Digital Forensics (IFS4102) Lab 8: Timeline Analysis & Incident Response

Lab Objectives

In this lab, you will perform **timeline analysis** as well as some basic **incident response tasks**. More specifically, you want:

1. To use Autopsy's **Plaso ingest module** and **Timeline** feature.
2. *(Optional)* To conduct a timeline analysis on time-containing artefacts and a disk image file using **Log2timeline/Plaso** + **Timeline Explorer**.
3. To run various Windows' **`wmic`** commands in a live analysis.
4. *(Optional)* To use process-monitoring tools in Windows, including **Process Explorer**, **Process Hacker**, and **Autoruns**
5. To use **KAPE** for incident response's evidence extraction and parsing.

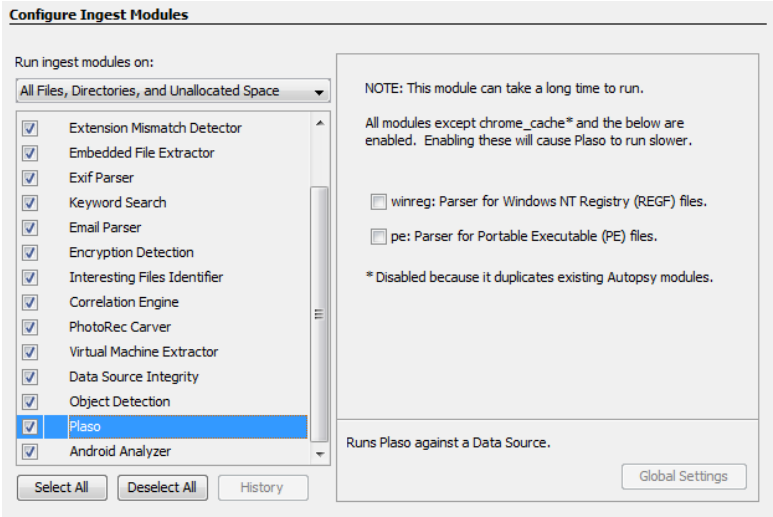
Task 1 (Win-FWS): Using Autopsy's Plaso Ingest Module and Timeline Feature

Notes:

- In this task, you can use Autopsy to run its **Plaso ingest module** and then its **Timeline** feature.
- If you are curious how Plaso really works, you can subsequently try performing optional **Task 2 below**. In that task, you use Timeline Explorer, which allows for ***manual inspection*** of all the extracted timestamp entries.
- You can use the previously shared "SuspectDrive1.E01" disk image downloadable from https://drive.google.com/file/d/1SiJOaWEmKYyXXKI-PEJP3-I2j1iOC_p3/view?usp=sharing.

Steps:

1. Run Autopsy, then add your disk image file as a data source.
2. To get the most out of the Autopsy's **Timeline**, you'll want to select **most/all** ingest modules, or **at least**: the hash lookup module (and use the NSRL to ignore known files), the recent activity module, the picture analyzer module, and other ingest modules that apply to your data (e.g. the email parser module if you have email data).
3. Enable the **Plaso ingest module** as shown below. In its configuration, you can use to enable or disable the `winreg` and `pe` modules which can take a long time to run.



4. Note that the Plaso events will be shown in the Autopsy's Timeline, and are *not* displayed in the Autopsy's tree viewer. Hence, run the **Autopsy's Timeline** by first selecting the "Tool" menu, then selecting the "Timeline" menu item.
5. To use the **Timeline's UI controls**, please follow the steps shown in the "*Autopsy's Timeline Analysis Tutorial*" tutorial video from Brian Carrier's Basis Technology: https://www.sleuthkit.org/autopsy/timeline.php. Note, however, that this tutorial video is based on Autopsy 3, which still has no "List" view. Nevertheless, the other UI panes and controls in Autopsy 3's Timeline run just like in Autopsy 4's Timeline.

[Optional] Task 2 (Lin-FWS & Win-FWS): Performing Super Timeline Analysis using Log2timeline/Plaso & Timeline Explorer

Notes:

- In this exercise, you will ***automatically extract*** timestamps from multiple time-containing artefact files using **Log2timeline/Plaso** (https://github.com/log2timeline/plaso) on your Linux forensics workstation, then export the resulting data into an **Excel file** for inspection & analysis.

- You can use your **Linux-based forensic workstation**.
  The **SANS Investigative Forensic Toolkit (SIFT) workstation** already has Log2timeline/Plaso installed. As explained in your Lab 1, you can download the SIFT VM appliance (.ova) file from https://www.sans.org/tools/sift-workstation/. Alternatively, you can install Log2timeline/Plaso on your **Linux, e.g. Ubuntu, machine** by following the steps given at: https://plaso.readthedocs.io/en/latest/sources/user/Users-Guide.html#installing-the-packaged-release.

- You will additionally export the resulting data into a ***CSV file*** for access by **Timeline Explorer** (https://binaryforay.blogspot.com/2017/04/introducing-timeline-explorer-v0400.html) on your Windows forensics workstation.

- For a sample time-containing artefact file, you can use a **Windows event log file** from Lab 5, `Security.evtx`, which can be downloaded from: https://drive.google.com/file/d/1Q4wR_cae08I0PdVSu97w5aZQI1rgYIAl/view?usp=sharing. Its MD5 value is 83fe57fd75239fda659aff2998e6f4c0. Additionally, you can use **a disk image file** "SuspectDrive1.E01".

Steps:

1. Test running **log2timeline** and **check its version** by invoking the following:
   `$ log2timeline.py -V`

2. List **all the options** of log2timeline by running:
   `$ log2timeline.py -h | less`

3. List the options for its available **timezone setting** by running:
   `$ log2timeline.py -z list`
   You should be able to see the options: "`Asia/Singapore : +08:00`" and "`Singapore : +08:00`"

4. Now, create a **folder** called `EventLogs`. Download the `Security.evtx` file, and put it into the `EventLogs` folder.
   You can also put some other time-containing artefact files into the folder.

   (***Note***: Instead of a target folder, you can even specify a target **disk image file** such as the `SuspectDrive1.E01` file. Yet, be careful that analyzing an image file usually *takes much longer*! To deal with this issue, you can filter artefact types that you want to process by specifying your **selected parsers** with `--parsers` option. The list of parsers can be seen at: https://plaso.readthedocs.io/en/latest/sources/user/Parsers-and-plugins.html).

5. Run the following command to produce a SQLite-based "***Plaso storage***" **file** named `events.plaso` from all the files inside the input folder:
   ```
   $ log2timeline.py -z "Singapore" events.plaso
     EventLogs/
   ```

   By default, the log2timeline command outputs times in the **UTC time zone**. Hence, if you need to set a specific time zone like Singapore time zone, you can do with the **`-z` option** as shown in the command above.

The log2timeline command will give you some information like in the following screenshot:

```
plaso - log2timeline version 20200121

Source path             : /home/sansforensics/EventLogs
Source type             : directory
Processing time         : 00:00:21

Tasks:          Queued  Processing      Merging         Abandoned       Total
                0       0               0               0               3

Identifier      PID     Status          Memory          Sources         Events          File
Main            2940    completed       148.3 MiB       3 (0)           19762 (0)
Worker_00       2947    idle            115.4 MiB       0 (0)           11137 (0)       OS:/home/sansforensics/EventLogs/System.evtx
Worker_01       2949    idle            116.4 MiB       2 (0)           8625 (0)        OS:/home/sansforensics/EventLogs/Security.evtx

Processing completed.

Number of warnings generated while extracting events: 214.

Use pinfo to inspect warnings in more detail.
```

6. **Inspect** the outputted Plaso storage file by running:

   `$ pinfo.py events.plaso`

   You should see the information about the storage file, and also the number of events generated by the conducted log2timeline parsing.

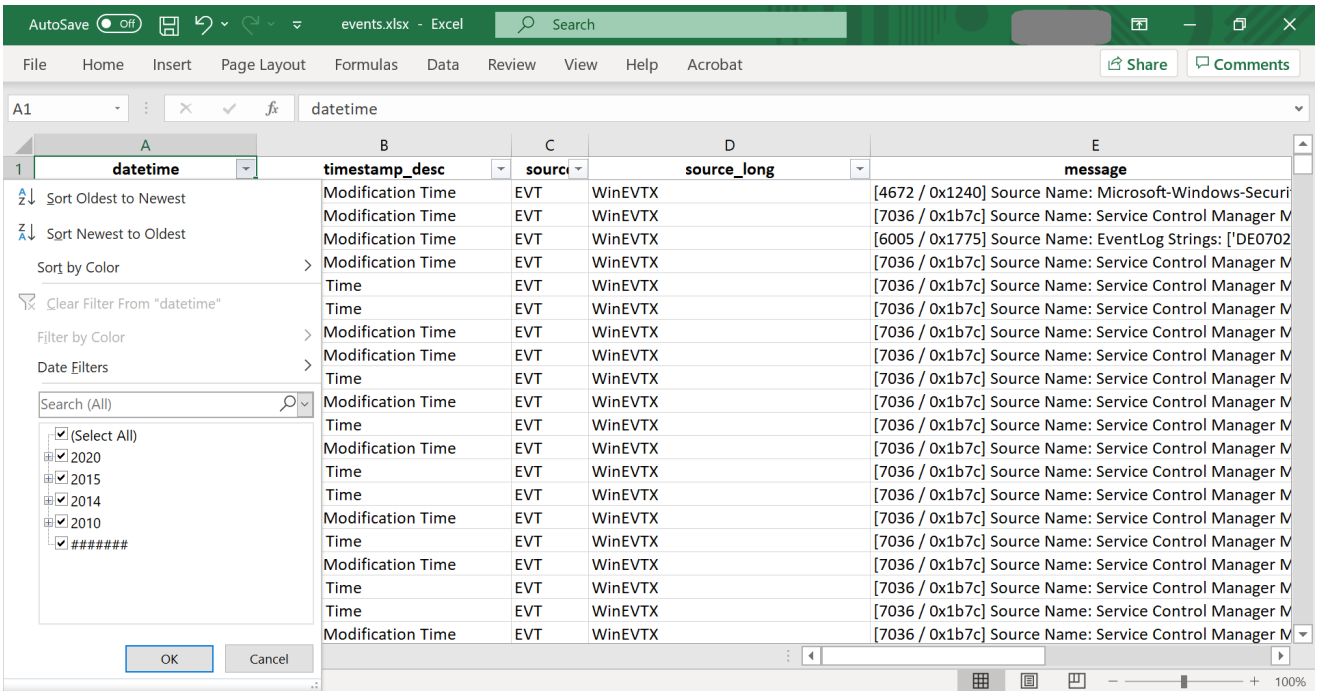
7. Now, **convert** the Plaso storage file into an **inspectable Excel file** by executing:

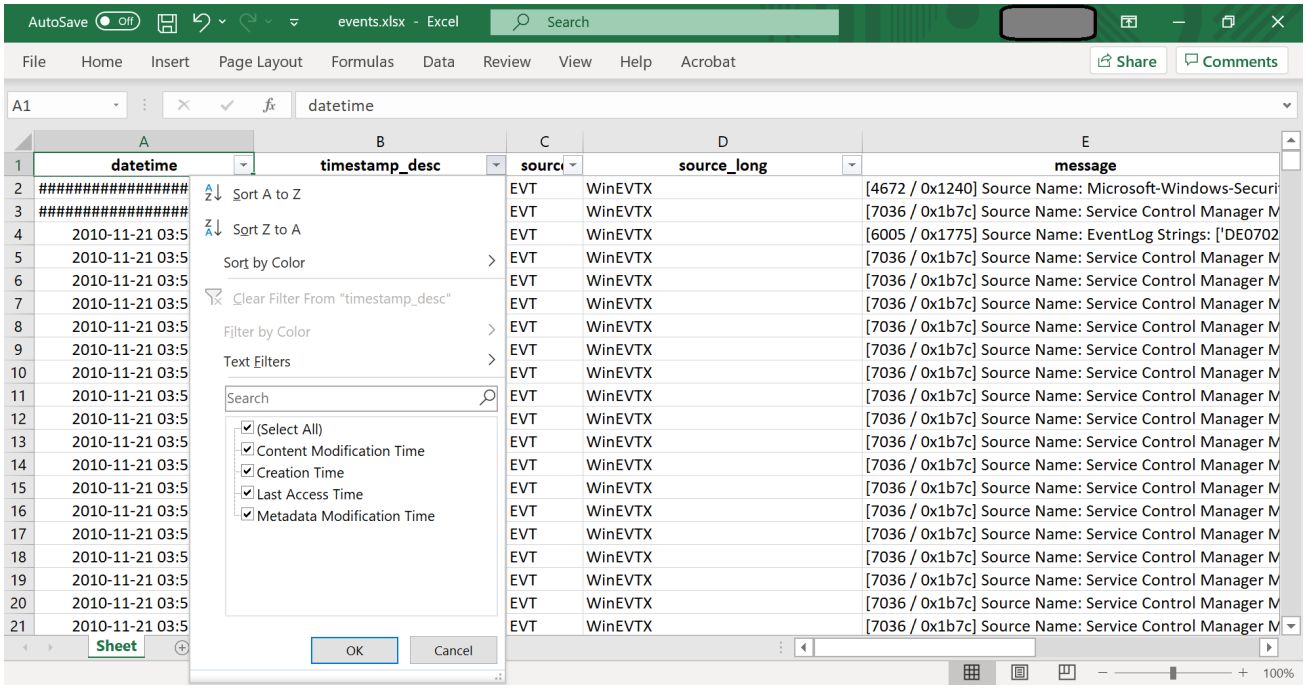
   `$ psort.py -z "Singapore" -o xlsx -w events.xlsx events.plaso`

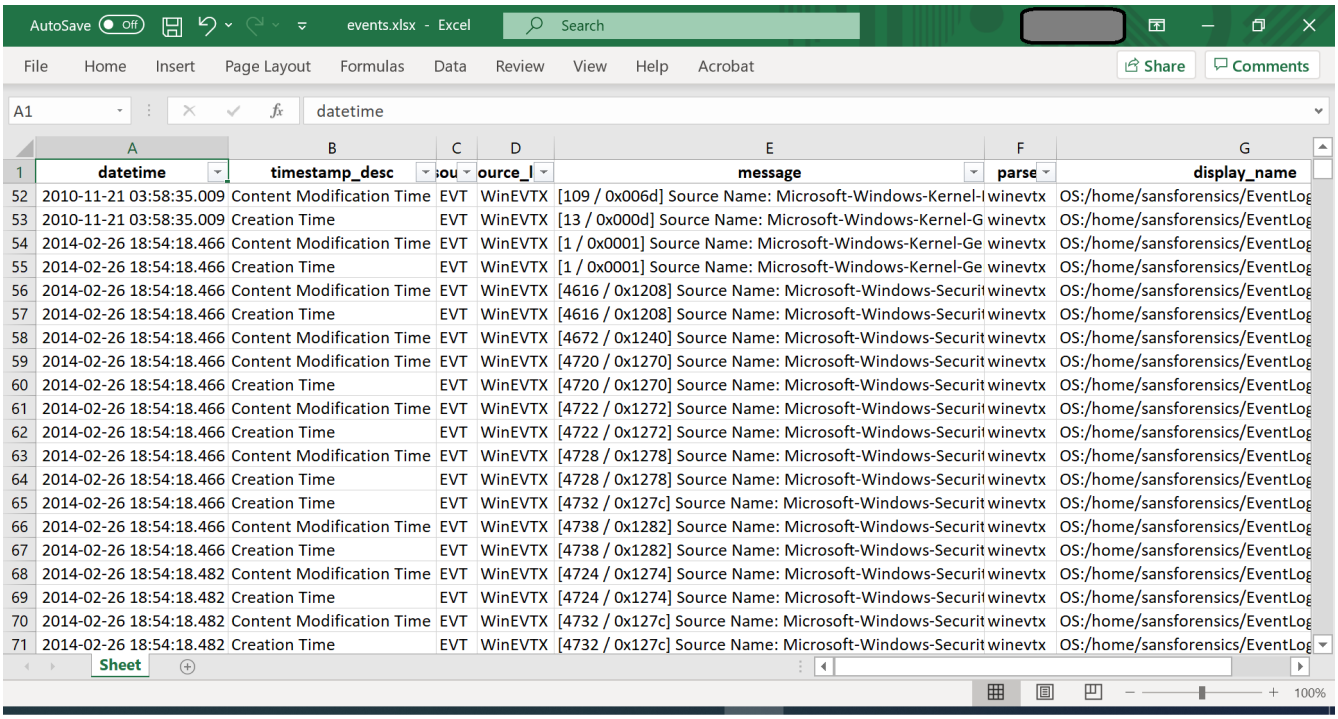
   The psort command should give you some information as follows:

```
plaso - psort version 20200121

Storage file            : events.plaso
Processing time         : 00:00:16

Events:         Filtered        In time slice   Duplicates      MACB grouped    Total
                0               0               0               19762           19762

Identifier              PID     Status          Memory          Events          Tags            Reports
Main                    3010    exporting       84.0 MiB        19762 (276)     0 (0)           0 (0)

Processing completed.
```

8. Open the resulting Excel file events.xlsx. You should be able to inspect the **columns** and **available options** of the Excel file as shown in the following three screenshots.


AutoSave Off
events.xlsx - Excel
Search
File Home Insert Page Layout Formulas Data Review View Help Acrobat
Share Comments
A1
datetime
Sort Oldest to Newest
Sort Newest to Oldest
Sort by Color
Clear Filter From "datetime"
Filter by Color
Date Filters
Search (All)
(Select All)
2020
2015
2014
2010
#######
OK
Cancel
100%


AutoSave Off
events.xlsx - Excel
Search
File Home Insert Page Layout Formulas Data Review View Help Acrobat
Share Comments
A1
datetime
Sort A to Z
Sort Z to A
Sort by Color
Clear Filter From "timestamp_desc"
Filter by Color
Text Filters
Search
(Select All)
Content Modification Time
Creation Time
Last Access Time
Metadata Modification Time
OK
Cancel
Sheet
100%



AutoSave Off events.xlsx - Excel Search

File Home Insert Page Layout Formulas Data Review View Help Acrobat Share Comments

A1 datetime

	A	B	C	D	E	F	G
1	datetime	timestamp_desc	ou	ource_I	message	parse	display_name
52	2010-11-21 03:58:35.009	Content Modification Time	EVT	WinEVTX	[109 / 0x006d] Source Name: Microsoft-Windows-Kernel-I	winevtx	OS:/home/sansforensics/EventLog
53	2010-11-21 03:58:35.009	Creation Time	EVT	WinEVTX	[13 / 0x000d] Source Name: Microsoft-Windows-Kernel-G	winevtx	OS:/home/sansforensics/EventLog
54	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[1 / 0x0001] Source Name: Microsoft-Windows-Kernel-Ge	winevtx	OS:/home/sansforensics/EventLog
55	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[1 / 0x0001] Source Name: Microsoft-Windows-Kernel-Ge	winevtx	OS:/home/sansforensics/EventLog
56	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[4616 / 0x1208] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
57	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[4616 / 0x1208] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
58	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[4672 / 0x1240] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
59	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[4720 / 0x1270] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
60	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[4720 / 0x1270] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
61	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[4722 / 0x1272] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
62	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[4722 / 0x1272] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
63	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[4728 / 0x1278] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
64	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[4728 / 0x1278] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
65	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[4732 / 0x127c] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
66	2014-02-26 18:54:18.466	Content Modification Time	EVT	WinEVTX	[4738 / 0x1282] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
67	2014-02-26 18:54:18.466	Creation Time	EVT	WinEVTX	[4738 / 0x1282] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
68	2014-02-26 18:54:18.482	Content Modification Time	EVT	WinEVTX	[4724 / 0x1274] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
69	2014-02-26 18:54:18.482	Creation Time	EVT	WinEVTX	[4724 / 0x1274] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
70	2014-02-26 18:54:18.482	Content Modification Time	EVT	WinEVTX	[4732 / 0x127c] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog
71	2014-02-26 18:54:18.482	Creation Time	EVT	WinEVTX	[4732 / 0x127c] Source Name: Microsoft-Windows-Securit	winevtx	OS:/home/sansforensics/EventLog

Sheet 100%

9. Alternatively, you can run psort in Step 8 to produce **a CSV file** as follows:

```
$ psort.py -z "Singapore" -o l2tcsv -w events.csv
events.plaso
```

10. Subsequently, open and analyze the CSV file using **Timeline Explorer**:
    a. First, you can inspect its legend information from the Help menu to see its color-coding scheme.
    b. Then, do click the “Search options” at the bottom right and specify a condition that you want to search.
    c. Lastly, you can click on the “Source Description” column header to select/filter selected artefact types.

Task 3 (Win-FWS): Running `wmic` in Live Analysis

Note:

- Let us run some **`wmic` commands** to inspect the state of a Windows machine assumed to be under an ongoing intrusion/attack.

Steps:

1. * To get the **total number of CPU cores** in your Windows machine:
   `wmic cpu get numberofcores`
2. To get the **Product ID** of your computer system product (hardware):
   `wmic csproduct get name`
3. To get the list of **all installed applications** in your system
   (***Note***: this command may take some time to finish):
   `wmic product get name`
4. To list **all user accounts** in **a particular remote computer**:
   `wmic /node:`<*remote-IP-address*> `/user:`<*username*>
   `useraccount list full`
5. * To get which user is **logged on** in your system:
   `wmic computersystem get username`
6. To get the process ID and executable pathname of all **running processes**:
   `wmic process get processid, executablepath`
7. To get the executable pathname, process ID, and parent's process ID of a **certain running process** (e.g. `chrome.exe`):
   `wmic process where "name='chrome.exe'" get executablepath, processid, parentprocessid`
8. To list **Auto Start processes**:
   `wmic startup list full`

[Optional] Task 4 (Win-FWS): Using Some GUI-based Process-Monitoring Tools in Windows

Note:

- If your are curious, you can additionally run some **GUI-based process monitoring tools** in Windows, including SysInternals' Process Explorer, Process Hacker, and SysInternals' Autoruns.

Task 4-1 (Win-FWS): Using SysInternals' Process Explorer

Note:

- ***Process Explorer*** is a popular task manager and system monitor for Windows. It provides the functionality of Windows Task Manager along with other additional features for collecting information about processes running on a system.

Steps:

1. **Download *Process Explorer*** from https://docs.microsoft.com/en-us/sysinternals/downloads/process-explorer.
2. **Run** Process Explorer as Administrator.
3. Inspect all the **running processes**, if necessary, by following the descriptions given at the site above.

Task 4-2 (Win-FWS): Using Process Hacker

Note:

- ***Process Hacker*** is a popular alternative multi-purpose tool that helps you monitor processes (including those that have active network connections). It can be useful for software debugging and malware detection as well.

Steps:

1. **Download** Process Hacker from https://processhacker.sourceforge.io/.
2. **Run** Process Hacker as Administrator.
3. Inspect the **running processes** by following some **usage tips** given at:
   a. https://www.varonis.com/blog/process-hacker
   b. https://www.socinvestigation.com/process-hacker-tool-that-helps-analyst-to-debug-software-and-detect-malware/

Task 4-3 (Win-FWS): Using SysInternals' Autoruns

Notes:

- ***Autoruns*** inspects auto-starting locations of any start-up monitor.
  It shows you **what programs are configured to run** during system bootup or login; and when you start various built-in Windows applications like Internet Explorer, Explorer and media players.
- Autoruns also reports Explorer shell extensions, toolbars, browser helper objects (BHOs), Winlogon notifications, auto-start services, among others.

Steps:

1. **Download** Autoruns from https://docs.microsoft.com/en-us/sysinternals/downloads/autoruns.
2. **Run** Autoruns as Administrator.
3. **Inspect** the reported entries, if necessary, by following the descriptions given at the site above.

Task 5 (Win-FWS): Using KAPE for Incident Response

Notes:

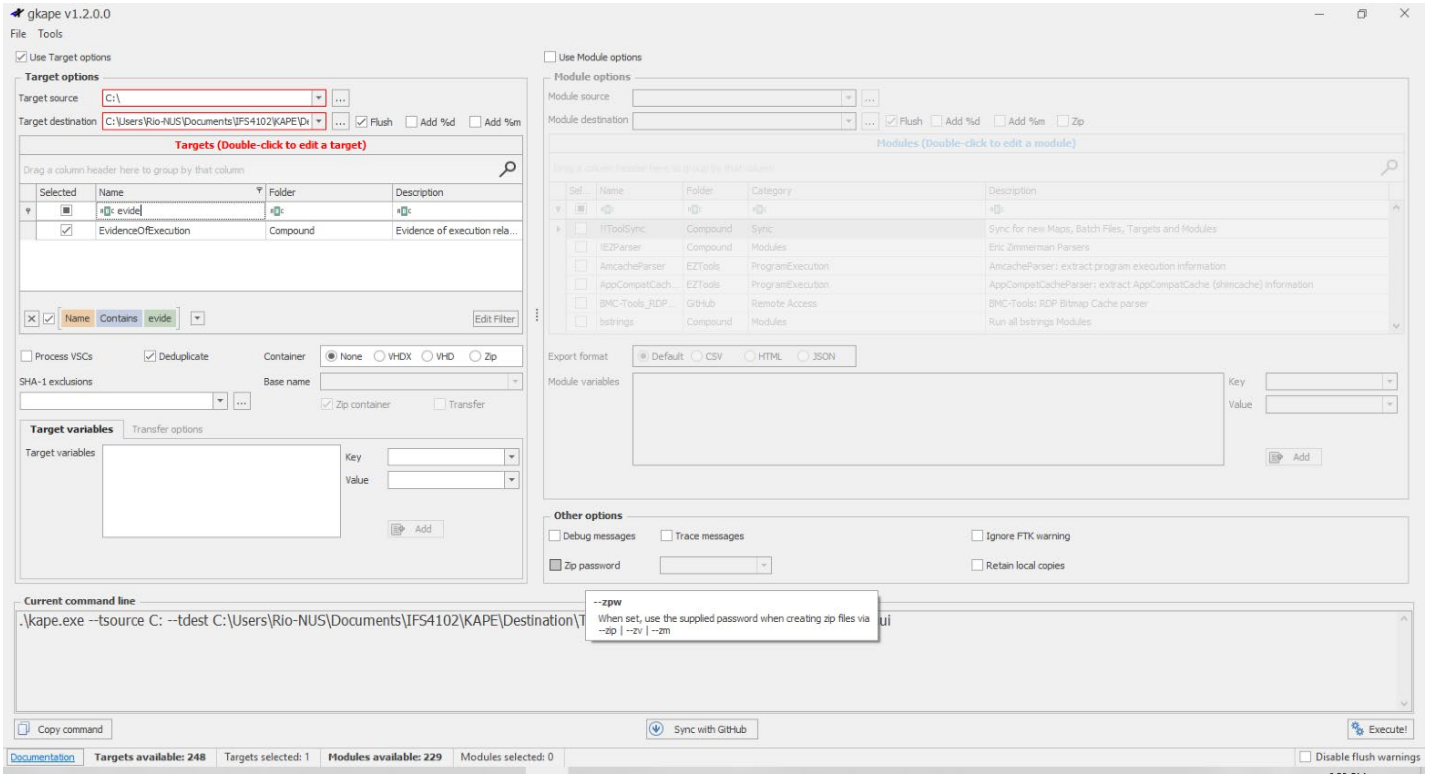
- Let's now run **KAPE** for **incident response's evidence extraction & parsing**. For the KAPE targets and modules to be used, let us focus on *past/recent program executions*.
- To simplify KAPE's usage, we will use its GUI interface called **gkape**.

Steps:

1. **Download** KAPE's zipped executable-program files from: https://www.kroll.com/en/insights/publications/cyber/kroll-artifact-parser-extractor-kape. Do provide an **accessible** email address since the download link is sent to your entered email address.
2. KAPE is a **standalone program**, which does *not* need to be installed. Hence, just **extract** the downloaded zip file to your selected directory. You should notice that the two executable program files, `kape.exe` and `gkape.exe`, are in the folder.
3. Run **`gkape.exe`** as **Administrator**. (***Note***: The two KAPE programs require Administrator rights when run. As such, when invoking the executables, always launch them with Administrator rights.)
4. Tick the "**Use Target options**" so that you can perform the evidence collection stage.
5. You need to specify your "**Target source**" as highlighted below.
   For this, you can select **a drive** where your local Windows is installed.
   Alternatively, you can **mount a disk image file** like in your Lab 2,
   and then select the corresponding accessible drive.
   (Note, however, that KAPE recommends the free "Arsenal Image Mounter

(AIM)" tool from https://arsenalrecon.com/, and not FTK Imager or Dokan, for mounting a disk image file as mentioned in: https://ericzimmerman.github.io/KapeDocs/#!Pages%5C2.-Getting-started.md.)

Then, select a folder as your "**Target destination**" as highlighted below.

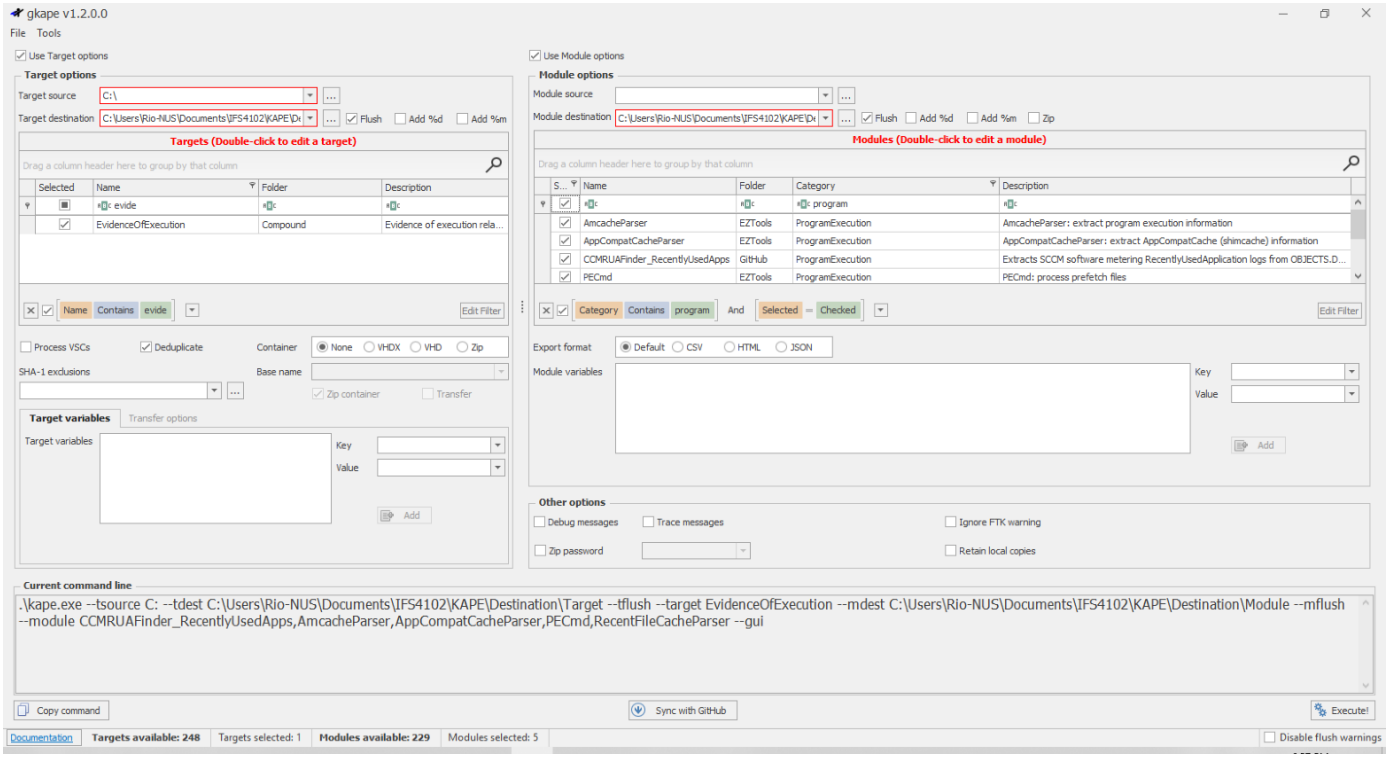


6. In the **Targets pane**, right under the "Name" column heading, do type "evidence". You should see a matching target entry named "**EvidenceOfExecution**". Do select it as your target entry like shown in the screenshot above.

7. Notice how gkape also shows the corresponding **command-line version** of the evidence-extraction operation. Now, click the "**Execute**" button. If the "Flush" option is selected, a warning is shown informing that the contents of the given target destination and/or module destination will be deleted prior to KAPE execution.

8. Then, check your **target destination folder**. Open a file ending with `_ConsoleLog.txt` for the latest's KAPE target execution log.
   Also review a folder created for your given target source.
   Inside the folder, you should see **a number of created sub-folders** where the relevant artefact files are extracted from the target source.

9. Now, let's also run some **KAPE modules**. Tick the "**Use Module options**". You need to specify a folder for your **module destination** as highlighted below. To analyze recent program executions, in the Modules pane, you can type "program" right under the Category column heading.
   Do select the shown entries under the "**ProgramExecution" category**, including the following KAPE modules: AmcacheParser, AppCompatCacheParser, PECmd, RecentFileCacheParser.



10. Finally, check your **module destination folder**. Open a file ending with `_ConsoleLog.txt` for the latest's KAPE module execution log.
    Browse an output folder created, and also review the outputted files.

Graded Lab Tasks #5 (1.5 Marks)

From your Lab 8, you will need to submit **your 2 answers** according to the following instructions:

- The selected **2 instructions** in this lab are:
  - (0.75 marks) **Task 3, Step 1 (page 8)**: Get the total number of CPU cores in your Windows machine.
  - (0.75 marks) **Task 3, Step 5 (page 8)**: Get which user is logged on in your system.
- You can just report the outputs in **your Windows (VM) system** used, and you will earn a total of **1.5 marks**.
- This graded lab task assignment is an **individual** assignment. Hence, you MUST finish the assignment and report **independently**.
- Please prepare your answers in a self-contained **PDF file** by using your name and matric number as part of your file name. For example, Jack Lee with Matric No A001 should submit the filename JackLee-A001-**GLT5**.pdf. Your report should also contain your name, matric number, and email address on its first page.
- Upload your PDF file using **`Graded-Lab-Tasks-5`** Canvas Assignment by **Saturday, 25 March 2023, 23:59 SGT**. Note that this deadline is a ***firm & final*** **deadline**. There will be ***no*** deadline extensions. As such, you are advised to submit **well before** the cut-off time so as to avoid any technical issues with Canvas or your uploading!
- *Have fun with your assigned lab tasks!* :)