	/de	v/blo	ock/p]	at-	Form/soc/624	1000.u-	fshc/by-name # ls -l
t	otal 0							
ι	TWXFWXFWX		root	root	16	2017-01-01	02:06	OP -> /dev/block/sda16
1	.rwxrwxrwx	1	root	root	15	2017-01-01	02:06	aboot -> /dev/block/sde6
1	rwxrwxrwx	1	root	root	15	2017-01-01	02:06	abootbak -> /dev/block/sde7
1	FWXFWXFWX	÷	FOOT	root	16	2017-01-01	02:00	apdp -> /dev/block/sde26 boot -> /dev/block/sde1
								cache -> /dev/block/sdel
								cdt -> /dev/block/sdd3
l	.rwxrwxrwx		root	root	16	2017-01-01	02:06	cmnlib -> /dev/block/sde22
ι	.rwxrwxrwx		root	root	16	2017-01-01	02:06	cmnlib64 -> /dev/block/sde24 cmnlib64bak -> /dev/block/sde25
1	.rwxrwxrwx	1	root	root	16	2017-01-01	02:06	cmnlib64bak -> /dev/block/sde25
	.FWXFWXFWX	÷	root	root	16	2017-01-01	02:06	cmnlibbak -> /dev/block/sde23 ddr -> /dev/block/sdd1
								devcfg -> /dev/block/sdel6
								devcfgbak -> /dev/block/sde17
								devinfo -> /dev/block/sdb6
1	.rwxrwxrwx		root	root	15	2017-01-01	02:06	dip -> /dev/block/sdb5
1	TWXTWXTWX	1	root	root	16	2017-01-01	02:06	dpo -> /dev/block/sde28 drm -> /dev/block/sda4
1	TWXFWXFWX	1	root	root	15	2017-01-01	02:00	drm -> /dev/block/sda4 eksst -> /dev/block/sda9
1	TWXTWXTWX	1	root	root	15	2017-01-01	02:00	encrypt -> /dev/block/sda9
								factory -> /dev/block/sda7
								fota -> /dev/block/sdb3
l	.rwxrwxrwx		root	root	15	2017-01-01	02:06	fsc -> /dev/block/sdf3
1	.rwxrwxrwx	1	root	root	15	2017-01-01	02:06	fsg -> /dev/block/sdb4 grow -> /dev/block/sda18
	TWXTWXTWX		root	root	16	2017-01-01	02:06	grow -> /dev/block/sda18 grow2 -> /dev/block/sdb7
1	TWXFWXFWX	ī	root	root	15	2017-01-01	02:06	grow3 -> /dev/block/sdc3
ι	TWXTWXTWX		root	root	15	2017-01-01	02:06	grow4 -> /dev/block/sdd4
ι	.rwxrwxrwx		root	root	16	2017-01-01	02:06	grow5 -> /dev/block/sde29
ι	.rwxrwxrwx		root	root	15	2017-01-01	02:06	grow6 -> /dev/block/sdf4
1	.rwxrwxrwx	1	root	root	15	2017-01-01	02:06	grow7 -> /dev/block/sdg2
1	TWXTWXTWX	÷	root	root	16	2017-01-01	02:00	hyp -> /dev/block/sdel2 hypbak -> /dev/block/sdel3
ĩ	TWXTWXTWX	i	root	root	16	2017-01-01	02:06	keymaster -> /dev/block/sde20
ι			root	root	16	2017-01-01	02:06	keymasterbak -> /dev/block/sde21
								keystore -> /dev/block/sdal2
								laf -> /dev/block/sdal
								lafbak -> /dev/block/sda2 misc -> /dev/block/sda6
ĩ	TWXTWXTWX	i	root	root	16	2017-01-01	02:00	modem -> /dev/block/sda0
ί	TWXFWXFWX	ĩ	root	root	15	2017-01-01	02:06	modem -> /dev/block/sde18 modemst1 -> /dev/block/sdf1
ι	TWXTWXTWX	1	root	root	15	2017-01-01	02:06	modemst2 -> /dev/block/sdf2
ι	FWXFWXFWX		root	root	15	2017-01-01	02:06	mpt -> /dev/block/sda3
								msadp -> /dev/block/sde27
								persist -> /dev/block/sdal3 persistent -> /dev/block/sdgl
1	TWXTWXTWX	1	root	root	16	2017-01-01	02:06	pmic -> /dev/block/sdel4
1	.rwxrwxrwx	1	root	root	16	2017-01-01	02:06	pmic -> /dev/block/sde14 pmicbak -> /dev/block/sde15
ι	TWXTWXTWX	1	root	root	15	2017-01-01	02:06	raw_resources -> /dev/block/sde8
								<pre>raw_resourcesbak -> /dev/block/sde9</pre>
								rct -> /dev/block/sdal0
								recovery -> /dev/block/sde2 recoverybak -> /dev/block/sde3
ι	TWXTWXTWX	1	root	root	15	2017-01-01	02:06	reserve -> /dev/block/sdd2
1	TWXTWXTWX	1	root	root	16	2017-01-01	02:06	rpm -> /dev/block/sde10
l	.rwxrwxrwx		root	root	16	2017-01-01	02:06	rpm -> /dev/block/sdel0 rpmbak -> /dev/block/sdell
ι	TWXTWXTWX	1	root	root	16	2017-01-01	02:06	<pre>sec -> /dev/block/sde19</pre>
								sns -> /dev/block/sda5
								<pre>ssd -> /dev/block/sdall system -> /dev/block/sdal4</pre>
								tz -> /dev/block/sda14
1	TWXTWXTWX	1	root	root	15	2017-01-01	02:06	tzbak -> /dev/block/sde5
1	.rwxrwxrwx		root	root	16	2017-01-01	02:06	tzbak -> /dev/block/sde5 userdata -> /dev/block/sda17
l	.rwxrwxrwx	1	root	root	15	2017-01-01	02:06	xbl -> /dev/block/sdb1
1	TWXTWXTWX	1	root	root	15	2017-01-01	02:06	xbl2 -> /dev/block/sdc1 xbl2bak -> /dev/block/sdc2
1	TWXTWXTWX	1	root	root	15	2017-01-01	02:00	xblzbak -> /dev/block/sdc2 xblbak -> /dev/block/sdb2
	- Contraction of the			2000	1.1			

Figure 11. Partition table symlinks view of LG G6 device via ADB command.

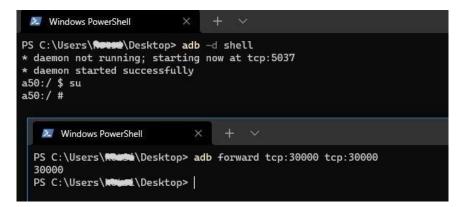
This could be very useful when a targeted acquisition is needed (e.g., userdata partition).

1	lrwxrwxrwx 1	. ro	ot	root	16	2017-01-27	01:00	userdata	->	/dev/block/sda17
---	--------------	------	----	------	----	------------	-------	----------	----	------------------

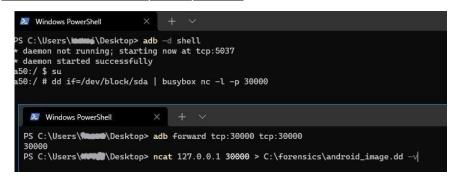
- > Acquiring data. The procedure is divided in three parts:
 - Firstly, in a terminal a bridge must be established between the host computer and the device in order for the acquired data to be copied back to the host:

adb forward tcp:30000⁴

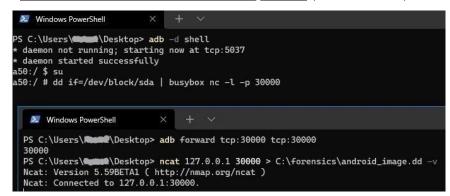
⁴ The usage of a high port number is recommended because windows 10 OS tends to block any connection in lower ones.



- Secondly, in another terminal which is already running inside the shell of the android device the partition that must be acquired can be copied through the desired port to the host computer. By choosing the sda partition in Samsung A50 device, a full physical image acquisition is conducted:
- # dd if=/dev/block/sda | nc -l -p 30000



- Thirdly, the acquired image can be "pushed" to a chosen path in the first terminal via the netcat⁵ executable:
 - ncat 127.0.0.1 30000 > android_image.dd -v (-v for verbose)



On some occasions, when specific folders and files are aimed, a targeted extraction can be applied. A forensic copying could be done, to a SD card⁶, by a combination of "adb pull" and copy (cp) commands. The main concern in this scenario is to preserve the source's timestamps.

⁵ "nc" in Linux or "ncat" in Windows.

⁶ An external SD card was used in order not to ensure the integrity of the internal storage of the examined device.

This can be achieved by adding the "-a" parameter in the adb pull command. For instance, the commands used for the forensic extraction of a specific Viber application's folder were:

 cp -pvR /data/data/com.viber.voip /sdcard0/forensics/ (-p for preserve, -v for verbose, -R for recursively)

or

- cp -R --preserve=timestamps /data/data/com.viber.voip/databases/* sdcard0/ forensics/ (coping only the databases' folder while preserving the timestamps)
- adb pull -a /sdcard0/forensics/com.viber.voip C:\forensics\viber_data_package (-a preserves file timestamp and mode)

The usage of two different commands was chosen because, when trying to execute the command "adb pull -a" directly from the «/data/data» folder it was unsuccessful, since the "adb root" command was not working. This feature is by default disabled on production build devices and in order to enable it the "ro. secure" file (Skulkin et al., 2018) has to be modified.

2.3. Extracting the data via software

PROVIDE FEEDBACK

In contrast to the command line approach (adb), several user-friendly forensic tools can be used for the acquisition of an android device. Some of them are presented below.

> Magnet Acquire.

Magnet ACQUIRE				
			OPTI	ONS
SELECT IMAG	E TYPE			
0	Please select the type of image you want to acquire:			
ANDROID	Full			
<u> </u>	Entire contents of the device More info			
.GE .G-H870	Logical			
GH8705d0b09f6	O Entire contents of user data folders More info			
Privileged access	Guick			
	O Native and 3rd party application data, media, and external shared storage (SD card)	More	info	

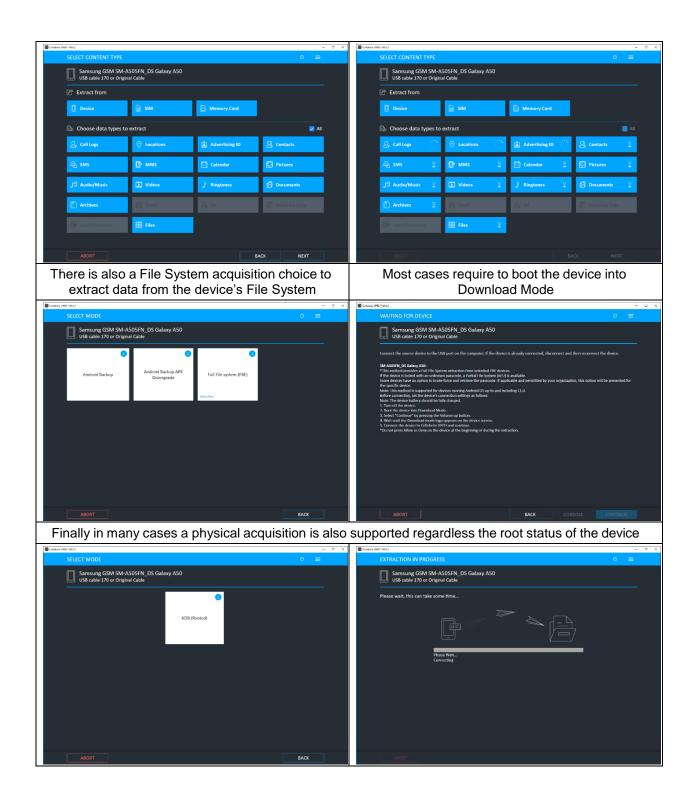
Figure 12. Magnet Acquire user interface while LG G6 is connected.

BACK

NEXT

> UFED 4PC.

Software GUI	When the device is connected and auto detected by the software
Collebrite UFED ○ =	Consumer and a second by Roman Boy R
Mobile device Mobile device UFED camera Drone	Sensorg GOM SM-MODIN, IS Galay AGO
Version 7.49.0.2 ⑤ 7:51:40 πμ 📅 29/6/2021	ABORT BACK CONSOLE BROWSE DEVICES AUTO DCTECT
The options are more or less the same for every android device	An Advanced Logical Acquire is feasible by enabling some parameters under the Developer Options Settings
Select EXTRACTION TYPE © Samung GSM SM-A505FN_DS Galaxy A50 Uss cable 170 or Original cable	WAITING FOR DEVICE C ☑ Samsung GSM SM-A505FN_DS Galaxy A50 U Uso able 170 or original Cable
	Connect the source device to the USH port on the computer. If the device is already connected, discorrect and then reconnect the device.
Advanced Logical File writem File writem Screenshot Chat Capture E	In cost of a gare
ABORT BACK After doing that the device info is re	ABONT BACK CONSOLE CONTINUE
WANTING FOR DEVICE O	CANAGENETING - 0 X DEVICE DETECTED © =
Please Wait Reading Phone Info	<section-header><section-header><section-header><section-header><section-header><section-header><section-header><text><text><text><text></text></text></text></text></section-header></section-header></section-header></section-header></section-header></section-header></section-header>
Next step is choosing the a	



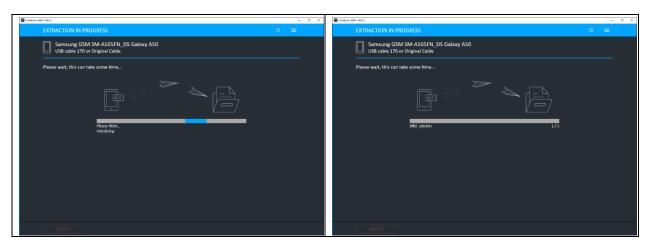
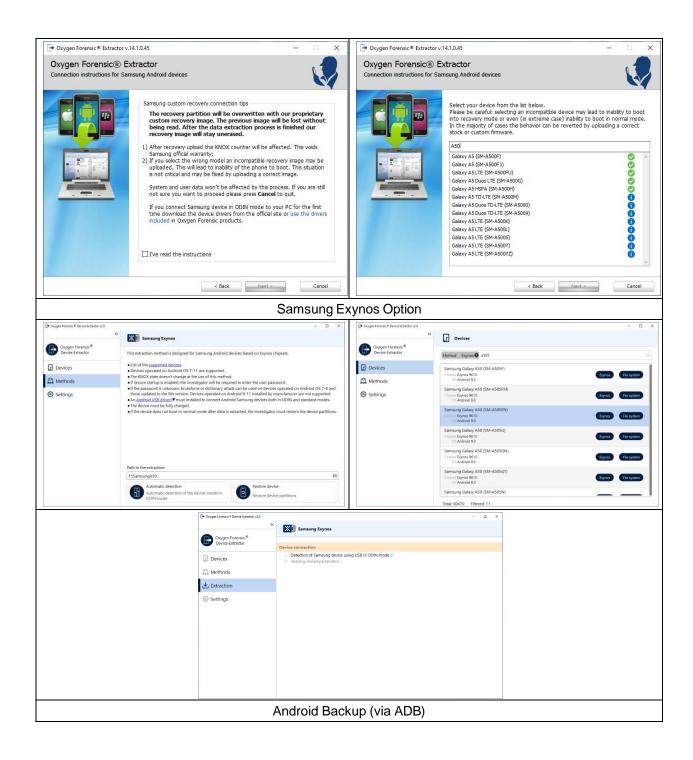


Table 2. Cellebrite UFED 4PC user interface. Acquisition methods while the Samsung A50 device is connected.

> OXYGEN

	Software GUI
Oxygen Forensic® Extractor v.14.1.0.45 Oxygen Forensic® Extractor Select a type of device backup or phys	tor 🚬
Android extracti Android physical (via ADB) Android manual OsyAgent MTK Android LG Android) Android backup (via ADB) Android OxyAgent extraction t extraction Android OxyAgent over WiFI extraction Android file system d Spreadhum Android Samsung Android Samsung Exynos omm EDL Huawel Kirin Sony MTK Huawel Qualcomm APK Downgrade ns
	Cancel
Samsung Android Option	Selection of Supported Samsung Devices in Samsung Android Option (searching model "A50")



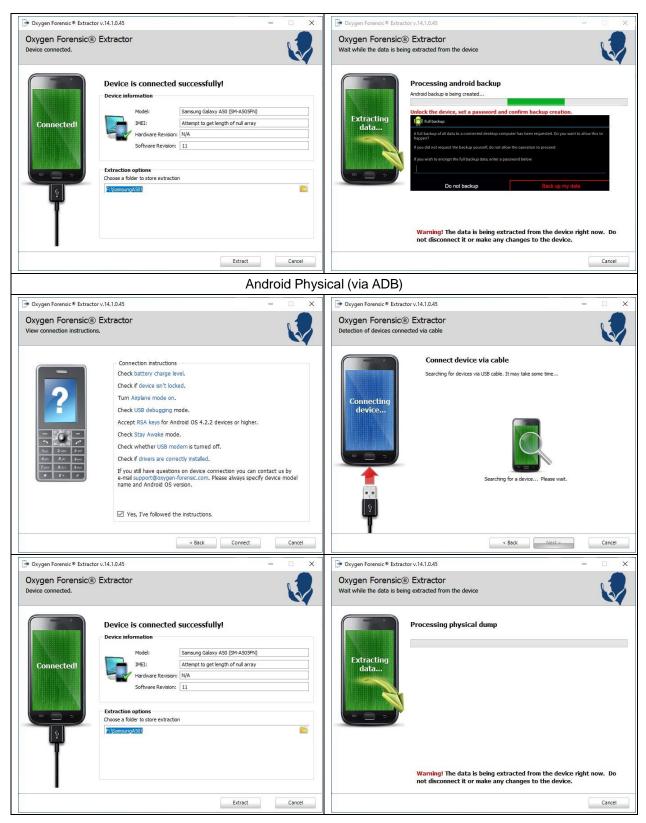
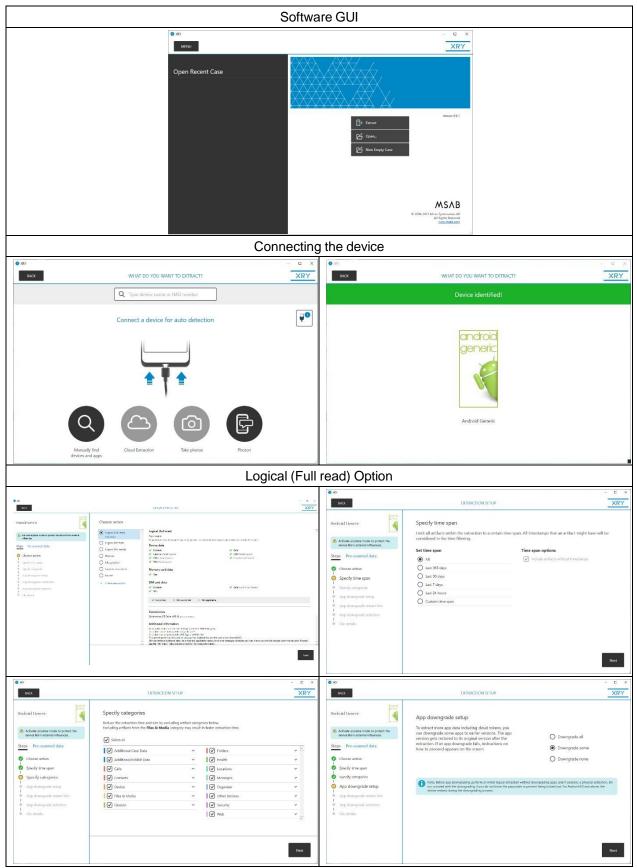


Table 3. OXYGEN Forensic Extractor user interface. Acquiring methods while the Samsung A50 device is connected.

> XRY



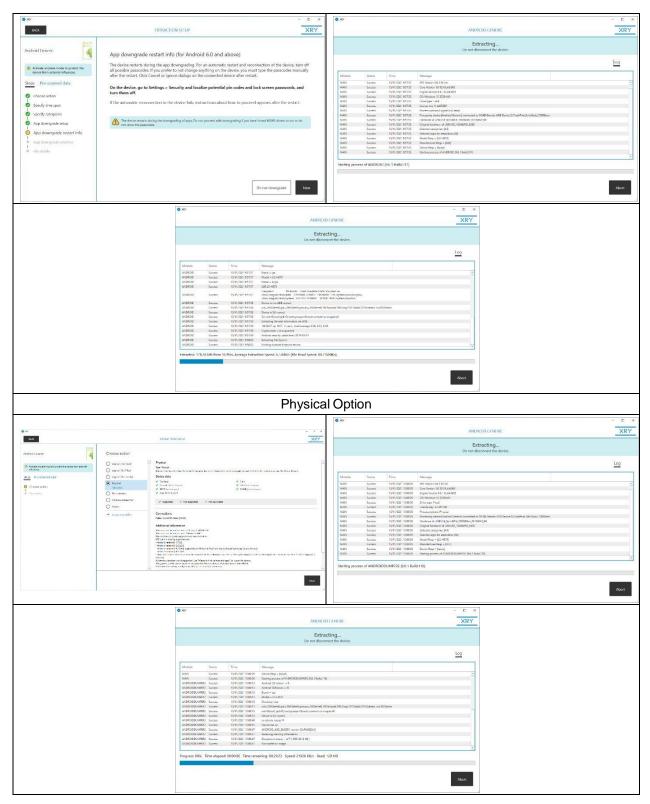


Table 4. MSAB XRY user interface. Acquiring methods while the LG G6 device is connected.

2.3.1. Speed of imaging and transferring data

The speed of data extraction of a device's storage is depended on the quality and length of used cables and of course the device itself. Some vendors, as a cost reduction method, implement an older USB 2.0 chipset on the devices irrespective the existence of a high-speed USB – C port on them. To take a glance of how the transfer speed is affected by these two factors in two different devices with the same acquiring setup (same PC, cable and software), an example between Samsung A50 and LG G6 is presented below (**Figures 13** and **14**). As shown the write speed is dramatically lower in Samsung, with value of 43MB/sec, compared with LG' s 108MB/s while using the Magnet Acquire forensic tool.

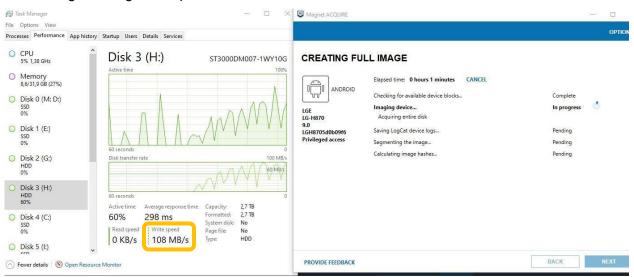


Figure 13. Speed monitoring while acquiring LG G6 device with MAGNET Acquire.

🕎 Task Manager	- 🗆 X	Magnet ACQUIRE			- 🗆 X
File Options View Processes Performance App h	nistory Startup Users Details Services				OPTIONS
 CPU 7% 2,51 GHz Memory 7,2/31,9 GB (23%) Disk 0 (M: D:) 30% Disk 1 (E:) 30% Disk 2 (G:) HDD 0% Disk 3 (H:) 20% Disk 4 (C:) 31% Disk 5 (f:) 	Active time 00% 60 seconds 0 Bisk transfer rate 00% 60 seconds 0 Bisk transfer rate 00 MB/s 60 seconds 0 Bisk transfer rate 00 MB/s 60 seconds 0 Bisk transfer rate 00 MB/s 60 seconds 0 Bisk transfer rate 00 MB/s Bisk transfer rate 00 MB/s Bisk transfer rate 100 MB/s Bisk trate 100 MB/s Bisk	CREATING FU ANDROID samsung SM-A305TN 11 R38M90KYS7V Privileged access	Elapsed time: 0 hours 8 minutes CANCEL Checking for available device blocks Imaging device Acquiring entire disk Saving LogCat device logs Segmenting the image Calculating image hashes	Complete In progress Pending Pending Pending	e
 DISK 3 (i.) Fewer details Solution Open Re 	• esource Monitor	PROVIDE FEEDBACK		BACK	NEXT

Figure 14. Speed monitoring while acquiring Samsung A50 device with MAGNET Acquire.

2.4. Examination of the acquired data

There is a wide range of options for software and tools specialized in the examination and analysis of the acquired databases. The criteria of choice are the examiners' experience and the available budget. While non-commercial tools may lack of special features, technical support and a user-friendly interface, the commercial ones have a significant cost that tend to be in the form of an annual subscription. Below are depicted some of the user interfaces of the abovementioned tools that are referred in paragraph §1.4.2. It has to be mentioned that the following used tools are for demonstration reasons and in-depth exploring their capabilities is out of scope of this Thesis. The choice of the used tool was depended on the examination and the analysis of the instant messaging application's databases.

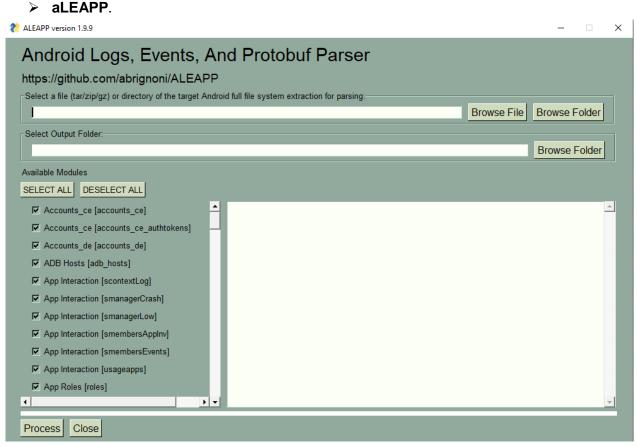


Figure 15. aLEAPP user interface.

> DB Browser for SQLite.

Name	^ IIC	COL.	view Tools Help				1						
Tables (24)	GN	ew Data	abase 🔒 Open Da	atabase 🚬 🛱 Write	Chang	ges 🕼 Revert C	hanges 🛛 🎧 Open F	roject 👔 Save I	Project Attach	Database	💢 Close Da	tabase	
> 🗾 applications	Data	base Str	ucture Browse Da	ta Edit Pragmas	Exe	cute SQL			Plot				8
> backgrounds		1000				No.	- 1		Columns	x			Y. /
> 📑 blocked_data	Table	me	essages	× 🛿 😼	0		. 🖪 🐴 » F	Iter in any column	Row #	Ň		Y1	т. г Г
> 📑 chat_extensions		id	conversation id	order key		msg date	token	conversatio	_rowid_				Ľ
chatex_suggestions	l r	-	Filter	Filter		Filter	Filter	Filter	jd	a 🗍			Ē
> conference_calls		-inter			-	1 contraction of the second se		10.121	conversation_i			Ц	F
> conversations	1	1	1	561419557183734	19295	1630586306677	56141955718373493	295	msg date	H		П	F
> gc_file	2	5	2	561419975976218	37124	1630587305138	5614199759762187	124	token				Ē
> group_delete_all_from_participant	3	6	2	561419983804048	32659	1630587323855	56141998380404820	59	conversation_ participant_id			H	F
> hidden_gems	- T	-						10.7	<		_		Ĕ
> iii hidden_gems_data	4	7	2	56141998069690	/8209	163058/316509	56141998069690782	209					
> messages	5	8	2	561419985727975	54766	1630587328323	5614199857279754	766	5				
> messages_calls	6	9	2	561419987830580	00540	1630587333439	5614199878305800	540	- i				
> 📑 messages_likes	7	10	2	E61410000643E10	17517	1600507007705	5614199896425193						
messages_reminders									4				
> iii participants	8	11	2	561419991705697	75610	1630587342609	5614199917056975	510					
> 🔄 participants_info	9	12	2	561419993986979	94957	1630587348037	5614199939869794	957	3				
> public_accounts	10	13	2	561410005816115	54465	1630587352425	5614199958161154	165	° -				
> 🛅 purchase								10203					
> i recent_stickers	11	14	2	561419997828542	24787	1630587357211	5614199978285424	/87	2				
> 🔄 remote_banners									-				
> 📑 sqlite_sequence													
> 🗾 stickers									1				
> stickers_packages													
Y 📎 Indices (40)													
> > idx_CONVERSATION_date_desc									0	1	1	<u>i.</u>	ينتند
idx_CONVERSATION_favourite_conversation_AND									0 1	2	2	-	5
> > idx_CONVERSATION_flags	<							>	Line type: Line	✓ Point sh	hape: Disc	\sim	
> 📎 idx_CONVERSATION_group_id		1-:	10 of 11 🕨 🔰			Go to:	1		SQL Log Plot				
> <pre>> idx_CONVERSATION_participant_id_1</pre>	~								SQLLOG MOL				

Figure 16. DB Browser for SQLite user interface (database: "viber_messages.db").

> SQlite Expert.

ē 🔤 🥇 🛃 🌄 🍕 🙀 🛛	😺 🛅 🗔 🛅 😡	👒 👒 🔘 🔘 🔽 🗽	; <mark>28</mark> 26 26	Ø D D) 🌡	A* A* 🐵 🗔	🚺 Nativ	e
atabase: viber_messages(2) File: C:\Users\: `\Desktop\database	es\16_03_30 messages sent\data3\databases\v	iber_messages				SQLite library: sqlite3.c	dll 3.36.0 [FTS3 F	TS4 FTS5 RTREE
viber_messages	Database Extensions	Schema Data DDL Design SQL builder	SOL Scripting					
viber_messages(1)	Databases							
applications	Databases							
- backgrounds	Database Status	File Name					File S	
blocked_data	main Conne	cted (read-write) C:\Users\kotsi\Desktop\datab	ases\16_03_30 message	s sent\data3\database	s\viber_mer	ssages	3	97312
chatex_suggestions								
chat_extensions	<							
conference_calls	Database properties			Library/Session/Con	nection pro	perties		
conversations	Name	Value	^	Name		Value		
gc_file	▶ application_id	0		automatic index		on		
group_delete_all_from_participant	and the second sec	full		busy_timeout		3000		
hidden_gems	auto_vacuum cache_size	-2000		cache_spill		483		
hidden_gems_data					- V	off		
messages	collation_list	[BINARY], [NOCASE], [RTRIM]		case_sensitive_like	1			
messages_calls	encoding	UTF-8		cell_size_check		off		
messages_likes	foreign_keys	on		checkpoint_fullfsy	nc			
messages_reminders	freelist_count	0				[COMPILER=msvc-1928], [E [ENABLE EXPLAIN COMM		
participants	journal_mode	delete		compile_options		[ENABLE_FTS3_PARENTHES		
participants_info	journal_size_limit	-1		1.050		[ENABLE_ICU], [ENABLE_JS	ON1], [ENABLE_	
public_accounts	max_page_count	1073741823				[SYSTEM_MALLOC], [THRE	ADSAFE=1]	
purchase	mmap_size	0		data_version		1		
recent_stickers	page_count	97		defer_foreign_key	5	off		
remote_banners stickers	page_size	4096		fullfsync		off		
	schema_version	72		ignore_check_con	straints	off		
United Stickers_packages	user_version	207		legacy_file_format	Ł			
applications	writable_schema	off		locking_mode		normal		
backgrounds				query_only		off		
blocked_data				read_uncommitte	:d	off		
- chatex_suggestions				recursive_triggers		off		
chat_extensions				reverse_unordered	d_selects	off		
conference calls				secure_delete		off		
conversations				soft_heap_limit		0		
qc_file				synchronous		full		
group_delete_all_from_participant	<		, [*]			1.4.5		
hidden gems								
hidden_gems_data	Apply >	Cancel	Refresh	Check (& Vacuum	Reindex All Tables	Repair	Backs

Figure 17. SQlite Expert user interface (database: "viber_messages.db").

> Oxygen forensic SQlite Viewer.

Open 🔹 Close 🛛 🔀 Visual Query B	luilder 📋 Recover de	eleted records in View	er 🔑 Search 🛛 🔂 Export + 🎃 Print + 🔛 Column options 🍓 Options 🥥 Help		Filte	ering criteria
Tables 🔺	Blocks containing deleted	data 🗎 SQL Editor	👪 Visual Query Bullder 🛛 🗎 DDL	📮 »	Туре	Value
	le name 🔹	Address A	 Hex data 	Text data	UTF-8	
	nema_table	00000122 - 00004070			Unicode (UTF-16.	*************
	nversations	00008204 - 00012152			Unicode (UTF-16.	
	ite_sequence	00012314 - 00016194			OLE Automation .	<can't convert=""></can't>
	ite_sequence	00016240 - 00016257	conversations	conversations	Unix Epoch Time	. <can't convert=""></can't>
	blic_accounts	00016392 - 00020479			Unix Epoch Time L	E <can't convert=""></can't>
	rticipants	00020496 - 00024516			Unix Epoch Time	<can't convert=""></can't>
	rticipants	00024542 - 00024552			Unix Epoch Time	<can't convert=""></can't>
	rticipants_info	00024590 - 00028041			Unix Epoch Time	. <can't convert=""></can't>
group_delete_all_fr mes	ssages	00028702 - 00030998			Unix Epoch Time	. <can't convert=""></can't>
	ssages	00031241 - 00031344	h % + M鎖 @p[]q-M鎖 @6 Qmhp5XZ)v50=`@ Mustw,	h	OS X Epoch Time.	<can't convert=""></can't>
	ssages_calls	00032776 - 00036863			OS X Epoch Time.	<can't convert=""></can't>
messages(11/0) mes	ssages_likes	00036872 - 00040959			OS X Epoch Time.	<can't convert=""></can't>
messages_calls(0/0) stic	kers packages	00040976 - 00044641			OS X Epoch Time.	<can't convert=""></can't>
	tkers	00049166 - 00053241	z^D* �§t[A& ≅ + hN3 ± }cI. �© uZ@& �*i IP5 磨 ¬x_D* H© sW< 爏µ eJ0 I «	z.^.D.*ô.Û.Á.§t.[,A.&I.Ô.\$h.N.3ÿ.a.Ĕ.±	BlackBerry Time B	E <can't convert=""></can't>
messages_reminders(0 rec	ent_stickers	00057352 - 00061439			BlackBerry Time L	E <can't convert=""></can't>
participants(4/0) bac	ckgrounds	00061448 - 00065535			Chrome Time BE	<can't convert=""></can't>
participants_info(3/0) app	plications	00065544 - 00069631			Chrome Time LE	<can't convert=""></can't>
public_accounts(0/0) blo	cked_data	00069640 - 00073727			Symbian Epoch	<can't convert=""></can't>
g purchase(0/0) pur	rchase	00077832 - 00081919			Symbian Epoch	2/12/34938 8:01:
rem	note banners	00086024 - 00090111			MS File Time BE	1/1/1601
remote_banners(0/0) gro	oup delete all from p	00090120 - 00094207			Unsigned-8	<can't convert=""></can't>
gite_sequence(9/0)	nference calls	00098312 - 00102399			Signed-8	<can't convert=""></can't>
	at extensions	00106518 - 00109515			Unsigned-16 BE	<can't convert=""></can't>
stickers_packages(4/0) cha	atex suggestions	00110622 - 00114634	.}]C, ± II&提◎CG+ i34 oY7! Nº s[7 混 ~iH3 �Y yCA) 天� nQ0 £o< 賺×R '		Signed-16 BE	<can't convert=""></can't>
III All Deleted Data	lden gems	00114702 - 00118778	至8 ¶ ^D1 + P-28 u[* Ki C. 田 dE+ 長+ } U@ 健iyU@ 舉? {K4 馆i{fF f= uX ± ~		Unsigned-16 LE	<can't convert=""></can't>
	den gems data	00122890 - 00125917		, , ,	Signed-16 LE	<can't convert=""></can't>
me	ssages reminders	00126984 - 00131071			Unsigned-32 BE	<can't convert=""></can't>
OC.	file	00131082 - 00134952			Signed-32 BE	<can't convert=""></can't>
	nema table	00299026 - 00299681		.š	Unsigned-32 LE	<can't convert=""></can't>
	nema_table	00303148 - 00303482	2^ TT '? I ; indexidx_MESSAGES_extra_mimemessages'CREATE INDEX idx_MESSAGES_extra_mime ON [Signed-32 LE	<can't convert=""></can't>
sch	nema table	00305548 - 00305592	-C indexsqlite autoindex purchase 1purch	Cindexsglite autoindex purchase 1purch	Signed-64 BE	<can't convert=""></can't>
	nema_table	00307240 - 00307825	x !k5 indexidx_MESSAGES_user_i _ indexpart_idx_MESSAGES_community_empty_messageme		Signed-64LE	<can't convert=""></can't>
sch	nema_table	00311328 - 00311792	u D < -C indexsqlte_autoindex_purchase_tpurchase y EtablepurchasecREATE TABLE purchase	.u.D. <cindexsglite_autoindex< td=""><td>Single</td><td><can't convert=""></can't></td></cindexsglite_autoindex<>	Single	<can't convert=""></can't>
sch	nema table	00315438 - 00315777	HH D 3% Otriggertrigger apps updateapplicationsCREATE TRIGGER trigger apps update AFTER UPDATE ON applications		Double	<can't convert=""></can't>
	nema table	00318090 - 00318700	c indexpart idx MESSAGES community empty message M Y' triggerbusiness inbox grouping insert triggerconversationsC	cf.indexpart idx MESSAGES community empty mess	Uuencoded	20 20 20 20 3
	ckers	00319782 - 00319790			Base64	□ÿ□□ZnW©j׳
stic	skers	00323772 - 00325246			Base64 (UTF8)	□1□@j+br*Zgw
	atex suggestions	00327926 - 00327965			Base64 (UTF-16.	# 湖口豊牢焼口
	atex_suggestions	00331960 - 00331967			Base64 (UTF-16 .	De掘圣種持 * 黛麗
	atex suggestions	00336050 - 00336085		· · · · · · · · · · · · · · · · · · ·		
< (>		

Figure 18. Oxygen forensic SQlite Viewer user interface (database: "viber_messages.db").

> Magnet Axiom Forensics.

		messages												^] Column vie	
L EVIDEN \vmware app_Mis	QL	TE VIEWER											*	Modifier 2/9/2021	
app_tex app_we	Select table messages *														
	ND BUILD QUERY EXPORT SHOW / HIDE COLLIMNS														
databas at	Ŧ	location_lng T	extra_uri ₹	destination_uri \mathbb{T}	extra_status 🔻	message_global_id 🔻	extra_upload_id \mathbb{T}	extra_download_id \mathbb{T}	extra_bucket_name T	sticker_id 🔻	extra_duration \mathbb{T}	body T	description \mathbf{T}	2/9/2021	
files												timebomb/60		2/9/2021	
no_back shared												Once upon a midnight dreary, while I		2/9/2021	
Jindico_		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	pondered, weak and		2/9/2021	
												weary, While I nodded, nearly		2/9/2021	
		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	napping, suddenly there came a tapping,		2/9/2021	
												Over many a quaint and		2/9/2021	
		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	curious volume of forgotten lore—		2/9/2021	
		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	As of some one gently rapping, rapping at my		2/9/2021	
		0	(null)	(null)	°	0	0	(null)	(nuii)	(null)	0	chamber door.		2/9/2021	
		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	"Tis some visitor," I muttered, "tapping at		2/9/2021	
			(()				()	()	(my chamber door-		2/9/2021	
		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	Only this and nothing more."		2/9/2021	
		0	(null)	(null)	3	0	0	(null)	(null)	(null)	0	Ah, distinctly I remember it was in the	•		
4													4		

Figure 19. Magnet Axiom Forensics user interface (database: "viber_messages.db").

> UFED Physical Analyzer.

÷	File View Tools Cloud Extract Python Plug-ins Report Help	UFED CLOUD insided (Search Q) Atheneced	[0]
습 Home	G LG H87 GUble wizard Double-click or drag the database tables to the work area and link database fields.	× ×	
\odot	Search Query builder DB viewer	Trans	slate Go t
Timeline	Analyzed ▲ III main > III AF > III AF > III AF > III AF	messages (main) 🗶	ttachments (0)
Analyzed Data	Co Co	A tobeled inte A tobeled inte di integer conversion/d orde_key intege	
ile Systems			
-Q- Insights	Aligned Content and Anti-Anti-Anti-Anti-Anti-Anti-Anti-Anti-	eEntry#	Point","p ,"_si":56
Tags	idx CONVERSATION group id	From messages	itus":"Ο", ν~*ιι)21 5:22:12 μμ
Reports	discher Stades conversation, ype AND_coler_participant converse converse conversation and conversationa	Preview Max results: 10 - Preview Count	
Cloud			
	2 c c > 0 Metar(core),xxy	Close Previous Next	
	> 📮 Messages (4)	*	
	Total: 8 Deduplication: 0	Items: 8/8 Selected: 8	

Figure 20. UFED Physical Analyzer user interface (database: "viber_messages.db").

> Autopsy Suite.

2	C Listing SOLite Database VI	RED MESSAGES	Table MPSSAGPS									33 Res
Data Sources	Table Thumbnail		Table MESSAGES									33 Ne
Pile Views											Save T	Table as CS
X Deleted Files	destination_uri	extra_status	message_global_id	extra_upload_id	extra_download_id	extra_bucket_name	sticker_id	extra_duration	body	description	msg_info	msg_ini
Data Artifacts	1.0000000000000000000000000000000000000	0	0	0				0	timebonb/60		0	BLOBS
SOLITE Database VIBER MESSAGES Table CHAT EXTENSIONS (21)		3	0	0				0	Once upon a midnight dreary, while I pondered, weak and		-0-	BLOBS 1
- 4 SQUIte Database VIBER_MESSAGES Table CHATEX_SUGGESTIONS (2733)		3	0	0				0	While I nodded, nearly napping, suddenly there came a ta		0	BLOBS
SQLite Database VIBER_MESSAGES Table CONVERSATIONS (6)		3	0	0				0	Over many a quaint and curious volume of forgotten lore-		0.	BLOBS
SQUte Database VIBER_MESSAGES Table GC_FILE (3) SQUte Database VIBER_MESSAGES Table HIDDEN GEMS (474)		3	0	0				0	As of some one gently rapping, rapping at my chamber door		42	ELOBS
SQUIte Database VIBER_MESSAGES Table HIDDEN_GEMS_DATA (3)		3	0	0				0	"Tis some visitor," I muttered, "tapping at my chamber doo.		0	BLOBS
SQUITE Database VIDER_MESSAGES Table MESSAGES (33)		3	0	0				0	Only this and nothing more."		0	BLOBS
SQUte Database VIBER_MESSAGES Table PARTICIPANTS (12)		3	0	0				0	Ah, distinctly Tremember it was in the bleak December;		0	BLOBS
SQUIte Database VIBER_MESSAGES Table PARTICIPANTS_INFO (9)		3	0	0				0	And each separate dying ember wrought its ghost upon th		0	BLOBS
SQLite Database VIBER_MESSAGES Table SQLITE_SEQUENCE (27) SQLite Database VIBER_MESSAGES Table STICKERS (699)		3	0	0				0	Eagerly I wished the morrow;vainly I had sought to borrow	v	0	BLOBS
 SOLITE Database VIBER MESSAGES Table STICKERS PACKAGES (12) 		3	0	0				0	message_deleted/Qhnb3h733pE=		0	BLOBS
Analysis Results		0.	0	0				0	timebomb/60		0	BLOBS
Keyword Hits (I)	۲.					17						3
OS Accounts	and the second se											
Tags Reports	Stings Indexed											



\triangleright	Para	abe	en E	3.													
0 # *	(UTC+02:0	00) Athen	s, Buchares	it 🔻	+ → ≣	· •							Parabe	en's E3:FREE - ca	ase.e3		
CASE	EVIDENCE	ANALYS	IS REP	PORTS	TOOLS VI	EW I	EXPORT										
Add Evidence	Find Chat	/	Reload	Remove	Start		Cloud Root Android	Validate	Change	View Skip	Add to	Create	Print	Calculate	Add File MD5 to		
Add Evidence		Source	Evidence		Acquisition		Import Device	Data	Settings	List	Skip List	Attachment List	PTITIC	Subkey Hash	Hash Database		
	Evic	dence					Mobile Data		Settings		Mai	lstorages		Registry	Files		
Case Cont	ont			а	hessage	s 1-11	Sorted Files	Hash [Databases	1							
Case Cont	CIL			-							12						
Case	Content	Sor.	ted Files		Internal Pat	h: e3://	Case/viber_messages/dat	abase/tables/m	essages/1-	15?item=row	_1						
	content	001	CLUTT BLU			rowid	_id	-		conversatio	n_id	order_key	90 1	msg		token	convers
Items			In	Total		1	1			1		5,614,195,	571,837,34	1,630 1,630	,586,306,677	5,614,195,571,837,349,295	0
🕶 🗌 🚔 Ca	se					5	5			2		5,614,199,	759,762,18	37,124 1,630	,587,305,138	5,614,199,759,762,187,124	0
▼□류	viber_message	es				6	6			2		5,614,199,	838,040,48	32,659 1,630	,587,323,855	5,614,199,838,040,482,659	0
•	SQLite Data	abase				7	7			2		5,614,199,	806,969,07	78,209 1,630	,587,316,509	5,614,199,806,969,078,209	0
•	Tables					8	8			2		5,614,199,	857,279,75	54,766 1,630	,587,328,323	5,614,199,857,279,754,766	0
	I alit	e_master	79			9	9			2		5,614,199,	878,305,80	00,540 1,630	,587,333,439	5,614,199,878,305,800,540	0
	Conv	ersations	2			10	10			2		5,614,199,	896,425,19	93,517 1,630	,587,337,735	5,614,199,896,425,193,517	0
	I h sqlit	e_sequer	nce 9			11	11			2		5,614,199,9	917,056,97	75,610 1,630	587,342,609	5,614,199,917,056,975,610	0
	Din publ					12	12			2		5,614,199,	939,869,79	4,957 1,630	,587,348,037	5,614,199,939,869,794,957	0
	De parti		4			13	13			2		5,614,199,5	958,161,15	54,465 1,630	,587,352,425	5,614,199,958,161,154,465	0
	Di parti		info 3			14	14			2		5,614,199,5	978,285,42	24,787 1,630	,587,357,211	5,614,199,978,285,424,787	0
	▼ □ 🔂 mess	sages nessages	11 1 11														

Figure 22. Paraben E3 user interface (database: "viber_messages.db").

> Belkasoft X.

pen file: \\OPENMV\Thesis\images\0	2_09_202	1\lg g6-experiment2\le	g g6\02_09_2021\15	_58_10 messages sent\data	\databases\viber_message	S	
applications (0)	Data	Structure					
packgrounds (0)		Record type	_id	conversation id	order_key	msg_date	token
olocked_data (0)							564 44000 700054
hat_extensions (7)	0		14	2	56141999782854247	1630587357211	56141999782854
hatex_suggestions (1671)	0		1	1	56141955718373492	1630586306677	56141955718373
onference_calls (0)	0		13	2	56141999581611544	1630587352425	56141999581611
onversations (4)	0		12	2	56141999398697949	1630587348037	56141999398697
pc_file (1) proup delete all from participant (0)	0		11	2	56141999170569756	1630587342609	56141999170569
idden_gems (395)	0		10	2	56141998964251935	1630587337735	56141998964251
nidden_gems_data (1)	0		9	2	56141998783058005	1630587333439	56141998783058
nessages (23)			8	2	56141998572797547	1630587328323	56141998572797
nessages_calls (0)	_		7				
nessages_likes (0)	0			2	56141998069690782	1630587316509	56141998069690
nessages_reminders (0)			6	2	56141998380404826	1630587323855	56141998380404
participants (4)	0		5	2	56141997597621871	1630587305138	56141997597621
participants_info (3)		Journal	14	2	56141999782854247	1630587357211	56141999782854
oublic_accounts (0)	0	Journal	1	1	56141955718373492	1630586306677	56141955718373
ourchase (0)	0	Journal	13	2	56141999581611544	1630587352425	56141999581611
ecent_stickers (67) emote_banners (0)	0	Journal	12	2	56141999398697949	1630587348037	56141999398697
glite_sequence (9)	_						
ickers (466)	0	Journal	11	2	56141999170569756	1630587342609	56141999170569
ickers packages (4)	0	Journal	10	2	56141998964251935	1630587337735	56141998964251
Jnallocated space		Journal	9	2	56141998783058005	1630587333439	56141998783058

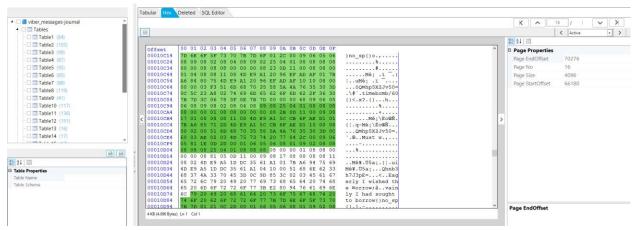
Figure 23. Belkasoft X user interface (database: "viber_messages.db").

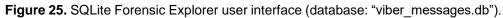
Name Topse Topse Topse Contract Mondified Texacit Analysis Analysis Metadata Decision = (Rod directory) existing 21.146 21.146 70.6016 <		ages							urs ago					
a metry messages 12 exiting already viewed spliteds S01 VB 2021/10/23d165154/2021/09/02d205648 A Page size 4006 File c	Name	•	Description	🝸 Туре		Created	Modified	Record changed	Attr.	1st sector	Analysis	Metadata		T Device ty
dickers.psckage.stav virtual (for examination p trv 501 B UTF-8 indicers.psckage.stav virtual (for examination p trv 67.8 B UTF-8 indicers.psckage.stav virtual (for examination p trv 09.18 B UTF-8 UTF-8 indicers.psckage.stav virtual (for examination p trv 167.8 UTF-8 UTF-8 indicer.gem.stav virtual (for examination p trv 167.8 UTF-8 UTF-8 indicer.gem.stav virtual (for examination p trv 17.8 B UTF-8 UTF-8 indicer.gem.stav virtual (for examination p trv 18.1 R UTF-8 UTF-8 inder.gem.stav virtual (for examination p trv 18.1 R UTF-8 UTF-8 conversation.stav virtual (for examination p trv 13.8 R UTF-8 UTF-8 chate.stavion.id order, key msd.gem.gem.stav UTF-8 UTF-8 UTF-8 conversation.stav virtual (for examination p trv 43.3 KB UTF-8 UTF-8 UTF-8 conversation.stav virtual (for examination p trv 43.3 KB UTF-8 UTF-8 UTF-8 UTF-8 UTF-8 UTF-8														
ticketstv virtual (for examination p tsv 6.7.88 UTF-8 splite sequence.tsv virtual (for examination p tsv 181.8 UTF-8 participants.tsv virtual (for examination p tsv 0.9.48 UTF-8 indiden.gems.dsta.tsv virtual (for examination p tsv 167.8 UTF-8 indiden.gems.dsta.tsv virtual (for examination p tsv 17.8 UTF-8 indiden.gems.dsta.tsv virtual (for examination p tsv 17.8 UTF-8 indiden.gems.dsta.tsv virtual (for examination p tsv 17.8 UTF-8 c.fict.v virtual (for examination p tsv 27.5 8 UTF-8 UTF-8 c.onversation.tsv virtual (for examination p tsv 13.88 UTF-8 UTF-8 c.onversation.id order key mg.dafe token conversation.ype participant.id group.id extra_dscte.tr						2021/10/23d16:	51:54 2021/09/02d20:56:4	8	Α				96 File c	
Indices UTF-8 participants, info.tsv Virtual (for examination p tsv 0.9 K8 VIT-8 VIT-8 participants, info.tsv Virtual (for examination p tsv 0.9 K8 VIT-8 VIT-8 metsignets, info.tsv Virtual (for examination p tsv 167.8 VIT-8 VIT-8 metsignets, info.tsv Virtual (for examination p tsv 157.8 VIT-8 VIT-8 inided gene, gene, stat.tsv Virtual (for examination p tsv 58.88 VIT-8 VIT-8 VIT-8 conversations.tsv Virtual (for examination p tsv 25.88 VIT-8 VIT-8 VIT-8 conversations.tsv Virtual (for examination p tsv 25.88 VIT-8 VIT-8 VIT-8 read_message time sciol post sciad call Exerct and call group (d extra_tsuckt_name scicker_1 d ext														
participants_info.tsv virtual (for examination p tsv 0.9 KB UTF-8 metasgetsv virtual (for examination p tsv 167.8 UTF-8 inder.gems_data.tsv virtual (for examination p tsv 2.7 KB UTF-8 inder.gems_data.tsv virtual (for examination p tsv 2.1 KB UTF-8 c.f.fictsv virtual (for examination p tsv 2.5 KB UTF-8 c.c.fictsv virtual (for examination p tsv 2.5 KB UTF-8 c.c.metrastion.stv virtual (for examination p tsv 1.3 KB UTF-8 c.newsration.stv virtual (for examination p tsv 1.3 KB UTF-8 c.newsration.stv virtual (for examination p tsv 1.3 KB UTF-8 c.newsration.gl of order !key msg_dat tokats_suggestion.stv UTF-8 virtual for examination p tsv 1.3 KB UTF-8 UTF-8 conversation.gl of order !key msg_dat tokats_suggestion.stv UTF-8 virtual for examination p tsv 1.3 KB UTF-8 UTF-8 conversation.gl of order !key msg_dat towata														
participants.tov virtual (for examination p., itov 167 8 UTF-8 Intelegets/v virtual (for examination p., itov 2.1 KB UTF-8 UTF-8 indder.gerns.tv virtual (for examination p., itov 1.1 KB UTF-8 UTF-8 inder.gerns.tv virtual (for examination p., itov 5.8 KB UTF-8 UTF-8 conversations.tv virtual (for examination p., itov 0.8 KB UTF-8 UTF-8 conversations.tv virtual (for examination p., itov 0.8 KB UTF-8 UTF-8 conversation.tv virtual (for examination p., itov 0.8 KB UTF-8 UTF-8 conversation.tv virtual (for examination p., itov 45.3 KB UTF-8 UTF-8 volume File Preview Details Calendar Eleged Raw Sync All extra_flags deleted send_type extra_duration body description 1 ofter/staflags odlated send_type for extra_unation for extra_mine user_jd extra_unation for extra_mine user_jd group jd extra_flags	sqlite_s	sequence.tsv	virtual (for examination p	tsv										
Intestages.bv Vitual (or examination p bv 2.7.KB UIT=8 Inider.gerns_dats.tv Vitual (or examination p bv 1.1 KB UIT=8 Inider.gerns_dats.tv Vitual (or examination p btv 2.5 KB UIT=8 or.glicit.tv Vitual (or examination p btv 2.5 KB UIT=8 c.file.tv Vitual (or examination p btv 0.8 KB UIT=8 c.new station.stv Vitual (or examination p btv 0.8 KB UIT=8 c.new station.id Volume File Preview Details Callend# EE Legend Raw Sync Vitual (or examination p btv 45.3 KB UIT=8 Volume File Preview Details Calend# EE Legend Raw Sync VIII=8 UIT=8 conversation.id order, kv msdadats.mgi.d quoted message data my.reaction 1 1011 Calend# 0.5 KB 1 5614195571837349295 File 0 0 0 0 0 0 0 0 0 0 0 0 0	🛛 particij	ipants_info.tsv	virtual (for examination p	tsv	0.9 KB							UTF-8		
hidden_gems_data.tsv virtual (for examination p., tsv 1.1 KB UTF-8 hidden_gems.tsv virtual (for examination p., tsv 5.8 KB UTF-8 conversation.stv virtual (for examination p., tsv 273 B UTF-8 conversation.stv virtual (for examination p., tsv 273 B UTF-8 conversation.stv virtual (for examination p., tsv 1.3 KB UTF-8 chate_stogestion.stv virtual (for examination p., tsv 1.3 KB UTF-8 volume File Preview Details Calledar III: Legend 1 avera diago extra_uind destination_uin extra_globacket_name sticker_id extra_flags deleted send_type extra_flags deleted send_type extra_flags extra_uind deleted send_type extra_flags deleted send_type extra_flags extra_download dexta_uinda (dextra_download) dextra_download) extra_download extra_download transition_id destra_tables 0 0 0 0 0 0 1 fold destra_tables 0 0 0 0 0 0 0	partici	ipants.tsv	virtual (for examination p	tsv	167 B							UTF-8		
inidad gens.tsv virtual (for examination p., tsv 5.8 KB UTF-8 gc_file.tsv virtual (for examination p., tsv 2758 UTF-8 chat_extensions.tsv virtual (for examination p., tsv 0.8 KB UTF-8 chat_extensions.tsv virtual (for examination p., tsv 0.8 KB UTF-8 chat_extensions.tsv virtual (for examination p., tsv 1.3 KB UTF-8 chat_extensions.tsv virtual (for examination p., tsv 45.3 KB UTF-8 chat_extanuin destination unital (for examination p., tsv 45.3 KB UTF-8 volume File Preview Details Galendr (E: Legend) I: Raw Sync	messa	iges.tsv	virtual (for examination p	tsv										
gc_file.tv virtual (for examination p tv 275 8 UTF-8 conversation.stv virtual (for examination p tv 0.8 kB UTF-8 chate_cetension.stv virtual (for examination p tv 0.8 kB UTF-8 chate_cetension.stv virtual (for examination p tv 1.3 KB UTF-8 volume File Preview Details Galendar EE Legend 1 Raw Sync All All UTF-8 UTF-8 volume File Preview Details Galendar EE Legend 1 Raw Sync All Group and transpace UTF-8 UTF-8 UTF-8 volume File Preview Details Galendar EE Legend 1 Raw Sync All Transpace UTF-8 U	hidden	n_gems_data.tsv	virtual (for examination p	tsv	1.1 KB							UTF-8		
Conversations.tsv virtual (for examination p tsv 0.8 kB UTF-8 chate_staggestions.tsv virtual (for examination p tsv 1.3 kB UTF-8 Volume File Preview Details Gallery Gallery group_id extra_flags deleted send_type extra_flags if conversation_id order_key msg_date token conversation_idversation_unicking extra_flags deleted send_type extra_function unicking user_id 1 conversation_idversation_unicking group id extra_flags deleted send_type extra_flags deleted send_type extra_function unicking 1 conversation_idversation_unickings 0 0 11011 QmhpSZLSS QmhpSZLSS 2 0 2 561419957062187124 4 1 4226 0 0	hidden	n_gems.tsv	virtual (for examination p	tsv	5.8 KB							UTF-8		
Chat_extensions.tsv virtual (for examination p tsv 1.3 KB UTF-8 Chate_suggestions.tsv virtual (for examination p tsv 43.3 KB OTF-8 Volume File Preview Details Gallery Calendar Esteped Raw Sync AB Sync AB Sync Volume File Preview UTF-8 UTF-8 conversation id order (ksy msg_date token conversation type participant id unread flag group id extra_flags deleted send type extra_mime user_id add_message timebomb/60 0 0 0 0 1 1011 Conversation type extra_duration body description 1 561419957137342925 100 0	gc_file.	e.tsv	virtual (for examination p	tsv	275 B							UTF-8		
Chetex_suggestions.tsv virtual (for examination p tsv 45.3 KB UTF-8 Volume File Preview Details Gallery Calendar Example Participant, id unread flag extra_uni output extra_table destination extra_duration body description 1 5614195571837349295 1503668306677 5614195571837349295 1 1011 Qmhp5ZJF20 2 0 0 0 60 108543788 0 <	conver	rsations.tsv	virtual (for examination p	tsv	0.8 KB							UTF-8		
Volume File Preview → Details © Gallery Callendar Example Preview → Details © Gallery Callendar Example Preview → Details © Gallery Callendar Example Preview → Details © Gallery Conversation Type participant Id unread flag extra_duction extra_duction body description read message time scill_pos broadcast mode message data my read message time bomb/60 0 0 data time bomb/60 0 data time bomb/60 0 data tim bomb/60 da	chat_e	extensions.tsv	virtual (for examination p	tsv	1.3 KB							UTF-8		
Volume File Preview Details Callendar Example Sync Ass	chatex	suggestions.tsv	virtual (for examination p	tsv	45.3 KB							UTF-8		
0 Once upon a midnight dreary, while I pondered, weak and weary, f) 1 0 n.o. sp 0		ad_message_time scroll 5614195571837349295	pos broadcast_msg_ic 1630586306677 56141955	d qu	oted mes	sage_data m	xtra_download_id ny_reaction	extra_bucket_nam	ne st	icker_id	extra_durat	ion body	descriptio	on i
2 5614199838040426269 1630587323855 164199838040482659 1 4096 0	rea 1	ad_message_time scroll 5614195571837349295 0 timebo	pos broadcast_msg_ic 1630586306677 56141955 pmb/60 {} 0	d qu 71837349 0	oted_mes 1295 0	sage_data m 1 0 60 1	xtra_download_id ny_reaction 4112 0 08543758 0	extra_bucket_nam 134217728 0 0 {} 0	ne st 1	icker_id 1011	extra_durat Qmhp5XZJ	ion body /50= 2073650	descriptio 32 2	on r ? (
0 While I noded, neady napping, suddenly there came a tapping, so that 199806969078209 1 0 no.sp 0 <t< td=""><td>rea 1 2</td><td>ad_message_time scroll 5614195571837349295 0 timebo 5614199759762187124</td><td>pos broadcast_msg_id 1630586306677 56141955 pmb/60 { 0 1630587305138 56141997</td><td>d qu 71837349 0 759762187</td><td>oted_mes 1295 0 '124 0</td><td>sage_data m 1 0 60 1 4 1</td><td>xtra_download_id ny_reaction 4112 0 08543758 0 4226 0</td><td>extra_bucket_nam 134217728 0 0 {} 0 1024 0 0</td><td>ne st 1 0</td><td>icker_id 1011</td><td>extra_durat Qmhp5XZJ</td><td>ion body /50= 2073650</td><td>descriptio 32 2</td><td>on r 2 (</td></t<>	rea 1 2	ad_message_time scroll 5614195571837349295 0 timebo 5614199759762187124	pos broadcast_msg_id 1630586306677 56141955 pmb/60 { 0 1630587305138 56141997	d qu 71837349 0 759762187	oted_mes 1295 0 '124 0	sage_data m 1 0 60 1 4 1	xtra_download_id ny_reaction 4112 0 08543758 0 4226 0	extra_bucket_nam 134217728 0 0 {} 0 1024 0 0	ne st 1 0	icker_id 1011	extra_durat Qmhp5XZJ	ion body /50= 2073650	descriptio 32 2	on r 2 (
2 5614199806969078209 1630587316509 5614199806969078209 0	rea 1 2 0	ad message time scroll 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight	pos broadcast_msg_ic 1630586306677 56141955 pmb/60 { 0 1630587305138 56141997 dreary, while I pondered, wea	d qu 71837349 0 59762187 ak and we	oted_mes 1295 0 1224 0 ary,	sage_data m 1 0 60 1 4 1 {} 1	xtra_download_id ny_reaction 4112 0 08543758 0 4226 0 0 no_sp	extra_bucket_nam 134217728 0 0 {} 0 1024 0 0 0 0 0	ne st 1 0 0	icker_id 1011 Qhnt {}	extra_durat Qmhp5XZJ o3h7J3pE= 0 0	ion body /50= 2073650 2	descriptio 32 2 0 (on r <u>?</u> () (
0 Over many a quaint and curious volume of forgotten lorg	rea 1 2 0	ad_message_time scroll 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight 5614199838040482659	pos broadcast_msg_ic 1630586306677 56141955 0mb/60 { 0 1630587305138 56141997 dreary, while I pondered, wea 1630587323855 56141998	d qu 71837349 0 59762187 ak and we 38040482	oted_mes 1295 0 1224 0 ary, 2659 0	sage_data m 1 0 60 1 4 1 {} 1 4 1 4 1	xtra_download_id ny_reaction 4112 0 08543758 0 4226 0 0 no_sp 4096 0	extra_bucket_nam 134217728 0 0 {} 0 1024 0 0 0 0 0 0 0 0 0 0 0	ne st 1 0 0 0	icker_id 1011 Qhnb Qhnb Qhnb	extra_durat Qmhp5XZJ o3h7J3pE= 0 0 o3h7J3pE= 21	ion body /50= 2073650 2	descriptio 32 2 0 (on 1 2 (
2 5614199867279764766 163058723823 5614199867279764766 0 1 4096 0 0 0 Qhnb3h7Å3pE= 211551537 2 0 0 As of some gently rapping, rapping at my chamber door. 0 1 0 no sp 0 <t< td=""><td>rea 1 2 0 2</td><td>ad_message_timescroll 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec</td><td>pos broadcast_msg`id 1630586306677 56141955 mb/60 {} 0 1630587305138 56141997 dreary, while I pondered, were 1630587305138 163058730523855 56141993 greary, while I pondered, were 1630587323855 1 1630587323855 y 163048733855</td><td>d qu 71837349 0 59762187 ak and we 38040482 here cam</td><td>oted_mes 1295 0 1224 0 ary, 2659 0 e a tapping</td><td>sage_data m 1 0 60 1 4 1 (} 1 4 1 9, {}</td><td>xtra_download_id ny_reaction 4112 0 08543758 0 4226 0 0 no_sp 4096 0 1 0</td><td>extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td><td>ne st 1 0 0 0 0</td><td>icker_id 1011 Qhnb Qhnb Qhnb 0</td><td>extra_durat Qmhp5XZJ v3h7J3pE= 0 0 v3h7J3pE= 21 {} 0</td><td>ion body /50= 2073650 2 1535150</td><td>descriptio 32 2 0 (2 (</td><td>2 () (</td></t<>	rea 1 2 0 2	ad_message_timescroll 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec	pos broadcast_msg`id 1630586306677 56141955 mb/60 {} 0 1630587305138 56141997 dreary, while I pondered, were 1630587305138 163058730523855 56141993 greary, while I pondered, were 1630587323855 1 1630587323855 y 163048733855	d qu 71837349 0 59762187 ak and we 38040482 here cam	oted_mes 1295 0 1224 0 ary, 2659 0 e a tapping	sage_data m 1 0 60 1 4 1 (} 1 4 1 9, {}	xtra_download_id ny_reaction 4112 0 08543758 0 4226 0 0 no_sp 4096 0 1 0	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ne st 1 0 0 0 0	icker_id 1011 Qhnb Qhnb Qhnb 0	extra_durat Qmhp5XZJ v3h7J3pE= 0 0 v3h7J3pE= 21 {} 0	ion body /50= 2073650 2 1535150	descriptio 32 2 0 (2 (2 () (
2 5614199878305800540 1630587333439 5614199878305800540 1 4096 0 0 0 Qhmb3h7J3pE= 211572020 2 0 0 "Tis some visitor," I muttered, "tapping at my chamber door— 0 1 0 no_sp 0	rea 1 2 0 2	ad message time scroll 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddeo 5614199806969078209	pos broadcast_msg_ic 1630586306677 56141955 omb/60 { 0 1630587305138 56141997 dreary, while I pondered, wea 1630587323855 56141998 I, nearly napping, suddenly ti 1630587316509 56141998	d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078	oted_mes 1295 0 124 0 ary, 2659 0 e a tapping 209 0	sage_data m 1 0 60 1 4 1 (} 1 4 1 9, {}	xtra_download_id yy_reaction	extra_bucket_nam 134217728 0 0 8 0 1024 0 0 0 0 0 0 0 0 no_sp 0 0 0 0 0 0	ne st 1 0 0 0 0 0 0	icker_id 1011 Qhnt Q Qhnt 0 Qhnt 0 Qhnt	extra_dural Qmhp5XZJ 03h7J3pE= 0 0 03h7J3pE= 21 0 03h7J3pE= 21	ion body /50= 2073650 2 1535150	descriptio 32 2 0 (2 (on (
0 "Tis some vision," I muttered, "tapping at my chamber door— 0 1 0 no_sp 0 <td>rea 1 2 0 2 2</td> <td>tad_message_time_scroll 5614195571837349295 0 timebo 5514199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 5614199806969078209 0 Over many a q</td> <td>pos broadcast_msg_it i 630586306677 56141952 pmb/60 0 umb/60 0 0 i 630587305138 56141992 dreary, while I pondered, were i 1630587323855 56141998 j. j. j. 1630587323855 56141998 j. j. j. j. 1630587316509 56141998 j. j. j. j. uaint and curious volume of 1 j. j. j. j. j.</td> <td>d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l</td> <td>oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore—</td> <td>sage_data m 1 0 60 1: 4 1 0 1 4 1 g, 0 4 1 g, 0 4 1 y 0 4 1</td> <td>xtra_download_id yy_reaction 4112 0 08543758 0 4226 0 0 no_sp 4096 0 1 0 4226 0 1 0</td> <td>extra_bucket_nam 134217728 0 0 { 0 1024 0 0 1024 0 0 0 0 0 0 0 0 no_sp 0 0 0 0 0 no_sp 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>ne st 1 0 0 0 0 0 0 0 0 0</td> <td>icker_id 1011 Qhnt {} Qhnt 0 Qhnt 0</td> <td>extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 8 0 03h7J3pE= 21 8 0</td> <td>ion body y50= 2073650 2 1535150 1502380</td> <td>descriptio 32 2 0 (2 (2 (</td> <td>on i 2 () () (</td>	rea 1 2 0 2 2	tad_message_time_scroll 5614195571837349295 0 timebo 5514199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 5614199806969078209 0 Over many a q	pos broadcast_msg_it i 630586306677 56141952 pmb/60 0 umb/60 0 0 i 630587305138 56141992 dreary, while I pondered, were i 1630587323855 56141998 j. j. j. 1630587323855 56141998 j. j. j. j. 1630587316509 56141998 j. j. j. j. uaint and curious volume of 1 j. j. j. j. j.	d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l	oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore—	sage_data m 1 0 60 1: 4 1 0 1 4 1 g, 0 4 1 g, 0 4 1 y 0 4 1	xtra_download_id yy_reaction 4112 0 08543758 0 4226 0 0 no_sp 4096 0 1 0 4226 0 1 0	extra_bucket_nam 134217728 0 0 { 0 1024 0 0 1024 0 0 0 0 0 0 0 0 no_sp 0 0 0 0 0 no_sp 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ne st 1 0 0 0 0 0 0 0 0 0	icker_id 1011 Qhnt {} Qhnt 0 Qhnt 0	extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 8 0 03h7J3pE= 21 8 0	ion body y50= 2073650 2 1535150 1502380	descriptio 32 2 0 (2 (2 (on i 2 () () (
2 5614199896425193517 1630587337735 5614199986425193517 0 1 4096 0 0 0 Qhnb3h7J3pE= 211592502 2 0 0 Only this and nothing more." 1 0 no_sp 0 0 0 0 Qhnb3h7J3pE= 211592502 2 0 2 5614199917050575610 1503587342009 561419991705087342009 4 1 4096 0 0 0 Qhnb3h7J3pE= 211608888 2 0 2 5614199917050587342009 561419993066974957 1 0 no_sp 0 0 Qhnb3h7J3pE= 211633466 2 0 2 561419993066974957 1503687340037 56141999393069794957 4 1 4096 0 0 0 Qhnb3h7J3pE= 211633466 2 0 3 And each separate dying ember wought its ghost upon the filtoner. 1 1 0 no_sp 0 <td>rea 1 2 0 2 2 2 2</td> <td>tad_message_time_scroll 5614195571837349295 0 timebr 55114199759762187124 Once upon a midnight 5614199808040482659 0 While I noddee 5514199806562078209 0 Over many a q 5614199857279754766 0 As of some on</td> <td>pos broadcast msg ici 16305686306677 56141955 mb/60 0 0 16305867305138 56141997 dreary, while I pondered, wes 1630587323855 1630587323855 56141999 1 1630587323855 1 1630587323855 1 1630587323855 1 163058732823 1 163058732823 1 163058732823 1 163058732823</td> <td>d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l 57279754 my chaml</td> <td>oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore— 1766 0 per door.</td> <td>sage_data m 1 0 60 1 4 1 0 1 4 1 9 0 4 1 9 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 0 4 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>tra_download_id y_reaction 4112 08543758 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 4096 1 0 4096 1</td> <td>extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>ne st 1 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0</td> <td>extra_dural Qmhp5XZJ x3h7J3pE= 0 0 x3h7J3pE= 21 0 x3h7J3pE= 21 0 x3h7J3pE= 21 0 0 x3h7J3pE= 21 0 0 0 x3h7J3pE= 21</td> <td>ion body /50= 2073650 2 1535150 1502380 1551537</td> <td>descripti 32 2 0 (2 (2 (2 (</td> <td>2 (2 () () () (</td>	rea 1 2 0 2 2 2 2	tad_message_time_scroll 5614195571837349295 0 timebr 55114199759762187124 Once upon a midnight 5614199808040482659 0 While I noddee 5514199806562078209 0 Over many a q 5614199857279754766 0 As of some on	pos broadcast msg ici 16305686306677 56141955 mb/60 0 0 16305867305138 56141997 dreary, while I pondered, wes 1630587323855 1630587323855 56141999 1 1630587323855 1 1630587323855 1 1630587323855 1 163058732823 1 163058732823 1 163058732823 1 163058732823	d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l 57279754 my chaml	oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore— 1766 0 per door.	sage_data m 1 0 60 1 4 1 0 1 4 1 9 0 4 1 9 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 0 4 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0	tra_download_id y_reaction 4112 08543758 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 4096 1 0 4096 1	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ne st 1 0 0 0 0 0 0 0 0 0 0 0 0	icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0	extra_dural Qmhp5XZJ x3h7J3pE= 0 0 x3h7J3pE= 21 0 x3h7J3pE= 21 0 x3h7J3pE= 21 0 0 x3h7J3pE= 21 0 0 0 x3h7J3pE= 21	ion body /50= 2073650 2 1535150 1502380 1551537	descripti 32 2 0 (2 (2 (2 (2 (2 () () () (
0 Only this and nothing more." 1 0 no_sp 0	rea 1 2 0 2 2 2 2	ad message time scroll 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 5614199806969078209 0 Over many a q 56141998752/39754766 0 As of some on 5614199878305800540	pos bradcast msg. ici 1630566306677 56141955 pmb/60 0 0 0 16305687305188 56141997 dreary, while I pondered, wes 16305687323855 56141998 uaint and curious volume of 1 1630567328323 56141998 uaint and curious volume of 1 1630567328323 56141998 e gently rapping, rapping at 1 163056733439 56141998	d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l 57279754 my chaml 78305800	oted_mes 1295 0 (124 0 ary, 1659 0 e a tapping 1209 0 ore	sage_data m 1 0 60 1 4 1 0 1 4 1 9 0 4 1 9 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 0 4 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0	xTra_download_id 4112 4112 08543758 0 4226 0 1 0 4226 1 0 4226 1 0 4226 1 0 4096 1 0 4096 1 0 4096	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0	ne	icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt	extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21	ion body /50= 2073650 2 1535150 1502380 1551537	descripti 32 2 0 (2 (2 (2 (on r 2 () () () () (
2 56141999170/56976510 16305873242009 56141999170/56976510 1 4096 0 0 0 0 Ah, distinctly I remember it was in the bleak December: 1 0 no_sp 0	rea 1 2 0 2 2 2 2 2	ad_message_time_scroll 5614195571837349295 0 timebc 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 56141998065959078209 0 Over manya q 5614199867279754766 0 As of some on 5614199878305600540 0 "Tis some visil	pos broadcast msg ici 1630586306677 56141955 mb/60 0 0 16305867305138 56141997 dreary, while I pondered, wese 1630587323855 56141999 dreary, while I pondered, wese 1630587323855 56141998 dreary, while I bondered, wese 1630587323855 56141998 uaint and curious volume of I 163058732323 56141998 e gently rapping, rapping at I 163058733439 56141998 roc, 'I muttered, 'Tapping at I 163058733439 56141998	d qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l 57279754 my chamb	oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore— 1766 0 ore door. 1540 0 er door.	sage_data m 1 0 60 1 4 1 0 1 4 1 9 0 4 1 9 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 0 4 1 0 0 1 0 0 1 0 0 1 0 1 0 0 1 0 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0	ktra_download_id iy_reaction 08543758 0 0 no_sp 4096 0 sp 4096 0 s 1 0 4096 0 1 0 4096 0 1 0 1 0 1 0 1 0	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0<	ne - st 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	icker_id 1011 Qhnt Qhnt O Qhnt O Qhnt O Qhnt O	extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 03h7J3pE= 21 0 03h7J3pE= 21 0 03h7J3pE= 21 0 03h7J3pE= 21 0 0	ion body v50= 2073650 2 1535150 1502380 1551537 1572020	descripti 32 2 0 () 2 () 2 () 2 () 2 () 2 ()	on () () () () () () () () ()
0 Ah, distinctly Iremember it was in the blask December; 0; 1 0 no_sp 0 0; 0; 1 0 no_sp 0; 0; 0; 1; 0;	rea 1 2 0 2 2 2 2 2	ad message time scroll 5614195571837349295 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 56141998069078209 0 Over mary a q 56141998723075800540 0 As forme on 5614199878305800540 0 "Tis some visi 5614199895425193517	pos broadcast msg. ici 16305686306677 5611455 pmb/60 10 10 1630587306138 56141997 dreany, while 1 pondered, wez 1630587325385 56141998 1630587316509 56141998 uaint and curious volume of 1 16305873233439 56141998 e gently rapping, rapping at 1 1630587333439 56141998 to 7,1 muttered, Tapping at 1 1630587333439 56141998	1 qu 71837349 0 59762187 ak and we 38040482 here cam 06969078 forgotten l 57279754 my chamb 96425193	oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore— 1766 0 ore door. 1540 0 er door.	sage data m 1 0 1 4 1 0 1 4 1 0 1 4 1 3 0 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1	xTra_download_id yreaction 4112 08543758 0 4226 0 4226 0 4226 0 4226 1 4226 1 4226 1 4096 1 4096 1 4096 1 0 4096	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0<	ne - st 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	icker_id 1011 Qhnt Qhnt O Qhnt O Qhnt O Qhnt O	extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 03h7J3pE= 21 0 03h7J3pE= 21 0 03h7J3pE= 21 0 03h7J3pE= 21 0 0	ion body v50= 2073650 2 1535150 1502380 1551537 1572020	descripti 32 2 0 () 2 () 2 () 2 () 2 () 2 ()	2 () () () () () (
2 5614199939869794957 1630587348037 5614199939869794957 0 4 1 4096 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	rea 1 2 2 2 2 2 2 2	tad message time scroll 5614195571837343295 0 timebr 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 5614199857279754766 0 Aso f some on 5614199878305800540 0 "Tis some visil 5614199878305800540 0 Only this and 1 0 Only this and 1	pos bradcast msg ici 1630586306677 56141955 mb/60 0 0 0 1630587305138 56141997 dreary, while I pondered, wei 1630587323855 56141998 di, nearly napping, suddenly t 1630587316509 56141998 uaint and curious volume of 1 163058733232 356141998 de gently rapping, rapping at 1630587337339 56141998 tor," I muttered, Tapping at 1630587337735 56141998	1 qu 71837349 0 759762187 ak and we 38040482 here cam 06969078 forgotten l 157279754 my chamb 178305800 ny chamb 96425133 1	oted_mes 1295 0 124 0 ary, 1659 0 e a tappinq 1209 0 ore	sage_data m 1 0 60 1 4 1 0 1 4 1 3 0 4 1 4 1 4 1 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1	xīta_download_id iy_rēaction 4112 08543758 0 4226 0 4226 1 0 4226 1 0 4226 1 0 4226 1 0 4226 1 0 4096 1 0 4096 1 0 4096 1 0 0	extra_bucket_nam 134217728 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ne	icker_id 1011 Qhnt Qhnt Qhnt Qhnt Qhnt Qhnt Qhnt	extra_dural Qmhp5XZJ 33h7J3pE= 0 0 33h7J3pE= 21 0 33h7J3pE= 21 0 33h7J3pE= 21 0 33h7J3pE= 21 0 33h7J3pE= 21	ion body 2073650 2 1535150 1502380 1551537 1572020 1592502	descripti 32 2 0 (2 (2 (2 (2 (2 (2 (2 (2 (2 () () () () () () (
0 And each separate dying ember wrought its ghost upon the floor. 1 0 no_sp 0	rea 1 2 2 2 2 2 2 2	ind message time scoll 5614195571837349295 561419957962187124 Once upon a midnight 5614199838040482659 0 While I noddec 56141998705762787124 Once upon a midnight 561419987059762107954766 0 While I noddec 561419987279754766 0 Over many a q 5614199876305800540 0 Tis some viai 5614199876305800540 0 Tis some viai 561419987630580540 0 Tis some viai 561419987630580540 0 Officiant some viai 561419987630580540 0 Tis some viai 561419987630580540 0 Officiant some viai 561419987630580540 Officiant some viai 5614199876426793517 O 0 Only this and 1 5614199876510 Some some some some some some some some s	pos bradcast msg. ici 163056806677 5611455 mb/60 1630587305138 56141997 dreary, while pondered, we: 1630587325385 56141998 1630587315699 56141998 uaint and curious volume of i 1630587333439 56141998 cor, il muttered, tapping at m 163058737325 56141998 totsing more. 163058737325 56141998 163058737355 5614998 163058737355 5614998 163058737355 5614998 163058737355 5614998 163058737325 5614998 1630587373255 5614998 1630587373255 5614998 1630587373255 5614998 1630587373255 5614998 1630587373255 5614998 163058737255 5614998 163058737255 5614998 16305873732609 5614998 16305873732609 5614998 163058737342609 5614998 163058737342609 5614998 1630587342609 5614998 1630587342609 5614998 1630587342609 5614998 1630587342609 5614998 1630587342609 5614998 1630587342609 5614989 1630587342609 5614988 1630587342609 5614989 1630587342609 5614989 163058734509 5614989 163058734509 5614989 163058734509 5614989 163058734509 5614989 163058734509 5614989 163058734509 5614989 163058734509 5614989 163058734509 163058734509 163058734509 163058734509 163058734509 163058734509 163058734509 163058734509 163058	d qu 71837345 0 59762187 ak and wea 38040482 here cam 106969078 forgotten l 105969078 forgotten l 1057297754 my chamb 196425193 1117056975	oted_mes 1295 0 124 0 ary, 1659 0 e a tappiny 1209 0 ore- 1566 0 ber door. 1540 0 er door. 1517 0 0 0 0 0 0 0 0 0 0 0 0 0 0	sage_data m 1 0 60 1 4 1 0 1 4 1 3 0 4 1 4 1 4 1 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1	xTra_download_id yreaction 4112 08543758 0 4226 0 4296 10 4226 10 4226 10 4096 1 4096 1 4096 4096 4096 4096 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	extra_bucket_nam 134217728 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	ne - st 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	icker_id 1011 } Qhnt 0 Qhnt 0 Qhnt 0 Qhnt Qhnt Qhnt	extra_dural Qmhp5XZJ 33h7J3pE= 0 0 33h7J3pE= 21 0 33h7J3pE= 21 0 33h7J3pE= 21 0 33h7J3pE= 21 0 33h7J3pE= 21	ion body 2073650 2 1535150 1502380 1551537 1572020 1592502	descripti 32 2 0 (2 (2 (2 (2 (2 (2 (2 (2 (2 () () () () () () (
2 561419996816115445 1630587352425 5614199958161154465 0 4 1 4096 0 0 0 0 0 Qhnb3h7J3pE= 211649852 2 0 0 Eagerly livished the morrow;—vainly lad sought to borrow { 1 0 no_sp 0 0 0 0 0 0 0 0 0 2 0 0 0 0 0 0 0 0 0	rea 1 2 2 2 2 2 2 2 2 2 2	ad_message_time_scroll 5614195571837349295 0 5614199759762187124 Once upon a midnight 561419983040482659 0 While I noddec 5614199853069078209 0 While I noddec 5614199853054069078209 0 Oer many a q 561419987305800540 0 As of some on 561419987305800540 0 "Tis some visit 5614199896425193517 0 Ohyt this and r 5614199917056975610 0 Ah distinctly I	pos bradcast msg. ici 1630586306677 56141955 bmb/60 0 0 16305867305138 56141997 dreary, while 1 pondered, wes 1630587323855 56141998 uaint and curious volume of 1 163058732523 56141998 uaint and curious volume of 1 163058732533 56141998 to 1630587337373 56141998 to 153058733735 56141998 to 153058733735 56141998 163058733735 56141998 1630587342609 56141999 1630587342609 56141999	d qu 71837345 0 59762187 ak and wea 38040482 here cami 06969078 forgotten li 57279754 forgotten li 57279754 forgotten li 57279754 forgotten li 57279754 horgotten li 57279754 horgot	oted_mes 1295 0 124 0 ary, 1659 0 e a tapping 1209 0 ore- 1766 0 ore- 1766 0 ore- 1766 0 ore- 1766 0 ore- 1766 0 ore- 1766 0 ore- 1770 0 0 0 0 0 0 0 0 0 0 0 0 0 0	sage_data m 1 0 60 1 4 1 0 1 4 1 3 0 4 1 4 1 4 1 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1	xTra_download_id 4112 4112 08543758 0 4226 0 4226 1 0 4226 1 0 4226 1 0 4096 1 0 4096 1 0 4096 1 0 4096 1 0 4096 0 0 0 0 0	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0<	ne - st 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	icker_id 1011 Qhnt Qhnt Qhnt Qhnt Qhnt Qhnt Qhnt Qhnt	extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 0	ion body 2 2 1535150 1502380 1551537 1572020 1592502 1608888	descripti 32 2 0 () 2 () 2 () 2 () 2 () 2 () 2 () 2 () 2	on - 1 2 (1) (1) (1) (1) (1) (1) (1) (1)
2 5614199978285424787 1630587357211 5614199978285424787 0 4 1 4096 0 0 0 1008 Qhnb3h7J3pE= 211670334 2 0	rea 1 2 2 2 2 2 2 2 2 2 2	ad message time scoul 5614195571837349295 0 timebo 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 5614199806969078209 0 Over many a q 5614199878305800540 0 As of some on 5614199878305800540 0 "Tis some visi 5614199896425193517 0 Only this and 5614199917056975610 0 Ah distinctly 1 5614199917056975610	pos broadcast msg. ici 1630566306677 56141955 instructure instructure instructure 16305687305138 56141957 instructure instructure <td>d qu 71837345 0 59762187 ak and we 38040482 here came 106969078 forgotten l 157279754 my chamb 196425193 117056975 k Deceml 139869794 s ghost up</td> <td>oted_mes 1295 0 124 0 ary, 12659 0 e a tappiny 1209 0 ore 1540 0 or door 1517 0 0 1517 0 0 0 1517 0 0 0 0 0 0 1517 0 0 0 0 1517 0 0 0 0 0 1517 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>sage data m 1 0 60 1 4 1 0 1 4 1 4 1 5, 0 4 1 4 1 4 1 0 4 1 4 1 0 4 1 4 1 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1</td> <td>xTra_download_id yreaction 4112 08543758 0 4226 0 4226 1 0 4226 1 4226 1 4096 1 4096 1 4096 1 0 4096 0 0 4096 0 0 0 0 1 0 4096 0 0 0 0 0 0 0 0 0 0 0 1</td> <td>extra_bucket_nam 134217728 0 0 0 1024 0 0 0</td> <td>ne - st 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 0 Qhnt 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td> <td>extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 0</td> <td>ion body 2 2 1535150 1502380 1551537 1572020 1592502 1608888</td> <td>descripti 32 2 0 () 2 () 2 () 2 () 2 () 2 () 2 () 2 () 2</td> <td>on 7 1 2 () 0 () 0 () 0 () 0 () 0 () 0 ()</td>	d qu 71837345 0 59762187 ak and we 38040482 here came 106969078 forgotten l 157279754 my chamb 196425193 117056975 k Deceml 139869794 s ghost up	oted_mes 1295 0 124 0 ary, 12659 0 e a tappiny 1209 0 ore 1540 0 or door 1517 0 0 1517 0 0 0 1517 0 0 0 0 0 0 1517 0 0 0 0 1517 0 0 0 0 0 1517 0 0 0 0 0 0 0 0 0 0 0 0 0 0	sage data m 1 0 60 1 4 1 0 1 4 1 4 1 5, 0 4 1 4 1 4 1 0 4 1 4 1 0 4 1 4 1 4 1 0 4 1 0 4 1 0 4 1 0 4 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 1 0 0 1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1	xTra_download_id yreaction 4112 08543758 0 4226 0 4226 1 0 4226 1 4226 1 4096 1 4096 1 4096 1 0 4096 0 0 4096 0 0 0 0 1 0 4096 0 0 0 0 0 0 0 0 0 0 0 1	extra_bucket_nam 134217728 0 0 0 1024 0 0 0	ne - st 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 0 Qhnt 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	extra_dural Qmhp5XZJ 03h7J3pE= 0 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 03h7J3pE= 21 0	ion body 2 2 1535150 1502380 1551537 1572020 1592502 1608888	descripti 32 2 0 () 2 () 2 () 2 () 2 () 2 () 2 () 2 () 2	on 7 1 2 () 0 () 0 () 0 () 0 () 0 () 0 ()
	rea 1 2 2 2 2 2 2 2 2 2 2 2 2 2	tad message time scroll 5614195571837343295 0 timebr 5614199759762187124 Once upon a midnight 5614199838040482659 0 While I noddec 561419985729754766 0 As of some on 561419985729754766 0 As of some on 5614199878305800540 0 "Tis some visil 56141998783058425133517 0 Only this and 1 56141999170569756110 0 Ah, distinctly 1 5614199939869794957 0 And each sepp 5614199958161154465	pos broadcast msg. ici 1630586306677 56141955 mb/60 0 0 16305867305138 56141997 dreary, while I pondered, wesi 163058732585 1630587315609 56141998 uanit and curious volume of 1 163058733623 16305873373439 56141998 tor, 'I muttered, tapping at 1 163058733733 163058733735 56141998 tor,'' I muttered, tapping at 1 163058733735 163058733735 56141998 tor,'' I muttered, tapping at 1 163058733735 1630587337439 56141998 tor,'' I muttered, tapping at 1 16305873724205 1630587324209 56141998 tor,'' I muttered, tapping at 1 163058734209 163058734209 56141999 tor,'' I muttered, tapping at 1 163058734202 163058734202 56141999 tor,'' I muttered, tapping at 1 16305873422	qu qu 71837345 0 '59762187 0 '59762187 0 '38040482 0 '38040482 0 '060690778 0 '060690778 0 '178305800 1 '17056975 k '1717056975 k '170569794 5 '39869794 s '58161154 58161154	oted_mes 1295_0 1224_0 ary, 1659_0 e a tapping 1209_0 ore 1540_0 1540_0 1547_0 0 1610_000000000000000	sage_data m 1 0 60 11 4 1 0 1 4 1 4 1 4 1 4 1 4 1 4 1 4 1 4	xīta_download_id iy_reaction 4112 08543758 0 4226 1 0 4226 1 0 4226 1 0 4296 1 0 4096 1 0 4096 1 0 4096 0 4096 1 0 4096 1 0 4096 1 0 4096 1 0 4096 1 0 4096 1 0 4096 1 0 4096 1 0 4096 4096 <td< td=""><td>extra_bucket_nam 134217728 0 0 0 1024 0 0 0</td><td>ne</td><td>icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 0 Qhnt 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td><td>extra_durat Qmhp5XZJ 33h7J3pE= 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 0 3h7J3pE= 21 0 0 3h7J3pE= 21</td><td>ion body 50= 2073650 2 1535150 1502380 1551537 1572020 1592502 1608888 1633466</td><td>descriptii 32 2 0 () 2 () 2 () 2 () 2 () 2 () 2 () 2 () 2</td><td>on - r 2 (2) (2) (2) (2) (2) (3) (2) (3) (2) (4) (2) (2) (4) (2) (2) (4) (2) (2) (4) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2</td></td<>	extra_bucket_nam 134217728 0 0 0 1024 0 0 0	ne	icker_id 1011 Qhnt Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 Qhnt 0 0 Qhnt 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	extra_durat Qmhp5XZJ 33h7J3pE= 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 3h7J3pE= 21 0 0 3h7J3pE= 21 0 0 3h7J3pE= 21	ion body 50= 2073650 2 1535150 1502380 1551537 1572020 1592502 1608888 1633466	descriptii 32 2 0 () 2 () 2 () 2 () 2 () 2 () 2 () 2 () 2	on - r 2 (2) (2) (2) (2) (2) (3) (2) (3) (2) (4) (2) (2) (4) (2) (2) (4) (2) (2) (4) (2) (2) (2) (2) (2) (2) (2) (2) (2) (2
	rea 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	ad_message_time_scroll 5614195571837349295 56141995976187124 Once upon a midnight 5614199838040482659 0 While I noddec 5614199806950978209 0 Over many a q 5614199878305600540 0 As o some on 5614199857279754766 0 As of some on 5614199856425193517 0 Ohly this and 1 5614199917056975610 0 Ah distinctly 1 56141999398669734957 0 And each seps 56141999398151154456 0 Eagerly I wish	pos broadcast msg it 1630566306677 56114957 56114957 omb/60 8 6141997 dreary, while I pondered, west 1630567323855 56141997 nearly apping, suddenly t 1630567323855 56141998 unit and curious volume of t 1630587316509 56141998 unit and curious volume of t 1630587328323 56141998 tof, instructured, "tapping at n 16305873743037 56141999 rothing more." (0) 1630587342609 56141999 rothing more." (1) 1630587342609 56141999 rate dying ember twas in the blea 1630587342609 56141999 rated dying ember twas unit height 1630587342637 56141999	y qu 71837345 0 0 59762187 ak and weat 38040482 138040482 138040482 138040482 157279754 157279754 157279754 15729754 117056975 k Decemi 139869794 s ghost up 158161154 sought to 580641154	oted_mes 1295_0 1224_0 ary, 1659_0 e a tappin(1209_0 ore— 1766_0 ber door. 1540_0 er door. 1547_0 517_0 0 16610_0 ber; 1957_0 pon the flow 1465 borrow	sage data r T 0 60 1 4 1 9 0 4 1 4 1 4 1 4 1 4 1 4 1 0 4 1 0 4 1 0 1 0 1 0 1 0 1 0 1 0 1	xTra_download_id yreaction 4112 08543758 0 4226 0 4096 1 0 4226 0 4096 1 0 4096 1 4096 0 4096 0 4096 0 0 4096 0 0 4096 0 4096 0 0 4096 0 0 1 0 4096 1 0 1 1 1 1	extra_bucket_nam 134217728 0 0 0 0 1024 0 0 0 0 0	ne	icker_id 1011 Qhnt Q Qhnt Q Qhnt Qhnt Qhnt Qhnt Qhnt	extra_durat Qmhp5XZJ 3h7J3pE= 0 0 3h7J3pE= 21 0 3h7J3pE= 21 0 0 0 0 0 0 0 0 0 0 0 0 0	ion body ,60= 2073650 2 1535150 1502380 1551537 1572020 1592502 1608888 1633466 1649852	descriptii 32 2 0 () 2 () 2 () 2 () 2 () 2 () 2 () 2 () 2	pn r 2 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

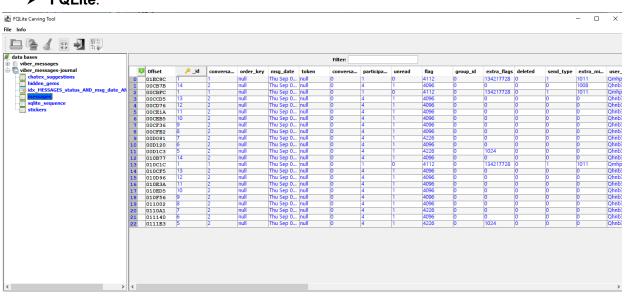
> X-ways Forensics.

Figure 24. X-ways Forensics user interface (database: "viber_messages.db").

> SQLite Forensic Explorer.







> FQLite.

Figure 26. FQLite user interface (database: "viber_messages.db").

> Andriller.

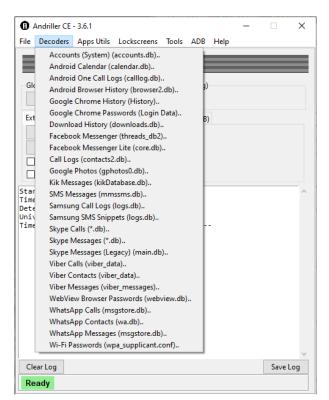


Figure 27. Andriller user interface (default decoders for databases).

> Forensic Toolkit for SQLite.

101(1	Main												
 PELDS tradh_ORIGDF tradh_ORIGDF tradh_ORIGDF beletion Time Da tradh_MEDIAT tradh_WEDIAT tradh_WEDIAT tradh_WEDIAT tradh_RESTOR FROM tradh 		DI BURSTGROUPID INT BESTIMACE INT COLODERVERID TXT COLODERVERID TXT RESTOREDITA TXT VOLMENALD INT											
5	Outpu	ut Expression	Alias	Sort Type	Sort Order	Appreciate	Grouping	Criteria	lor	lor	lor	1	~
1				2.5								8	
	a 🗹												
			'u Deletion Time										v .
steTime(trashD) rashABSPATH, rashMEDIATYPE, rashWIDIH, rashHEIGHT, rashRESTOREEXTI		Z / 1000, 'unimepoch') AS "De	letion Time",										
ateTime(trashD) rashASSPATH, rashMEDIATYPE, rashWIDTH, rashHEIGHT, rashREIGHT, rashREIGHT, fash		Σ / 1000, 'unixepoch') λS "De	letion Time",										
resh_ORIGHTITI herdine (trash_DU resh_ABSPATH, resh_MEDIATUPE, resh_MIDIATUPE, resh_HEIGHT, resh_HEIGHT, resh MICHT, Rows = 4 UGNPATH		E / 1000, 'unisspoch') AS "De	_ORIGDWITLE			Deleton Tim		BSPATH			MEDIATYPE	_WIDTH	-Heisu Teedoresian
ateTime (trashB) rashABSPATH, rashHOIATYPE, rashHOIATYPE, rashHOIATYPE, rashHOIATY rashBESTOREEXTI H trash Rows = 4 JGDPATH	23.	Z / 1000, 'uninepoch') KS *De 3e7c3/3ef9c300ee43890ba3kef66-4.pp	_ORIGD/TITLE	c300ee42d9d0a	a 36efdd-V. ju				(/Android/-Trash/com.sec	android.gallery3d/4495019437814		WIDTH	("s350video")fabes,"s07m"fabes,"s0Favourite")fabes,"fleDuration"\0,"recordingMode"\0 2244; "afflieS.dsTipe="0,"setFileType="0,"setFileS355355,"group_type="0,"setFileS357356 "abbude="10,"ongtude="0,"mmeType="1 mage/type="0,"ecordingType="0,"setFileS35736 "doubdhorgnaf3set="0,"doubdhorwing="1,"_edaveFaller")"?icturesVideoV', "orign: file_has "doubdhorgnaf3set="0,"doubdhorwing="1,"_edaveFaller")"?icturesVideoV', "orign: file_has
ateline israshBSATH, reshASSATH, reshMIDIATUPS, reshWIDIATUPS, reshWIGHT, reshMIGHT, reshMIGHT, Rows = 4 IGDWATH gg/emulated/0/Pictures/	93. Vber/D4G-f3		_ORIGD/ITTLE			pg 2021-10-31	11:41:59 /sto	rage/emulated/0			816045 1		("_s3600ideo":faise,"_s0rm"faise,"_s5Favourite":faise,"_fieDuration"10,"_recordingMode":0 3240; efficieSubType"10, _sefficieType"10, _ssef "9315355,"_group_type"10, _deaTaken":16356758 381046":0; and another the time Type "Theory Uncol, "recordingType"10, _deaTaken":16356758
tod'ier (trash) ashASDFATH, ashSDIATYPE, teahSUIATYPE, ashSUIATYPE, ashSUIATYPE, ashSUIATYPE, SUIATYPE,	23, Viber /DHG-F3 Viber /DHG-F3	Se hafbefördborerläfelna Stefför - Jap	_ORIGD/ITTLE	c300ee42d9d0a		pg 2021-10-31	11:41:59 /sto	rage/emulated/0 rage/emulated/0	(Android). Trash/com.sec		1816045 1 1816045 1	5760	[

Figure 28. Forensic Toolkit for SQLite user interface.

Chapter 3.

Android Incident Response Scenario

This Thesis focused on DFIR actions and their consequences. This was achieved by creating and executing a possible realistic scenario. In this chapter are referenced the retained scenario, the desired goals of the experiments, the methodology and the procedure on which the choice of instant messaging applications was based. General information about the Viber, WhatsApp and Telegram applications and their communication encryption are mentioned. What is more, the location of the applications communication servers for Greece (Athens) territory is presented via an IDS solution implementation (ntopng with geoip plugin) and the difficulty to retrieve content from the companies' cloud servers based on an FBI document.

3.1. The scenario

Some organizations contain and manage much, sensitive, information for example, nuclear power plant or a military intelligence facility. A person inside the organization facilities wants to send a classified piece of information which saw on a desk or on a desktop. This person could be either an insider (Industrial espionage) or a random visitor. So, this individual uses a mobile phone and via a social chatting application sends a message (saved or non-saved-contact) with the intel-photo included. After having sent the message, the in-question chat thread is deleted by the user, before the incident response team has the chance to spot and seize the mobile device. The device is either a LG G6 or a Samsung A50 which were detailed in the previous chapter. In these scenarios both hypothetical situations were included where the shared images were captured directly using the chat application or/and the device's default camera application.

3.2. Experiments Goals

The volatility or the persistence of evidence is a crucial subject in forensics. Specially in mobile forensics one common question which often arises is related to the handling of the device after its seizure and the actions that must be taken by the IFR. For this purpose, a simple scenario was developed, which is not far from reality. In this scenario an employee (insider), or a random visitor, of an organization/industry using a smartphone's instant messaging application, takes pictures containing classified information from a screen of an unlocked workstation and sends them to a third party alongside some text messages. After that, the suspect immediately deletes the in-question chat thread. These actions come to security team's attention, who apprehends the suspect and seizes the device.

Three key aspects regarding the seized device should be examined to determine the best course of actions from an IFR perspective for the retrieval of the incriminating messages and the receiver's identity:

- The time that has passed from the seizure of the device and until its forensic acquisition
- The status of the device

- The interaction with the messaging application either by the user before the seizure or by the IFR after it.

The main goal of these experiments is to determine under which conditions and actions the in-question data are preserved.

3.3. Messaging Application Choice

For the scenario two devices (LG G6 and Samsung A50), running a different Android OS Version, were prepared according to the procedures described in **Chapter 3** of this Thesis. Different sim cards were activated. Nine of the most popular messaging application were installed using the same version of apk (through "adb -d install" command), a new user was added, and chat messages were exchanged. In **Table 5** is presented the popularity of chosen applications, as it is referenced on the official's digital distributor site and Android Market (Google Play Store). Furthermore, no additional applications were installed (e.g., google accounts) except the underexamination ones to mitigate the level of "noise" and unrelated data.

Application	Total Downloads
Facebook Messenger	5B+
WhatsApp	5B+
Viber	1B+
Telegram	1B+
Skype	1B+
LINE	500M+
imo	500M+
WeChat	100M+
kik	100M+
Signal	50M+

Table 5. Chat applications' total downloads (Source: Google Play Store).

All the user data including contacts, messages and call logs are stored in SQLite databases inside the applications' folders in the device's filesystem. As it was analyzed in paragraph §1.3.1.2 the key factor for recovering deleted records (e.g., deleted messages in the scenario's case) depends on the settings and features of these databases. With the usage of forensic tools, described in paragraph §1.4, those nine applications' databases were compared. The forensically interesting settings of these applications are presented in Table 6.

Application	Version	Journaling Mode	Encryption	Auto Vacuum mode	Messages Databases
Facebook Messenger	339.0.0.18.118	Journal	No	Full	com.facebook.orca/databases/threads.db2
imo	2021.11.2051	WAL	No	Full	com.imo.android.imoim/databases/imofriends_xxxxx.db
kik	15.38.1.25235	WAL	No	Full	kik.android/databases/xxxxxx.kikDatabase.db

LINE	11.19.1	Journal	No	Full	jp.naver.line.android/databases/naver_line
Signal	5.26.11	WAL	Yes	Unknown	org.thoughtcrime.securesms/databases/signal.db
Skype	8.78.0.164	WAL	No	Full	com.skype.raider/databases/s4I-live:cid.xxxxx.db
Telegram	7.9.3	WAL	No	None	org.telegram.messenger/files/cache4.db
Viber	16.4.0.8	Journal	No	Full	com.viber.voip/databases/"viber_messages.db"
WeChat	8.0.16	WAL	Yes	None	com.tencent.mm/files/Micromsg/xxxxxx/EnMicroMsg.db
WhatsApp	2.21.22.27	WAL	No	Full	com.whatsapp/databases/"msgstore.db"

Table 6. Chat application databases' characteristics (sort alphabetically).

From the above nine applications, three of them were chosen based on the differences of their databases' settings such as the type of journaling they use (rollback journal/wal). Those apps are Viber, WhatsApp and Telegram, which have also been correlated in the past as all together by Botha et al. (2019) and Sutikno et al. (2016), or with other instant messaging applications (Onovakpuri P., 2018). Even though the encryption of the WeChat app was bypassed (Rathi et al., 2018) its examination would not add any further value, since by itself alone does not affect under which conditions the database is modified.

The above methodology is depicted in **Figure 29**.



Figure 29. Workflow of the applications' choice.

3.4. Followed Methodology

Before the main scenario's experiments there was a reconnaissance phase involving:

- The understanding of the folder structure, file size and parameters of the databases inside the application after a user has been added. This was achieved by initializing this new user, removing the application, and repeating the process several times.
- The analysis of the interested databases regarding to their tables and columns.
- The understanding of the behavior of the database after examining a single record (text message) in several states (not sent, sent, not delivered, delivered, received etc.) and pinpointing the modified columns.
- The identification of the affected files and databases in the operating system besides the ones residing in the applications' folders when a media or a text message is exchanged.

The main phase of the scenario's experiments involved the examination of the two devices regarding the three key factors, which were previously mentioned:

- The time that has elapsed until and after the device's seizure.
- The status of the device before and after the seizure.
- The application's database modification by sending, receiving, reading or deleting additional messages during the above-mentioned time.

There is a correlation among these three key factors. For example, the longer the device is connected to a network the more likely an application's database modification might occur. The media files (pictures) and the messages which were sent/received and must be retrieved are mentioned in Appendix A2 and are exactly the same for all the experiments. For the creation of a realistic populated database, the exchange of those messages occurred among the exchange of other, unrelated, text messages between four (4) active users as it depicted in Appendix A1. On some occasions (especially in Telegram) the number of the messages of Appendix A1 were not enough to draw any definite conclusions. In this case there was an exchange among the four (4) users of more messages in the form "This is message X", where X represents a sequential numbering (i.e., "This is message 200"). The experimentation phases had to be adjusted according to the unique features and characteristics of each tested application. For example, some of the messages of Appendix A2 in the Viber application included "autodelete" feature (timebomb), whereas in WhatsApp application this feature was ignored since autodeletion feature was activated after 24-hours. Apart of that, forensic images of the devices were acquired in several time frames and in devices' different statuses and conditions to determine if any modification of the databases can occur based solely on these two factors. The main time frames and under examination conditions are represented in Table 7.

Status (Always after deleting the messages of Appendix A2)	Time Frame	Action
Standby	As soon as possible	None
Standby	After 2/6/24 hours	None
Standby	As soon as possible	Removing Sim
Airplane mode	After 2/6/24 hours	None
After reboot	24 hours	Removed Sim
After shutdown	48 hours	None
Faraday box	After 2/6/24 hours	None

Table 7. Chat application databases' main characteristics (sort alphabetically).

After the scenario's main phase was completed, more experiments were conducted, which were unique for each tested application regarding some of their distinctive features. For example, in the Viber application the behavior of the journal file was examined regarding its size and correlation with the number of active users. In WhatsApp application was examined the behavior of the database after a local backup has been created and the ability to decrypt it. The above-mentioned methodology was organized into a workflow consisted of distinct phases (**Figure 30**). The experiments that were conducted were identical for both the devices but differed based on the examined application. Their total number for both the devices is presented on **Table 8**.

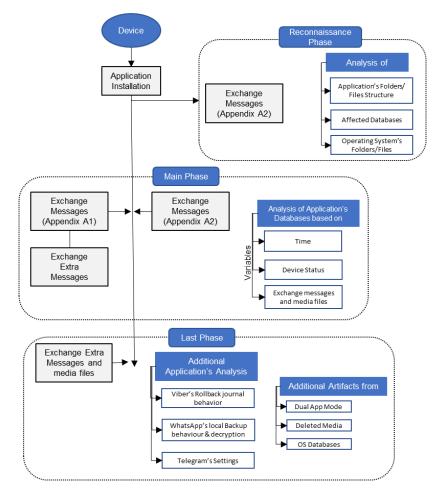


Figure 30. Workflow of methodology.

	Physical	Acquiring I	Exchanged			
Application	Images			Last Phase	Messages	
Viber	48	10	164	30	336	
WhatsApp	12	6	168	-	660	
Telegram	12	6	180	-	2.200	

Table 8. Total conducted experiments' number.

3.5. General information about Viber, WhatsApp & Telegram

3.5.1. Viber

3.5.1.1 General Information

It was founded by Israelian Talmon Marco back in 2010 and joined Google Play Store two years after, in 2012. In 2014 Viber was bought by the Japanese company Rakuten and in 2016 the end-to-end encryption as a default setting was implemented. Viber has implemented the feature of automatically disappearing messages after a brief period of time that they have been

read by the receiver. This period can vary from 10 seconds to 24 hours. Viber application can be installed in an Android or an iOS device and only after this initial setup the user can be install a version of it on a personal computer running Windows, Mac, or Linux operating system. According to Viber statistics (99Content, 2021) nowadays Viber market has been expanded to over 190 countries and in March 2020 there were almost 1.17 billion registered users worldwide. Among the countries in which Viber has significant popularity are Ukraine, Greece, Belarus, Russia, and Philippines.

3.5.1.2 Viber's Communication Encryption

The Viber app provides an end-to-end encryption, including calls, one-on-one or group messages and media sharing. Encryption keys are stored only on the communicating "clients", and no one can access them, including Viber servers.

For exchanging messages for example, as official Viber's site refers (Rakuten Viber, 2021): Every time a device A sends a message to a device B, an ephemeral one-time 128-bit symmetric key is being generated and used for the encryption of the message body, using the Salsa20 encryption algorithm. «Then the ephemeral message key is encrypted using each recipient's session key. The sender device sends a unified message to the server, containing one encrypted cyphertext and a set of encrypted ephemeral keys. A server-side fan-out slices this message and delivers the relevant parts to each target device. The two devices take turns at advancing the session keys in a process called ratcheting. Each time the direction of the conversation changes, the device whose turn it is randomly generates a new Ratchet key pair, and once again performs the following sequence: TempKey = HMAC SHA256(RootKey, DH (RatchetA, RatchetB)) New RootKey = HMAC_SHA256(TempKey, "root") SessionKey = HMAC_SHA256(TempKey, "mesg") With Ratchetthis device being the private part of the newly derived key-pair. Alongside each message, the public part of the Ratchetthis device is also sent. The recipient runs DH with its last private ratchet together with the sender's public ratchet» (Rakuten Viber, 2021). By implementing this double-ratchet method there is a forward and backwards secrecy of and even if there is a compromise of keys, previous or/and future messages cannot be decrypted. Moreover, peer's authentication is being maintained by the algorithm, because the root keys' DH (Diffie-Hellman) chain began with both devices' ID Keys. So, the entire chain is considered as trusted, if the peer's ID key confirmed as trusted at any point. Calls are being encrypted through an ephemeral 256-bit Curve25519 key-pair and the conversion of the RTP stream to SRTP via Salsa20 algorithm. Media sharing encryption uses an ephemeral, symmetric Salsa20 key, an HMAC (Hash-based message authentication code) signature and an MD5 signature for the file encryption in the Viber's storage server.

3.5.2. WhatsApp

3.5.2.1 General Information

It was founded, by Brian Acton and Jan Koum, former employees of Yahoo!, in 2009. In 2013 Google made an offer of ten (10\$) billion US dollars, but the offer was rejected. In 2014 it was acquired by Facebook, currently known as Meta Platforms. The acquisition has risen some privacy concerns among the users causing them to migrate at different messaging platforms. WhatsApp has a different approach to implementing the feature of automatically disappearing messages where there is a longer lasting period of time for these messages. This period of time

can be either 24 hours,7 days or 90 days. WhatsApp is a cross platform app and can be installed and used on Android or iOS mobile devices and on Windows PC, Mac or Linux desktop or laptop. According to WhatsApp statistics (99Content, 2021) in India the application has over half a billion users. Overall, there more than two billion users accessing the application every month. In several countries including China, the United Arab Emirates, Iran, Syria, North Korea and Cuba the usage of the application is blocked or restricted. The application has often been accused, especially in India, for the spread of misinformation, propaganda and fake news. For this reason, WhatsApp in 2020 forced a limit of chat forwarding (Singh S., 2020).

3.5.2.2 WhatsApp's Communication Encryption

WhatsApp features an end-to-end encryption to all types of communications. The term "end-to-end encryption" is defined by app's programmers (WhatsApp, 2021) «as communications that remain encrypted from a device controlled by the sender to one controlled by the recipient, where no third parties (not directly participants), not even WhatsApp or parent company Facebook, can access the content in between». As official WhatsApp's site refers for exchanging and transmitting messages, media files, calls etc., when a session is established, the content is encrypted because "clients" exchange messages with a Message Key using AES256 in CBC mode for encryption and HMAC-SHA256 for authentication. The Message Key for each message is an ephemeral and the one-time key, is derived from a sender's Chain Key that "ratchets" forward with every sent message. In addition, a new Chain Key is created by a new ECDH agreement which performed with each message roundtrip. The combination of both an immediate "hash ratchet" and a round trip "DH ratchet" provides more secrecy. Another step forward to security, is that all communications between application's clients and servers use Noise Pipes with Curve25519, AES-GCM, and SHA256 from the Noise Protocol Framework for long running interactive connections. Thus, there are a superfast connection setup and resume, a metadata encryption, and no private authentication keys/credentials storing on servers (only public authentication keys).

3.5.3. Telegram

3.5.3.1 General Information

It was founded by the Russian brothers Pavel and Nikolai Durov, who had already lunched an online Russian social media and networking service called "VK" (V Kontakte). It is a cloudbased messaging app which was launched in 2013. The main characteristics of the application are the open-source code and the ability to create secret, end-to-end encrypted and selfdestructing messages and chats (Telegram, 2021). Today, Telegram is available in 155 countries with more than 550 million active users per month. The country with the most installations (over 210 million) is India, followed by Russia and Indonesia (Dean B., 2021). Telegram can be installed on every platform such as Android, iOS, Windows PC, Mac or Linux.

3.5.3.2 Telegram's Communication Encryption

Unlike Viber and WhatsApp, Telegram does not enable, by default, end-to-end encryption. Once a chat message arrives at the Telegram servers, it is encrypted using MTProto (v2), while at rest on the servers. However, Telegram could read chat data since it handles encryption and decryption of the messages at its (Telegram) servers. E2E encryption is supported only if the secret chat feature is chosen. MTProto is a custom mobile protocol designed by Telegram and it has been criticized many times by cryptography experts (Saribekyan & Margvelashvili, 2017, Lee et al., 2017, Jakobsen & Orlandi, 2016). MTProto (v2) uses Diffie-Hellman (DH) key exchange, Secure Hash Algorithm 256 (SHA-256), Key Derivation Function (KDF), and AES-256 in IGE mode as cryptographic primitive (End-to-End Encryption, Secret Chats) (telegram.org). Even if it is differentiated from the first version with features like extra padding bytes in the computation of msg_key, researchers from the University of London and ETH Zurich have documented significant problems in terms of Telegram's current encryption system security (Albrecht et al., 2021).

3.5.3.3 Telegram's Privacy Concerns

The by default lack of E2E encryption has risen several privacy concerns. According to app's privacy policy it is possible to «...collect metadata such as your IP address, devices, and Telegram apps you have used, history of username changes, etc.». This in conjunction with the fact that the messages are being encrypted on Telegram servers raise some red flags regarding users' privacy.

3.6. Applications' Cloud Servers

Obtaining data such as account information or exchanged messages, from messaging applications' private company servers is a difficult and complex process. The two main reasons are:

- 1. The locations of both servers (**Table 8**) and companies where different legislation is implemented. To put this in perspective:
 - Viber is owned by a Japanese company, based in Luxembourg. The messages originating from Greece were relayed/routed from servers in U.S.A.
 - WhatsApp is owned by an American company. The messages originating from Greece were relayed/routed from servers in Germany.
 - Telegram is registered as an American LLC and UK LLP, with headquarters in Dubai. The messages originating from Greece were relayed/routed from servers in England.

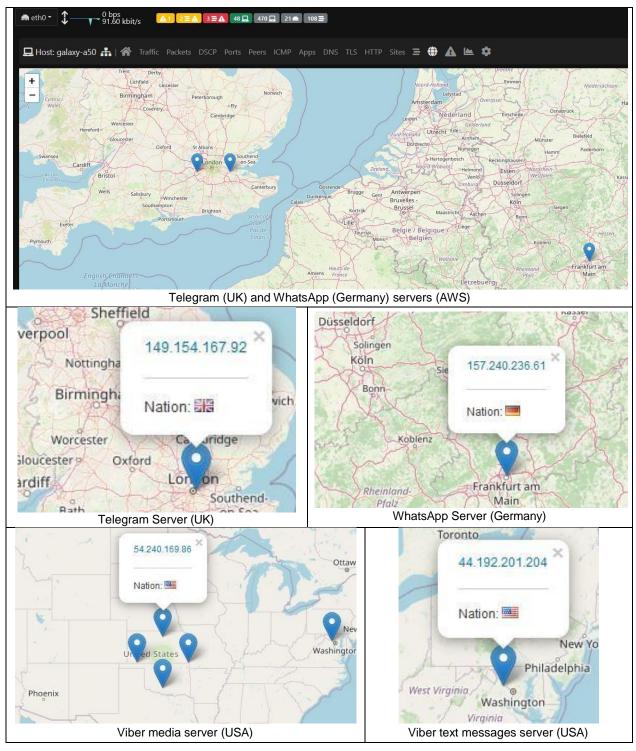


Table 9. WhatsApp, Telegram and Viber applications' cloud servers where the devices connected (tool used: ntopng with geoip plugin, device examined: Samsung A50).

2. Even if this Kafkaesque legal system is bypassed and a court order is executed, as an FBI training document indicates, there is little that could be done to gain access to the content of encrypted messages from secure messaging services (Cimpanu C., 2021). In Table 9 is presented the contents of the above-mentioned document for the messaging applications that

are examined in this Thesis concerning the data that can be retrieved from the company's servers.

Арр	Legal process & additional details
Viber	* No message content.
	* Provides account (i.e., phone number) registration data and IP address at time of creation.
	* Message history: time, date, source number, and destination number.
WhatsApp	* Message content limited.
	* Subpoena: can render basic subscriber records.
	* Court order: Subpoena return as well as information like blocked users.
	* Search warrant: Provides address book contacts and WhatsApp users who have the target in their address book contacts.
	* Pen register: Sent every 15 minutes, provides source and destination for each message.
	* If target is using an iPhone and iCloud backups enabled, iCloud returns may contain WhatsApp data, to include message content.
Telegram	* No message content.
	* No contact information provided for law enforcement to pursue a court order. As per Telegram's privacy statement, for confirmed terrorist investigations, Telegram may disclose IP and phone number to relevant authorities.

Table 10. FBI' ability legally content access of Viber, WhatsApp and Telegram. Content of FBI's training document released on Jun. 2021 (Cimpanu C., 2021).

It follows from all the foregoing that, if the data (messages) are not preserved and forensically retrieved from the devices it will be extremely difficult to do so from anywhere else.

Chapter 4.

Viber application Scenario

In this chapter the Viber application's examination and the related results are presented regarding text and media exchanged messages and the volatility of the database in which they are stored. At first, there is a reference on the application database's location, it's features and the differences between an Android 9 and an Android 11 examined device. After that there is an analysis of the tables and columns of the database and the queries that were built in order to view the desired ones. In addition, the volatility of deleted text and media messages is examined through multiple experiments. Furthermore, some unique characteristics of the application are examined, and final conclusions are drawn.

4.1. Viber's Examination

The used .apk for both the devices was the Viber v.16.4.0.8 which was released in October 2021. The application's file and folder structure is well known and documented over the years for both educational/research (Hermawan et al., 2021, Anglano et al., 2016) and forensic purposes (Belkasoft, 2021). In this Thesis the focus is on the messaging database and the modifications which occur after sending, receiving, reading and deleting messages or/and elapsed time and the device's status. The goal is to examine under which conditions deleted records can be recovered. Since this is not a black box experiment and as matter of fact, there is a total control of the device it is much safer to draw some conclusions. The above-mentioned "messaging database" is named "viber mesages.db" and is located. among others. in the path: «data/data/com.viber.voip/databases». The access to this path is normally prohibited, so the examiner must either have "root" access to the device or acquire a physical image of it.

4.1.1. Android 9 vs Android 11 Comparison

The structure of the database's folder («data/data/com.viber.voip/databases») in the 16.4.0.8 version of the application in the two devices had some minor differences:

- Firstly, the google API ("com.google.android.datatransport.events.db") and "cdr.db" (Call Details Records - Metadata) databases in the Android 9 device used the newer WAL journaling mode. On the contrary the newer Android 11 device used the legacy journal mode. Those two databases do not have any participation in the experiment or differentiation on the results and therefore were not taken into consideration.
- Secondly, the initial size of the "viber_messages" databases and the overall size of their corresponding journal files differentiated in the two devices. The initial database (right after the first login of a user), in five different occasions on each device, had a small difference in size. This was depended on the volume of the initial pushed data by the company's server to the "chat_extension", "chat_suggestions" etc. tables.

4.1.2. General information about the under-examination database

Generally, the initial size of the "viber_mesages" database, before any transactions, was approximately 380 kb and the journal's size was 80 kb. After having created several transactions (sending/receiving) among four (4) users, it was observed that the maximum size of the journal file, was related to the number of the active users and the type of transaction that had occurred. The size of the main database is constantly altered as a result of the "auto_vacuum" PRAGMA statement (**Figure 31**).

-rw 1 u0_a250 u0_a250 475136 2021-11-07	20:00 withon macrogar
-rw 1 uθ_a250 uθ_a250 390392 2021-11-07	
	20:09 viber_prefs
	20:09 viber_prefs-journal
-rw-rw 1 u0_a250 u0_a250 671744 2021-11-07	
	19:15 google_app_measurement_local.db
	18:59 cdr.db-shm
	18:59 exoplayer.db
	18:59 exoplayer.db-shm
	18:59 exoplayer.db-wal
	18:59 com.google.android.datatransport.events-shm
-rw 1 u0_a250 u0_a250 122880 2021-11-07	
	12:08 viber_data-journal
	12:08 cdr.db-wal
-rw-rw 1 u0_a250 u0_a250 4096 2021-11-07	
	12:07 com.google.android.datatransport.events
	12:07 com.google.android.datatransport.events-wal
lucye:/data/data/com.viber.voip/databases # ls	
total 2308	
	20:13 viber_messages
-rw 1 u0_a250 u0_a250 390392 2021-11-07	
-rw-rw 1 uθ_a250 uθ_a250 798720 2021-11-07	
	20:12 viber prefs
-rw 1 u0_a250 u0_a250 49760 2021-11-07	20:12 viber_prefs-journal
-rw-rw 1 u0_a250 u0_a250 16384 2021-11-07	19:15 google_app_measurement_local.db
	18:59 cdr.db-shm
-rw-rw 1 u0 a250 u0 a250 4096 2021-11-07	18:59 exoplayer.db
-rw 1 u0_a250 u0_a250 32768 2021-11-07	18:59 exoplayer.db-shm
	18:59 exoplayer.db-wal
-rw 1 u0_a250 u0_a250 32768 2021-11-07	18:59 com.google.android.datatransport.events-shm
-rw 1 u0_a250 u0_a250 122880 2021-11-07	12:08 viber_data
-rw 1 u0_a250 u0_a250 86696 2021-11-07	12:08 viber_data-journal
-rw 1 u0_a250 u0_a250 61832 2021-11-07	12:08 cdr.db-wal
-rw-rw 1 u0_a250 u0_a250 4096 2021-11-07	12:07 cdr.db
-rw-rw 1 u0_a250 u0_a250 4096 2021-11-07	12:07 com.google.android.datatransport.events
-rw 1 u0_a250 u0_a250 53592 2021-11-07	12:07 com.google.android.datatransport.events-wal
lucye:/data/data/com.viber.voip/databases # ls	
total 2464	
-rw-rw 1 u0_a250 u0_a250 929792 2021-11-07	
-rw 1 u0_a250 u0_a250 <mark>512000 2021-11-07</mark>	20:18 viber messages
-rw 1 u0_a250 u0_a250 390392 2021-11-07	20:18 viber_messages-journal
-rw 1 u0_a250 u0_a250 94208 2021-11-07	20:18 viber_prefs
	20:18 viber_prefs-journal
	19:15 google_app_measurement_local.db
-rw 1 u0_a250 u0_a250 32768 2021-11-07	18:59 cdr.db-shm
	18:59 exoplayer.db
-rw 1 u0_a250 u0_a250 32768 2021-11-07	
-rw 1 u0_a250 u0_a250 65952 2021-11-07	
	18:59 com.google.android.datatransport.events-shm
-rw 1 u0_a250 u0_a250 122880 2021-11-07	12:08 viber_data

Figure 31. Comparison of the "viber_messages-journal" file size while increasing the size of main database with transaction between two (2) active users. The journal file is always 381 kb (Examined Device: LG Device).

The way the size of journal file was increased or decreased followed a certain pattern. For four (4) active users, after the transmission and receival of several text, photos and video messages, the size of the journal file was always 409 kb for the Samsung device and 393 kb for the LG Device. After a total deletion of the exchanged messages, the size of the file grown (721 kb for Samsung and 777 kb for the LG) but remained stable through the rest of the experiments even after sending and receiving more content on the two devices (**Figures 32** and **33**).

-гw 1 и0_a267 и0_a267 419120 2021-12-05 20:44 viber_messages-journal -гw 1 и0_a267 и0_a267 593920 2021-12-05 20:44 viber_messages	
-rw 1 u0_a267 u0_a267 419120 2021-12-05 20:48 viber_messages-journal -rw 1 u0_a267 u0_a267 667648 2021-12-05 20:48 viber_messages	-rw 1 u0_a250 u0_a250 1187840 2021-12-05 21:09 viber_messages -rw 1 u0_a250 u0_a250 402704 2021-12-05 21:09 viber_messages-journal
-rw 1 u0_a267 u0_a267 419120 2021-12-05 20:52 viber_messages-journal -rw 1 u0_a267 u0_a267 823296 2021-12-05 20:52 viber_messages	-rw 1 uθ_a250 uθ_a250 1204224 2021-12-05 21:24 viber_messages -rw 1 uθ_a250 uθ_a250 402704 2021-12-05 21:24 viber_messages-journal
-rw 1 u0_a267 u0_a267 419120 2021-12-05 20:54 viber_messages-journal -rw 1 u0_a267 u0_a267 827392 2021-12-05 20:54 viber_messages	-rw 1 uθ_a250 uθ_a250 1351680 2021-12-05 21:30 viber_messages -rw 1 uθ_a250 uθ_a250 402704 2021-12-05 21:30 viber_messages-journal
-rw 1 u0_a267 u0_a267 419120 2021-12-05 20:57 viber_messages-journal -rw 1 u0_a267 u0_a267 880640 2021-12-05 20:57 viber_messages -rw 1 u0_a267 u0_a267 1036288 2021-12-05 21:00 viber_messages	тw 1 и0_a250 и0_a250 1507328 2021-12-05 21:35 viber_messages rw 1 и0_a250 и0_a250 402704 2021-12-05 21:35 viber_messages-journal
-τw 1 u0_a267 u0_a267 419120 201-12-05 21:00 viber_messages -τw 1 u0_a267 u0_a267 419120 2021-12-05 21:00 viber_messages	-rw 1 u0_a250 u0_a250 991232 2021-12-05 21:36 viber_messages
-rw 1 u0_a267 u0_a267 419120 2021-12-05 21:09 viber_messages-journal	-rw 1 u0_a250 u0_a250 796688 2021-12-05 21:36 viber_messages-journal rw 1 u0_a250 u0_a250 782336 2021-12-05 21:36 viber_messages rw 1 u0_a250 u0_a250 796688 2021-12-05 21:36 viber_messages-journal -rw 1 u0_a250 u0_a250 532400 2021-12-05 21:37 viber_messages -rw 1 u0_a250 u0_a250 796688 2021-12-05 21:37 viber_messages-journal -rw 1 u0_a250 u0_a250 671744 2021-12-05 21:40 viber_messages
-rw 1 u0_a267 u0_a267 743336 2021-12-05 21:20 viber_messages-journal -rw 1 u0_a267 u0_a267 622592 2021-12-05 21:24 viber_messages -rw 1 u0_a267 u0_a267 74336 2021-12-05 21:24 viber_messages-journal -rw 1 u0_a267 u0_a267 54864 2021-12-05 21:27 viber_messages -rw 1 u0_a267 u0_a267 743336 2021-12-05 21:27 viber_messages	rw 1 uθ_a256 uθ_a256 796688 2021-12-05 21:40 viber_messages -rw 1 uθ_a250 uθ_a250 536576 2021-12-05 21:40 viber_messages -rw 1 uθ_a250 uθ_a250 796688 2021-12-05 21:40 viber_messages-journal
Figure 32. Comparison of "viber_messages.db" journal file's size while increasing the size of main database with transaction among four (4) active users on the Samsung A50 device.	Figure 33. Comparison of "viber_messages.db"- journal file's size while increasing the size of main database with transaction among four (4) active users on the LG G6 device.

There was a clear correlation between the size of the journal file and the number of the active users. A conclusion that could be drawn from this is that the size of the journal file of "viber_messages" database, does not necessarily correlate with the volume of the recovered deleted content from one user/chat thread. The active pragma statements for the databases in the two devices can be examined from the file headers. The examination and comparison which is depicted in **Figures 34** and **35** involves copies of the examined databases by using the nonforensic tools: "SQLite Expert" and "DB Browser for SQLite". When a non-forensic tool is used there is a probability that the data of main database could be modified, especially if the database coexists in the same path as its journal/wal file.

Previo	us changes N	ext changes Ignore case Ignore wh	nitespace							
elect	First Database:	viber_messages	✓ Select	Second Data	base: viber	_messag	es(1)			
1	Database: [viber messages]	1	Database:	[viber m	essages	3 (1)]			
2	Connect	ion: read-write	2	Conne	ction: re	ad-writ	;e			
3	File na	me: H:\Thesis\Viber\SAMSUNG	A505\viber\v 3	File	name: H:\	Thesis\	Viber\LGH	870\v:	iber\o	rigin
4	File si	ze: 397312 bytes	4	File	size: 331	776 byt	es			
5	Page si	ze: 4096	5	Page	size: 409	6				
6	Encodin	g: UTF-8	6	Encod	ing: UTF-	8				
7	Auto va	cuum: 1	7	Auto	vacuum: 1					
8	Tables:	25	8	Table	s: 25					
9	Views:	0	9	Views	: 0					
10	Virtual	Tables: 0	10	Virtu	al Tables	: 0				
11			11							
12	Table [appl	ications]	12	Table [ap	plication	s]				
13	Fields:	13	13	Field	s: 13					
14	[_i	d]: INTEGER	14	[_id]: INT	EGER				
15		NOT NULL	15		NOT NU	LL				
16	[na:	me]: TEXT	16	[name]: TE	XT				
17	[ty	pe]: TEXT	17	[type]: TE	XT				
18	[pa	ckage_name]: TEXT	18	[package_n	ame]: 1	EXT			
19	[st	ore_id]: TEXT	19	[store_id]	: TEXT				
20	[ur	l_scheme]: TEXT	20	[url_schem	e]: TEX	T			
21	[st	atus]: INTEGER	21	[status]:	INTEGER	R .			
22	[f1	ags]: INTEGER	22	[flags]: I	NTEGER				
23		DEFAULT '0'	23		DEFAUL	T '0'				
24	[la	st_modified]: INTEGER	24		last_modi	fied]:	INTEGER			
Match	es: 924	Modified: 3	Ad	ded: 0		• • • • •	Deleted: 0			

Figure 34. Comparison of "viber_messages" database in Samsung A50 (left) and in LG G6 (right) (tool used: SQLite Expert).

Auto Vacuum 😟	Full ~
Automatic Index 😟	
Case Sensitive Like 🥹	
Checkpoint Full FSYNC 😣	
Foreign Keys 🥹	
Full FSYNC 🥹	
Ignore Check Constraints 😣	
Journal Mode 😡	Delete 🗸
Journal Size Limit 🙆	-1
Locking Mode 😡	Normal ~
Max Page Count 😟	1073741823 🜩
Page Size 😡	4096 🗸
Recursive Triggers 😣	
Secure Delete 🥹	
Synchronous 🥹	Full 🗸
Temp Store 😣	Default 🗸
User Version 😣	214
WAL Auto Checkpoint 🥹	1000 🜩

Figure 35. SQLite Pragma Statements of "viber_messages.db" in edit mode (tool used: DB Browser for SQLite).

The analysis of the active pragma statements from the database-file headers yield that both databases:

- ✓ had the Full auto_vacuum enabled and did not create freelist pages,
- \checkmark used rollback journal as a journaling mode with unlimited size (-1), in theory.

The meaning and forensic value of these pragma statements (enabled options) have been analyzed in paragraph §1.3.1.2. All these settings render the data, especially the deleted ones, extremely volatile. The settings of the "viber_messages" database have been changed over the years. In older versions there were some key differences in the database which allowed an examiner to recover records with a greater success rate. At least until the 2017 versions of the application, freelist pages were created and kept in the database (**Table 10**).

Name	Value		Name	Value		Name	Value	
application_id	0		application_id	0		application_id	0	
auto_vacuum	none		auto_vacuum	full		auto_vacuum	full	
cache_size	-2000		cache_size	-2000		cache_size	-2000	
collation_list	[BINARY], [NOCASE], [F		collation_list	[BINARY], [NOCASE], [RT		collation_list	[BINARY], [NOCASE], [RTI	
encoding	UTF-8		encoding	UTF-8		encoding	UTF-8	
foreign_keys	on		foreign_keys	on		foreign_keys	on	
freelist_count	47		freelist_count	0		freelist_count	0	
journal_mode	delete		journal_mode	delete		journal_mode	delete	
journal_size_limit	-1		journal_size_limit	-1		journal_size_limit	-1	
max_page_count	1073741823	1	max_page_count	1073741823	Ð	max_page_count	1073741823	
mmap_size	0		mmap_size	0		mmap_size	0	
page_count	254		page_count	95		page_count	125	
page_size	4096		page_size	4096		page_size	4096	
schema_version	49		schema_version	72		schema_version	72	
user_version	99		user_version	207		user_version	214	
writable_schema	off		writable_schema	off		writable_schema	off	
Version 49- Se	eptember 2017		Version 72- June 2021			Version 72- October 2021		

 Table 11. Comparing settings of "viber_messages.db" on different versions of the application (tool used:

 SQL Expert).

4.1.3. Analysis of the "viber_messages.db"

The "viber_messages" database was composed by twenty-four (24) tables (**Figure 36**). The two (2) most forensically interesting tables, for the examined scenario were the "messages" (**Figure 37**) and "participants_info".

	Tables (24)
<hr/>	applications
1	
>_	🔟 backgrounds
>	blocked_data
>	chat extensions
	chatex_suggestions
٢.	conference_calls
>	conversations
> `	gc_file
2	group_delete_all_from_participant
>	hidden_gems
>	hidden_gems_data
s Ì	messages
ĺ.	
2	messages_calls
>	messages_likes
>	messages_reminders
Ś	participants
/	participants_info
>	public_accounts
>	purchase
5	recent stickers
Į.	remote banners
1	
>_	sqlite_sequence
>	stickers
>	stickers packages
-	in other presented

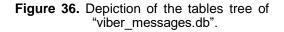


Figure 37. Some of the columns that are consisted in "messages" table.

read_message_time

INTEGER

The "**messages**" table contained the most interesting columns. Some of the column names are self-explanatory, such as the "**body**" which contains the body of the message. On the contrary others need to be explained regarding their type of content and form:

- The "**msg_date**" column contained the timestamp when a message was created, and it was encoded in Unix Millisecond (Java Time).
- The "**send_type**" column's value indicated the direction of the message. When the value was "1" meant that it was an outgoing message. Any other value indicated that it was an incoming one.
- The "**unread**" column's value was "1" or "0". The value "1" indicated that the message had not yet been read and value "0" that the message had been viewed.
- The "status" column contained information about whether the message was not sent ("0") (e.g., problem in the device's network connection), sent but not delivered ("1"), or delivered ("2").
- The "**read_message_time**" column contained information about the timestamp of the read time of a message by its receiver.
- The "**msg_info**" column contained several types of information e.g., the metadata of a transmitted media file or if a message has been edited by the user.

The "participants_info" table contained information about the participants. In this table, the phone number and the contact's name could be found. The data in the columns of this table were combined with the data of the columns of the "messages" table in order to create new tables which contained easily understandable results. This was achieved through simple SQL queries (Figures 38 and 39) and by joining the "member_id" and "user_id" columns of these two tables.

10	WILEN T	THEN Message was	not received.						
19	when '2'	THEN 'Message rece	ived'						
20	ELSE 'Uni	known Status'							
21	LEND AS status,	,							
22	CASE messages	.read_message_time							
23		THEN 'Message was n	ot opened'						
24		ssage was Opened'							
25		ge Read Status'							
26	FROM messages								
27		ants_info ON messag ages.msg date ASC	es.user_1d = p	articipants_in	ro.memper_1d				
	Message	Message Date	user_id	display_name	number	Direction	Opened	status	Message Read Status
1	This is a test	2021-11-10 10:43:48	DrmrchQhcUo=	Bad Actor 1	+3069407	Outgoing	True	Message received	Message was Opened
2	This is also a test	2021-11-10 10:48:32	DrmrchQhcUo=	Bad Actor 1	+3069407	Incoming	False	Message received	Message was not opened
3	This is a test 2	2021-11-10 10:49:21	Qhnb3h7J3pE=	Μιχάλης	+3069472_ 1	Outgoing	True	Message received	Message was not opened
4	This is third test	2021-11-10 10:49:46	Qhnb3h7J3pE=	Μιχάλης	+306947271.5	Outgoing	True	Message received	Message was not opened
5	This is test number 4	2021-11-10 10:50:56	CfunDWmeF1o=	Edgar Allan Poe	+306986f	Outgoing	True	Message was not received	Message was not opened
6	This is test 5		Sector Contraction of the Contraction of the	the second second second	100000000	Outration	+22222	www.construction.com.com.com.com.com.com	
	This is test 5	2021-11-10 10:51:17	CfunDWmeF1o=	Edgar Allan Poe	+300980	Outgoing	True	Message was not received	Message was not opened
7	This is test 6	2021-11-10 10:51:17 2021-11-10 10:51:41		Edgar Allan Poe Μιχάλης	+306986	Outgoing	True	Message was not received Message was not sent	5
			Qhnb3h7J3pE=				302030-403		Message was not opened Message was not opened Message was not opened

Figure 38. SQL query combining some of the columns and tables.

25

```
CASE
WHEN messages.msg_info LIKE '%edit%' THEN 'Message has been edited'
END AS 'Message info'
26
```

27

SELECT

messages.body As 'Message',

participants_info.number, CASE messages.send_type

WHEN '1' THEN 'Outgoing' ELSE 'Incoming'

WHEN '0' THEN 'True'

WHEN '1' THEN 'False' ELSE messages.opened

WHEN '0' THEN 'Message was not sent'

WHEN '1' THEN 'Message was not received'

participants_info.display_name,

messages.user_id,

END AS 'Direction',

LEND AS 'Opened',

CASE messages.status

CASE messages.unread

datetime(messages.msg_date/1000,'unixepoch','localtime') AS 'Message Date',

1

2

3

4

5 6

7

8 9

10 11

12

14

15 16

17 18

þ 13

28 FROM messages

```
JOIN participants info ON messages.user_id = participants info.member_id
29
```

ORDER BY messages.msg_date ASC 30

	Message	Message Date	user_id	display_name	number	DIRECTION	Opened	status	Message Opened	Message info
1	This is a test and test 12	2021-11-11 23:48:25	DrmrchQhcUo=	Bad Actor 1	+306940713831	Incoming	True	RECEIVED	Message Opened	Message has been edited
2	This a test 2 and 11	2021-11-11 23:48:58	DrmrchQhcUo=	Bad Actor 1	+306940713831	Outgoing	True	RECEIVED	Message Opened	Message has been edited
3	This is a test 4	2021-11-11 23:54:10	DrmrchQhcUo=	Bad Actor 1	+306940713831	Incoming	True	RECEIVED	Message Opened	NULL
4	This is test 5	2021-11-11 23:56:34	DrmrchQhcUo=	Bad Actor 1	+306940713831	Outgoing	True	RECEIVED	Message Opened	NULL
5	This is test 6	2021-11-11 23:57:31	DrmrchQhcUo=	Bad Actor 1	+306940713831	Incoming	True	RECEIVED	Message Opened	NULL
6	This is test 7	2021-11-11 23:57:39	DrmrchQhcUo=	Bad Actor 1	+306940713831	Incoming	True	RECEIVED	Message Opened	NULL
7	This is test 8	2021-11-11 23:57:47	DrmrchQhcUo=	Bad Actor 1	+306940713831	Incoming	True	RECEIVED	Message Opened	NULL
8	This is test 9	2021-11-12 00:01:38	DrmrchQhcUo=	Bad Actor 1	+306940713831	Outgoing	True	RECEIVED	Message Opened	NULL
9	This is test 10 and 11	2021-11-12 00:01:52	DrmrchQhcUo=	Bad Actor 1	+306940713831	Outgoing	True	RECEIVED	Message Opened	Message has been edited
10	This a test 2 and 11	2021-11-13 10:00:42	DrmrchQhcUo=	Bad Actor 1	+306940713831	Outgoing	True	RECEIVED	Message not opened	Message has been edited

Figure 39. Adding another case statement in the previous SQL query, it can be viewed if the message has been edited or not.

At this point it must be mentioned that the columns' names inside the tables and the meaning of the corresponding flags/data are modified among versions. For example, in the Viber v.16.4.7.2, the "unread" column was absent and was replaced with the "open" column. Those two

columns differ not only in name but also in meaning. In the "**unread**" column the value of "1" meant that the message was not read by the user, in contrast with the column "**open**" where the flag "1" meant the exact opposite (that the message was read by the user). This means that the SQL queries that were executed in this Thesis might not be applicable in other versions of these applications.

4.1.4. Volatility of the "viber_messages.db"

The experiments confirmed that the crucial data in the "viber_messages" database and its corresponding journal file are not susceptible to any of device status changes or time, including:

- ✓ Leaving the device powered on with active network. Take note that in the experiments that were executed it was not taken under consideration the possibility of a remote wipe of the device. Forensically is highly recommended to disable all the network connections of the device.
- ✓ Leaving the device in a faraday box with network.
- ✓ Putting the device in airplane mode.
- \checkmark Rebooting the device.
- ✓ Shutting down the device.
- ✓ Removing the SIM card (which was the only thing associated with account), while the device was live and after it was deactivated and then powered on without the SIM.

All the above experiments were repeated in several time frames and did not modify by any means the two files. During the experiments there was only one case, in which "killing" (closing) the application manually from the background and reopening it, modified the files. That exception had to do with "bad timing" because at that moment Viber's server happened to push some new data in the "**chatex_suggestions**" table in "viber_messages.db". As long as there were not any interactions with the application, the examined database was not modified even after several hours (**Figure 40**) or even days.

4	-rw-rw	1	u0_a250	u0_a250	4096	2021-11-05	22:29	cdr.db
32	-rw	1	u0_a250	u0_a250	32768	2021-11-06	00:47	cdr.db-shm
36	6 -rw	1	u0_a250	u0_a250	32992	2021-11-05	22:29	cdr.db-wal
4	-rw-rw	1	u0_a250	u0_a250	4096	2021-11-05	22:29	com.google.android.datatransport.events
32	-rw	1	u0_a250	u0_a250	32768	2021-11-06	00:47	com.google.android.datatransport.events-shm
56	5 -rw	1	u0_a250	u0_a250	53592	2021-11-05	22:29	com.google.android.datatransport.events-wal
4	-rw-rw	1	u0_a250	u0_a250	4096	2021-11-05	22:31	exoplayer.db
32	-rw	1	u0_a250	u0_a250	32768	2021-11-05	22:31	exoplayer.db-shm
68	-rw	1	u0_a250	u0_a250	65952	2021-11-05	22:31	exoplayer.db-wal
16	-rw-rw	1	u0_a250	u0_a250	16384	2021-11-06	20:50	google_app_measurement_local.db
716	-rw-rw	1	u0_a250	u0_a250	729088	2021-11-06	20:50	mixpanel
120	-rw	1	u0_a250	u0_a250	122880	2021-11-06	00:45	viber_data
64	-rw	1	u0_a250	u0_a250	62072	2021-11-06	00:45	viber data-iournal
348	-rw	1	u0_a250	u0_a250	356352	2021-11-06	01:02	viber_messages
96	-rw	1	u0_a250	u0_a250	94904	2021-11-06	01:02	viber_messages-journal
100	-rw	1	u0_a250	u0_a250	98304	2021-11-06	10:57	viber_prefs
52	-rw	1	u0_a250	u0_a250	45656	2021-11-06	10:57	viber_prefs-journal
luc	ye:/data/da	ta	/com.vibe	er.voip/	databas	es #		
	5							へ 遠 記 句》 ENG ^{8:52 μμ} 6/11/2021

Figure 40. The device's "viber_messages.db" was not modified after 20 hours, even changing its state several times.

As mentioned before, the schema which Viber has implemented in the last versions, renders the data extremely volatile. By sending just one message the record was present three times in the main database and its journal file. Every time the record changes it's state (from sent to delivered, or from delivered to have been read by the receiver) its previous version is moved to the journal file. Only the most recent status of the record resides in the main database while the previous states of the same record reside in the journal file (**Figure 41**).

1000										
\checkmark		6/11/2021 8:09:48 µµ	How are you my friend?	5637864372885770891	1	0	4096	0	CfunDWmeF1o=	0
	Ť.	6/11/2021 8:09:48 µµ	How are you my friend?	5637864372885770891	1	0	4096	0	CfunDWmeF1o=	0
	ά	6/11/2021 8:09:48 µµ	How are you my friend?	0	1	0	4096	0	CfunDWmeF1o=	0

Figure 41. How "viber_messages.db" and its journal file are altered after one message was sent (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50).

This was verified through the FQLite tool. In **Figure 42** it can be observed that the same message (record) existed two times in the journal file and one time in the main database (latest state). The same rule applied when more messages were sent as it can be viewed in **Figures 43** to **45**.

idx_CONVERSATION_participant_id_1									Filt	er:								
idx_MESSAGES_LIKES_message_token	e evt	user_id	seq	status	opened	sync read	location I	location L.	extra uri	destinati	extra_sta	message	extra uni	extra do	extra bu	sticker id	extra du	body
idx MESSAGES conversation id AND order k	0	CfunDWmeF1o=	18465945		O	0	0	0	CHOID_BIT				0	Cito C_dom	Croid_Dam	serence_ro	0	How are you my friend
idx_MESSAGE5_conversation_type_AND_extra	l o	CfunDWmeF1o=	18465945		0	0	0	ő				0	0				0	How are you my frien
idx_MESSAGES_conversation_type_AND_toke	P	cluibwiner io=	1040354	2015	. jo	P	P ^V	lo.			12	v	No.				P	now are you my men
idx_MESSAGES_extra_mime																		
idx_MESSAGES_group_id_AND_order_key																		
dx_MESSAGES_seq																		
idx MESSAGES status AND msg date AND t																		
idx_MESSAGES_timebomb_read_message_tin																		
idx MESSAGES token																		
idx_MESSAGES_user_id																		
kdx_PARTICIPANT5_INF0_contact_id																		
idx_PARTICIPANTS_INFO_encrypted_number																		
idx_PARTICIPANTS_INFO_member_id_number																		
idx PARTICIPANTS INFO participant type Af																		
idx PARTICIPANTS conversation id																		
idx_PUBLIC_ACCOUNTS_group_id																		
idx_PUBLIC_ACCOUNTS_public_account_id																		
idx STICERS package id																		
messages																		
messages calls																		
messages_likes																		
messages reminders																		
part idx MESSAGES LIKES read 0																		
part idx MESSAGES broadcast msg id																		
part_idx_MESSAGES_community_empty_mes																		
part_idx_MESSAGES_conversation_type01_AI																		
part idx MESSAGES deleted																		
part idx MESSAGES extra status																		
part_idx_MESSAGES_order_key_AND_msg_da																		
part idx MESSAGES sync read																		
part_idx_MESSAGES_token_AND_sync_read																		
part idx MESSAGES unread																		
participants																		
participants_info																		
public_accounts																		
purchase																		
recent_stickers																		
remote_banners																		
sqlite_sequence																		
stickers																		
stickers_packages																		
viber_messages-journal																		
idx_MESSAGES_status_AND_msg_date_AND_t																		
Mitstages *																		
>	<																	

Figure 42. How "viber_messages.db" journal file was altered after one message was sent (tool used: FQLite, Examined Device: Samsung A50).

	1000			,							
1			6/11/2021 8:09:48 μμ	How are you my friend?	1	0	4096	0	CfunDWmeF1o=	2	0
2	\checkmark		6/11/2021 8:10:21 µµ	It is been a long time since we talked	1	0	4096	0	CfunDWmeF1o=	2	0
3		ti di	6/11/2021 8:10:21 µµ	It is been a long time since we talked	1	0	4096	0	CfunDWmeF1o=	1	0
4		竝	6/11/2021 8:10:21 µµ	It is been a long time since we talked	1	0	4096	0	CfunDWmeF1o=	3	0

Figure 43. How "viber_messages.db" journal file was altered after two messages were sent (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50).

1		6/11/2021 8:	How are you my friend?	1	0	4096	0	CfunDWmeF1o=	2	0
2		6/11/2021 8:	It is been a long time since we talked	1	0	4096	0	CfunDWmeF1o=	2	0
		6/11/2021 8:	Are you in Boston?	1	0	4096	0	CfunDWmeF1o=	2	0
4	Ť	6/11/2021 8:	Are you in Boston?	1	0	4096	0	CfunDWmeF1o=	1	0
5	ά	6/11/2021 8:	Are you in Boston?	1	0	4096	0	CfunDWmeF1o=	3	0

Figure 44. How "viber_messages.db" journal file was altered after three messages were sent (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50).

#	\	msg_date 🔺	 participant_id 	body ■	unread	▼ flag	 deleted 	✓ user_id ✓ sta	atus 🔹 opened	 read_message_time 	•
1		6/11/2021 8:09:48 µµ	1	How are you my friend?	0	4096	0	CfunDWmeF1o= 2	0	651788737	
2		6/11/2021 8:09:48 µµ	1	How are you my friend?	0	4096	0	CfunDWmeF1o= 2	0	0	
3	\checkmark	6/11/2021 8:10:21 µµ	1	It is been a long time since we talked	1 0	4096	0	CfunDWmeF1o= 2	0	651788737	
4		6/11/2021 8:10:21 µµ	1	It is been a long time since we talked	1 0	4096	0	CfunDWmeF1o= 2	0	0	
5	\checkmark	6/11/2021 8:10:38 µµ	1	Are you in Boston?	0	4096	0	CfunDWmeF1o= 2	0	651788737	
6		6/11/2021 8:10:38 µµ	1	Are you in Boston?	0	4096	0	CfunDWmeF1o= 2	0	0	
7		6/11/2021 8:10:38 µµ	1	Are you in Boston?	0	4096	0	CfunDWmeF1o= 1	0	0	
8		б/11/2021 8:10:38 µµ	1	Are you in Boston?	0	4096	0	CfunDWmeF1o= 3	0	0	

- **Figure 45.** How "viber_messages.db" journal file was altered after three messages were opened by the receiver. By changing the "read_message_time" flag, all records which were previously unread are moved to the journal (tool used: Oxygen Sqlite Viewer, Examined Device: Samsung A50).
 - So, the database and its journal file were modified when:
 - a message was created. It did not matter whether was successfully sent or not.
 - ! a message was received.
 - **!** a message was opened. This was regardless of if it was an outgoing message read by the receiver or an incoming read by the user.
 - ! a message or a thread was deleted by the user or by timebomb.
 - a bot message (ads) was received, opened, or deleted.
 - ! a message notification was removed (Figures 46 and 47).

backgrounds (0)		Record type	_id	conversation_id	order_key	↓ F msg_date	participant_id	send_type	user_id
blocked_data (0) chat_extensions (12)		Journal	16	4	56378672129207774	6/11/2021 8:21:06 PM	8	0	DrmrchQhcUo=
chatex_suggestions (1671)	0	Journal	17	4	56378672366563432	6/11/2021 8:21:12 PM	8	0	DrmrchQhcUo=
onference_calls (0)	0	Journal	18	4	56378681589329967	6/11/2021 8:24:51 PM	7	1	DrmrchQhcUo=
onversations (16)	0	Journal	19	4		6/11/2021 8:25:17 PM	7	1	DrmrchQhcUo=
_file (2)		Journal							
roup_delete_all_from_participant (0)	0		20	4		6/11/2021 8:26:56 PM	8	0	DrmrchQhcUo=
dden_gems (395)	0	Journal	21	4	56378686994743896	6/11/2021 8:27:00 PM	8	0	DrmrchQhcUo=
dden_gems_data (2)	0	Journal	22	4	56378687224843408	6/11/2021 8:27:06 PM	8	0	DrmrchQhcUo=
essages (34)	0	Journal	23	4	56378687647377592	6/11/2021 8:27:16 PM	8	0	DrmrchQhcUo=
essages_calls (0) essages likes (0)	0	Journal	24	4	56378688089625016	6/11/2021 8:27:26 PM	8	0	DrmrchQhcUo=
essages_nees (0) essages_reminders (0)	0	Journal	13	4	56378671401663797	6/11/2021 8:20:49 PM	8	0	DrmrchQhcUo=
articipants (8)	0	Journal	14	4	56378671685072925	6/11/2021 8:20:55 PM	8	0	DrmrchQhcUo=
rticipants_info (5)	0	Journal	15	4	56378671926245399	6/11/2021 8:21:01 PM	8	0	DrmrchQhcUo=
iblic_accounts (1)	0	Journal	36	1	56378721672450117	6/11/2021 8:40:47 PM	2	0	CfunDWmeF1o=
irchase (0) cent stickers (67)	0	Journal	37	1	56378722149258594	6/11/2021 8:40:59 PM	2	0	CfunDWmeF1o=
mote_banners (0)	0	Journal	38	1	56378722346516705	6/11/2021 8:41:03 PM	2	0	CfunDWmeF1o=
lite_sequence (20)	0	Journal	39	1	56378722537818914	6/11/2021 8:41:08 PM	2	0	CfunDWmeF1o=
ickers (444)	0		16	4	56378672129207774	6/11/2021 8:21:06 PM	8	0	DrmrchOhcUo=
ickers_packages (4)			17						DrmrchQhcUo=
Unallocated space	0		17	4	202/00/2300503432	6/11/2021 8:21:12 PM	8	0	DrmrchQncUo=

I if new data were pushed by Viber's server (e.g., download new emoticons suggestion).

Figure 46. Before removing Viber Notification: eighteen (18) messages existed in "viber_messages.db" and 16 messages existed in journal (tool used: Belkasoft-X, Examined Device: Samsung A50).

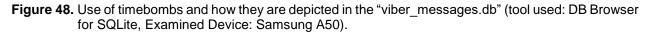
backgrounds (0)		Record type	_id	conversation_id	order_key	↓ F msg_date	participant_id	send_type	user_id
blocked_data (0) chat extensions (12)		Journal	16	4	56378672129207774	6/11/2021 8:21:06 PM	8	0	DrmrchQhcUo=
chatex_suggestions (1671)		Journal	17	4	56378672366563432	6/11/2021 8:21:12 PM	8	0	DrmrchQhcUo=
conference_calls (0)	0	Journal	18	4	56378681589329967	6/11/2021 8:24:51 PM	7	1	DrmrchQhcUo=
conversations (12)	0	Journal	19	4	56378682672928407	6/11/2021 8:25:17 PM	7	1	DrmrchQhcUo=
gc_file (2) group_delete_all_from_participant (0)	0	Journal	20	4	56378686799457097	6/11/2021 8:26:56 PM	8	0	DrmrchQhcUo=
hidden gems (395)	0	Journal	21	4	56378686994743896	6/11/2021 8:27:00 PM	8	0	DrmrchQhcUo=
hidden_gems_data (2)	0	Journal	22	4	56378687224843408	6/11/2021 8:27:06 PM	8	0	DrmrchQhcUo=
messages (51)		Journal	23	4	56378687647377592	6/11/2021 8:27:16 PM	8	0	DrmrchQhcUo=
messages_calls (0)	0	Journal	24	4	56378688089625016	6/11/2021 8:27:26 PM	8	0	DrmrchQhcUo=
messages_likes (0) messages_reminders (0)	0	Journal	13	4	56378671401663797	6/11/2021 8:20:49 PM	8	0	DrmrchQhcUo=
participants (8)	0	Journal	14	4	56378671685072925	6/11/2021 8:20:55 PM	8	0	DrmrchQhcUo=
participants_info (5)	0	Journal	15	4	56378671926245399	6/11/2021 8:21:01 PM	8	0	DrmrchQhcUo=
public_accounts (1)	0	Journal	36	1	56378721672450117	6/11/2021 8:40:47 PM	2	0	CfunDWmeF1o=
purchase (0)	0	Journal	37	1	56378722149258594	6/11/2021 8:40:59 PM	2	0	CfunDWmeF1o=
recent_stickers (67) remote banners (0)	0	Journal	38	1	56378722346516705	6/11/2021 8:41:03 PM	2	0	CfunDWmeF1o=
sqlite_sequence (20)	0	Journal	39	1	56378722537818914	6/11/2021 8:41:08 PM	2	0	CfunDWmeF1o=
stickers (444)	0	Journal	16	4		6/11/2021 8:21:06 PM	8	0	DrmrchQhcUo=
stickers_packages (4)	0	Journal	17	4		6/11/2021 8:21:12 PM	8	0	DrmrchQhcUo=
Unallocated space	0	Journal	18	4		6/11/2021 8:24:51 PM	7	1	DrmrchQhcUo=

Figure 47. After having Viber Notification removed: eighteen (18) messages existed in "viber_messages.db" and thirty-three (33) messages existed in it's journal file. Removing messages notification can potentially push something important out from the journal file by replacing it (tool used: Belkasoft-X, Examined Device: Samsung A50).

4.1.5. Retrieving the messages in question and the timebomb feature

During the experiments some of the exchanged messages of the scenario included the timebomb feature. Timebomb is a message autodeletion setting. The user predefines an autodeletion timer countdown which is activated after a message has been read by the receiver. After that the exchanged message is deleted from the chat thread. The Viber application treats these kinds of messages like normal ones, and the only element that differentiates them, is the flag "autodelete" which is displayed as "timebomb/X" (where X= 10,60,3600 seconds) in the body of the message (**Figure 48**).

19	I am sending the photo	2021-11-06 22:35:09	Qhnb3h7J3pE=
20	content://com.viber.voip.provider.internal_files/	2021-11-06 22:35:18	Qhnb3h7J3pE=
21	Did you receive it?	2021-11-06 22:35:43	Qhnb3h7J3pE=
22	Yes send the second one	2021-11-06 22:35:56	Qhnb3h7J3pE=
23	content://com.viber.voip.provider.internal_files/	2021-11-06 22:36:11	Qhnb3h7J3pE=
24	This will cost you 1000 euros	2021-11-06 22:36:26	Qhnb3h7J3pE=
25	timebomb/60	2021-11-06 22:36:44	Qhnb3h7J3pE=



By examining the database and its journal file with the X-Ways Forensics, or any other HEX Editor, and searching it by the value inside the "**user_id**" column, which contains the user that sent the messages, it could be observed that the messages with enabled timebomb setting still existed and could be retrieved as long as no other commits were executed to the database (**Figure 49**). To make things clearer: the timebomb feature added an extra state to the record when it deleted it. Since the journal file is the invalid previous version of the database it could still be retrieved.

L712-F	W OL	L-21-7			0		712	F	e	21	712.	-	~ ~		-712		~ ~	funDWmeF1o= % CfunDV
	% Qhn	03N/	JSPE															hnb3h7J3pE= % CfunDWr
				7	6 Qn	nbsn	/J3p			nb3h nb3h								hnb3h7J3pE= % Qhnb3h7 hnb3h7J3pE= % Qhnb3h7
									/6 Q1	incon	rispi							hnb3h7J3pE= % Qhnb3h7
b/60{}<&	帰れる	,	2			G		N_8	± ± ± 5	鲼z3	1_83		% Qr		n7J3 57J2		% Q	hnb3n/J3pE= %Qnnb3n/
o_sp<&?F			%	0	#	0				問((25) 贲?N=			OF	nh3	b713	E-t	imeh	omb/60{}<&镘 {} ?
t?{}no_sp{			%		?					· 橬=								Ve agreed on five hundred{
]no_sp&i			%							鲻?N=								na you receive it:{/no_sp{//
uros{}no_s		Я.	10	%														Yes send the second one{}n
}no_sp<8		117	2			G				·鲼z3I								his will cost you 1000 euros
_sp<&館			%			#				睬 1N								want more{ }no_sp<&?媚}
no_sp<&			%			#				嵘C昇								WE WILL SEE{}no_sp<&館
mbnailInf			%		1		N	=?#c	戚!	₩KN	=?#o	觑						think they saw me. I must o
e{}no_sp<			%	1	×	咵?				鲼BI								content://media/external/ir
o3h7J3pE=		nnb3	h7J3	pE='	' ?"Q	hnb	3h7J3	pE=	! ? Qł	nb3h	7J3pl	E= 1	% QI	nb3	h7J3	DE=	% Q	hnb3h7J3pE= % Qhnb3h7
	% QI	hnb3	h7J3	pE=	# ?30	Qhnb	3h7J	3pE=	-" ?"C) hnb3	h7J3	pE=!	? Qł	nb3	h7J3	oE=	% Q	hnb3h7J3pE= % Qhnb3h7
					% Qł	nnb3	h7J3	pE=#	# ?3Q	hnb3l	h7J3p	E="	?"Qł	nb3	h7J3	E=!	?Qh	nb3h7J3pE= % Qhnb3h7J3
									% Qł	nb3h	7J3p	E=#	?3Q	nb3	h7J3	•E="	?"Qł	nb3h7J3pE=! ? Qhnb3h7J3j
													% Qł	nb3	h7J3	DE=#	?3Q	hnb3h7J3pE=" ?"Qhnb3h7J
no_sp&?	擬}z,		%		3		N=	降N	o袞	鲻?N=	₿ÂN	0衰	Qł	nnb3	h7J3p	oE=n	iq. D	id you receive it?{}no_sp&
	圸/		%]	×	咵?		N=ឪ	窖席?	鲼BI	1=窖	席?	?QI	nnb3	h7J3p	oE=n	is祸(content://media/external/ir
n7J3pE=	% Qhn	b3h7	/J3pE	= 9	6 Qh	nb3h	7J3p	E=	% Qł	nb3h	7J3pl	E= 1	% <mark>Q</mark> ł	nnb3	h7J3p	DE=	Cfu	InDWmeF1o= % DrmrchQ
<																		
Se Vo	lume	Γ	Fil	e		Prev	iew	9	Det	ails		Galle	ery		Caler	ıdar		Legend 🔅 🕥 Syne
Offs	et	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	ANSI ASCII
00048	816	B0	96	ED	01	7C	F6	F8	43	95	4E	3D	BE	11	26	BO	96	°-i öøC•N=¾ &°-
00048	832	ED	04	10	00	08	00	00	00	51	68	6E	62	33	68	37	4A	í Qhnb3h7J
00048	848	33	70	45	3D	6E	77	C9	C4	02	03	57	45	20	57	49	4C	<mark>3pE=</mark> nwÉÄ WE WIL
00048	864	4C	20	53	45	45	7B	7D	6E	6F	5F	73	70	3C	26	F0	5E	L SEE{}no_sp<&ð^
00048	880	8A	7B	7D	0E	13	00	77	01	06	05	06	08	01	80	02	80	Š{} W
00048	896	04	08	09	08	25	04	01	08	08	08	08	00	00	01	08	80	9 6
00048	912	00	00	00	08	23	0D	11	00	09	08	17	01	04	08	08	11	#
00048						BE					F7						31	N=¾ Y/Ó÷ öø 1
00048						59					10			00			51	N=¾ Y/Ó÷ Q
00048		68	6E			68				70	45						02	hnb3h7J3pE=nwKù
00048			49			61					6F		65					I want more{}no
00048						26						81		1E		00	01	_sp<&ð<"{} ,
00049		06				01					08	09						*
00049		08				00					00			47				G
		09	08			04		08						BD			6F	N=3:₽Úo
00049								77	22	45	3D	BD	DE				33	
00049	056		33															Ò3 ö÷z3N=≒⊉ÚoÒ3
00049 00049 00049	056 072	03	10	00	02	00	51	68	6E	62	33		37		33	70	45	Qhnb3h7J3pE
00049 00049 00049 00049	056 072 088	03 3D	10 6E	00 74	02 AB	00 F7	51 02	68 03	6E 54	62 68	33 69	73	20	77	69	6C	6C	Qhnb3h7J3pE =nt«÷ This will
00049 00049 00049 00049 00049	056 072 088 104	03 3D 20	10 6E 63	00 74 6F	02 AB 73	00 F7 74	51 02 20	68 03 79	6E 54 6F	62 68 75	33 69 20	73 31	20 30	77 30	69 30	6C 20	6C 65	Chnb3h7J3pE =nt«÷ This will cost you 1000 e
00049 00049 00049 00049 00049 00049	056 072 088 104 120	03 3D 20 75	10 6E 63 72	00 74 6F 6F	02 AB 73 73	00 F7 74 7B	51 02 20 7D	68 03 79 6E	6E 54 6F 6F	62 68 75 5F	33 69 20 73	73 31 70	20 30 26	77 30 F0	69 30 3C	6C 20 94	6C 65 7B	Qhnb3h7J3pE =nt«÷ This will cost you 1000 e uros{}no_sp&8<"{
00049 00049 00049 00049 00049 00049 00049	056 072 088 104 120 136	03 3D 20 75	10 6E 63 72 7C	00 74 6F 6F 1C	02 AB 73 73 2C	00 F7 74 7B 00	51 02 20 7D 01	68 03 79 6E 06	6E 54 6F 6F 05	62 68 75 5F 06	33 69 20	73 31 70 01	20 30 26 08	77 30 F0	69 30 3C 08	6C 20 94 08	6C 65 7B 08	<pre>Chnb3h7J3pE =nt«÷ This will cost you 1000 e uros{}no_sp&ð<"{ } ,</pre>
00049 00049 00049 00049 00049 00049	056 072 088 104 120 136	03 3D 20 75	10 6E 63 72	00 74 6F 6F 1C	02 AB 73 73	00 F7 74 7B	51 02 20 7D	68 03 79 6E	6E 54 6F 6F	62 68 75 5F	33 69 20 73	73 31 70	20 30 26	77 30 F0	69 30 3C	6C 20 94	6C 65 7B	Qhnb3h7J3pE =nt«÷ This will cost you 1000 e uros{}no_sp&8<"{
00049 00049 00049 00049 00049 00049 00049	056 072 088 104 120 136 152	03 3D 20 75 7D 08	10 6E 63 72 7C 08	00 74 6F 6F 1C 25	02 AB 73 73 2C 04	00 F7 74 7B 00	51 02 20 7D 01 08	68 03 79 6E 06	6E 54 6F 6F 05 08	62 68 75 5F 06	33 69 20 73 08 00	73 31 70 01	20 30 26 08 01	77 30 F0 02	69 30 3C 08 08	6C 20 94 08 00	6C 65 7B 08	Chnb3h7J3pE =nt«÷ This will cost you 1000 e uros{}no_sp&&<"{ } , % ;
00049 00049 00049 00049 00049 00049 00049 00049	056 072 088 104 120 136 152 168	03 3D 20 75 7D 08 00	10 6E 63 72 7C 08 08	00 74 6F 1C 25 3B	02 AB 73 73 2C 04 0D	00 F7 74 7B 00	51 02 20 7D 01 08 00	68 03 79 6E 06 08 09	6E 54 6F 05 08 08	62 68 75 5F 06 08 17	33 69 20 73 08 00	73 31 70 01 00 04	20 30 26 08 01 08	77 30 F0 02 08 08	69 30 3C 08 08 11	6C 20 94 08 00 08	6C 65 7B 08 00 02	<pre>Qhnb3h7J3pE =nt«÷ This will cost you 1000 e uros{}no_sp&ð<"{ } , %</pre>
00049 00049 00049 00049 00049 00049 00049 00049 00049	056 072 088 104 120 136 152 168 184	03 3D 20 75 7D 08 00 4E	10 6E 63 72 7C 08 08 3D	00 74 6F 1C 25 3B BD	02 AB 73 73 2C 04 0D C1	00 F7 74 78 00 01 11	51 02 20 7D 01 08 00 B7	68 03 79 6E 06 08 09	6E 54 6F 05 08 08	62 68 75 5F 06 08 17 01	33 69 20 73 08 00 08	73 31 70 01 00 04 F6	20 30 26 08 01 08 F7	77 30 F0 02 08 08 08 04	69 30 3C 08 08 11	6C 20 94 08 00 08	6C 65 7B 08 00 02	<pre>Qhnb3h7J3p =nt«+ This wil cost you 1000 uros{}no_sp&8<" } , % ;</pre>

Figure 49. Searching timebombs messages content using "user_id" in "viber_messages.db" (tool used: X-Ways Forensics, Examined Device: Samsung A50).

The problem lies that with the speed that the applications are updating their settings and databases, it is very difficult for forensic tools, free or commercial, to keep up with the correct parsing of the application's databases and journal/wal files. The verification by using multiple tools could help, with the retrieval of most of the records, but at the end, it is necessary to dive straight into file itself in a hex level to ensure that nothing is missed. The experiments showed that the usage of the contents of the column "user_id" as a searchable string can preview some fast results. It is understandable that viewing the data of the file as hexadecimal is a hard task, but unfortunately necessary and vital in many cases (AskClees, 2020). Several forensic tools (free and commercial) and the way they parse the deleted timebomb records are depicted in (**Figures 50** to **52**).

On some occasions a forensic tool returned one message of the deleted ones after the timebomb feature has been activated (**Figure 50**), whereas examining the same database with different tools retrieved the most if not all of the erased, by timebomb, messages (**Figures 51** and **52**). This comparison is not made to point out the advantage of one software/tool over another but to point out that multiple ones have to be used in order to draw a final and detailed conclusion.

conversa	order_key	msg_date	body	user_id
2	null	Sat Nov 06 22:35:43 EET 2021	Did you receive it?	Qhnb3h7J
2	null	Sat Nov 06 22:35:56 EET 2021	Yes send the second one	Qhnb3h7J
2	null	Sat Nov 06 22:36:11 EET 2021	content://com.viber.voip.provider.internal_files/message/lo	Qhnb3h7J
2	null	Sat Nov 06 22:36:26 EET 2021	This will cost you 1000 euros	Qhnb3h7J
2	null		timebomb/60	Qhnb3h7J
2	null	Sat Nov 06 22:37:41 EET 2021	I think they saw me. I must delete those	Qhnb3h7J

Figure 50. How "viber_messages" journal file was modified after messages with timebomb were deleted (tool used: FQLite Examined Device: Samsung A50).

	Journal	2	6/11/2021 8:36:12 PM	Qhnb3h7J3pE=	content://com.viber.voip.provider.internal_files/message/lo
\Box	Journal	2	6/11/2021 8:35:44 PM	Qhnb3h7J3pE=	Did you receive it?
\Box	Journal	2	6/11/2021 8:37:41 PM	Qhnb3h7J3pE=	I think they saw me. I must delete those
\Box	Journal	2	6/11/2021 8:36:26 PM	Qhnb3h7J3pE=	This will cost you 1000 euros
\Box	Journal	2	6/11/2021 8:35:56 PM	Qhnb3h7J3pE=	Yes send the second one
\bigcirc	Journal	2	6/11/2021 8:36:45 PM	Qhnb3h7J3pE=	timebomb/60

Figure 51. How "viber_messages" journal file was modified after timebomb messages were deleted (tool used: Belkasoft-X, Examined Device: Samsung A50).

10	\checkmark	Ť.	6/11/2021 8:37:00 µµ	4	We agreed on five hundred	0	Qhnb3h7J3pE=
11		竝	6/11/2021 8:37:08 µµ	3	I want more	0	Qhnb3h7J3pE=
12		竝	6/11/2021 8:37:08 µµ	3	I want more	0	Qhnb3h7J3pE=
13		竝	6/11/2021 8:37:17 µµ	4	WE WILL SEE	0	Qhnb3h7J3pE=
14		竝	6/11/2021 8:37:41 µµ	3	I think they saw me. I must delete those	0	Qhnb3h7J3pE=

Figure 52. How "viber_messages" file was modified after timebomb messages were deleted (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50).

After the manual deletion of the conversation thread which also contained timebomb messages, the forensic tools, without tweaking them, were still able to recover several records, with different levels of success (**Figures 53** and **54**).

Journal	6/11/2021 10:36:12 P	Qhnb3h7J3pE=	content://com.viber.voip.provider.internal_files/message/local/thumb/b68ab198524939961cbb801a0a	66 69 6C 65 49 6E 66
Journal	6/11/2021 10:35:44 P	Qhnb3h7J3pE=	Did you receive it?	
Journal	6/11/2021 10:36:26 P	Qhnb3h7J3pE=	This will cost you 1000 euros	
Journal	6/11/2021 10:35:56 P	Qhnb3h7J3pE=	Yes send the second one	
Journal	6/11/2021 10:36:45 P	Qhnb3h7J3pE=	timebomb/60	

Figure 53. How "viber_messages" journal file was modified five minutes after the conversation thread was deleted (tool used Belkasoft-X, Examined Device: Samsung A50).

17	\checkmark	m	6/11/2021 8:35:09 µµ	I am sending the photo	3
18	\square	ά	6/11/2021 8:35:18 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/155012	3
19	\square	ά	6/11/2021 8:35:18 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/155012	3
20	\square	Ť	6/11/2021 8:35:43 µµ	Did you receive it?	3
21	\square	竝	6/11/2021 8:35:43 µµ	Did you receive it?	3
22	\square	竝	6/11/2021 8:35:56 µµ	Yes send the second one	4
23	\square	ά	6/11/2021 8:36:11 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/b68ab1	3
24		ά	6/11/2021 8:36:11 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/b68ab1	3
25	\checkmark	ά	6/11/2021 8:36:11 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/b68ab1	3
26	\square	ά	6/11/2021 8:36:26 µµ	This will cost you 1000 euros	3
27		ά	6/11/2021 8:36:26 µµ	This will cost you 1000 euros	3
28	\checkmark	ά	6/11/2021 8:36:26 µµ	This will cost you 1000 euros	3
29	\square	ά	6/11/2021 8:36:44 µµ	timebomb/60	4
30		ά	6/11/2021 8:36:44 µµ	timebomb/60	4
31		ά	6/11/2021 8:37:00 µµ	We agreed on five hundred	4
32	\square	ά	6/11/2021 8:37:08 µµ	I want more	3
33		ά	6/11/2021 8:37:08 µµ	I want more	3
34		竝	6/11/2021 8:37:17 µµ	WE WILL SEE	4
35		竝	6/11/2021 8:37:41 µµ	I think they saw me. I must delete those	3

Figure 54. How "viber_messages" journal file was modified five minutes after the conversation thread was deleted (tool used: Oxygen Forensic SQlite, Examined Device: Samsung A50).

It has to be mentioned that in the latter example the timebomb messages had not yet been vacuumed. In a realistic scenario the acquisition of the data after five minutes it is considered impossible. The full "auto_vacuum" feature of the database irretrievably removes the timebomb records after a few minutes, making it impossible to recover them, while the records/messages which were manually deleted (the ones that existed in the previous state of the database) are still recoverable. This procedure is independent of the device's brand or OS version. The exact timeframe in which full "auto_vacuum" is triggered could not be pinpointed. **Figures 55** to **58** depicts the way that the messages table in the Viber database in the LG device is modified through time using several forensic tools and software.

#	2	msg_date 🔺 💌	participant_id	• user_id •	status 💌	opened 🔹	body -	extra_uri 🔹
1		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	I am sending the photo	
2		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	content://com.viber.voip.provider.internal_files/message	content://media/external/images/media/949
3		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	content://com.viber.voip.provider.internal_files/message	content://media/external/images/media/949
4	\checkmark	31/10/2021 1	1	Qhnb3h7J3pE=	2	0	Did you receive it?	
5	\checkmark	31/10/2021 1	2	Qhnb3h7J3pE=	2	0	Yes send the second one	
6		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	content://com.viber.voip.provider.internal_files/message	content://com.viber.voip.provider.internal_files/message
7		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	content://com.viber.voip.provider.internal_files/message	content://com.viber.voip.provider.internal_files/message
8		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	This will cost you 1000 euros	
9		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	This will cost you 1000 euros	
10		31/10/2021 1	2	Qhnb3h7J3pE=	0	0	timebomb/60	
11		31/10/2021 1	2	Qhnb3h7J3pE=	2	0	We agreed on five hundred	
12		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	I think they saw me. I must delete these	
13		31/10/2021 1	1	Qhnb3h7J3pE=	2	0	I think they saw me. I must delete these	

Figure 55. How "viber_messages" journal file was modified five minutes after the timebomb messages were deleted (tool used: Oxygen Forensic SQlite, Examined Device: LG G6).

🗹 🗯	31/10/2021 10:08:05 nµ	1	I am sending the photo	Qhnb3h7J3pE=
☑ =	31/10/2021 10:08:05 nµ	1	I am sending the photo	Qhnb3h7J3pE=
🗹 🟛	31/10/2021 10:08:12 nµ	1	content://com.viber.voip.provider.internal_files/message/local/thumb/f61a2b5b541014969df6695d	Qhnb3h7J3pE=
☑ =	31/10/2021 10:08:12 nµ	1	content://com.viber.voip.provider.internal_files/message/local/thumb/f61a2b5b541014969df6695d	Qhnb3h7J3pE=
2	31/10/2021 10:08:26 nµ	1	Did you receive it?	Qhnb3h7J3pE=
2	31/10/2021 10:08:26 nµ	1	Did you receive it?	Qhnb3h7J3pE=
¹	31/10/2021 10:08:48 nµ	2	Yes send the second one	Qhnb3h7J3pE=
🗹 🟛	31/10/2021 10:08:48 nµ	2	Yes send the second one	Qhnb3h7J3pE=
🗹 🟛	31/10/2021 10:09:24 nµ	1	content://com.viber.voip.provider.internal_files/message/local/thumb/e27c001ac1c34e8c6dff95419	Qhnb3h7J3pE=
☑ =	31/10/2021 10:09:24 пµ	1	content://com_wiber_voin_provider.internal_files/message/local/thumb/e27c001ac1c34e8c6dff95419	Qhnb3h7J3pE=
2 🗖	31/10/2021 10:09:44 nµ	1	This will cost you 1000 euros	Qhnb3h7J3pE=
2 🗖	31/10/2021 10:09:44 nµ	1	This will cost you 1000 euros	Qhnb3h7J3pE=
Image: Second	31/10/2021 10:10:03 nµ	2	timebomb/60	Qhnb3h7J3pE=
☑ 🗖	31/10/2021 10:10:03 nµ	2	timebomb/60	Qhnb3h7J3pE=

Figure 56. How "viber_messages" journal file was modified 1, 6, 24 and 48 hours after the conversation thread was deleted with the device in several states. The timebomb messages were deleted permanently (tool used: Oxygen Forensic SQLite, Examined Device: LG G6).

000	- ~	"#	\bigcirc	×	K	0	C	† Timestamp ▼	From	Body
٥		4		×		Ø	ব	31/10/2021 10:08:12 πμ(UTC	+3069407 Bad Actor 1 (owner)	
Ö		5					ব	31/10/2021 10:08:26 πμ(UTC	+3069407 Bad Actor 1 (owner)	Did you receive it?
		6		×			ব	31/10/2021 10:08:26 πμ(UTC	+3069407 Bad Actor 1 (owner)	Did you receive it?
Ö	\checkmark	7					Ľ	31/10/2021 10:08:48 πµ(UTC	+3069472 Μιχάλης	Yes send the second one
		8		×			للا	31/10/2021 10:08:48 πµ(UTC	+3069472 Μιχάλης	Yes send the second one
į.	$\mathbf{\Sigma}$	9				0	ব	31/10/2021 10:09:24 πμ(UTC	+3069407 Bad Actor 1 (owner)	This is the second one
		10		×		0	3	31/10/2021 10:09:24 πµ(UTC	+3069407 Bad Actor 1 (owner)	This is the second one
Ū.		11					ব	31/10/2021 10:09:44 πµ(UTC	+3069407 Bad Actor 1 (owner)	This will cost you 1000 euros
		12		×			ব	31/10/2021 10:09:44 πμ(UTC	+3069407 Bad Actor 1 (owner)	This will cost you 1000 euros
Ü		13					Ľ	31/10/2021 10:10:03 πµ(UTC	+3069472	timebomb/60
0		14		×			Ľ	31/10/2021 10:10:03 πμ(UTC	+3069472 Μιχάλης	timebomb/60
		15		×			Ľ	31/10/2021 10:10:29 πµ(UTC	+3069472	We agreed on five hundred
0		16		*			C	31/10/2021 10:11:05 πμ(UTC	+3069407 Bad Actor 1 (owner)	I think they saw me. I must delete these

Figure 57. How "viber_messages" journal file was modified five minutes after the timebomb messages were deleted (tool used: UFED PA, Examined Device: LG G6)

000	- ~	#	S	×	K	0	C	↑ Timestamp 🔹	From	•	Body
	\checkmark	1		×			ব	31/10/2021 10:08:05 πμ(UTC	+3069407 Bad Actor 1 (own	ner)	I am sending the photo
	\checkmark	2		×		0	ব	31/10/2021 10:08:12 πμ(UTC	+3069407 Bad Actor 1 (own	ner)	
	\checkmark	3		×			3	31/10/2021 10:08:26 πµ(UTC	+3069407 Bad Actor 1 (own	ner)	Did you receive it?
		4		×			Ľ	31/10/2021 10:08:48 πμ(UTC	+3069472 Μιχάλης		Yes send the second one
		5		×		0	ব	31/10/2021 10:09:24 πμ(UTC	+3069407 Bad Actor 1 (own	ner)	This is the second one
0	\checkmark	6		×			3	31/10/2021 10:09:44 πµ(UTC	+3069407 Bad Actor 1 (own	ner)	This will cost you 1000 euros
0		7		*			Ľ	31/10/2021 10:10:03 πµ(UTC	+3069472. Μιχάλης		timebomb/60

Figure 58. How "viber_messages" journal file was modified 1, 6, 24 and 48 hours after the conversation thread was deleted with the device in several states. The timebomb messages were deleted permanently (tool used: UFED PA, Examined Device: LG G6).

As already shown even one commit to the database was enough to lose the majority off the "incriminating messages". By receiving just five (5) messages from a third participant almost all the previous commits were vacuumed and thus permanently deleting the messages of interest (**Figures 59** and **60**).

11	3	3	6/11/2021 8:35:09 µµ	I am sending the photo	Qhnb3h7J3pE=	2
12	3	3	6/11/2021 8:35:18 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/155012ebd1c93729a0dce8dc9	Qhnb3h7J3pE=	2
13	3	3	6/11/2021 8:36:11 µµ	content://com.viber.voip.provider.internal_files/message/local/thumb/b68ab198524939961cbb801a0	Qhnb3h7J3pE=	2

Figure 59. How "viber_messages" journal file is modified after five (5) additional messages were received and not opened by the user (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50). Only the messages that contained records from the "msg_info" column remained.

11 🗹	Qhnb3h7J3pE=	I am sending the photo		2
12 🗹	Qhnb3h7J3pE=	content://com.viber.voip.provider.internal_files/message/local/thumb/155012ebd1c93729a0dce8dc941d339a	content://media/external/images/media/66	2
13 🗹	Qhnb3h7J3pE=		content://media/external/images/me"(-	2
14 🗹	Qhnb3h7J3pE=	content://com.viber.voip.provider.internal_files/message/local/thumb/b68ab198524939961cbb801a0af9d458	content://media/external/images/media/65	2

Figure 60. How "viber_messages" journal file is modified after the notification, of the previously received messages, is removed from the device. The data are further corrupted introducing false positive records (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50).

In conclusion, just three actions-commits, involving receiving messages, reading incoming messages, or having outcoming messages, read by another participant (receiver) etc., were enough to erase every trace of the records of interest.

4.2. Viber's Media Files Attachments

In the experiments two media files, were sent, one which created through the default camera application of the device and the other, was created via the internal camera feature of the Viber application. The media file was stored at «media/0/Pictures/Viber» if it was captured by the internal feature of the application or at «media/0/DCIM/Camera/» if it was captured by the default camera application of the device. Since the application did not automatically remove the media file from the filesystem when the messages which included it, was deleted, it was removed manually. In addition, the trash bin folder of the photo gallery application was emptied. The trash bin folder is an OS feature where after a media file's deletion, it is moved and kept separately for a period of time. This period varies, and for the LG device is 7 days as for the SAMSUNG device is 30 days. For media files except the columns which were mentioned in paragraph §4.1.3, there were also two more, which were extremely useful. The "msg_info" (Figure 61) which contained the metadata of the sent files in JSON format and the "extra_uri" (Figure 62) in which the internal storage path of the file is mentioned.



Figure 61. The "extra_uri" and "body" columns of the table "messages" in "viber_messages.db" (Examined Device: LG G6).

("fielnfo":("ContentType": 'PIC", Duration":0.0, 'FieEsti"):09', 'Fielsah': 'W39KisWa)/TLGKWWWAVQ.003d/u003d
 'FieName": '20211031_120716.jpg'
 FieSze": 234345, 'mediaInfo":('Height': 4160, 'MediaType": 'image", 'Width': 3120), 'MediaType": 'image", 'Width': 3120), 'MediaType: 'image", 'Width': 3120), 'MediaType: 'image", 'Width': 488), 'MediaType: 'image", 'Width': 488), 'MediaType: 'image", 'Width': 488), 'MediaType: 'image', '

Figure 62. The "msg_info" column of the table "messages" in "viber_messages.db" (Examined Device: LG G6).

The thumbnails of the media files, which were created during the transmission, were stored at the path «/media/0/Android/data/com.viber.voip/files/.thumbnails/» (Figure 63). In order to correlate the content of the transmitted message with a media file (picture) with a picture's created thumbnail, a comparison was made between their timestamps. One other path where artifacts can be found is the cache folder of the Viber application at «/media/0/Android/data/com.viber.voip/ cache/ImageFetcherThumb» (Figure 64).

com.viber.voip (778)	□ ▼Name▲	T Description	Type	Y Size	Created	
🚊 🦲 cache (7)	□, = files (770)	existing		14.5 MB	10/31/2021	10:03:39 1
	🗆 🛄. = .thumbnails (3)	existing		58.0 KE	10/31/2021	10:08:09
video-cache (1)	🔲 🗋 .nomedia	existing, already viewed		0 E	10/31/2021	10:08:09
🖕 🧰 files (770)	IMG-4fe59f7acffa80f757a1aa3b65e43343-V.jpg	existing	ipq	25.5 KB	10/31/2021	10:09:24
emoticons (287)	🗌 🛅 IMG-53e38a2384184d8b55ea8d3ea1bd5742-V.jpg	existing	jpg	32.5 KB	10/31/2021	10:08:09



🚞 com.google.android.video	□ ▼Name ▲	T Description	Type	Y Size	Created	
🚞 com.google.android.youtu	= cache (7)	existing		68.3 KB	10/31/2021	10:07:42
🚞 com.lge.exchange (0)	🗆 🧰 = ImageFetcherThumb (6)	existing		68.3 KB	10/31/2021	10:07:42
🚞 com.lge.gametools.gamer	0E77FBF13BB6B20A4511776B5A71DA11.0	existing	0	32.5 KB	10/31/2021	10:08:12
🚞 com.lge.hiddenmenu (0)	19D4549EA4391290E38009CF5486FF4D.0	existing	0	10.1 KB	10/31/2021	10:08:07
com.lge.lgworld (0)	E4B10CCCA401CDB7F65663A6EBEC8B27.0	existing	0	25.4 KB	10/31/2021	10:09:24
com.lge.qmemoplus (0)	E4B10CCCA401CDB7F65663A6EBEC8B27.0.tmp	prev. existing, file content	tmp			
com.topjohnwu.magisk (0	🔲 📄 journal	existing		238 B	10/31/2021	10:07:42
com.viber.voip (778)	🗆 🔀 journal.tmp	prev. existing, file content	tmp			
🖕 🧰 cache (7)						
- ImageFetcherThum						

Figure 64. Media files cache location (Examined Device: LG G6).

4.2.1. LG device's media artifacts

In the LG device the thumbnails of the media gallery were stored at path: «media/0/DCIM/.thumbnails». The timestamp when the user deleted the files and when the trash bin was emptied could be retrieved from the contents of the "trashcan.db" database and it's WAL file located at the path: «/data./data/com.android.gallery3d/databases/» (**Figure 65**).



Figure 65. LG G6's "trashcan.db" WAL file record (tool used: Oxygen Forensic SQlite).

Furthermore, the conversion of the blob file named "imagecache.0", which is located at the path: «/media/0/Android/data/com.android.gallery3d/cache», with a free tool (slo-sleuth, 2021) to a SQLite database and then parsing it with DB Browser for SQLite, provides valuable information about the captured media files (**Figure 66**).



Figure 66. Parsing "imagecache0.sqlite" (tool used: DB Browser for SQLite, Examined Device: LG G6).

To summarize all the above-mentioned artifacts, they are presented in chronological order:

- At 31-10-2021 10:07:16 UTC a media file named «20211031_120716» was captured by the device's camera application and stored at path: «../DCIM/Camera/» as the item «949» in the device's "InternalPath". The file's content is presented at the "Thumbnail" column (Figure 66).
- At 31-10-2021 **10:08:07** UTC the file is imported at the application for transmission and a cache file is created (**Figure 64**).
- At 31-10-2021 **10:08:09** UTC the file's thumbnail is created and stored permanently at path: «media/0/Android/data/com.viber.voip/files/.thumbnails/» (**Figure 63**).
- At 31-10-2021 **10:08:12** UTC the file is sent as a message. The "extra_uri" and "body" columns depict the original internal path of the media (**Figure 61**), and the media's filename is shown inside the JSON file (**Figure 62**).
- The file is deleted and removed from the trash bin of the device's gallery application (**Figure 65**).

Finally, if the location tag's setting was enabled in the examined device, the geolocation of the captured pictures could be found in the file «/media/0/Android/data/com.android.gallery3d/cache/**rev_geocoding.0**».

4.2.1. Samsung device's media artifacts

The Samsung device has a modified version of stock android. It does not use the Android's default camera or photo gallery applications. The pictures' thumbnails are located in the Samsung gallery application at path: «/data/com.sec.android.gallery.3d/cache/». The analysis of the "local.db" database located at the path: «/data/com.sec.android.gallery.3d/databases/» and specifically the "trash" table of this database provided the timestamps of media deletion (**Figure 67**). Most of the column names of this table were self-explanatory.

Y FIELDS													
 Fash_ORIGIN trash_ORIGIN trash_ORIGIN Deletion Time Da trash_MEDIAT trash_MEDIAT trash_MEDIAT trash_HEIGHT trash_RESTOR FROM trash 		BURSTOROUPD DIT _BESTMAGE INT _GOLOBERVERID TEXT _GOLOBERVERID TEXT _GOLODERVERID _GOLOPEVERIE VOLUMEVALID INT											Hoto Book Service Serv
	+ Out	put Expression	Alias	Sort Type	Sort Order	Aggregate	Grouping	Criteria	Or	0r	Or		*
	1 2 2	trashORIGINPATH		1.1								-	
		trashORIGINTITLE											
,	• = E	DateTime(trashDELETETIME / 1000, '	'u Deletion Time										v .
ashABSPATH, ashMEDIATYP ashWIDTH, ashHEIGHT,	ΡE,	IME / 1000, 'unimepoch') AS "Del	letion Time",										
rashABSPATH, rashMEDIATYP rashWIDTN, rashHEIGHT, rashPESTOPEE M trash	ΡE,	ля / 1000, поллерост / жа тое.	letion Time",										
vateine traan. preshMSSPATH, breshMEDIATYP breshMIDIN, preshREIGHT, breshREIGHT, breshRESTORE S, Rows = 4 RIGNPATH	ΡE,	na / 1000, "Unisepoon") as "be.	ORIGIMITILE			Deleton Time	_A	BSPATH			MEDIATYPE	WIDTH	_HEIDHT_RESTOREDTRA
crashMSSPATH preshMSDIATH preshMSDIATH preshMSIOTH, preshRSSTOPEE CSI trash n,Rons = 4 RIGDPATH	je, extra			000ee+2d9d0a	36efdd-V.jp				(Android)/ Trash/com.s	ec.android.gellmy34(4495) (94378 548) 56045		WIDTH 5760	("is360Video";false, "isDrm";false, "isPavourite";false, "fleDuration";0, "_recordingMode";
rashMSPATR, rashMSDTATY rashMSDTATY rashMSDTATY RECOT, rashRECOT, rashRECOT, rashRECOT, (Conseq) (C	es/Vber/DMG-	-19a72ad7ad66ic300ee4389sta36efd54-(pg	ORIGINITILE IMG-f3e7c3cf3ef69c3			ng 2021-10-31 11	::41:59 /sto	age/emulated/C		ec.androd.gellery3/(4950)5917814630091 ec.androd.gellery3/(4950)5917814630095	1		(
creathASSPATH, creathHSDIATH, creathHSDIATH presthHSDIAT, creathRSCORE, HSDIATH scaleASSACT RIGNPATH age/enulated/0/Picture	νΣ, IXTRA es/Vber/DMG- es/Vber/DMG-	-19x723479469-300e+329883384699-4399 -19x723479469-300e+329883384699-4399	ORIGINITILE IMG-f3e7c3cf3ef69c3	300ee42:19:0a)		ng 2021-10-31 11 ng 2021-10-31 11	::41:59 /sto ::41:59 /sto	age/emulated/C age/emulated/C	(Android/ Trash/com.s		5 1	5760	[

Figure 67. The "trash" table of Samsung's "local.db" (tool used: Forensic Toolkit for SQLite, Examined Device: Samsung A50).

Two more databases which contained interesting artifacts were:

The "graph.db" located at path: «/data/com.samsung.mlp/databases» (Figure 68). MLP stands for Mobile Location Protocol, and it can be used, among others, to track the media files which were sent as messages.

/storage/emulated/0/DCIM/Camera/20211031_122307.jpg	65537	31/10/2021 10:23:56 пµ	31/10/2021 10:23:07 nµ
/storage/emulated/0/Pictures/Viber/IMG-f3e7c3cf3ef69c300ee42d9d0a36e	65537	31/10/2021 10:25:04 nµ	31/10/2021 10:24:26 nµ

- Figure 68. The "MediaAttribute" table of Samsung "graph.db" (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50).
 - The "core2.db" at path: «/data/com.sec.android.app.camera/databases/» (Figure 69), contained information only about the pictures created via the camera application.

1000 L		-	_	-		
\checkmark	2147483649	/storage/emulated/0/DCIM/Camera/20210905_124444.jpg	external_primary	5/9/2021 9:44:44 nµ	20210905_124444	20210905_124444.jpg
\checkmark	2147483650	/storage/emulated/0/DCIM/Camera/20210905_124454.jpg	external_primary	5/9/2021 9:44:54 nµ	20210905_124454	20210905_124454.jpg
\checkmark	2147483651	/storage/emulated/0/DCIM/Camera/20211031_104226.jpg	external_primary	31/10/2021 8:42:26 пµ	20211031_104226	20211031_104226.jpg
	2147483653	/storage/emulated/0/DCIM/Camera/20211031_122307.jpg	external_primary	31/10/2021 10:23:07 nµ	20211031_122307	20211031_122307.jpg

Figure 69. The "files" table of Samsung "core2.db" (tool used: Oxygen Forensic SQLite, Examined Device: Samsung A50).

4.3. Final Conclusions on Viber

To sum up, DFIR Team members should avoid interacting with the Viber Application on android devices (remove notifications, read messages etc). In addition, they must isolate the device from networks, so that the Viber server cannot push any new data (i.e., read a sent message from a third participant/receiver). The experiments showed that the elapsed time or the state of the device (turned on/off, reboot, removing SIM etc.) did not alter the in-question data (deleted). Information about the in-question messages, especially the ones that contain media files, might be retrieved from artifacts residing in files and databases of other OS applications.

Chapter 5.

WhatsApp application Scenario

In this chapter the WhatsApp application's examination and the related results are presented regarding text and media exchanged messages and the volatility of the database in which they are stored. Like in the previous chapter there is a reference on the application database's location, it's features and the differences between an Android 9 and an Android 11 examined device. After that there is an analysis of the tables and columns of the database and the queries that were built in order to view the desired ones. In addition, the volatility of deleted text and media messages is examined through multiple experiments. Furthermore, some unique characteristics of the application are examined, and final conclusions are drawn.

5.1. WhatsApp's Examination

The used .apk for both the devices was the WhatsApp v.2.21.22.27 and its release date was on 8th November 2021. Application's data's, file and folder structure is well known and documented over the years for both educational/research (Rathi et al., 2018, Zhang et al., 2018) and forensic purposes (Belkasoft, 2021). There is an interesting approach made by Mirza et al. (2020) in which artifacts can be retrieved, under specific circumstances from the notification system of the application. As it has already been mentioned this Thesis is focused on the actions of an DFIR and the methodology best suited to be followed regarding the preservation of deleted data (messages). There were two databases that contained forensically interesting artifacts: the "msgstore.db" which included the messages, and the "wa.db" which included information about the user contacts. Both databases located at were the path: «/data/data/com.whatsapp/databases». Furthermore, the private encryption key was retrieved from the path: «/data/data/com.whatsapp/files» and used for the decryption of the user's chat backup. Once again, it must be pointed out that access to these paths is normally prohibited, so either the examiner must have root access to the device or acquire a physical image of it to extract them.

5.1.1. Android 9 vs Android 11 Comparison

The initial size of the "msgstore.db" on both devices, running different Android version, was 988kb. However, the WAL file's initial size differed. On the LG Device (Android 9) its size was 12kb, growing to the maximum of 477kb, and on the Samsung Device (Android 11) it had a steady size of 512kb.

5.1.2. General information about the under-examination database

The header and PRAGMA statements of the "msgstore.db" database analysis, by using the non-forensic tools "SQL Expert" and "DB Browser for SQLite" showed that (**Figure 70**):

- ✓ had the Full auto_vacuum enabled and did not create freelist pages,
- ✓ used WAL as a journaling mode,

✓ the WAL Auto Checkpoint had been set at one thousand (1.000) commits (default size).

The meaning and forensic value of these pragma statements (enabled options) have been analyzed in paragraph §1.3.1.2. All these settings render the data, especially the deleted ones, volatile. However, the use of WAL as a journaling mode provides some leniency in comparison to the use of the rollback journal in the Viber application.

Name	Value	Auto Vacuum 🙆	Full
application_id	70	Automatic Index 😣	\checkmark
auto_vacuum	full	Case Sensitive Like 🥹	
cache size	-2000	Checkpoint Full FSYNC 🥹	
collation_list	[BINARY], [NOCASE], [RTRIM]	Foreign Keys 😣	
encoding	UTF-8	Full FSYNC 😣	
foreign_keys	on	Ignore Check Constraints	
freelist_count	0	Journal Mode 🥹	WAL ✓
journal_mode	wal	Journal Size Limit 🥹 Locking Mode 🚱	Normal V
journal_size_limit	-1	Max Page Count @	1073741823
max_page_count	1073741823	Page Size 😣	4096 ~
mmap_size	0	Recursive Triggers 🔞	
page_count	249	Secure Delete 😣	
page_size	4096	Synchronous 😡	Full 🗸
schema_version	315	Temp Store 😣	Default 🗸
user_version	1	User Version 😣	1
writable_schema	off	WAL Auto Checkpoint 😟	1000 韋

Figure 70. Analysis of "msgstore.db" properties (tools used: SQLite Expert (left) & DB Browser for SQLite (right)).

The "wa.db" database contained information about the contacts of the user and participants in the chat threads. Since it is not frequently modified (a user does not add or delete contacts every minute) it can be characterized as a "steadier" database and even though it will be utilized in the SQL queries, its properties, will not be analyzed any further.

5.1.3. Analysis of the "msgstore.db" and the "wa.db"

The "msgstore" database was composed by one hundred and forty-two (142) tables (**Figure 69**). The two (2) most forensically interesting tables for the examined scenario were the "messages" and "message_media". The "wa" database was composed by twenty-four (24) tables (**Figure 70**) with the most forensically interesting table to be the "wa_contacts".

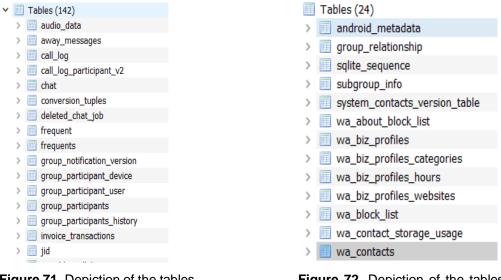


Figure 71. Depiction of the tables tree of "msgstore.db".

Figure 72. Depiction of the tables tree of "wa.db".

Among the most interesting columns of the "msgstore" database's table "messages" were:

- The "key remote jid" contained the WhatsApp ID of the contact.
- The "timestamp" contained the timestamp of the message creation, and it was encoded in Unix Millisecond (Java Time).
- The "key_from_me" contained information about the message direction.
- The "media_wa_type" identified the type of message (i.e., text, media, audio, video).
- The "data" contained the body of the text message.
- The "media_url" was the web location of the encrypted media in WhatsApp servers.
- The "media_caption" contained the title of the transmitted media.

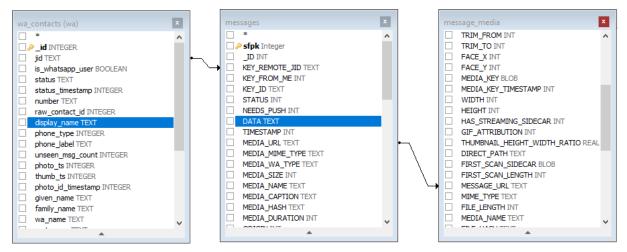
Among the most interesting columns of the "msgstore" database's table "message_media" were:

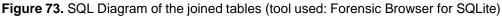
- The "file_path" contained the original file path of the media file in the device's filesystem.
- The "message_url" was the web location of the encrypted media's in WhatsApp servers.
- The "media_key" contained the decryption key in binary form (BLOB).

Among the most interesting columns of the "wa" database's table "wa_contacts" were:

- The "number" contained the phone number associated to the contact.
- The "display name" contained the contact's display name.

At this point it must be mentioned that the columns' names inside the tables and the meaning of the corresponding flags/data are modified among versions. A further analysis for most of the above-mentioned tables and their columns is made by Anglano C. (2014). To get an overview of the user's activity the two different databases ("msgstore" and "wa") had to be attached and the three tables had to be combined (**Figure 73**). The creation of the queries was a bit more complicated (**Figure 74**).





2 wa.w	CT								
	a contacts	display name AS '	User Name',						
		.number AS 'Phone I							
			'unixepoch') AS 'Me	essage Date',					
		.KEY_FROM_ME							
		THEN 'Message Receiv	ved"						
		Message was sent'							
		ge Direction',							
			MESTAMP/1000, 'unixe	poch') AS 'Date was	s Read',				
		AS Message,							
		media_wa_type							
		'HEN 'text'							
		'HEN 'image' 'HEN 'audio'							
		HEN 'video'							
		HEN 'contact card'							
		"HEN 'geo location'							
	AS 'Messag								
		CAPTION AS 'Media	Message'.						
20 mess	ages.MEDIA	URL AS 'Whatsapp	server media url',						
21 hex (message me	dia.MEDIA KEY) as	Media Decryption key	y',					
	age media.	FILE_PATH AS 'File	in the Device'						
23									
24 FROM	messages								
24 FROM 25									
24 FROM 25 26 Li	EFT JOIN .		ssage_media.MESSAGE_1						
24 FROM 25 26 Li 27 Li	EFT JOIN H	a.wa_contacts ON w	a.wa_contacts.jid = 1						
24 FROM 25 26 Li 27 Li 28 WH	EFT JOIN m EFT JOIN w ERE messag	wa.wa_contacts ON wa mes.key_remote_jid N	a.wa contacts.jid = 1 NOT like '-1'						
24 FROM 25 26 Li 27 Li 28 WH 29 OR	EFT JOIN m EFT JOIN w ERE messag	a.wa_contacts ON w	a.wa contacts.jid = 1 NOT like '-1'						
24 FROM 25 26 Li 27 Li 28 WH 29 OR	EFT JOIN m EFT JOIN w ERE messag	wa.wa_contacts ON wa mes.key_remote_jid N	a.wa contacts.jid = 1 NOT like '-1'						
24 PROM 25 26 Li 27 Li 28 WH 29 OR 30	EFT JOIN m EFT JOIN w ERE messag DER BY mes	va.wa_contacts ON w ges.key_remote_jid 1 ssages.timestamp AS0	a.wa_contacts.jid = 1 NOT like '-1' C	messages.KEY_REMOTI	E_JID	1.000 100			
24 PROM 25 26 Li 27 Li 28 WH 29 OR 30	EFT JOIN m EFT JOIN w ERE messag	va.wa_contacts ON w ges.key_remote_jid 1 ssages.timestamp AS0	a.wa contacts.jid = 1 NOT like '-1'			Message Type	Media Message	Whatsapp server media url	Media Decryption key
24 PROM 25 26 Li 27 Li 28 WH 29 OR 30	EFT JOIN m EFT JOIN w ERE messag DER BY mes	wa.wa_contacts ON w. pes.key_remote_jid 1 ssages.timestamp AS4 Message Date	A.wa_contacts.jid = 1 NOT like '-1' C Message Direction	messages.KEY_REMOTI	E_JID	Message Type text	Media Message	Whatsapp server media url	Media Decryption key
24 FROM 25 26 Li 27 Li 28 WH 29 OR 30 Jser Name	EFT JOIN m EFT JOIN w ERE messag DER BY mes one Numb +30694	wa.wa_contacts ON w. pes.key_remote_jid 1 ssages.timestamp AS4 Message Date	A.wa contacts.jid = n NOT like '-1' C Message Direction The Message was sent	messages.KEY_REMOTI	g_JID Message				Media Decryption key
24 PROM 25 26 Li 27 Li 28 WH 29 OR 30 Jser Name 1 Ιχάλης	EFT JOIN m EFT JOIN w ERE messag DER BY mes one Numb +30694 +30694	Message Date 2021-11-14 18:25:40 2021-11-14 18:46:59	A.wa contacts.jid = n NOT like '-1' C Message Direction The Message was sent	Date was Read	Message MULL This is the third	text	NULL	NULL	Media Decryption key
24 PROM 25 26 Li 27 Li 28 WH 29 OR 30 J Jser Name 1 1ιχάλης 2 1ιχάλης 3 1ιχάλης	EFT JOIN m EFT JOIN w ERE messag DER BY mes one Numb +30694 +30694	Message Date 2021-11-14 18:25:40 2021-11-14 18:46:59	A.va contacts.jid = P NOT like '-1' Message Direction The Message was sent Message Received The Message was sent	Date was Read	Message MULL This is the third	text text	NULL NULL NULL	NULL NULL	Media Decryption key
24 PROM 25 26 LL 27 LL 28 WH 29 OR 30 Jser Name 1 ⁴ Ιχάλης 2 ⁴ Ιχάλης 3 ⁴ Ιχάλης 4 ad Actor 1	EFT JOIN m EFT JOIN m ERE messag DER BY mes one Numb +30694 +30694 +30694	A. va_contacts ON va pes.key_remote_11d l ses.key_remote_11d l Message Date 2021-11-14 18:25:40 2021-11-14 18:46:59 2021-11-14 18:48:23 2021-11-17 10:47:27	A.va contacts.jid = P NOT like '-1' Message Direction The Message was sent Message Received The Message was sent	Date was Read NULL 2021-11-14 18:49:00 NULL	Message NULL This is the third This is the fourth	text text	NULL NULL NULL	NULL NULL NULL	
24 FROM 25 26 27 28 WE 29 OR 30 30 30 30 30 30 30 30 30 30 30 30 30	EFT JOIN m EFT JOIN w ERE messag DER BY mes one Numt +30694 +30694 +30694	ra.va_contacts ON wa pres.key_remote_11d l seges.timestamp AS Message Date 2021-11-14 18:25:40 2021-11-14 18:46:59 2021-11-14 18:48:23 2021-11-17 10:47:27 2021-11-17 10:47:29	a wa_contacts.jid = 1 NOT like '-1' Message Direction The Message was sent Message Received The Message was sent Message Received	Date was Read NULL NULL NULL NULL NULL NULL	Message NULL This is the third This is the fourth NULL NULL	text text text image	NULL NULL NULL This is an elephant NULL	NULL NULL NULL https://mmg.whatsapp.net/d/f/	

Figure 74. SQL query combining several columns from three different tables in two different databases.

A misperception is that since the WAL's checkpoint is at one thousand (1.000) commits there is "plenty of room" for many messages to be stored. Examinations showed that by sending one plain text message there were fifteen (15) different commits (**Figure 75**) to the "msgstore" database and by sending one media file fifty (50) commits occurred (**Figure 76**).

🙀 KS DB Merge Tools for SQLite - Pro v 1.5.1 - Trial - H	\Thesis\Whatsapp\San	isung A505\Database_cop	oying\first_test - Copy\0.clea	an_database\msgstor	e.db vs H:\Thesis\Wh	atsapp\Samsung A505\Datab	ase_copying\first_tes	: - Copy\2.first_message_re	ad\msgstore.
rs 🕫 🖿 🔳 🖪 🖁					1 🖪 🖪 🛛	2 🖉 🛠 👬			
😭 Project 🗴 🔚 Batch Data Diff 🗴									
Run () Only total record count (fast () Total + new counts (slower, N () All - total, new and changed	/A for views)	N/A for views)	✓ Show tables Show views						
Table name	Type	Changed columns	Common primary key	Total (left)	Total (right)	New (left) New	right) Chang	ed Error	
Chat	Table	No	Yes	1	2	0	1	0	
message_ftsv2	Table	No	No	0	1	0	1	0	
message ftsv2 content	Table	No	Yes	9	1	0	1	0	
message ftsv2 docsize	Table	No	Yes	C	1	0	1	0	
message ftsv2 segdir	Table	No	Yes	0	1	0	1	0	
message ftsv2 stat	Table	No	Yes	0	1	0	1	0	
messages	Table	No	Yes	1	3	0	2	0	
primary device version	Table	No	Yes	C	1	0	1	0	
props	Table	No	Yes	15	17	0	2	0	
sqlite sequence	Table	No	No	6		0	0	3	

Figure 75. Differences in the "msgstore.db" after the receipt of one message (tool used: KS DB Merge Tools).

78 P3 T6 TE II. L2	H		- * * 7				🚄 🛃 🔗		()	
Project 🗴 💼 Batch Data Diff 🗴										
Run Only total record count O Total + new counts (slo All - total, new and ch	wer, N/A	for views)	Sh	ow tables 👎 ow views						
Table name	Type	Changed columns	Common primary key	Total (left)	Total (right)	New (left)	New (right)	Changed	Error	
chat	Table	No	Yes	1	3		0	2	0	
frequents	Table	No	Yes	(1		0	1	0	
jid	Table	No	Yes		10		0	3	0	
message ftsv2	Table	No	No	(4		0	4	0	
message ftsv2 content	Table	No	Yes	(4		0	4	0	
message ftsv2 docsize	Table	No	Yes	(4		0	4	0	
message ftsv2 segdir	Table	No	Yes	(4		0	4	0	
message ftsv2 stat	Table	No	Yes	(1		0	1	0	
message media	Table	No	Yes	<u>(</u>	2		0	2	0	
message_streaming_sidecar	Table	No	Yes	9	2		0	2	0	
message thumbnails	Table	No	No	(2		0	2	0	
messages	Table	No	Yes	1	1		0	6	0	
primary device version	Table	No	Yes	9	2		0	2	0	
props	Table	No	Yes	13	20		0	5	0	
receipt device	Table	No	Yes	9	2		0	2	0	
receipt user	Table	No	Yes	(2		0	2	0	
sglite seguence	Table	No	No		9		0	3	4	

Figure 76. Differences in the "msgstore.db" after sending one media file (tool used: KS DB Merge Tools).

5.1.4. Volatility of the "msgstore.db"

The experiments confirmed that both the "msgstore" database and its WAL file were extremely volatile. However, certain actions/statuses/time did not seem to alter the data inside the "messages" or "message_media" tables. Those tested circumstances were:

- ✓ Leaving the device powered on and having enabled network connections.
- ✓ When device's boot sequence finished (reboot/restart) and the application auto starts.
- ✓ At opening of the application after closing it.
- ✓ Every few hours even in airplane mode.
- ✓ Leaving the device in a faraday box with or without network.

The database's "messages" or "message_media" tables were modified when:

- ! a message was created. It did not matter if it was successfully sent or not.
- ! a message was received.
- **!** a message was opened. This was regardless of whether it was an outgoing message which was read by the receiver or an incoming which was read by the user.

! a message or a thread was deleted.

During the experiments the devices did not receive any bot messages (ads). Another finding was that the incoming message notifications did not modify the "messages" table as it did on the examination of the Viber application. The record of one message was present in the WAL file multiple times due to its state changes (sent, delivered, read) or it's status alterations (live, deleted). So, there were cases where a message record was created, sent and deleted and never committed to the main database. All these records resided only in the WAL file.

5.1.5. Retrieving the messages in question

At the time of the experimentation phase WhatsApp's implementation for messages autodeletion was a predefined period of time of seven (7) days. This was the only available option and because of the long duration of this feature it was not taken under consideration. The reason was that in an DFIR scenario the procedures and actions are supposed to occur in a short period of time. As already mentioned, it is very difficult for the forensic tools, free or commercial, to keep up with the applications' settings and databases updating speed. The automated parsing, of the forensics tools, specifically regarding the WAL file, still had different levels of success as is depicted in **Figures 77** to **79**. In general, in order to verify and preview results, related to the communication with a specific contact, the searchable string of its WhatsApp ID (column "key_remote_jid") was used in a Hex Viewer (e.g., X-Ways Forensics) (**Figure 80**).

ų	Offset	commit	dbpage	walframe	salt1	salt2	🔑 _id	key_rem	key_from	key_id	status	needs_pu	data	timestamp
7	01E011	true	249	29	772401287	2520792042	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
9	02D179	true	249	44	772401287	2520792042	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
2	03C2E1	true	249	59	772401287	2520792042	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
8	0443A1	true	249	67	772401286	1649287693	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
5	053509	true	249	82	772401286	1649287693	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
	05F629	true	249	94	772401286	1649287693	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
	06E791	true	249	109	772401286	1649287693	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
	00EEA9	true	249	14	772401287	2520792042	149	306986605	0	3EB0A149	0	0	This is message No.64	Sat Nov 20 11:50:26 EET 2021
6	01DF8D	true	249	29	772401287	2520792042	150	306986605	0	3EB02DCA	0	0	This is message No.65	Sat Nov 20 11:50:29 EET 2021

Figure 77. Parsing messages' content in "msgstore.db". The tool failed to parse any of the records in question (tool used: FQLite, Examined Device: Samsung A50).

20/11/2021 11:50:49 A	20/11/2021 11:50:49 A			This is message No.70	text
20/11/2021 12:20:02 PM	20/11/2021 12:20:02 PM			Yes send the second	text
20/11/2021 12:20:38 PM	20/11/2021 12:20:38 PM			We agreed on five hu	text
20/11/2021 12:20:50 PM	20/11/2021 12:20:50 PM			WE WILL SEE	text
20/11/2021 2:05:42 PM	20/11/2021 2:05:42 PM			This is message No.81	text
20/11/2021 2:05:42 PM	20/11/2021 2:05:42 PM			This is message No.81	text
20/11/2021 2:17:35 PM	20/11/2021 2:17:35 PM			This is message No.82	text
20/11/2021 2:17:35 PM	20/11/2021 2:17:35 PM			This is message No.82	text
20/11/2021 2:44:36 PM		20/11/2021 2:44:38 PM	20/11/2021 2:44:36 PM	This is message No.85	text
20/11/2021 5:43:34 PM		20/11/2021 5:43:35 PM	20/11/2021 5:43:34 PM	This is message No.86	text
20/11/2021 5:43:58 PM	20/11/2021 5:43:59 PM			This is message No.87	text
20/11/2021 7:12:53 PM	20/11/2021 7:12:54 PM			This is messagge No.91	text
20/11/2021 7:12:53 PM	20/11/2021 7:12:54 PM			This is messagge No.91	text

Figure 78. Parsing messages' content in "msgstore.db". The tool managed to retrieve/parse some of the records in question (tool used: Magnet Axiom, Examined Device: Samsung A50).

ructure								
Record type	key_rer	key_from_me	key_id	status	needs_push	data	↓ ₹ timestamp	media_url
Wal	30694	0	3EB0CB207C50C9F62	0	0	Yes send the second o	20/11/2021 12:20:02	
Wal	30694	1	2610587DBDA7E836I	13	0	Did you receive it?	20/11/2021 12:19:45	
Wal	30694	1	4D649768FCB0B1AAI	13	0		20/11/2021 12:19:33	https://mmg.whatsa
Wal	30694	1	160E872777521405C	13	0		20/11/2021 12:20:11	https://mmg.whatsa
Wal	30694	1	7ACD3648BD21ADF6	13	0	This will cost you 1000	20/11/2021 12:20:28	
Wal	30694	0	3EB0D01EA8F933458	0	0	We agreed on five hur	20/11/2021 12:20:38	
Wal	30694	1	76439CC22CB756B47	13	0	I want more	20/11/2021 12:20:45	
Wal	30694	0	3EB083A83BBB8CEBC	0	0	WE WILL SEE	20/11/2021 12:20:50	
Wal	30694	1	777DAED4AD24CAA5	0	0	I think thet saw mel r	20/11/2021 12:21:21	
Wal	306941	1	5BBAAA5B96E8813D	13	0	This is message No.86	20/11/2021 5:43:34 P	
Wal	30694	0	523F0A099E5AB0565	0	0	This is message No.87	20/11/2021 5:43:58 P	
Wal	30698	1	A5656100BB62EFE56	13	0	This is message No.85	20/11/2021 2:44:36 P	

Figure 79. Parsing messages' content in "msgstore.db". The tool was able to retrieve all the records in question (tool used: Belkasoft-X, Examined Device: Samsung A50).

hys. offs. L	.og. offs	i. Y	Des				hits									-	Name	YT.	ype TCreate	l ⁱ Hit cou	Term count
	73A3	5 CF	125	2	A9E	C2ED	5720	2 Did	you	recei	ve it	?}<(JèH0	ÿ } <l< th=""><th>Jæ</th><th></th><th>] msgstore.db-wal</th><th>db-v</th><th>val</th><th>1</th><th>1</th></l<>	Jæ] msgstore.db-wal	db-v	val	1	1
Se Volum	ne [Fi	le		Prev	view	5	Detai	ls	6 U	Galle	ry		Caler	ndar		Legend 🔅 🔂 Sync	~	~ #4≡ ₫	₽≡ Φ≡	
Offset	0	1	2	3	4	5	6	7	8	9	A	В	С	D	Е	F	ANSI ASCII	~	msgstore.db-		
00073960	00 00	7E	00	06	00	00	00	20	8B	04	3E	85	84	A9	6C	67	~ < >,©lg`		C:\Users\dfir\l		100 message
00073970	D E4	E6	85	5A	DC	CB	BF	B7	СВ	B4	62	F8	95	ED	5D	6B	äæ…ZÜË¿∙Ë′bø•i]k		c.(oseis(diii))	Jesktop (20	
00073980	9F	D9	1A	2E	E8	3B	A8	08	70	70	FF	01	7D	3C	DB	BE	ŸÙ .è;" ppÿ }<Û¾		File size:		
00073990	D FO	01	7D	30	DB	BE	FO	01	7D	3C	DC	OD	10	42	71	38	ð }<Û%ð }<Ü Bq8				
000739A	34	4A	6C	46	6F	2F	72	49	46	31	74	4F	50	55	4C	51	4J1Fo/rIF1t0PULQ		Without slack		
000739B	50	53	72	6A	46	6F	4A	35	4B	36	50	67	4A	6E	4C	77	PSrjFoJ5K6PgJnLw				
00073900) 6E	59	52	71	4B	6B	77	30	3D	81	15	81	28	2C	00	43	nYRqKkw0= (, C		Read-only mo	de	
000739D	0 09	4D	01	08	33	05	00	00	OF	80	00	00	00	08	08	08	M 3		Creation time		
000739E	08	00	00	08	01	05	05	05	00	00	08	00	00	08	00	00			creation time.		
000739F	08	00	00	80	08	09	08	80	08	33	30	36	39	34	37	32	3069				
00073A0(33	32	37	37	32	40	73	2E	77	68	61	74	73	61	70	70	@s.whatsapp		Last write time	5	
00073A1() 2E	6E	65	74	32	36	31	30	35	38	37	44	42	44	41	37	.net2610587DBDA7				
00073A20	45	38	33	36	42	30	41	32	41	39	45	43	32	45	44	35	E836B0A2A9EC2ED5		Last access tin	ne:	
00073A3(37	32	44	32	OD	44	69	64	20	79	6F	75	20	72	65	63	72D2 Did you rec				
00073A4(65	69	76	65	20	69	74	3F	01	7D	3C	DB	E8	48	30	FF	eive it <mark>?</mark> }<ÛèHOÿ		Attributes:		
00073A50	0 01	7D	30	DB	E6	00	01	7D	3C	DB	E9	E8	01	7D	3C	DC	}<Ûæ }<Ûéè }<Ü		Attributes:		

Figure 80. Searching for messages content using "key_remote_jid" in "msgstore.db" (tool used: X-Ways Forensics, Examined Device: Samsung A50).

To monitor the experiments, course the "msgstore" database and its WAL file were combined into a new database. After having done that, several queries were executed. At first, the undeleted records could be detected in the WAL file after the messages were sent/received (**Figure 81**). The messages were characterized as "committed". This meant that when the one thousand (1000) commits-checkpoint was reached they would be moved to the main database.

1	SEI	LECT								
2		ssagesId AS								
3		_		AS 'User Name',						
4		ssages.sflsLiv								
5		ssages.sfSourc								
6		ssages.sfPage								
7 8				1000, 'unixepoch	', 'localtı	.me') AS 'Message Da	ite',			
9	E	ASE messages.	N 'Message F	ecoired!						
10	T		ssage was se							
11	END AS 'Message Direction',									
12	- RAD AS 'nessage Direction', messages.DATA AS Message,									
13	CASE messages.media_wa_type									
14	WHEN '0' THEN 'text'									
15										
16		WHEN '2' THE	N 'audio'							
17		WHEN '3' THE								
18			IN 'contact o							
19	P		IN 'geo locat	ion'						
20 21	-ENI	D AS 'Message	TAbe.							
21	FRO	OM messages								
23		ar messages								
24		LEFT JOIN mes	sage media C	DN message media.	MESSAGE URI	= messages.MEDIA_U	JRL			
25			_	_		sages.KEY REMOTE JI				
26	1	WHERE messages	.key_remote_	jid NOT like '-1	WHERE	wa.wa_contacts.dis	splay_name LIKE 'Μιχ	άλης'		
27	0	ORDER BY messa	ages.timestan	np ASC						
	A/A	User Name	Live Record	Record Location	Leaf Page	Message Date	Message Direction	Message	Message Type	
49	166	Μιχάλης	True	DB	250	2021-11-20 12:14:36	The Message was sent	This is message No.76	text	
50	167	Μιχάλης	True	DB	250	2021-11-20 12:14:39	The Message was sent	This is message No.77	text	
51	168	Μιχάλης	True	DB	250	2021-11-20 12:14:43	The Message was sent	This is message No.78	text	
52	169	Μιχάλης	True	DB	250	2021-11-20 12:14:47	The Message was sent	This is message No.79	text	
53	170	Μιχάλης	True	DB	250	2021-11-20 12:14:50	The Message was sent	This is message No.80	text	
54	171	Μιχάλης	True	DB	250	2021-11-20 12:22:01	The Message was sent	I am sending the photo	text	
55	172	Μιχάλης	True	WAL-10-commit	251	2021-11-20 12:22:09	The Message was sent	NULL	image	
56	173	Μιχάλης 📒	True 📒	WAL-10-commit	251	2021-11-20 12:22:24	The Message was sent	Did you receive it?	text	
57	174	Μιχάλης	True	WAL-10-commit	251	2021-11-20 12:22:43	Message Received	Yes send the second one	text	
58	175	Μιχάλης	True	WAL-10-commit	251	2021-11-20 12:22:50	The Message was sent	NULL	image	
59	176	Μιχάλης	True	WAL-10-commit	251	2021-11-20 12:23:06	The Message was sent	This will cost you 1000 euros	text	
60	176	Μιχάλης	False	WAL-20-commit	251	2021-11-20 12:23:06	The Message was sent	This will cost you 1000 euros	text	
61	176	Μιχάλης	False	WAL-21-commit	251	2021-11-20 12:23:06	The Message was sent	This will cost you 1000 euros	text	

Figure 81. The undeleted records after transmission using SQL queries in a reconstructed database made from "msgstore.db" and its WAL file (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50).

When the messages thread was deleted, and after 2 hours with no activity by the user, it was still feasible to detect them. Their status had been altered ("Live Record Status" and "Record Location" columns) (**Figure 82**).

_	A/A	User Name	Live Record	Record Location	Leaf Page	Message Date	Message Direction	Message Me	ssage Type
5	172	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:22:09	The Message was sent	NULL	image
5	173	Μιχάλης	False 🗧	WAL-10-commit	251	2021-11-20 12:22:24	The Message was sent	Did you receive it?	text
7	174	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:22:43	Message Received	Yes send the second one	text
8	175	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:22:50	The Message was sent	NULL	image
9	176	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:23:06	The Message was sent	This will cost you 1000 euros	text
0	177	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:23:19	Message Received	We agreed on five hundred	text
1	178	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:23:26	The Message was sent	I wan more	text
2	178	Μιχάλης	False	WAL-66-commit	251	2021-11-20 12:23:26	The Message was sent	I wan more	text
3	178	Μιχάλης	False	WAL-67-commit	251	2021-11-20 12:23:26	The Message was sent	I wan more	text
4	178	Μιχάλης	False	WAL-68-commit	251	2021-11-20 12:23:26	The Message was sent	I wan more	text
5	178	Μιχάλης	False	WAL-75-commit	251	2021-11-20 12:23:26	The Message was sent	I wan more	text
6	179	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:23:42	Message Received	WE WILL SEE	text
7	180	Μιχάλης	False	WAL-10-commit	251	2021-11-20 12:23:58	The Message was sent	I think they saw me. I must delete those	e text
8	180	Μιχάλης	False	WAL-0-commit	251	2021-11-20 12:23:58	The Message was sent	I think they saw me. I must delete those	e text
9	180	Μιχάλης	False	WAL-1-commit	251	2021-11-20 12:23:58	The Message was sent	I think they saw me. I must delete those	e text
0	180	Μιχάλης	False	WAL-8-commit	251	2021-11-20 12:23:58	The Message was sent	I think they saw me. I must delete those	e text
1	180	Μιχάλης	False	DB	251	2021-11-20 12:23:58	The Message was sent	I think they saw me. I must delete those	e text

Figure 82. How "msgstore.db" and its WAL file were modified after 2 hours with no activity by the user (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50).

After several messages were received, sent, read and/or deleted, the status of the inquestion messages inside the WAL had been changed ("Record Location" column) (**Figure 83**).

		· · · · · · · · · · · · · · · · · · ·	Record Location			Message Direction	Message	Message Type
175	Μιχάλης	False	WAL-114	251	2021-11-20 12:22:50	The Message was sent	NULL	image
176	Μιχάλης	False	WAL-114	251	2021-11-20 12:23:06	The Message was sent	This will cost you 1000 euros	text
177	Μιχάλης	False	WAL-114	251	2021-11-20 12:23:19	Message Received	We agreed on five hundred	text
178	Μιχάλης	False	WAL-114	251	2021-11-20 12:23:26	The Message was sent	I wan more	text
179	Μιχάλης	False	WAL-114	251	2021-11-20 12:23:42	Message Received	WE WILL SEE	text
180	Μιχάλης	False	WAL-114	251	2021-11-20 12:23:58	The Message was sent	I think they saw me. I must delete those	text
181	Edgar Allan Poe	True	DB	248	2021-11-20 14:05:35	Message Received	This is message No.81	text
182	Edgar Allan Poe	True	DB	248	2021-11-20 14:17:43	Message Received	This is message No.82	text
184	Μιχάλης	True	DB	248	2021-11-20 14:33:05	Message Received	This is message No.83	text
183	Μιχάλης	True	DB	248	2021-11-20 14:33:05	The Message was sent	NULL	text
185	Μιχάλης	True	DB	248	2021-11-20 14:33:08	Message Received	This is message No.84	text
186	Edgar Allan Poe	True	DB	248	2021-11-20 14:44:58	The Message was sent	This is messagge No.85	text
187	Bad Actor 2	True	DB	248	2021-11-20 17:43:34	Message Received	This is message No.86	text
188	Bad Actor 2	True	WAL-82-commit	249	2021-11-20 17:43:58	The Message was sent	This is message No.87	text
188	Bad Actor 2	False	DB	249	2021-11-20 17:43:58	The Message was sent	This is message No.87	text
188	Bad Actor 2	False	WAL-0-commit	249	2021-11-20 17:43:58	The Message was sent	This is message No.87	text
188	Bad Actor 2	False	WAL-98-commit	249	2021-11-20 17:43:58	The Message was sent	This is message No.87	text
189	Μιχάλης	True	WAL-82-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-26-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-27-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-28-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-35-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
190	Μιχάλης	True	WAL-82-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-54-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-55-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-56-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-63-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
191	Μιχάλης	True	WAL-82-commit	249	2021-11-20 18:03:58	Message Received	This is message No.90	text

Figure 83. The "msgstore.db" and its corresponding WAL file were modified after several messages received/sent/read/deleted (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50).

When a second deletion of a chat thread related to the same user was made, the two devices behaved differently because of their WAL file size. As mentioned in paragraph §5.1.1 the Samsung device's WAL size was able to retain the in-question messages (Figure 84), unlike the LG device where the same messages were removed (Figure 85). The difference in size and therefore the capacity for temporarily storing data in the WAL file of the two devices produces varied results. In the Samsung device the deleted records were preserved, even after the exchange of ten more messages than in the LG device. Only after twenty more messages were exchanged the in-question messages were also removed.

One more thing to notice is that in the Samsung device even though the in-question messages were retained (*records 167 to 175*), the was a removal of two more recently deleted records (*records 183 to 185*). So, there was an assumption about WAL file behavior, that confirmed several times. The assumption was that the application chose to remove as little data as possible from its database.

22	155	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 11:50:49	Message Received	This is message No.70	text
23	167	Μιχάλης	False	WAL-114	251	2021-11-20 12:19:32	The Message was sent	NULL	imag
24	168	Μιχάλης	False	WAL-114	251	2021-11-20 12:19:44	The Message was sent	Did you receive it?	text
25	169	Μιχάλης	False	WAL-114	251	2021-11-20 12:20:02	Message Received	Yes send the second one	text
26	170	Μιχάλης	False	WAL-114	251	2021-11-20 12:20:10	The Message was sent	NULL	imag
27	171	Μιχάλης	False	WAL-114	251	2021-11-20 12:20:28	The Message was sent	This will cost you 1000 euros	text
28	172	Μιχάλης	False	WAL-114	251	2021-11-20 12:20:38	Message Received	We agreed on five hundred	text
29	173	Μιχάλης	False	WAL-114	251	2021-11-20 12:20:44	The Message was sent	I want more	text
30	174	Μιχάλης	False	WAL-114	251	2021-11-20 12:20:50	Message Received	WE WILL SEE	text
31	175	Μιχάλης	False	WAL-114	251	2021-11-20 12:21:21	The Message was sent	I think thet saw meI must delete those	text
32	176	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 14:05:42	Message Received	This is message No.81	text
33	177	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 14:17:35	Message Received	This is message No.82	text
34	181	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 14:44:36	The Message was sent	This is message No.85	text
35	182	Bad Actor 1	True	WAL-59-commit	249	2021-11-20 17:43:34	The Message was sent	This is message No.86	text
36	183	Bad Actor 1	True	WAL-59-commit	249	2021-11-20 17:43:58	Message Received	This is message No.87	text
37	187	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 19:12:53	Message Received	This is messagge No.91	text
38	188	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 19:12:56	Message Received	This is messagge No.92	text
39	189	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 19:12:59	Message Received	This is messagge No.93	text
40	190	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 19:13:02	Message Received	This is messagge No.94	text
41	191	Edgar Allan Poe	True	WAL-59-commit	249	2021-11-20 19:13:05	Message Received	This is messagge No.95	text
42	193	Μιχάλης	True	WAL-59-commit	249	2021-11-20 19:15:32	Message Received	This is message No.96	text
43	192	Μιχάλης	True	WAL-59-commit	249	2021-11-20 19:15:32	The Message was sent	NULL	text
44	194	Μιχάλης	True	WAL-59-commit	249	2021-11-20 19:15:35	Message Received	This is message No.97	text
45	195	Μιχάλης	True	WAL-59-commit	249	2021-11-20 19:15:37	Message Received	This is message No.98	text
46	196	Μιχάλης	True	WAL-59-commit	249	2021-11-20 19:15:40	Message Received	This is message No.99	text
47	197	Μιχάλης	True	WAL-59-commit	249	2021-11-20 19:15:43	Message Received	This is message No.100	text

Figure 84. How "msgstore.db" and its WAL file are modified after a second deletion of a chat thread. The in-question messages were still retrievable. The messages 83 and 84 were removed (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: Samsung A50).

157	Edgar Allan Poe	True	DB	249	2021-11-20 11:51:38	Message Received	This is message No.67	text
158	Edgar Allan Poe	True	DB	249	2021-11-20 11:51:40	Message Received	This is message No.68	text
159	Edgar Allan Poe	True	DB	249	2021-11-20 11:51:43	Message Received	This is message No.69	text
160	Edgar Allan Poe	True	DB	249	2021-11-20 11:51:47	Message Received	This is message No.70	text
181	Edgar Allan Poe	True	DB	249	2021-11-20 14:05:35	Message Received	This is message No.81	text
182	Edgar Allan Poe	True	DB	249	2021-11-20 14:17:43	Message Received	This is message No.82	text
184	Μιχάλης	False	WAL-84-commit	248	2021-11-20 14:33:05	Message Received	This is message No.83	text
183	Μιχάλης	False	WAL-84-commit	248	2021-11-20 14:33:05	The Message was sent	NULL	text
185	Μιχάλης	False	WAL-84-commit	248	2021-11-20 14:33:08	Message Received	This is message No.84	text
186	Edgar Allan Poe	True	DB	249	2021-11-20 14:44:58	The Message was sent	This is messagge No.85	text
187	Bad Actor 2	True	DB	249	2021-11-20 17:43:34	Message Received	This is message No.86	text
188	Bad Actor 2	True	DB	249	2021-11-20 17:43:58	The Message was sent	This is message No.87	text
189	Μιχάλης	False	WAL-26-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-27-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-28-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-35-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
189	Μιχάλης	False	WAL-67-commit	249	2021-11-20 18:03:27	The Message was sent	This is message No.88	text
190	Μιχάλης	False	WAL-54-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-55-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-56-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-63-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
190	Μιχάλης	False	WAL-65-commit	249	2021-11-20 18:03:44	The Message was sent	This is message No.89	text
191	Μιχάλης	False	WAL-82-commit	249	2021-11-20 18:03:58	Message Received	This is message No.90	text
		1						

Figure 85. How "msgstore.db" and its WAL file alters after a second deletion of a chat thread. The messages of interest (between 70 and 81) were removed. The deleted messages 83 and 84 were still retrievable (tools used: Forensic Browser for SQLite & DB Browser for SQLite, Examined Device: LG G6).

As demonstrated the size, the type, and the complexity of the database, in comparison with Viber's one, works in favor of an DFIR team rather against it. The in-question messages were deleted after all the below actions were executed:

- i. receiving approximately thirty-three (33) new messages (forty-three (43) at the Samsung device),
- ii. reading fourteen (14) messages,
- iii. sending two (2) messages and
- iv. deleting five (5) additional messages.

These numbers are not absolute since different conditions and user actions could, slightly, modify these results. For example, the reading of an incoming message has a smaller impact (less commits) in the WAL file than creating and sending one. This means that there is a bigger margin for reading incoming messages than sending ones, until a WAL checkpoint is reached and so the in question deleted messages are permanently removed.

5.1.6. WhatsApp Backup

One of WhatsApp application's feature is the ability to store a local or cloud backup of the user's messages database in an encrypted form. The backups have file names ending with ".cryptNN" extension, where "NN" is the number of the encryption version such as "crypt5", "crypt7", "crypt12" (Ghannam H. A., 2018). The latest is "crypt14". The decryption of the backup requires the usage of the private key which is stored at the path «/data/data/com.whatsapp/files». There are many tools which have the ability to decrypt these backups, but not all of them can decrypt the latest versions. One solution for decrypting the "crypt14" backup version is depicted below (**Figure 86**).

ASUS-DESKTOP:/mnt/h/Thesis/Whatsapp/WhatsApp-Crypt14-Decrypter-master\$ file msgstore.db.crypt14 ASUS-DESKTOP:/mnt/h/Thesis/Whatsapp/WhatsApp-Crypt14-Decrypter-master\$ python3 decrypt14.py key msgstore.db.crypt14 msgstore.db Decryption of crypt14 file was successful. MagaSUS-DESKTOP:/mnt/h/Thesis/Whatsapp/WhatsApp-Crypt14-Decrypter-master\$ file msgstore.db msgstore.db: SQLite 3.x database, user version 1, last written using SQLite version 3022000 MagaSUS-DESKTOP:/mnt/h/Thesis/Whatsapp/WhatsApp-Crypt14-Decrypter-master\$

Figure 86. Decrypting a crypt14 backup with free tool «WhatsApp Crypt14 Database Decrypter» (ElDavoo D., 2021).

The backup can be executed either manually by the user or at a time interval. Before the backup the application committed all changes residing in the WAL file to the main database and then executed the "auto_vacuum" PRAGMA statement, leading to the permanent removal of all the deleted records. A more detailed reference on WhatsApp backup was made by Orr & Castro (2018), which includes the usage of a specific tool, named WhatsApp Key/DB Extractor (2016), to extract the key of the device's RAM. Unfortunately, this tool is obsolete and does not support newer versions of Android or WhatsApp. A downgrade procedure of the application is required.

5.2. WhatsApp's Media Files Attachments

The media transmission procedure was the same as it was explained in paragraph §4.2. In WhatsApp application when a chat thread including media, is deleted, these media files and their temporary created ones are also erased from the device's filesystem. WhatsApp transmitted files were stored at path: «/media/0/Android/media/com.whatsapp/WhatsApp/Media/WhatsApp Images» (Figure **87**), while their hashes were stored at path: «/media/0/Android/media/com.whatsapp/WhatsApp/.Shared» (Figure 88). There was also a folder at path: «/media/0/Android/media/com.whatsapp/WhatsApp/.trash», where deleted files were located (Figure 89). As depicted in below figures, the files are erased and only their record traces could be found.

■ YName▲	T Description	Туре	🝸 Size	TCreated T
□□. = Media (18)	existing		0 8	2021/11/14d18:10:2
🗆 🦲 . = WhatsApp Images (5)	existing		0 B	2021/11/14d18:10:2
Private (1)	existing		0 B	2021/11/14d18:10:2
🗆 🧰 Sent (2)	existing		0 B	2021/11/14d18:10:2
🔲 📈 IMG-20211117-WA0000.jpg	prev. existing, 1st cluster not availa	ıb jpg	0 B	2021/11/17d15:21:0
IMG-20211117-WA0001.jpeg	prev. existing, 1st cluster not availa	b jpeg	0 B	2021/11/17d10:49:1

Figure 87. The path that the WhatsApp transmitted files were stored (tool used: X-WAYS Forensic, Examined Device: LG G6).

TName 🔺	Description	Туре	🍸 Size	TCreated 🖘
🛄 = WhatsApp (29)	existing		83.5 KB	2021/11/14d18:10:2
🦳 = .Shared (4)	existing		0 B	2021/11/14d18:10:2
nomedia	existing, already viewed		0 B	2021/11/14d18:10:2
X 8+ ewC+ dk7dmHviGmGusWmfykvkSB3+LxxxGR8QqB5ko=.chk.tmp	prev. existing, file contents unknown	tmp		
8+ ewC+ dk7dmHviGmGusWmfykvkSB3+ LxxxGR8QqB5ko=.tmp	prev. existing, 1st cluster not availab	tmp	0 B	2021/11/17d15:21:0
QEsMKAr6s7kOmEfKwlxjhtVJ+Wdy-2nLJ9aUwJ6joQY=.enc.tmp	prev. existing, 1st cluster not availab	tmp	0 B	2021/11/17d10:49:0

Figure 88. The path that the WhatsApp transmitted files' hashes were stored (tool used: X-WAYS Forensic, Examined Device: LG G6).

\media\0\Android\media\com.whatsapp\WhatsApp\.trash\e07b3f0b-3cbf-42b1-94a7-12345c3e99f4									
■ ▼Name▲	Description	Type `	Y Size	▼ Created ¬					
□□. = .trash (1)	existing		0 B	2021/11/15d00:29:13					
e07b3f0b-3cbf-42b1-94a7-12345c3e99f4 (1)	existing		0 B	2021/11/17d10:47:33					
C X c35c80e7-67c4-4e72-b392-2968bfa64839	prev. existing, file contents unknown	1							

Figure 89. The WhatsApp media files trash folder (tool used: X-WAYS Forensic, Examined Device: LG G6).

5.2.1. Media persistence on WhatsApp Server

Even though the media files cannot be recovered locally, there is a possibility to do it remotely. After their transmission and encryption with a unique key, media are stored at WhatsApp media server to be received by any participant. The duration of the persistence on the server varies. Angus M. (2018) have made the hypothesis that if the encrypted media on the server are to be forwarded from user to user, the deletion timer is being reset. The files can be downloaded via a "wget" script or simply by using a web browser. In the conducted examinations it was possible to download files via WhatsApp server even after 36 hours of the original transmission time (Figure 90). As claimed in 2018 when the above paper was published, the media were saved at an unencrypted format. Nowadays they are encrypted with AES-256. The decryption key was found at the "message_media" table under the "media_key" column (Figure 91). This column was characterized as blob (thumbnails). By converting the contents of the column to a hex value and with the usage of a python script (i.e., "whatsapp-media-decrypt") the media was able to be retrieved (Figure 92). Unfortunately, even though the decryption key could be retrieved, after the deletion of the in-question messages it had a smaller persistence than the other records. The record was wiped after the exchange of just five (5) messages in any case (notice that the other rules about the non-alteration of the database still applied, like reboot, elapsed time, etc).

Q https://mmg.w	hatsapp.net/d/f/ApVWv	S-fuJNTyJ8MiWns8tmJ1unuc3E_GyhGFdbnwTiX.enc
ce your bookmarks here	on the bookmarks toolbar.	
or Browser?	Opening file.enc	×
et started.	from: https:/	reshark capture file (118 KB) //mmg.whatsapp.net
		Browser do with this file?
	Open with	Wireshark (default) v
	Save File	
		아. Cancel

Figure 90. Download transmitted files via WhatsApp server.

Message Date	Whatsapp server media url	Media Decryption key
2021-11-20 10:19:23	NULL	
2021-11-20 10:19:32	https://mmg.whatsapp.net/d/f/ApVWvS-fuJNTyJ8MiWns8tmJ1unuc3E_GyhGFdbnwTiX.enc	8B043E8584A96C67E4E6855ADCCBBFB7CBB462F
2021-11-20 10:19:44	NULL	
2021-11-20 10:20:02	NULL	
2021-11-20 10:20:10	https://mmg.whatsapp.net/d/f/AkgxHetqP2sU5jF5SNV554yA1gB8z0KK4C6ArDMMLWJE.enc	267D3EC39E5B0696E4E96CB9ED171F4AEB9EFBA

Figure 91. Media urls and decryption keys were stored on "message_media" table of "msgstore.db".

🛲 @ASUS-DESKTOP:/mnt/h/Thesis/Whatsapp/whatsapp-media-decrypt-main\$ python3 decrypt.py -j "88043E8584A96C67E4E6855ADCCBBFB7CBB462F895ED5D6B9FD91A2EE83BA808" file.enc
Decrypted
<pre>@ASUS-DESKTOP:/mnt/h/Thesis/Whatsapp/whatsapp-media-decrypt-main\$ file file.enc</pre>
file.enc: data
Action of the second se
file.jpg: JPEG image data, JFIF standard 1.01, aspect ratio, density 1x1, segment length 16, progressive, precision 8, 960x1280, components 3
wwwwwsos-beskipp://mit/n/inesis/wnatsapp/wnatsapp-media-decrypt-mains

Figure 92. Decrypting downloaded from WhatsApp server media files, by using python script named «whatsapp-media-decrypt».

5.2.2. LG device's media artifacts

In the LG device, even if there was no content of the media that were captured through the WhatsApp application's camera feature, thumbnails of those captured through the device's camera application and then sent via WhatsApp application could still be found and parsed at the path: «media\0\Android\data\com.android.gallery3d\cache\imgcache.0» (**Figure 93**).

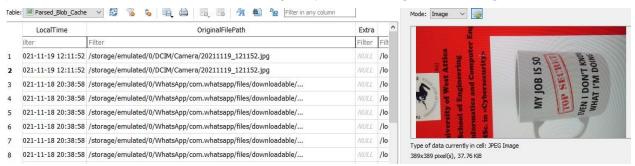


Figure 93. Viewing the "imgcache.sqlite" file (BLOB) (tool used: DB Browser for SQLite, Examined Device: LG G6).

5.2.3. Samsung device's media artifacts

The Samsung device has a modified version of stock android. It does not use the Android's default camera or photo gallery applications, but not the ones that were captured through the WhatsApp application's camera feature. The pictures' thumbnails are located in the Samsung gallery application at path: «/data/com.sec.android.gallery.3d/cache/». As presented in paragraph §4.2.1 analysis the "local" database located the of at the path: «/data/com.sec.android.gallery.3d/databases/» and specifically the "trash" table could provide the media deletion. "graph.db" database timestamps of In located at «/data/com.samsung.mlp/databases», it was still possible to track the media files which were sent as messages via the WhatsApp application (Figure 94).

ID	COL1	COL2	COL3	COL4	COL5	COL6	COL7	COL8	⊽ (
7	8		1	/storage/emulated/0/Pictures/Viber/IMG-81eac0eb87d2e3e132a5900fcd59593b-V.jpg	65537		2021/11/05 18:35:01	2021/11/05 18:34:2	8
5	7		1	/storage/emulated/0/Pictures/Viber/IMG-d80dfc7b3d8a0349e93cb1148ff4dbb4-V.jpg	65537		2021/11/05 18:34:04	2021/11/05 18:33:1	.9
1	3		1	/storage/emulated/0/DCIM/Camera/20211031_104226.jpg	65537		2021/10/31 08:43:06	2021/10/31 08:42:2	6
4	6		1	/storage/emulated/0/Pictures/Viber/IMG-f3e7c3cf3ef69c300ee42d9d0a36efdd-V.jpg	65537		2021/10/31 10:25:04	2021/10/31 10:24:2	6
3	5		1	/storage/emulated/0/DCIM/Camera/20211031_122307.jpg	65537		2021/10/31 10:23:56	2021/10/31 10:23:0	7
18	18			/storage/emulated/0/Android/media/com.whatsapp/WhatsApp/Media/WhatsApp Images/IMG-20211118-WA0001.jpeg	65537		2021/11/18 21:11:48	2021/11/18 21:11:0	6
15	17		1	/storage/emulated/0/DCIM/Camera/20211118_230817.jpg	65537		2021/11/18 21:08:58	2021/11/18 21:08:1	.7
14	16			/storage/emulated/0/Android/media/com.whatsapp/WhatsApp/Media/WhatsApp Images/IMG-20211117-WA0001.jpeg	65537		2021/11/17 10:50:01	2021/11/17 10:49:1	.1
13	14		1	/storage/emulated/0/DCIM/Screenshots/Screenshot_20211115-194127_One UI Home.jpg	65537		2021/11/15 20:40:26	2021/11/15 17:41:2	7
12	12		1	/storage/emulated/0/Pictures/Viber/IMG-133563d0e654758e1b2602a36af42559-V.jpg	65537		2021/11/15 13:51:14	2021/11/15 13:50:4	0
10	11		1	/storage/emulated/0/Pictures/Viber/IMG-71e92ae13d4db3ef81b9e5926049c3f3-V.jpg	65537		2021/11/15 13:49:11	2021/11/15 13:48:2	2
9	10		1	/storage/emulated/0/Pictures/Viber/IMG-a081b57754a8936d0401434b27bfcd3b-V.jpg	65537		2021/11/10 20:30:29	2021/11/10 10:47:1	1
8	9		1	/storage/emulated/0/Pictures/Viber/IMG-6759053542d2ed2a449eb370dd09f6a2-V.jpg	65537		2021/11/10 10:44:11	2021/11/10 10:43:4	3

Figure 94. Exploring "MediaAttribute" table of Samsung "graph.db" (tool used: Oxygen Forensic SQlite, Examined Device: Samsung A50).

5.3. Final Conclusions on WhatsApp

To sum up, it is impossible to avoid the modification of "msgstore" database's WAL file, however certain types of actions (e.g., reboot, closing/opening the application) does not seem to alter the data of the database's critical "messages" table. The usage of the WAL file as a journaling mode provides some leniency to the DFIR Team members, because as the experiments proved it was able to retain a sufficient amount of deleted data. However, there is a time pressure regarding a potential application's auto backup execution and the deletion of the media files from the media server.

Chapter 6.

Telegram application Scenario

In this chapter the Telegram application's examination and the related results are presented regarding text and media exchanged messages and the volatility of the database in which they are stored. Like in the two previous chapters there is a reference on the application database's location, it's features and the differences between an Android 9 and an Android 11 examined device. After that there is an analysis of the tables and columns of the database and the queries that were built in order to view the desired ones. In addition, the volatility of deleted text and media messages is examined through multiple experiments. Furthermore, some unique characteristics of the application are examined, and final conclusions are drawn.

6.1. Telegram's Examination

The used .apk for both the devices was the Telegram v.7.9.3 with released date in August of 2021. Application's data, file and folder structure are not as well-known as to other applications. Anglano et al. (2017) have made an excellent job in terms of educational and research spectre. The examined database for this application was the "cache4.db" located at the path: «data/data/org.telegram.messenger/files». In this scenario's experiments more than one thousand and one hundred (1.100) messages were exchanged in total for each device, among four users. The different approach which was taken this time, was related to the creation and existence of freelist pages in the database and the need to study them. This was achieved by creating a large amount of data traffic inside the application's database. The access to these files and folders is normally prohibited to the user, so either the examiner must have root access to the device or acquire a physical image of it.

6.1.1. Android 9 vs Android 11 Comparison

There were no differences in the folder structure or the files of the application on both examined devices. In any case the initial size of "cache4.db" was 1.336 kb and its corresponding WAL file was 4.475kb.

6.1.2. General information about the under-examination database

The header and PRAGMA statements of the "cache4" database analysis, by using the non-forensic tools "SQL Expert" and "DB Browser for SQLite" showed that (**Figure 95**):

- ✓ created freelist pages since the auto_vacuum feature was disabled,
- ✓ used WAL as a journaling mode,
- ✓ the WAL Auto Checkpoint had been set at one thousand (1.000) commits (default size).

Γ	Name	Value	Auto Vacuum 😟	None 🗸 🗸
•	application_id]0	Automatic Index 😣	
	auto_vacuum	none	Case Sensitive Like 😣	
Γ	cache_size	-2000	Checkpoint Full FSYNC 🥹	
F	collation_list	[BINARY], [NOCASE], [RTRIM]	Foreign Keys 🥹	
F	encoding	UTF-8	Full FSYNC 😣	
F	foreign_keys	on	Ignore Check Constraints 🥹	
F	freelist count	0	Journal Mode 😡	WAL V
F	journal_mode	wal	Journal Size Limit 🥹	-1 🔹
┢	journal_size_limit	-1	Locking Mode 😡	Normal V
┝		1073741823	Max Page Count 😉	1073741823 🗘
⊢	max_page_count		Page Size 😣	4096 🗸
L	mmap_size	0	Recursive Triggers 😣	
	page_count	334	Secure Delete 🥹	
	page_size	4096	Synchronous 😣	Full 🗸
Γ	schema_version	87	Temp Store 😣	Default $$
	user_version	80	User Version 😣	80 ≑
	writable_schema	off	WAL Auto Checkpoint 😟	1000 ≑

Figure 95. Properties of "cache4.db" (tools used: SQLite Expert (left) & DB Browser for SQLite (right)).

6.1.3. Analysis of the "cache4.db"

The "cache4" database was composed by fifty-two (52) tables (**Figure 96**). The six (6) most forensically interesting tables for the examined scenario were the "**dialogs**", "**enc_chats**", "**messages**", "**user_contacts_v7**", "**user_phones_v7**" and "**users**".

Among the most interesting columns of the "dialogs" table were:

- The "did", contained the dialog id of the secret or normal chat.
- The "date", contained the timestamp of a chat-thread's last message in Unix Seconds.
- The "unread count", contained information about the unread messages of the dialog.
- The "last_mid", "inbox_max" and "outbox_max" which in conjunction could clarify the size of the dialog.

Under the "**enc_chats**" table the "**uid**", "**user**" and "**name**" columns were the most valuable. The records included in these columns had to be compared with some of the records on the "**messages**" table to get information about the participants of a secret chat.

Among the most interesting columns of the "messages" table were:

- The "uid", contained the participant of a normal or a secret chat.
- The "read_state", had information about the message's read status.
- The "send_state", clarified whether the outgoing message was actually sent (i.e., network unavailability).
- The "date", contained the timestamp of the message in Unix Seconds.

- The "body" contained the body of the text message, along with several other information, in a binary serialized form. From now on, these types of data will be referred as "TDS" (Telegram Data Structure). This term was introduced by Anglano et al. (2017).
- The "out", contained information about the message direction.
- The "**TTL**", kept the timer set by the user for autodeleting the message.

The "**user_contacts_v7**", "**user_phones_v7**" and "**users**" tables contained information about the users participating in chat threads.



Figure 96. Depiction of the tables tree of "cache4_db".

The form of data inside the TDSs was an obstacle to preview results from a created SQL query. (**Figure 97**).

SELECT							0000 d2 83 e3 bc 02 03 00 0 0010 43 1c 71 71 6d bc b1 9	0 34 00 00 00 6d d a3 79 a8 72 d5
users.name A								3 20 61 20 73 6e
users.data A							0030 d7 50 51 69 01 00 00 0	0 65 7a 19 fb 00
		, 'unixepoch', '	localtime") >	AS 'Date',			0040 4d b9 31 1b de 1d 09 5	
E WEEN '1'	' THEN 'Not						0050 19 03 00 00 00 34 61 a	
LEND AS 'Mean							0060 cl 27 ff 61 8c 7l la 4	7 8d 03 00 00 d5
FICASE message		1					0080 95 01 17 28 cf 8a 12 e	b bc fd da 95 e3
	THEN 'Inco	ming*					0090 b8 27 d7 b5 25 a4 a4 3	
BLSE 100	atgoing!						00-0 -C 0- 0+ C2 52 44 b4 4	1 En 16 mil no 9h
LEND AS 'Mess								
ECASE message							Тура	
	THEN 'Non' THEN 'Rece						432 byte(s)	
END AS 'Ness							Remote	
messages.dat							hemore	
FROM message							Identity Select an identity to connect ~	
		<pre>rs.uid = messages sta) LIKE*D283E3B</pre>						
TWHERE hext	(nessages.da	ata) LINE'FA55555	St! Secret b	messages Messages			DBHub.io Local Current Database	
GRDER BY mes							63 %	
Name	Info	Date N	4essage Status	Message Direction	Message status	Body	Name	Last mod
				Message Direction Outgoing	Message status Non Received	Body	Name	Last mod
edgar allan poe;;;	; 82.08 2021-	-11-25 10:14:07	yu.			10.00	Name	Last mod
edgar allan poe;;; edgar allan poe;;;	; 8008 2021- ; 8008 2021-	-11-25 10:14:07 //	W.L.	Outgoing	Non Received	BLOB	× Name	Last mod
edgar allan poe;;; edgar allan poe;;; edgar allan poe;;;	; 81.08 2021 ; 81.08 2021 ; 81.08 2021 ; 81.08 2021	-11-25 10:14:07 //	at at at	Outgoing Outgoing	Non Received Non Received	BLOB BLOB	* Name	Last mod
edgar allan poe;;; edgar allan poe;;; edgar allan poe;;; edgar allan poe;;; bad actor 2;;;	\$ 81.08 2021- \$ 81.08 2021- \$ 81.08 2021- \$ 81.08 2021- \$ 2021-	-11-25 10:14:07 /// -11-25 10:14:14 // -11-25 10:14:22 //	nt nt: nt	Outgoing Outgoing Outgoing	Non Received Non Received Non Received	BLOB BLOB BLOB	A Nane	Last mod
edgar allan poe;;; edgar allan poe;;; edgar allan poe;;; edgar allan poe;;; bad actor 2;;; bad actor 2;;;	\$ \$6.08 2021 \$ \$6.08 2021 \$ \$6.08 2021 \$6.08 2021 \$6.08 2021 \$6.08 2021	-11-25 10:14:07 // -11-25 10:14:14 // -11-25 10:14:22 // -11-26 11:20:21 //	nt nt: nt nt nt	Outgoing Outgoing Outgoing Outgoing	Non Received Non Received Non Received Received	BLOB BLOB BLOB BLOB	n Hane	Last mod
1 edgar allan poe;;; 2 edgar allan poe;;; 3 edgar allan poe;;; 4 bad actor 2;;; 5 bad actor 2;;; 6 //////////////////////////////////	6 000 2021- 6 000 2021- 7 000 2021- 8 000 2021- 8 000 2021- 10 00 2021- 10 000 2000- 10 000 2000- 10 000 2000- 10 0000- 10 0000	-11-25 10:14:07 /// -11-25 10:14:14 /// -11-25 10:14:22 // -11-26 11:20:21 // -11-26 11:21:09 // -11-26 11:22:10 // -11-26 11:22:10 //	URE URE: URE URE URE URE URE URE URE	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing	Non Received Non Received Non Received Received Received Received Received	BLOB BLOB BLOB BLOB HLOB BLOB	n Hane	Last mod
 edgar allan poe;;;; edgar allan poe;;;; edgar allan poe;;;; bad actor 2;;; bad actor 2;;; <i>Mati</i> <i>Mati</i> bad actor 2;;; 	8/09 2021- 6/09 2021- 7/09 2021- 8/09 2021- 8/09 2021- 7/01 2021- 7/01 2021- 7/01 2021- 8/09 2021-	-11-25 10:14:07 -11-25 10:14:14 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:21:09 -11-26 11:22:10 -11-26 11:22:43 -11-26 11:26:24	NA NA NA NA NA NA NA NA	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming	Non Received Non Received Non Received Received Received Received Received Received	BLOB BLOB BLOB BLOB HLOB BLOB HLOB	A Nane	Last mod
1 edgar allan poe;;;; 2 edgar allan poe;;; 3 edgar allan poe;;; 4 bad actor 2;;; 5 bad actor 2;;; 6 ////////////////////////////////////	8008 2021- 1008 2021- 1008 2021- 1008 2021- 1009 2021- 1009 2021- 1009 2021- 1009 2021- 1009 2021-	-11-25 10:14:07 -11-25 10:14:14 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:22:10 -11-26 11:22:10 -11-26 11:22:43 -11-26 11:56:24 -11-26 11:56:24	RE RE RE RE RE RE RE RE RE RE RE RE RE R	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming	Non Received Non Received Non Received Received Received Received Received Received Received	BLOB BLOB BLOB BLOB MLOB BLOB MLOB BLOB	n Hane	Last mod
1 edgar allan poe;;;; 2 edgar allan poe;;;; 3 edgar allan poe;;;; 4 bad actor 2;;; 5 bad actor 2;;; 6 MEL 7 MEL 8 bad actor 2;;; 9 bad actor 2;;; 9 bad actor 2;;;	80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021-	-11-25 10:14:17 -11-25 10:14:14 -11-25 10:14:12 -11-26 11:20:21 -11-26 11:20:01 -11-26 11:21:09 -11-26 11:22:40 -11-26 11:22:43 -11-26 11:26:43 -11-26 11:56:24 -11-26 11:57:11 -11-26 11:57:56	RE RE RE RE RE RE RE RE RE RE RE RE RE	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming Incoming	Non Received Non Received Non Received Received Received Received Received Received Received	81.08 81.08 81.08 81.08 81.08 81.08 81.08 81.08 81.08 81.08 81.08	n Nane	Last mod
1 edgar allan poe;;;; 2 edgar allan poe;;; 3 edgar allan poe;;; 4 bad actor 2;;; 5 bad actor 2;;; 6 ///.2. 7 ///.2. 9 bad actor 2;;; 1 ///.2.	80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021-	-11-25 10:14:07 -11-25 10:14:14 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:20:21 -11-26 11:22:40 -11-26 11:22:43 -11-26 11:57:11 -11-26 11:57:56 -11-26 11:58:37	RE RE RE RE RE RE RE	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming Incoming	Non Received Non Received Non Received Received Received Received Received Received Received Received	8L08 8L08 8L08 8L08 8L08 8L08 8L08 8L08	n Nane	Last mod
1 edgar allan poe;;;; 2 edgar allan poe;;; 3 edgar allan poe;;; 4 bad actor 2;;; 5 bad actor 2;;; 6 max 7 Mcd. 8 bad actor 2;;; 9 bad actor 2;;; 9 bad actor 2;;; 9 max 1 max 1 max 1 max 2 edgar allan poe;;;	8008 2021 9008 2021 8008 2021 8008 2021 8008 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021 8009 2021	-11-25 10:14:07 -11-25 10:14:14 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:20:21 -11-26 11:22:40 -11-26 11:22:40 -11-26 11:22:41 -11-26 11:57:11 -11-26 11:57:56 -11-26 11:58:37 -11-27 12:24:16 Me	004 004 004 004 004 004 004 004 004 004	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming Incoming Outgoing	Non Received Non Received Non Received Received Received Received Received Received Received Received Non Received	BLOB BLOB BLOB BLOB BLOB BLOB BLOB BLOB	n Hane	Last mod
1 edger allen poe;;;; 2 edger allen poe;;;; 3 edger allen poe;;;; 5 bad actor 2;;;; 5 bad actor 2;;;; 9 bad actor 2;;;; 9 bad actor 2;;;; 9 bad actor 2;;; 9 bad actor 2;;; 9 MAL 7 MAL 7 MAL 2 edger allen poe;;; 3 edger allen poe;;;	90.08 2021- 90.08 2021- 90.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.08 2021- 80.09 2021- 80.09 2021- 80.09 2021- 80.09 2021- 90.04 2021- 90.04 2021- 90.04 2021- 90.04 2021-	-11-25 10:14:07 -11-25 10:14:14 -11-25 10:14:12 -11-26 11:20:21 -11-26 11:20:21 -11-26 11:22:10 -11-26 11:22:43 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:37 -11-26 11:57:37 -11-27 12:24:28 Net	DAL DAL DAL DAL DAL DAL DAL DAL DAL SEA X Send X Send	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming Incoming	Non Received Non Received Non Received Received Received Received Received Received Received Received	8L08 8L08 8L08 8L08 8L08 8L08 8L08 8L08	n Nane	Last mod
1 edger allen poe;;; 2 edger allen poe;;; 3 edger allen poe;;; 4 bad actor 2;;; 5 bad actor 2;;; 6 //dcs 7 //dcs 9 bad actor 2;;; 9 bad actor 2;;; <td< td=""><td>9408 2021* 9408 2021* 9408 2021* 8408 2021* 9408 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021*</td><td>-11-25 10:14:07 -11-25 10:14:12 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:20:21 -11-26 11:20:43 -11-26 11:22:43 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:13 -11-27 11:22:41:28 -11-27 12:24:28 No -11-27 12:24:28 No</td><td>024 024 024 024 024 024 024 024 024 024</td><td>Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming Incoming Outgoing</td><td>Non Received Non Received Non Received Received Received Received Received Received Received Received Non Received</td><td>BLOB BLOB BLOB BLOB BLOB BLOB BLOB BLOB</td><td>n Hane</td><td>Last mod</td></td<>	9408 2021* 9408 2021* 9408 2021* 8408 2021* 9408 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021*	-11-25 10:14:07 -11-25 10:14:12 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:20:21 -11-26 11:20:43 -11-26 11:22:43 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:13 -11-27 11:22:41:28 -11-27 12:24:28 No -11-27 12:24:28 No	024 024 024 024 024 024 024 024 024 024	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Incoming Incoming Incoming Outgoing	Non Received Non Received Non Received Received Received Received Received Received Received Received Non Received	BLOB BLOB BLOB BLOB BLOB BLOB BLOB BLOB	n Hane	Last mod
1 edger allen poe;;;; 2 edger allen poe;;; 3 edger allen poe;;; 4 bad actor 3;;; 5 bad actor 2;;; 6 MARL 7 MARL 8 bad actor 2;;; 9 bad actor 2;;; 9 bad actor 2;;; 9 MARL 7 MARL 2 edger allen poe;;; 3 edger allen poe;;;	9408 2021* 9408 2021* 9408 2021* 8408 2021* 9408 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021* 9409 2021*	-11-25 10:14:07 -11-25 10:14:12 -11-25 10:14:22 -11-26 11:20:21 -11-26 11:20:21 -11-26 11:20:43 -11-26 11:22:43 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:11 -11-26 11:57:13 -11-27 11:22:41:28 -11-27 12:24:28 No -11-27 12:24:28 No	024 024 024 024 024 024 024 024 024 024	Outgoing Outgoing Outgoing Outgoing Outgoing Outgoing Dutgoing Incoming Incoming Outgoing Outgoing Outgoing	Non Received Non Received Non Received Received Received Received Received Received Received Non Received Non Received	8L08 8L08 8L08 8L08 8L08 8L08 8L08 8L08	Anne -	Last mod

Figure 97. SQL query combining several columns from different tables from "cache4.db".

6.1.3.1 Analysis of TDSs

Anglano et al. (2017) explain that the serialized data are structured in the Telegram Language as composite types (include integers and strings) and defined by a 4-byte integer number in little Indian. Each 4-bytes integer comprise one object type. In April 2020 through the examination of 42.000 lines of Telegram code, 1.340 objects were found (dfirfpi, 2020). Telegram Team adds, removes or redefines these objects by adding or removing application's features. Boiko M. (2018) explains some of these objects, even though many of them have been redefined. Analysis of some of the TDSs and objects found in the examined application's version are presented below throughout examples. In the first example there is an exchange of two messages (one incoming and one outgoing) of a normal chat thread (records 162 and 163). In every TDS specific 4-byte integer (in little Indian) have a specific meaning (**Table 11**).

101		Nam	-		Info	202	-	ate				ge Sta	atus		essag		ection	Message	status	Body
162	bad a	ctor 2	2;;;	B	LOB	202	1-11-2	27 23	52:3	3 NU	NULL				Incoming			Received		
163	bad a	ctor 2	2;;;	В	BLOB 2021-11-27 23:53:31						NULL			Out	going			Received		BLOB
							R	ecora	ls inc	ludin	g nor	mal c	hat m	nessa	ges.					
						Ī	able:	users			\sim	2 1	5 👈		, 🖨	5				
							u	iid		n	ame			status	da	ita				
							Filter		Filter				Filt	ter	Filt	ter				
						1	19032	238211	zugni	feyda;	;;		16	380469	12 <i>BL</i>	ОВ				
						2	19236	543811	bad a	ctor 2;	;;			-1	00 <i>BL</i>	ОВ				
						3	21351	101684	μιχάλι	וי;;				-1	00 <i>BL</i>	ОВ				
						4	21389	80292	edgar	allan p	ooe;;;			-1	00 BL	ОВ				
		The "	users	s" tab	ole of	the '	'cach	e4.db	". The	e reco	ord 1	(devid	e's ι	user) a	and r	ecord	2 are t	the participa	ants.	
000	d 2	83	e3	bc	00	01	00	00	c5	00	00	00	6d	bc	bl	9d			m.	
010	a3	79	a8	72	6d	bc	bl	9d	a3	79	a8	72	al	a8	a2	61	• y	.rm	y.r	.a
020	16	54	68	69	73	20	69	73	20	6d	65	73	73	61	67	65	.T	nis is	messa	ge
	20	40	6F	20	31	36	30	00	01	20	00	00					N	.160		

						0	bject	jects in record 162 (Normal Incoming Message)									
BCE3	883D	2		Integer flagging this TDS as a normal message between two users													
72A8	79A	3		The hex value of the user's id sending the message in the conversation (1923643811													
72A8	79A	3		The	hex	valu	e of t	the u	iser'	s id p	partio	cipat	ing i	n the	e con	versa	ation (1923643811)
61A2	A8A	1		The	time	starr	np of	the	mes	sage	e (16	3804	1995	3) in	Uni	x Sec	conds
0000	d2	83	e3	bc	02	03	00	00	c6	00	00	00	6d	bc	bl	9d	m
0010	43	lc	71	71	6d	bc	bl	9d	a3	79	a8	72	db	a8	a2	61	C.qqmy.ra
0020	16	54	68	69	73	20	69	73	20	6d	65	73	73	61	67	65	.This is message
0030	20	4e	6f	2e	31	36	31	00	20	63	ed	3d	01	20	00	00	No.161. c.=
0040																	
						0	bject	s in re	ecora	163	(Norn	nal O	utgoi	ng M	essag	ge)	
BCE3	83D	2		Integ	jer fl	aggii	ng th	nis Tl	DS a	s a r	norm	al m	essa	age l	oetw	een t	wo users
7171	1C4	3		The hex value of the user's id sending the message in the conversation (1903238211)											conversation (1903238211)		
72A8	79A	3		The hex value of the user's id participating in the conversation (1923643811)													
61A2	A8D	В		The	time	stam	p of	thei	nes	sage	(16	3805	001	1) in	Unix	k Sec	onds

 Table 12. TDSs (Telegram Data Structure) objects in Normal Chat thread

In the second example there is an exchange of two messages (one incoming and one outgoing) of a secret chat thread (records 225 and 226). In every TDS specific 4-byte integer (in little Indian) have a specific meaning (**Table 12**).

	225	NUL	L		Λ	IULL	2021	-11-29	20:2	4:26	NULL			Out	going			Received	BLOB
	226	NUL	L		Λ	IULL	2021	-11-29	20:2	4:39	NULL			Inco	ming	I		Received	BLOB
I								Rec	ords	inclu	ding	secre	t cha	t mes	sage	s.			
							<u>T</u> able:	use	ers		``	, N	76	-	Q (
								uid			nam	e		stat	us	data			
								lter		ter				Filter		Filter	1		
								903238 923643						163804			J		
						1		135101								BLOB	1		
							U —	138980				;;				BLOB	J		
		Th	e "us	ers"	table	of th				-			vice	s use	r) an	nd rec	ord 3	are the participa	nts.
000	0 1			55	55	03	03	00	00	fd	ca	fc	ff	00	00	00	00	.000	
001	0 4	13	lc	71	71	6d	bc	bl	9d	f4	10	43	7f	da	la	a5	61	C.qqm	Ca
002	0 1	ld	54	68	69	73	20	77	69	6c	6c	20	63	6f	73	74	20	.This wi	ll cost
003	0 7	79	6f	75	20	31	30	30	30	20	65	75	72	6f	73	00	00	you 1000) euros
004	0 2	20	63	ed	3d	15	c4	b5	lc	00	00	00	00	00	00	00	00	c.=	
005	0						~			I	00F /								
				r			0	ojects	in re	cora	225 (3	secre	t Out	going	wes	sage)			_
555	555	FA		Inte	eger	flag	ging	this	TDS	as a	a sec	ret n	ness	age l	betv	veen	two ı	users	
717	11C	43		Th	e he	x val	ue c	of the	use	r's ic	lsen	ding	the	mess	sage	e in th	ne co	nversation (19	903238211)
7F4	310)F4		Th	e he	x va	ue c	of the	use	r's ic	l rece	eivinę	g the	mes	ssag	je in t	the c	onversation (2	2135101684)
61A	51A	DA		Th	e tirr	nesta	mp	of th	e me	essa	ge (1	6382	2102	66) iı	n Ur	nix Se	econ	ds	

0000	fa	55	55	55	01	03	00	00	fc	ca	fc	ff	00	00	00	00	.000
0010	f4	10	43	7f	6d	bc	bl	9d	43	1c	71	71	e7	1a	a5	61	C.mC.qqa
0020	19	57	65	20	61	67	72	65	65	64	20	6f	6e	20	66	69	.We agreed on fi
0030	76	65	20	68	75	6e	64	72	65	64	00	00	20	63	ed	3d	ve hundred c.=
0040	15	c4	b5	lc	00	00	00	00	00	00	00	00					
						C	bject	s in r	ecord	226	(Secr	et Ind	omin	g Me	ssag	e)	I
5555	55FA	A	lr	ntege	er flag	ggin	g this	s TD	S as	a se	cret	mes	sage	e bet	wee	n two	users
7F43	10F4	1	Т	he h	ex va	alue	of th	e us	er's i	d se	ndin	g the	mes	ssag	e in	the co	onversation (2135101684)
7171	1C43	3	Т	he h	ex va	alue	of th	e us	er's i	d rea	ceivii	ng th	e me	essa	ge ir	n the c	conversation (1903238211)
61A5	1AE	7	Т	he ti	nest	tamp	of th	ne m	essa	ige (1638	3210	279)	in U	nix S	Secon	ıds

 Table 13. TDSs (Telegram Data Structure) objects in Secret Chat thread.

In **Table 13** the objects of an outgoing message containing media file is presented. As occurred the TDS structure had the same logic.

0000	fa	55	55	55	03	03	02	00	cb	ca	fc	ff	00	00	00	00	.000
0010	43	1c	71	71	6d	bc	bl	9d	f4	10	43	7f	a7	07	a7	61	C.qqmCa
0020	1c	54	68	69	73	20	69	73	20	61	20	67	75	6e	20	6f	.This is a gun o
0030	75	74	67	6f	69	6e	67	20	70	68	6f	74	6f	00	00	00	utgoing photo
0040	d7	50	51	69	03	00	00	00	65	7a	19	fb	00	00	00	00	.PQiez
0050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0060	00	00	00	00	a7	07	a7	61	15	c4	b5	1c	02	00	00	00	a
0070	lb	b6	bf	77	01	73	00	00	cd	c6	7f	bc	00	00	00	80	w.s
0080	ff	ff	ff	ff	8d	cb	fc	ff	5a	00	00	00	32	00	00	00	Z2
0090	98	06	00	00	lb	b6	bf	77	01	79	00	00	54	55	55	55	w.yTUUU
00a0	04	00	00	00	fe	09	00	00	88	8f	38	51	37	91	3c	44	8Q7. <d< td=""></d<>
00b0	a7	39	51	a2	bc	49	13	c2	20	0a	9b	c5	7b	83	8c	b7	.9QI{
00c0	b5	86	29	6e	29	d0	08	9a	95	d5	29	45	55	6a	ab	ba)n))EUj
00d0	50	e8	3d	3d	7d	b6	ca	3b	cl	00	00	00	20	9c	d6	2e	P.==};
00e0	85	46	12	05	8a	0c	57	41	28	d4	la	99	63	95	fc	a7	.FWA(c
00f0	5e	ff	a2	8d	33	da	8c	ea	42	01	4c	b2	f0	00	00	00	^3B.L
0100	00	05	00	00	d0	02	00	00	8b	0f	02	00	00	00	00	00	
0110	15	c4	b5	1c	00	00	00	00	00	00	00	00	00	00	00	00	
0120	ee	7c	7c	6f	72	69	67	69	6e	61	6c	50	61	74	68	7c	. originalPath
0130	3d	7c	2f	73	74	6f	72	61	67	65	2f	65	6d	75	6c	61	= /storage/emula
0140	74	65	64	2f	30	2f	41	6e	64	72	6f	69	64	2f	64	61	ted/0/Android/da
0150	74	61	2f	6f	72	67	2e	74	65	6c	65	67	72	61	6d	2e	ta/org.telegram.
0160	6d	65	73	73	65	6e	67	65	72	2f	63	61	63	68	65	2f	messenger/cache/
0170	49													30	37	32	IMG 20211201 072
0180	36	34	34	5f	35	34	35	2e	6a	70	67	31	30	31	37	30	644 545.jpg10170
0190	30	37	5f	31	36	33	38	33	33	36	34	30	34	30	30	30	07 1638336404000
01a0	7c	7c	66	69	6e	61	6c	7c	3d	7c	31	7c	7c	67	72	6f	final = 1 gro
01b0	75	70	49	64	7c	3d	7c	30	7c	7c	2f	73	74	6f	72	61	upId = 0 /stora
01c0	67	65	2f	65	6d	75	6C	61	74	65	64	2f	30	2f	41	6e	ge/emulated/0/An
01d0	64	72	6f	69	64	2f	64	61	74	61	2f	6f	72	67	2e	74	droid/data/org.t
01e0														6e	67	65	elegram.messenge
01f0														34	38		r/cache/-2147483
0200	36	34	38	5f	2d	32	31	30	30	33	36	2e	6a	70	67	00	648 -210036.jpg.
0210																	

555555FA	Integer flagging this TDS as a secret message between two users
71711C43	The hex value of the user's id sending the message in the conversation (1903238211)
7F4310F4	The hex value of the user's id receiving the message in the conversation (2135101684)
61A707A7	The timestamp of the message (1638336423) in Unix Seconds
695150D7	End of the media message caption
The original path of the readable in ASCII charac	media file and the temporary path in the cache folder of the application are sters.

Table 14. Media file TDS (Telegram Data Structure) objects in Secret Chat thread.

Another interesting artifact was that in secret chats the user participant's "**uid**" in the conversation is hidden, unlike in the normal ones. The way to decode this number is by converting it to a hex value, and then converting the first eight (8) bytes back to decimal.

218	782	2138	980292	2 0			2	021-11-29 20	:09:58 B
19	783	2138	980292	2 0			2	021-11-29 20	:10:13 B
220	-210174	-6895359982712	127488	3 0			2	021-11-29 20	:23:09 Bi
21	-210175	-6895359982712 Reco		3 0 messages" tabl	le where the "uid	" of the cl	2 nat participant is encod	021-11-29 20 ed	:23:15 B
Γh	e value	e(dec) 6895	359982712	2127488 is co	onverted to (he	x) 5FB1	3D5F 0000 0000		
	nc_cha	ts" table.		1	Tab		√ 8 %	e locateo	in the
	able:	enc_chats	~	ອີ 🍾 💊 🏻 [
		uid	user	name		uid Filter	name Filter	status Filter	data Filter
	Filter		Filter	Filter	1	1903238211	zugni feyda;;;	1638046912	BLOB
	· · · · · · · · · · · · · · · · · · ·	-2056838969	1923643811	bad actor 2;;;	2	1923643811	bad actor 2;;;	-100	BLOB
	1	2030030909							
	1		2135101684	μιχάλης;;;	3	2135101684	μιχάλης;;;	-100	BLOB
	1	-1605451103		μιχάλης;;; edgar allan poe;	4		μιχάλης;;; edgar allan poe;;;		BLOB

Table 15. How to decode participant's "uid" from a Telegram's Secret Chat.

With the volume of objects being added, removed and modified, forensic tools are unable to decode a part of them. A simple example is that the latest versions, of two popular forensic tools, at the time of this Thesis, could decode several types of TDSs.

	☑ 🙁	Μιχάλης					zugni fe	iyda 🛞 1 5:18:48 πμ(UTC+0)	
		This is an inc		nedia messa 021 5:28:20 πj	The second se		1/ 12/202	Sources (1)	
		Sources (1)				zugni feyda		8	
		Μιχάλης				This is a secre		test message No.1 1 5:26:27 πμ(UTC+0)	
		This is a dog				_		Sources (1)	
						zu	gni feyda	8	
			photo/jpg	9 77954051495	52 121.ing	Th	is is a gu	n outgoing photo	
							A	y photo/y	
			1/12/	/2021 5:29:01 t	τμ(UTC+0)			У	
		Sources (2)					1/12/202	- 1 5:27:03 πμ(UTC+0)	
		Μιχάλης						Sources (1)	
		This is my do	Dg			zugni feyda		8	
			photo/jpc			This i a ma	ichine gu	n outgoing photo	
				- 77954051499	53_121.jpg		photo/j	y .	
			1/12/	/2021 5:29:15 т	ти(UTC+0)		У		
		Sources (2)					1/12/202	1 5-27:46 πμ(UTC+0)	
								Sources (1)	
UFE	ED PA vī	7.50 decode	d norm	nal chat n	nedia messages	UFED PA v7.50 did not	deco	de secret media mes	sages
					 Time stamp (UTC)[×] 	Text	File size	File time stamp (UTC)	Message
Z ☆ Z ☆	0	2135101684 2135101684	30694 30694		01/12/2021 05:29:15 nµ (UTC+0) 01/12/2021 05:29:01 nµ (UTC+0)	This is my dog This is a dog	25,0 KB 24,7 KB	01/12/2021 05:29:15 nµ (UTC+0) 01/12/2021 05:29:00 nµ (UTC+0)	Photo Photo
2 ☆	0	2135101684	30694		01/12/2021 05:28:20 nµ (UTC+0)			,	Text
						where the second second second			Text
	0	2135101684	30694	Μιχάλης	01/12/2021 05:26:06 nµ (UTC+0)	This is a test message. for media paths No.3			
	0	2135101684	30694	Μιχάλης	01/12/2021 05:25:41 nµ (UTC+0)	This is a crocodile	21,6 KB	01/12/2021 05:25:40 пµ (UTC+0)	Photo
	0	2135101684 2135101684	30694 30694	Μιχάλης Μιχάλης	01/12/2021 05:25:41 пµ (UTC+0) 01/12/2021 05:25:20 пµ (UTC+0)	This is a crocodile This is a test message for media paths. No.2			Text
	0	2135101684 2135101684 2135101684	30694 30694 30694	Μιχάλης Μιχάλης Μιχάλης	01/12/2021 05:25:41 nµ (UTC+0) 01/12/2021 05:25:20 nµ (UTC+0) 01/12/2021 05:24:55 nµ (UTC+0)	This is a crocodile This is a test message for media paths. No.2 This is a snake	21,6 KB 20,3 KB	01/12/2021 05:25:40 nµ (UTC+0) 01/12/2021 05:24:55 nµ (UTC+0)	Text Photo
	0	2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Μιχάλης	01/12/2021 05:25:41 πμ (UTC+0) 01/12/2021 05:25:20 πμ (UTC+0) 01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:33 πμ (UTC+0) 01/12/2021 05:24:06 πμ (UTC+0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths	20,3 KB	01/12/2021 05:24:55 пµ (UTC+0) 01/12/2021 05:24:05 пµ (UTC+0)	Text
	0 0 0	2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Μιχάλης	01/12/2021 05:25:41 πμ (UTC+0) 01/12/2021 05:25:20 πμ (UTC+0) 01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:33 πμ (UTC+0) 01/12/2021 05:24:06 πμ (UTC+0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf	20,3 KB	01/12/2021 05:24:55 пµ (UTC+0) 01/12/2021 05:24:05 пµ (UTC+0)	Text Photo Text
	0 0 0	2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694 Oxygei	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Μιχάλης Νιχάλης	01/12/2021 05:25:41 πμ (UTC+0) 01/12/2021 05:25:20 πμ (UTC+0) 01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:33 πμ (UTC+0) 01/12/2021 05:24:06 πμ (UTC+0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Ode normal chat media me	20,3 KB	01/12/2021 05:24:55 m² (UTC+0) 01/12/2021 05:24:05 m² (UTC+0) 25	Text Photo Text Photo
	0	2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694 Oxygei	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Μιχάλης Νιχάλης	01/12/2021 05:25:41 nµ (UTC+0) 01/12/2021 05:25:20 nµ (UTC+0) 01/12/2021 05:24:55 nµ (UTC+0) 01/12/2021 05:24:33 nµ (UTC+0) 01/12/2021 05:24:05 nµ (UTC+0) ics v14.1 did not deco	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Ode normal chat media me	20,3 KB	01/12/2021 05:24:55 m² (UTC+0) 01/12/2021 05:24:05 m² (UTC+0) 25	Text Photo Text Photo
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 4361899022663	30694 30694 30694 30694 Oxygei 811072 300 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Μιχάλης π Forens 69 Μιχάλης 69 Μιχάλης	01/12/2021 05:25:41 πμ (UTC+0) 01/12/2021 05:25:20 πμ (UTC+0) 01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:36 πμ (UTC+0) 01/12/2021 05:24:06 πμ (UTC+0) ics v14.1 did not deco	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media media This is incoming rpg -7 photo This is my dog	20,3 KB	01/12/2021 05:24:55 m² (UTC+0) 01/12/2021 05:24:05 m² (UTC+0) 25	Text Photo Text Photo
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 3 4361899022663 2135101684 2135101684	30694 30694 30694 30694 30694 811072 300 811072 300 300 300 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης π Forens	01/12/2021 05:25:41 пµ (UTC+0) 01/12/2021 05:25:20 пµ (UTC+0) 01/12/2021 05:24:55 пµ (UTC+0) 01/12/2021 05:24:33 пµ (UTC+0) 01/12/2021 05:24:06 пµ (UTC+0) ics v14.1 did not deco	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media media This is incoming rpg -7 photo This is my dog This is a dog	20,3 KB	01/12/2021 05:24:55 m² (UTC+0) 01/12/2021 05:24:05 m² (UTC+0) 25	Text Photo Text Photo
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694 811072 300 811072 300 300 300 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Μιχάλης π Forens 69 Μιχάλης 69 Μιχάλης	01/12/2021 05:25:41 πμ (UTC+0) 01/12/2021 05:25:20 πμ (UTC+0) 01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:36 πμ (UTC+0) 01/12/2021 05:24:06 πμ (UTC+0) ics v14.1 did not deco	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dede normal chat media mes This is incoming rpg -7 photo This is my dog This is a dog This is a incoming media message No.4	20,3 KB	01/12/2021 05:24:55 m² (UTC+0) 01/12/2021 05:24:05 m² (UTC+0) 25 -2147483	Text Photo Text Photo
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694 811072 300 811072 300 300 300 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης π Forens	01/12/2021 05:25:41 пµ (UTC+0) 01/12/2021 05:25:20 пµ (UTC+0) 01/12/2021 05:24:55 пµ (UTC+0) 01/12/2021 05:24:53 пµ (UTC+0) 01/12/2021 05:24:06 пµ (UTC+0) 01/12/2021 05:29:57 пµ (UTC+0) 01/12/2021 05:29:57 пµ (UTC+0) 01/12/2021 05:29:57 пµ (UTC+0) 01/12/2021 05:29:15 пµ (UTC+0) 01/12/2021 05:29:01 пµ (UTC+0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dede normal chat media mes This is incoming rpg -7 photo This is my dog This is a dog This is a incoming media message No.4	20,3 KB	01/12/2021 05:24:55 m² (UTC+0) 01/12/2021 05:24:05 m² (UTC+0) 25 -2147483	Text Photo Text Photo
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 Oxyger 811072 300 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης n Forens 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης	01/12/2021 05:25:41 пµ (UTC+0) 01/12/2021 05:25:20 пµ (UTC+0) 01/12/2021 05:24:55 пµ (UTC+0) 01/12/2021 05:24:53 пµ (UTC+0) 01/12/2021 05:24:06 пµ (UTC+0) 01/12/2021 05:29:57 пµ (UTC+0) 01/12/2021 05:29:57 пµ (UTC+0) 01/12/2021 05:29:57 пµ (UTC+0) 01/12/2021 05:29:15 пµ (UTC+0) 01/12/2021 05:29:01 пµ (UTC+0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media media This is incoming rpg -7 photo This is incoming rpg -7 photo This is a dog This is a dog This is a incoming media message No.4 This i a medhine gun outgoing photo	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text Photo 64821003
		2135101684 2135100555555555555555555555555555555555	30694 30694 30694 30694 Oxyger 811072 300 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης n Forens 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης	01/12/2021 05:25:41 πμ (UTC+0) 01/12/2021 05:25:20 πμ (UTC+0) 01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:33 πμ (UTC+0) ics v14.1 did not deco	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media media This is incoming rpg -7 photo This is incoming rpg -7 photo This is a dog This is a dog This is a incoming media message No.4 This i a medhine gun outgoing photo	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text Photo 648210035
		2135101684 2135100000000000000000000000000000000000	30694 30694 30694 30694 811072 300 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης n Forens 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης	01/12/2021 05:25:41 пµ (UTC +0) 01/12/2021 05:25:20 пµ (UTC +0) 01/12/2021 05:24:55 пµ (UTC +0) 01/12/2021 05:24:35 пµ (UTC +0) 01/12/2021 05:24:06 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:27:46 пµ (UTC +0) 01/12/2021 05:27:46 пµ (UTC +0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media media This is incoming rpg -7 photo This is incoming rpg -7 photo This is a dog This is a incoming media message No.4 This i a machine gun outgoing photo This is a gun outgoing photo	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text Photo 648210035
		2135101684 2135100000000000000000000000000000000000	30694 30694 30694 30694 30694 811072 300 811072 300 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης n Forens 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης 69 Μιχάλης	01/12/2021 05:25:41 пµ (UTC +0) 01/12/2021 05:25:20 пµ (UTC +0) 01/12/2021 05:25:50 пµ (UTC +0) 01/12/2021 05:24:35 пµ (UTC +0) 01/12/2021 05:24:36 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:15 пµ (UTC +0) 01/12/2021 05:29:16 пµ (UTC +0) 01/12/2021 05:27:46 пµ (UTC +0) 01/12/2021 05:27:46 пµ (UTC +0) 01/12/2021 05:27:03 пµ (UTC +0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media media This is incoming rpg -7 photo This is incoming rpg -7 photo This is a dog This is a dog This is a incoming media message No.4 This i a medhine gun outgoing photo	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text Photo 648210035
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης π Forens 69 Μιχάλης 69	01/12/2021 05:25:41 пµ (UTC +0) 01/12/2021 05:25:20 пµ (UTC +0) 01/12/2021 05:24:55 пµ (UTC +0) 01/12/2021 05:24:33 пµ (UTC +0) 01/12/2021 05:24:33 пµ (UTC +0) 01/12/2021 05:24:05 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:50 пµ (UTC +0) 01/12/2021 05:29:01 пµ (UTC +0) 01/12/2021 05:27:03 пµ (UTC +0) 01/12/2021 05:26:05 пµ (UTC +0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a snake This is a wolf Doce normal chat media media This is a wolf This is incoming rpg -7 photo This is a dog This is a incoming media message No.4 This is a machine gun outgoing photo This is a gun outgoing photo This is a secret media test message No.1 This is a test message. for media paths No.3 This is a crocodile	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text Photo 648210035
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 34361899022663 4361899022663 4361899022663 4361899022663 4361899022663 4361899022663 4361899022663 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 4361890265 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 43618905 4361895 4361895 4361895 4361895 4361895 4361895 4361895 4361895 4361895 4361895 4361895 4361895 4361895 436185 4361855 436185 4361855 436185555 43618555555555555555555555	30694 30694 30694 30694 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης Πολλης π 69 Μιχάλης	01/12/2021 05:25:41 nµ (UTC +0) 01/12/2021 05:25:20 nµ (UTC +0) 01/12/2021 05:24:55 nµ (UTC +0) 01/12/2021 05:24:53 nµ (UTC +0) 01/12/2021 05:24:06 nµ (UTC +0) ics v14.1 did not deco 01/12/2021 05:29:57 nµ (UTC +0) 01/12/2021 05:29:57 nµ (UTC +0) 01/12/2021 05:29:57 nµ (UTC +0) 01/12/2021 05:29:57 nµ (UTC +0) 01/12/2021 05:29:01 nµ (UTC +0) 01/12/2021 05:27:46 nµ (UTC +0) 01/12/2021 05:27:46 nµ (UTC +0) 01/12/2021 05:27:03 nµ (UTC +0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a snake This is a wolf Dode normal chat media message This is a wolf This is incoming rpg -7 photo This is a dog This is a local message No.4 This is a gun outgoing media message No.4 This is a gun outgoing photo This is a secret media test message No.1 This is a test message.for media paths No.3 This is a test message.for media paths. No.3 This is a test message for media paths. No.3	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text Photo 648210039
		2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684 2135101684	30694 30694 30694 30694 30694 811072 300 811072 300	Μιχάλης Μιχάλης Μιχάλης Μιχάλης π Forens 69 Μιχάλης 69	01/12/2021 05:25:41 пµ (UTC +0) 01/12/2021 05:25:20 пµ (UTC +0) 01/12/2021 05:24:55 пµ (UTC +0) 01/12/2021 05:24:33 пµ (UTC +0) 01/12/2021 05:24:33 пµ (UTC +0) 01/12/2021 05:24:05 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:57 пµ (UTC +0) 01/12/2021 05:29:50 пµ (UTC +0) 01/12/2021 05:29:01 пµ (UTC +0) 01/12/2021 05:27:03 пµ (UTC +0) 01/12/2021 05:26:05 пµ (UTC +0)	This is a crocodile This is a test message for media paths. No.2 This is a snake This is a message for media paths This is a wolf Dode normal chat media message This is a wolf This is incoming rpg -7 photo This is an incoming media message No.4 This is a machine gun outgoing photo This is a gun outgoing photo This is a secret media test message No.1 This is a test message for media paths No.3 This is a crocodile This is a snake	20,3 KB	01/12/2021 05:24:55 πμ (UTC+0) 01/12/2021 05:24:05 πμ (UTC+0) 25 -2147483 -2147483	Text Photo Text

 Table 16. Comparison of how UFED PA (above) and Oxygen Forensics (below) were able to decode different kind of TDS (Telegram Data Structure)

6.1.4. Volatility of the "cache4.db"

The experiments confirmed that both the "cache4" database and its corresponding WAL file were not susceptible to certain device status changes, which did not seem to alter the data inside the "messages" table. Those actions included:

- ✓ Leaving the device powered on and with network.
- ✓ After the finishing boot sequence of the device and the application auto starts when network was available.
- ✓ At opening of the app after closing it.

Under the following circumstances the "cache4" database was not modified:

- ✓ Leaving the device powered on in airplane mode.
- ✓ Rebooting/turning off the device while it was in airplane mode.
- ✓ At closing the app and removing it from the task manager of the android device.
- Removing the SIM card (which was the only thing associated with account), while the device is live and after it was deactivated and then powered on without the SIM.
- ✓ Removing notifications of incoming messages.

The database's "messages" table was modified when:

- a message was created. It does not matter if it was successfully sent or not.
- ! a message was received.
- **!** a message was opened. This was regardless of whether it was an outgoing message which was read by the receiver or an incoming which was read by the user.
- ! a message or a thread was deleted.

6.1.5. Retrieving the messages in question and the timebomb feature

Telegram implements a timebomb/autodelete feature but only in the secret chat section (the E2E encrypted section). The autodelete feature could vary, from one (1) second to a few days. Nevertheless, this feature did not affect the ability to recover this kind of messages, since it did not have any other additional properties except the data in column "TTL", which represents the timer until deletion, in seconds. Every time a message's state changed the whole record was moved inside the WAL file creating a new separate record.

			대마네 [- 마비 [- 마비] ,	CASE messages.SEND_S WHEN '1' THEN '1	<pre>sage ID', ', date, 'unixepoch', 'localtime') AS STATE he message has not reached Server' ge was delivered to the Server' al Status', Incoming' ection', state Non Received' Keceived'</pre>	'Date',				
			נייי בישע בישע בייי	messages.MID AS 'Mee users.data AS 'Info' datetime (messages.c CASE messages.SEND', ELSE 'IThe messag END AS 'Message Lord CASE messages.out WHEN '0' THEN '1 ELSE 'Outgoing' RD AS 'Message Dire CASE messages.read, WHEN '2' THEN '1 WHEN '2' THEN '1	<pre>', 'unixepoch', 'localtime') AS STATE The message has not reached Server' ye was delivered to the Server' al Status', Incoming' ection', state Non Received' Received'</pre>	'Date',				
			נישיע באיע באיע איי	users.data AS 'Info' datetime (messages.CAS messages.CSUD) WHEN '1' THEN '1 ELSE 'The message END AS 'Message Locd CASE messages.out WHEN '0' THEN '1 ELSE 'Outgoing' END AS 'Message Dire CASE messages.read, WHEN '2' THEN '1 WHEN '3' THEN '1	<pre>', 'unixepoch', 'localtime') AS STATE The message has not reached Server' ye was delivered to the Server' al Status', Incoming' ection', state Non Received' Received'</pre>	'Date',				
			נישיים בישעו	datatime (messages.c CASE messages.sEND_2 WHEN '1' THEN '1 ELSE 'The message END AS 'Hessage Lord CASE messages.out WHEN '0' THEN '1' WHEN '2' THEN '1' WHEN '2' THEN '1'	<pre>date, 'unixepoch', 'localtime') AS STATE The message has not reached Server' ge was delivered to the Server' al Status', Incoming' ection', state Non Received' Keccived'</pre>	'Date',				
			נישים ששים שישים	CASE messages.SEND_ WHEN '1' THEN '1 ELSE 'The message END AS 'Message Locc CASE messages.out WHEN '0' THEN '1 ELSE 'Outgoing' END AS 'Message Dire CASE messages.read_ WHEN '3' THEN '1 WHEN '3' THEN '1	STATE The message has not reached Server' per was delivered to the Server' al Status', Incoming' ection', state Non Received' Neceived'	'Date',				
			יה-היי ה-היי	WHEN '1' THEN '1 ELSE 'The message EDA S' Message Loca CASE messages.out WHEN '0' THEN '1 ELSE 'Outgoing' END AS 'Message Dire CASE messages.read_ WHEN '3' THEN '1 WHEN '3' THEN '1	The message has not reached Server' ye was delivered to the Server' al Status', Incoming' ection', state Non Received' Received'					
			החרי הרחי	ELSE 'The messag END AS 'Message Loca CASE messages.out WHEN '0' THEN'' ELSE 'Outgoing' END AS 'Message Dire CASE messages.read_e WHEN '3' THEN 'E WHEN '3' THEN 'E	ge was delivered to the Server' al Status', Incoming' ection', state Non Received' Received'					
			החרי הרחי	END AS 'Message Loc CASE messages.out WHEN '0' THEN '1 ELSE 'Outgoing' END AS 'Message Dire CASE messages.read_s WHEN '2' THEN '1 WHEN '3' THEN '1	l Status', Incoming' ection', state Non Received' Received'					
			החרי הרחי	CASE messages.out WHEN '0' THEN 'J ELSE 'Outgoing' END AS 'Message Dire CASE messages.read WHEN '2' THEN 'J WHEN '3' THEN 'S	Incoming' action', state Non Received' Keceived'					
				WHEN '0' THEN 'D ELSE 'Outgoing' END AS 'Message Dire CASE messages.read_ WHEN '2' THEN 'N WHEN '3' THEN 'F	ection', state Non Received' Received'					
			Į Į	ELSE 'Outgoing' END AS 'Message Dire CASE messages.read_s WHEN '2' THEN '1 WHEN '3' THEN 'F	ection', state Non Received' Received'					
			Į Į	CASE messages.read_s WHEN '2' THEN 'N WHEN '3' THEN 'F	state Non Received' Received'					
			Į Į	WHEN '2' THEN 'N WHEN '3' THEN 'F	Non Received' Received'					
				WHEN '3' THEN 'F	Received'					
				END AS 'Message stat	tus',					
				messages.data AS 'Bo						
				messages.TTL AS 'Tim messages.sfSource AS						
				FROM messages	5 JOULCE					
					users.uid = messages.uid					
				Quer	y which was executed					
Name	UID	Message ID	Info	Date	Message Local Status	Message Direction	Message status	Body	Timer	Source
ILL	-7424347666923913216	-210225	NULL	2021-11-29 22:44:05	The message has not reached Server	Outgoing	Non Received	BLOB	30	WAL-96-commit
ILL	-7424347666923913216	-210225	NULL	2021-11-29 22:44:06	The message was delivered to the Server	Outgoing	Non Received	BLOB	30	WAL-111-comm
ILL	-7424347666923913216	-210225	NULL	2021-11-29 22:44:06	The message was delivered to the Server	Outgoing	Received	BLOB	30	WAL-115-comm
ILL		-7424347666923913216 -7424347666923913216 -7424347666923913216	-7424347666923913216 -210225 -7424347666923913216 -210225 -7424347666923913216 -210225	Name UD Message ID Info -7424347666923913216 -210225 NULL -7424347666923913216 -210225 NULL -7424347666923913216 -210225 NULL	UD Message ID Info Date -7424347666923913216 -210225 NULL 2021-11-29 22:44:05 -7424347666923913216 -210225 NULL 2021-11-29 22:44:05 -7424347666923913216 -210225 NULL 2021-11-29 22:44:05 -7424347666923913216 -210225 NULL 2021-11-29 22:44:05	-7424347666923913216 -210225 NULL 2021-11-29 22:44:05 The message has not reached Server -7424347666923913216 -210225 NULL 2021-11-29 22:44:06 The message was delivered to the Server -7424347666923913216 -210225 NULL 2021-11-29 22:44:06 The message was delivered to the Server -7424347666923913216 -210225 NULL 2021-11-29 22:44:06 The message was delivered to the Server	ORDER EY messages.date ASC Name UID Message ID Info Date Message Local Status Message Direction -7424347666923913216 -210225 V/UL 2021-11-29 22:44:05 The message has not reached Server Outgoing -7424347666923913216 -210225 V/UL 2021-11-29 22:44:05 The message was delivered to the Server Outgoing	ODDER BY messages.date ASC Name UID Message ID Info Date Message Local Status Message Direction Message status -742437666923913216 -210225 W/L 2021-11-29 22:44:05 The message has not reached Server Outgoing Non Received -742437666923913216 -210225 W/L 2021-11-29 22:44:05 The message was delivered to the Server Outgoing Non Received	Name UID Message ID Info Date Message Local Status Message Direction Message status Body -7424347666923913216 -210225 NUL 202111-29 22:44:05 The message has not reached Server Outgoing Message status Body -7424347666923913216 -210225 NUL 202111-29 22:44:05 The message was delivered to the Server Outgoing Non Received Body -7424347666923913216 -210225 NUL 202111-29 22:44:05 The message was delivered to the Server Outgoing Non Received Bid -7424347666923913216 -210225 NUL 202111-29 22:44:05 The message was delivered to the Server Outgoing Received Bid -7424347666923913216 -210225 NUL 202111-29 22:44:05 The message was delivered to the Server Outgoing Received Bid -7424347666923913216 -210225 NUL 202111-29 22:44:05 The message was delivered to the Server Outgoing Received Bid	ORDER BY messages.tate JSC DRDER BY messages.tate JSC Name UID Message ID Info Date Message Local Status Message Direction Message status Body Timer -7424347666923913261 -210225 VVV 2021-11-29 22:44:05 The message has not reached Server Outgoing Message status Body Timer -7424347666923913261 -210225 VVV 2021-11-29 22:44:05 The message was delivered to the Server Outgoing Non Received 8/00 3/0 -7424347666923913261 -210225 VVVV 2021-11-29 22:44:05 The message was delivered to the Server Outgoing Received 8/00 3/0 -7424347666923913261 -210225 VVVV 2021-11-29 22:44:05 The message was delivered to the Server Outgoing Received 8/00 3/0 -7424347666923913261 -210225 VVVV 2021-11-29 22:44:05 The message was delivered to the Server Outgoing Received 8/00 3/0 -7424347666923913261 -210225 VVVV 2021-11-29 22:44:05 The message was delivered to the Server Outgoing Received 8/00

Table 17. Records of a Secret Chat in "cache4" database's WAL file.

Telegram did not have the full "auto_vacuum" PRAGMA statement enabled and thus it creates freelist pages. As mentioned in paragraph §1.3.1.4 the existence of freelist pages enables the possibility of retrieving deleted messages. However, even if it had not the "secure_delete" PRAGMA statement enabled it had implemented some kind of failsafe for wiping the deleted records. This was tested multiple times in both devices by deleting hundreds of records, creating freelist pages, and attempting to recover them. In every case, immediately after the deletion, most of the messages were recoverable, but after committing some new ones (incoming or outgoing) all the messages residing in the main database were erased (**Table 17**). This was verified by searching specific messages as a string, in the database and WAL file, with a hex editor.

Name	Value
	-
application_id	0
auto_vacuum	none
cache_size	-2000
collation_list	[BINARY], [NOCASE], [RTRIM]
encoding	UTF-8
foreign_keys	on
freelist_count	0
journal_mode	wal
journal_size_limit	-1
max_page_count	1073741823
mmap_size	0
page_count	334
page_size	4096
schema_version	94
user_version	80
writable_schema	off

2 3	count (message	urce AS 'Source', s.sfSource)	application_id	0
4	FROM messages		auto_vacuum	none
5 6	GROUP by mess	ages.sfSource t(messages.sfSource) DESC	cache_size	-2000
_			collation_list	[BINARY], [NOCASE], [RTRIM]
	Source	count(messages.sfSource)	encoding	UTF-8
1	DB	198	foreign_keys	on
2	WAL-338-commit	29	freelist_count	22
3	WAL-218-commit	29	journal_mode	wal
4	WAL-422-commit	28	journal_size_limit	-1
5	WAL-139	11	max_page_count	1073741823
6	WAL-449	10	mmap_size	0
			page_count	383
7	WAL-153	7	page_size	4096
8	WAL-1002-commit	7	schema_version	94
9	WAL-986	1	user_version	80
10	WAL-968	1	writable_schema	off
	file be	d from the database and its fore deletion	State of the database (1.568 kb in size) before dele records
1 2	SELECT messages.sfS	ource AS 'Source',	Name	Value
3	count (message		application_id	0
4	FROM message:		auto_vacuum	none
5	_	<pre>sages.sfSource nt(messages.sfSource) DESC</pre>	cache_size	-2000
_	CIUSIK DI COU	To (mosougos, sisource) Disc	collation_list	[BINARY], [NOCASE], [RTRIM]
	Source	count(messages.sfSource)	encoding	UTF-8
1	DB	225	foreign_keys	on
2	WAL-170-commit	12	freelist_count	39
3	WAL-1002-commi	t 7	journal_mode	wal
4	WAL-416-commit	2	journal_size_limit	-1
		2	max_page_count	1073741823
5			mmap_size	0
6		2	page_count	383
7	WAL-392-commit	2	page_size	4096
8	WAL-386-commit	2	schema_version	94
9	WAL-380-commit	2	user_version	80
1	0 WAL-374-commit	2	writable_schema	off
1 2 3 4 5	file a		Name application_id auto_vacuum	(1.568 kb in size) after dele records Value 0 none
5	ORDER BY count	(messages.sfSource) DESC	cache_size	-2000
_	Source	count(messages.sfSource)	collation_list	[BINARY], [NOCASE], [RTRIM
1	WAL-224-commit	29	encoding	UTF-8
_			foreign_keys	on
2	WAL-170-commit	29	freelist_count	39
3	WAL-344-commit	28	journal_mode	wal
4	WAL-1002-commit	23	journal_size_limit	-1
	WAL-23	20	max_page_count	1073741823
5	WAL-486	15	mmap_size	0
5 6	WAL-61	11	page_count	383
			page_size	4096
			schema_version	94
	WAL-698	5		
		5	user_version	80
6 7 8 9	WAL-698		user_version writable_schema	80 off

Table 18. The state of "cache4.db" and its WAL file in different conditions.

There was a difference in the gravity of the acts. An outgoing message occupied more records and space in the WAL file than reading an incoming message. The element of luck is also involved. A crucial factor is the timing of the in-question messages deletion. Whether it occurs right after a WAL commit to the database, or before it. This is a parameter purely random and cannot be controlled in realistic situations. In **Table 18** is presented a list of the different experiments which were executed (*Test No*), the prior condition before deleting the in-question messages (*Parameters prior to deletion*), the different actions in which they were still recoverable and are green colored (*Incoming/Outgoing/Opened/Deleted Messages*) and the action which finally led to the removal of the chat thread from the main database or the WAL file and is red coloured (*Parameters prior to deletion*).

Test No	Parameters prior to deletion		Incoming Messages	Outgoing Messages	Messages that were opened	Messages that were deleted	Action that removed the messages in-question
1	None	9	15	5	15	None	10 outgoing messages
2	All messages of the database were deleted	T THREAD	50	None	20	None	the rest 30 messages were opened
3	60 messages included on the thread alongside with in-question ones were deleted	UESTION CHAT	50	None	None	80 (From other chat thread)	50 messages were opened
4	None	IN-QUE	20		60	None	10 incoming messages
5	WAL commit was forced ⁷	OF THE	20	15	None	None	40 messages were opened
6	All messages of the database were deleted, and WAL commit was forced	DELETION	70	None	None	None	70 read
7	None		50	None	None	None	70 read

Table 19. Progress of experiments' results on "cache4.db" and its WAL file.

6.1.6. Wiped Databases

In Telegram there were two occasions where no records of the database were recoverable. i. When the user logged out of the device, triggering a complete database's wipe and ii. By clearing the local database from the settings menu. In any case the difference between the timestamp of "cache4.db" and the Android' OS usage history application, might be a strong indicator that one of the above occasions have occurred.

6.2. Telegram's Media Files Attachments

The media transmission procedure was the same as it was explained in paragraph §4.2. Telegram administrated the shared media files, in a different way based on the type of the chat thread (normal/secret).

⁷ The database was monitored in real time by exchanging single messages, until a WAL commit occurred.

6.2.1. Normal chat media transmission

In a normal chat thread, when the in-app camera feature was used, the media files were saved to the following paths:

- media\0\Pictures\Telegram. The files stored by their captured file name (e.g., «IMG_20211125_102653_487.jpg») and
- media\0\Telegram\Telegram Images. The files stored by their, by the transmitted, via the Telegram server, name (e.g., «-5834547226205076084_121.jpg»).

After a media file deletion some artifacts which have remained on the android operating system were utilized to retrieve its information. To be more accurate this was done by comparing those artifacts with the records located in "cache4" database. Those artifacts' paths were analyzed in previous corresponding paragraphs (§4.2 and §5.2), for each device. Samsung device's path was «data\com.sec.android.gallery3d\cache», and the LG device's paths were: i. «media\0\DCIM\.thumbnails» and ii. «media\0\Android\data\com.android.gallery3d\cache\ imgcache.0». An example of the latter is presented in Table 19.

MID			UID			RE	AD_	STAT	E	SEND	_st/	ATE			DA	TE 🕫		DATA	OUT
Filter	Filte	r				Filt	er		F	Filter			Filte	r				Filter	Filter
49			19	2364	3811				2			0			2021	-11-2	5 10:27:00	BLOB	1
								Re	cord	in th	e "ca	che4.	db"						
	_																		
0000	d2	83	e3	bc	02	03	02	00	97	cb	fc	ff	6d	bc	bl	9d			m
0010	43	lc	71	71	6d	bc	bl	9d	a3	79	a 8	72	d4	48	9f	61	C.qqm	y.1	.H.a
0020	00	00	00	00	d7	50	51	69	03	00	00	00	65	7a	19	fb	PC	Qi	ez
0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00			
0040	00	00	00	00	00	00	00	00	d4	48	9f	61	15	c4	b5	lc		Н.а	i
0050	02	00	00	00	1b	b6	bf	77	01	73	00	00	cd	c6	7f	bc		w.s	
0060	00	00	00	80	ff	ff	ff	ff	b0	cb	fc	ff	5a	00	00	00			z
0070	32	00	00	00	04	07	00	00	1b	b6	bf	77	01	79	00	00	2		/.y
0080	cd	c6	7f	bc	00	00	00	80	ff		ff	ff	af	cb	fc	ff	•••••		
0090	00	05	00	00	d0	02	00	00	a4	8e	02	00	00	00	00	00	•••••		
00a0	00	00	00	00	00	00	00	00	d6	7c	7c	6f	72	69	67	69			-
00b0	6e	61	6c	50	61	74	68	7c	3d	7c	2f	73	74	6f	72	61	nalPath		
00c0	67	65	2f	65	6d	75	6C	61	74	65	64	2f	30	2f	50	69	ge/emul		
0000	63	74	75	72	65	73	2f	54	65	6C	65	67	72	61		2f	ctures/		
00e0	49	4d	47	5f	32	30	32	31	31	31	32	35	5f	31	30	32	IMG 202		
0010	36	35	33	5f		38	37	2e	6a		67	31	31	30	39	36	653 487		
0100	38	35	5f	31	36	33	37	38	32		38	31	33	30	30	30	85 1637		
0110	7c	7c	66	69		61	6C	7c	3d		31	7c	7c	67	72	6f	final		
0120	75	70	49	64	7c	3d	7c	30	7c		2f	73	74	6f	72	61	upId =		
0130	67	65	2f	65	6d	75	6c	61	74	65	64	2f	30	2f	41	6e	ge/emul		
0140	64	72	6f	69	64	2f	64	61	74	61	2f	6f	72	67	2e	74	droid/d		-
0150	65	6C	65	67	72	61	6d			65	73	73	65	6e	67	65	elegram		_
0160	72	2f	63	61		68				32	31	34	37	34		33	r/cache		
0170	36	34	38	5f	2d	32	31	30	30	30	31	2e	6a	70	67	00	64821	10001.	jbd.
0180									•		•								
									Con	tents	of the	e TDS	5						

1. ar	normal med	ia message is sent					
2. by	the user 1	03238211					
3. to	the user 19	23643811					
4. at	163782882	0					
5. wh	oro modio	ilo'a nome in IMC 20211125102652487					
.	lere media	ile's name is IMG 20211125102653487					
итс		OriginalFilePath	Extra	InternalPath	Thumbnail	^	
UTC			Extra Filter	InternalPath Filter	Thumbnail Filter	^	
UTC	LocalTime +1 Filter	OriginalFilePath			Filter	^	and the second
UTC Filter 2021-11-25 08:2	LocalTime ▲1 Filter 26:53 2021-11-25 10:26	OriginalFilePath Filter	Filter	Filter	Filter 5 BLOB	^	
UTC Filter 2021-11-25 08:2 2021-11-19 10:1	LocalTime *1 Filter 26:53 2021-11-25 10:26 14:15 2021-11-19 12:14	OriginalFilePath Filter i3 /storage/emulated/0/Pictures/Telegram/IMG_20211125_102653_487.jpg	Filter	Filter /local/image/item/4405	Filter 5 BLOB 5 BLOB	^	
UTC Filter 2021-11-25 08:2 2021-11-19 10:1 2021-11-19 10:1	LocalTime *1 Filter 26:53 2021-11-25 10:26 14:15 2021-11-19 12:14 14:15 2021-11-19 12:14	OriginalFilePath Filter 3 /storage/emulated/0/Pictures/Telegram/IMG_20211125_102653_487.jpg 5 /storage/emulated/0/WhatsApp/Media/WhatsApp Images/IMG-20211119-WA0001.jpeg	Filter NULL NULL	Filter /local/image/item/4405 /local/image/item/2255	Filter 5 BLOB 5 BLOB 5 BLOB 5 BLOB	^	
UTC Filter 2021-11-25 08:2 2021-11-19 10:1 2021-11-19 10:1 2021-11-19 10:1	LocalTime +1 Filter 26:53 2021-11-25 10:26 14:15 2021-11-19 12:14 14:15 2021-11-19 12:14 11:52 2021-11-19 12:11	OriginalFilePath Filter 3 /storage/emulated/0/Pictures/Telegram/IMG_20211125_102653_487.jpg 5 /storage/emulated/0/WhatsApp/Media/WhatsApp Images/IMG-20211119-WA0001.jpeg 5 /storage/emulated/0/WhatsApp/Media/WhatsApp Images/IMG-20211119-WA0001.jpeg	Filter NULL NULL NULL	Filter /local/image/item/4405 /local/image/item/2255 /local/image/item/2255	Filter BLOB BLOB BLOB BLOB	~	
UTC Filter 2021-11-25 08:2 2021-11-19 10:1 2021-11-19 10:1 2021-11-19 10:1 2021-11-19 10:1	LocalTime +1 Filter 26:53 2021-11-25 10:26 14:15 2021-11-19 12:14 14:15 2021-11-19 12:11 11:52 2021-11-19 12:11	OriginalFilePath Filter 3 /storage/emulated/0/Pictures/Telegram/IMG_20211125_102653_487.jpg 5 /storage/emulated/0/WhatsApp/Media/WhatsApp Images/IMG-20211119-WA0001.jpeg 5 /storage/emulated/0/DCIM/Camera/20211119_121152.jpg	Filter NULL NULL NULL NULL	Filter /local/image/item/4405 /local/image/item/2255 /local/image/item/2255 /local/image/item/2254	Filter 5 BLOB 5 BLOB 6 BLOB 8 BLOB	^	

Table 20. Correlation of a sent image file's record in "cache4.db" with "imgcache.sqlite" file (tool used: DBBrowser for SQLite, Examined Device: LG G6).

6.2.2. Secret chat media transmission

In a secret chat when the media were captured with in-app camera feature never resided outside the cache folder of the application. After deleting the secret chat messages, it was impossible to retrieve/determine the content of the files (attachments) which were sent. During the experiments, it was observed that the corresponding thumbnail files had always a minus one in its file name values than the ones in the database. For instance, the thumbnail of the transmitted media named "-2147483648_-21002.jpg", but in the database record its name was "-2147483648_-21003.jpg". This was a recurring pattern, but it was not decoded.

1611	-210029	-373324493013123072	3	0	2021-11-25 10:28:00	BLOB	1	0	

0000	fa	55	55	55	03	03	02	00	93	cb	fc	ff	00	00	00	00	.000
0010	43	lc	71	71	6d	\mathbf{bc}	bl	9d	a3	79	a 8	72	10	49	9f	61	C.qqmy.r.I.a
0020	00	00	00	00	d 7	50	51	69	03	00	00	00	65	7a	19	fb	PQiez
0030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0040	00	00	00	00	00	00	00	00	0f	49	9f	61	15	$\mathbf{c4}$	b5	1c	I.a
0050	02	00	00	00	1b	b6	bf	77	01	73	00	00	\mathbf{cd}	c 6	7f	bc	w.s
0060	00	00	00	80	ff	ff	ff	ff	ae	$^{\rm cb}$	fc	ff	5a	00	00	00	Z
0070	32	00	00	00	e3	05	00	00	1b	b6	bf	77	01	79	00	00	2w.y
0080	54	55	55	55	04	00	00	00	f6	0a	00	00	4 f	61	£9	50	TUUUOa.P
0090	57	7b	£0	4a	5c	f6	38	3d	31	35	de	6d	20	35	62	15	W{.J\.8=15.m 5b.
00a0	£4	75	1c	72	ee	9e	6e	f6	75	9b	\mathbf{bc}	42	64	6a	$^{\rm cb}$	bb	.u.rn.uBdj
00b0	d2	3a	58	dl	0e	83	19	b2	83	8e	04	2e	ba	00	00	00	.:X
00c0	20	a7	\mathbf{cd}	f6	20	46	b5	\mathbf{bf}	2a	2e	95	4e	30	00	ef	\mathbf{fd}	F*NO
00d0	12	a2	e7	15	14	47	39	26	27	31	d7	1e	60	с6	68	7f	G9&'l`.h.
00e0	49	00	00	00	00	05	00	00	d0	02	00	00	3c	b9	01	00	I<
00f0	00	00	00	00	15	c4	b5	1c	00	00	00	00	00	00	00	00	•••••
0100	00	00	00	00	ed	7c	7c	6f	72	69	67	69	6e	61	6C	50	originalP
0110	61	74	68	7c	3d	7c	2f	73	74	6f	72	61	67	65	2f	65	ath = /storage/e
0120	6d	75	6C	61	74	65	64	2f	30	2f	41	6e	64	72	6f	69	mulated/0/Androi
0130	64	2f	64	61	74	61	2f	6f	72	67	2e	74	65	6c	65	67	d/data/org.teleg
0140	72	61	6d	2e	6d	65	73	73	65	6e	67	65	72	2f	63	61	ram.messenger/ca
0150	63	68	65	2f	49	4d	47	5f	32	30	32	31	31	31	32	35	che/IMG 20211125
0160	5f	31	30	32	37	34	39	5f	37	31	37	2e	6a	70	67	39	102749 717.jpg9
0170	34	32	39	35	37	5f	31	36	33	37	38	32	38	38	36	39	42957 1637828869
0180	30	30	30	7c	7c	66	69	6e	61	6c	7c	3d	7c	31	7c	7c	000 final = 1
0190	67	72	6f	75	70	49	64	7c	3d	7c	30	7c	7c	2f	73	74	groupId = 0 /st
01a0	6f	72	61	67	65	2£	65	6d	75	6c	61	74	65	64		30	orage/emulated/0
01b0	2f		6e	64	72	6f	69	64	2f	64	61	74	61	2f		72	/Android/data/or
01c0	67	2e	74	65	6C	65	67	72	61	6d	2e		65	73	73	65	g.telegram.messe
01d0	6e	67	65	72	2f	63	61	63	68	65	2f		32	31	34	37	nger/cache/-2147
	34	38	33	36	34	38	5f	2d	32	31	30	30	30	33	2e	6a	483648210003.j
01f0	70	67	00	00													pg
<u>iki</u> kik.a i=-iii org.t	aver.line.an Indroid (0)	droid (0)	46)	1 1 58	147483648	21847711_1 210004.jp 65501686_1 210002.jp	789548450 9 257274199 9									existing existing existing	xisting, 1st cluster n temp 0 B 2021/11/25d10:30:03 g, tagged, already v jpg 80.1 KB 2021/11/25d10:29:49 g, tagged jpg 1.7 KB 2021/11/25d10:29:49 g, tagged, already v jpg 110 KB 2021/11/25d10:27:59 g, tagged, already v jpg 115 KB 2021/11/25d10:27:59 g, tagged, already v jpg 1.5 KB [2021/11/25d10:27:59 g, tagged, already v jpg 1.5 KB [2021/11/25d10:27:59
	cache (acae i acae files (0)	50) che (13)				× -21	4748364	82100 165501	04.jpg 686_125	9548450 7274199	.jpg	prev. ex prev. ex	isting, 1 isting, 1	st clust st clust	er not a er not a	vailable, . vailable, . vailable, . vailable, .	jpg 0 B 2021/11/25d10-20-49 jpg 0 E 2021/11/25d10:27:59

Table 21. Detection of a sent image file in "cache4.db" from a Secret chat thread.

One last interesting fact was that the secret media transmission was the only one that was not recorded in the "graph.db" located in the path «/data/com.samsung.mlp/databases» of the Samsung device. On the contrary the media sent as normal chat message could be retrieved from the "graph" database (**Table 21**).

col2	col3						col4						col5	col6			ol7 🗚	
ilter	Filter	Filter					0011								Filter			Filter
VULL	1	/storag	e/emu	lated/0/	Pictures/	Telegra	r IMG_2	2021112	5_10285	4_845.j	pg		65537	NULL	2021-11-2	5 08:29	:26.884	2021-11-25 08:28:54.45
VULL	1	/storag	e/emu	lated/0/.	Android/	media/o	om.what	tsapp/W	hatsApp/	/Media/\	WhatsAp	o	65537	NULL	2021-11-1	8 21:11	:48.835	2021-11-18 21:11:06.54
VULL	1	/storag	e/emu	lated/0/	DCIM/Ca	mera/20	211118	_230817	.jpg				65537	NULL	2021-11-1	8 21:08	:58.14	2021-11-18 21:08:17.50
							" /	/ledia	Attrib	ute"	table	of Sa	msu	ng "g	raph.c	lb"		
1256	5	43			19	0323				2			0					2021-11-25 08:28:56 BLOB
000	00	d2	83	e3	bc	02	03	02	00	94	cb	fc	ff	6d	l bc	bl	9d	m
001	LO	a3	79	a 8	72	6d	bc	bl	9d	43	1c	71	71	48	49	9f	61	.y.rmC.qqHI.a
002	20	00	00	00	00	d7	50	51	69	03	00	00	00	65	7a	19	fb	PQiez
003	30	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
004	ŧo	00	00	00	00	00	00	00	00	48	49	9f	61	15	c4	b5	1c	HI.a
005	50	02	00	00	00	1b	b6	bf	77	01	73	00	00	cd	L C6	7f	bc	w.s
006	50	00	00	00	80	ff	ff	ff	ff	af	cb	fc	ff	5a	00	00	00	z
007	70	32	00	00	00	27	08	00	00	1b	b6	bf	77	01	. 79	00	00	2'w.y
008	30	cd	c6	7f	bc	00	00	00	80	ff	ff	ff	ff	ae	cb	fc	ff	
009	90	00	05	00	00	d0	02	00	00	e9	d5	02	00	00	00	00	00	
00a	10	00	00	00	00	00	00	00	00	d6	7c	7c	6f	72	69	67	69	origi
001	00	6e	61	6c	50	61	74	68	7c	3d	7c	2f	73	74	6f	72	61	nalPath = /stora
000	:0	67	65	2f	65	6d	75	6c	61	74	65	64	2f	30	2f	50	69	ge/emulated/0/Pi
000	10	63	74	75	72	65	73	2f	54	65	6c	65	67	72	61	6d	2f	ctures/Telegram/
006	20	49	4d	47	5f	32	30	32	31	31	31	32	35	5f	31	30	32	IMG 20211125 102
001	ΞO	38	35	34	5f	38	34	35	2e	6a	70	67	31	38	37	33	32	854 845.jpg18732
010	00	36	35	5f	31	36	33	37	38	32	38	39	33	34	30	30	30	65 1637828934000
011	LO	7c	7c	66	69	6e	61	6c	7c	3d	7c	31	7c	7c	67	72	6f	<pre> final = 1 gro</pre>
012	20	75	70	49	64	7c	3d	7c	30	7c		2f	73	74	6f	72	61	upId = 0 /stora
013			65	2f	65	6d	75	6c	61	74	65	64	2f	30		41	6e	ge/emulated/0/An
014	10	64	72	6f	69	64	2f	64	61	74	61	2f	6f	72	67	2e	74	droid/data/org.t
015		65	6c	65	67	72	61	6d	2e	6d	65	73	73	65		67	65	elegram.messenge
016	-	_	2f	63	61	63	68	65	2f	2d	32	31	34	37		38	33	r/cache/-2147483
017			34		5f	2d	32	31	30	30	30	32	2e	6a		67	00	648 -210002.jpg.
018	-															-		
							Norm	al Ch	ats's	Reco	ord an	d its	TDS i	from	"cach	e4.db	"	1

Table 22. Retrieving artifacts from the "graph.db" for media transfers of Telegram's Normal Chat thread (tool used: DB Browser for SQLite, Examined Device: Samsung A50).

6.2.3. Final Conclusions on Telegram

To sum up, certain types of actions modified the "cache4.db" database's WAL file but did not seem to alter data of the database's crucial "messages" table. The usage of the WAL file as journaling mode and its size, almost 5mb, provides a sufficient level of leniency to the DFIR Team members, because as the experiments proved it was able to retain a vast amount of deleted data. In the case of shared media files via an encrypted (secret) chat it was impossible to retrieve any content of them.

Chapter 7.

Dual Applications Feature

In this chapter usage of android's dual application feature was examined. This feature involves the creation of different accounts in the same device's application. It is branded with different names depending on the device like "Dual apps" for LG or "Dual Messenger" for Samsung (**Figure 98**). When this feature is enabled a second user is created in the filesystem with individual folders and files and separate read/execute/write permissions (**Figures 99** and **100**). Specifically for the creation of a Viber's second user's profile it is not mandatory to have a second SIM installed on the device. The second user's profile verification can be done through a separate device.



Figure 98. Use of dual Viber application. (Device used: Samsung A50)

▼Name ▲"	Description	Type Type	7
🛄 = com.viber.voip (171)	existing		
🛄. = databases (16)	existing		
google_app_measurement_local.db-journal	existing, already viewed	db-jou	
viber_data-journal	existing		
viber_data	existing		
cdr.db-journal	existing, already viewed	db-jou	
cdr.db	existing	db	
com.google.android.datatransport.events-journal	existing, already viewed	events	
com.google.android.datatransport.events	existing	events	
📄 exoplayer.db	existing	db	
📄 exoplayer.db-journal	existing, already viewed	db-jou	
viber_messages-journal	existing		
viber_messages	existing		
google_app_measurement_local.db	existing	db	
📄 mixpanel-journal	existing, already viewed		
inixpanel inixpanel	existing		
viber_prefs	existing		
viber_prefs-journal	existing		

Figure 99. Depiction of second Viber account's separate databases (path: «/user/95/com.viber.voip/databases», Examined Device: Samsung A50).

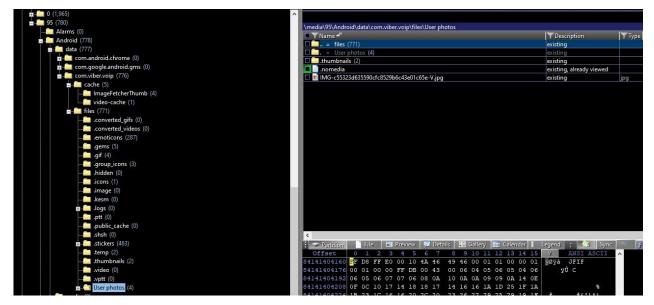


Figure 100. The second Viber user (95) has a separate folder structure for media located in «/media/95/Adroid/data/com.viber.voip/file/User photos». The original user of the device is located in folder (0) (Examined Device: Samsung A50).

The feature adds a new layer of complexity and difficulty to the investigation for two reasons. Firstly, if the user logs in from another device (e.g., the one with the SIM associated with the account) the investigator can only see the initial login screen of the application's second version. This action does not remove the user's data from the filesystem. A second reason is that if the user disables the application's second profile from the device's system settings, the corresponding folder structure is deleted, while the second profile's folder remains. This would be an indicator that additional examination like carving the acquired image for deleted databases must be done.

Chapter 8.

Applications Comparison

In this chapter the results of every examined instant messaging application are summarized and compered, in order to gain an overall view.

The volatility of each application's data in different conditions is presented in **Table 23**. The crucial data is never altered by the elapsed time or the changes at the device's status. The interaction with system notifications affected vital data only in the Viber application. Any message creation, reading and deletion affected all examined applications. WhatsApp and Telegram had some unique features (e.g., WhatsApp local backup) which might modify application's data. Regarding exchanged media files, Viber leaved significant number of artifacts on the device's internal storage. The same applied to certain occasions in Telegram (on normal chat messages). In case of WhatsApp application there was also a possibility of retrieving exchanged media files from its cloud server.

Applications	Time elapsed	Device's status	Interacting with Notifications	Message (Read/Sent/Delete)	Unique features	Media Retrieval from device's storage	Media Retrieval from app's Servers
Viber	x	X	\checkmark	\checkmark	x	\checkmark	×
WhatsApp	√ *	√ *	×	\checkmark	\checkmark	×	\checkmark
Telegram	x	√ *	x	\checkmark	\checkmark	√ **	×

Table 23. The volatility of each application in different conditions.

The usage of rollback journal as a journaling mode in Viber application rendered the data extremely volatile. After exchanging a few messages all the in-question records (deleted messages) were erased. On the other hand, the usage of the WAL as a journaling mode in WhatsApp and Telegram applications, resulted in the preservation of the in-question records after multiple actions (exchange messages/read/delete). The Telegram WAL's file size which was ten times bigger compared to the same file of the WhatsApp application, could retain a larger amount of data, including deleted ones.

All in all, the Viber application is the most volatile one related to retrieving deleted messages records and the least one when it comes to retrieving deleted media files. Telegram application is the least volatile in respect of retrieving deleted messages records, however the recovery of artifacts as for deleted media files depends on the chat thread type (normal/secret). The WhatsApp is the only application that even there were no traces left of a deleted exchanged media file on the examined device, it was possible to retrieve it from its Server.

^{*} Not crucial data.

^{**} Depending on the message's type (normal/secret).

Conclusions, Challenges & Future Work

To sum everything up, throughout the experiments done in this thesis several conclusions could be drawn about the most acceptable and forensic-sound first steps while handling an incident relating to instant messaging applications on an android-based mobile device. First and foremost, the seizure of the device must happen as soon as possible. By leaving the device at the suspect's possession there is a certainty that more messages will be exchanged and thus application's data will be further modified. There is always the element of chance as an unforeseen factor, such as multiple incoming messages, unexpectedly modifying irreversibly vital data (i.e., erase in-question deleted records). Once the device is seized the very first step is to block the incoming data, which will lead to the modification of database's volatile data and hence a potential key evidence removal. This could be achieved by disconnecting the device from the network (e.g., airplane mode, faraday bag or shutdown). The successful recovery of deleted content (messages/media) is determined by two key factor categories.

- > The first category is related to IR team's actions regarding the incident. These actions are:
 - Block the device's network connections. The used method is irrelevant (deactivate the device, airplane mode).
 - Not remove any devices notification messages.
 - Not try to create new messages. There are reported occasions in which members of the IR team tried to lure the third-party participants into revealing their identity by exchanges messages.
 - Not read unread messages.

As the experiments showed, once the aforementioned policies are followed the elapsed time does not modify the crucial data. The same applies to device's different statuses (powered on/off, reboot, plugged in, etc.).

- The second category is related to IR team's followed methodology regarding the evidence (data) forensic examination. This methodology must include:
 - Validation of the result. There is no known forensic tool able to decrypt and examine all chat messaging application databases. So, the results must be confirmed and validated by using multiple forensic tools. There are instances where the validation is feasible by diving into the raw data and examining the files in their hexadecimal representation.
 - Experimentation and Analysis. During the development of application versions different database schema/pragma statements and folder structure is implemented. The evidence's specific installed and running application version should be examined through research and development to test the validity of the results.

From the scope of digital evidence forensics, there are always three main challenges on two different areas. From a legal standpoint there are obstacles of bureaucracy which emerge and delay the technical procedures of starting the seizure and the examination of the suspect device. From a technical standpoint there are two main problems. On one hand there is the device's acquisition, which should be a physical one in order to access most of its data. Such acquisition is not always supported all devices' vendors and models. On the other hand, there is the decoding/parsing of the volatile database files is not always up to date or not even supported by forensic tool. It is more than necessary to have updated digital forensic tools and simplify the IR-related legislation.

As future work, more android applications should be examined to find out if there are more volatile data on their databases and how react in different occasions.

References

- Albrecht, M. R., Mareková, L., Paterson, K. G. & Stepanovs, I. (2021), Four Attacks and a Proof for Telegram, In 43rd IEEE Symposium on Security and Privacy (IEEE S&P 2022), San Francisco, CA, USA, 22 – 26/05/2022, deposited on: 21/09/2021, DOI: 10.3929/ethz-b-000506353.
- Anglano, C., Canonico, M. & Guazzone, M. (2017), Forensic analysis of telegram messenger on android smartphones, Digital Investigation, vol. 23, pp. 31-49, published on: 20/09/2017, DOI: 10.1016/j.diin.2017.09.002.
- Anglano, C., Canonico, M., & Guazzone, M. (2016), Forensic analysis of the ChatSecure instant messaging application on android smartphones, Digital Investigation, vol 19, 2016, pp. 44-59, published on: 15/10/2016, DOI: 10.1016/j.diin.2016.10.001, last accessed 13/11/2021.
- Anglano, C. (2014), Forensic Analysis of WhatsApp Messenger on Android Smartphones, Digital Investigation, vol. 11(3), pp. 201-213, published on: 27/05/2014, DOI: 10.1016/j.diin.2014.04.003.
- Angus, M. (2018), WhatsApp server-side media persistence, Digital Investigation, vol. 25, pp. 114-115, published on: 25/04/2018, DOI: 10.1016/j.diin.2018.04.005.
- Bair, J. (2017), Seeking the Truth from Mobile Evidence, Basic Fundamentals, Intermediate and Advanced Overview of Current Mobile Forensic Investigations, Elsevier, London, UK, published on: 13/11/2017, ISBN: 978-0-12-811056-0.
- Botha, J., Van 't Wout, C. & Leenen, L. (2019), Proceedings of the 18th European Conference on Cyber Warfare and Security (ECCWS 2019), Hosted by University of Coimbra, Portugal, 4-5/7/2019, Academic Conferences and Publishing International Limited Reading, UK, pp. 55-62, ISBN: 978-1-912764-28-0.
- SWGDE (2014), Digital and Multimedia Evidence (Digital Forensics) as a Forensic Science Discipline, Scientific Working Group on Digital Evidence, published on: 05/09/2014, retrieved from: https://drive.google.com/file/d/1OBux0n7VZQe7HSgObwAtmhz5LgwvX0o Y/view, last accessed 31/01/2022.
- Easttom, C. (2021), An In-Depth Guide to Mobile Device Forensics, CRC Press, Boca Raton, Florida, published on: 22/10/2021, DOI: 10.1201/9781003118718.
- Easttom, C. (2021), Digital forensics, investigation, and response Fourth Edition, Information Systems Security & Assurance Series, Jones & Bartlett Learning, Burlington, Massachusetts, USA, published on 24/08/2021, ISBN 9781284226065.
- Ghannam, H. A. (2018), Forensic Analysis of Artifacts of Giant Instant Messaging "WhatsApp" in Android Smartphone. Journal of Applied Information, Communication and Technology, vol.5(2), pp. 63-72, published on: 03/10/2018, DOI: 10.33555/ejaict.v5i2.55.
- Gogolin, G. (2021), Digital Forensics Explained Second Edition, CRC Press, Boca Raton, Florida, published on: 12/04/2021, ISBN 9780367503437.

- Jakobsen, J. & Orlandi, C. (2016), On the CCA (in) security of MTProto, In Proceedings of the 6th Workshop on Security and Privacy in Smartphones and Mobile Devices, pp. 113-116, published on: 10/2016, retrieved from URL: https://eprint.iacr.org/2015/1177.pdf.
- Lee, J., Choi, R., Kim, S. & Kim, K. (2017), Security analysis of end-to-end encryption in Telegram, 2017 Symposium on Cryptography and Information Security (SCIS 2017), Naha, Japan, The Institute of Electronics, Information and Communication Engineers, published on: 01/2017, retrieved from URL: https://caislab.kaist.ac.kr/publication/paper_files/2017/SCIS17_JU.pdf.
- Hermawan, T., Suryanto, Y., Alief, F. & Roselina, L. (2021), Android Forensic Tools Analysis for Unsend Chat on Social Media, 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI), 2020, pp. 233-238, published on: 13/01/2021, DOI: 10.1109/ISRITI51436.2020.9315364, last accessed 13/11/2021.
- Meng, C., Baier, H. (2019), bring2lite: A Structural Concept and Tool for Forensic Data Analysis and Recovery of Deleted SQLite Records, Digital Investigation, vol. 29, pp. 31-41, published on: 14/07/2019, DOI: 10.1016/j.diin.2019.04.017, last accessed 09/11/2021.
- Mahalik, H., Bair, J, Brignoni, A., Coates, S., Dickinson, M., Epifani, M., Hyde, J., Katalov, V., Koenig, S., Lorentz, P., Poirier, C., Reiber, L., Westman, M., Williamson, M., Whiffin, I. & Skulkin, O. (2021), Six Steps To Successful Mobile Validation, SANS Institute Reading Room site.
- Mirza, M. M., Salamh, F. E. & Karabiyik, U. (2020), An Android Case Study on Technical Anti-Forensic Challenges of WhatsApp Application, 2020 8th International Symposium on Digital Forensics and Security (ISDFS), pp. 1-6, published on: 15/06/2020, DOI: 10.1109/ISDFS49300.2020f.9116192.
- Nemetz, S., Schmitt, S. & Freiling, F. (2018), A standardized corpus for SQLite database forensics, Digital Investigation, vol. 24, pp. 121-130, published on: 21/03/2018, DOI: 10.1016/j.diin.2018.01.015, last accessed 09/11/2021.
- Onovakpuri, P. (2018), Forensics Analysis of Skype, Viber and WhatsApp Messenger on Android Platform, International Journal of Cyber-Security and Digital Forensics (IJCSDF), Hong Kong, vol. 7, no. 2, pp. 119-131, published on: 06/2018, DOI: 10.17781/P002369.
- Orr, D. A., Castro, A. (2018), Whatsapp Messenger on the Android Platform: A Forensic Examination of a Physical Device, Forensic Science: Methods and Techniques, issue 2018(1), pp. 04-19, published on: 20/09/2018, DOI: 10.33513/FSMT/1801-05.
- Pawlaszczyk, D., Hummert, C. (2021), Making the Invisible Visible –Techniques for Recovering Deleted SQLite Data Records, The International Journal of Cyber Forensics and Advanced Threat Investigations under a Creative Commons License, vol. 1(1-3), pp. 27-41, published on: 15/02/2021, DOI: 10.46386/ijcfati.v1i1-3.17, last accessed 09/11/2021.
- Raji, M., Wimmer, H. & Haddad, R. J. (2018), Analyzing Data from an Android Smartphone while Comparing between Two Forensic Tools, SoutheastCon 2018, 2018, pp. 1-6, published on: 04/10/2018, DOI: 10.1109/SECON.2018.8478851, last accessed 14/11/2021.

- Rathi, K., Karabiyik, U., Aderibigbe, T. & Chi, H. (2018), Forensic analysis of encrypted instant messaging applications on Android, 2018 6th International Symposium on Digital Forensic and Security (ISDFS), pp. 1-6, published on: 07/05/2018, DOI: 10.1109/ISDFS.2018.8355344.
- Saribekyan, H. & Margvelashvili, A. (2017), Security Analysis of Telegram 6.857 Final Project, published on 2017, retrieved from URL: https://www.semanticscholar.org/paper/Security-Analysis-of-Telegram-6-.-857-Final-Project-Saribekyan/fabe0871280513af67b23438ce17ca8c6c79a2d8, last accessed on 05/12/2021.
- Skulkin, O., Tindall, D. & Tamma, R. (2018), Learning Android Forensics Analyze Android Devices With The Latest Forensic Tools And Techniques, 2nd Edition, published by Packt Publishing Ltd, December 2018, Birmingham UK, pp. 73-94 & 135-148, ISBN: 978-1-78913-101-7.
- Sutikno, T., Handayani, L., Stiawan, D., Riyadi, M.A. & Subroto, I. (2016). WhatsApp, Viber and Telegram: which is the Best for Instant Messaging?, International Journal of Electrical and Computer Engineering (IJECE), vol. 6, no. 3, pp. 909-914, published on: 26/05/2016, DOI: 10.11591/ijece.v6i3.10271.
- Tamma, R., Skulkin, O., Mahalik, H., Bommisetty, S. (2020), Practical Mobile Forensics Fourth Edition, Forensically investigate and analyze iOS, Android, and Windows 10 devices, Packt Publishing, Birmingham, UK, published on: 04/2020, ISBN 978-1-83864-752-0.
- Vasilaras, A. (2016), Android Forensics in Android Smart Devices, MSc Thesis, Advanced Informatics and Computing Systems, Security Management Engineering, Department of Informatics, University of Piraeus, created on 03/2016.
- Zhang, H., Chen, L. & Liu, Q. (2018), Digital Forensic Analysis of Instant Messaging Applications on Android Smartphones, 2018 International Conference on Computing, Networking and Communications (ICNC), pp. 647-651, published on: 21/06/2018, DOI: 10.1109/ICCNC.2018.8390330.

Internet Sites/Forums/Blogs:

- askclees1711 (2020), SQLite Databases at hex level, AskClees, published on: 20/10/2020, retrieved from URL: https://askclees.com/2020/11/20/sqlite-databases-at-hex-level, last accessed 13/11/2021.
- Belkasoft (2021), ANDROID VIBER FORENSICS WITH BELKASOFT X, retrieved from URL: https://belkasoft.com/android_viber_forensics, last accessed 13/11/2021.
- Belkasoft (2021), Whitepaper: WhatsApp Forensics on Smartphones, retrieved from URL: https://belkasoft.com/whitepaper_whatsapp_forensics, last accessed 24/11/2021.
- Воіко (Бойко), М. (2018), TELEGRAM ELECTRONIC CORRESPONDENCE RESEARCH ON ANDROID DEVICES («ДОСЛІДЖЕННЯ ЕЛЕКТРОННОЇ ПЕРЕПИСКИ TELEGRAM НА ПРИСТРОЯХ ANDROID»), Kiev, 2018, CACHE4.DB FILE OF TELEGRAM FOR ANDROID (PART 1), Digital Forensics (dflab.blogspot), published on: 03/01/2019, retrieved from URL: https://dflab.blogspot.com/2019/01/cache4db-file-of-telegram-forandroid.html, last accessed on 05/12/2021.

- Cimpanu, C. (2021), FBI document shows what data can be obtained from encrypted messaging apps, created on: 30/11/2021, retrieved from URL: https://therecord.media/fbi-document-shows-what-data-can-be-obtained-from-encrypted-messaging-apps, last accessed 09/12/2021.
- Cynet (2021), NIST Incident Response Plan: Building Your Own IR Process Based on NIST Guidelines, last updated on 03/11/2021, retrieved from URL: https://www.cynet.com/incident-response/nist-incident-response, last accessed 09/11/2021.
- Cynet (2021), Incident Response SANS: The 6 Steps in Depth, last updated on 03/11/2021, retrieved from URL: https://www.cynet.com/incident-response/incident-response-sans-the-6-steps-in-depth, last accessed 09/11/2021.
- Cynet (2021), Incident Response Team: A Blueprint for Success, last updated on 03/11/2021, retrieved from URL: https://www.cynet.com/incident-response/incident-response-team-a-blueprint-for-success, last accessed 09/11/2021.
- Cynet (2021), What is Incident Response? People, Processes and Tools [2021 Guide], last updated on 06/2021, retrieved from URL: https://www.cynet.com/incident-response, last accessed 09/11/2021.
- Dean, B. (2021), How Many People Use Telegram in 2021? 55 Telegram Stats, BACKLINKO, published on: 14/10/2021, retrieved from URL: https://backlinko.com/telegram-users, last accessed 05/12/2021.
- dfirfpi (2020), teleparser, ZENA FORENSICS something about digital forensics and something not, published on: 06/04/2020, retrieved from URL: https://blog.digital-forensics.it/2020/04/teleparser.html, last accessed on 05/12/2021.
- ElDavoo, D. (2021), WhatsApp Crypt14 Database Decrypter, GitHub, created on: 29/08/2021, retrieved from URL: https://github.com/ElDavoo/WhatsApp-Crypt14-Decrypter, last accessed 24/11/2021.
- González, J. (2012), From PowerOn to Android The Boot Sequence, My Cellar Door, published on: 24/08/2012, retrieved from URL: https://javigon.com/2012/08/24/from-poweron-toandroid-the-boot-sequence, last accessed 14/11/2021.
- Macdonald, I. (2020), multidisabler-samsung, GitHub, created on: 30/12/2020, retrieved from URL: https://github.com/ianmacd/multidisabler-samsung, last accessed 09/11/2021.
- Magisk Manager (2021), Download Magisk Manager Latest Version 23.0 For Android 2021, retrieved from URL: https://magiskmanager.com, last accessed 09/11/2021.
- Rakuten Viber (2021), Viber Encryption Overview, retrieved from URL: https://www.viber.com/app/uploads/viber-encryption-overview.pdf, last accessed 22/11/2021.
- Precise Law (2017), What is a Forensic Image?, Precise, published on: 09/11/2017, retrieved from URL: https://discovery.precise-law.com/what-is-a-forensic-image, last accessed 09/11/2021.

- Rahmanpour, A. (2019), Preparing For Computer Forensics Investigation (Partitions), Threat Hunting, published on: 31/12/2019, retrieved from URL: https://www.threathunting.se/2019/12/31/preparing-for-computer-forensics-investigation-partitions, last accessed 09/11/2021.
- redymedan (2021), [TWRP][Root] Unofficial TWRP 3.5.0 Android 11 for Galaxy A50, created on: 13/03/2021, XDA Developers, retrieved from URL: https://forum.xdadevelopers.com/t/twrp-root-unofficial-twrp-3-5-0-android-11-for-galaxy-a50.4246581, last accessed 09/11/2021.
- SalvationDATA (2020), [Case Study] Mobile Forensics: Forensic Data Extraction from Android Devices Using ADB (Android Debug Bridge) Part II, published on 05/06/2020, retrieved from URL: https://blog.salvationdata.com/2020/06/05/case-study-mobile-forensicsforensic-data-extraction-from-android-devices-using-adb-android-debug-bridge-part-ii, last accessed 09/11/2021.
- Singh S. (2020), WhatsApp has a fake news problem: Here's everything that it's doing to fight it in wake of Coronavirus outbreak, Financial Express, created on: 07/04/2020, retrieved from URL: https://www.financialexpress.com/industry/technology/whatsapp-has-a-fakenews-problem-heres-everything-that-its-doing-to-fight-it-in-wake-of-coronavirusoutbreak/1921515/, Last accessed 08/12/2021.
- SQLite.org (2021), Atomic Commit In SQLite, retrieved from URL: https://www.sqlite.org/atomiccommit.html, last accessed 09/11/2021.
- SQLite.org (2021), PRAGMA Statements (journal mode), retrieved from URL: https://www.sqlite.org/pragma.html#pragma_auto_vacuum, last accessed 09/11/2021.
- SQLite.org (2021), Database File Format, retrieved from URL: https://www.sqlite.org/fileformat2.html, last accessed 09/11/2021.
- SQLite.org (2021), PRAGMA Statements, retrieved from URL: https://www.sqlite.org/pragma.html#pragma_secure_delete, last accessed 09/11/2021.
- SQLite.org (2021), PRAGMA Statements (journal mode), retrieved from URL: https://www.sqlite.org/pragma.html#pragma_journal_mode, last accessed 09/11/2021.
- SQLite.org (2021), PRAGMA Statements (journal mode), retrieved from URL: https://www.sqlite.org/pragma.html#pragma_journal_size_limit, last accessed 09/11/2021.
- O' Dea, S. (2021), Forecast number of mobile devices worldwide from 2020 to 2025 (in billions), statista, published on 24/09/2021, retrieved from URL: https://www.statista.com/statistics/245501/multiple-mobile-device-ownership-worldwide, last accessed 31/01/2022.
- Tahiri, S. (2016), Android forensic logical acquisition, INFOSEC, published on: 16/03/2016, retrieved from URL: https://resources.infosecinstitute.com/topic/android-forensic-logical-acquisition, last accessed 09/11/2021.
- Telegram (2021), The Evolution of Telegram, retrieved from URL: https://telegram.org/evolution, last accessed on 05/12/2021.

- Telegram (2021), End-to-End Encryption, Secret Chats, retrieved from URL: https://core.telegram.org/api/end-to-end, last accessed on 05/12/2021.
- WhatsApp (2021), WhatsApp Encryption Overview, Technical white paper, retrieved from URL: https://www.whatsapp.com/security/WhatsApp-Security-Whitepaper.pdf, last accessed on 22/11/2021.
- Xiaomi MIUI (2021), Unlock Your Mi Device, retrieved from URL: https://en.miui.com/unlock/download_en.html, last accessed 09/11/2021.
- zefie (2017), [RECOVERY][UNOFFICIAL][US997/H870][rel_o2/rel_t2][2018-06-08] Melina TWRP v3.2.1, created on: 20/12/2017, XDA Developers, retrieved from URL: https://forum.xdadevelopers.com/t/recovery-unofficial-us997-h870-rel_o2-rel_t2-2018-06-08-melina-twrpv3-2-1.3722199, last accessed 09/11/2021.
- 99Content (2021), Viber Statistics, 99FIRMS BLOG, retrieved from URL: https://99firms.com/blog/viber-statistics, last accessed 13/11/2021.
- 99Content (2021), Whatsapp Statistics, 99FIRMS BLOG, retrieved from URL: https://99firms.com/blog/whatsapp-statistics, last accessed 23/11/2021.

Software, Tool and Utilities' Official Sites:

- Access Data (FTK): https://accessdata.com/products-services/forensic-toolkit-ftk
- Acquire Forensics (SQLite Forensic Explorer): https://www.acquireforensics.com/services/database-forensics.html

aLEAPP: https://github.com/abrignoni/ALEAPP

Andriller: https://0x1.gitlab.io/phone/Andriller

Android Gallery BlobCache Files: https://github.com/slo-sleuth/android_blob_cache

Autopsy: http://sleuthkit.org/autopsy/docs/user-docs/4.19.2

Autopsy-Plugins: https://github.com/markmckinnon/Autopsy-Plugins

Belkasoft: https://belkasoft.com/android

bring2lite: https://github.com/bring2lite/bring2lite

Cellebrite UFED: https://www.cellebrite.com/en/ufed

Cellebrite Physical Analyzer: https://www.cellebrite.com/en/physical-analyzer

Forensic SQLite Data Recovery Tool (FQLite): https://www.staff.hs-mittweida.de/~pawlaszc/fqlite

Forensic Toolkit for SQLite: https://sqliteforensictoolkit.com/sqlite-forensic-toolkit

Hancom (MD-Series): http://www.hancomgmd.com/product & https://www.youtube.com/watch?v=YVkh7qFBVHc&ab_channel=TeelTechnologiesCan ada

KS DB Merge Tools for SQLite: https://www.db-merge-tools.net/sqlite-diff-merge-overview.html

- Magnet Forensics (Axiom & Acquire): https://www.magnetforensics.com/products/magnet-axiomcyber & https://www.magnetforensics.com/resources/magnet-acquire
- Melina TWRP v3.2.1: https://forum.xda-developers.com/t/recovery-unofficial-us997-h870-rel_o2-rel_t2-2018-06-08-melina-twrp-v3-2-1.3722199
- MOBILedit: https://www.mobiledit.com/forensic-express-personal

MSAB (XRY): https://www.msab.com/law-enforcement/forensic-examiners

multidisabler-samsung: https://github.com/ianmacd/multidisabler-samsung

ncat: https://nmap.org/ncat

Oxygen forensics: https://www.oxygen-forensic.com/en/products/oxygen-forensic-detective

Paraben Forensics (E3 Forensic Platform): https://paraben.com/free-dfir-tools

Salvationdata (DBF6300): https://www.salvationdata.com/database-forensic-analysis-system

Sqlitebrowser: https://github.com/sqlitebrowser/sqlitebrowser

SQLite Expert database administration: http://www.sqliteexpert.com

Undark: https://github.com/alitrack/undark

WhatsApp Crypt14 Database Decrypter: https://github.com/EIDavoo/WhatsApp-Crypt14-Decrypter.

WhatsApp Key/DB Extractor: https://github.com/EliteAndroidApps/WhatsApp-Key-DB-Extractor

whatsapp-media-decrypt: https://github.com/sh4dowb/whatsapp-media-decrypt

X-Ways: http://www.x-ways.net/forensics

Appendix

A1.

A/A	Sender	Receiver	Content	Acquisition
1	Bad Actor_2	Edgar	How are you my friend?	data0_0_1
2	Bad Actor_2	Edgar	It is been a long time since we talked	data0_0_2
3	Bad Actor_2	Edgar	Are you in Boston?	data0_1
4		data0_1_1		
5	Michael	Bad Actor_2	5 lyrics from the poem "The Crow" (1-5)	data0_2
6	Bad Actor_2	Edgar	Did Mike really send you a great poem?	data0_3
			He should send me more	
			Who wrote it?	
7		Reading	a random message bot message	data0_3_1
8	Bad Actor_1	Bad Actor_2	5 lyrics from the poem "The Crow" (6-10)	data0_4
9		data0_4_1		
10	Bad Actor_2	Bad Actor_1	Mike sent some lyrics from the same poem	data1
			Where did you find it?	
11	Deleted chat thread with Edgar		data2	
12	Bad Actor_1	Bad Actor_2	5 lyrics from the poem "The Crow" (11-15)	data3
13	Reading the messages sent from Michael			data3_1
14	Bad Actor_2	Michael	Scenario (10 messages from which 2 photos) (see Appendix A2)	data4
				(Non-Deleted)
				data4_1
				(Deleted)
15		data5		
16	Edgar	Bad Actor_2	5 lyrics from the poem "The Crow" (16-20)	data6
17		data7		
18		data8		
19		data9		
20	Bad Actor_2	Michael	WHAT IS WRONG WITH YOU TWO AND THIS POEM	data10
21		data11		
22	Michael	Bad Actor_2	1 lyric from the poem "The Crow" (21) Rebooting the device	data12
23		data13		
24		data14		
25	Michael	Bad Actor_2	4 lyrics from the poem "The Crow" (22-25)	data15
26	Edgar	Bad Actor_2	5 lyrics from the poem "The Crow" (26-30)	data16
27	Bad Actor_1	Bad Actor_2	5 lyrics from the poem "The Crow" (31-35)	data17

A2.

A/A	Sender	Receiver	Action / Content	Parameters
1	Bad Actor_1	Michael	Message: I am sending the photo	

2	Bad Actor_1	Michael	Action: Sending photo from gallery	
3	Bad Actor_1	Michael	Message: Did you receive it?	
4	Michael	Bad Actor_1	Message: Yes send the second one	
5	Bad Actor_1	Michael	Action: Sending photo from app's camera feature	
6	Bad Actor_1	Michael	Message: This will cost you 1000 euros	
7	Michael	Bad Actor_1	Message: We agreed on five hundred	Timebomb at 60 seconds
8	Bad Actor_1	Michael	I want more	Timebomb at 60 seconds
9	Michael	Bad Actor_1	WE WILL SEE	Timebomb at 60 seconds
10	Bad Actor_1	Michael	I think they saw me. I must delete those	Timebomb at 60 seconds