ISSN 2460-9056
http://socj.telkomuniversity.ac.id/ijoict/

Intl. Journal on ICT
Vol. 6, Issue. 1, June 2020. pp. 1-9
doi: 10.21108/IJOICT.2020.61.489

# WhatsApp Chat Visualizer: A Visualization of WhatsApp Messenger's Artifact Using the Timeline Method

Hardiansyah Shidek [1], Niken Dwi Wahyu Cahyani [1*], Aulia Arif Wardana [1]

[1] *School of Computing, Telkom University*
*Bandung, Indonesia*

[*] nikencahyani@telkomuniversity.ac.id

**Abstract**

WhatsApp is a medium that everyone can use to interact and to share information effectively and efficiently. However, it can be misused for criminal activities. Analyzing WhatsApp' artifacts is quite challenging as the suspect may have a lot of random conversational data to be considered. This makes it difficult for the trial process to obtain digital traces that can be identified in malicious activities such as knowing who was involved when the conversation was held and the timespan. Therefore, in this research, a social media investigation of WhatsApp was carried out by acquiring data from rooted Android devices that were used as target devices for forensic activities. A python-based application is developed to show the content of the conversation, and a web-based application is presented to visualize the data using the Timeline method. Experimental results in this research display important timeline information such as information about who was involved when and what time the conversation was carried out with the suspect.

**Keywords:** social media, court, forensic, visualization, timeline.

## I. Introduction

The development and the usage of social media that continue to grow are followed by the increasing number of digital devices under criminal investigation. This has caused major problems that affect the social media investigation process in order to present the results of fast and precise analysis and raises a challenge in social media investigative research (Pomalingo et al., 2019).

One type of Social Media is WhatsApp Messenger (WA). The last report on WhatsApp statistics (dating back to Q1 2020) stated the increasing number of WhatsApp Messenger users and recorded two billion monthly users over 180 countries, as illustrated in Fig. 1. According to Koum & Acton (2016), WhatsApp Messenger is the most popular messenger application, followed by Facebook Messenger and WeChat. WA is supported by an encryption feature to ensure the data security of its users. However, the popularity and features provided by WA can be misused by the public for criminal purposes, for example, in terrorism crime. Nevertheless, under forensically sound digital investigation process, the authorities can use the data in WA as evidence (Umar, Riadi, Zamroni, et al., 2018). To fulfill all the evidence to support the trial, the forensic team must present the evidence in a clear and understandable manner for various groups, especially people who do not understand the digital forensics (Casey, 2011).

Forensic methods are needed to ensure the success of the process of retrieving these data, including the contents of text messages and where text messages are sent. The data is then visualized to get information about who is involved, time information, tracking activities, and timeline matching (Pomalingo et al., 2019).

Received on 13 May 2020. Accepted on 28 May 2018

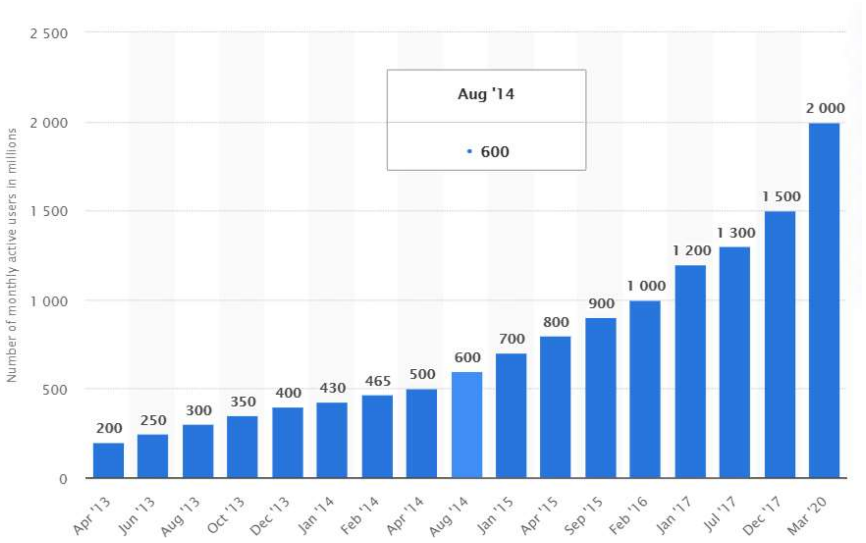


Fig. 1. WhatsApp users statistic (Clement, 2020)

In order to facilitate the illustration of the results of the WhatsApp investigation, this research develops a system using the Timeline visualization method. Timeline visualization is a method in which a series of events and visualization is recorded to present a series of events along with a graph that has a time axis and place it at the point of time where something happens or the range where something ends (Plaisant, Milash, Rose, Widoff, & Shneiderman, 1996). In this research, the Timeline method is chosen from the many of visualization methods, because according to Nguyen, Xu, Walker, & Wong (2016) the Timeline method can help coordinate a series of events in a chronological dataset to facilitate the uncover temporal relationships and reduce the analyst's effort in memorizing an event. Timeline visualization usually uses a rectangle graph or chart to show the time of events and horizontal lines for one interval. This usually has an explanation of text that describes the event. To display the visualized data, it will represent different colors, icons, or shapes (Plaisant et al., 1996).

A thorough forensic analysis has been done by Anglano (2014) on the WhatsApp Messenger application. The research analyzed WhatsApp Messenger artifacts on an Android device. Another research conducted by Yadav, Prakash, Dayal, & Singh (2020) discusses an analysis of the WhatsApp acquisition process that uses several tools, compare them, and show which application that is more suitable for forensic analysis in WhatsApp Messenger. There is also research conducted by C. F. Tassone, Martini, & Choo (2017) about the visualization of the results of digital forensic analysis. They stated that visualization could be done by summarizing and narrowing the data needed to be observed by the forensic team. Based on this literature search, we found that currently, there is no research related to the visualization of the results of an analysis of WhatsApp artifacts on an Android Smartphone, while visualization can be used as supporting evidence in some cases.

Therefore, this research proposes the visualization of the WhatsApp Messenger artifact that focused on the chat database using the Timeline method that can support efforts in mapping data from various existing information for criminal investigation purposes. The data visualization method, in the form of a connected timeline graph, is used to present the results of the analysis.

## II. LITERATURE REVIEW

Many researchers have contributed to forensic analysis methodologies on social media apps and data visualization. Anglano (2014) focused on forensic analysis of WhatsApp Messenger on the Android platform. He thoroughly analyzed the artifacts of WhatsApp Messenger and categorized these artifacts in detail, but has not mentioned about the visualization. Then Yadav et al. (2020) analyzed the acquisition process of WhatsApp data using various tools and compared them to show which tool is better for the acquisition of WhatsApp data on Android devices.

C. Tassone, Martini, & Choo (2017) conducted a survey of several digital visualization methods currently being developed and also used in the academic field. To test some of these visualization methods, they identified several court cases that were related to digital evidence. They highlighted the device used and the data practitioner in charge when analyzing data of the forensic process. They also identified several visualization methods that could visualize different types of data, one of which was the timeline visualization method to visualize data related to time chronicles. However, this paper does not explain what type of data they visualize. Olsson & Boldt (2009) focus on making a timeline visualization tool. This is similar to our works. However, they use the tool to visualize the event activity of the evidence collection.

Hariyadi, Winarno, & Luthfi (2016) analyzed artifacts of Blackberry Messenger on the Blackberry platform and visualized data analysis results using the wordcloud method. Meanwhile, Anwar & Riadi (2017) focused on analyzing the web-based WhatsApp Messenger, but no data visualization. Similarly, Pomalingo, Sugiantoro, & Prayudi (2019) presented forensic analysis on social media, and added visualization of what social media the person is using and visualized it using directed graph method

In this research, a visualization system will be developed to contribute to the gap of artifacts visualization of forensic analysis on WA apps, by making a case study of crimes prioritized for investigation in Indonesia, in which data is fictitious and focused on terrorism crimes. Timeline visualization is used in order to clearly know the time span of the incident.

### A. WhatsApp Messenger

Thakur (2013) reported that WhatsApp Messenger generated artifacts on Android operating system devices that are stored in certain files, including the contact database, chat database, and a key file are listed in Table I.

From Table I, WhatsApp Messenger stores all message information in the msgstore.db file which is located in the preferences directory, which can be analyzed to enable the reconstruction of the chronology of the messages exchanged, such as to determine when messages were exchanged, the substance of the data conveyed, the people engaged with the discussion, what, and when the recipient has actually received the message (Anglano, 2014)

WhatsApp has end-to-end encryption that came with the various version of encryption depends on what version of WhatsApp is installed. This ensures only the owner of the device and the person his communicating with can read either what is sent or received, and nobody in between, not even WhatsApp itself (Umar et al., 2018). Due to this security system give some difficulties for investigators to analyze the device, forensic investigation support tools must be aware of the development of the WhatsApp system and keep up with it (Sai, Prasad, & Dekka, 2015).

### B. Timeline Visualization

The timeline visualization method is the most popular method for temporary data. Each object or event is visualized as a line, indicating the start and the end time (Wang & Yuan, 2014). Timeline is a graph that illustrates how a set of events over time. If managing a software project and want to illustrate who is doing what and when, or if holding a conference and need to schedule a meeting room, the timeline is often a very appropriate visualization choice. One popular type of timeline is the Gantt Chart (Pavlik & MacIntoch, n.d.).

TABLE I
WHATSAPP ARTIFACTS

| Num | Name | Path | File |
|---|---|---|---|
| 1 | contacts database | /data/data/com.whatsapp/databases | wa.db (SQLite File Format) |
| 2 | cipher key | /data/data/com.whatsapp/files | key (XML File Format) |
| 3 | chat database files | /data/data/com.whatsapp/databases | msgstore.db.crypt<version> (Encrypted SQLite File Format) |
| 4 | chat backup database | /sdcard/Whatsapp/Databases | msgstore-<date>.db.crypt<version> |
| 5 | log files | /data/data/com.whatsapp/files/Logs | whatsapp.log, whatsapp-< date>.log |
| 6 | received files | /sdcard/Whatsapp/Media | different kind of file type such as picture, videos, etc |
| 7 | sent files | /sdcard/Whatsapp/Media/Sent | different kind of file type such as picture, videos, etc |
| 8 | user settings and preferences | /data/data/com.whatsapp/ files | config file |

There are many different methods related to visualization of time-dependent data information. Mostly, this visualization utilizes a course of events representations that mastermind set of information data along with a period pivot (Stab, Nazemi, & Fellner, 2010). For example, Allen (1995) uses an interactive timeline to visualize information data based on an interactive textbook or a digital library. There are numerous courses of events visualizations that appear on the screen into a rectangle bar shape to arrange the imagined occasions in a level or even in a vertical shape (Stab et al., 2010).

Timelines are used in many domains, including autobiography, data analysis, health records, music release times, and history. Meanwhile, visualizations are commonly classified based on specific categories or sets. For instance, organizations, have more than one task segment or division; simply like news stories that have various classifications, i.e., governmental issues, health, sports, sci-tech, and others (Plaisant et al., 1996). An example of a Timeline graph can be seen in Fig. 2.

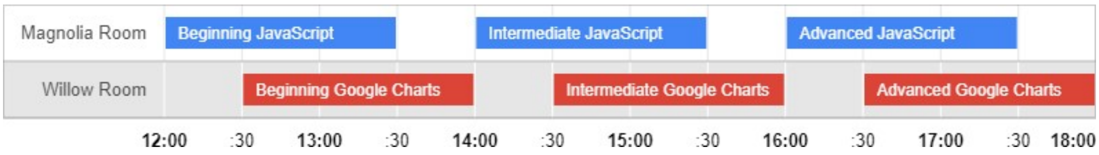


Fig. 2. Example of Timeline Chart (Google, 2020)

## III. RESEARCH METHOD

The stages of this research are illustrated in Fig. 3, to explain the flow of the research process. In order to obtain digital text evidence, a logical acquisition using FTKImager tools is conducted. The WhatsApp Messenger application has digital evidence in the form of an encrypted SQLite database file, msg-store.db.crypt12.

The file can be obtained in the Device Storage/Whatsapp/Databases/msgstore.db.crypt12 directory. The file is encrypted by WhatsApp and to open it requires a key file that can be obtained in the /data/data/com.whatsapp/files/key directory. Root access is required to get the key file. In this research, the WhatsApp Messenger application with version 2.20.69 is used on Android devices with Android 5.1 that has been given a root access.

Key file and msgstore.db.crypt12, which have been acquired, are then decrypted using decrypt12 tools to allow us to read the file. Without the key file, the msgstore.db.crypt12 file will only display blank data. After decrypted, it will generate a decrypted msgstore.db file, which means the file has been decrypted and can be read. In order to do this, we made a tool called WhatsApp Reader.

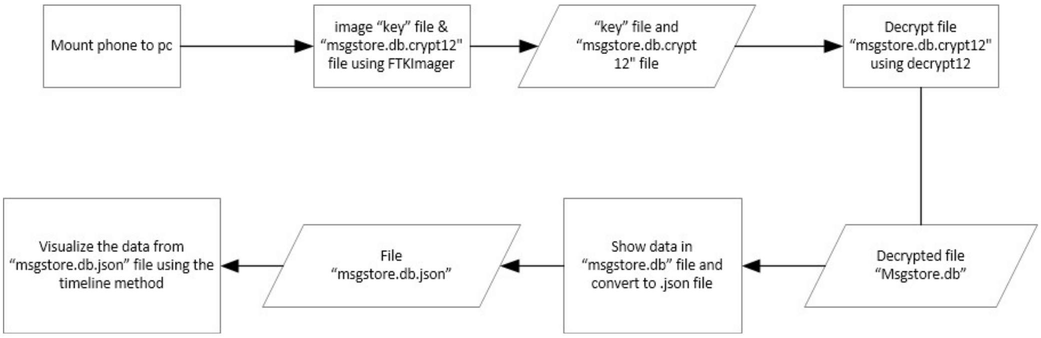

Mount phone to pc
image "key" file & "msgstore.db.crypt12" file using FTKImager
"key" file and "msgstore.db.crypt 12" file
Decrypt file "msgstore.db.crypt12" using decrypt12
Decrypted file "Msgstore.db"
Show data in "msgstore.db" file and convert to .json file
File "msgstore.db.json"
Visualize the data from "msgstore.db.json" file using the timeline method

Fig. 3. Research Process Flow

### A. *Acquisition on Rooted Android Device*

Every Android device has different ways for its rooting process. Therefore, there is no single root application can root all devices. In this research, we use the Oppo A33w Android device with version 5.1, and we root the device via recovery mode using fileroot.zip. After the rooting process, the phone can give permission to access the root file, where the WhatsApp key file exists. A forensic image of the WhatsApp key and an encrypted file is obtained using FTK Imager and use the USB Flash Drive as the evidence devices after we mount the phone to PC, and we proceed to process WhatsApp data on the PC that runs Windows 10 operating system.

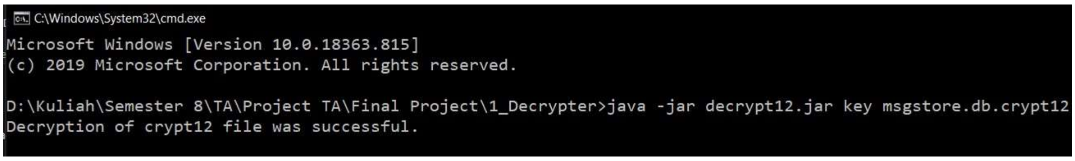
### B. *Decrypt and Read the Data*

WhatsApp utilizes a 192-bit AES algorithm to encrypt the message database (Anglano, 2014). However, according to Cortjens, Spruyt, & Wieringa (2012), it can be decrypted using a key file that is stored in the root directory on devices that has a WhatsApp installed on it. In order to do the decryption, we use a tool called decrypt12. The decrypt12 tool decrypts the msgstore.db.crypt12 file into msgstore.db, which can be read using the DB Browser for SQLite application. The database contains various data in WhatsApp, such as call logs, media, chat lists, and messages. For our purposes, we only use chat lists and messages, and we use our own-created tool called WhatsApp Reader to filter and display the data. WhatsApp Reader itself can show all chat messages and convert them into a JSON file. We use the sqlite3 library to filter the chat data from the Chat List table and select the attributes that are listed in Table II.

TABLE II
CHAT LIST ATTRIBUTES

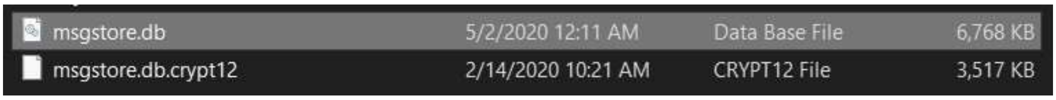
| **Attribute** | **Function** |
|---|---|
| `primary_key` | the primary key of the record |
| `contact_id` | contact id |
| `contact_name` | name or number of the contact whom the suspect make a contact with |
| `from_me` | a status which indicates that the message is sent by the suspect |
| `msg_date` | the date of the message |
| `Text` | content of the conversation |
| `msg_status` | message status |
| `last_message_date` | date of last time the message are sent or received |

For decrypting file msgstore.db.crypt12, the decrypt12 tool is utilized. It is a java program in the form of a JAR file. In order to use the tool, we need to enter the command via command prompt on the operating system. The command is "java -jar decrypt12.jar key msgstore.db.crypt12", it can be seen in Fig. 4. It should be noted that to decrypt the msgstore.db.crypt12 file, a key file is required. If the key file does not exist, it cannot decrypt the msgstore.db.crypt12 file and only show an empty data. The results of the decryption will produce a file msgstore.db which has a size larger than msgstore.db.crypt12 as shown in Fig. 5. That file is a SQL database file, and it can be viewed using the SQLite Database Browser application, as shown in Fig. 6.



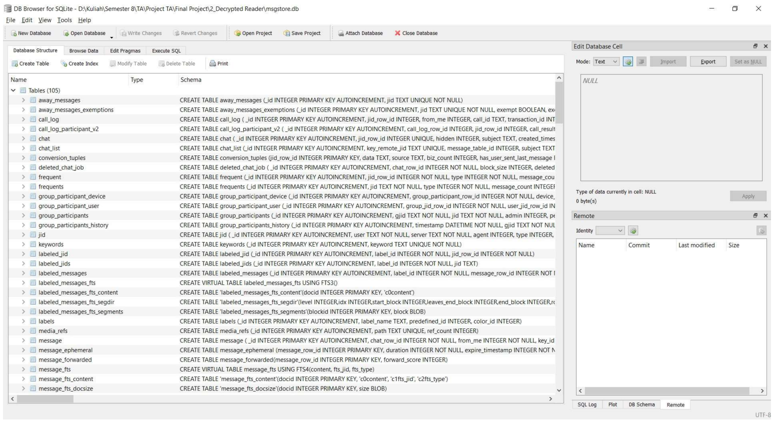
```
Microsoft Windows [Version 10.0.18363.815]
(c) 2019 Microsoft Corporation. All rights reserved.

D:\Kuliah\Semester 8\TA\Project TA\Final Project\1_Decrypter>java -jar decrypt12.jar key msgstore.db.crypt12
Decryption of crypt12 file was successful.
```

Fig. 4. Decrypt12 Command



| | | | |
|---|---|---|---|
| msgstore.db | 5/2/2020 12:11 AM | Data Base File | 6,768 KB |
| msgstore.db.crypt12 | 2/14/2020 10:21 AM | CRYPT12 File | 3,517 KB |

Fig. 5. Decrypted msgstore.db file



DB Browser for SQLite - D:\Kuliah\Semester 8\TA\Project TA\Final Project\2_Decrypted Reader\msgstore.db

| Name | Type | Schema |
|---|---|---|
| Tables (105) | | |
| away_messages | | CREATE TABLE away_messages (_id INTEGER PRIMARY KEY AUTOINCREMENT, jid TEXT UNIQUE NOT NULL) |
| away_messages_exemptions | | CREATE TABLE away_messages_exemptions (_id INTEGER PRIMARY KEY AUTOINCREMENT, jid TEXT UNIQUE NOT NULL, exempt BOOLEAN, ex |
| call_log | | CREATE TABLE call_log ( _id INTEGER PRIMARY KEY AUTOINCREMENT, jid_row_id INTEGER, from_me INTEGER, call_id TEXT, transaction_id INT |
| call_log_participant_v2 | | CREATE TABLE call_log_participant_v2 ( _id INTEGER PRIMARY KEY AUTOINCREMENT, call_log_row_id INTEGER, jid_row_id INTEGER, call_result |
| chat | | CREATE TABLE chat ( _id INTEGER PRIMARY KEY AUTOINCREMENT, jid_row_id INTEGER UNIQUE, hidden INTEGER, subject TEXT, created_times |
| chat_list | | CREATE TABLE chat_list (_id INTEGER PRIMARY KEY AUTOINCREMENT, key_remote_jid TEXT UNIQUE, message_table_id INTEGER, subject TEXT |
| conversion_tuples | | CREATE TABLE conversion_tuples (jid_row_id INTEGER PRIMARY KEY, data TEXT, source TEXT, biz_count INTEGER, has_user_sent_last_message I |
| deleted_chat_job | | CREATE TABLE deleted_chat_job ( _id INTEGER PRIMARY KEY AUTOINCREMENT, chat_row_id INTEGER NOT NULL, block_size INTEGER, deleted |
| frequent | | CREATE TABLE frequent (_id INTEGER PRIMARY KEY AUTOINCREMENT, jid_row_id INTEGER NOT NULL, type INTEGER NOT NULL, message_cou |
| frequents | | CREATE TABLE frequents (_id INTEGER PRIMARY KEY AUTOINCREMENT, jid TEXT NOT NULL, type INTEGER NOT NULL, message_count INTEGEI |
| group_participant_device | | CREATE TABLE group_participant_device (_id INTEGER PRIMARY KEY AUTOINCREMENT, group_participant_row_id INTEGER NOT NULL, device_ |
| group_participant_user | | CREATE TABLE group_participant_user (_id INTEGER PRIMARY KEY AUTOINCREMENT, group_jid_row_id INTEGER NOT NULL, user_jid_row_id IN |
| group_participants | | CREATE TABLE group_participants (_id INTEGER PRIMARY KEY AUTOINCREMENT, gjid TEXT NOT NULL, jid TEXT NOT NULL, admin INTEGER, pe |
| group_participants_history | | CREATE TABLE group_participants_history (_id INTEGER PRIMARY KEY AUTOINCREMENT, timestamp DATETIME NOT NULL, gjid TEXT NOT NUL |
| jid | | CREATE TABLE jid ( _id INTEGER PRIMARY KEY AUTOINCREMENT, user TEXT NOT NULL, server TEXT NOT NULL, agent INTEGER, type INTEGER, |
| keywords | | CREATE TABLE keywords (_id INTEGER PRIMARY KEY AUTOINCREMENT, keyword TEXT UNIQUE NOT NULL) |
| labeled_jid | | CREATE TABLE labeled_jid (_id INTEGER PRIMARY KEY AUTOINCREMENT, label_id INTEGER NOT NULL, jid_row_id INTEGER NOT NULL) |
| labeled_jids | | CREATE TABLE labeled_jids (_id INTEGER PRIMARY KEY AUTOINCREMENT, label_id INTEGER NOT NULL, jid TEXT) |
| labeled_messages | | CREATE TABLE labeled_messages (_id INTEGER PRIMARY KEY AUTOINCREMENT, label_id INTEGER NOT NULL, message_row_id INTEGER NOT I |
| labeled_messages_fts | | CREATE VIRTUAL TABLE labeled_messages_fts USING FTS3() |
| labeled_messages_fts_content | | CREATE TABLE 'labeled_messages_fts_content'(docid INTEGER PRIMARY KEY, 'c0content') |
| labeled_messages_fts_segdir | | CREATE TABLE 'labeled_messages_fts_segdir'(level INTEGER,idx INTEGER,start_block INTEGER,leaves_end_block INTEGER,end_block INTEGER,rc |
| labeled_messages_fts_segments | | CREATE TABLE 'labeled_messages_fts_segments'(blockid INTEGER PRIMARY KEY, block BLOB) |
| labels | | CREATE TABLE labels (_id INTEGER PRIMARY KEY AUTOINCREMENT, label_name TEXT, predefined_id INTEGER, color_id INTEGER) |
| media_refs | | CREATE TABLE media_refs (_id INTEGER PRIMARY KEY AUTOINCREMENT, path TEXT UNIQUE, ref_count INTEGER) |
| message | | CREATE TABLE message ( _id INTEGER PRIMARY KEY AUTOINCREMENT, chat_row_id INTEGER NOT NULL, from_me INTEGER NOT NULL, key_id |
| message_ephemeral | | CREATE TABLE message_ephemeral (message_row_id INTEGER PRIMARY KEY, duration INTEGER NOT NULL, expire_timestamp INTEGER NOT N |
| message_forwarded | | CREATE TABLE message_forwarded(message_row_id INTEGER PRIMARY KEY, forward_score INTEGER) |
| message_fts | | CREATE VIRTUAL TABLE message_fts USING FTS4(content, fts_jid, fts_type) |
| message_fts_content | | CREATE TABLE 'message_fts_content'(docid INTEGER PRIMARY KEY, 'c0content', 'c1fts_jid', 'c2fts_type') |
| message_fts_docsize | | CREATE TABLE 'message_fts_docsize'(docid INTEGER PRIMARY KEY, size BLOB) |

NULL

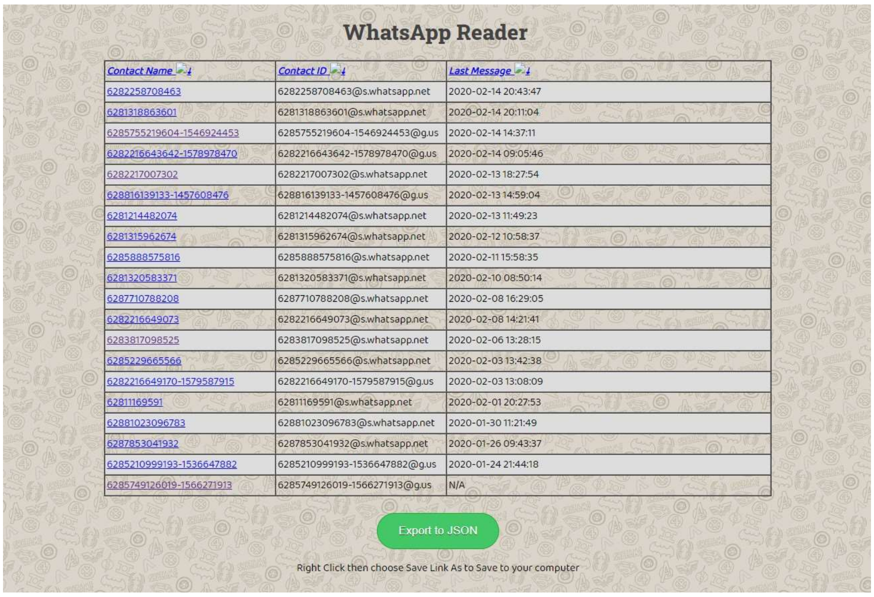
Type of data currently in cell: NULL
0 byte(s)

| Name | Commit | Last modified | Size |
|---|---|---|---|

Fig. 6. Whatsapp Chat Database

The database file contains data, including conversation, media, audio, and contact number. In order to filter and choose the intended data, such as specific chat data of someone and its date, a python-based application is created. This application can display data that has been filtered, then displayed in the form of a table like in Fig. 7.

In addition, to displaying and viewing the contents of conversation data, this application can convert each conversation, which will then be combined into a JSON file. The saved file will be in JSON size, depending on how many chats are to be converted. In this case, all chat data are converted, which results in a msgstore.db.json file that has a size of 203 KB.



WhatsApp Reader

| Contact Name | Contact ID | Last Message |
|---|---|---|
| 6282258708463 | 6282258708463@s.whatsapp.net | 2020-02-14 20:43:47 |
| 6281318863601 | 6281318863601@s.whatsapp.net | 2020-02-14 20:11:04 |
| 6285755219604-1546924453 | 6285755219604-1546924453@g.us | 2020-02-14 14:37:11 |
| 6282216643642-1578978470 | 6282216643642-1578978470@g.us | 2020-02-14 09:05:46 |
| 6282217007302 | 6282217007302@s.whatsapp.net | 2020-02-13 18:27:54 |
| 628816139133-1457608476 | 628816139133-1457608476@g.us | 2020-02-13 14:59:04 |
| 6281214482074 | 6281214482074@s.whatsapp.net | 2020-02-13 11:49:23 |
| 6281315962674 | 6281315962674@s.whatsapp.net | 2020-02-12 10:58:37 |
| 6285888575816 | 6285888575816@s.whatsapp.net | 2020-02-11 15:58:35 |
| 6281320583371 | 6281320583371@s.whatsapp.net | 2020-02-10 08:50:14 |
| 6287710788208 | 6287710788208@s.whatsapp.net | 2020-02-08 16:29:05 |
| 6282216649073 | 6282216649073@s.whatsapp.net | 2020-02-08 14:21:41 |
| 6283817098525 | 6283817098525@s.whatsapp.net | 2020-02-06 13:28:15 |
| 6285229665566 | 6285229665566@s.whatsapp.net | 2020-02-03 13:42:38 |
| 6282216649170-1579587915 | 6282216649170-1579587915@g.us | 2020-02-03 13:08:09 |
| 62811169591 | 62811169591@s.whatsapp.net | 2020-02-01 20:27:53 |
| 62881023096783 | 62881023096783@s.whatsapp.net | 2020-01-30 11:21:49 |
| 6287853041932 | 6287853041932@s.whatsapp.net | 2020-01-26 09:43:37 |
| 6285210999193-1536647882 | 6285210999193-1536647882@g.us | 2020-01-24 21:44:18 |
| 6285749126019-1566271913 | 6285749126019-1566271913@g.us | N/A |

Right Click then choose Save Link As to Save to your computer

Fig. 7. Chat Data on WhatsApp Reader

### C. Visualization using the Timeline Method

This research was conducted with the Social Network Analysis approach, where the visualization of WhatsApp chat data uses the timeline model. Data and information used in this research are based on a controlled experiment to produce WhatsApp conversation data conducted by the suspect. In Timeline visualization, data/events can be seen chronically and present the timespan when the chat is done. Visualizing an event with a timeline is simple; however, showing the relationship and the detail between some events is quite challenging (Nguyen, Xu, Walker, & Wong, 2014).

As we conducted a controlled experiment, this research may have a fictitious case of terrorism where conversations are made on WhatsApp. This case is not related to the real case, including the name, time, place, and any other related information. In this case, the owner of the device has a plan to initiate the terrorist operation and making a conversation with the funder, the courier, the maker, and the person who to execute the crime. But, before the operation was successfully executed, the owner of the phone was arrested and will be judged at the court. The forensic team found evidence of the WhatsApp conversation, but it was not clear what, who, when, and how the activity was carried out because there were many other chats. Therefore, we create a web-based application called WhatsApp Chat Visualizer to visualize the WhatsApp chat data into a timeline graph that has a time span and a conversation period over time with a search feature and displayed live chat data, not just a graph displays.

## IV. RESULTS AND DISCUSSION

This research converts the JSON file to the form of a timeline graph, as can be seen in Fig. 8 to support material evidence and to facilitate understanding of people who do not have a technical background in digital forensics. Data is represented by a timeline graph to display the contact name and timeline of a person's conversation with the suspect. Fig. 9 shows the detail of the conversation contents. It will be displayed by clicking on one of the timelines of a person's contact name.

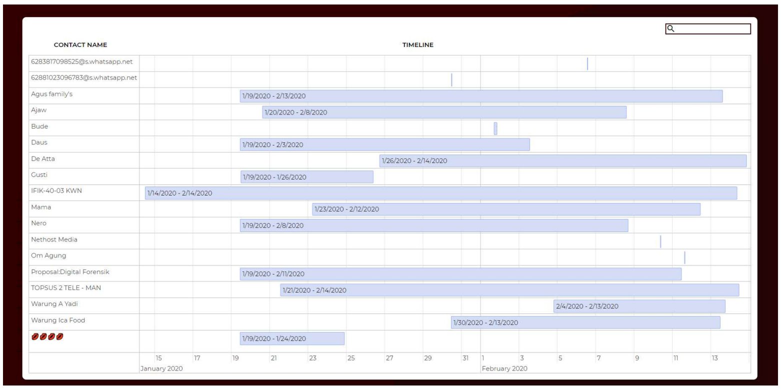
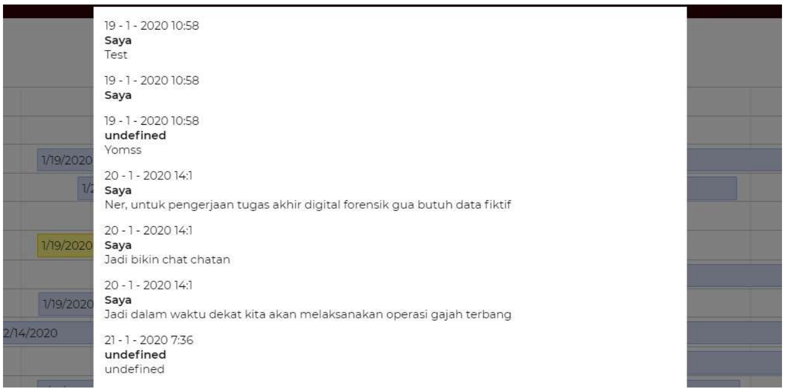


Fig. 8. Timeline Graph of Whatsapp Chat Data



19 - 1 - 2020 10:58
**Saya**
Test

19 - 1 - 2020 10:58
**Saya**

19 - 1 - 2020 10:58
**undefined**
Yomss

20 - 1 - 2020 14:1
**Saya**
Ner, untuk pengerjaan tugas akhir digital forensik gua butuh data fiktif

20 - 1 - 2020 14:1
**Saya**
Jadi bikin chat chatan

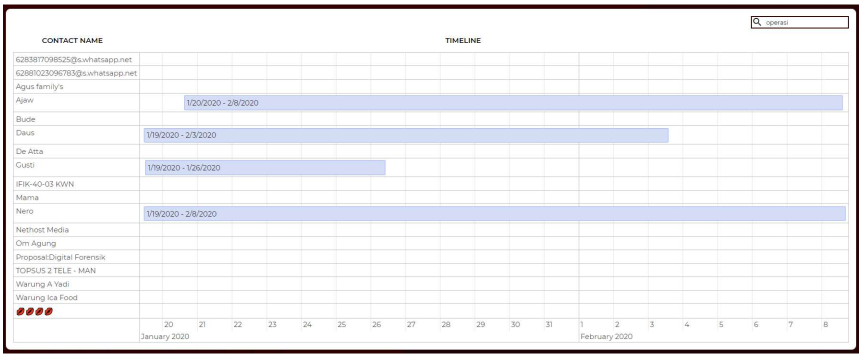
20 - 1 - 2020 14:1
**Saya**
Jadi dalam waktu dekat kita akan melaksanakan operasi gajah terbang

21 - 1 - 2020 7:36
**undefined**
undefined

Fig. 9. Detail of the chat showed in WhatsApp Reader

While the timeline visualization in Fig. 8 depicts the entire WhatsApp conversation, we still need to indicate specific conversation that is related to a crime. To get a timeline visualization of this specific data, we can filter it by enabling keywords or unique words based on the signs of terrorist activity. These keywords might be supplied by the authority that handle the case.

The filtering results will display certain visualized conversation data on the timeline. Data conversations that appear are conversations between the owner of the cell phone with other parties that have the match keywords. Fig. 10 shows a visualization of the timeline using the keyword "operasi" and presents four people who are related to the mobile owner.



| CONTACT NAME | TIMELINE |
|---|---|
| 6283817098525@s.whatsapp.net | |
| 62881023096783@s.whatsapp.net | |
| Agus family's | |
| Ajaw | 1/20/2020 - 2/8/2020 |
| Bude | |
| Daus | 1/19/2020 - 2/3/2020 |
| De Atta | |
| Gusti | 1/19/2020 - 1/26/2020 |
| IFIK-40-03 KWN | |
| Mama | |
| Nero | 1/19/2020 - 2/8/2020 |
| Nethost Media | |
| Om Agung | |
| Proposal:Digital Forensik | |
| TOPSUS 2 TELE - MAN | |
| Warung A Yadi | |
| Warung Ica Food | |

Fig. 10. Filtered contact name using a keyword

As seen in the visualization in Fig. 10, the overall information from the chat database, which is one of the WhatsApp Artifact, can be visualized in the form of a Timeline that shows the time span of the conversation for each conversation and can then be used to answer investigation questions such as presented in Table III. The answer itself is based on the fictitious case mentioned in Section III and the result of the filtered contact name in Fig. 10.

TABLE III
THE ANSWER TO THE DIGITAL FORENSIC SOP QUESTION

| Question | Answer |
|---|---|
| What type of crime is it? | Preparation of terrorist operation |
| Who are the people involve | Shidek, Daus, Ajaw, Nero, and Gusti |
| Where is the location of the crime | Indonesia |
| When did the crime happen | Between 2020-01-19 10:58 until 2020-02-08 16:29 |
| How did the crime happen | Shidek makes conversations with several people on WhatsApp Messenger to plan a terrorist operation |

This visualization may facilitate us to understand the crime events and their related data. We can prove that using the timeline visualization method can help in conducting an investigative analysis of chat database artifact on WhatsApp Messenger. By using the WhatsApp Chat Visualizer application, it will get information quickly. The search is based on several keywords related to existing case studies. Information about the content of the conversation and timestamp can be displayed with the timeline, making it easier to trace the information needed.

## V. CONCLUSION

Conversation data can be visualized in a timeline form. In this research, we show how visualization may help the forensic analysis process on WhatsApp's artifacts by creating an application called WhatsApp Chat Visualizer. This application is based on the Timeline method to delineate the event and relation between data conversation and contacts. By incorporating a filter using specific keywords, it is also shown that we can indicate related data and its timespan, specific to the case under investigation. As the extracted WhatsApp database is encrypted, we also show how to prepare the decryption in order to read the data. This application provides a quick and easy way to analyze the interrelation of other parties in the alleged visualization of crime in the form of a timeline. Future research is expected to be developed using another visualization method to increase data comprehension in determining the alleged crime in conversation data on WhatsApp Messenger.

## REFERENCES

Allen, R. B. (1995). Interactive timelines as information system interfaces. In *Symposium on digital libraries* (Vol. 175, p. 180).

Anglano, C. (2014). Forensic analysis of whatsapp messenger on android smartphones. *Digital Investigation*, *11*(3), 201–213.

Anwar, N., & Riadi, I. (2017). Analisis investigasi forensik whatsapp messenger smartphone terhadap whatsapp berbasis web. *Jurnal Ilmu Teknik Elektro Komputer dan Informatika (JITEKI)*, *3*(1), 1–10.

Casey, E. (2011). *Digital evidence and computer crime: Forensic science, computers, and the internet.* Academic press.

Clement, J. (2020, Apr). *Number of monthly active whatsapp users as of 2013-2020.* statista.com. Retrieved from `https://www.statista.com/statistics/260819/ number-of-monthly-active-whatsapp-users/`

Cortjens, D., Spruyt, A., & Wieringa, W. (2012). *Whatsapp database encryption project report* (Tech. Rep.). Technical report, 2011. Available at https://www. os3. nl/media/2011-2012.

Google. (2020, Jan). *Timelines | charts | google developers.* Author. Retrieved from `https:// developers.google.com/chart/interactive/docs/gallery/timeline`

Hariyadi, D., Winarno, W. W., & Luthfi, A. (2016). Analisis konten dugaan tindak kejahatan dengan barang bukti digital blackberry messenger. *Teknomatika STMIK Jenderal Achmad Yani Yogyakarta*, *9*(1), 81–89.

Koum, J., & Acton, B. (2016, Apr). *Whatsapp.* Retrieved from `https://blog.whatsapp.com/ end-to-end-encryption`

Nguyen, P. H., Xu, K., Walker, R., & Wong, B. W. (2014). Schemaline: Timeline visualization for sensemaking. In *2014 18th international conference on information visualisation* (pp. 225–233).

Nguyen, P. H., Xu, K., Walker, R., & Wong, B. W. (2016). Timesets: Timeline visualization with set relations. *Information Visualization*, *15*(3), 253–269.

Olsson, J., & Boldt, M. (2009). Computer forensic timeline visualization tool. *digital investigation*, *6*, S78–S87.

Pavlik, M., & MacIntoch, S. (n.d.). John, & shawn.(2015). *Converging Media*, 189.

Plaisant, C., Milash, B., Rose, A., Widoff, S., & Shneiderman, B. (1996). Lifelines: visualizing personal histories. In *Proceedings of the sigchi conference on human factors in computing systems* (pp. 221–227).

Pomalingo, S., Sugiantoro, B., & Prayudi, Y. (2019). Data visualisasi sebagai pendukung investigasi media sosial. *ILKOM Jurnal Ilmiah*, *11*(2), 143–151.

Sai, D. M., Prasad, N., & Dekka, S. (2015). The forensic process analysis of mobile device. *Int. J. Comput. Sci. Inf. Technol*, *6*(5), 4847–4850.

Stab, C., Nazemi, K., & Fellner, D. W. (2010). Sematime-timeline visualization of time-dependent relations and semantics. In *International symposium on visual computing* (pp. 514–523).

Tassone, C., Martini, B., & Choo, K.-K. (2017). Forensic visualization: survey and future research directions. In *Contemporary digital forensic investigations of cloud and mobile applications* (pp. 163–184). Elsevier.

Tassone, C. F., Martini, B., & Choo, K.-K. R. (2017). Visualizing digital forensic datasets: a proof of concept. *Journal of forensic sciences*, *62*(5), 1197–1204.

Thakur, N. S. (2013). Forensic analysis of whatsapp on android smartphones.

Umar, R., Riadi, I., Zamroni, G. M., et al. (2018). Mobile forensic tools evaluation for digital crime investigation. *Int. J. Adv. Sci. Eng. Inf. Technol*, *8*(3), 949.

Wang, Z., & Yuan, X. (2014). Urban trajectory timeline visualization. In *2014 international conference on big data and smart computing (bigcomp)* (pp. 13–18).

Yadav, S., Prakash, S., Dayal, N., & Singh, V. (2020). Forensics analysis of whatsapp in android mobile phone. *Available at SSRN 3576379.*