Chapter 12
Timeline Analysis

Learning Objectives
The objectives of this chapter are to:

- Understand the fundamentals of timeline analysis and its processes
- Become familiar with popular tools to perform timeline analysis
- Be able to analyze filesystem timeline using Autopsy

Digital forensics requires applying computer science to answer investigative questions, such as when a digital artifact occurs. It is very helpful, therefore to arrange events on a computer system chronologically so as to tell what happened to an incident occurred [1]. It is referred to as timeline analysis. Timeline analysis is the process of analyzing event data to determine when and what has occurred on a computer system for forensic purposes. In this chapter, we'll learn fundamentals of timeline analysis. The standard process for filesystem timeline analysis will be introduced. Also, you will become familiar with popular tools to perform timeline analysis.

12.1 Principle of Timeline Analysis

12.1.1 Timeline

Due to the rapid development of digital technologies and their pervasiveness in everyday life, more computer forensics techniques are required to answer questions related to an investigation from different aspects. One question commonly asked in digital forensics is "what happened and when?". One of the simplest ways to answer

X. Lin, *Introductory Computer Forensics*,
https://doi.org/10.1007/978-3-030-00581-8_12

Fig. 12.1 A timeline is an approach for representing a set of events in sequential arrangement

this question is to organize events chronologically. The sequence ordered by time is called a timeline. In other words, a timeline is an approach for representing a set of events in sequential arrangement [2], sometimes described as a project artifact as shown in Fig. 12.1. Timelines can use any time scale, depending on the subject and data.

Obviously, we must first extract timestamps associated with events to create a timeline. The most common timestamp sources are file system times [2]. MAC times are pieces of filesystem metadata used to record events and event times regarding a particular file. MAC times refer to three types of time that metadata records; M—modification time, A—access time, and C—created time. Thus, the creation of timeline can be seen as a process of iterating over the files and their metadata and outputting a chronologically ordered event sequence, for example, showing when a file had been created, modified and accessed or deleted. This kind of timeline allows us to have a global overview of the events occurring before, during and after a given incident. However, events occurring on a certain file system can be enormous and complicated to analyze. The complexity of events makes the interpretation of the timeline difficult and therefore decision making can be erroneous. Thus, timeline must be intuitive in how it organizes relevant information in a very convenient way so that better assisting in forensic investigations.

Traditionally, we collect larger archives of data; later simplifying their representation to assist decision making. Digging deeper in a forensic volume we aim to collect as much related information as possible from artifacts or data logs. These are used to create larger more complex time lines. We call these super time lines. Super time lines will be discussed later.

12.1.2 Timeline Event

Timeline analysis is any investigation processes that involve timeline data. In other words, timeline data is used when any time-related investigative questions need to be answered, for example, when was a file deleted?, when did a user visit a website?, or when was the last time that a user logged into your system? Prior to conducting your own timeline analysis it would be prudent to explore the components of timeline data. A primary component of timeline data is events; these are often used as the primary data sources for timeline analysis. These are collected by timelines along with their associated timestamps. Events can be classified into three categories:

- Filesystems
- Web activity
- Miscellaneous

Timeline data is significant for a forensic investigation. This can shed light on when files or resources were created, accessed or deleted. Chronological evidence may expose whether a perpetrator had sufficient exposure to any resources to commit an offense. Now, we will explore each of events categories in further detail.

12.1.2.1 Filesystems

A filesystem controls how the data is stored and retrieved in computer systems. A filesystem organizes the files and keeps track of them on disk or in partitions. If partitions or disks are used as file systems, the disk or partition should be initialized before usage. This process is called formatting or making a filesystem, which writes data structures to the disk [3].

Filesystems determine the structure of data on disk. These data structures differ among various filesystems. Despite differences, timeline databases collect information of important events. These include file modified events, file accessed events and file created events. File created events occur when an instance of a file has been created. Data on file creation events cannot be changed unless modified through third party software. File modified events occur when an instance of file is written or modified. Renaming a file does not change the modification timestamp. File Accessed events occur when files are read or overwritten. Most timeline analysis tools can extract the events mentioned above.

12.1.2.2 Web Activity

Web Activity is a broad categorization of internet browsing activity. This includes but is not limited internet web page browse actions, downloads, cookies, bookmarks, history and searches. Download events occur when users download files from remote servers. Cookies save events occur when the user is logging into web systems. Bookmark events occur when the user saves pages to the bookmark. History events occur when the user visits pages and search events occur when the user uses address bar for searching.

12.1.2.3 Miscellaneous

There are a number events vital to digital forensics which cannot be fitted into the categories above. These are usually labeled as miscellaneous. These are often e-mail events, recent file events, installed program, devices attached, and so on. These activities are usually trivial but vital in timeline investigation and analysis.

From this section we have a brief understanding of the data sources and some aspects of timeline analysis. However, collecting event data is only the first step in timeline analysis. In the following section we introduce more definitions and timeline analysis tools by discussing timeline definition.

12.2 Timeline Analysis Process

Timeline analysis is the process of collecting and analyzing event data to determine when and what has occurred on a filesystem for forensic purposes. Results are organized chronologically to illustrate a chain of events in a concise manner. Timeline analyses generally comprise of two stages.

- First is time and event collection (or timeline creation), where information on events and their associated times are collected from numerous sources and organized into a database.
- The second is organization and analysis (or timeline analysis), where information is sorted and filtered based on the requirements of the investigation and represented in a manageable fashion.

Data sources may include system logs, MAC times, firewall logs, and application data.

12.2.1 Timeline Creation

As previously discussed, we ask “what happened and when?” We conjecture that a timeline is consisted of two parts including time data and event data. Examples of timestamps sources are event logs, registry files, internet history, email files, recycle bin, thumbs.db file, chat logs, restore points, capture files, and archive files. For a better demonstration we discuss the timeline in TSK (The Sleuth Kit®). In TSK the software uses time stamps to store the time data including atime, mtime, ctime and the crtime.

- Atime (Access time) stamp contains data of last access time to a file.
- Mtime (Modification time) stamp contains data of last modification time to a file.
- Ctime stamp represents different meaning in different file system. In NTFS it means the last time the MFT was modified. In EXT3 it means inode changes. In FAT the time stamp means the time when the file is created.
- Crtime stamp means the time a file was created in NTFS and it is used for file deletion in EXT3 and not used in FAT.

From the time stamp we know the milestones on the timeline. This solves the problem of "When" in the timeline analysis. Events occurring at certain instances solve the issue of "What". The event context contains the data to describe the events related to a certain time stamp.

12.2.2 Timeline Analysis

As mentioned before, one of the solutions to the shortcomings of traditional timeline analysis is expanding it with information from multiple sources to get a better picture of the events. This is called super timeline. Having analyzed a timeline, there should be criteria to constrain the quality of the timeline analysis. There are three factors which can determine the quality of timeline analysis. They are size or volume of collected information, data representation and time to perform analysis.

In the first step of timeline analysis, investigator should prepare the data source for investigation. It is usually an image or a copy of hard disk. It should be imported into the investigation software. Tools usually provide an overall investigation of the timeline. Analysis usually focuses on a small fraction of time. So investigators should zoom the timeline for further investigation. There are three kinds of zooming in timeline analysis. They are temporal zooming, event type zooming, and description level zooming. Temporal zooming are techniques to investigate the timeline in different time scale. By zooming the time scale investigator not only investigate the timeline by hour but also by seconds. Event type zooming enables investigator to inspects the timeline by events classification. Description level zooming can provide the content data inside the timeline at different levels of description. After zooming to a proper scale investigator need to filter the targeted events for investigation. Filter aims to reduce the volume of data and to hide uninteresting events. An investigator uses timeline context analysis or other analysis methods to reach their target. We then conclude the investigation and visualizing our data for further analysis. The process of standard timeline analysis is shown in Fig. 12.2.

Until now, we introduced a standard process of timeline analysis. The process of timeline analysis can be different based on demands and the target of the investigation. The standard process can satisfy common situations in timeline analysis.

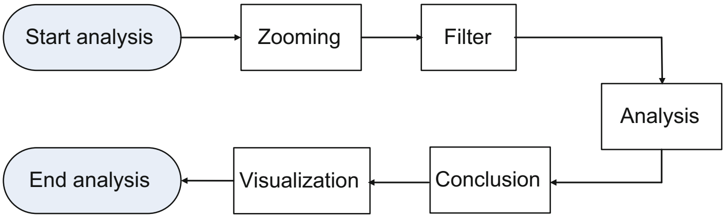


Fig. 12.2 Standard timeline analysis process

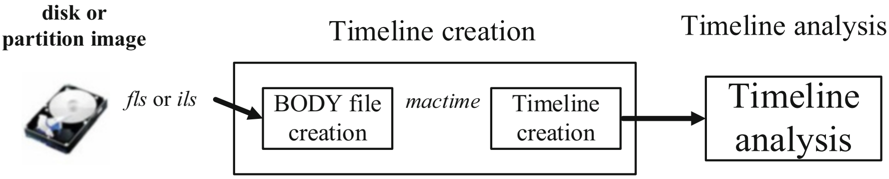


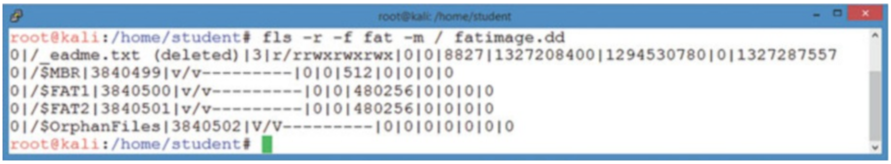
Fig. 12.3 MAC timeline analysis procedure

12.2.3 MAC Timeline Creation and Analysis with TSK

The most popular data sources for timeline analysis are MAC times. In this section, we will provide a simple case study to show how to build MAC timelines from filesystem metadata for a given system image, particularly the only partition formatted with FAT file system within the disk image "thumbimage_fat.dd" provided in the book. As mentioned before timeline analysis forensics is important to ease investigation and make the examiners to get the big picture of what exactly happened in chronological order. MAC times are pieces of filesystem metadata used to record events and event times regarding a particular file. The procedure for MAC timeline analysis consists of timeline creation then timeline analysis, shown in Fig. 12.3. In MAC timeline creation, we first extract information from unallocated inodes and unallocated directory entries, whereas MAC Timeline analysis examines file event and time data to reconstruct the events which have occurred on a system. Timeline Creation phase consists of two stages [7]:

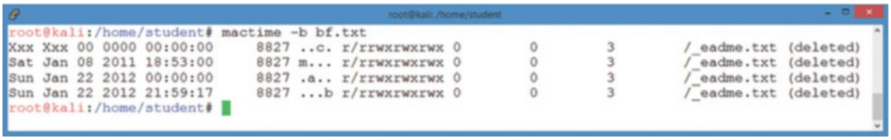
Stage 1—BODY file creation: BODY file is an intermediate file when creating a timeline of file activity. It is a pipe ("|") delimited text file that contains one line for each file (or other even type, such as a log or registry key), shown in Fig. 12.4. For example, the TSK tools *fls* and *ils* all output this data format [4]. Each line of ouput has the following format: "MD5|name|inode|mode_as_string|UID|GID|size|atime|mtime|ctime|crtime". In Fig. 12.4, we can clearly see that the UNIX epoch is used for times. The UNIX epoch, also known as Unix timestamps, stands for the number of seconds from January 1, 1970, but it's not user friendly.

Stage 2—Timeline creation: It runs the TSK tool *mactime* to turn the body file into something a bit more user friendly. The mactime tool reads this file and sorts the



```
root@kali:/home/student# fls -r -f fat -m / fatimage.dd
0|/_eadme.txt (deleted)|3|r/rrwxrwxrwx|0|0|8827|1327208400|1294530780|0|1327287557
0|/$MBR|3840499|v/v---------|0|0|512|0|0|0|0
0|/$FAT1|3840500|v/v---------|0|0|480256|0|0|0|0
0|/$FAT2|3840501|v/v---------|0|0|480256|0|0|0|0
0|/$OrphanFiles|3840502|V/V---------|0|0|0|0|0|0|0
root@kali:/home/student#
```

Fig. 12.4 Example body file



```
root@kali:/home/student# mactime -b bf.txt
Xxx Xxx 00 0000 00:00:00     8827 ..c. r/rrwxrwxrwx 0        0        3         /_eadme.txt (deleted)
Sat Jan 08 2011 18:53:00     8827 m... r/rrwxrwxrwx 0        0        3         /_eadme.txt (deleted)
Sun Jan 22 2012 00:00:00     8827 .a.. r/rrwxrwxrwx 0        0        3         /_eadme.txt (deleted)
Sun Jan 22 2012 21:59:17     8827 ...b r/rrwxrwxrwx 0        0        3         /_eadme.txt (deleted)
root@kali:/home/student#
```

Fig. 12.5 Example timeline

contents (therefore the format is sometimes referred to as the "mactime format"). Specifically, these epoch timestamps will be converted to more human readable dates, shown in Fig. 12.5. Most significant, we start to see meaningful events (file deletion) are associated with timestamps.

Timeline analyses are useful in how we represent IT forensic evidence. Once relevant evidence has been extracted they can be used to acquit or indict an accused based on results. With the dynamic and unstable nature of file systems, timeline analysis provide a perspective into what was done on a system, when and for how long.

Now, we show how to build MAC timelines from filesystem metadata for the given FAT system image from the disk image "thumbimage_fat.dd" provided in the book. We assume the image named "fatimage.dd" is the extracted FAT file system image. First, we create some events by mounting the extracted FAT file system onto Forensics Workstation and then deleting the "readme.txt" file. Afterwards, we unmount the file system.

```
# mount -o rw fatimage.dd /mnt/forensics/
# cd /mnt/forensics/
# rm -f readme.txt
# cd ..
# umount /mnt/forensics
```

Note that after you delete the readme.txt file, you must move out of the folder of /mnt/forensics before unmounting the file system.

Now we can create the body file from the extracted FAT file system by using the TSK tool fls, shown in Fig. 12.4

For the purpose of further investigation, we save the body file into a text file named bf.txt by running the TSK tool *fls* using output redirection.

```
# fls -r -f fat -m / fatimage.dd > bf.txt
```

Afterwards, we can create timelines by using the TSK tool *mactime*. Specifically, we turn the body file into something a bit more user friendly, shown in Fig. 12.5.

Until now, we can clearly see that there is a deleted file in the FAT file system, which is obvious since we just deleted it in this example. We also see a list of actions happened to the file along with when they occurred.

12.3 Forensic Timeline Analysis Tools

In forensic analysis, timeline information will prove crucial, these include point of creation, modification, access and deletion. **Timeline Analysis** is a process of collating extracted data, using time-stamps from the file system and other sources such as log files and internal file meta-data. In simpler words timeline analysis is designed to recover all the chronological events which occurred on the disk. Some tools have been developed to perform automated timeline analysis such as Leading Forensic Analysis Tools, Log2timeline, SIMILE Visual Timeline, EnCase, and Forensic Tool Kit (FTK). These tools have their own strengths and shortcomings.

Regardless of differences between tools, all are designed to achieve efficient timeline analysis. The criteria of evaluating the quality of timeline analysis is to focus on two aspects. These are investigation time and data volume reduction. One goal of timeline analysis is to reduce investigation time and to collect a smaller volume of data set to inspect. Although investigation time and data set volume are very useful in timeline analysis evaluation, investigators should make intuitive plan based on their investigative requirements. Timeline creation and analysis should be proposed to meet the forensic investigation requirements.

To achieve the goals of better quality in timeline analysis, many tools have been developed to solve problems. We can categorize these tools to the following categories [2]. Then we will provide a brief description of number of them:

- Timelines based on file system times—e.g. EnCase, Sleuth Kit
- Timelines including times from inside files—e.g. Cyber Forensic Time Lab (CFTL), Log2timeline
- Visualizations—e.g. EnCase, Zeitline, Aftertime

12.3.1 Log2timeline

Log2timeline [5] is timeline analysis tool developed by Kristenn Gudjonsson. This tool is open source and developed using Perl language. This framework is designed to parse events from log files of suspect systems and create an output file that used to produce a timeline. The output file contains time and date information for files in the system with csv format.

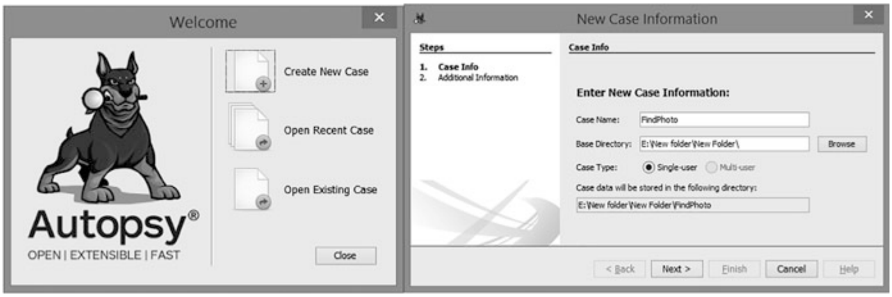
12.3.2 EnCase

EnCase [6] is a commercial framework provide many features for forensics investigations. Guidance Software company developed different versions of this product. One of these features by EnCase is timeline creation which is a useful tool. Using Encase, examiner can parse event logs and export it to csv output file. Encase parser can perform customized commands to extract the needed data.

12.4 Case Study

Time line analysis is widely applied in crime investigation and computer forensics. So we assume a real world problem as the lab exercise. We assume that Bob is a photographer and he is also one of your best friend. He takes about 600 wedding photos for a few couples in 1 day. Normally not all the photos will be presented to his customers. First he should select around 100 satisfactory photos to an independent folder for customer's selection. Then the photographer select and fix 12 photos as reference and recommendation to its users. After the photograph works Bob divided his work into three folder and save the files into USB flash memory. The ALL PHOTO folder contains all photos and will be submitted to his company as a backup. The SELECTED folder contains the photos for user selection. The Demo folder contains the 12 photo for demo. After doing the job, Bob leaves his laptop in his bedroom and his young son deletes all photos in the demo folder accidentally. Finding photos from the hundreds of photos are time consuming. So Bob asks you to help him to locate the deleted files in a short time. We can use time line analysis to investigate the events in file system.

To investigate the file system in the USB flash memory, we first insert the flash memory into the computer. Then we open the Autopsy software and create a new investigation case as Fig. 12.6.



Enter New Case Information:

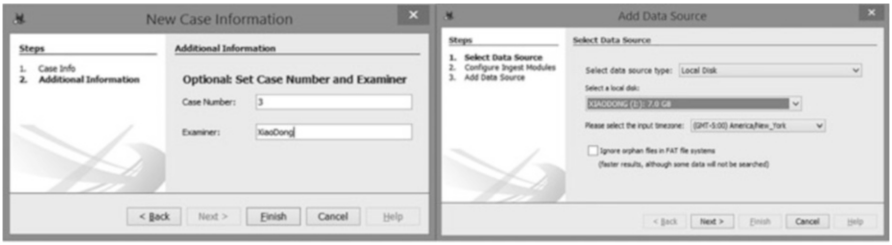
Case Name: FindPhoto

Base Directory: E:\New folder\New Folder\

Case Type: ● Single-user ○ Multi-user

Case data will be stored in the following directory:
E:\New folder\New Folder\FindPhoto

Fig. 12.6 Creation of new case



Optional: Set Case Number and Examiner

Case Number: 3

Examiner: XiaoDong

Select data source type: Local Disk

Select a local disk: XIAODONG (E:): 7.0 GB

Please select the input timezone: (GMT-5:00) America/New_York

□ Ignore orphan files in FAT file systems
(faster results, although some data will not be searched)

Fig. 12.7 Add data source to the autopsy

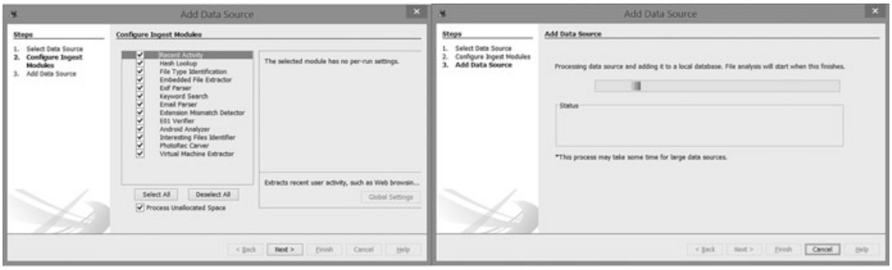
Then we give the investigation case a name and click the next button. Then we input the case number and case examiner. Then we select the local disk and target USB flash disk as data source. In this section we use the standard process of timeline analysis. The process in Figs. 12.6 and 12.7 are preparation for the timeline analysis.

After we select the source from the USB flash disk, we configure the ingest modules to determine the coverage of the investigation. In our lab we choose the default configuration (Fig. 12.8).

After initial preparation we should have a visualized timeline result. Then we should zoom it into a proper scale as to determine critical events. We will then filter the file modified events and analyze the scaled timeline. Finally we can determine which files are related to the file modified events as shown in Fig. 12.9. From this figure we can see the file modified events and then select the files we want. This problem was solved by using timeline analysis.

Review Questions

1. Describe in your own words, what is timeline analysis?
2. List three categories of timeline events and then describe each of them?
3. List five examples of timestamp sources.
4. What is MAC Timeline Analysis Process?



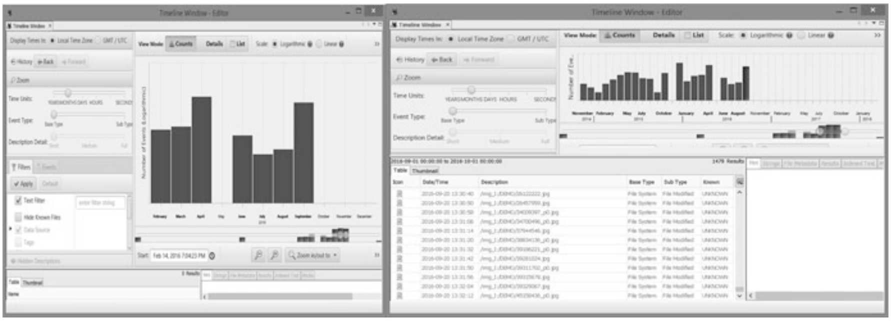
The selected module has no per-run settings.

Extracts recent user activity, such as Web browsin...

Processing data source and adding it to a local database. File analysis will start when this finishes.

*This process may take some time for large data sources.

Fig. 12.8 Ingest configuration and timeline generation



Timeline Window - Editor
Display Times In: ● Local Time Zone ○ GMT / UTC
History
Back
Forward
Zoom
Time Units:
Event Type:
Description Detail:
Filters
Text Filter
Hide Known Files
Data Source
Tags
View Mode: Counts Details List
Scale: ● Logarithmic ○ Linear
Number of Events (Logarithmic)
Timeline Window - Editor
Table Thumbnail
Icon Date/Time Description Base Type Sub Type Known

Fig. 12.9 Timeline analysis processes

12.5 Practice Exercise

The objective of this exercise is to build MAC timelines from filesystem metadata.

12.5.1 Setting Up the Exercise Environment

For this exercise, you need to prepare a custom file system image

- Create a 100 MB file with random values

```
# dcfldd if=/dev/zero bs=1M count=100 of=fat.dd
```

- Format the file with the FAT32 file system

```
# mkfs.vfat -F 32 fat.dd
```

where “fat.dd” is now a FAT32 file system. Note that you can use the TSK tool *fsstat* to check with it. For example,

```
# fsstat -f fat fat.dd
```

- Mount the file system as read/write

```
# mount -o loop,rw fat.dd /mnt/forensics
```

- Create two 1 KB random files in it

```
# cd /mnt/forensics
# dcfldd if=/dev/urandom of=file1.dat bs=512 count=2
# dcfldd if=/dev/urandom of=file2.dat bs=512 count=2
```

Note that it is highly recommended that you introduce a delay when you create the second file to make this experiment more visible.

- Delete two files you created

```
# rm -f file 1.dat
# rm -f file 2.dat
```

Note that it is highly recommended that you introduce a delay when you delete the second file to make this experiment more visible.

- Unmount the file system

```
# cd ..
# umount /mnt/forensics
```

12.5.2 Exercises

In this exercise, you are required to build MAC timelines from the FAT file system you have created, and answer the following questions:

Table 12.1 MAC meaning by FAT file system

m	a	c	b
Written	Accessed	Not applicable	Created

Q1. How many files are found in the image?
Q2. Which file is created first, “file 1.dat” or “file 2.dat”?
Q3. What is the exact date and time when “file 1.dat” is created?

Note that you can find out the MAC meaning by FAT file system in Table 12.1.

References

1. Derek Edwards. Computer Forensic Timeline Analysis with Tapestry. https://www.sans.org/reading-room/whitepapers/tools/computer-forensic-timeline-analysis-tapestry-33836
2. Hargreaves, C., & Patterson, J. (2012). An automated timeline reconstruction approach for digital forensic investigations. Digital Investigation, 9, S69-S79
3. https://en.wikipedia.org/wiki/File_system
4. https://wiki.sleuthkit.org/index.php?title=Body_file
5. https://github.com/log2timeline/plaso
6. https://www.guidancesoftware.com/encase-forensic
7. Timeline Analysis Part I: Creating a Timeline of a Live Windows System http://thedigitalstandard.blogspot.com/2010/03/creating-timeline-of-live-windows.html