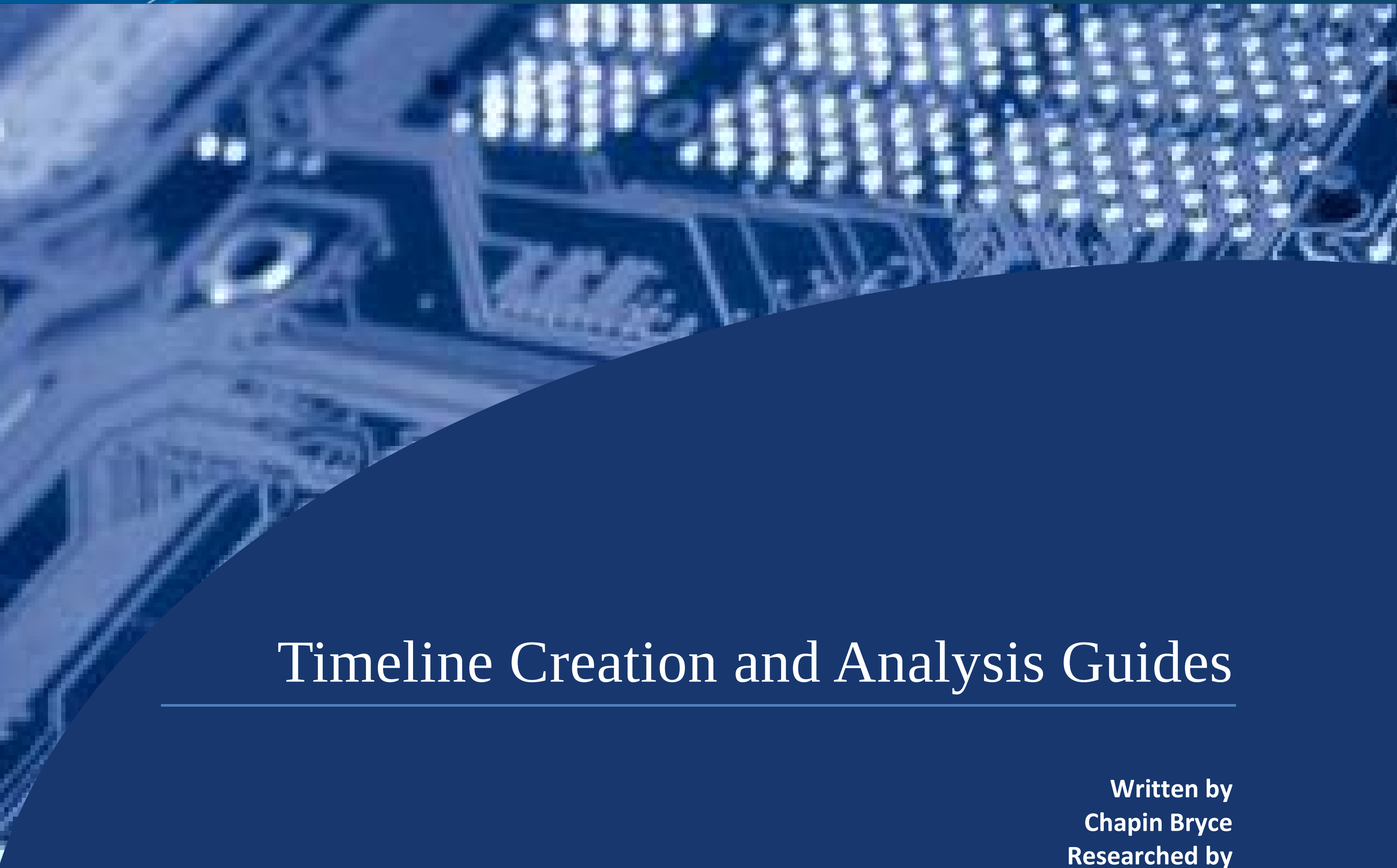
CHAMPLAIN COLLEGE

1878

LCDI

The Senator Patrick Leahy
Center for Digital Investigation



Timeline Creation and Analysis Guides

Written by
Chapin Bryce
Researched by
Chapin Bryce

175 Lakeside Ave, Room 300A
Phone: 802/865-5744
Fax: 802/865-6446
http://www.lcdi.champlin.edu

Disclaimer:

This document contains information based on research that has been gathered by employee(s) of The Senator Patrick Leahy Center for Digital Investigation (LCDI). The data contained in this project is submitted voluntarily and is unaudited. Every effort has been made by LCDI to assure the accuracy and reliability of the data contained in this report. However, LCDI nor any of our employees make no representation, warranty or guarantee in connection with this report and hereby expressly disclaims any liability or responsibility for loss or damage resulting from use of this data. Information in this report can be downloaded and redistributed by any person or persons. Any redistribution must maintain the LCDI logo and any references from this report must be properly annotated.

Contents

1 Introduction

1.1 How to use this guide

This document has 5 guides that provide information about timeline creation and analysis for several different tools and platforms. Navigate to the desired guide via the table of contents. Since the tools each have a different set of strengths, sections 1.2 through 1.6 evaluate the different options and use cases for timeline generation. For most situations involving heavy use of timeline analysis, Log2Timeline is going to be the best fit for the situation, since the tools was designed for the specific task of timeline generation.

1.2 Overview of EnCase

EnCase is a popular industry tool that is used for a lot of investigations. Timeline creation with EnCase is useful if your case is already loaded into and processed by the tool. Especially when timelines are lightly used within the case, the EnCase timeline feature is the best option.

1.3 Overview of Forensic Tool Kit

AccessData's Forensic Tool Kit (FTK) provides another industry method for case analysis. Though FTK does not has a timeline view, like EnCase, it can be more difficult to seamlessly integrate the timeline data from FTK. Once again, FTK is best suited for use when timeline analysis is not the main focus, but a secondary component in the investigation since it requires additional processing.

1.4 Overview of Log2Timeline

Log2Timeline is an open source command line tool created by Kristenn Gudjonsson. Log2Timeline has become known for its ability to efficiently and accurately provide timeline information for mounted logical devices, while operating at the partition level. Log2Timeline is written in Perl, and it works by reading through a filesystem and outputting the metadata in a report format (.csv by default). The outputted report provides the date, time, date/time type, module used, and file location of the data on the partition. Log2Timeline is capable of scanning through folders and directories individually or an entire mounted partition at once. Please note that since these partitions must be mounted (as read-only for data integrity purposes), the host machine determines which file systems Log2Timeline is capable of parsing through.

1.4.1 Log2Timeline Modules

Log2Timeline is a framework that comes with a series of modules for examiners to use to extract specific information from a device. This includes clusters of modules for Linux, Macintosh, Windows 7, Windows XP, and Windows Server. In addition the module clusters, Log2Timeline also includes task specific modules, such as web history, and options for running the tool against images without registry information. For a complete list of modules, see Figure 1. These modules can be run individually, or if a cluster is selected, it will run the modules within the cluster. The modules ability of Log2Timeline makes it flexible for case work and allows the user to select the scope that Log2Timeline runs.

1.4.2 Log2Timeline Time zones

In addition to the range of available modules that Log2Timeline can run, it supports 555 different time zone options covering a large number of cities as well as every continental time zone. Log2Timeline has an extensive list of redundant time zones to ensure that daylight savings time is included/excluded based on the region also. For a complete list of time zones, run **log2timeline –z list** in the command prompt.

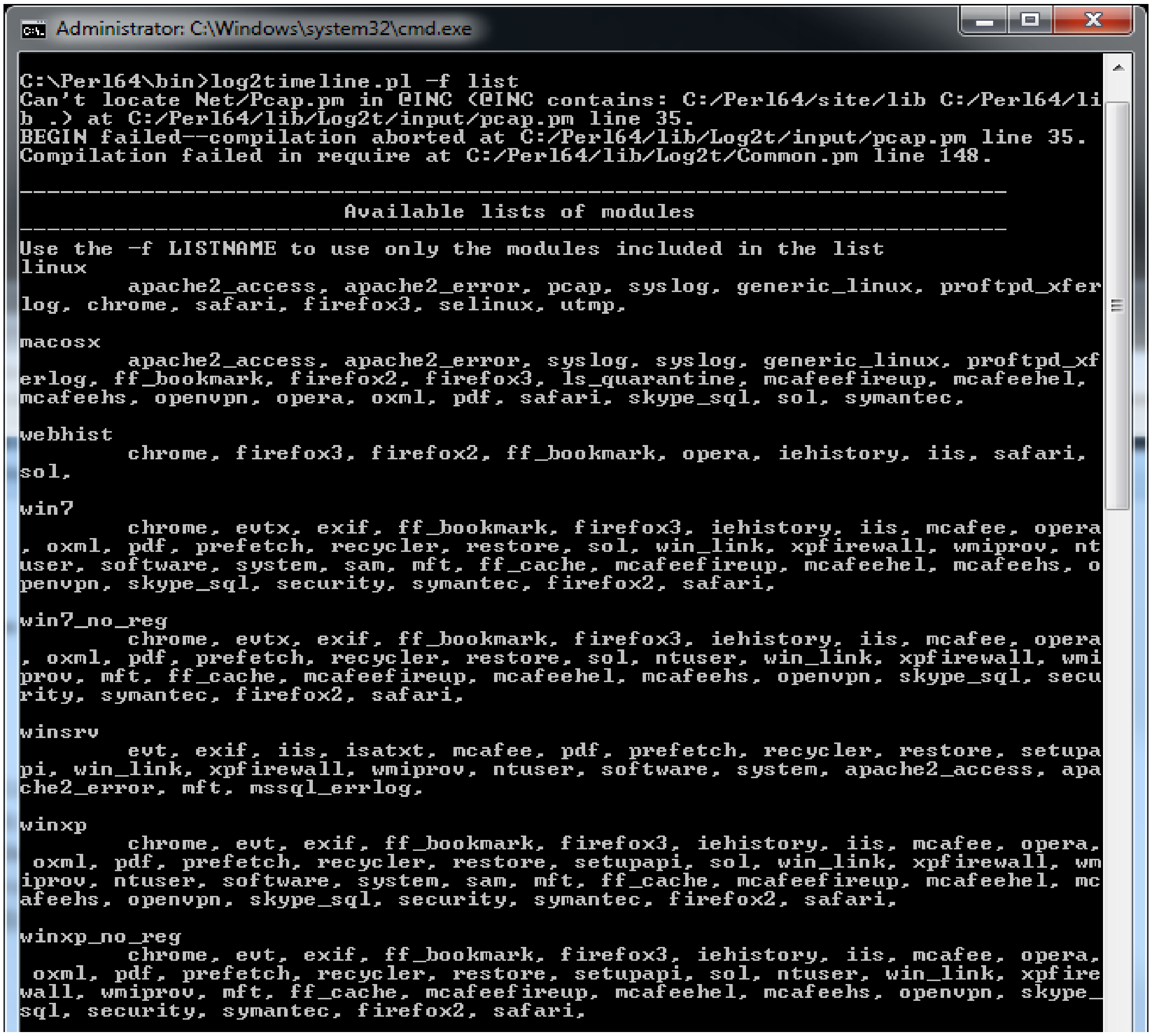
1.4.3 Log2Timeline Output

Log2Timeline supports multiple output formats for the data retrieved and parsed from the target source. The default output is CSV (Comma Separated Values), which can be opened in Microsoft Excel or any other worksheet application. CSV is a universal format and is simple for many applications to interpret and display. In addition to the default CSV format is the Mactime format. The Mactime format is closely focused on UNIX file systems, so it would not provide some information for Windows file systems (such as inode information) because the two file systems use different organizational structures. This format is used by Sleuth Kit (TSK), which outputs Date, Time, Size of file, Record Type, UNIX Permissions, User ID, Group ID, inode (for UNIX systems), and filename[1]. Mactime is stored as a bodyfile format and is encoded in plain text.

Another unique file output type is the TLN format. This output is a tab delimited format, though it is also used by Harlan Carvey's tools. The TLN is in the ASCII format and can be opened with most worksheet applications such as Microsoft Excel; it can be converted to CSV format as well.

Another common file output type is SQLite, which will place the information in a database to allow querying for timeline information versus reading an entire file. The SQLite format can be useful as the output files may become extremely large, preventing them from being opened by common applications such as Microsoft Excel or Notepad ++ (we have encountered output files as large as 2 GB). These are only a few of the supported file formats; the rest can be found at: http://log2timeline.net/#output.

[1] Information from sluethkit.org regarding sample output of the mactime format. http://wiki.sleuthkit.org/index.php?title=Mactime_output



```
C:\Perl64\bin>log2timeline.pl -f list
Can't locate Net/Pcap.pm in @INC (@INC contains: C:/Perl64/site/lib C:/Perl64/li
b .) at C:/Perl64/lib/Log2t/input/pcap.pm line 35.
BEGIN failed--compilation aborted at C:/Perl64/lib/Log2t/input/pcap.pm line 35.
Compilation failed in require at C:/Perl64/lib/Log2t/Common.pm line 148.

-------------------------------------------------------------------------
                        Available lists of modules
-------------------------------------------------------------------------
Use the -f LISTNAME to use only the modules included in the list
linux
        apache2_access, apache2_error, pcap, syslog, generic_linux, proftpd_xfer
log, chrome, safari, firefox3, selinux, utmp,

macosx
        apache2_access, apache2_error, syslog, syslog, generic_linux, proftpd_xf
erlog, ff_bookmark, firefox2, firefox3, ls_quarantine, mcafeefireup, mcafeehel,
mcafeehs, openvpn, opera, oxml, pdf, safari, skype_sql, sol, symantec,

webhist
        chrome, firefox3, firefox2, ff_bookmark, opera, iehistory, iis, safari,
sol,

win7
        chrome, evtx, exif, ff_bookmark, firefox3, iehistory, iis, mcafee, opera
, oxml, pdf, prefetch, recycler, restore, sol, win_link, xpfirewall, wmiprov, nt
user, software, system, sam, mft, ff_cache, mcafeefireup, mcafeehel, mcafeehs, o
penvpn, skype_sql, security, symantec, firefox2, safari,

win7_no_reg
        chrome, evtx, exif, ff_bookmark, firefox3, iehistory, iis, mcafee, opera
, oxml, pdf, prefetch, recycler, restore, sol, ntuser, win_link, xpfirewall, wmi
prov, mft, ff_cache, mcafeefireup, mcafeehel, mcafeehs, openvpn, skype_sql, secu
rity, symantec, firefox2, safari,

winsrv
        evt, exif, iis, isatxt, mcafee, pdf, prefetch, recycler, restore, setupa
pi, win_link, xpfirewall, wmiprov, ntuser, software, system, apache2_access, apa
che2_error, mft, mssql_errlog,

winxp
        chrome, evt, exif, ff_bookmark, firefox3, iehistory, iis, mcafee, opera,
 oxml, pdf, prefetch, recycler, restore, setupapi, sol, win_link, xpfirewall, wm
iprov, ntuser, software, system, sam, mft, ff_cache, mcafeefireup, mcafeehel, mc
afeehs, openvpn, skype_sql, security, symantec, firefox2, safari,

winxp_no_reg
        chrome, evt, exif, ff_bookmark, firefox3, iehistory, iis, mcafee, opera,
 oxml, pdf, prefetch, recycler, restore, setupapi, sol, ntuser, win_link, xpfire
wall, wmiprov, mft, ff_cache, mcafeefireup, mcafeehel, mcafeehs, openvpn, skype_
sql, security, symantec, firefox2, safari,
```

Figure 1

1.4.4 Other Log2Timeline Options

Log2Timeline allows the standard output (STDOUT), the information displayed in the command/terminal window, to be saved as a text file for later review. The **–log file/path/to/log.txt** option allows you to export essential information regarding errors encountered, the run modules, start and end time, timezone selected, and output module selected to a log file. This information is worth saving, making the **-log** option a necessity in running the command. Another useful option is the **–r** option, which tells Log2Timeline to run recursively across the source directory. This is useful when running across an entire partition or all of the subfolders within the source directory. When the recursive option is selected, the preprocessing option, **-p**, should be run to allow Log2Timeline to parse information from the source for use with some of the modules.

With the versatility to run across multiple platforms and the options to generate timelines for sources with a variety of time zones and operating systems, the Log2Timeline framework is the cornerstone of forensic timeline creation. What makes this framework so incredibly popular and widely known is its integration with different forensic platforms such as SIFT and TAPEWORM.

1.5 Log2Timeline in SIFT

SIFT is a preconfigured virtual machine appliance developed, managed, and released by SANS Forensics. SANS Investigative Forensic Toolkit (SIFT) is a suite of preinstalled open source forensic tools, allowing investigators to download a prebuilt virtual machine (VM) they can run and use for analysis of a case. SIFT prevents investigators from facing tedious installation and configuration of open source tools. This VM includes the Log2Timeline framework as well as a custom version of Log2Timeline that can be run from the terminal within SIFT to further simplify the timeline creation process.

1.5.1 Preparing Sources for Log2Timeline in SIFT

Forensic acquisitions use a variety of file types and extensions. The DD, also known as RAW, format saves a bit for bit copy of the drive as a file. This format does not support compression or encryption. This format can also be interpreted by the Linux mount command, used in SIFT, to mount acquired exhibits as virtual read-only partitions.

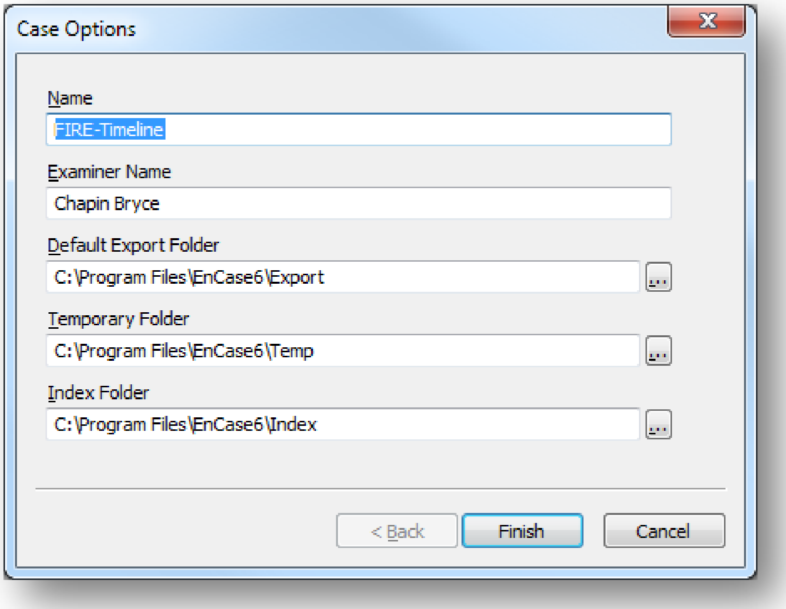
Since DD images do not support compression or encryption, the E01 format is commonly used to provide both compression and encryption

2 Timeline Creation and Analysis with EnCase 6.19 and 7.04

2.1 Timeline Analysis with EnCase 6.19

EnCase offers a built-in timeline feature capable of beginning a timeline analysis from the case file already loaded into the software. This easy-to-use timeline feature can either be used as a method to initially focus on a known date and time or to narrow results using a date and time window.

To begin timeline analysis with EnCase 6.19, a case must be created or opened and evidence must be added to it (see **Figure 1**).



Name: FIRE-Timeline

Examiner Name: Chapin Bryce

Default Export Folder: C:\Program Files\EnCase6\Export

Temporary Folder: C:\Program Files\EnCase6\Temp

Index Folder: C:\Program Files\EnCase6\Index

Figure 2 – Creating a Case in EnCase 6.19

Once evidence is added into EnCase, and the software is done parsing the MFT (Master File Table) and other pre-processing requirements, the timeline feature can be used. By green plating an evidence file (whether it is an entire case, partition, folder, or uncompressed compound file), all of the entries within the evidence file will be displayed. Then, after selecting the “Timeline” view from the right pane, the program will display a graphical representation of the drive in relation to the calendar displayed (See Figure 2).

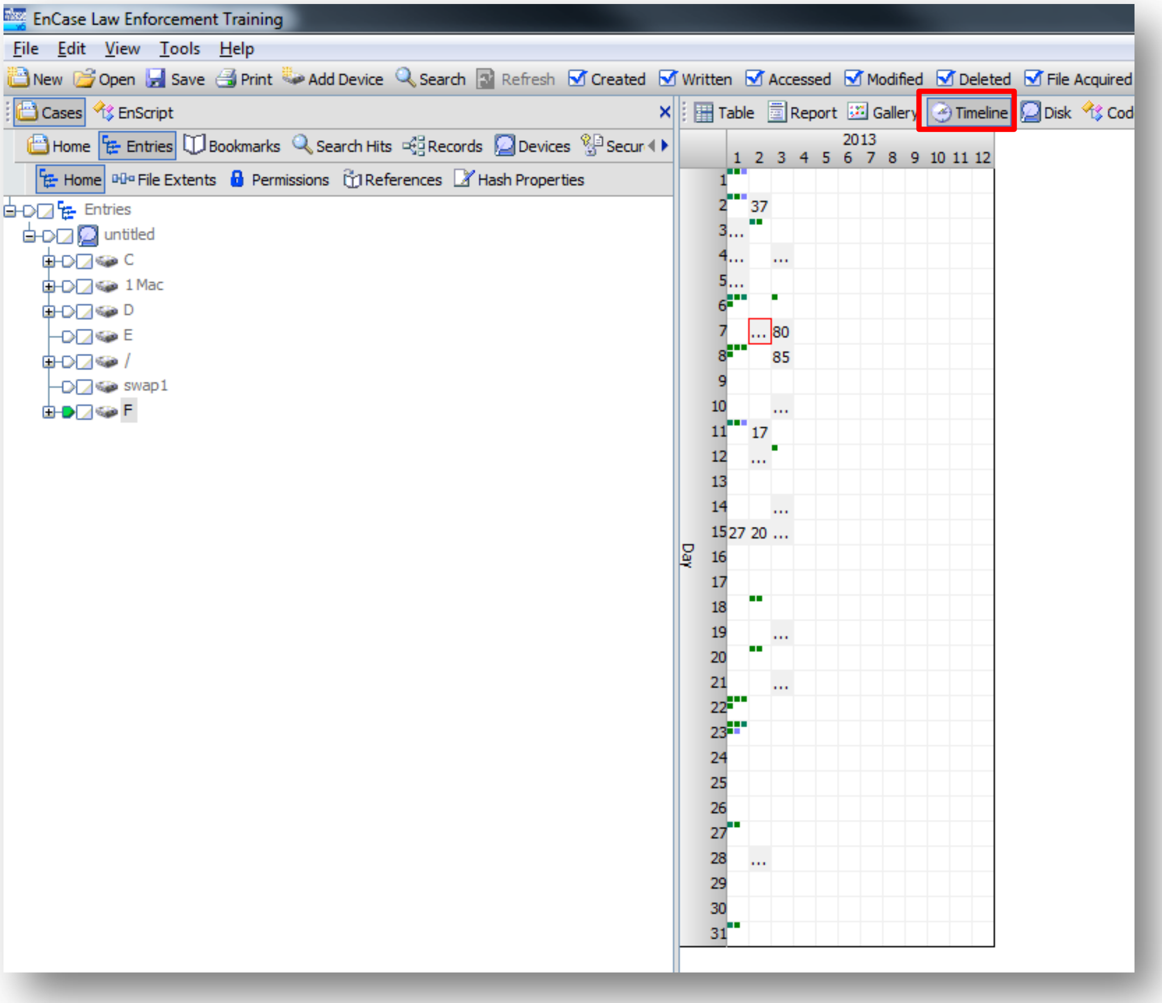
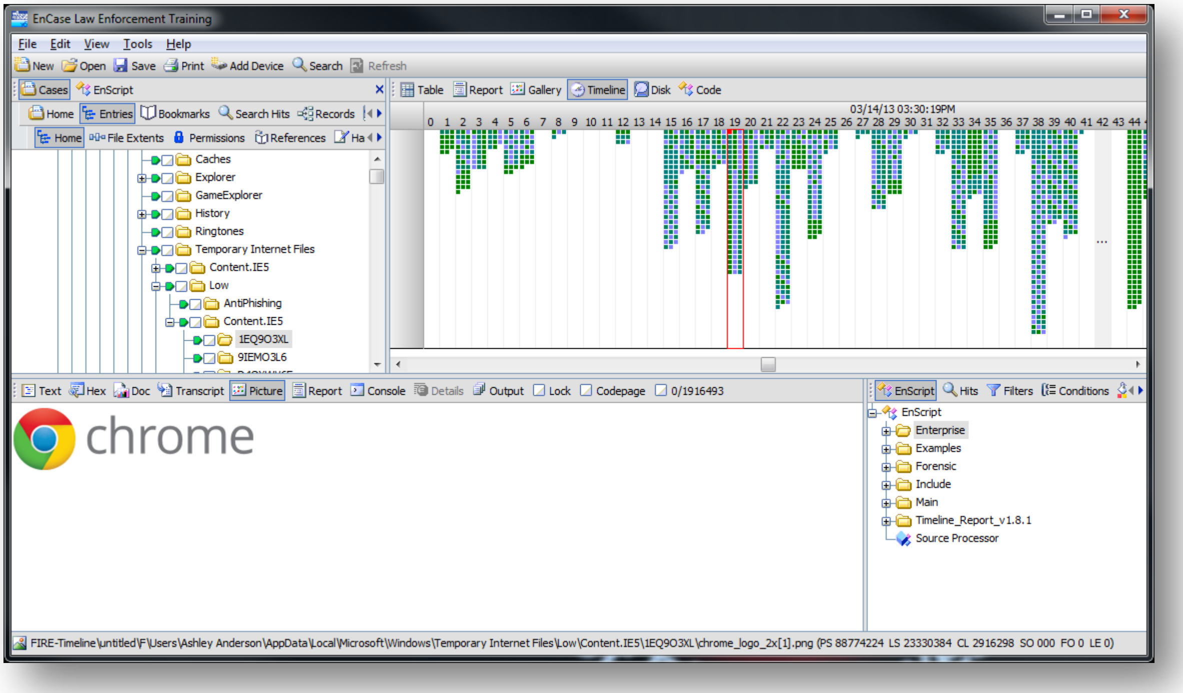

EnCase Law Enforcement Training
File Edit View Tools Help
New Open Save Print Add Device Search Refresh Created Written Accessed Modified Deleted File Acquired
Cases EnScript
Home Entries Bookmarks Search Hits Records Devices Secur
Home File Extents Permissions References Hash Properties
Entries
untitled
C
1 Mac
D
E
/
swap1
F
Table Report Gallery Timeline Disk Cod
2013
Day

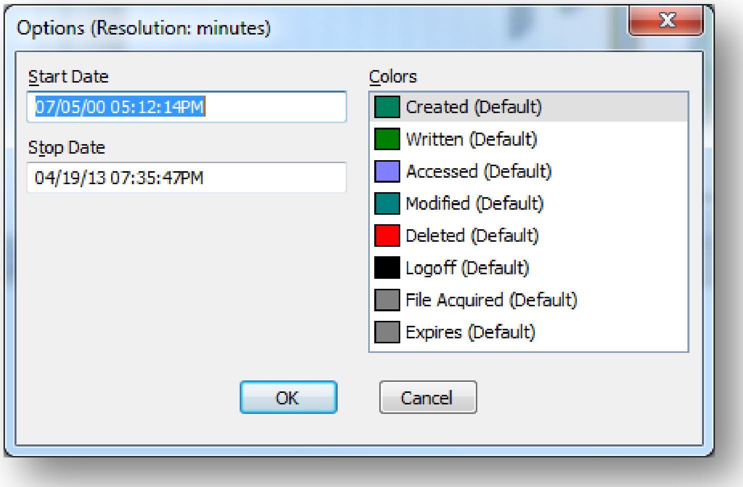
Figure 3 – Timeline Year View in Encase 6.19

To change the focus, a user can zoom in on the data in a specific range by double clicking on a cell. The right click menu can be used to zoom in and out as well. The focus ranges from a year view (**Figure 2**) to a minute view (**Figure 3**), and the focus can also be modified using the options menu and adjusting the resolution (See **Figure 4**).


EnCase Law Enforcement Training
File Edit View Tools Help
New Open Save Print Add Device Search Refresh
Cases EnScript
Home Entries Bookmarks Search Hits Records
Home File Extents Permissions References
Caches
Explorer
GameExplorer
History
Ringtones
Temporary Internet Files
Content.IE5
Low
AntiPhishing
Content.IE5
1EQ9O3XL
9IEMO3L6
Table Report Gallery Timeline Disk Code
03/14/13 03:30:19PM
Text Hex Doc Transcript Picture Report Console Details Output Lock Codepage 0/1916493
chrome
EnScript Hits Filters Conditions
EnScript
Enterprise
Examples
Forensic
Include
Main
Timeline_Report_v1.8.1
Source Processor

FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\1EQ9O3XL\chrome_logo_2x[1].png (PS 88774224 LS 23330384 CL 2916298 SO 000 FO 0 LE 0)

Figure 4 – Timeline Minute View in EnCase 6.19


Options (Resolution: minutes)

Start Date
07/05/00 05:12:14PM

Stop Date
04/19/13 07:35:47PM

Colors
- 🟩 Created (Default)
- 🟩 Written (Default)
- 🟦 Accessed (Default)
- 🟩 Modified (Default)
- 🟥 Deleted (Default)
- ⬛ Logoff (Default)
- ⬜ File Acquired (Default)
- ⬜ Expires (Default)

Figure 5 – Adjusting the Timeline Resolution in Encase 6.19

EnCase 6.19 does not have a native timeline exporting tool comparable to Log2Timeline. EnCase is intended for case analysis while Log2Timeline is designed as a preprocessing platform. EnCase excels at case analysis, however, and has the ability to mount files in the case so that each one is revealed. This allows each file to be individually identified and processed, a feature missing from Log2Timeline.

2.2 Timeline Analysis with EnCase 7.04

EnCase 7.04 is a newer version of Guidance Software's forensic tool suite with an added graphic interface upgrade, along with a number of other usability modifications. The Timeline feature was improved through the major version update, and this new version offers more usability than EnCase 6.19.

As with EnCase 6.19, a case needs to be created and evidence must be added before the timeline feature can be utilized. Once the evidence is added, it is good practice to run the file mounter option in the case processor, to ensure that all files within compound files are expanded and accurately represented by the EnCase timeline.

Similar to EnCase 6.19, a green plated evidence file can be viewed by the timeline tab within the evidence pane. Within the Timeline tab, there are buttons to increase/decrease the timeline resolution, making it more convenient to use the timeline feature within EnCase.

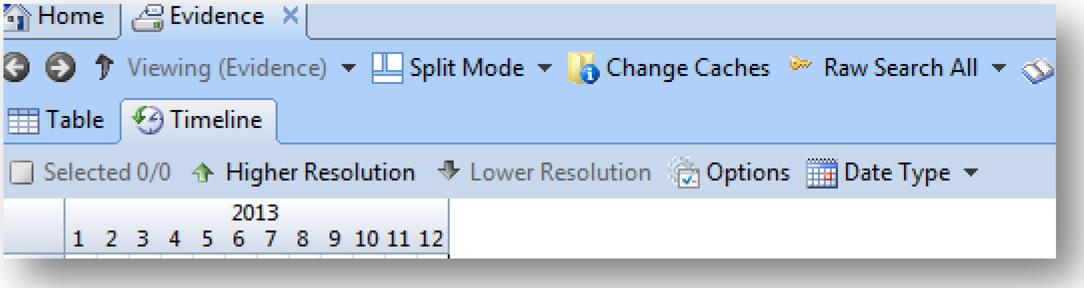


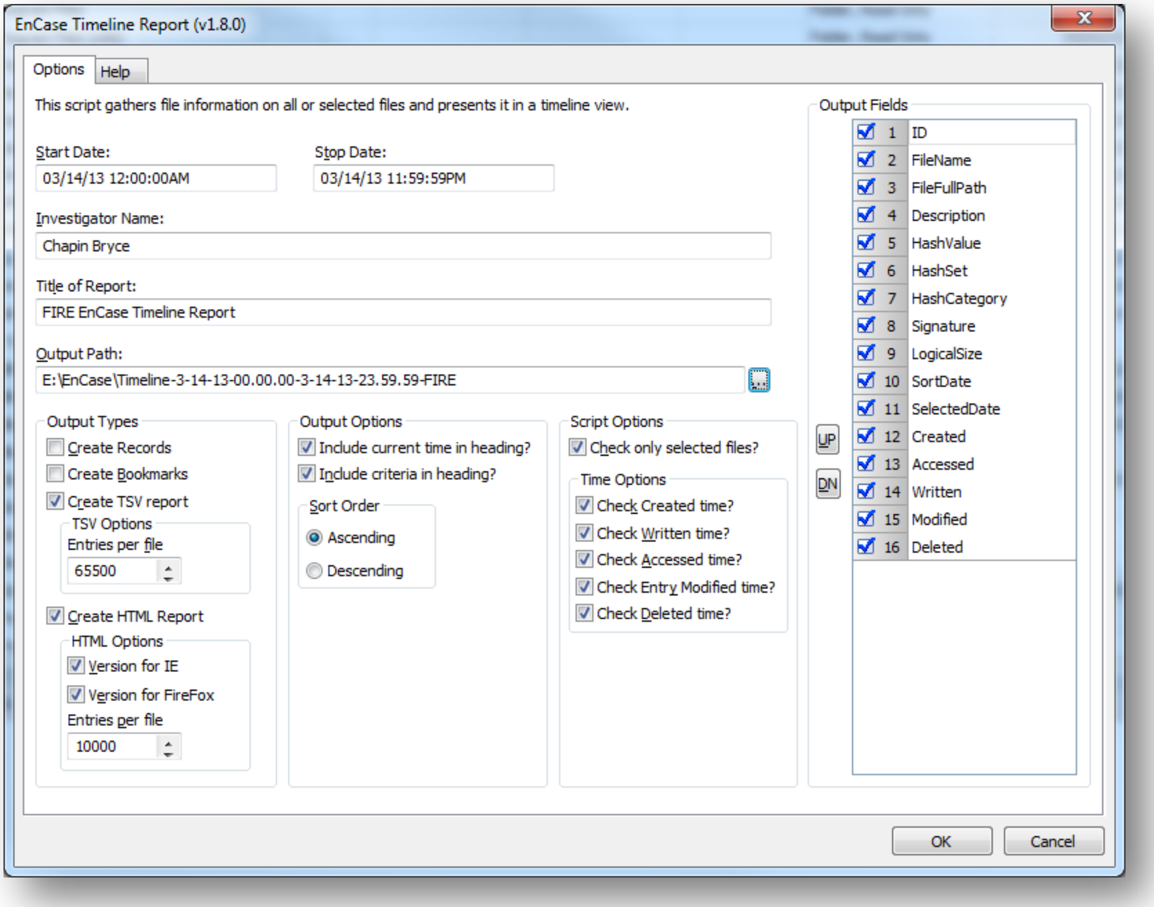
Figure 6 – Updated Resolution Buttons

Similarly to EnCase 6.19, EnCase 7.04 does not offer a timeline exporting tool similar to Log2Timeline, though EnCase 7.04 is meant for case analysis not preprocessing as Log2Timeline is. However, it should be noted that Encase 6.19 and 7.04 both support EnScripts, which can use the EnCase platform to create an exported timeline similar to Log2Timeline.

2.3 Timeline Analysis with Geoff Black's Timeline EnScript

EnCase 6.19 and 7.04 both contain timeline functionality, although it is only available for use within the program and without an easy timeline exporting tool, such as with Log2Timeline. The EnScripting feature of EnCase allows for EnScripts to be written for creating additional functionality. Geoff Black created a Timeline EnScript that uses the evidence loaded into an opened case, using EnCase as the source for creating timelines. His script may be downloaded at: GeoffBlack's Timeline EnScript. This file must be unzipped and placed within the EnCase installation directory in the EnScript folder.

To create a timeline using this EnScript, the target evidence must be loaded into a case in EnCase (see **Figure 1**). Once loaded in, it is best practice to always run the file mounter EnScript, to ensure all of the compound files can be accurately represented. For the EnScript, use the blue checking to select which files should be included in the exported timeline. Once the files are selected, open the Timeline Report EnScript within the EnScript pane of EnCase, and the option window will appear (see **Figure 6**). In this window, a series of options can be selected for creating a timeline (see **Figure 6**).



This script gathers file information on all or selected files and presents it in a timeline view.

Start Date: 03/14/13 12:00:00AM

Stop Date: 03/14/13 11:59:59PM

Investigator Name: Chapin Bryce

Title of Report: FIRE EnCase Timeline Report

Output Path: E:\EnCase\Timeline-3-14-13-00.00.00-3-14-13-23.59.59-FIRE

Output Types
- □ Create Records
- □ Create Bookmarks
- ☑ Create TSV report
  - TSV Options: Entries per file 65500
- ☑ Create HTML Report
  - HTML Options
    - ☑ Version for IE
    - ☑ Version for FireFox
    - Entries per file 10000

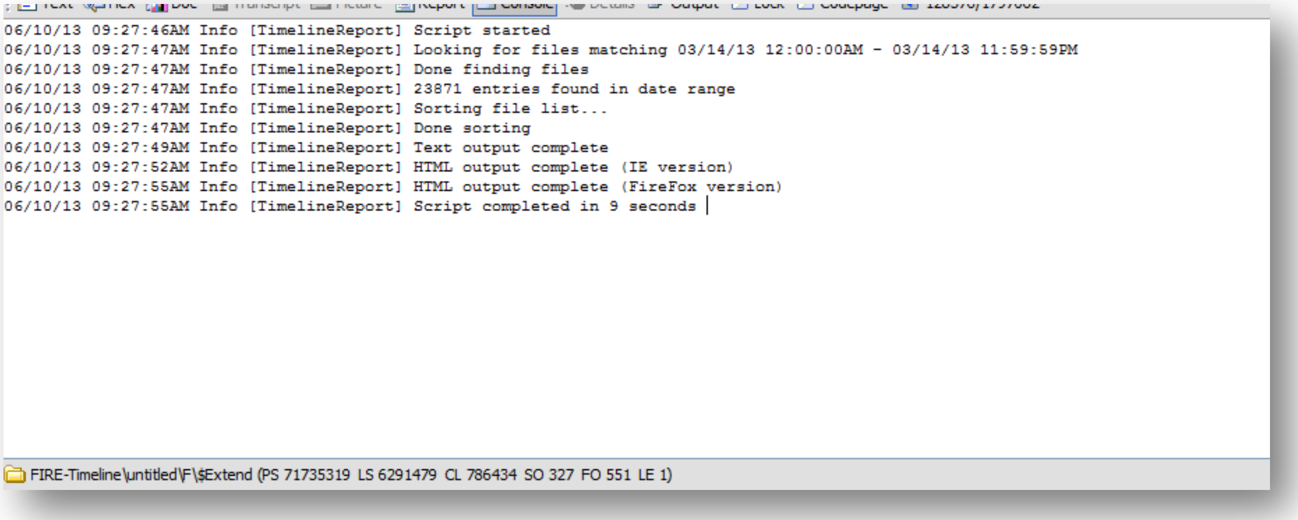
Output Options
- ☑ Include current time in heading?
- ☑ Include criteria in heading?
- Sort Order
  - ● Ascending
  - ○ Descending

Script Options
- ☑ Check only selected files?
- Time Options
  - ☑ Check Created time?
  - ☑ Check Written time?
  - ☑ Check Accessed time?
  - ☑ Check Entry Modified time?
  - ☑ Check Deleted time?

Output Fields

☑	1	ID
☑	2	FileName
☑	3	FileFullPath
☑	4	Description
☑	5	HashValue
☑	6	HashSet
☑	7	HashCategory
☑	8	Signature
☑	9	LogicalSize
☑	10	SortDate
☑	11	SelectedDate
☑	12	Created
☑	13	Accessed
☑	14	Written
☑	15	Modified
☑	16	Deleted

Figure 7 – EnCase Timeline Report EnScript

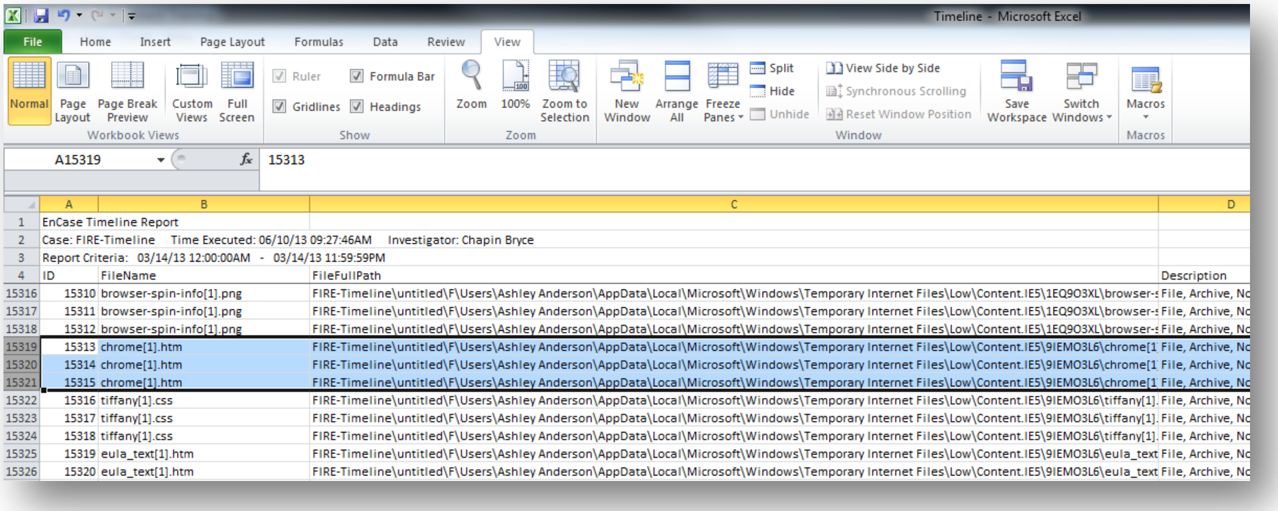


```
06/10/13 09:27:46AM Info [TimelineReport] Script started
06/10/13 09:27:47AM Info [TimelineReport] Looking for files matching 03/14/13 12:00:00AM - 03/14/13 11:59:59PM
06/10/13 09:27:47AM Info [TimelineReport] Done finding files
06/10/13 09:27:47AM Info [TimelineReport] 23871 entries found in date range
06/10/13 09:27:47AM Info [TimelineReport] Sorting file list...
06/10/13 09:27:47AM Info [TimelineReport] Done sorting
06/10/13 09:27:49AM Info [TimelineReport] Text output complete
06/10/13 09:27:52AM Info [TimelineReport] HTML output complete (IE version)
06/10/13 09:27:55AM Info [TimelineReport] HTML output complete (FireFox version)
06/10/13 09:27:55AM Info [TimelineReport] Script completed in 9 seconds
```

FIRE-Timeline\untitled\F\$Extend (PS 71735319 LS 6291479 CL 786434 SO 327 FO 551 LE 1)

Figure 8 – Run Notes from EnScript

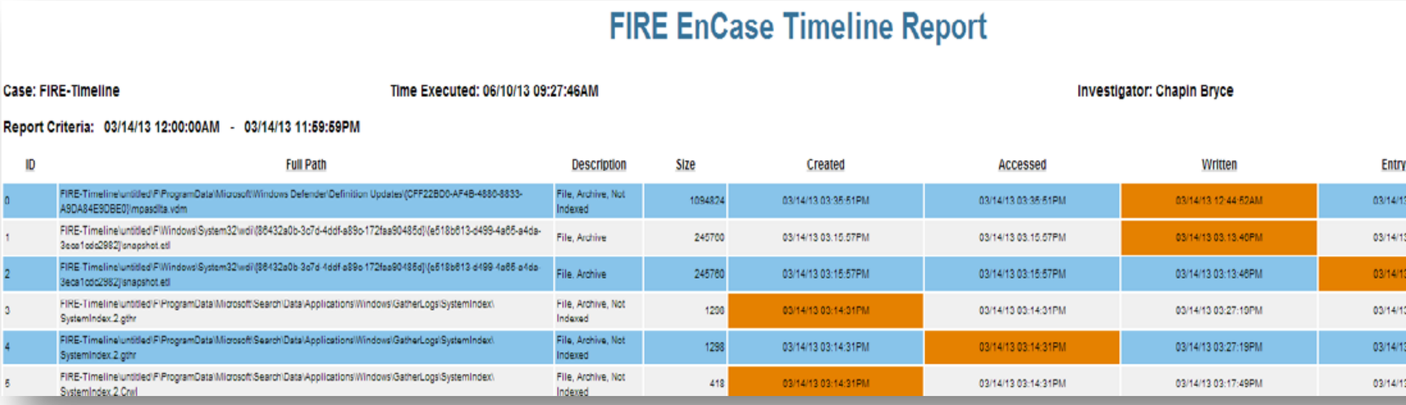
The script ran quickly and took roughly nine seconds to process the entire partition [See **Figure 7**]. While faster than Log2Timeline, the time to create the case and set it up should be accounted for as well. This timeline created by EnScript will contain more data than its log2timeline counterpart, as it has expanded compound files, but it also creates additional entries due to its structure. The exported TSV document of the timeline is shown in **Figure 8,** and the HTML output is show in **Figure 9.**



A15319 | 15313

	A	B	C	D
1	EnCase Timeline Report			
2	Case: FIRE-Timeline Time Executed: 06/10/13 09:27:46AM Investigator: Chapin Bryce			
3	Report Criteria: 03/14/13 12:00:00AM - 03/14/13 11:59:59PM			
4	ID	FileName	FileFullPath	Description
15316	15310	browser-spin-info[1].png	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\1EQ9O3XL\browser-s	File, Archive, No
15317	15311	browser-spin-info[1].png	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\1EQ9O3XL\browser-s	File, Archive, No
15318	15312	browser-spin-info[1].png	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\1EQ9O3XL\browser-s	File, Archive, No
15319	15313	chrome[1].htm	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\chrome[1	File, Archive, No
15320	15314	chrome[1].htm	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\chrome[1	File, Archive, No
15321	15315	chrome[1].htm	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\chrome[1	File, Archive, No
15322	15316	tiffany[1].css	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\tiffany[1].	File, Archive, No
15323	15317	tiffany[1].css	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\tiffany[1].	File, Archive, No
15324	15318	tiffany[1].css	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\tiffany[1].	File, Archive, No
15325	15319	eula_text[1].htm	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\eula_text	File, Archive, No
15326	15320	eula_text[1].htm	FIRE-Timeline\untitled\F\Users\Ashley Anderson\AppData\Local\Microsoft\Windows\Temporary Internet Files\Low\Content.IE5\9IEMO3L6\eula_text	File, Archive, No

Figure 9 – CSV EnScript Output



FIRE EnCase Timeline Report

Case: FIRE-Timeline Time Executed: 06/10/13 09:27:46AM Investigator: Chapin Bryce

Report Criteria: 03/14/13 12:00:00AM - 03/14/13 11:59:59PM

ID	Full Path	Description	Size	Created	Accessed	Written	Entry
0	FIRE-Timeline\untitled\F\ProgramData\Microsoft\Windows Defender\Definition Updates\{CFF22BD0-AF4B-4880-8833-A9DA84E9DBE0}\mpasdlta.vdm	File, Archive, Not Indexed	1094824	03/14/13 03:35:51PM	03/14/13 03:35:51PM	03/14/13 12:44:52AM	03/14/1
1	FIRE-Timeline\untitled\F\Windows\System32\wdi\{86432a0b-3c7d-4ddf-a89c-172faa90485d}\{e518b613-d499-4a65-a4da-3eca1cdc2982}\snapshot.etl	File, Archive	245760	03/14/13 03:15:57PM	03/14/13 03:15:57PM	03/14/13 03:13:40PM	03/14/1
2	FIRE-Timeline\untitled\F\Windows\System32\wdi\{86432a0b-3c7d-4ddf-a89c-172faa90485d}\{e518b613-d499-4a65-a4da-3eca1cdc2982}\snapshot.etl	File, Archive	245760	03/14/13 03:15:57PM	03/14/13 03:15:57PM	03/14/13 03:13:46PM	03/14/1
3	FIRE-Timeline\untitled\F\ProgramData\Microsoft\Search\Data\Applications\Windows\GatherLogs\SystemIndex\SystemIndex.2.gthr	File, Archive, Not Indexed	1290	03/14/13 03:14:31PM	03/14/13 03:14:31PM	03/14/13 03:27:19PM	03/14/1
4	FIRE-Timeline\untitled\F\ProgramData\Microsoft\Search\Data\Applications\Windows\GatherLogs\SystemIndex\SystemIndex.2.gthr	File, Archive, Not Indexed	1298	03/14/13 03:14:31PM	03/14/13 03:14:31PM	03/14/13 03:27:19PM	03/14/1
5	FIRE-Timeline\untitled\F\ProgramData\Microsoft\Search\Data\Applications\Windows\GatherLogs\SystemIndex\SystemIndex.2.Crwl	File, Archive, Not Indexed	418	03/14/13 03:14:31PM	03/14/13 03:14:31PM	03/14/13 03:17:49PM	03/14/1

Figure 10 – HTML EnScript Output

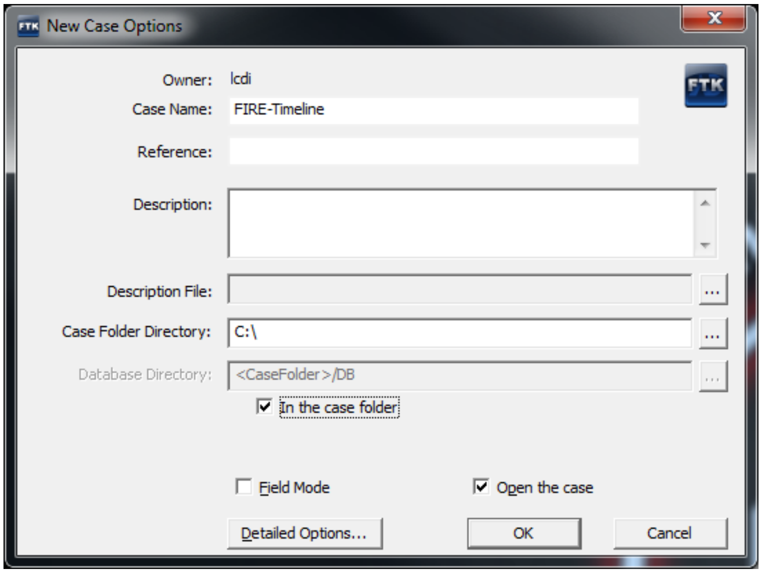
The HTML outputs are split into multiple files and are sorted by Firefox (FF) and Internet Explorer (IE) file types. Each file is split when it is around 11 MB in size. It appears that the TSV file does not split, although it was only tested at 15MB.

3 Timeline Creation and Analysis with Forensic Tool Kit 4

3.1 Timeline Analysis with Forensic Tool Kit 4

AccessData's Forensic Tool Kit (FTK) is a case analysis tool. To use FTK for timeline analysis, the preprocessing options (displayed when adding evidence to the case) must be used. Unlike EnCase, FTK does not have a built-in feature to display timeline information. To create a timeline report for viewing, a case must first be created [See **Figure 1**].

Figure 11



New Case Options

Owner: lcdi

Case Name: FIRE-Timeline

Reference:

Description:

Description File:

Case Folder Directory: C:\

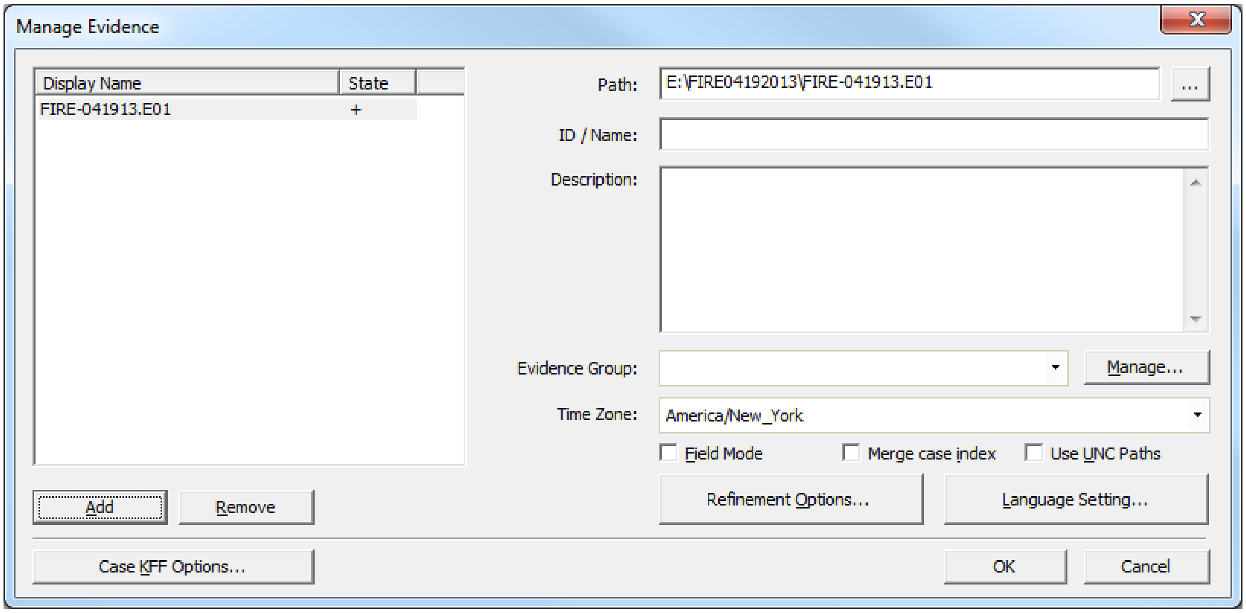
Database Directory: <CaseFolder>/DB

☑ In the case folder

☐ Field Mode ☑ Open the case

Once the new case is opened, evidence must be added to the case [See **Figure 2**]. FTK is built to handle multiple partitions and file system types, including HFS+, EXT4, NTFS, and FAT partitions, all found in the FIRE image used for this guide.

Figure 12



Manage Evidence

Display Name	State
FIRE-041913.E01	+

Path: E:\FIRE04192013\FIRE-041913.E01

ID / Name:

Description:

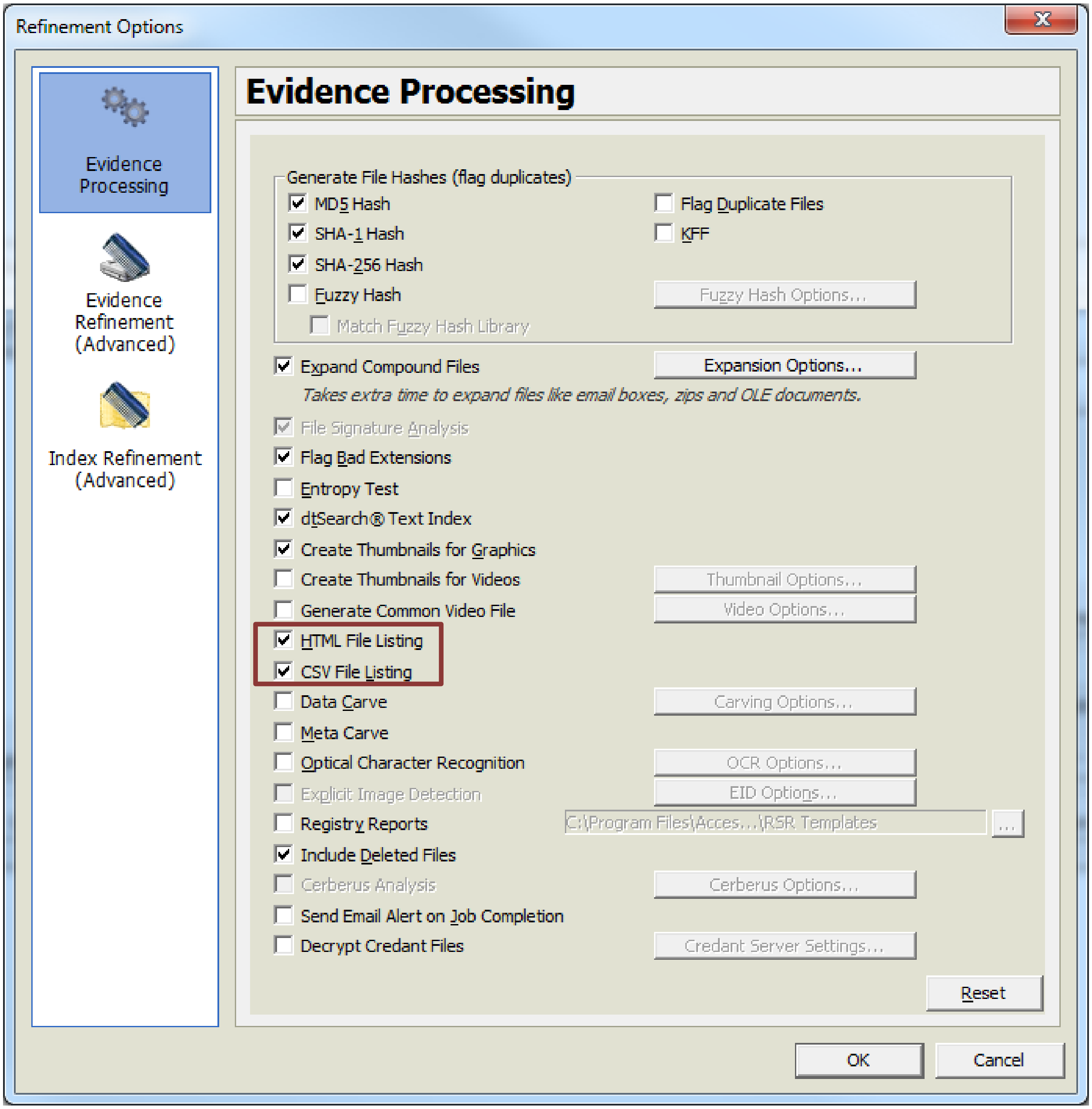
Evidence Group:

Time Zone: America/New_York

□ Field Mode □ Merge case index □ Use UNC Paths

Once evidence is added to the case, selecting the refinement options allows the desired processing options to run. To create a timeline in FTK, remember to select the HTML and/or CSV file listing options from the evidence processing window [See **Figure 3**]. These options tell FTK to create a list of files in HTML/CSV format, including the path, file name, MAC timestamps, and MD5, among others. This is the only feature allowing for timeline analysis of the drive, so be extra diligent in selecting them while processing the case.

Figure 13



Refinement Options

Evidence Processing

Generate File Hashes (flag duplicates)

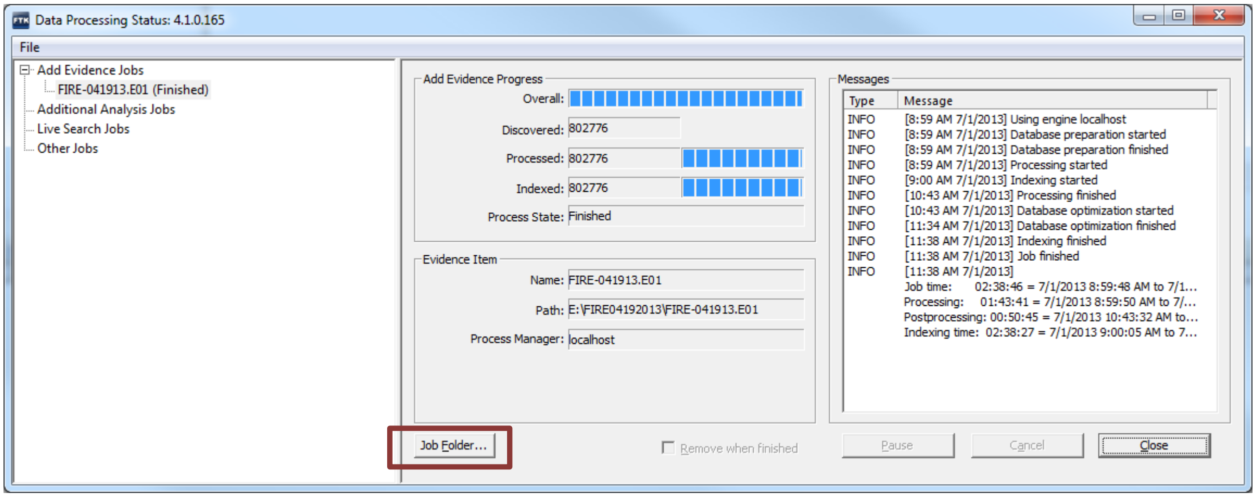
- ☑ MD5 Hash
- ☑ SHA-1 Hash
- ☑ SHA-256 Hash
- □ Fuzzy Hash
- □ Match Fuzzy Hash Library
- □ Flag Duplicate Files
- □ KFF

- ☑ Expand Compound Files

Takes extra time to expand files like email boxes, zips and OLE documents.

- ☑ File Signature Analysis
- ☑ Flag Bad Extensions
- □ Entropy Test
- ☑ dtSearch® Text Index
- ☑ Create Thumbnails for Graphics
- □ Create Thumbnails for Videos
- □ Generate Common Video File
- ☑ HTML File Listing
- ☑ CSV File Listing
- □ Data Carve
- □ Meta Carve
- □ Optical Character Recognition
- □ Explicit Image Detection
- □ Registry Reports — C:\Program Files\Acces...\RSR Templates
- ☑ Include Deleted Files
- □ Cerberus Analysis
- □ Send Email Alert on Job Completion
- □ Decrypt Credant Files

Figure 14



Data Processing Status: 4.1.0.165

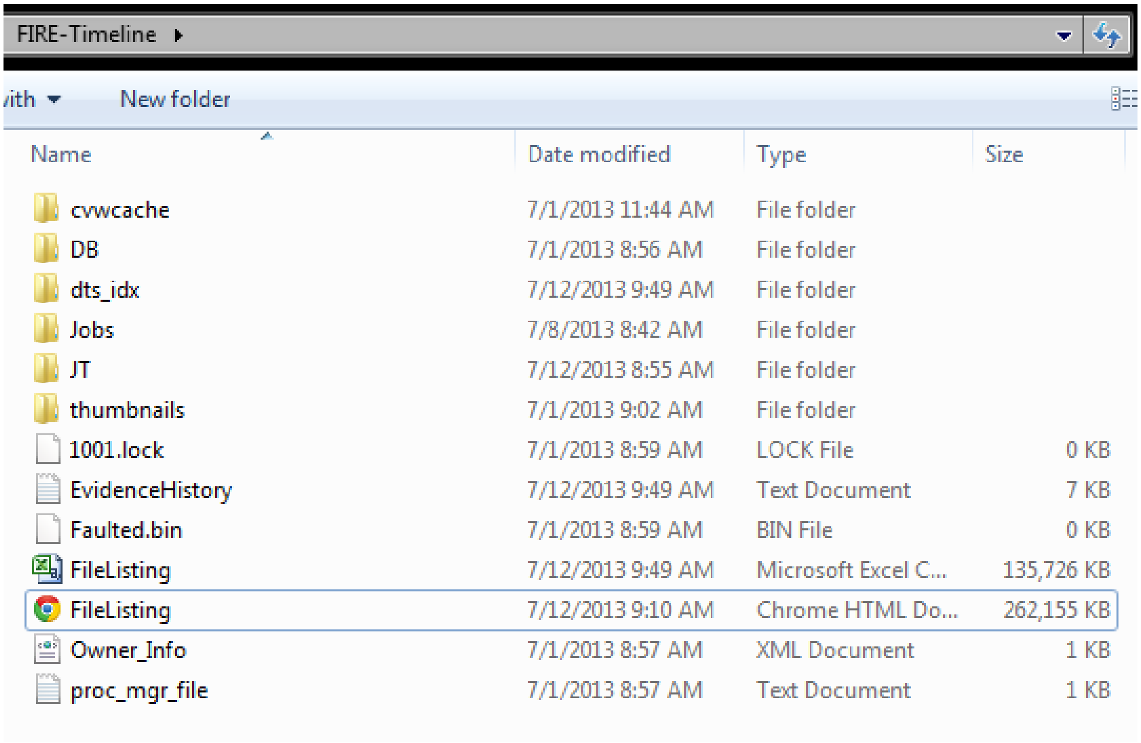
Overall:
Discovered: 802776
Processed: 802776
Indexed: 802776
Process State: Finished

Name: FIRE-041913.E01
Path: E:\FIRE04192013\FIRE-041913.E01
Process Manager: localhost

```
Type   Message
INFO   [8:59 AM 7/1/2013] Using engine localhost
INFO   [8:59 AM 7/1/2013] Database preparation started
INFO   [8:59 AM 7/1/2013] Database preparation finished
INFO   [8:59 AM 7/1/2013] Processing started
INFO   [9:00 AM 7/1/2013] Indexing started
INFO   [10:43 AM 7/1/2013] Processing finished
INFO   [10:43 AM 7/1/2013] Database optimization started
INFO   [11:34 AM 7/1/2013] Database optimization finished
INFO   [11:38 AM 7/1/2013] Indexing finished
INFO   [11:38 AM 7/1/2013] Job finished
INFO   [11:38 AM 7/1/2013]
       Job time:       02:38:46 = 7/1/2013 8:59:48 AM to 7/1...
       Processing:     01:43:41 = 7/1/2013 8:59:50 AM to 7/...
       Postprocessing: 00:50:45 = 7/1/2013 10:43:32 AM to...
       Indexing time:  02:38:27 = 7/1/2013 9:00:05 AM to 7...
```

Running the preprocessing with this many options selected can be time consuming and can also cause FTK to consume a large number of computer resources while it runs. Once the task is complete, navigate to the job folder or select the job folder button on the data processing window [See **Figure 4**]. Within the case directory are the HTML/CSV file listings generated by FTK [See **Figure 5**].

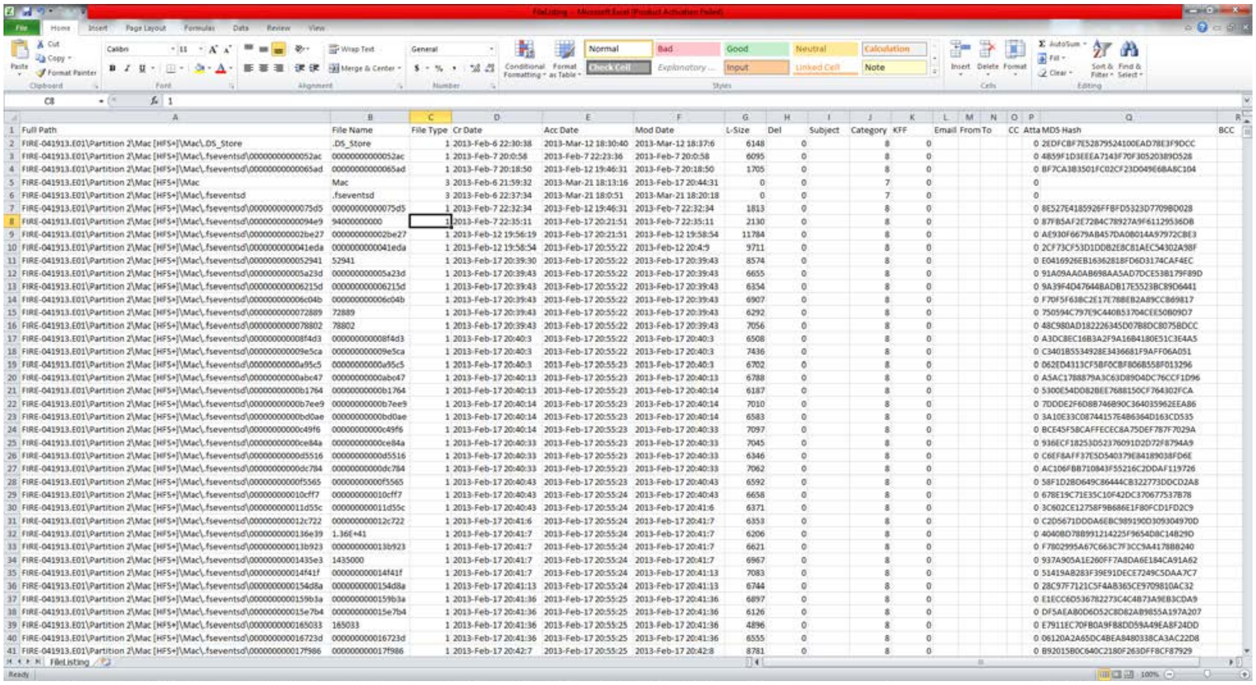
Figure 15



Name	Date modified	Type	Size
cvwcache	7/1/2013 11:44 AM	File folder	
DB	7/1/2013 8:56 AM	File folder	
dts_idx	7/12/2013 9:49 AM	File folder	
Jobs	7/8/2013 8:42 AM	File folder	
JT	7/12/2013 8:55 AM	File folder	
thumbnails	7/1/2013 9:02 AM	File folder	
1001.lock	7/1/2013 8:59 AM	LOCK File	0 KB
EvidenceHistory	7/12/2013 9:49 AM	Text Document	7 KB
Faulted.bin	7/1/2013 8:59 AM	BIN File	0 KB
FileListing	7/12/2013 9:49 AM	Microsoft Excel C...	135,726 KB
FileListing	7/12/2013 9:10 AM	Chrome HTML Do...	262,155 KB
Owner_Info	7/1/2013 8:57 AM	XML Document	1 KB
proc_mgr_file	7/1/2013 8:57 AM	Text Document	1 KB

The CSV report can be opened with Microsoft Excel and formatted as desired [See **Figure 6**].

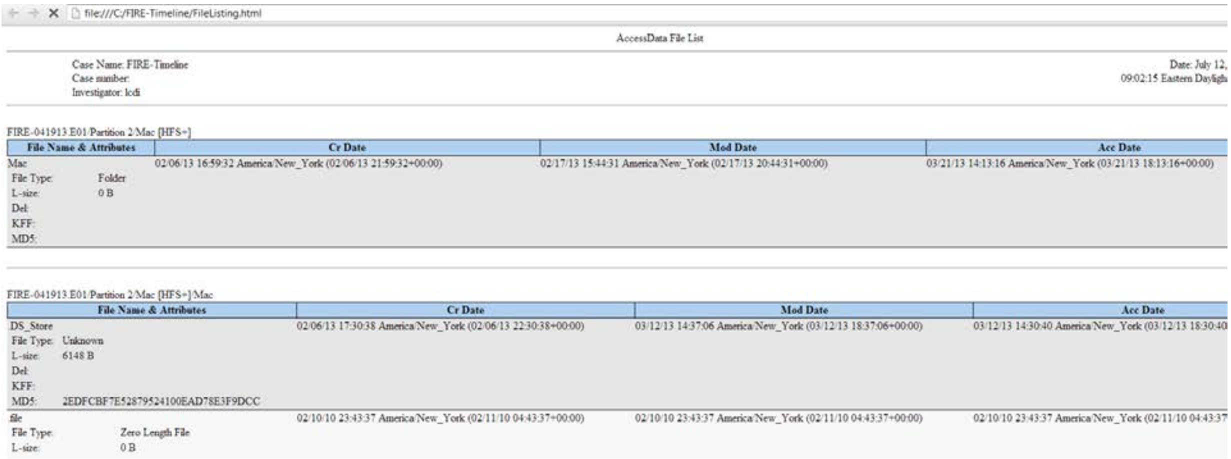
Figure 16



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
1	Full Path	File Name	File Type	Cr Date	Acc Date	Mod Date	L-Size	Del	Subject	Category	KFF	Email	From	To	CC	Atta	MD5 Hash	BCC
2	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.DS_Store	.DS_Store	1	2013-Feb-6 22:30:38	2013-Mar-12 18:30:40	2013-Mar-12 18:37:6	6148	0		8	0					0	2EDFCBF7E52879524100EAD78E3F9DCC	
3	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000052ac	00000000000052ac	1	2013-Feb-7 20:0:58	2013-Feb-7 22:23:36	2013-Feb-7 20:0:58	6095	0		8	0					0	4B59F1D3EEEA7143F70F30520389D528	
4	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000065ad	00000000000065ad	1	2013-Feb-7 20:18:50	2013-Feb-12 19:46:31	2013-Feb-7 20:18:50	1705	0		8	0					0	BF7CA3B3501FC02CF23D049E6BA8C104	
5	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac	Mac	3	2013-Feb-6 21:59:32	2013-Mar-21 18:13:16	2013-Feb-17 20:44:31	0	0		7	0					0		
6	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd	.fseventsd	3	2013-Feb-6 22:37:34	2013-Mar-21 18:0:51	2013-Mar-21 18:20:18	0	0		7	0					0		
7	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000075d5	00000000000075d5	1	2013-Feb-7 22:32:34	2013-Feb-12 19:46:31	2013-Feb-7 22:32:34	1813	0		8	0					0	8E527E4185926FFBFD5323D7709BD028	
8	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000094e9	9400000000	1	2013-Feb-7 22:35:11	2013-Feb-17 20:21:51	2013-Feb-7 22:35:11	2130	0		8	0					0	87FB5AF2E7284C78927A9F61129536DB	
9	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000002be27	000000000002be27	1	2013-Feb-12 19:56:19	2013-Feb-17 20:21:51	2013-Feb-12 19:58:54	11784	0		8	0					0	AE930F6679AB457DA0B014A97972CBE3	
10	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000041eda	0000000000041eda	1	2013-Feb-12 19:58:54	2013-Feb-17 20:55:22	2013-Feb-12 20:4:9	9711	0		8	0					0	2CF73CF53D1DDB2E8C81AEC54302A98F	
11	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000052941	52941	1	2013-Feb-17 20:39:30	2013-Feb-17 20:55:22	2013-Feb-17 20:39:43	8574	0		8	0					0	E0416926EB16362818FD6D3174CAF4EC	
12	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000005a23d	000000000005a23d	1	2013-Feb-17 20:39:43	2013-Feb-17 20:55:22	2013-Feb-17 20:39:43	6655	0		8	0					0	91A09AA0AB698AA5AD7DCE53B179F89D	
13	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000006215d	000000000006215d	1	2013-Feb-17 20:39:43	2013-Feb-17 20:55:22	2013-Feb-17 20:39:43	6354	0		8	0					0	9A39F4D47644BADB17E5523BC89D6441	
14	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000006c04b	000000000006c04b	1	2013-Feb-17 20:39:43	2013-Feb-17 20:55:22	2013-Feb-17 20:39:43	6907	0		8	0					0	F70F5F638C2E17E78BEB2A89CCB69817	
15	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000072889	72889	1	2013-Feb-17 20:39:43	2013-Feb-17 20:55:22	2013-Feb-17 20:39:43	6292	0		8	0					0	750594C797E9C440B53704CEE50B09D7	
16	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000078802	78802	1	2013-Feb-17 20:39:43	2013-Feb-17 20:55:22	2013-Feb-17 20:39:43	7056	0		8	0					0	48C980AD182226345D07B8DC8075BDCC	
17	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000008f4d3	000000000008f4d3	1	2013-Feb-17 20:40:3	2013-Feb-17 20:55:22	2013-Feb-17 20:40:3	6508	0		8	0					0	A3DC8EC16B3A2F9A16B4180E51C3E4A5	
18	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000009e5ca	000000000009e5ca	1	2013-Feb-17 20:40:3	2013-Feb-17 20:55:22	2013-Feb-17 20:40:3	7436	0		8	0					0	C3401B5534928E3436681F9AFF06A051	
19	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000a95c5	00000000000a95c5	1	2013-Feb-17 20:40:3	2013-Feb-17 20:55:23	2013-Feb-17 20:40:3	6702	0		8	0					0	062ED4313CF5BF0CBF806B558F013296	
20	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000abc47	00000000000abc47	1	2013-Feb-17 20:40:13	2013-Feb-17 20:55:23	2013-Feb-17 20:40:13	6788	0		8	0					0	A5AC1788879A3C63D89D4DC76CCF1D96	
21	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000b1764	00000000000b1764	1	2013-Feb-17 20:40:13	2013-Feb-17 20:55:23	2013-Feb-17 20:40:14	6187	0		8	0					0	5300E54DD82BEE7688150CF764302FCA	
22	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000b7ee9	00000000000b7ee9	1	2013-Feb-17 20:40:14	2013-Feb-17 20:55:23	2013-Feb-17 20:40:14	7010	0		8	0					0	7DDDE2F6D8B746B90C364035962EEA86	
23	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000bd0ae	00000000000bd0ae	1	2013-Feb-17 20:40:14	2013-Feb-17 20:55:23	2013-Feb-17 20:40:14	6583	0		8	0					0	3A10E33C08744157E4B6364D163CD535	
24	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000c49f6	00000000000c49f6	1	2013-Feb-17 20:40:14	2013-Feb-17 20:55:23	2013-Feb-17 20:40:33	7097	0		8	0					0	BCE45F58CAFFECEC8A75DEF787F7029A	
25	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000ce84a	00000000000ce84a	1	2013-Feb-17 20:40:33	2013-Feb-17 20:55:23	2013-Feb-17 20:40:33	7045	0		8	0					0	936ECF18253D52376091D2D72F8794A9	
26	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000d5516	00000000000d5516	1	2013-Feb-17 20:40:33	2013-Feb-17 20:55:23	2013-Feb-17 20:40:33	6346	0		8	0					0	C6EF8AFF37E5D540379E84189038FD6E	
27	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000dc784	00000000000dc784	1	2013-Feb-17 20:40:33	2013-Feb-17 20:55:23	2013-Feb-17 20:40:33	7062	0		8	0					0	AC106FBB710843F55216C2DDAF119726	
28	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000000f5565	00000000000f5565	1	2013-Feb-17 20:40:43	2013-Feb-17 20:55:23	2013-Feb-17 20:40:43	6592	0		8	0					0	58F1D2BD649C86444CB322773DDCD2A8	
29	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000010cff7	000000000010cff7	1	2013-Feb-17 20:40:43	2013-Feb-17 20:55:24	2013-Feb-17 20:40:43	6658	0		8	0					0	678E19C71E35C10F42DC370677537B78	
30	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000011d55c	000000000011d55c	1	2013-Feb-17 20:40:43	2013-Feb-17 20:55:24	2013-Feb-17 20:41:6	6371	0		8	0					0	3C602CE12758F9B686E1F80FCD1FD2C9	
31	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000012c722	000000000012c722	1	2013-Feb-17 20:41:6	2013-Feb-17 20:55:24	2013-Feb-17 20:41:7	6353	0		8	0					0	C2D5671DDDA6EBC989190D309304970D	
32	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000136e39	1.36E+41	1	2013-Feb-17 20:41:7	2013-Feb-17 20:55:24	2013-Feb-17 20:41:7	6206	0		8	0					0	40408D788991214225F9654D8C14829D	
33	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000013b923	000000000013b923	1	2013-Feb-17 20:41:7	2013-Feb-17 20:55:24	2013-Feb-17 20:41:7	6621	0		8	0					0	F7802995A67C663C7F3CC9A41788B240	
34	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\00000000001435e3	1435000	1	2013-Feb-17 20:41:7	2013-Feb-17 20:55:24	2013-Feb-17 20:41:7	6967	0		8	0					0	937A905A1E260FF7A8DA6E184CA91A62	
35	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000014f41f	000000000014f41f	1	2013-Feb-17 20:41:7	2013-Feb-17 20:55:24	2013-Feb-17 20:41:13	7083	0		8	0					0	51419AB283F39E91DECE7249C5DAA7C7	
36	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000154d8a	0000000000154d8a	1	2013-Feb-17 20:41:13	2013-Feb-17 20:55:24	2013-Feb-17 20:41:13	6744	0		8	0					0	28C97F7121C5F4AB365CE9709810AC32	
37	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000159b3a	0000000000159b3a	1	2013-Feb-17 20:41:36	2013-Feb-17 20:55:25	2013-Feb-17 20:41:36	6897	0		8	0					0	E1ECC6D536782273C4C4B73A9EB3CDA9	
38	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000015e7b4	000000000015e7b4	1	2013-Feb-17 20:41:36	2013-Feb-17 20:55:25	2013-Feb-17 20:41:36	6126	0		8	0					0	DF5AEA80D6D52C8D82A89855A197A207	
39	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\0000000000165033	165033	1	2013-Feb-17 20:41:36	2013-Feb-17 20:55:25	2013-Feb-17 20:41:36	4896	0		8	0					0	E7911EC70FB0A9F88DD59A49EA8F24DD	
40	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000016723d	000000000016723d	1	2013-Feb-17 20:41:36	2013-Feb-17 20:55:25	2013-Feb-17 20:41:36	6555	0		8	0					0	06120A2A65DC4BEA8480338CA3AC22D8	
41	FIRE-041913.E01\Partition 2\Mac [HFS+]\Mac\.fseventsd\000000000017f986	000000000017f986	1	2013-Feb-17 20:42:7	2013-Feb-17 20:55:25	2013-Feb-17 20:42:8	8781	0		8	0					0	B92015B0C640C2180F263DFF8CF87929	

The HTML document can be opened with any web browser [See **Figure 7**].

Figure 17



AccessData File List

Case Name: FIRE-Timeline
Case number:
Investigator: lcdi

Date: July 12,
09:02:15 Eastern Dayligh

FIRE-041913.E01/Partition 2/Mac [HFS+]

File Name & Attributes	Cr Date	Mod Date	Acc Date
Mac File Type: Folder L-size: 0 B Del: KFF: MD5:	02/06/13 16:59:32 America/New_York (02/06/13 21:59:32+00:00)	02/17/13 15:44:31 America/New_York (02/17/13 20:44:31+00:00)	03/21/13 14:13:16 America/New_York (03/21/13 18:13:16+00:00)

FIRE-041913.E01/Partition 2/Mac [HFS+]/Mac

File Name & Attributes	Cr Date	Mod Date	Acc Date
DS_Store File Type: Unknown L-size: 6148 B Del: KFF: MD5: 2EDFCBF7E52879524100EAD78E3F9DCC	02/06/13 17:30:38 America/New_York (02/06/13 22:30:38+00:00)	03/12/13 14:37:06 America/New_York (03/12/13 18:37:06+00:00)	03/12/13 14:30:40 America/New_York (03/12/13 18:30:40
file File Type: Zero Length File L-size: 0 B	02/10/10 23:43:37 America/New_York (02/11/10 04:43:37+00:00)	02/10/10 23:43:37 America/New_York (02/11/10 04:43:37+00:00)	02/10/10 23:43:37 America/New_York (02/11/10 04:43:37

It is important to note that both report types contain file names & attributes, such as MD5 and logical sizes, along with created/modified/accessed dates for each entry found within the FTK 4.1 case for the evidence.

Log2Timeline Guide for TAPEWORM

Written by: Chapin Bryce

Updated: July, 2013

The Senator Patrick Leahy Center for Digital Investigation

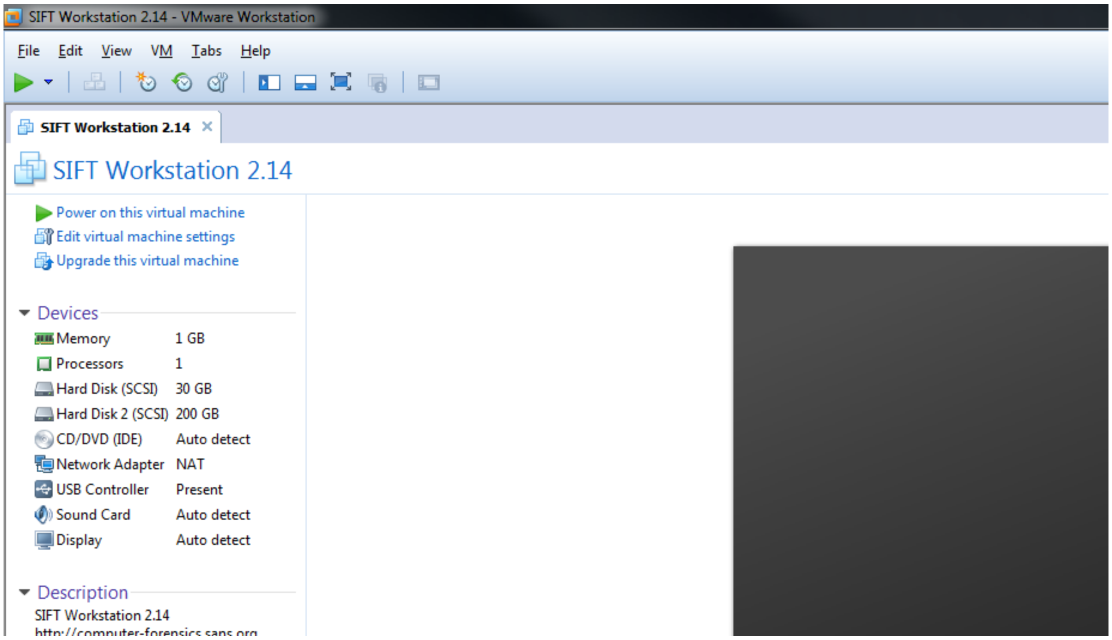
Champlain College

Table of Contents

4 Timeline Creation and Analysis with Log2Timeline in SIFT 2.14

4.1 Configuring SIFT for Log2Timeline

SIFT 2.14 can be downloaded after registering for an account. A custom download link will be made available, and the .zip file containing the preconfigured virtual appliance can be downloaded at that point. Once the download is completed and the files are extracted from the archive, the virtual machine (VM) can be opened in VMWare, or another virtualization product. See **Figure 1** for an example of SIFT opened in VMWare Workstation 9.


SIFT Workstation 2.14 - VMware Workstation
File
Edit
View
VM
Tabs
Help
SIFT Workstation 2.14
Power on this virtual machine
Edit virtual machine settings
Upgrade this virtual machine

SIFT Workstation 2.14

Devices

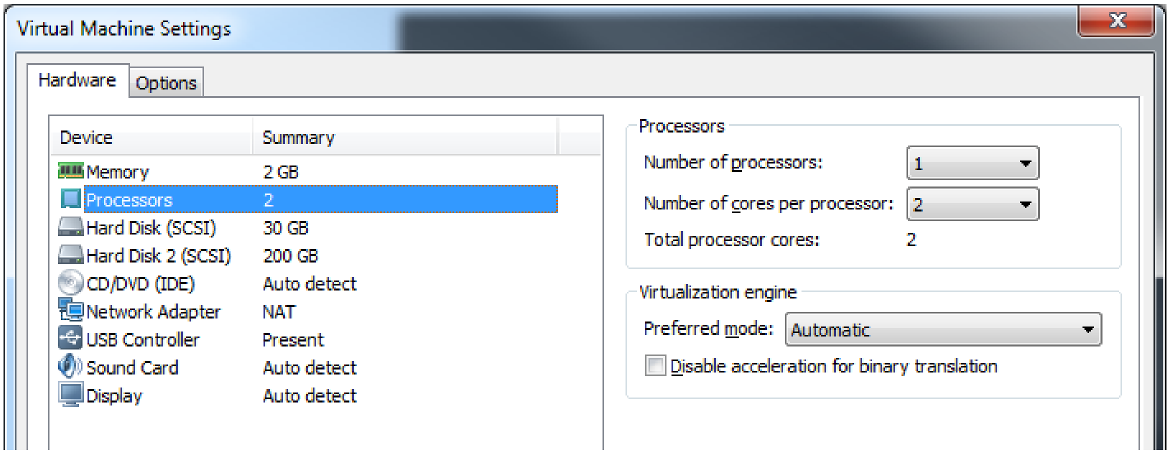
Memory	1 GB
Processors	1
Hard Disk (SCSI)	30 GB
Hard Disk 2 (SCSI)	200 GB
CD/DVD (IDE)	Auto detect
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto detect
Display	Auto detect

Description

SIFT Workstation 2.14
http://computer-forensics.sans.org

Figure 18 – SIFT VM unzipped and ready for use

The minimum specifications are preconfigured into this VM(**Figure 1**), but can be altered if more memory or processing is available. **Figure 2** shows increased allocations for the SIFT VM.



Device	Summary
Memory	2 GB
Processors	2
Hard Disk (SCSI)	30 GB
Hard Disk 2 (SCSI)	200 GB
CD/DVD (IDE)	Auto detect
Network Adapter	NAT
USB Controller	Present
Sound Card	Auto detect
Display	Auto detect

Processors

Number of processors: 1

Number of cores per processor: 2

Total processor cores: 2

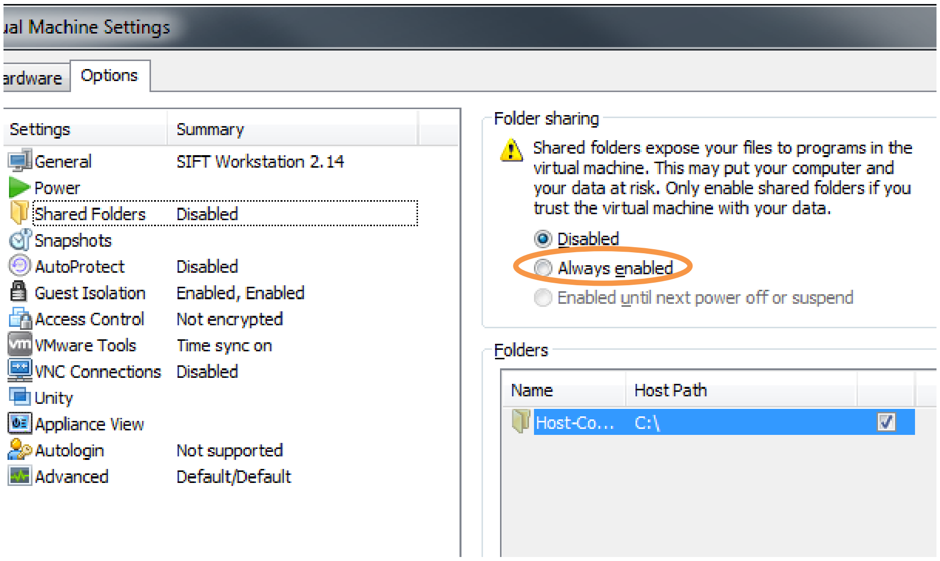
Virtualization engine

Preferred mode: Automatic

□ Disable acceleration for binary translation

Figure 19 – Configuring the Virtual Machine

Additionally, you can adjust the virtual machine settings to enable shared folders between the host and guest machines. This allows for the virtual machine to read exhibits from the host machine and to export the output to the host computer. How to enable this feature is shown in **Figure 3.**



ual Machine Settings

ardware | Options

Settings	Summary
General	SIFT Workstation 2.14
Power	
Shared Folders	Disabled
Snapshots	
AutoProtect	Disabled
Guest Isolation	Enabled, Enabled
Access Control	Not encrypted
VMware Tools	Time sync on
VNC Connections	Disabled
Unity	
Appliance View	
Autologin	Not supported
Advanced	Default/Default

Folder sharing

Shared folders expose your files to programs in the virtual machine. This may put your computer and your data at risk. Only enable shared folders if you trust the virtual machine with your data.

● Disabled
○ Always enabled
○ Enabled until next power off or suspend

Folders

Name	Host Path	
Host-Co...	C:\	☑

Figure 20 – Enable Shared Folders

With shared folders enabled, the virtual machine can be powered on, and SIFT is ready for use on the workstation. The login screen will appear, prompting for the password (forensics) before granting access (see **Figure 4**).

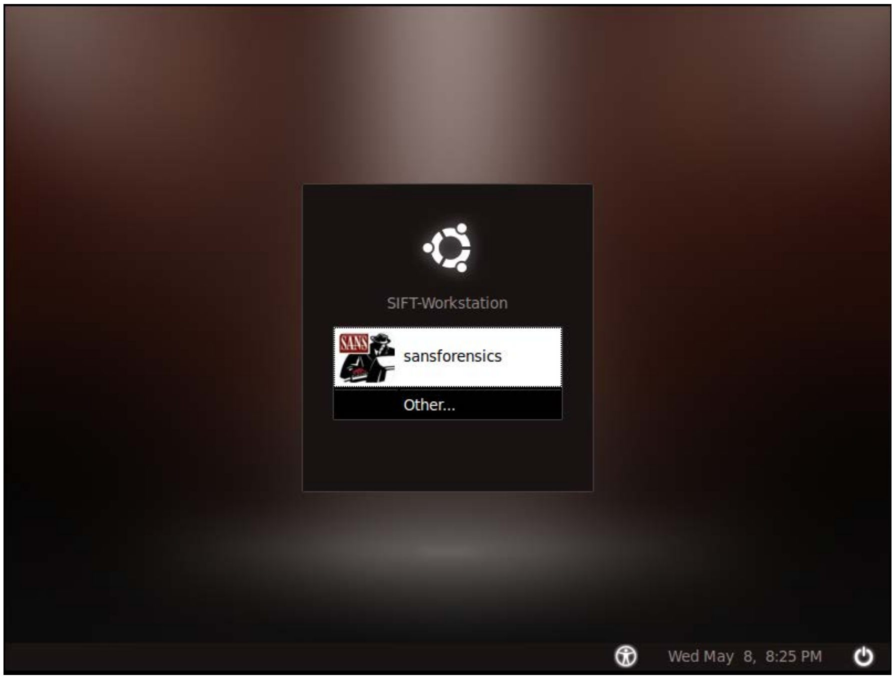
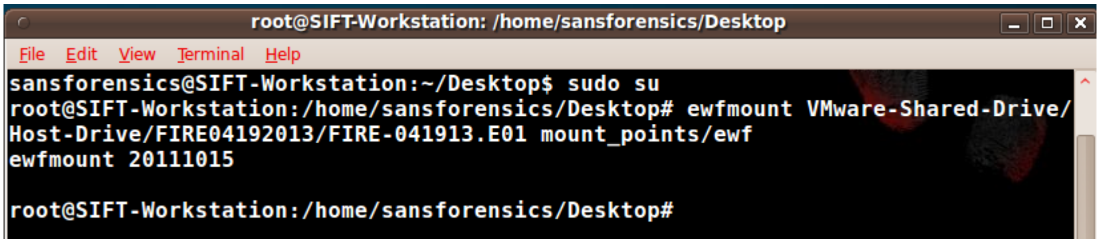


Figure 21 – SIFT 2.14 login window

4.2 Preparing Exhibits for Log2Timeline in SIFT

Once a user is logged in, SIFT's desktop appears and presents a terminal window. Since the shared folders have been enabled before startup, the exhibits saved on the local machine are accessible via the virtual machine, with the opposite being true as well. Exhibits can be acquired in a few different formats, including E01 and DD (Raw). There are a number of different methods for readying these acquisitions for use. Since the acquisition is a block file, it has to be mounted so the data inside of it can be read as a block device[2]. The E01 (EnCase evidence file) format is extremely popular, as it allows for compression, encryption, and is compatible with most forensic tools. To convert the E01 for use in SIFT the **ewfmount** command must be run. This command will convert the source E01 file (**VMware-Shared-Drive/Host-Drive/FIRE04192013/FIRE-041913.E01**) to a mount point where it can be interpreted as an uncompressed block file (**mount_points/ewf**) (**Figure 5**).

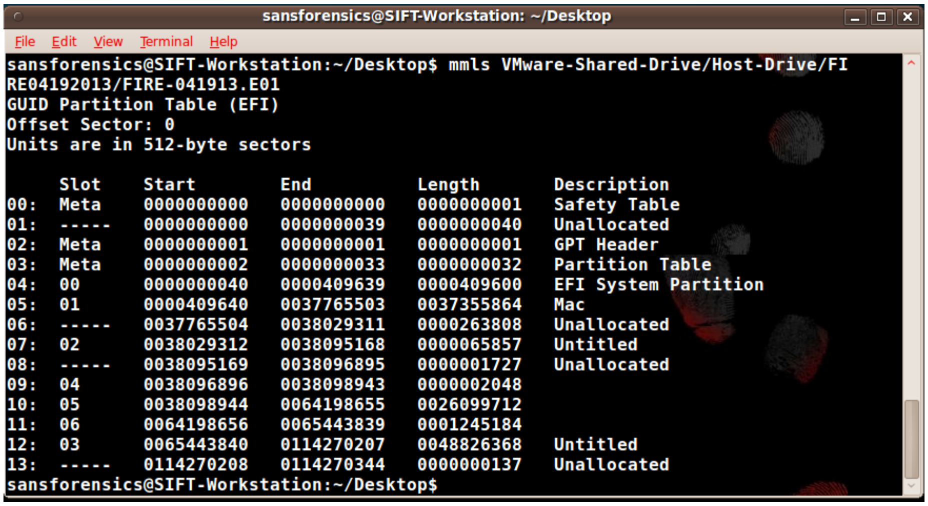

root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help

```
sansforensics@SIFT-Workstation:~/Desktop$ sudo su
root@SIFT-Workstation:/home/sansforensics/Desktop# ewfmount VMware-Shared-Drive/
Host-Drive/FIRE04192013/FIRE-041913.E01 mount_points/ewf
ewfmount 20111015

root@SIFT-Workstation:/home/sansforensics/Desktop#
```

Figure 22 – Using ewfmount to prepare the E01 file for use with Log2Timeline

If the exhibit is Microsoft Windows only, skip to the next section to the section about log2timline-sift which is Windows only. Drives with partitions other than NTFS (Microsoft Windows) need further processing before use with Log2Timeline. To ensure that Macintosh or Linux partitions are completely processed, they must be fully mounted. Once the **ewfmount** command is run, the individual partitions must be mounted from the newly converted block file. SIFT also has the **mmls** utility bundle, allowing for partition structures to be viewed.


sansforensics@SIFT-Workstation: ~/Desktop
File Edit View Terminal Help

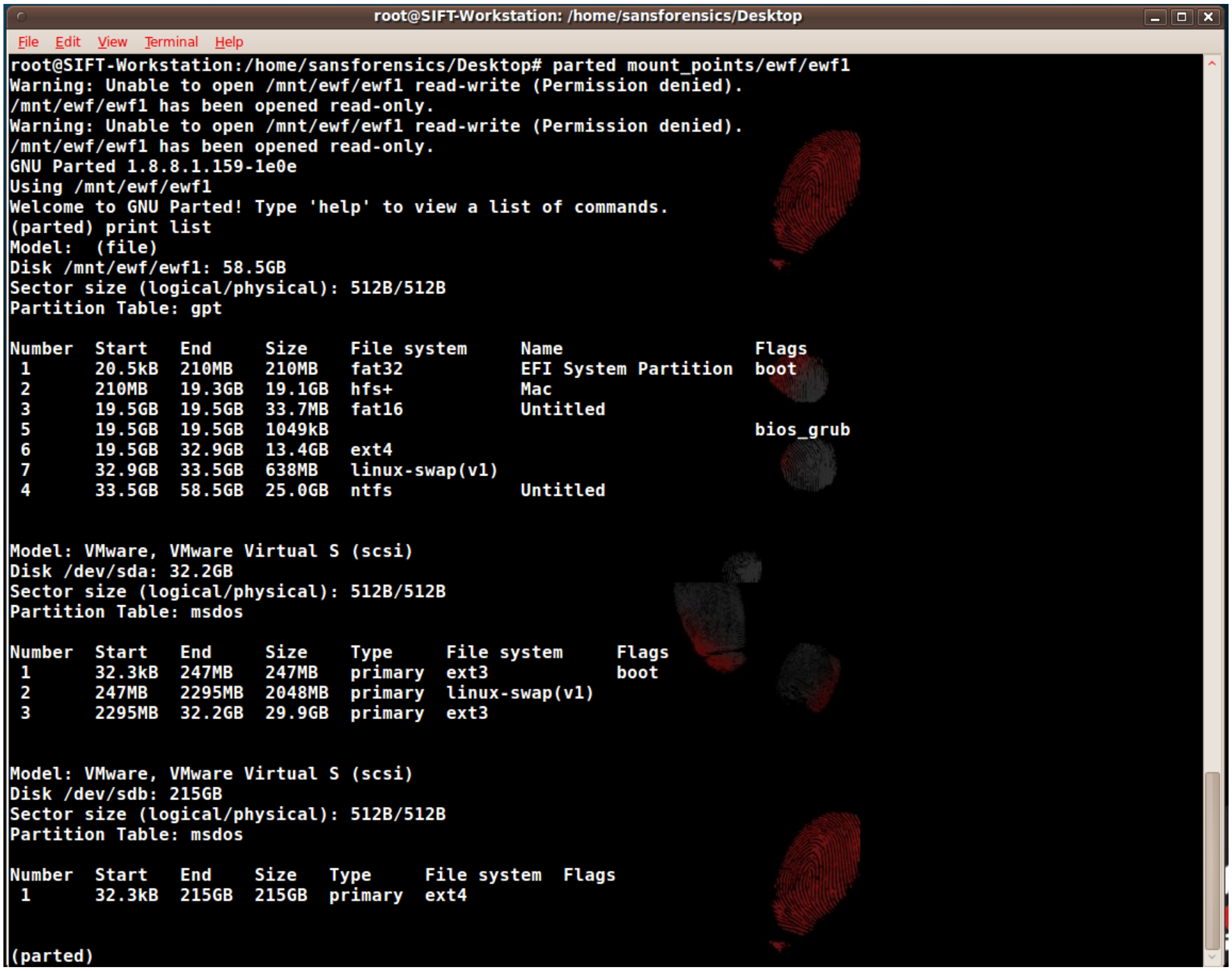
```
sansforensics@SIFT-Workstation:~/Desktop$ mmls VMware-Shared-Drive/Host-Drive/FI
RE04192013/FIRE-041913.E01
GUID Partition Table (EFI)
Offset Sector: 0
Units are in 512-byte sectors

     Slot    Start        End          Length       Description
00:  Meta    0000000000   0000000000   0000000001   Safety Table
01:  -----   0000000000   0000000039   0000000040   Unallocated
02:  Meta    0000000001   0000000001   0000000001   GPT Header
03:  Meta    0000000002   0000000033   0000000032   Partition Table
04:  00      0000000040   0000409639   0000409600   EFI System Partition
05:  01      0000409640   0037765503   0037355864   Mac
06:  -----   0037765504   0038029311   0000263808   Unallocated
07:  02      0038029312   0038095168   0000065857   Untitled
08:  -----   0038095169   0038096895   0000001727   Unallocated
09:  04      0038096896   0038098943   0000002048
10:  05      0038098944   0064198655   0026099712
11:  06      0064198656   0065443839   0001245184
12:  03      0065443840   0114270207   0048826368   Untitled
13:  -----   0114270208   0114270344   0000000137   Unallocated
sansforensics@SIFT-Workstation:~/Desktop$
```

Figure 23 – MMLS partition table listing

[2] For more information and references on block files and loop devices: http://wiki.osdev.org/Loopback_Device

Another command that provides essential information when preparing the block file to be mounted is the **parted** framework. This framework allows the partition information to be read from a block file. **Parted** shows information regarding the specific file systems in place on the block file as well as byte information, as opposed to sector information provided my **mmls (Figure 7).**


root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help

```
root@SIFT-Workstation:/home/sansforensics/Desktop# parted mount_points/ewf/ewf1
Warning: Unable to open /mnt/ewf/ewf1 read-write (Permission denied).
/mnt/ewf/ewf1 has been opened read-only.
Warning: Unable to open /mnt/ewf/ewf1 read-write (Permission denied).
/mnt/ewf/ewf1 has been opened read-only.
GNU Parted 1.8.8.1.159-1e0e
Using /mnt/ewf/ewf1
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) print list
Model:  (file)
Disk /mnt/ewf/ewf1: 58.5GB
Sector size (logical/physical): 512B/512B
Partition Table: gpt

Number  Start   End     Size    File system     Name                  Flags
 1      20.5kB  210MB   210MB   fat32           EFI System Partition  boot
 2      210MB   19.3GB  19.1GB  hfs+            Mac
 3      19.5GB  19.5GB  33.7MB  fat16           Untitled
 5      19.5GB  19.5GB  1049kB                                        bios_grub
 6      19.5GB  32.9GB  13.4GB  ext4
 7      32.9GB  33.5GB  638MB   linux-swap(v1)
 4      33.5GB  58.5GB  25.0GB  ntfs            Untitled


Model: VMware, VMware Virtual S (scsi)
Disk /dev/sda: 32.2GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos

Number  Start   End     Size    Type     File system     Flags
 1      32.3kB  247MB   247MB   primary  ext3            boot
 2      247MB   2295MB  2048MB  primary  linux-swap(v1)
 3      2295MB  32.2GB  29.9GB  primary  ext3


Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdb: 215GB
Sector size (logical/physical): 512B/512B
Partition Table: msdos

Number  Start   End    Size   Type     File system  Flags
 1      32.3kB  215GB  215GB  primary  ext4


(parted)
```

Figure 24 – Linux Parted command providing more partition information

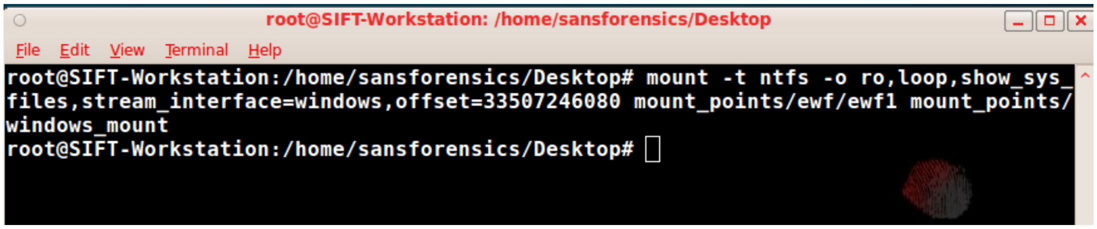
To convert the sector offsets listed by **mmls** to the byte offset needed for the mount command, the starting sector should be multiplied by the Bytes per sector (See **Table 1** for an example conversion).

Starting Sector x Bytes/Sector = Offset in Bytes Total Sectors x Bytes/Sector = Bytes per Slot Bytes/sector = 512		
'Slot 01' is HFS+ (Macintosh)		
Start: 0000409640	End:003776503	Total Sectors:0037355864
0000409640 x 512 = 209735680 bytes from beginning of physical drive		0037355864 x 512 = 19126202368 bytes or ~17.812 GB
'Slot 03' is NTFS (Windows)		
Start: 0065443840	End: 014270207	Total Sectors: 0048826368

0065443840 x 512 = 33507246080 bytes from beginning of physical drive		0048826368 x 512 = 24999100416 or ~23.28 GB
'Slot 05' is EXT4 (Linux)		
Start: 0038098944	End: 0064198655	Total Sectors: 0026099712
0038098944 x 512 = 19506659328 bytes from beginning of physical drive		0026099712 x 512 = 13363052544 or ~12.45GB

Table 1 – Converting mmls sector offset to byte offset.

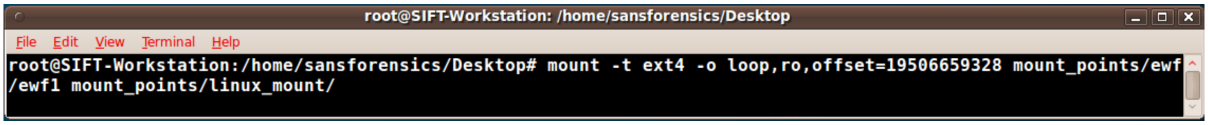
Now that the partitions within the block file have been identified, they can be individually mounted. Using the Linux mount command, the Windows partition is mounted to a pre-made directory illustrated in **Figure 8.**



```
root@SIFT-Workstation:/home/sansforensics/Desktop# mount -t ntfs -o ro,loop,show_sys_
files,stream_interface=windows,offset=33507246080 mount_points/ewf/ewf1 mount_points/
windows_mount
root@SIFT-Workstation:/home/sansforensics/Desktop# 
```

Figure 25 – Mounting the Windows partition with the Linux mount command

This same mount process can be repeated for the offsets provided by the **mmls** conversions (**Figure 9** shows Linux mounting only).



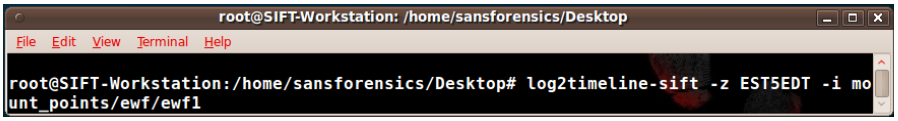
```
root@SIFT-Workstation:/home/sansforensics/Desktop# mount -t ext4 -o loop,ro,offset=19506659328 mount_points/ewf
/ewf1 mount_points/linux_mount/
```

Figure 26 – Mounting Linux partition with the Linux Mount Command.

Once the partitions are mounted, SIFT is able to run Log2Timeline against the exhibit. Skip to the section Running Log2Timeline to create a timeline from the new mount point.

4.3 Running Log2Timeline-SIFT

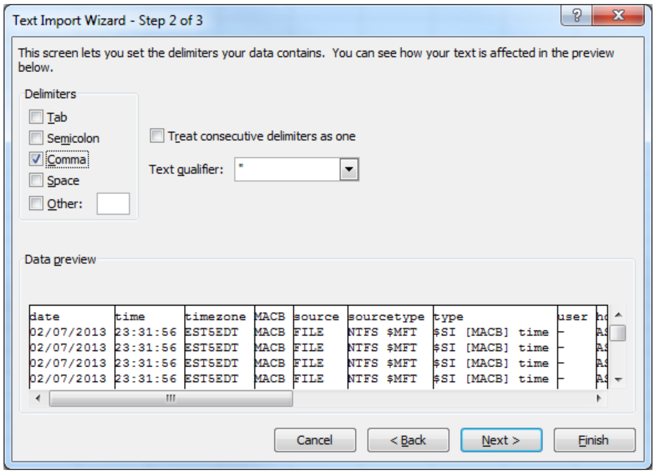
SIFT has created a custom version of Log2Timeline with the command **log2timeline-sift,** easing the process of mounting the exhibit and running the command against a Windows only partition. This automation selects the default options for log2timeline and begins processing. **Figure 10** is a capture of the command used to run **log2timeline-sift** against the exhibit. Note the command needs the timezone (-z) and image mount point (-i) from the **ewfmount** step (if the image was in a DD/raw format, select that file for use with **log2timeline-sift**).


root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help

```
root@SIFT-Workstation:/home/sansforensics/Desktop# log2timeline-sift -z EST5EDT -i mo
unt_points/ewf/ewf1
```

Figure 27 – Using log2timeline-sift to create a timeline for the Windows partition.

When the command is complete, the output is placed in the "cases" directory on the SIFT desktop. From here, the CSV file can be copied onto the host machine and opened with Microsoft Excel. Once the file is opened in Excel, the columns must be delimited, as Excel does not automatically parse the information into columns (See **Figure 11**). If it parses correctly, it will look similar to the worksheet shown in **Figure 12**.



This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

Delimiters

☐ Tab
☐ Semicolon
☑ Comma
☐ Space
☐ Other:

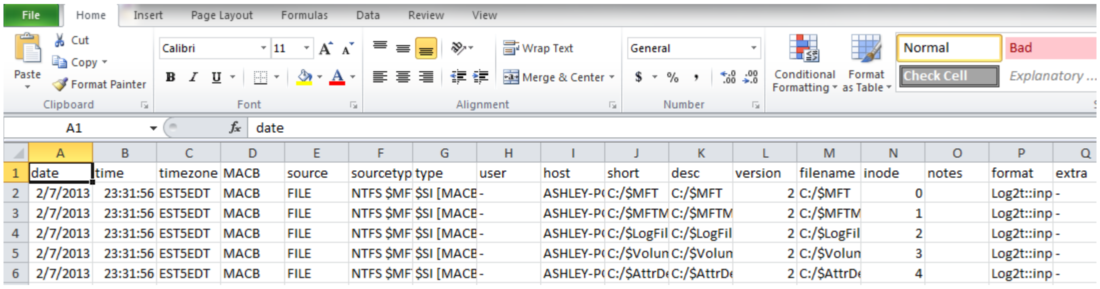
☐ Treat consecutive delimiters as one

Text qualifier: "

Data preview

```
date       time     timezone MACB source sourcetype type             user ho
02/07/2013 23:31:56 EST5EDT  MACB FILE   NTFS $MFT  $SI [MACB] time  -    AS
02/07/2013 23:31:56 EST5EDT  MACB FILE   NTFS $MFT  $SI [MACB] time  -    AS
02/07/2013 23:31:56 EST5EDT  MACB FILE   NTFS $MFT  $SI [MACB] time  -    AS
02/07/2013 23:31:56 EST5EDT  MACB FILE   NTFS $MFT  $SI [MACB] time  -    AS
```

Figure 28 – Delimiting CSV document in Excel



A1 date

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	date	time	timezone	MACB	source	sourcetyp	type	user	host	short	desc	version	filename	inode	notes	format	extra
2	2/7/2013	23:31:56	EST5EDT	MACB	FILE	NTFS $MF	$SI [MACE	-	ASHLEY-P(	C:/$MFT	C:/$MFT	2	C:/$MFT	0		Log2t::inp	-
3	2/7/2013	23:31:56	EST5EDT	MACB	FILE	NTFS $MF	$SI [MACE	-	ASHLEY-P(	C:/$MFTM	C:/$MFTM	2	C:/$MFTM	1		Log2t::inp	-
4	2/7/2013	23:31:56	EST5EDT	MACB	FILE	NTFS $MF	$SI [MACE	-	ASHLEY-P(	C:/$LogFil	C:/$LogFil	2	C:/$LogFil	2		Log2t::inp	-
5	2/7/2013	23:31:56	EST5EDT	MACB	FILE	NTFS $MF	$SI [MACE	-	ASHLEY-P(	C:/$Volun	C:/$Volun	2	C:/$Volun	3		Log2t::inp	-
6	2/7/2013	23:31:56	EST5EDT	MACB	FILE	NTFS $MF	$SI [MACE	-	ASHLEY-P(	C:/$AttrD(	C:/$AttrD(	2	C:/$AttrD(	4		Log2t::inp	-

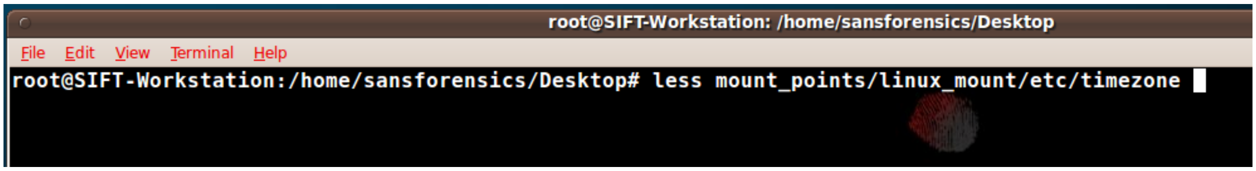
Figure 29 – Log2Timeline-SIFT output in Excel

4.4 Running Log2Timeline for Linux and Macintosh Partitions

The **log2timeline-sift** command automates actions for timeline creation with Windows partitions in SIFT. To create a timeline with log2timeline against other partition types, such as Linux and Macintosh, log2timeline has to be run manually. Please refer to the section on mounting for information on how to mount and prepare evidence for use with log2timeline.

4.4.1 Running Log2Timeline on a Linux Exhibit

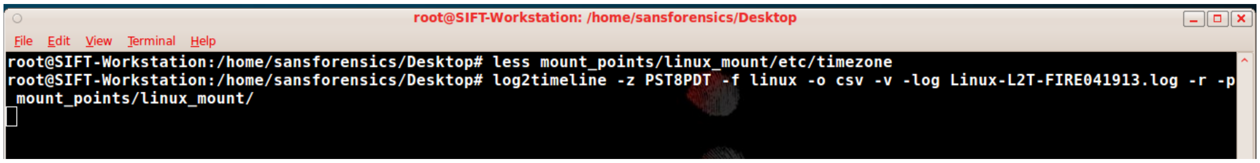
After the image has been mounted correctly and the partition has its own mount point, the time zone of the Linux partition should be verified. To do this, run the command **less** (directory path to Linux mount point)**/etc/timezone** as seen in **Figure 13**. The **less** command allows a file to be previewed in the command window. To exit the **less** windows press **q**.


root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help

```
root@SIFT-Workstation:/home/sansforensics/Desktop# less mount_points/linux_mount/etc/timezone
```

Figure 30 – Previewing the Time Zone Set for the Linux Partition

Once the command is run, the timezone stored in the /etc/timezone system file of the partition will be displayed and log2timeline can be run. The command to run log2timeline is **log2timeline –z** [exhibit time zone] **–f** [selected modules] **–o** [output format] **–v –log** [logfile output location] **–r –p** [Evidence/exhibit mount point location] (**Figure 14)**.

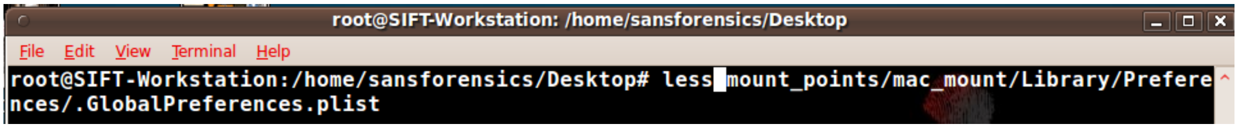

root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help

```
root@SIFT-Workstation:/home/sansforensics/Desktop# less mount_points/linux_mount/etc/timezone
root@SIFT-Workstation:/home/sansforensics/Desktop# log2timeline -z PST8PDT -f linux -o csv -v -log Linux-L2T-FIRE041913.log -r -p
 mount_points/linux_mount/
```

Figure 31 – Running Log2Timeline Manually Against Linux Partition

4.4.2 Running Log2Timeline on a Macintosh Exhibit

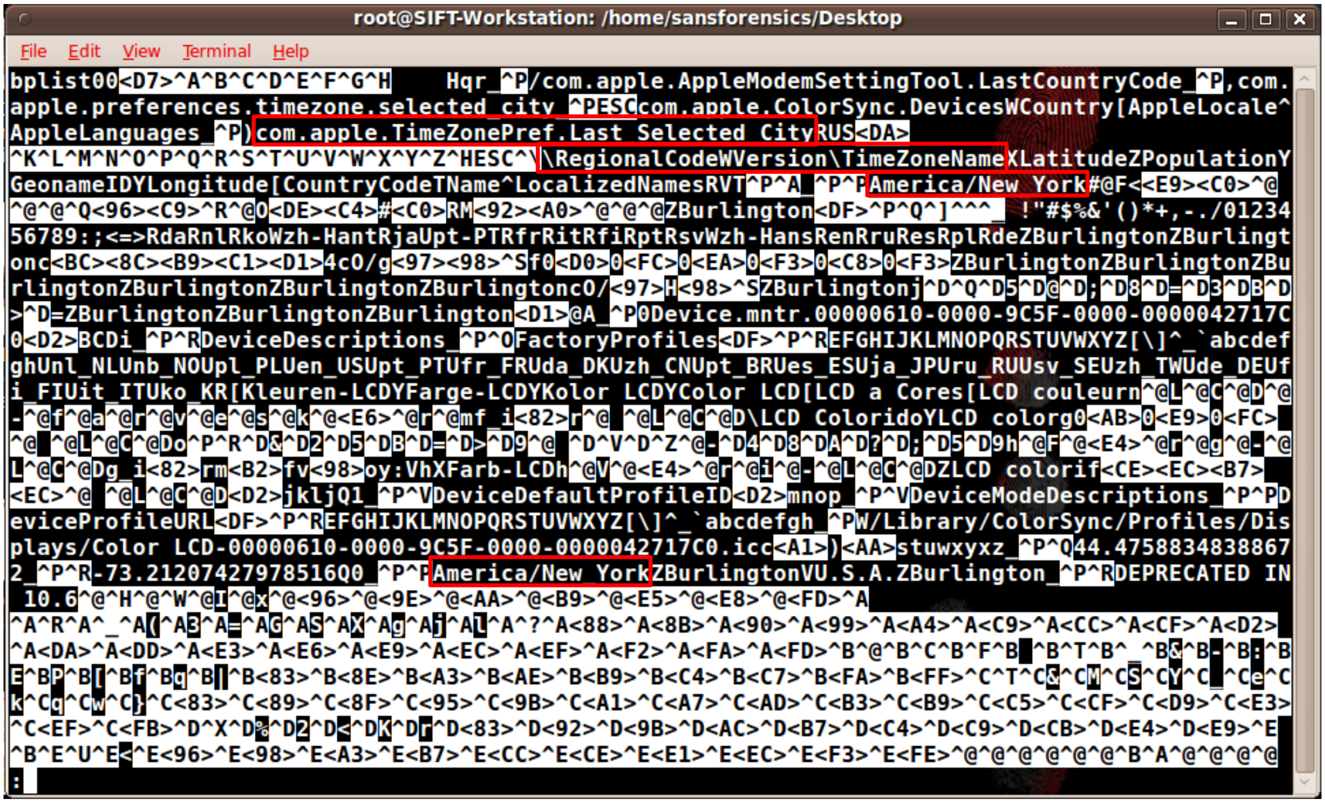
With Linux, most system information is stored in plain text, such as the timezone file referenced above. For Macintosh OSX, the timezone, along with other system information, is stored in a .plist (property list) file. Plist files are either XML or binary format, and they act similarly to the registry. The timezone information is saved in binary format on the Macintosh OSX in the GlobalPreference.plist. Plist files aremore difficult for most text editors to interpret, despite the key information being in plain text. To read the information from the plist file, use the command **less** [path to mount point]**/Library/Preferences/.GlobalPreferences.plist** (**Figure 15)**.


root@SIFT-Workstation: /home/sansforensics/Desktop
File Edit View Terminal Help

```
root@SIFT-Workstation:/home/sansforensics/Desktop# less mount_points/mac_mount/Library/Prefere
nces/.GlobalPreferences.plist
```

Figure 32 – Previewing the Time Zone Set for the Macintosh Partition

The text highlighted in **Figure 16** suggests the timezone is America/New_York, known as EST5EDT in log2timeline.



```
bplist00<D7>^A^B^C^D^E^F^G^H    Hqr_^P/com.apple.AppleModemSettingTool.LastCountryCode_^P,com.
apple.preferences.timezone.selected_city_^PESCcom.apple.ColorSync.DevicesWCountry[AppleLocale^
AppleLanguages_^P)com.apple.TimeZonePref.Last_Selected_CityRUS<DA>
^K^L^M^N^O^P^Q^R^S^T^U^V^W^X^Y^Z^HESC^\\RegionalCodeWVersion\TimeZoneNameXLatitudeZPopulationY
GeonameIDYLongitude[CountryCodeTName^LocalizedNamesRVT^P^A_^P^PAmerica/New_York#@F<<E9><C0>^@
^@^@^Q<96><C9>^R^@0<DE><C4>#<C0>RM<92><A0>^@^@^@ZBurlington<DF>^P^Q^]^^^_ !"#$%&'()*+,-./01234
56789:;<=>RdaRnlRkoWzh-HantRjaUpt-PTRfrRitRfiRptRsvWzh-HansRenRruResRplRdeZBurlingtonZBurlingt
onc<BC><8C><B9><C1><D1>4c0/g<97><98>^Sf0<D0>0<FC>0<EA>0<F3>0<C8>0<F3>ZBurlingtonZBurlingtonZBu
rlingtonZBurlingtonZBurlingtonZBurlingtonc0/<97>H<98>^SZBurlingtonj^D^Q^D5^D@^D;^D8^D=^D3^DB^D
>^D=ZBurlingtonZBurlingtonZBurlington<D1>@A_^P0Device.mntr.00000610-0000-9C5F-0000-000042717C
0<D2>BCDi_^P^RDeviceDescriptions_^P^OFactoryProfiles<DF>^P^REFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdef
ghUnl_NLUnb_NOUpl_PLUen_USUpt_PTUfr_FRUda_DKUzh_CNUpt_BRUes_ESUja_JPUru_RUUsv_SEUzh_TWUde_DEUf
i_FIUit_ITUko_KR[Kleuren-LCDYFarge-LCDYKolor LCDYColor LCD[LCD a Cores[LCD couleurn^@L^@C^@D^@
-^@f^@a^@r^@v^@e^@s^@k^@<E6>^@r^@mf_i<82>r^@ ^@L^@C^@D\LCD ColoridoYLCD colorg0<AB>0<E9>0<FC>
^@ ^@L^@C^@Do^P^R^D&^D2^D5^DB^D=^D>^D9^@ ^D^V^D^Z^@-^D4^D8^DA^D?^D;^D5^D9h^@F^@<E4>^@r^@g^@-^@
L^@C^@Dg_i<82>rm<B2>fv<98>oy:VhXFarb-LCDh^@V^@<E4>^@r^@i^@-^@L^@C^@DZLCD colorif<CE><EC><B7>
<EC>^@ ^@L^@C^@D<D2>jkljQ1_^P^VDeviceDefaultProfileID<D2>mnop_^P^VDeviceModeDescriptions_^P^PD
eviceProfileURL<DF>^P^REFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefgh_^PW/Library/ColorSync/Profiles/Dis
plays/Color LCD-00000610-0000-9C5F-0000-000042717C0.icc<A1>)<AA>stuwxyxz_^P^Q44.4758834838867
2_^P^R-73.21207427978516Q0_^P^PAmerica/New_YorkZBurlingtonVU.S.A.ZBurlington_^P^RDEPRECATED IN
 10.6^@^H^@^W^@I^@x^@<96>^@<9E>^@<AA>^@<B9>^@<E5>^@<E8>^@<FD>^A
^A^R^A^_^A(^A3^A=^AG^AS^AX^Ag^Aj^Al^A^?^A<88>^A<8B>^A<90>^A<99>^A<A4>^A<C9>^A<CC>^A<CF>^A<D2>
^A<DA>^A<DD>^A<E3>^A<E6>^A<E9>^A<EC>^A<EF>^A<F2>^A<FA>^A<FD>^B^@^B^C^B^F^B ^B^T^B^_^B&^B-^B:^B
E^BP^B[^Bf^Bq^B|^B<83>^B<8E>^B<A3>^B<AE>^B<B9>^B<C4>^B<C7>^B<FA>^B<FF>^C^T^C&^CM^CS^CY^C_^Ce^C
k^Cq^Cw^C}^C<83>^C<89>^C<8F>^C<95>^C<9B>^C<A1>^C<A7>^C<AD>^C<B3>^C<B9>^C<C5>^C<CF>^C<D9>^C<E3>
^C<EF>^C<FB>^D^X^D%^D2^D<^DK^Dr^D<83>^D<92>^D<9B>^D<AC>^D<B7>^D<C4>^D<C9>^D<CB>^D<E4>^D<E9>^E
^B^E^U^E<^E<96>^E<98>^E<A3>^E<B7>^E<CC>^E<CE>^E<E1>^E<EC>^E<F3>^E<FE>^@^@^@^@^@^@^B^A^@^@^@^@
:
```

Figure 33 – Previewing a Binary Plist File with Time Zone Information

After determining the time zone information from the Macintosh partition, follow the steps outlined for creating a timeline for Linux (**Figure 14).**

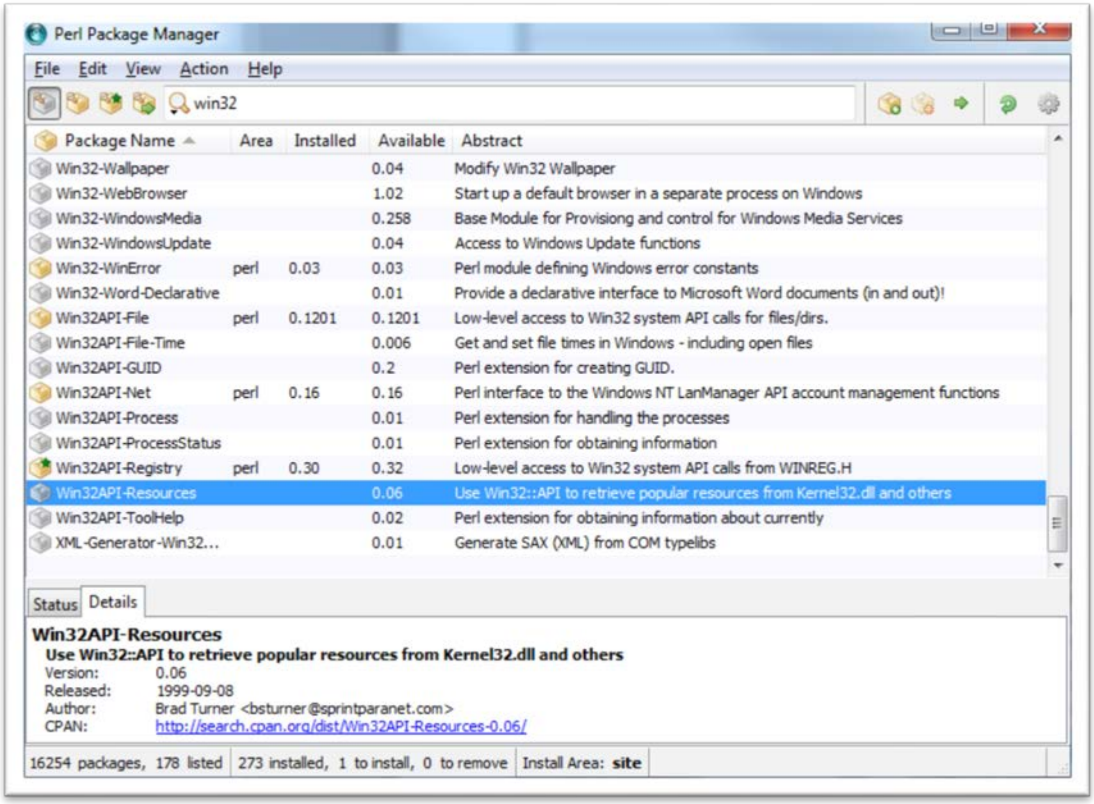
5 Timeline Creation and Analysis with Log2Timeline in Windows 7 & 8

5.1 Log2Timeline with Windows 7

5.1.1 Configuring Log2Timeline with Windows 7

Windows 7 is a popular desktop environment, and most forensic tools are built to run natively on the Windows platform, including Log2Timeline, which is cross platform. To install Log2Timeline on Windows, the ActiveState Perl framework, Log2timeline framework, and two additional libraries (Library 1, Library 2) have to be downloaded and unzipped. Once all of these are downloaded and unzipped, and ActiveState Perl is installed, launch the Perl Package Manager (PPM) to download the Perl dependencies (See **Figure 1**). Each bullet point below is a different dependency that must be installed for Log2Timeline to run properly. Install each package by searching for the package name, right clicking, and selecting "Install."

- datetime
- win32::api
- date::manip
- xml::libxml
- carp::assert
- digest::crc
- data::hexify
- image::exiftool
- file::mork
- datetime::format::strptime
- parse::win32registry
- html::scrubber



Package Name	Area	Installed	Available	Abstract
Win32-Wallpaper			0.04	Modify Win32 Wallpaper
Win32-WebBrowser			1.02	Start up a default browser in a separate process on Windows
Win32-WindowsMedia			0.258	Base Module for Provisiong and control for Windows Media Services
Win32-WindowsUpdate			0.04	Access to Windows Update functions
Win32-WinError	perl	0.03	0.03	Perl module defining Windows error constants
Win32-Word-Declarative			0.01	Provide a declarative interface to Microsoft Word documents (in and out)!
Win32API-File	perl	0.1201	0.1201	Low-level access to Win32 system API calls for files/dirs.
Win32API-File-Time			0.006	Get and set file times in Windows - including open files
Win32API-GUID			0.2	Perl extension for creating GUID.
Win32API-Net	perl	0.16	0.16	Perl interface to the Windows NT LanManager API account management functions
Win32API-Process			0.01	Perl extension for handling the processes
Win32API-ProcessStatus			0.01	Perl extension for obtaining information
Win32API-Registry	perl	0.30	0.32	Low-level access to Win32 system API calls from WINREG.H
Win32API-Resources			0.06	Use Win32::API to retrieve popular resources from Kernel32.dll and others
Win32API-ToolHelp			0.02	Perl extension for obtaining information about currently
XML-Generator-Win32...			0.01	Generate SAX (XML) from COM typelibs

Win32API-Resources

Use Win32::API to retrieve popular resources from Kernel32.dll and others

Version:	0.06
Released:	1999-09-08
Author:	Brad Turner <bsturner@sprintparanet.com>
CPAN:	http://search.cpan.org/dist/Win32API-Resources-0.06/

16254 packages, 178 listed | 273 installed, 1 to install, 0 to remove | Install Area: **site**

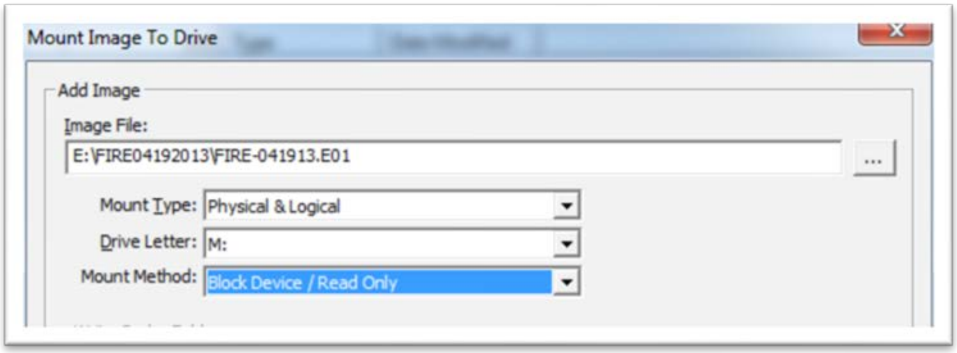
Figure 34 – PPM Package Manager

Copy the entire directory of XML-Entities/lib/XML folder into the Perl installation/lib/XML directory. After you have copied this, create a folder within the Perl installation lib/ directory named Mac. Copy the files from the Mac-Properties lib directory into the newly created Mac folder next. Then you can follow the steps outlined in **Table 1**.

1. Inside the log2timeline directory
    a. Delete the file lib/Log2t/input/pcap.pm
    b. Copy the content of the lib/Parse/* to c:/perl/lib/Parse/
    c. Copy the content of the folder lib/Log2t to c:/perl/lib/Log2t/*
    d. Copy lib/Log2Timeline.pm to c:/perl/lib/
    e. Copy log2timeline to c:/perl/bin/log2timeline.pl
    f. Copy l2t_process to c:/perl/bin/l2t_process.pl
    g. Copy timescanner to c:/perl/bin/timescanner.pl

Table 2

After completing the installation, the acquired image must be mounted. To do this, use FTK Imager to open the E01 file (in this example it is FIRE-041913.E01). Right click on the image and select “Image Mounting…” from the context menu. Then select the “Physical & Logical” mount type and assign a drive letter using the “Block Device/Read Only” mount method (See **Figure 2**). Clicking mount will now virtually mount each partition of the E01 as a drive, allowing Log2Timeline to parse through the partitions.



Mount Image To Drive

Add Image

Image File: E:\FIRE04192013\FIRE-041913.E01

Mount Type: Physical & Logical

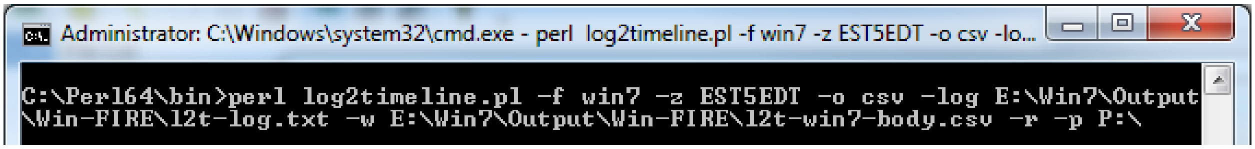
Drive Letter: M:

Mount Method: Block Device / Read Only

Figure 35 – Mounting the Image for Log2Timeline Processing

5.1.2 Running Log2Timeline on Windows 7

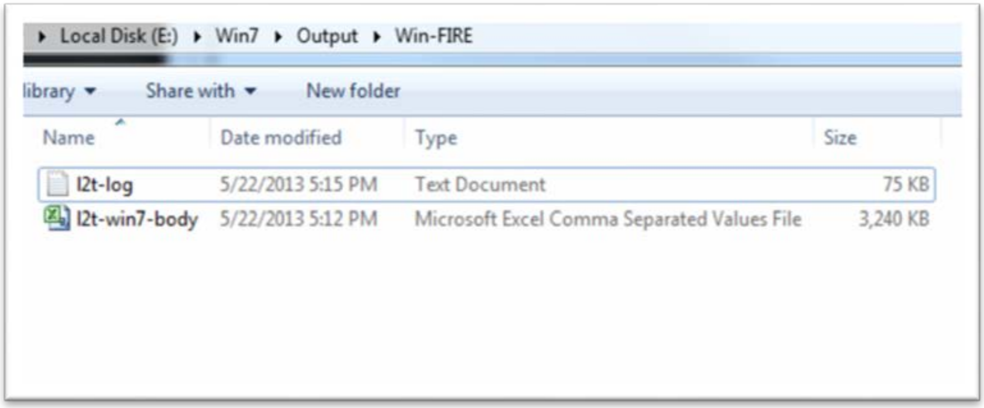
When the previous process is completed, the log2timeline process can be run by opening the command prompt in the C:\Perl\bin\ directory. Holding shift, then right click and select “open command window here” from the context menu. At this point, type **perl log2timeline –f win7 –z EST5EDT –o csv –log E:\path\to\log\file.txt –w: E:\path\to\output\file.csv –r –p M:\** where drive M: is the target to scan. This will generate the log text file and the output csv file in the directory selected (see **Figure 3**).



Administrator: C:\Windows\system32\cmd.exe - perl log2timeline.pl -f win7 -z EST5EDT -o csv -lo...

```
C:\Perl64\bin>perl log2timeline.pl -f win7 -z EST5EDT -o csv -log E:\Win7\Output
\Win-FIRE\l2t-log.txt -w E:\Win7\Output\Win-FIRE\l2t-win7-body.csv -r -p P:\
```

Figure 36 – Command used to create timeline



Name	Date modified	Type	Size
l2t-log	5/22/2013 5:15 PM	Text Document	75 KB
l2t-win7-body	5/22/2013 5:12 PM	Microsoft Excel Comma Separated Values File	3,240 KB

Figure 37 – Output files and locations

the test image contained 3 partitions, and the run above run was only for the Windows 7 partition. Windows 7 only recognizes the NTFS, exFAT, and FAT file systems. Since FTK virtually mounts as a drive on Windows, Windows does not mount the drives as usable. Windows 7 Is not able to interpret data on EXT or HFS+ file systems, preventing the Macintosh and Linux partitions from being mounted.

5.2 Configuring and Running Log2Timeline with Windows 8

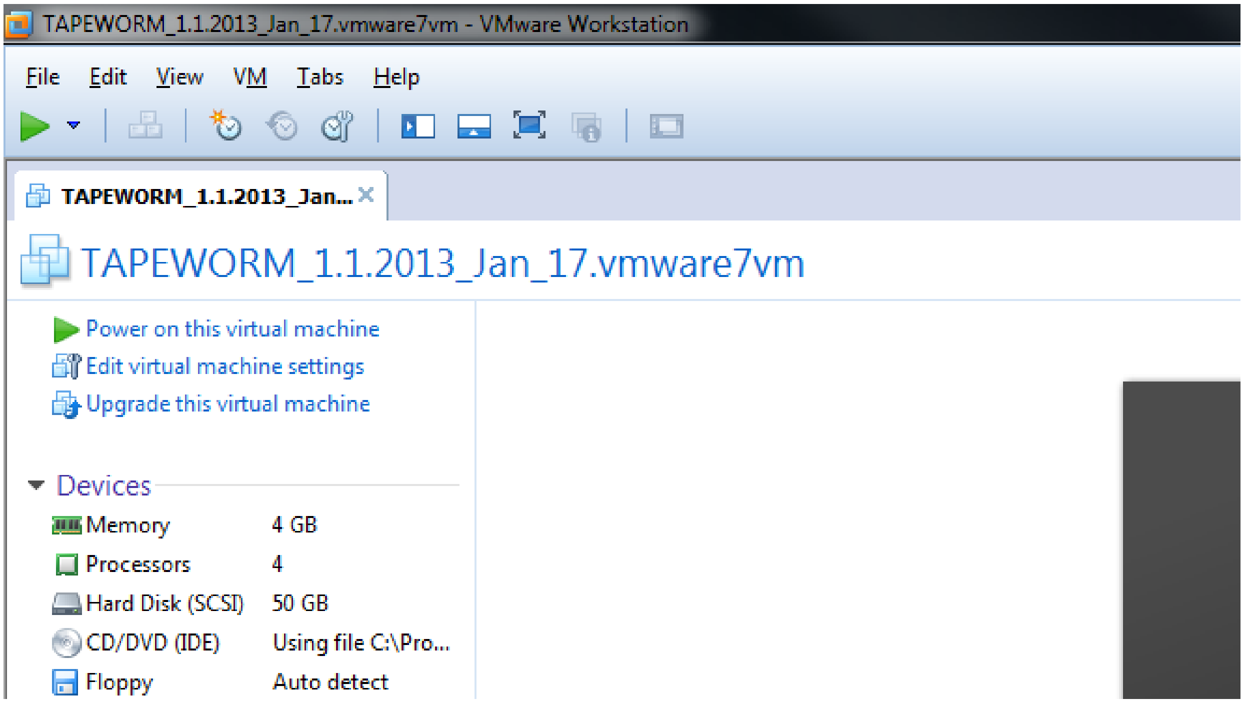
The process for configuration and execution of Log2Timeline on Windows 7 and Windows 8 is identical.

6 Timeline Creation and Analysis with Log2Timeline in TAPEWORM

6.1 Configuring TAPEWORM for Log2Timeline

TAPEWORM is an open sourced workflow automation system bundled inside a Debian based virtual machine and can be downloaded at: http://tapeworm.s3.amazonaws.com/TAPEWORM_1.1.2013_Jan_17.vmware7vm.zip.

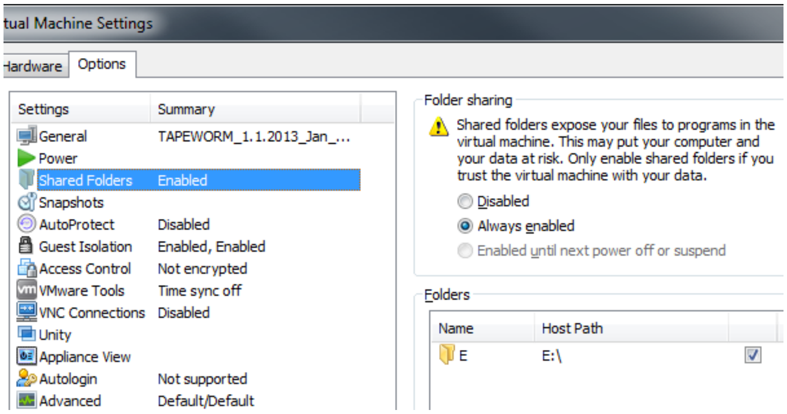
Once the zip file for TAPEWORM is downloaded and unzipped, it can be opened in VMWare Workstation 7, 8, 9,;VMWare Fusion; or VMWare Player, 3, 4, or 5. Additionally, it can be converted for use with VirtualBox (using the latest version of any software is recommended). Once it has been opened in the virtualization product, VMWare (shown in **Figure 1**) can be configured for the host machine.


TAPEWORM_1.1.2013_Jan_17.vmware7vm - VMware Workstation
File
Edit
View
VM
Tabs
Help
TAPEWORM_1.1.2013_Jan...
TAPEWORM_1.1.2013_Jan_17.vmware7vm
Power on this virtual machine
Edit virtual machine settings
Upgrade this virtual machine
Devices

Memory	4 GB
Processors	4
Hard Disk (SCSI)	50 GB
CD/DVD (IDE)	Using file C:\Pro...
Floppy	Auto detect

Figure 38 – TAPEWORM VM in VMWare Workstation 9.0.2

Opening the "Edit virtual machine settings" window allows the user to change the number of processors, amount of RAM, and shared folders to be configured for use. The recommended specifications are preset into the virtual machine. **Figure 2** shows the shared folder configuration.



tual Machine Settings

Hardware | Options

Settings	Summary
General	TAPEWORM_1.1.2013_Jan_...
Power	
Shared Folders	Enabled
Snapshots	
AutoProtect	Disabled
Guest Isolation	Enabled, Enabled
Access Control	Not encrypted
VMware Tools	Time sync off
VNC Connections	Disabled
Unity	
Appliance View	
Autologin	Not supported
Advanced	Default/Default

Folder sharing

Shared folders expose your files to programs in the virtual machine. This may put your computer and your data at risk. Only enable shared folders if you trust the virtual machine with your data.

○ Disabled
● Always enabled
○ Enabled until next power off or suspend

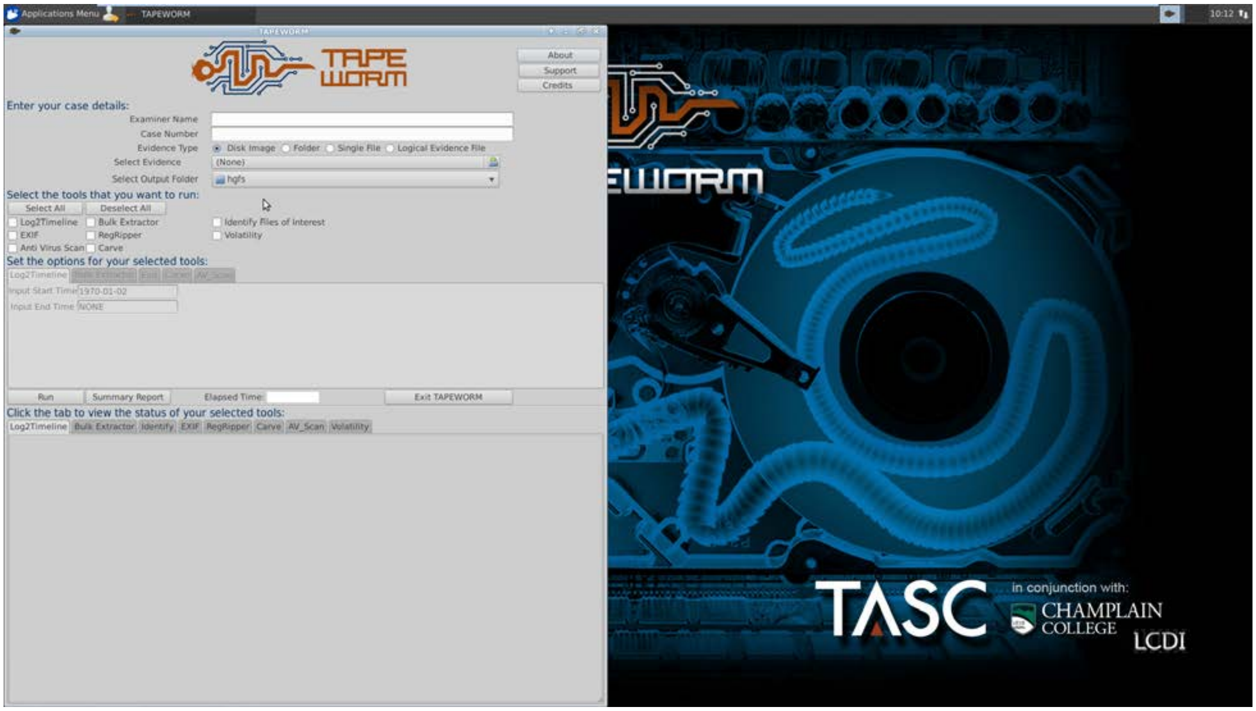
Folders

Name	Host Path	
E	E:\	☑

Figure 39 – Shared Folder Configuration

6.2 Running Log2Timeline in TAPEWORM

Once the virtual machine is running, the desktop will show the main tapeworm window (See **Figure 3**). From this interface, the automated graphic user interface (GUI) can be used, and the preprocessing and timeline creation for any exhibit can begin. It should be noted that if the main window is minimized, other tools may be accessed either from the terminal or separate GUI applications from the desktop.



Enter your case details:

Examiner Name
Case Number
Evidence Type ● Disk Image ○ Folder ○ Single File ○ Logical Evidence File
Select Evidence (None)
Select Output Folder hgfs

Select the tools that you want to run:

Select All | Deselect All

□ Log2Timeline □ Bulk Extractor □ Identify Files of Interest
□ EXIF □ RegRipper □ Volatility
□ Anti Virus Scan □ Carve

Set the options for your selected tools:

Log2Timeline | Bulk Extractor | Exif | Carver | AV_Scan

Input Start Time 1970-01-02
Input End Time NONE

Run | Summary Report | Elapsed Time: | Exit TAPEWORM

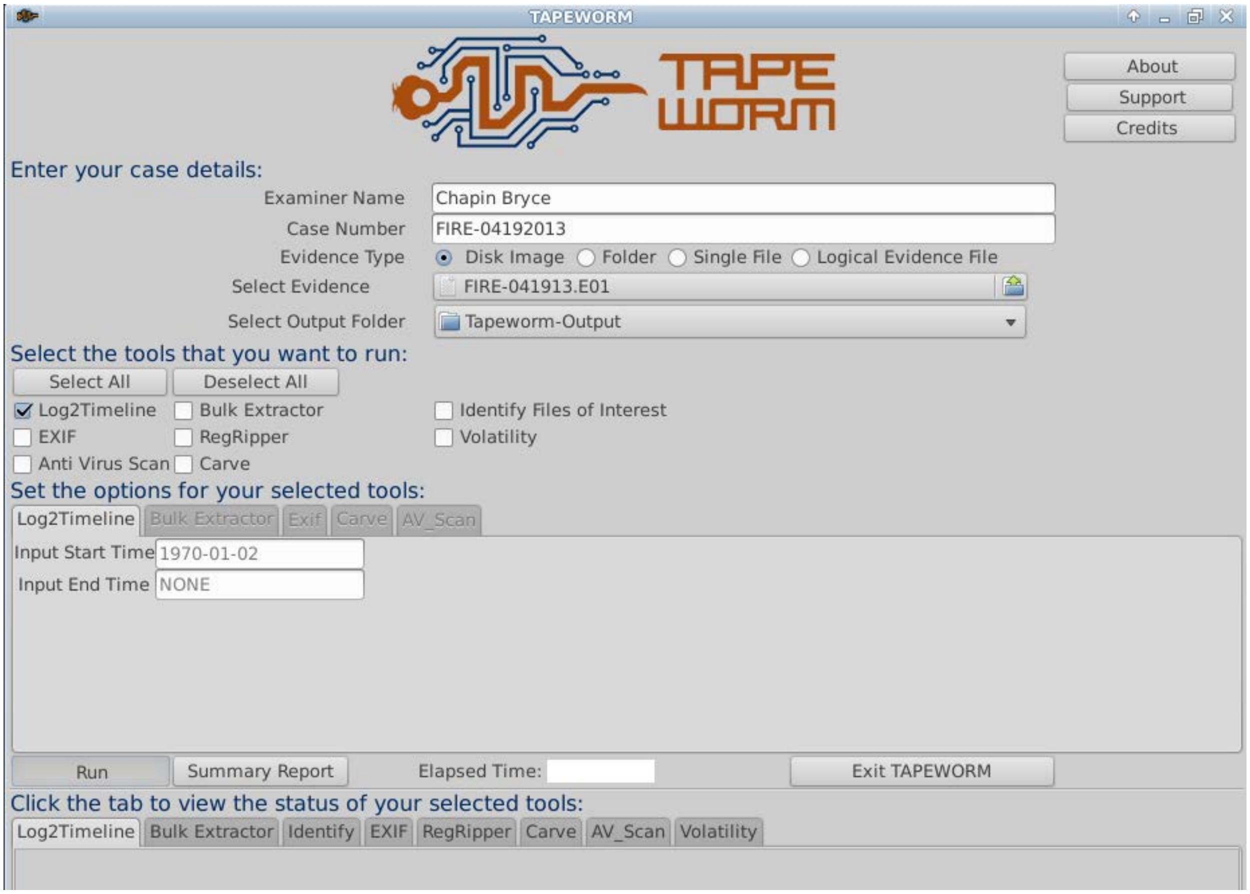
Click the tab to view the status of your selected tools:

Log2Timeline | Bulk Extractor | Identify | EXIF | RegRipper | Carve | AV_Scan | Volatility

Figure 40 – TAPEWORM Interface

As seen in **Figure 4**, the TAPEWORM interface has case options that need to be filled out in regards to the specific case being processed. After filling out case details, select the evidence type. For creating a timeline, the disk image option is most often used due to the E01, DD, or AFF files used to acquire exhibits. Additionally, Log2Timeline can be run against a folder or logical image (but not a single file) using TAPEWORM's interface.

Once the evidence type is selected, select your evidence and output location. After choosing the input and output locations and selecting Log2Timeline, hit run. TAPEWORM will automate the process. It allows multiple tools to run in succession as well, so if the evidence requires other processing, you can select the appropriate tools and options to run alongside the timeline at this point.



Enter your case details:

Examiner Name: Chapin Bryce
Case Number: FIRE-04192013
Evidence Type: ● Disk Image ○ Folder ○ Single File ○ Logical Evidence File
Select Evidence: FIRE-041913.E01
Select Output Folder: Tapeworm-Output

Select the tools that you want to run:

☑ Log2Timeline ☐ Bulk Extractor ☐ Identify Files of Interest
☐ EXIF ☐ RegRipper ☐ Volatility
☐ Anti Virus Scan ☐ Carve

Set the options for your selected tools:

Log2Timeline | Bulk Extractor | Exif | Carve | AV_Scan

Input Start Time: 1970-01-02
Input End Time: NONE

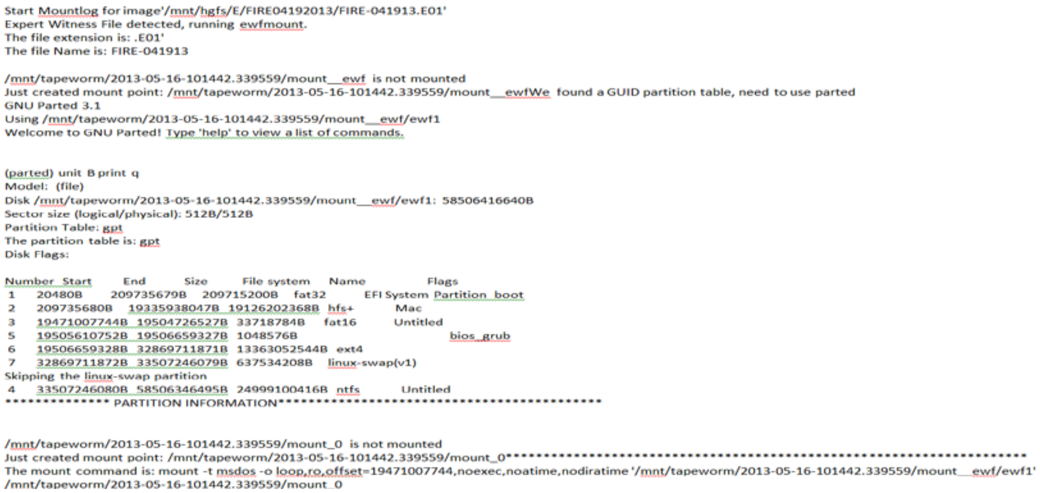
Elapsed Time:

Click the tab to view the status of your selected tools:

Log2Timeline | Bulk Extractor | Identify | EXIF | RegRipper | Carve | AV_Scan | Volatility

Figure 41 – TAPEWORM Case Information

TAPEWORM utilizes a number of the same steps taken when creating a timeline in SIFT 2.14. TAPEWORM begins by mounting the image, determining the partition types, creating necessary mount points, and doing the math for offsets, as well as starting and ending sectors.



```
Start Mountlog for image'/mnt/hgfs/E/FIRE04192013/FIRE-041913.E01'
Expert Witness File detected, running ewfmount.
The file extension is: .E01'
The file Name is: FIRE-041913

/mnt/tapeworm/2013-05-16-101442.339559/mount__ewf  is not mounted
Just created mount point: /mnt/tapeworm/2013-05-16-101442.339559/mount__ewfWe  found a GUID partition table, need to use parted
GNU Parted 3.1
Using /mnt/tapeworm/2013-05-16-101442.339559/mount__ewf/ewf1
Welcome to GNU Parted! Type 'help' to view a list of commands.


(parted) unit B print q
Model:  (file)
Disk /mnt/tapeworm/2013-05-16-101442.339559/mount__ewf/ewf1:  58506416640B
Sector size (logical/physical): 512B/512B
Partition Table: gpt
The partition table is: gpt
Disk Flags:

Number  Start        End          Size         File system    Name            Flags
 1      20480B       209735679B   209715200B   fat32          EFI System Partition  boot
 2      209735680B   19335938047B  191262023688B  hfs+        Mac
 3      19471007744B  19504726527B  33718784B   fat16         Untitled
 5      19505610752B  19506659327B  1048576B                                  bios_grub
 6      19506659328B  32869711871B  13363052544B  ext4
 7      32869711872B  33507246079B  637534208B   linux-swap(v1)
Skipping the linux-swap partition
 4      33507246080B  58506346495B  24999100416B  ntfs          Untitled
************** PARTITION INFORMATION*********************************************


/mnt/tapeworm/2013-05-16-101442.339559/mount_0  is not mounted
Just created mount point: /mnt/tapeworm/2013-05-16-101442.339559/mount_0**********************************************************************
The mount command is: mount -t msdos -o loop,ro,offset=19471007744,noexec,noatime,nodiratime '/mnt/tapeworm/2013-05-16-101442.339559/mount__ewf/ewf1'
/mnt/tapeworm/2013-05-16-101442.339559/mount_0
```

Figure 42 – TAPEWORM Mount Log

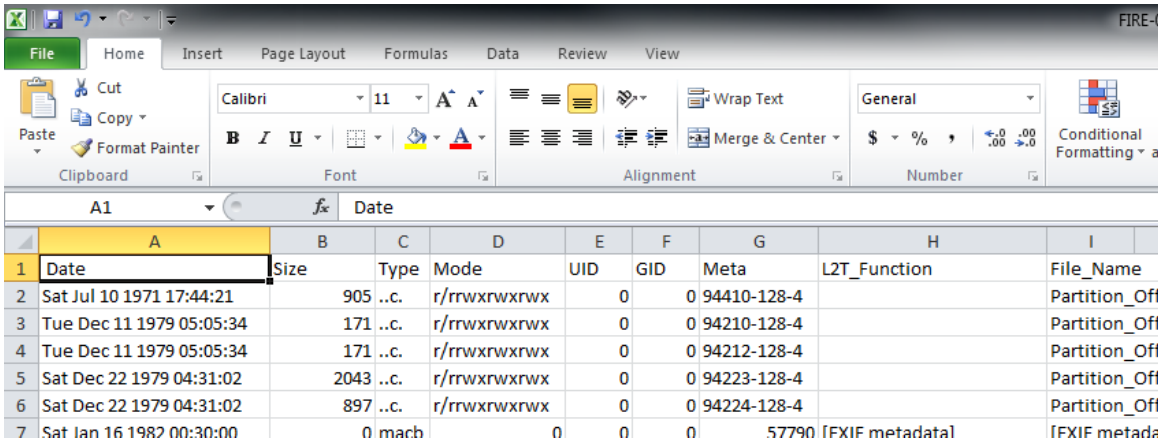
For the test image used in this tutorial (FIRE 04192013.E01), TAPEWORM was able to create a timeline in 25 hours and 17 minutes. It is important to note that this exhibit had 3 operating systems installed: Windows 7, Mac OSX, and BackTrack Linux. Each of these Operating Systems runs on a separate time zone, each with unique user data. If the 25 hour and 17 minute run may seem extremely long, note that TAPEWORM ran Log2Timeline against every partition in the correct time zone, as well as in every format type in one instance without the use of a single command.

Once the timeline is completed, a new folder can be found in the output directory containing an organized (by tool) output. Inside the log2timeline directory are the file types and logs for running log2timeline, with the partition offset in the file names. TAPEWORM exports the timelines in CSV, bodyfile, mactime, or TLN formats. The CSV format is fairly common and can be opened with Microsoft Excel or any other spreadsheet program. The bodyfile and mactime format can be used with the Sleuth Kit and other open source forensic tools for timeline analysis. The TLN format is a custom format that can function with Harlan Carvey's tools.

6.3 Reading the output from TAPEWORM

For this guide, the CSV files will be used to look at the output. In TAPEWORM, the CSV documents are split when they are too large for excel to open, preventing any error for maximum file size. Also in the output folder are two different CSV files, one with modules and one without. TAPEWORM adds an additional column to the Log2timeline CSV output, to include the module used to process the particular artifact. For example, if the artifact listed in the CSV file originated from an index.dat file, the index.dat will be listed in the module column. This seems to be a lot of extra information, but when performing an investigation and attempting to look into specific information such as a timeline of Firefox history, sorting by module becomes extremely helpful. The CSV files will also contain the data within any specified date range, while the other file types do not allow for that sort of filtering. The CSV will be the most accurate source of information.

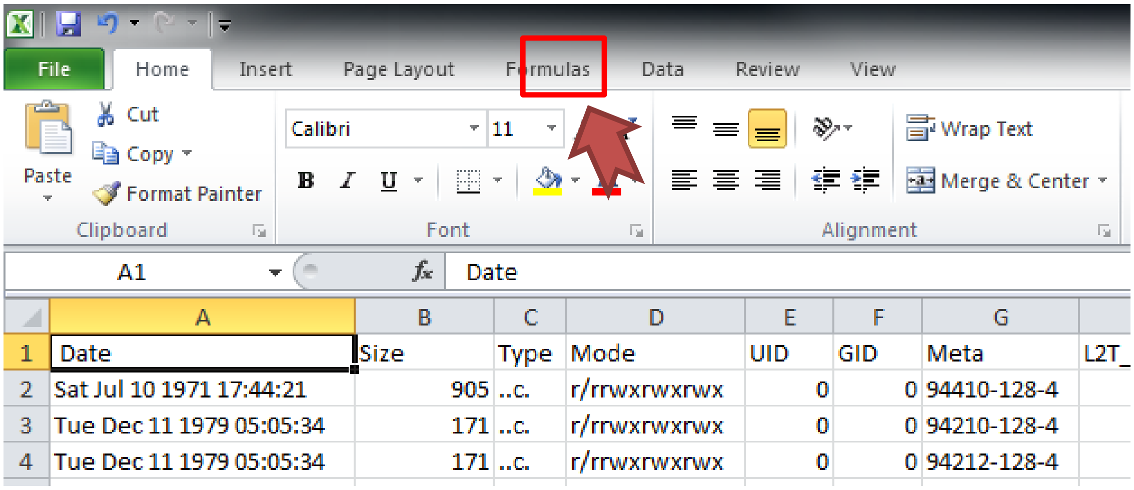
Opening the CSV files in Microsoft Excel shows the data neatly organized by column: sorted by date and time from oldest to newest (See **Figure 6**). Using Excel's data sort feature, the first row can be used to sort and filter content throughout the document.



	A	B	C	D	E	F	G	H	I
1	Date	Size	Type	Mode	UID	GID	Meta	L2T_Function	File_Name
2	Sat Jul 10 1971 17:44:21	905	..c.	r/rrwxrwxrwx	0	0	94410-128-4		Partition_Off
3	Tue Dec 11 1979 05:05:34	171	..c.	r/rrwxrwxrwx	0	0	94210-128-4		Partition_Off
4	Tue Dec 11 1979 05:05:34	171	..c.	r/rrwxrwxrwx	0	0	94212-128-4		Partition_Off
5	Sat Dec 22 1979 04:31:02	2043	..c.	r/rrwxrwxrwx	0	0	94223-128-4		Partition_Off
6	Sat Dec 22 1979 04:31:02	897	..c.	r/rrwxrwxrwx	0	0	94224-128-4		Partition_Off
7	Sat Jan 16 1982 00:30:00	0	mach	0	0	0	57790	[EXIF metadata]	[EXIF metada

Figure 43 – CSV Timeline Output

To enable the sorting feature, select the "data" tab in excel (See **Figure 7**).



A1 *fx* Date

	A	B	C	D	E	F	G	
1	Date	Size	Type	Mode	UID	GID	Meta	L2T_
2	Sat Jul 10 1971 17:44:21	905	..c.	r/rrwxrwxrwx	0	0	94410-128-4	
3	Tue Dec 11 1979 05:05:34	171	..c.	r/rrwxrwxrwx	0	0	94210-128-4	
4	Tue Dec 11 1979 05:05:34	171	..c.	r/rrwxrwxrwx	0	0	94212-128-4	

Figure 44 – Data Filtering with Excel

Then select filter (See **Figure 8**).

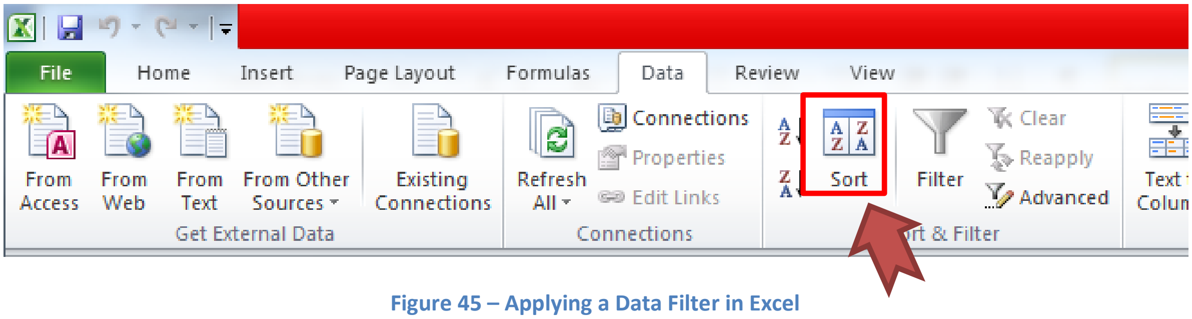


Figure 45 – Applying a Data Filter in Excel

Log2Timeline Guide for SIFT

Written by: Chapin Bryce

Updated: July, 2013

The Senator Patrick Leahy Center for Digital Investigation

Champlain College

Table of Contents