KROLL

How to Detect Timestomping with KAPE

June 15, 2022

Upcoming KAPE Intensive Training and Certification Sessions

- Virtual live sessions
- Max 25 students

Full Calendar Available here:
https://bit.ly/KAPE2022

SCHEDULE	INSTRUCTORS
June 23, 2022 10:00 a.m. - 7:00 p.m. ET	Eric Zimmerman Sean Straw Scott Zuberbuehler Andrew Rathbun
September 27, 2022 10:00 a.m. - 7:00 p.m. ET	Eric Zimmerman Sean Straw Scott Zuberbuehler Andrew Rathbun
October 4, 2022 8:00 a.m. - 5:00 p.m. BST	James Thoburn Paul Wells Guillermo Roman

Table of Contents

1. What is Timestomping?
2. Timestomping Tools
3. Investigative Process Using KAPE
4. Analysis with Timeline Explorer: NewFileTime Scenario
5. Analysis with Timeline Explorer: Total Commander Scenario
6. Important Considerations
7. Real Life Examples of Timestomping
8. Advanced Timestomping Methods/Tools

Andrew Rathbun

Vice President, Cyber Risk

Let's Connect

- LinkedIn (andrewrathbun)
- Twitter (@bunsofwrath12)
- GitHub (AndrewRathbun)
- Discord (rathbuna#0679)

Work Experience

- 2020-Present: Vice President at Kroll
  - Digital Forensics & Incident Response
  - KAPE Instructor
- 2019-2020: HHS OIG
  - Forensic Computer Examiner (2210)
- 2012-2019: Michigan State University Police Department
  - 2016-2019: Detective (digital forensics and general investigations)
  - 2012-2015: Police Officer
- 2005-2011: USMC Veteran
  - Rifleman (0311)
  - Deployment to Fallujah, Iraq (2006-2007)

Side Projects

- 2018-Present: Administrator of the Digital Forensics Discord Server
  - 2020 DFIR Resource of the Year – Winner
  - 2019 DFIR Resource of the Year – Winner
- 2019-Present: AboutDFIR.com Contributor
  - 2020 DFIR Resource of the Year – Nominee
  - 2019 DFIR Resource of the Year – Nominee
- 2020-Present: Steward of many GitHub repos
- 2021-Present: SANS DFIR Summit Advisory Board Member
- 2022: SANS Gold Paper (GCFE) – Post-submission, in review

What is Timestomping?

What is Timestomping?

Making sure we're all on the same page

- Common anti-forensic technique observed in incident response matters
- Altering timestamps on an NTFS file system
  - MAC(B) – Modified, Accessed, Change, and Birth (File Creation)
  - Usually 0x10, but 0x30 can be altered with extra effort/knowledge
- Why would someone timestomp files?
  - Often used to hide tools or tool output from incident responders when conducting file system timestamp analysis in the $MFT
  - Can also be used to make those files you downloaded back in the early days of the internet have the 1996 timestamps that were lost over time transferring from one storage medium to another, from one ZIP file to another, etc.
- In 2 scenarios, this webinar will cover:
  - Examples of the most widely observed techniques seen by Kroll in our day-to-day engagements
  - Example of detection and analysis techniques using KAPE and Timeline Explorer

Investigative Mindset

Things to keep in mind when hunting for Timestomping

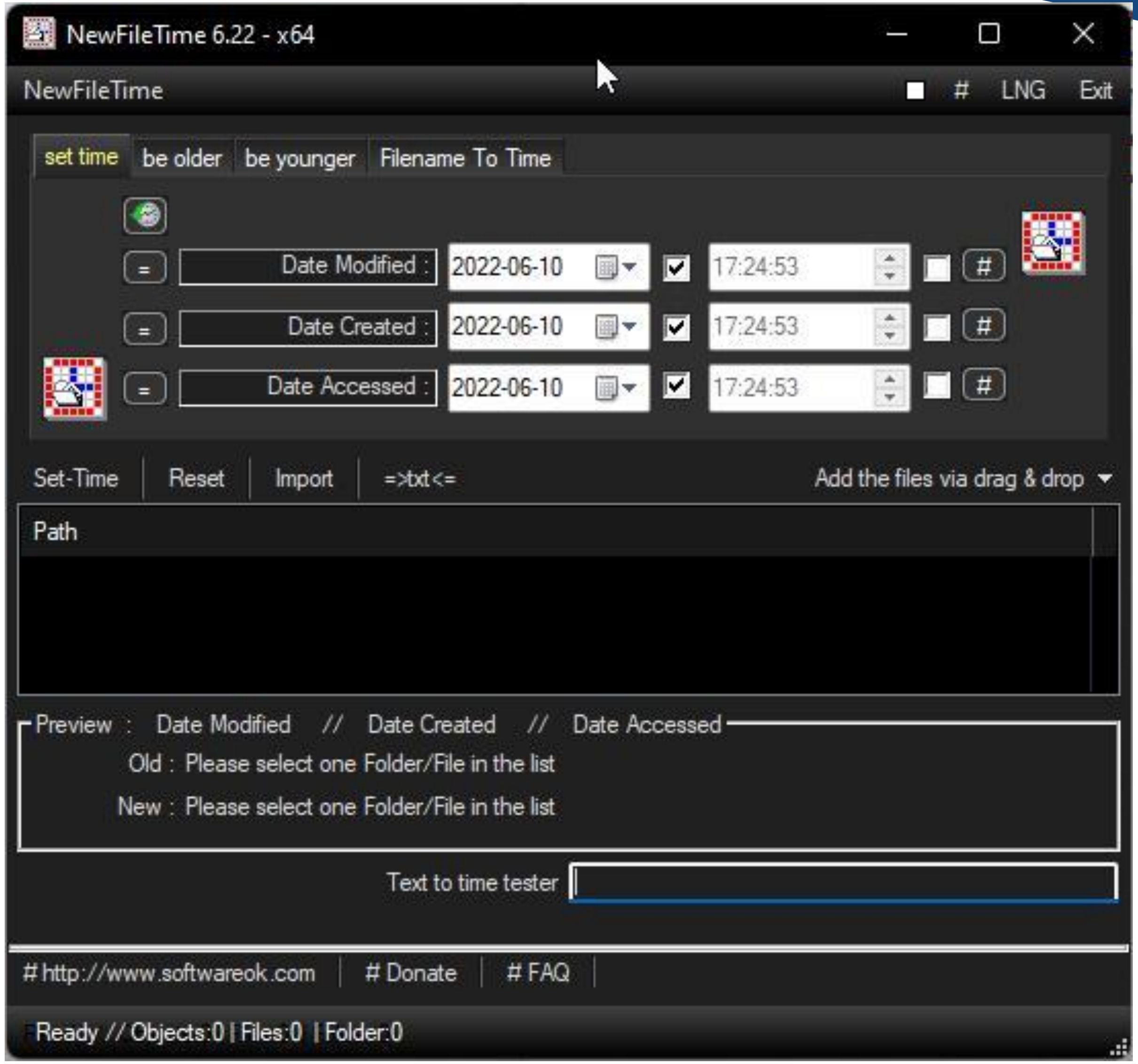
- Timestomping is common enough to where one should expect to see it on most IR cases where threat actors have something to hide
- Examiners don't know if timestomping is present on a host until they do
  - Normally, I discover suspicious files based on file name or location, and THEN notice indicators of timestomping
  - Once I've seen it once on a host, I usually find multiple other instances of it on the host and within the network
- Understand that as with everything, context is king
  - Plenty of applications don't record the sub-seconds

Timestomping Tools

NewFileTime

Popular Timestamp Manipulation Tool

- Covered extensively in the blog post series
- Timestomping a File with NewFileTime
- Free
- Easy to use
- Can modify the Modified, Created, and Accessed timestamps of any file
- Can also adopt the timestamp of another file
- NewFileTime 6.22 Download (softwareok.com)



NewFileTime 6.22 - x64

Date Modified :	2022-06-10	☑	17:24:53
Date Created :	2022-06-10	☑	17:24:53
Date Accessed :	2022-06-10	☑	17:24:53

Preview : Date Modified // Date Created // Date Accessed

Old : Please select one Folder/File in the list

New : Please select one Folder/File in the list

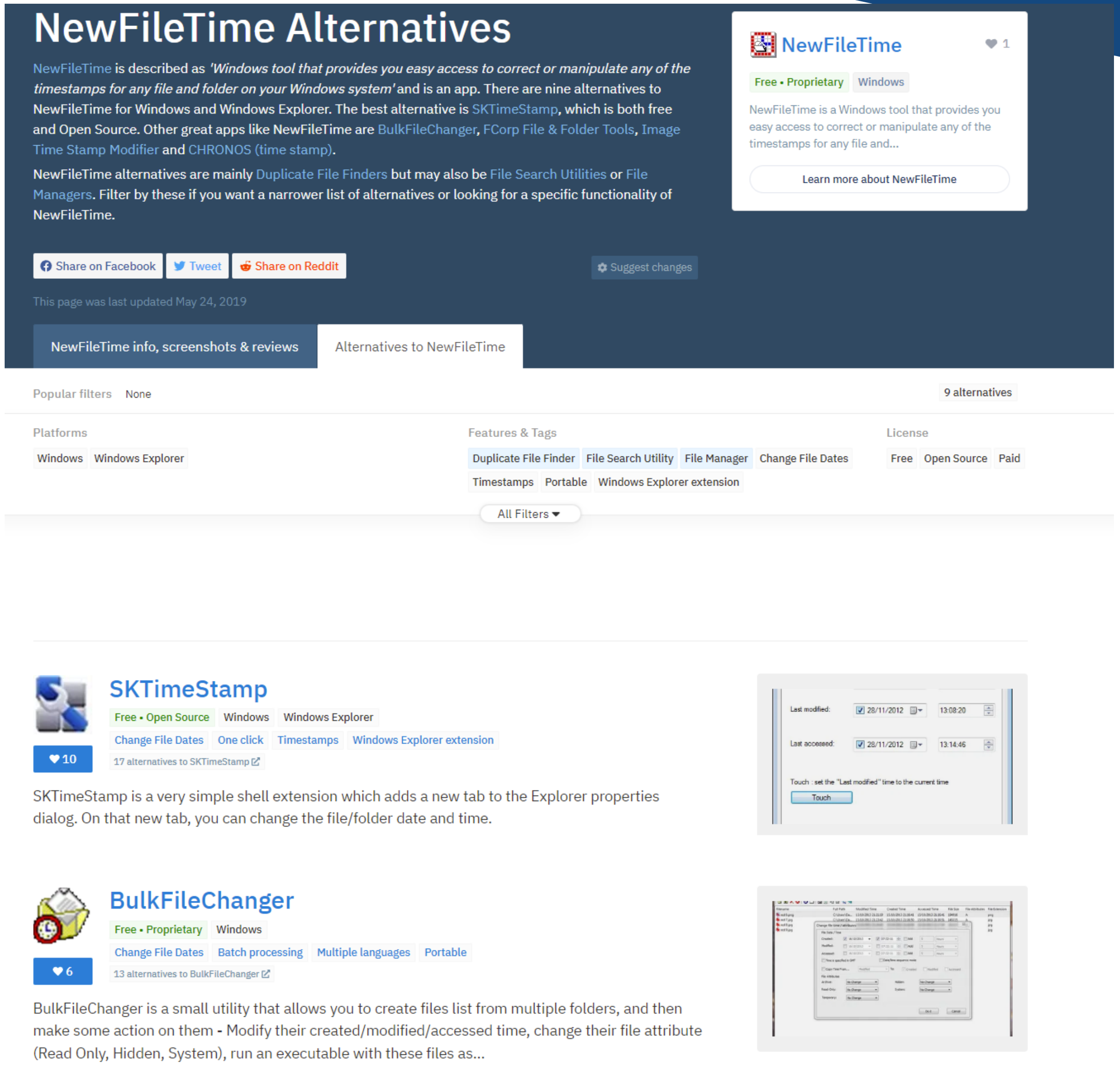
Text to time tester

Ready // Objects:0 | Files:0 | Folder:0

NewFileTime Alternatives

AlternativeTo.net – List of NewFileTime Alternatives

- SKTimeStamp
- BulkFileChanger
- Chronos (time stamp)
- Attribute Magic
- TouchPro
- Ninotech Date Edit
- File version info editor



NewFileTime Alternatives

NewFileTime is described as *'Windows tool that provides you easy access to correct or manipulate any of the timestamps for any file and folder on your Windows system'* and is an app. There are nine alternatives to NewFileTime for Windows and Windows Explorer. The best alternative is SKTimeStamp, which is both free and Open Source. Other great apps like NewFileTime are BulkFileChanger, FCorp File & Folder Tools, Image Time Stamp Modifier and CHRONOS (time stamp).

NewFileTime alternatives are mainly Duplicate File Finders but may also be File Search Utilities or File Managers. Filter by these if you want a narrower list of alternatives or looking for a specific functionality of NewFileTime.

This page was last updated May 24, 2019

NewFileTime info, screenshots & reviews
Alternatives to NewFileTime

NewFileTime ♥ 1

Free • Proprietary Windows

NewFileTime is a Windows tool that provides you easy access to correct or manipulate any of the timestamps for any file and...

Popular filters None

9 alternatives

Platforms: Windows, Windows Explorer

Features & Tags: Duplicate File Finder, File Search Utility, File Manager, Change File Dates, Timestamps, Portable, Windows Explorer extension

License: Free, Open Source, Paid

SKTimeStamp

♥ 10

Free • Open Source Windows Windows Explorer

Change File Dates One click Timestamps Windows Explorer extension

17 alternatives to SKTimeStamp

SKTimeStamp is a very simple shell extension which adds a new tab to the Explorer properties dialog. On that new tab, you can change the file/folder date and time.

Last modified: 28/11/2012 13:08:20
Last accessed: 28/11/2012 13:14:46
Touch : set the "Last modified" time to the current time
Touch

BulkFileChanger

♥ 6

Free • Proprietary Windows

Change File Dates Batch processing Multiple languages Portable

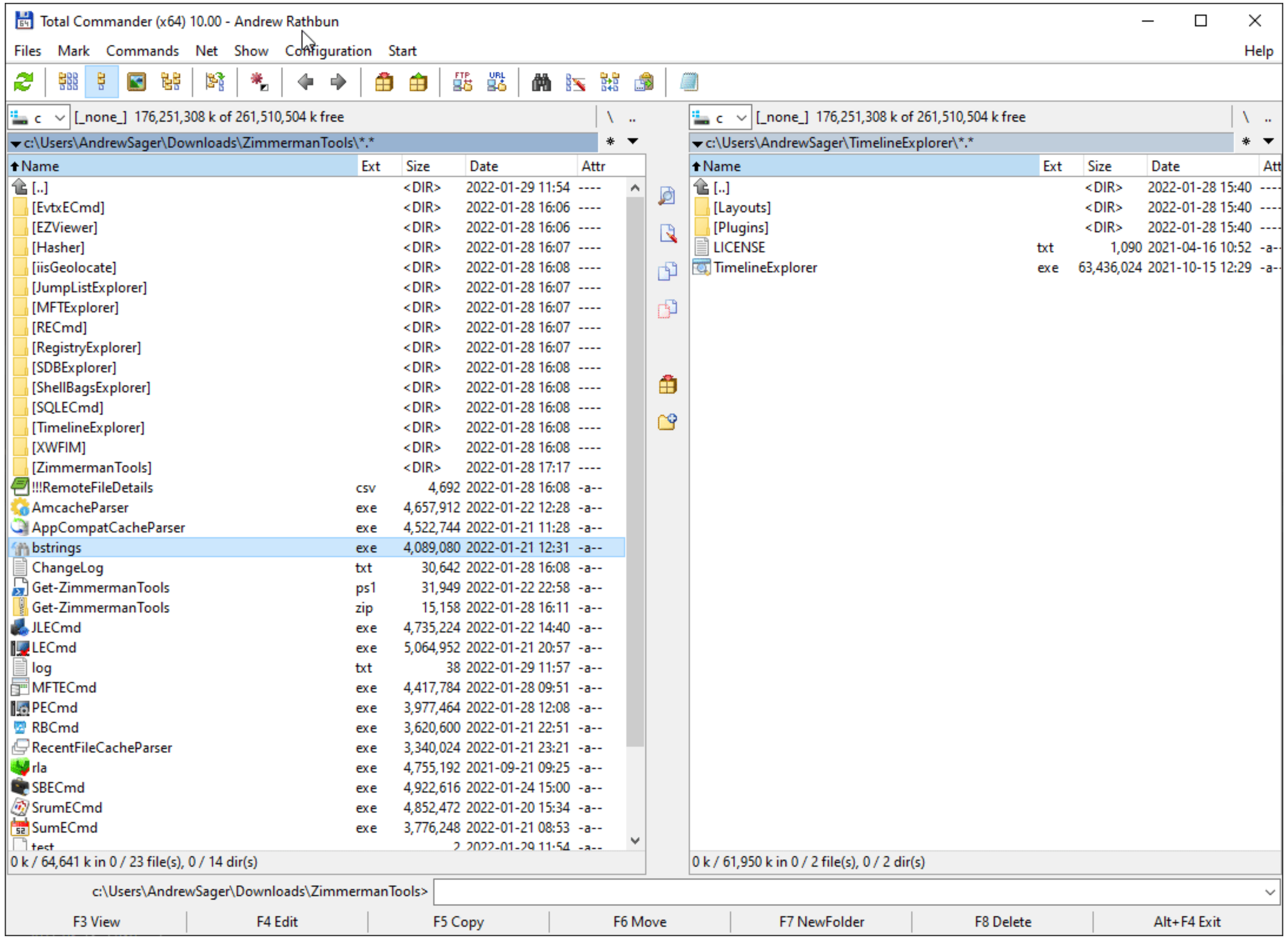
13 alternatives to BulkFileChanger

BulkFileChanger is a small utility that allows you to create files list from multiple folders, and then make some action on them - Modify their created/modified/accessed time, change their file attribute (Read Only, Hidden, System), run an executable with these files as...

Total Commander

Windows File Explorer Alternative

- Dual Pane Windows File Explorer alternative with lots of functionality and extensibility via plugins
  - Total Commander - Plugins (ghisler.com)
- Can be used as a data exfil tool
- File -> Change Attributes
- Lots of alternatives on AlternativeTo
  - Sidenote: KAPE Targets exist for most of them ☺
  - Compound Target: FileExplorerReplacements.tkape
- Total Commander - home (ghisler.com)



c [_none_] 176,251,308 k of 261,510,504 k free

c:\Users\AndrewSager\Downloads\ZimmermanTools\*.*

Name	Ext	Size	Date	Attr
[..]		<DIR>	2022-01-29 11:54	----
[EvtxECmd]		<DIR>	2022-01-28 16:06	----
[EZViewer]		<DIR>	2022-01-28 16:06	----
[Hasher]		<DIR>	2022-01-28 16:07	----
[iisGeolocate]		<DIR>	2022-01-28 16:08	----
[JumpListExplorer]		<DIR>	2022-01-28 16:07	----
[MFTExplorer]		<DIR>	2022-01-28 16:07	----
[RECmd]		<DIR>	2022-01-28 16:07	----
[RegistryExplorer]		<DIR>	2022-01-28 16:07	----
[SDBExplorer]		<DIR>	2022-01-28 16:08	----
[ShellBagsExplorer]		<DIR>	2022-01-28 16:08	----
[SQLECmd]		<DIR>	2022-01-28 16:08	----
[TimelineExplorer]		<DIR>	2022-01-28 16:08	----
[XWFIM]		<DIR>	2022-01-28 16:08	----
[ZimmermanTools]		<DIR>	2022-01-28 17:17	----
!!!RemoteFileDetails	csv	4,692	2022-01-28 16:08	-a--
AmcacheParser	exe	4,657,912	2022-01-22 12:28	-a--
AppCompatCacheParser	exe	4,522,744	2022-01-21 11:28	-a--
bstrings	exe	4,089,080	2022-01-21 12:31	-a--
ChangeLog	txt	30,642	2022-01-28 16:08	-a--
Get-ZimmermanTools	ps1	31,949	2022-01-22 22:58	-a--
Get-ZimmermanTools	zip	15,158	2022-01-28 16:11	-a--
JLECmd	exe	4,735,224	2022-01-22 14:40	-a--
LECmd	exe	5,064,952	2022-01-21 20:57	-a--
log	txt	38	2022-01-29 11:57	-a--
MFTECmd	exe	4,417,784	2022-01-28 09:51	-a--
PECmd	exe	3,977,464	2022-01-28 12:08	-a--
RBCmd	exe	3,620,600	2022-01-21 22:51	-a--
RecentFileCacheParser	exe	3,340,024	2022-01-21 23:21	-a--
rla	exe	4,755,192	2021-09-21 09:25	-a--
SBECmd	exe	4,922,616	2022-01-24 15:00	-a--
SrumECmd	exe	4,852,472	2022-01-20 15:34	-a--
SumECmd	exe	3,776,248	2022-01-21 08:53	-a--
test		2	2022-01-29 11:54	-a--

0 k / 64,641 k in 0 / 23 file(s), 0 / 14 dir(s)

c [_none_] 176,251,308 k of 261,510,504 k free

c:\Users\AndrewSager\TimelineExplorer\*.*

Name	Ext	Size	Date	Att
[..]		<DIR>	2022-01-28 15:40	----
[Layouts]		<DIR>	2022-01-28 15:40	----
[Plugins]		<DIR>	2022-01-28 15:40	----
LICENSE	txt	1,090	2021-04-16 10:52	-a-
TimelineExplorer	exe	63,436,024	2021-10-15 12:29	-a-

0 k / 61,950 k in 0 / 2 file(s), 0 / 2 dir(s)

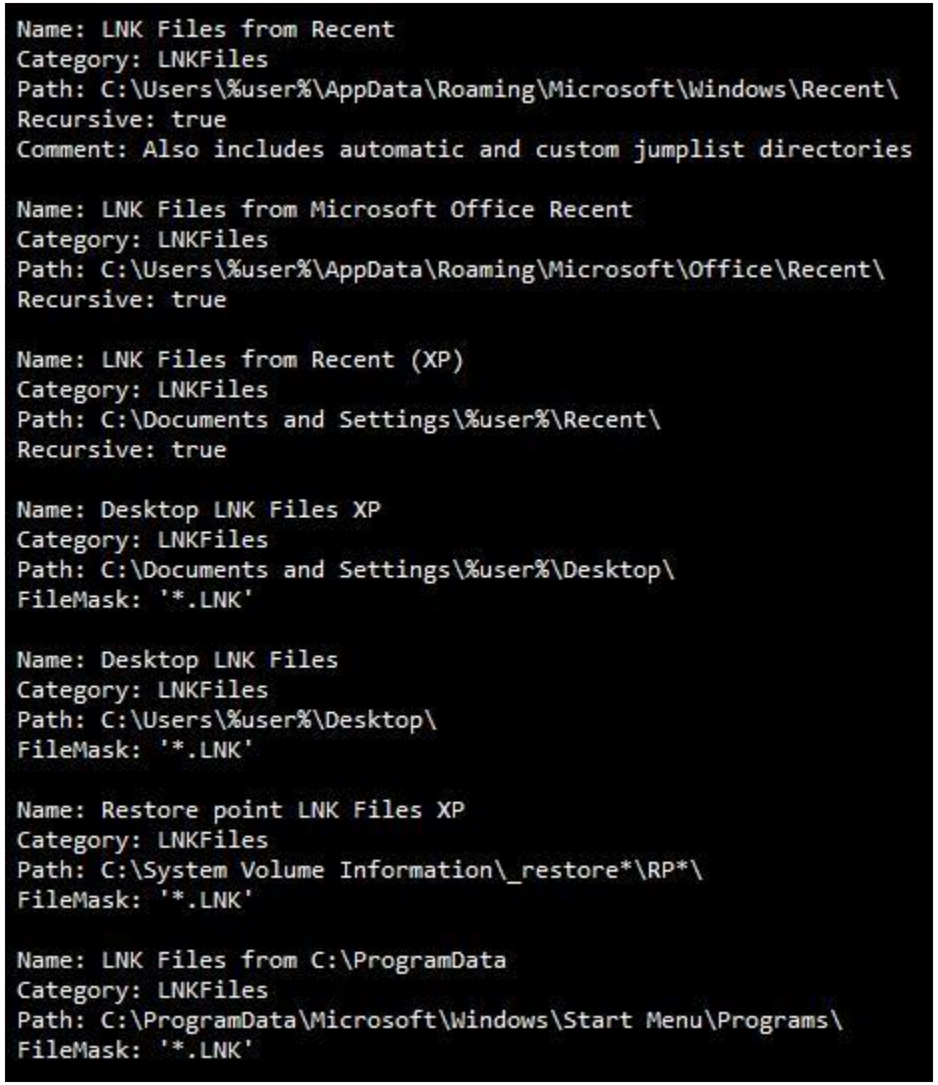
c:\Users\AndrewSager\Downloads\ZimmermanTools>

Investigative Process Using KAPE

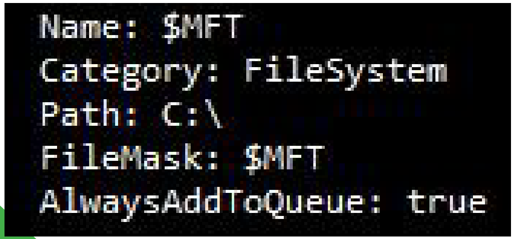
Acquiring Necessary Artifacts Using KAPE

Featuring Snips from $MFT, $J, and LNKFilesAndJumpLists Targets

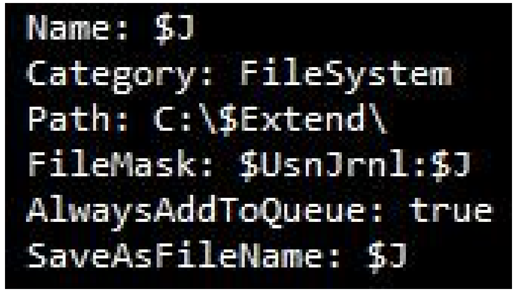
- Artifacts needed:
  - $MFT – index of all files on an NTFS-formatted file system
- Artifacts that aren't needed but can be helpful
  - $J – journal of changes to files (does not store actual file contents that are changed)
  - .LNK files – evidence of file access, stores timestamps of the target file within
- All of the above (and more) are acquired using the KapeTriage Compound Target



```
Name: LNK Files from Recent
Category: LNKFiles
Path: C:\Users\%user%\AppData\Roaming\Microsoft\Windows\Recent\
Recursive: true
Comment: Also includes automatic and custom jumplist directories

Name: LNK Files from Microsoft Office Recent
Category: LNKFiles
Path: C:\Users\%user%\AppData\Roaming\Microsoft\Office\Recent\
Recursive: true

Name: LNK Files from Recent (XP)
Category: LNKFiles
Path: C:\Documents and Settings\%user%\Recent\
Recursive: true

Name: Desktop LNK Files XP
Category: LNKFiles
Path: C:\Documents and Settings\%user%\Desktop\
FileMask: '*.LNK'

Name: Desktop LNK Files
Category: LNKFiles
Path: C:\Users\%user%\Desktop\
FileMask: '*.LNK'

Name: Restore point LNK Files XP
Category: LNKFiles
Path: C:\System Volume Information\_restore*\RP*\
FileMask: '*.LNK'

Name: LNK Files from C:\ProgramData
Category: LNKFiles
Path: C:\ProgramData\Microsoft\Windows\Start Menu\Programs\
FileMask: '*.LNK'
```



```
Name: $MFT
Category: FileSystem
Path: C:\
FileMask: $MFT
AlwaysAddToQueue: true
```

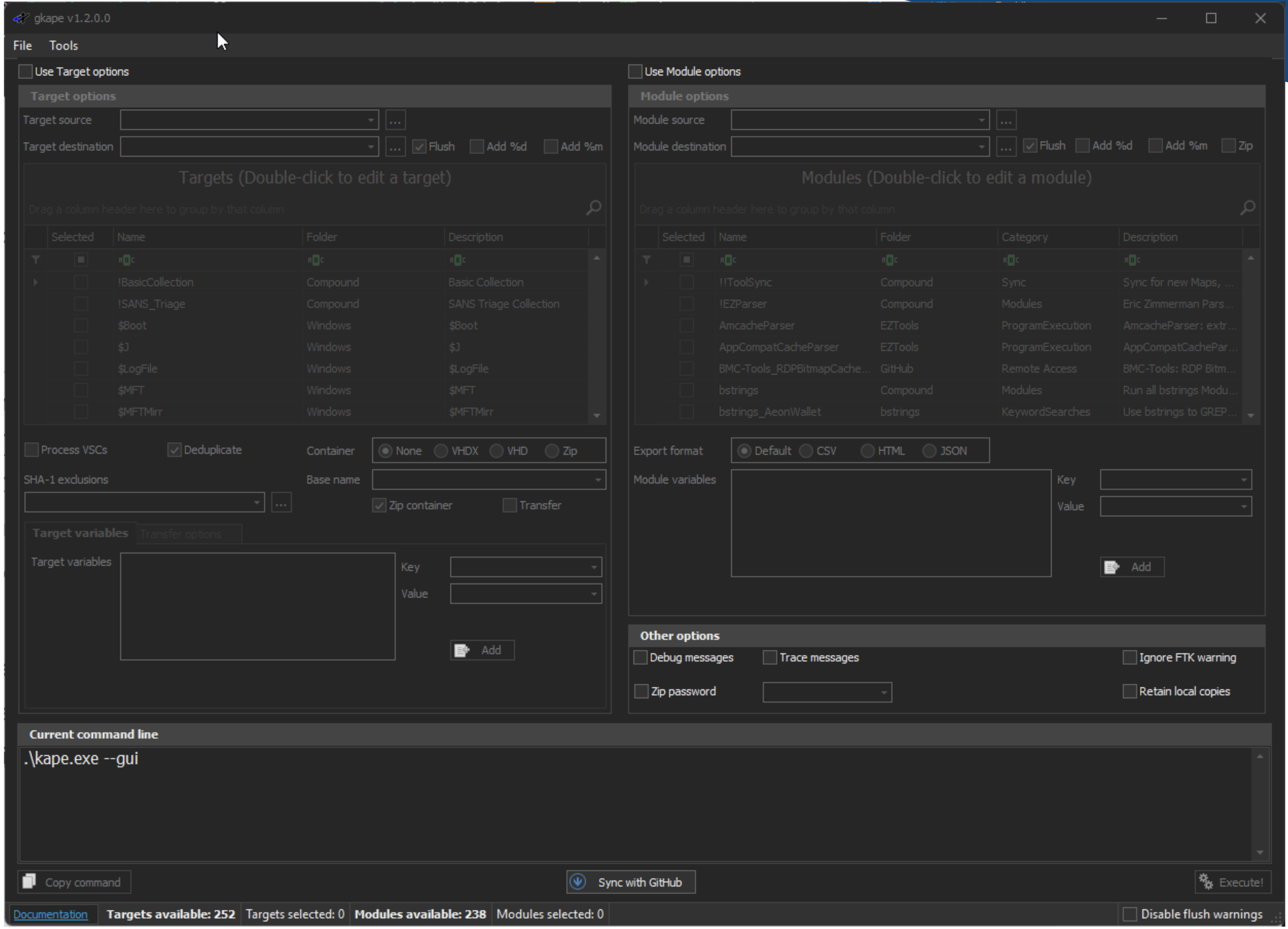


```
Name: $J
Category: FileSystem
Path: C:\$Extend\
FileMask: $UsnJrnl:$J
AlwaysAddToQueue: true
SaveAsFileName: $J
```

KAPE Workflow

Using gkape

- My common workflow:
  - KapeTriage Target
  - !EZParser Module
- Grabs (most) everything I need to answer 95% of the questions I initially have
- Produces output for most artifacts pulled using KapeTriage Compound Target
- Every EZ Tool is put to work so long as the corresponding artifact exists



□ Use Target options

Target options

Target source

Target destination ☑ Flush □ Add %d □ Add %m

Targets (Double-click to edit a target)

Drag a column header here to group by that column

Selected	Name	Folder	Description
□	!BasicCollection	Compound	Basic Collection
□	!SANS_Triage	Compound	SANS Triage Collection
□	$Boot	Windows	$Boot
□	$J	Windows	$J
□	$LogFile	Windows	$LogFile
□	$MFT	Windows	$MFT
□	$MFTMirr	Windows	$MFTMirr

□ Process VSCs ☑ Deduplicate

Container ● None ○ VHDX ○ VHD ○ Zip

SHA-1 exclusions

Base name

☑ Zip container □ Transfer

Target variables Transfer options

Target variables

Key

Value

□ Use Module options

Module options

Module source

Module destination ☑ Flush □ Add %d □ Add %m □ Zip

Modules (Double-click to edit a module)

Drag a column header here to group by that column

Selected	Name	Folder	Category	Description
□	!!ToolSync	Compound	Sync	Sync for new Maps, ...
□	!EZParser	Compound	Modules	Eric Zimmerman Pars...
□	AmcacheParser	EZTools	ProgramExecution	AmcacheParser: extr...
□	AppCompatCacheParser	EZTools	ProgramExecution	AppCompatCachePar...
□	BMC-Tools_RDPBitmapCache...	GitHub	Remote Access	BMC-Tools: RDP Bitm...
□	bstrings	Compound	Modules	Run all bstrings Modu...
□	bstrings_AeonWallet	bstrings	KeywordSearches	Use bstrings to GREP...

Export format ● Default ○ CSV ○ HTML ○ JSON

Module variables

Key

Value

Other options

□ Debug messages □ Trace messages □ Ignore FTK warning

□ Zip password □ Retain local copies

Current command line

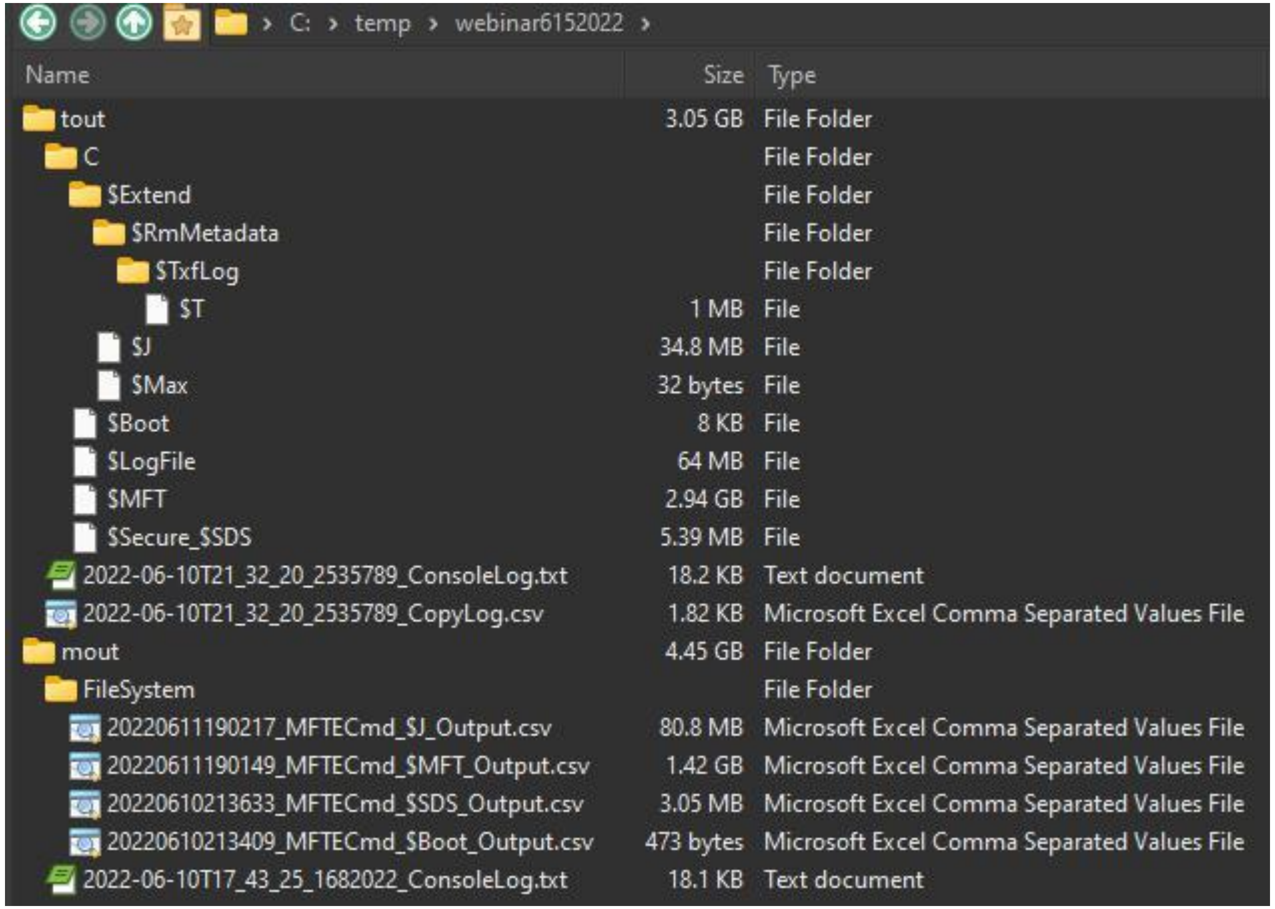
```
.\kape.exe --gui
```

Targets available: 252 Targets selected: 0 **Modules available: 238** Modules selected: 0 □ Disable flush warnings

KAPE Output

tout and mout folders

- tout
  - Target Destination location
  - Will have the files we acquired using KAPE Targets with recreated directories
- mout
  - Module Destination location
  - Will have the parsed output from the $MFT, $J, and more
- With this workflow, we have the raw files and parsed output for us to analyze in Timeline Explorer
- Full disclosure: this example was only the FileSystem/MFTECmd, NOT KapeTriage/!EZParser



C: › temp › webinar6152022 ›

Name	Size	Type
tout	3.05 GB	File Folder
C		File Folder
$Extend		File Folder
$RmMetadata		File Folder
$TxfLog		File Folder
$T	1 MB	File
$J	34.8 MB	File
$Max	32 bytes	File
$Boot	8 KB	File
$LogFile	64 MB	File
$MFT	2.94 GB	File
$Secure_$SDS	5.39 MB	File
2022-06-10T21_32_20_2535789_ConsoleLog.txt	18.2 KB	Text document
2022-06-10T21_32_20_2535789_CopyLog.csv	1.82 KB	Microsoft Excel Comma Separated Values File
mout	4.45 GB	File Folder
FileSystem		File Folder
20220611190217_MFTECmd_$J_Output.csv	80.8 MB	Microsoft Excel Comma Separated Values File
20220611190149_MFTECmd_$MFT_Output.csv	1.42 GB	Microsoft Excel Comma Separated Values File
20220610213633_MFTECmd_$SDS_Output.csv	3.05 MB	Microsoft Excel Comma Separated Values File
20220610213409_MFTECmd_$Boot_Output.csv	473 bytes	Microsoft Excel Comma Separated Values File
2022-06-10T17_43_25_1682022_ConsoleLog.txt	18.1 KB	Text document

Timeline Explorer

Brief introduction to Timeline Explorer

- Our KAPE workflow acquired files and generated output from those files for us to analyze
- Timeline Explorer is the best companion for the examiners who primarily analyze CSV output from EZ Tools or any other tool
- Built by an examiner, for examiners
- Opens much larger files than Excel can handle, plus many other quality of life features
- Timeline Explorer - AboutDFIR - The Definitive Compendium Project

Timeline Explorer v2.0.0.1

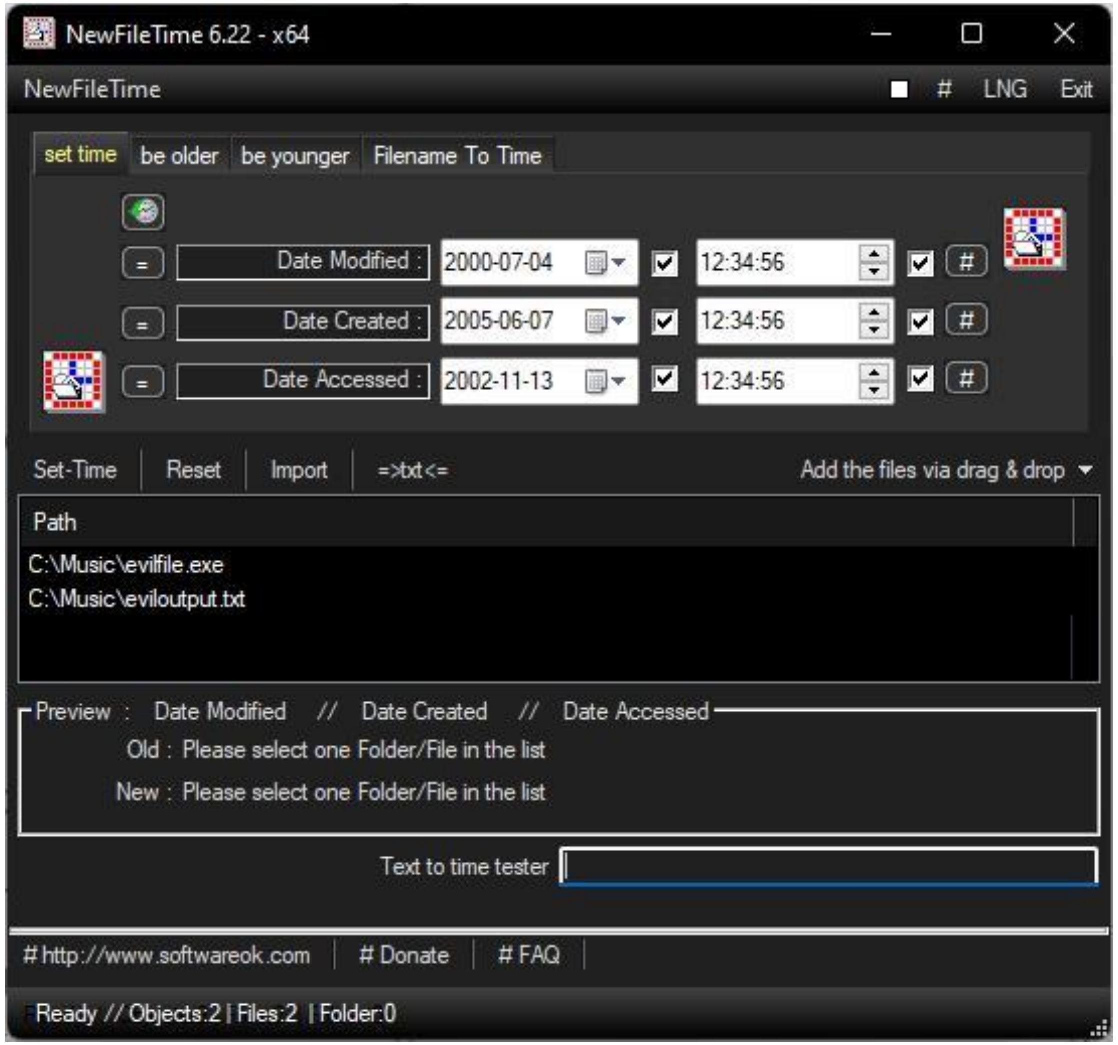
Analysis with Timeline Explorer:

NewFileTime Scenario

NewFileTime Timestomping Analysis: $MFT

Change Attributes settings and subsequent $MFT output from MFTECmd

- Timestamps in NewFileTime (right) are in Local Time (EST)
- Timestamps in MFTECmd output are in UTC
  - i.e., 12:34:56 (EST) -> 16:34:56 (UTC)
- Temporal analysis of $MFT timestamps would show evilfile.exe and eviloutput.txt as created much earlier than reality (next slide)



Date Modified : 2000-07-04 ☑ 12:34:56 ☑

Date Created : 2005-06-07 ☑ 12:34:56 ☑

Date Accessed : 2002-11-13 ☑ 12:34:56 ☑

Path

C:\Music\evilfile.exe

C:\Music\eviloutput.txt

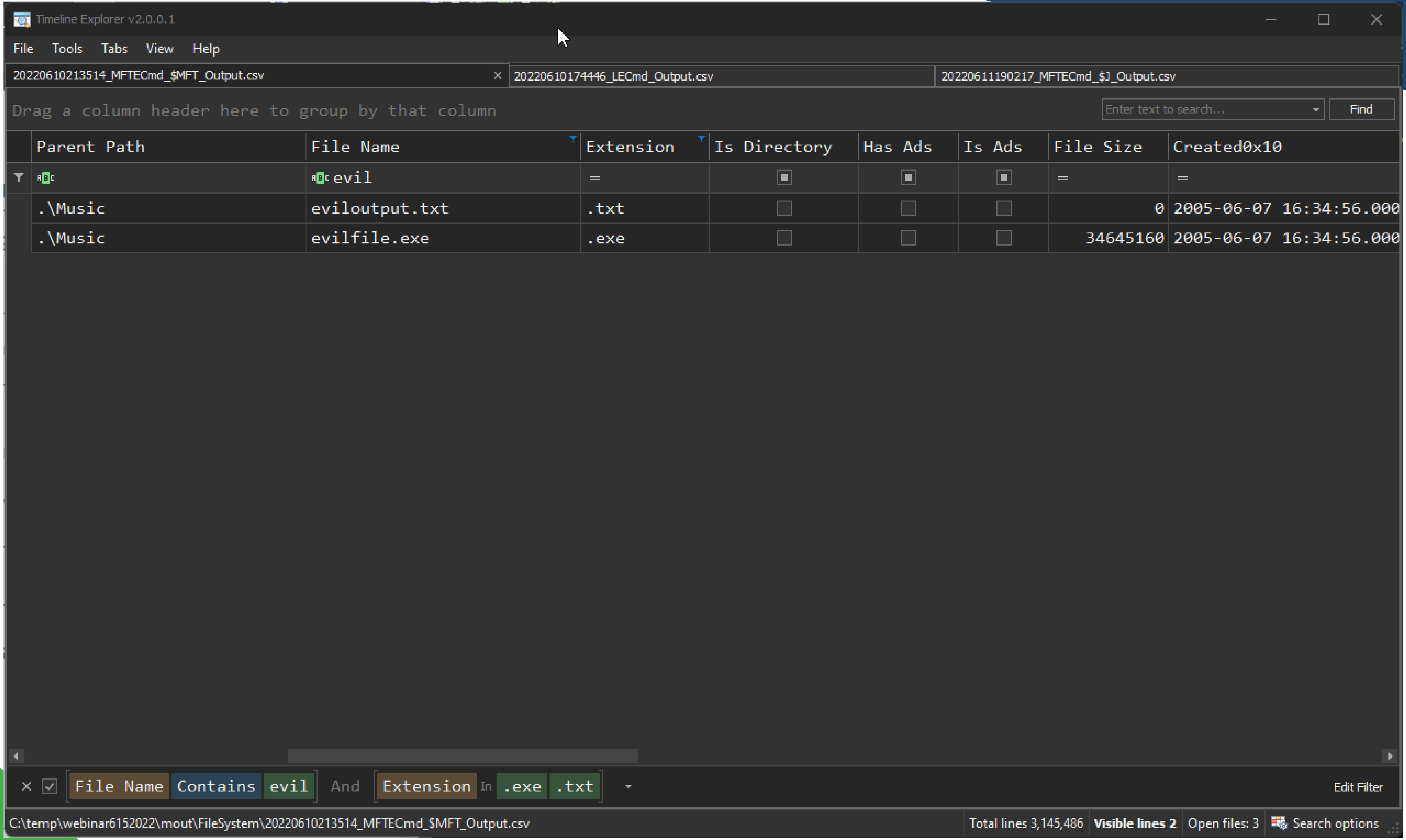
Preview : Date Modified // Date Created // Date Accessed

Old : Please select one Folder/File in the list

New : Please select one Folder/File in the list

Text to time tester

Ready // Objects:2 | Files:2 | Folder:0



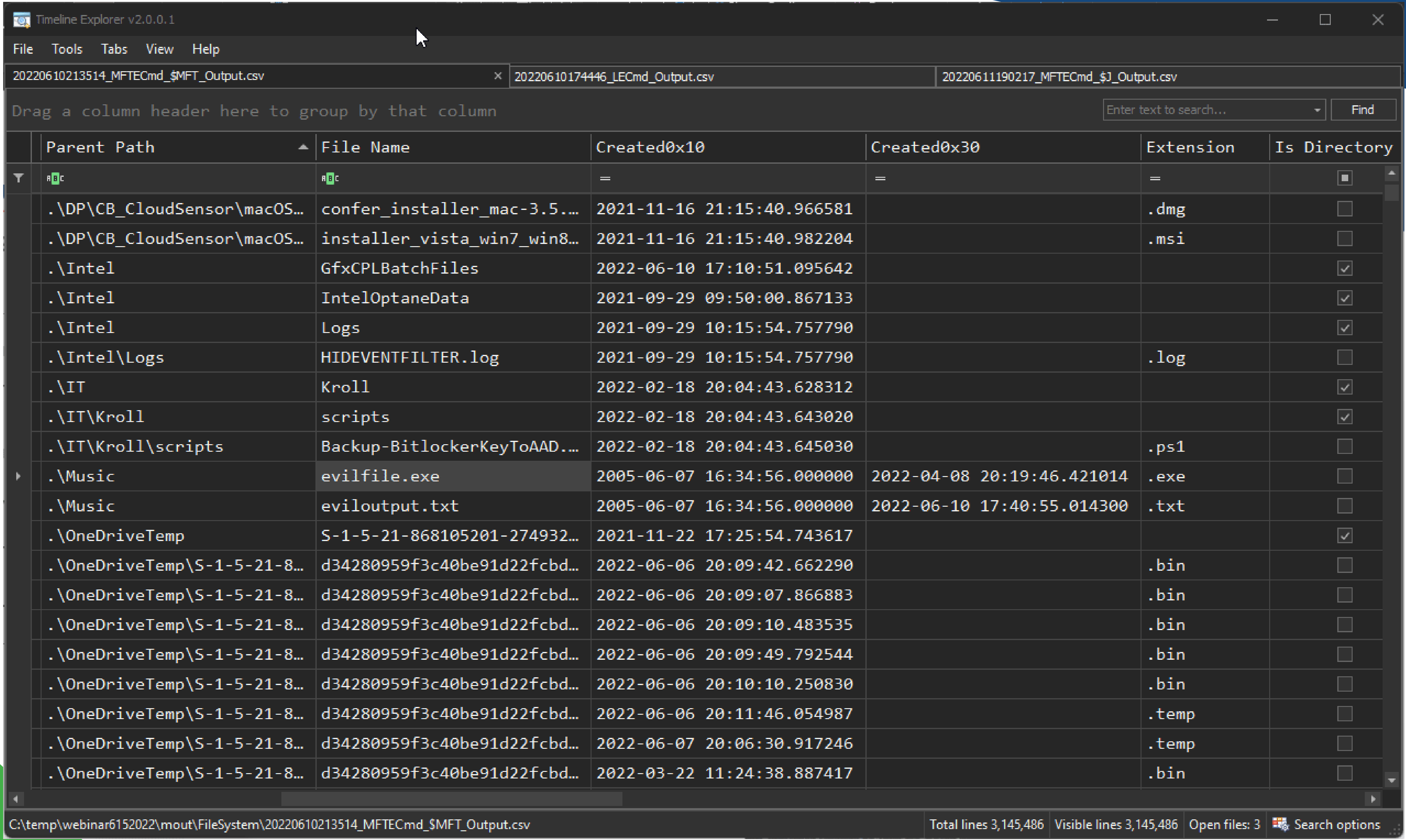
20220610213514_MFTECmd_$MFT_Output.csv | 20220610174446_LECmd_Output.csv | 20220611190217_MFTECmd_$J_Output.csv

Drag a column header here to group by that column

Parent Path	File Name	Extension	Is Directory	Has Ads	Is Ads	File Size	Created0x10
	evil	=	▣	▣	▣	=	=
.\Music	eviloutput.txt	.txt	□	□	□	0	2005-06-07 16:34:56.000
.\Music	evilfile.exe	.exe	□	□	□	34645160	2005-06-07 16:34:56.000

✕ ☑ File Name Contains evil And Extension In .exe .txt

C:\temp\webinar6152022\mout\FileSystem\20220610213514_MFTECmd_$MFT_Output.csv | Total lines 3,145,486 | **Visible lines 2** | Open files: 3 | Search options


Timeline Explorer v2.0.0.1
File
Tools
Tabs
View
Help
20220610213514_MFTECmd_$MFT_Output.csv
20220610174446_LECmd_Output.csv
20220611190217_MFTECmd_$J_Output.csv
Drag a column header here to group by that column
Enter text to search...
Find

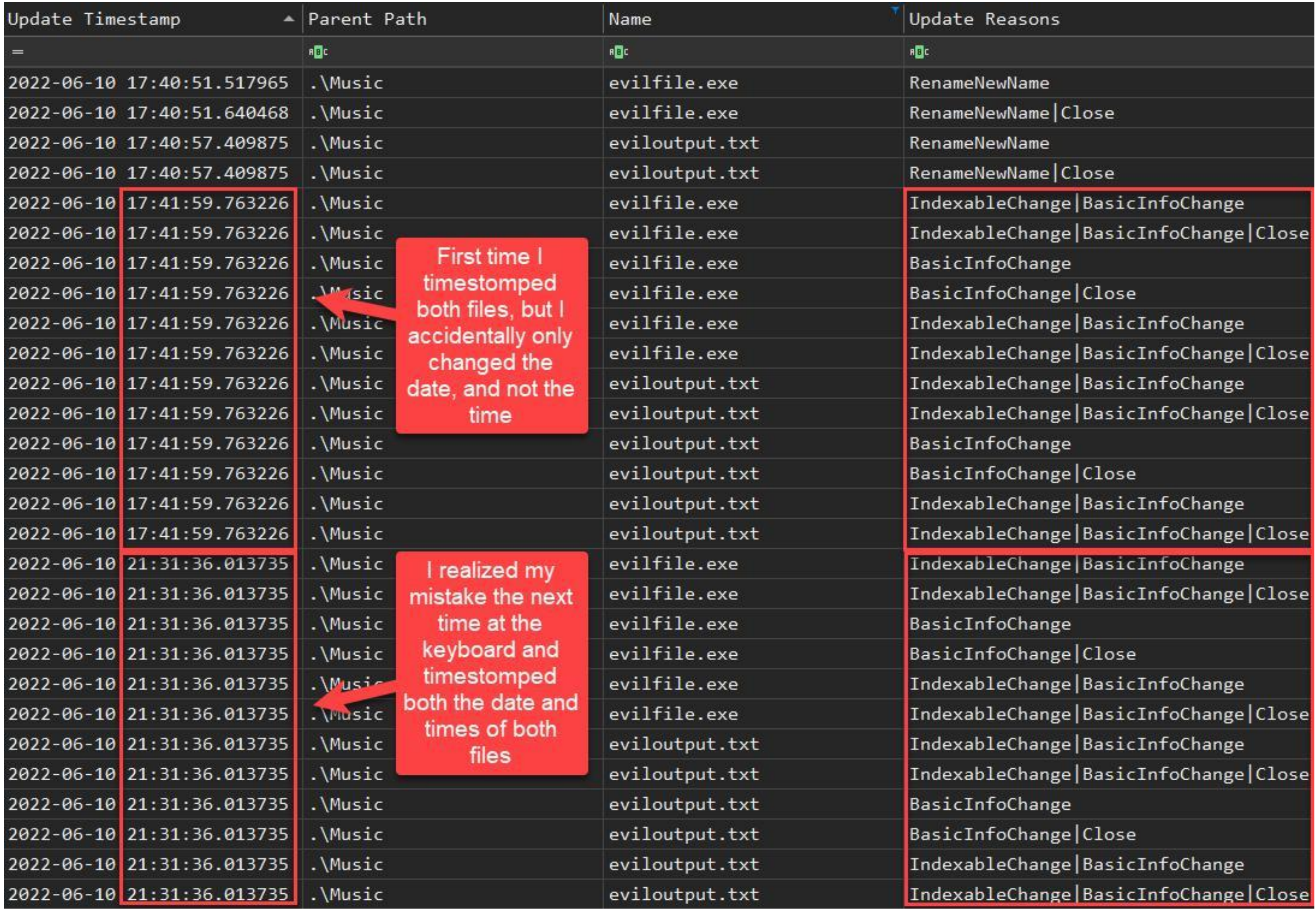
Parent Path	File Name	Created0x10	Created0x30	Extension	Is Directory
.\DP\CB_CloudSensor\macOS...	confer_installer_mac-3.5....	2021-11-16 21:15:40.966581		.dmg	☐
.\DP\CB_CloudSensor\macOS...	installer_vista_win7_win8...	2021-11-16 21:15:40.982204		.msi	☐
.\Intel	GfxCPLBatchFiles	2022-06-10 17:10:51.095642			☑
.\Intel	IntelOptaneData	2021-09-29 09:50:00.867133			☑
.\Intel	Logs	2021-09-29 10:15:54.757790			☑
.\Intel\Logs	HIDEVENTFILTER.log	2021-09-29 10:15:54.757790		.log	☐
.\IT	Kroll	2022-02-18 20:04:43.628312			☑
.\IT\Kroll	scripts	2022-02-18 20:04:43.643020			☑
.\IT\Kroll\scripts	Backup-BitlockerKeyToAAD....	2022-02-18 20:04:43.645030		.ps1	☐
.\Music	evilfile.exe	2005-06-07 16:34:56.000000	2022-04-08 20:19:46.421014	.exe	☐
.\Music	eviloutput.txt	2005-06-07 16:34:56.000000	2022-06-10 17:40:55.014300	.txt	☐
.\OneDriveTemp	S-1-5-21-868105201-274932...	2021-11-22 17:25:54.743617			☑
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-06 20:09:42.662290		.bin	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-06 20:09:07.866883		.bin	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-06 20:09:10.483535		.bin	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-06 20:09:49.792544		.bin	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-06 20:10:10.250830		.bin	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-06 20:11:46.054987		.temp	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-06-07 20:06:30.917246		.temp	☐
.\OneDriveTemp\S-1-5-21-8...	d34280959f3c40be91d22fcbd...	2022-03-22 11:24:38.887417		.bin	☐

C:\temp\webinar6152022\mout\FileSystem\20220610213514_MFTECmd_$MFT_Output.csv | Total lines 3,145,486 | Visible lines 3,145,486 | Open files: 3 | Search options

NewFileTime Timestomping Analysis: $J

$J parsed output from MFTECmd

- $J is one of my favorite artifacts!
- I've seen the $J cover anywhere from a few hours to 2+ years of file system activity
  – Sometimes it doesn't exist ☹
- If it exists and covers your time period of interest, it'll be a great asset to your investigation
- On a live system:
  – fsutil.exe usn queryJournal C:
- On an image:
  – C:\$Extend\$UsnJrnl:$J



Update Timestamp	Parent Path	Name	Update Reasons
2022-06-10 17:40:51.517965	.\Music	evilfile.exe	RenameNewName
2022-06-10 17:40:51.640468	.\Music	evilfile.exe	RenameNewName|Close
2022-06-10 17:40:57.409875	.\Music	eviloutput.txt	RenameNewName
2022-06-10 17:40:57.409875	.\Music	eviloutput.txt	RenameNewName|Close
2022-06-10 17:41:59.763226	.\Music	evilfile.exe	IndexableChange|BasicInfoChange
2022-06-10 17:41:59.763226	.\Music	evilfile.exe	IndexableChange|BasicInfoChange|Close
2022-06-10 17:41:59.763226	.\Music	evilfile.exe	BasicInfoChange
2022-06-10 17:41:59.763226	.\Music	evilfile.exe	BasicInfoChange|Close
2022-06-10 17:41:59.763226	.\Music	evilfile.exe	IndexableChange|BasicInfoChange
2022-06-10 17:41:59.763226	.\Music	evilfile.exe	IndexableChange|BasicInfoChange|Close
2022-06-10 17:41:59.763226	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange
2022-06-10 17:41:59.763226	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange|Close
2022-06-10 17:41:59.763226	.\Music	eviloutput.txt	BasicInfoChange
2022-06-10 17:41:59.763226	.\Music	eviloutput.txt	BasicInfoChange|Close
2022-06-10 17:41:59.763226	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange
2022-06-10 17:41:59.763226	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange|Close
2022-06-10 21:31:36.013735	.\Music	evilfile.exe	IndexableChange|BasicInfoChange
2022-06-10 21:31:36.013735	.\Music	evilfile.exe	IndexableChange|BasicInfoChange|Close
2022-06-10 21:31:36.013735	.\Music	evilfile.exe	BasicInfoChange
2022-06-10 21:31:36.013735	.\Music	evilfile.exe	BasicInfoChange|Close
2022-06-10 21:31:36.013735	.\Music	evilfile.exe	IndexableChange|BasicInfoChange
2022-06-10 21:31:36.013735	.\Music	evilfile.exe	IndexableChange|BasicInfoChange|Close
2022-06-10 21:31:36.013735	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange
2022-06-10 21:31:36.013735	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange|Close
2022-06-10 21:31:36.013735	.\Music	eviloutput.txt	BasicInfoChange
2022-06-10 21:31:36.013735	.\Music	eviloutput.txt	BasicInfoChange|Close
2022-06-10 21:31:36.013735	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange
2022-06-10 21:31:36.013735	.\Music	eviloutput.txt	IndexableChange|BasicInfoChange|Close

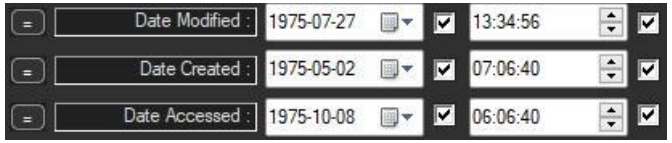
First time I timestomped both files, but I accidentally only changed the date, and not the time

I realized my mistake the next time at the keyboard and timestomped both the date and times of both files

NewFileTime Timestomping Analysis: LNK Files

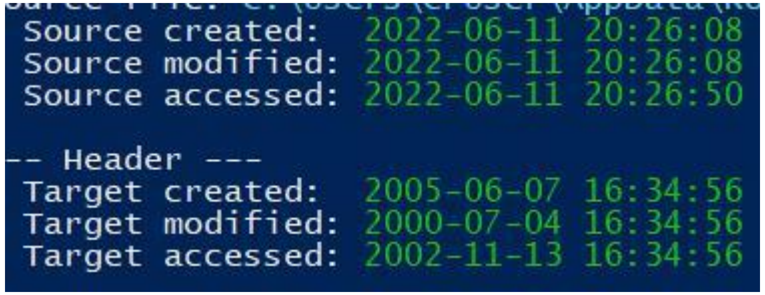
LNK file parsed output from LECmd

- After creating eviloutput.txt, no LNK file exists (yet)
- Opening eviloutput.txt the first time will generate a LNK file
- In this scenario, I timestomped eviloutput.txt before opening it for the first time
  - 1st: Parsed LNK file upon opening eviloutput.txt the first time
  - 2nd: Parsed LNK file upon timestomping again but NOT opening
  - 3rd: Parsed LNK file upon opening eviloutput.txt AFTER timestomping
  - 4th: Parsed LNK file upon timestomping files to 1975 but NOT opening



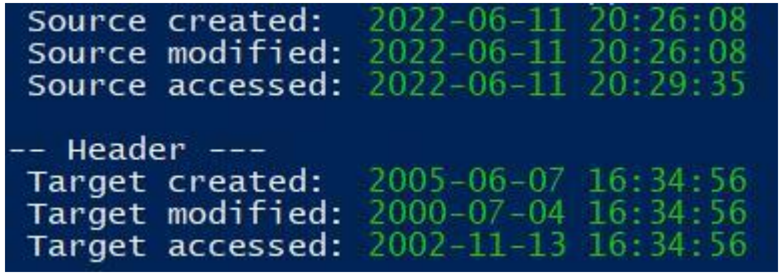
Date Modified :	1975-07-27	☑	13:34:56	☑
Date Created :	1975-05-02	☑	07:06:40	☑
Date Accessed :	1975-10-08	☑	06:06:40	☑

  - 5th: Cycle keeps going
- We find that LNK files refresh metadata upon opening of target file AFTER timestomping occurred



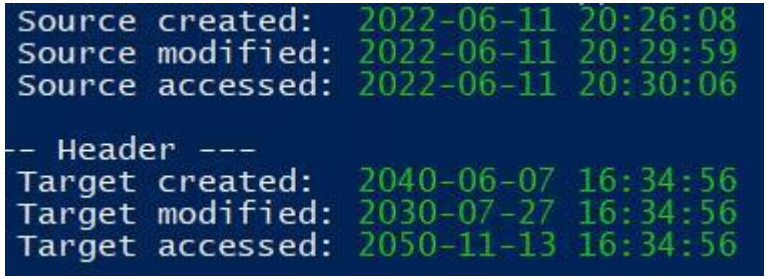
```
 Source created:   2022-06-11 20:26:08
 Source modified:  2022-06-11 20:26:08
 Source accessed:  2022-06-11 20:26:50

-- Header ---
 Target created:   2005-06-07 16:34:56
 Target modified:  2000-07-04 16:34:56
 Target accessed:  2002-11-13 16:34:56
```

1



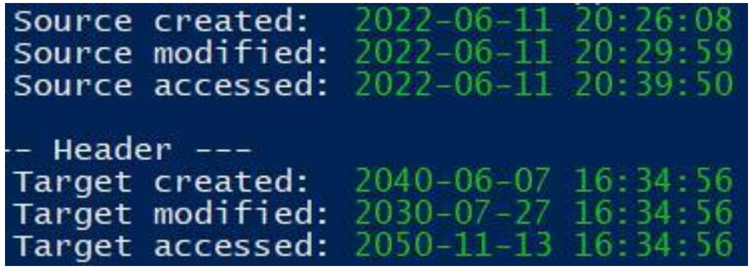
```
 Source created:   2022-06-11 20:26:08
 Source modified:  2022-06-11 20:26:08
 Source accessed:  2022-06-11 20:29:35

-- Header ---
 Target created:   2005-06-07 16:34:56
 Target modified:  2000-07-04 16:34:56
 Target accessed:  2002-11-13 16:34:56
```

2



```
 Source created:   2022-06-11 20:26:08
 Source modified:  2022-06-11 20:29:59
 Source accessed:  2022-06-11 20:30:06

-- Header ---
 Target created:   2040-06-07 16:34:56
 Target modified:  2030-07-27 16:34:56
 Target accessed:  2050-11-13 16:34:56
```

3



```
Source created:   2022-06-11 20:26:08
Source modified:  2022-06-11 20:29:59
Source accessed:  2022-06-11 20:39:50

- Header ---
Target created:   2040-06-07 16:34:56
Target modified:  2030-07-27 16:34:56
Target accessed:  2050-11-13 16:34:56
```

4

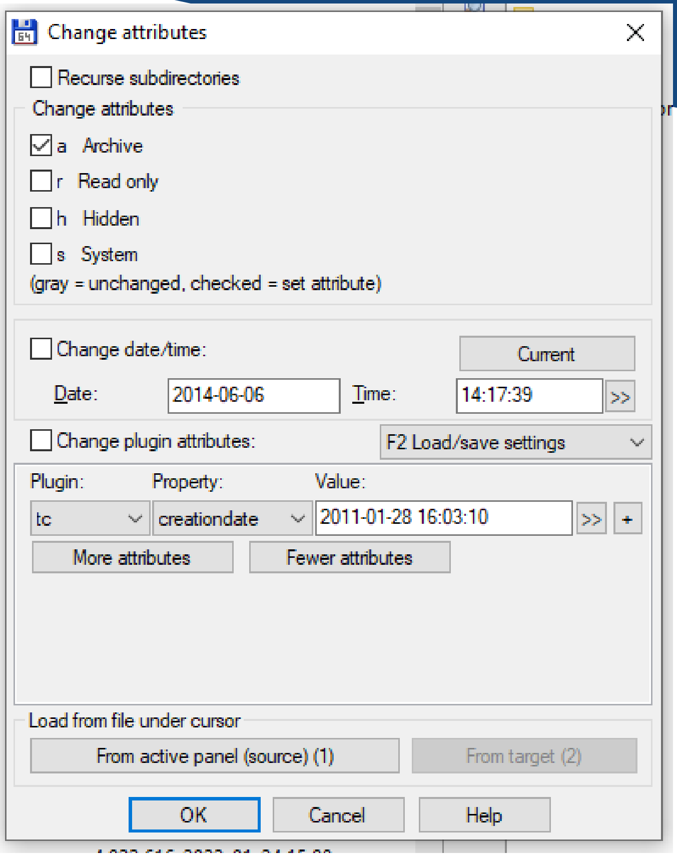
Analysis with Timeline Explorer:

Total Commander Scenario

Total Commander Timestomping Analysis: $MFT

Change Attributes settings and subsequent $MFT output from MFTECmd

- Timestamps in Total Commander (right) are in Local Time (EST)
- Timestamps in MFTECmd output (below) are in UTC
- Notice how I timestomped bstrings.exe back to 2011 (right)
- Temporal analysis of $MFT timestamps would show bstrings.exe as created a decade prior to my time period of interest (below)
- Total Commander and **most** other tools only can modify the 0x10 timestamps, but not the 0x30 timestamps



Change attributes

□ Recurse subdirectories

Change attributes

☑ a Archive

□ r Read only

□ h Hidden

□ s System

(gray = unchanged, checked = set attribute)

□ Change date/time: [Current]

Date: 2014-06-06 Time: 14:17:39 [>>]

□ Change plugin attributes: [F2 Load/save settings]

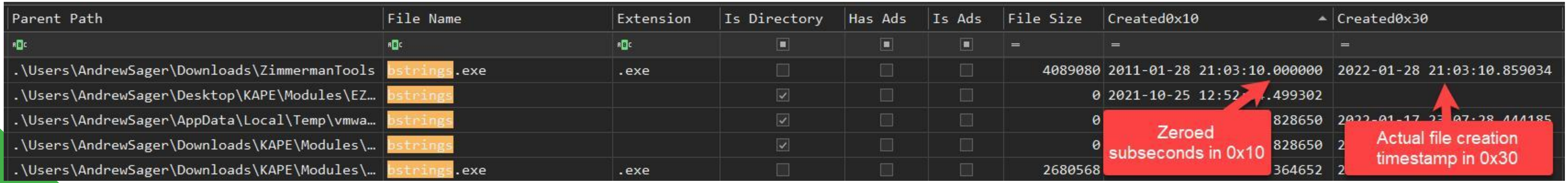
Plugin: tc Property: creationdate Value: 2011-01-28 16:03:10 [>>] [+]

[More attributes] [Fewer attributes]

Load from file under cursor

[From active panel (source) (1)] [From target (2)]

[OK] [Cancel] [Help]



Parent Path	File Name	Extension	Is Directory	Has Ads	Is Ads	File Size	Created0x10	Created0x30
.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	.exe	□	□	□	4089080	2011-01-28 21:03:10.000000	2022-01-28 21:03:10.859034
.\Users\AndrewSager\Desktop\KAPE\Modules\EZ…	bstrings		☑	□	□	0	2021-10-25 12:52[illegible].499302	
.\Users\AndrewSager\AppData\Local\Temp\vmwa…	bstrings		☑	□	□	0	[illegible]828650	2022-01-17 2[illegible]07:28.444185
.\Users\AndrewSager\Downloads\KAPE\Modules\…	bstrings		☑	□	□	0	[illegible]828650	2[illegible]
.\Users\AndrewSager\Downloads\KAPE\Modules\…	bstrings.exe	.exe	□	□	□	2680568	[illegible]364652	2[illegible]

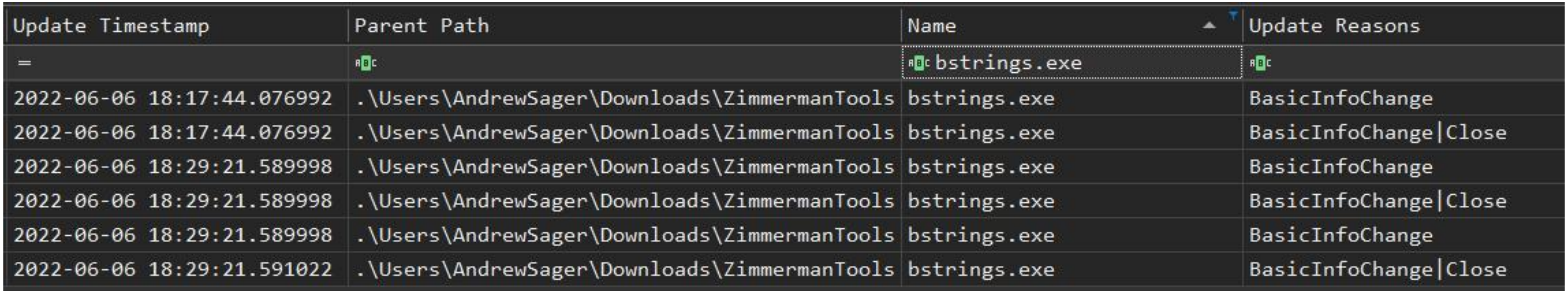
Zeroed subseconds in 0x10

Actual file creation timestamp in 0x30

Total Commander Timestomping Analysis: $J

$J parsed output from MFTECmd

- Identical fact pattern from the NewFileTime Timestomping example
- $J provides an awesome play-by-play of what happens on the file system

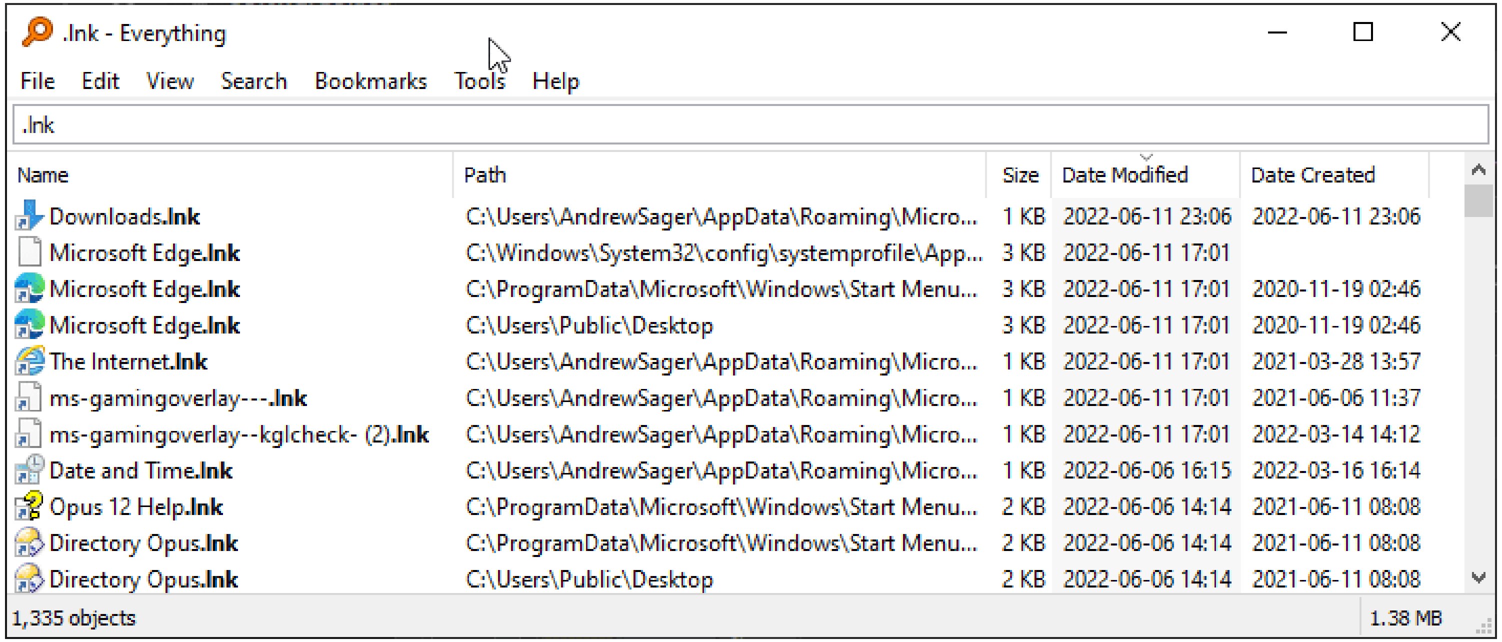


Update Timestamp	Parent Path	Name	Update Reasons
=		bstrings.exe	
2022-06-06 18:17:44.076992	.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	BasicInfoChange
2022-06-06 18:17:44.076992	.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	BasicInfoChange|Close
2022-06-06 18:29:21.589998	.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	BasicInfoChange
2022-06-06 18:29:21.589998	.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	BasicInfoChange|Close
2022-06-06 18:29:21.589998	.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	BasicInfoChange
2022-06-06 18:29:21.591022	.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	BasicInfoChange|Close

Total Commander Timestomping Analysis: LNK Files

LNK file parsed output from LECmd

- Same behavior as previous
- Key takeaways
  - Source timestamps refer to LNK file itself
  - Target timestamps refer to the file the LNK file is pointing to
  - LNK files are only created upon the first time a file is opened
  - LNK file is created for the file opened itself and the parent folder (right)
  - SourceModified provides the Last Modified timestamp of the LNK file itself, indicating last time the target file was opened
  - Timestamps within the LNK file are only updated once the target file is opened
  - LNK file timestamps do not appear to be refreshed in real time as files are timestomped (file access indicator)

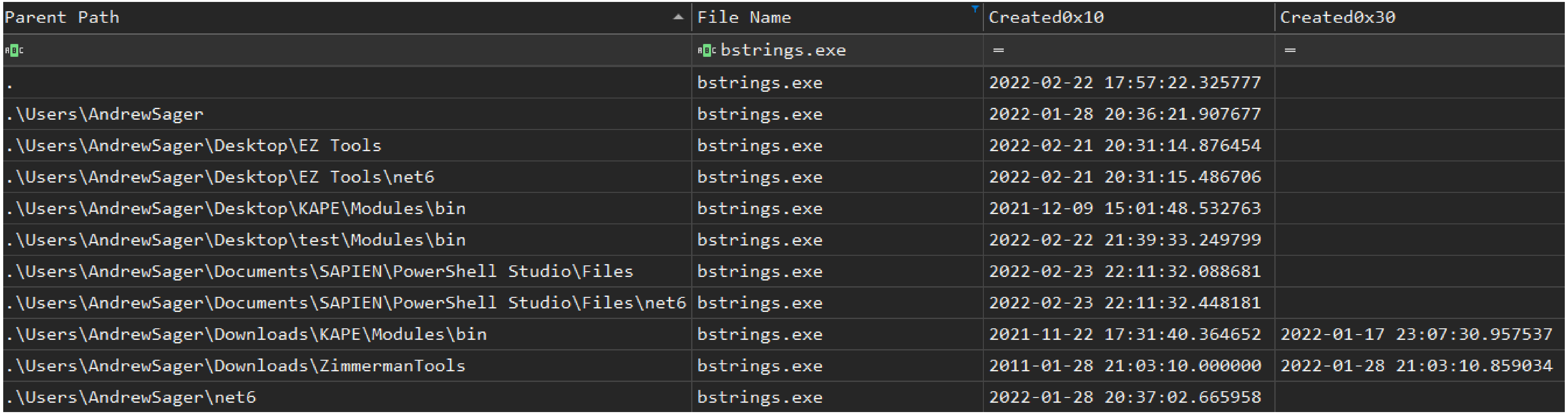


Name	Path	Size	Date Modified	Date Created
Downloads.lnk	C:\Users\AndrewSager\AppData\Roaming\Micro...	1 KB	2022-06-11 23:06	2022-06-11 23:06
Microsoft Edge.lnk	C:\Windows\System32\config\systemprofile\App...	3 KB	2022-06-11 17:01	
Microsoft Edge.lnk	C:\ProgramData\Microsoft\Windows\Start Menu...	3 KB	2022-06-11 17:01	2020-11-19 02:46
Microsoft Edge.lnk	C:\Users\Public\Desktop	3 KB	2022-06-11 17:01	2020-11-19 02:46
The Internet.lnk	C:\Users\AndrewSager\AppData\Roaming\Micro...	1 KB	2022-06-11 17:01	2021-03-28 13:57
ms-gamingoverlay---.lnk	C:\Users\AndrewSager\AppData\Roaming\Micro...	1 KB	2022-06-11 17:01	2021-06-06 11:37
ms-gamingoverlay--kglcheck- (2).lnk	C:\Users\AndrewSager\AppData\Roaming\Micro...	1 KB	2022-06-11 17:01	2022-03-14 14:12
Date and Time.lnk	C:\Users\AndrewSager\AppData\Roaming\Micro...	1 KB	2022-06-06 16:15	2022-03-16 16:14
Opus 12 Help.lnk	C:\ProgramData\Microsoft\Windows\Start Menu...	2 KB	2022-06-06 14:14	2021-06-11 08:08
Directory Opus.lnk	C:\ProgramData\Microsoft\Windows\Start Menu...	2 KB	2022-06-06 14:14	2021-06-11 08:08
Directory Opus.lnk	C:\Users\Public\Desktop	2 KB	2022-06-06 14:14	2021-06-11 08:08

1,335 objects 1.38 MB

Pop Quiz!

Can you spot the timestomped bstrings.exe?



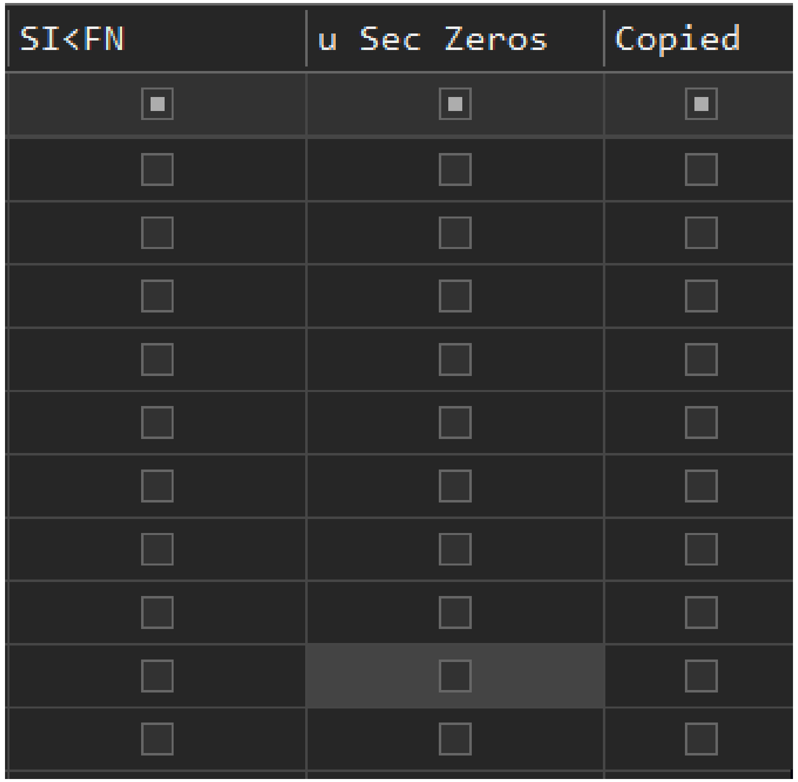
Parent Path	File Name	Created0x10	Created0x30
ABC	ABC bstrings.exe	=	=
.	bstrings.exe	2022-02-22 17:57:22.325777	
.\Users\AndrewSager	bstrings.exe	2022-01-28 20:36:21.907677	
.\Users\AndrewSager\Desktop\EZ Tools	bstrings.exe	2022-02-21 20:31:14.876454	
.\Users\AndrewSager\Desktop\EZ Tools\net6	bstrings.exe	2022-02-21 20:31:15.486706	
.\Users\AndrewSager\Desktop\KAPE\Modules\bin	bstrings.exe	2021-12-09 15:01:48.532763	
.\Users\AndrewSager\Desktop\test\Modules\bin	bstrings.exe	2022-02-22 21:39:33.249799	
.\Users\AndrewSager\Documents\SAPIEN\PowerShell Studio\Files	bstrings.exe	2022-02-23 22:11:32.088681	
.\Users\AndrewSager\Documents\SAPIEN\PowerShell Studio\Files\net6	bstrings.exe	2022-02-23 22:11:32.448181	
.\Users\AndrewSager\Downloads\KAPE\Modules\bin	bstrings.exe	2021-11-22 17:31:40.364652	2022-01-17 23:07:30.957537
.\Users\AndrewSager\Downloads\ZimmermanTools	bstrings.exe	2011-01-28 21:03:10.000000	2022-01-28 21:03:10.859034
.\Users\AndrewSager\net6	bstrings.exe	2022-01-28 20:37:02.665958	

Important Considerations

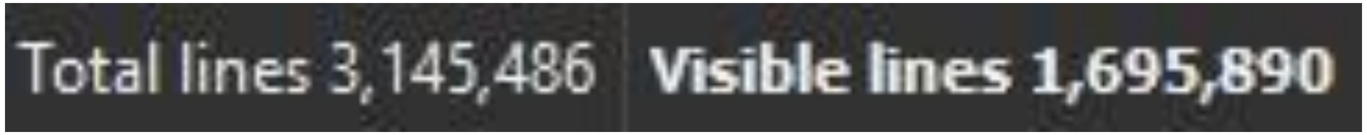
Lots of False Positives

If judging ONLY by .0000000 subseconds, you'll find a lot of false positives

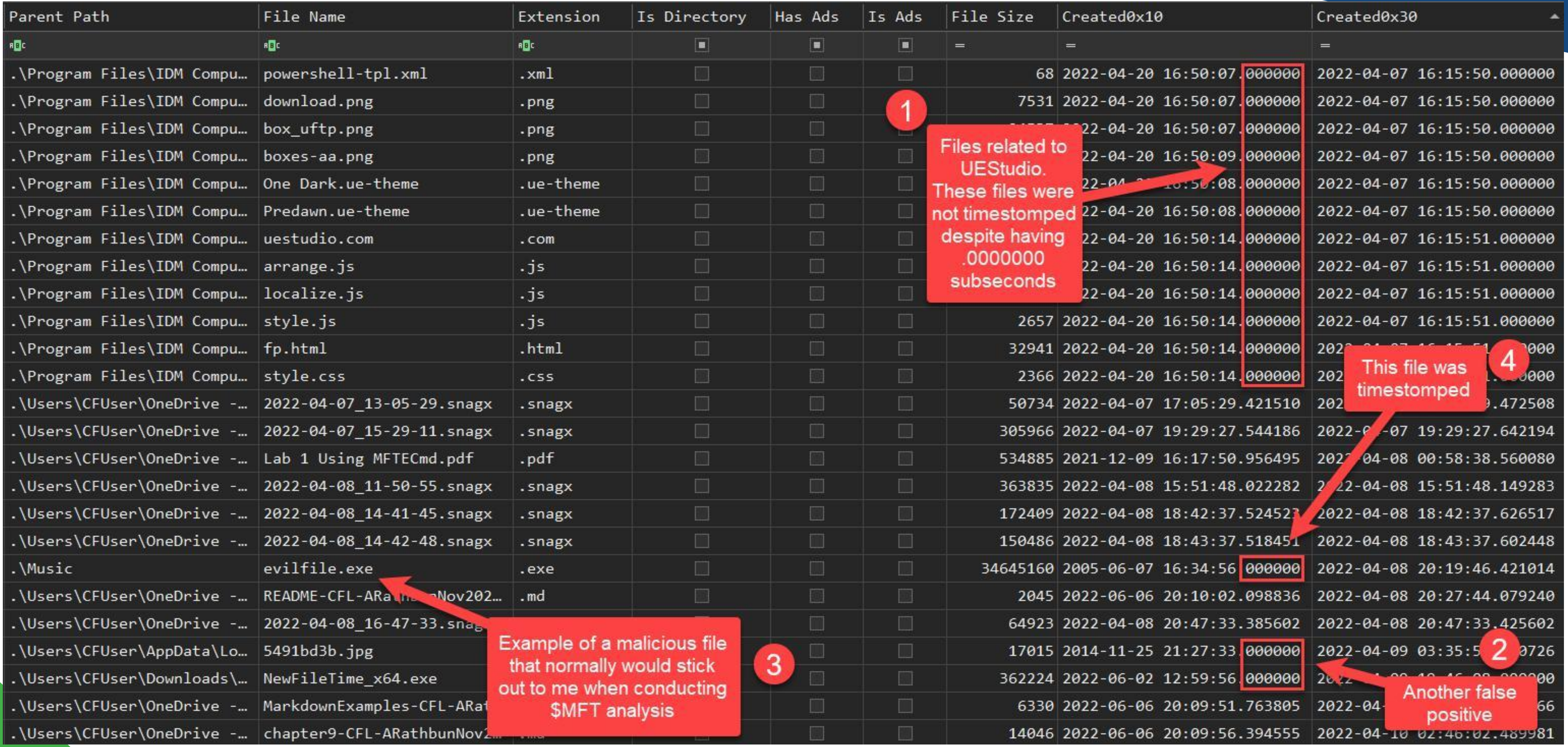
- Toggling the u Sec Zeros column header filter in Timeline Explorer with MFTECmd CSV output ingested will filter on all entries in the $MFT that have any timestamp with zeroed out subseconds (.0000000).
- Obviously, no threat actor is timestomping that many files on a victim's system
- This is exactly why context matters (as seen in the next slide)
  - – Example of an everyday $MFT on a victim system



SI<FN	u Sec Zeros	Copied
▣	▣	▣
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐
☐	☐	☐



Total lines 3,145,486 Visible lines 1,695,890



Parent Path	File Name	Extension	Is Directory	Has Ads	Is Ads	File Size	Created0x10	Created0x30
			■	■	■	=	=	=
.\Program Files\IDM Compu…	powershell-tpl.xml	.xml	☐	☐	☐	68	2022-04-20 16:50:07.000000	2022-04-07 16:15:50.000000
.\Program Files\IDM Compu…	download.png	.png	☐	☐	[illegible]	7531	2022-04-20 16:50:07.000000	2022-04-07 16:15:50.000000
.\Program Files\IDM Compu…	box_uftp.png	.png	☐	☐	[illegible]	[illegible]	22-04-20 16:50:07.000000	2022-04-07 16:15:50.000000
.\Program Files\IDM Compu…	boxes-aa.png	.png	☐	☐	☐	[illegible]	22-04-20 16:50:09.000000	2022-04-07 16:15:50.000000
.\Program Files\IDM Compu…	One Dark.ue-theme	.ue-theme	☐	☐	☐	[illegible]	:08.000000	2022-04-07 16:15:50.000000
.\Program Files\IDM Compu…	Predawn.ue-theme	.ue-theme	☐	☐	☐	[illegible]	22-04-20 16:50:08.000000	2022-04-07 16:15:50.000000
.\Program Files\IDM Compu…	uestudio.com	.com	☐	☐	☐	[illegible]	22-04-20 16:50:14.000000	2022-04-07 16:15:51.000000
.\Program Files\IDM Compu…	arrange.js	.js	☐	☐	☐	[illegible]	22-04-20 16:50:14.000000	2022-04-07 16:15:51.000000
.\Program Files\IDM Compu…	localize.js	.js	☐	☐	☐	[illegible]	22-04-20 16:50:14.000000	2022-04-07 16:15:51.000000
.\Program Files\IDM Compu…	style.js	.js	☐	☐	☐	2657	2022-04-20 16:50:14.000000	2022-04-07 16:15:51.000000
.\Program Files\IDM Compu…	fp.html	.html	☐	☐	☐	32941	2022-04-20 16:50:14.000000	202 000
.\Program Files\IDM Compu…	style.css	.css	☐	☐	☐	2366	2022-04-20 16:50:14.000000	202 000
.\Users\CFUser\OneDrive -…	2022-04-07_13-05-29.snagx	.snagx	☐	☐	☐	50734	2022-04-07 17:05:29.421510	202 .472508
.\Users\CFUser\OneDrive -…	2022-04-07_15-29-11.snagx	.snagx	☐	☐	☐	305966	2022-04-07 19:29:27.544186	2022-04-07 19:29:27.642194
.\Users\CFUser\OneDrive -…	Lab 1 Using MFTECmd.pdf	.pdf	☐	☐	☐	534885	2021-12-09 16:17:50.956495	2022-04-08 00:58:38.560080
.\Users\CFUser\OneDrive -…	2022-04-08_11-50-55.snagx	.snagx	☐	☐	☐	363835	2022-04-08 15:51:48.022282	2022-04-08 15:51:48.149283
.\Users\CFUser\OneDrive -…	2022-04-08_14-41-45.snagx	.snagx	☐	☐	☐	172409	2022-04-08 18:42:37.524523	2022-04-08 18:42:37.626517
.\Users\CFUser\OneDrive -…	2022-04-08_14-42-48.snagx	.snagx	☐	☐	☐	150486	2022-04-08 18:43:37.518451	2022-04-08 18:43:37.602448
.\Music	evilfile.exe	.exe	☐	☐	☐	34645160	2005-06-07 16:34:56.000000	2022-04-08 20:19:46.421014
.\Users\CFUser\OneDrive -…	README-CFL-ARa Nov202…	.md	☐	☐	☐	2045	2022-06-06 20:10:02.098836	2022-04-08 20:27:44.079240
.\Users\CFUser\OneDrive -…	2022-04-08_16-47-33.snag	[illegible]	[illegible]	☐	☐	64923	2022-04-08 20:47:33.385602	2022-04-08 20:47:33.425602
.\Users\CFUser\AppData\Lo…	5491bd3b.jpg	[illegible]	[illegible]	☐	☐	17015	2014-11-25 21:27:33.000000	2022-04-09 03:35:5 0726
.\Users\CFUser\Downloads\…	NewFileTime_x64.exe	[illegible]	[illegible]	☐	☐	362224	2022-06-02 12:59:56.000000	20 00
.\Users\CFUser\OneDrive -…	MarkdownExamples-CFL-ARat	[illegible]	[illegible]	☐	☐	6330	2022-06-06 20:09:51.763805	2022-04- 66
.\Users\CFUser\OneDrive -…	chapter9-CFL-ARathbunNov2…	.md	[illegible]	☐	☐	14046	2022-06-06 20:09:56.394555	2022-04-10 02:46:02.489981

1 Files related to UEStudio. These files were not timestomped despite having .0000000 subseconds

2 Another false positive

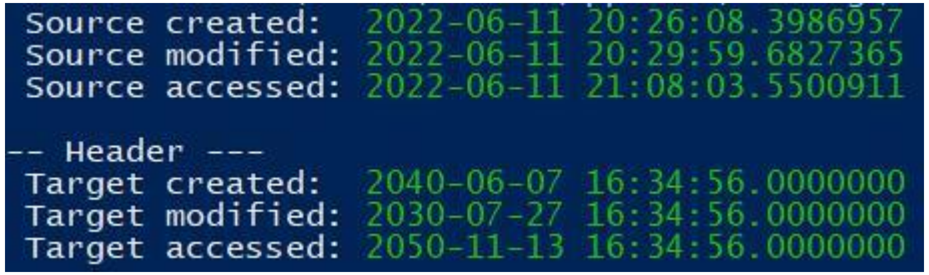
3 Example of a malicious file that normally would stick out to me when conducting $MFT analysis

4 This file was timestomped

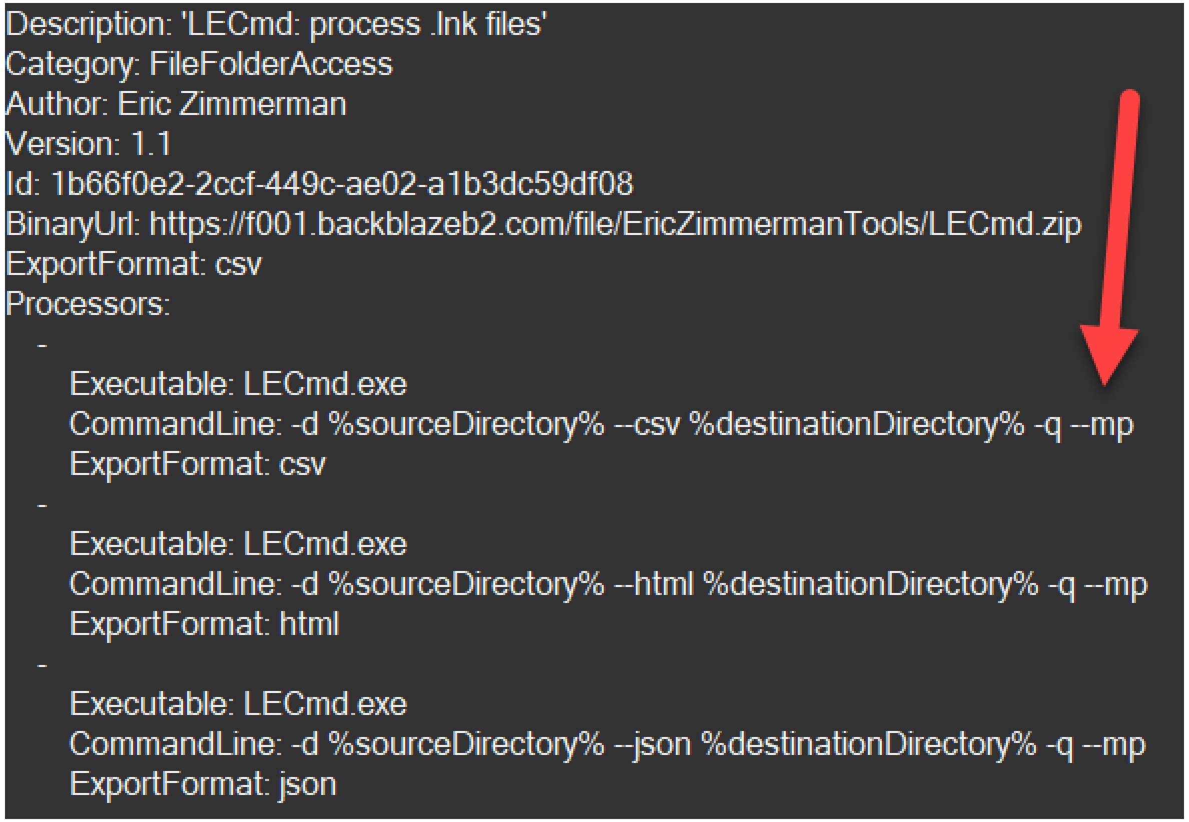
Was a File Opened After Timestomping?

Another way to enumerate threat actor activity

- If a LNK file reflects timestomped values for the target file, the file was opened AFTER timestomping
- If the LNK file doesn't reflect timestomped values for file(s) that you know were timestomped, they've not been opened AFTER timestomping
- LECmd
  - --mp switch for LECmd provides more precise timestamps (subseconds!)



```
  Source created:   2022-06-11 20:26:08.3986957
  Source modified:  2022-06-11 20:29:59.6827365
  Source accessed:  2022-06-11 21:08:03.5500911

-- Header ---
  Target created:   2040-06-07 16:34:56.0000000
  Target modified:  2030-07-27 16:34:56.0000000
  Target accessed:  2050-11-13 16:34:56.0000000
```

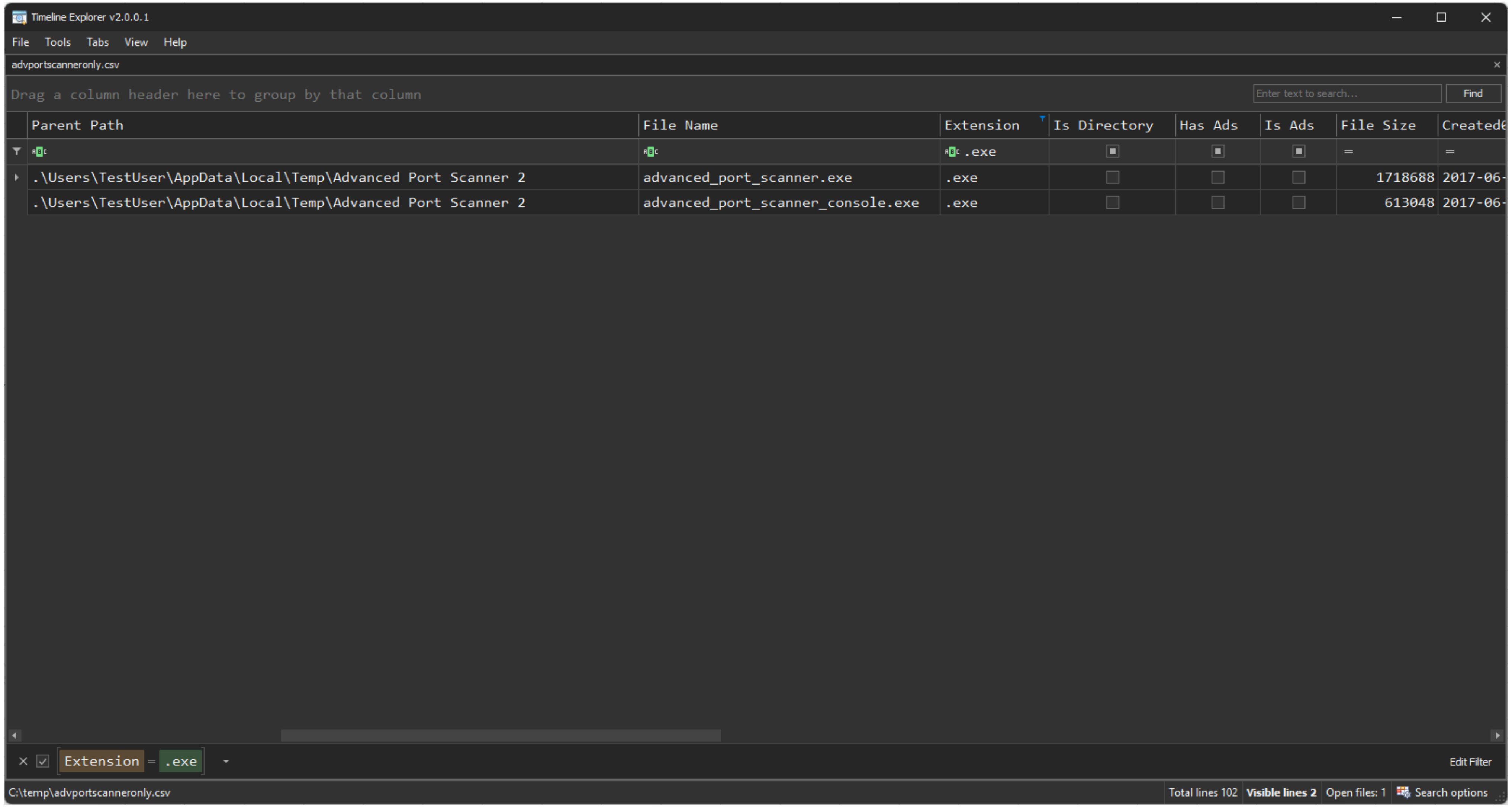


```
Description: 'LECmd: process .lnk files'
Category: FileFolderAccess
Author: Eric Zimmerman
Version: 1.1
Id: 1b66f0e2-2ccf-449c-ae02-a1b3dc59df08
BinaryUrl: https://f001.backblazeb2.com/file/EricZimmermanTools/LECmd.zip
ExportFormat: csv
Processors:
    -
        Executable: LECmd.exe
        CommandLine: -d %sourceDirectory% --csv %destinationDirectory% -q --mp
        ExportFormat: csv
    -
        Executable: LECmd.exe
        CommandLine: -d %sourceDirectory% --html %destinationDirectory% -q --mp
        ExportFormat: html
    -
        Executable: LECmd.exe
        CommandLine: -d %sourceDirectory% --json %destinationDirectory% -q --mp
        ExportFormat: json
```

Real Life Examples of Timestomping

Real-Life Examples of Timestomping

Advanced Port Scanner timestomped by threat actors



Parent Path	File Name	Extension	Is Directory	Has Ads	Is Ads	File Size	Created0
		.exe	■	■	■	=	=
.\Users\TestUser\AppData\Local\Temp\Advanced Port Scanner 2	advanced_port_scanner.exe	.exe	□	□	□	1718688	2017-06-
.\Users\TestUser\AppData\Local\Temp\Advanced Port Scanner 2	advanced_port_scanner_console.exe	.exe	□	□	□	613048	2017-06-

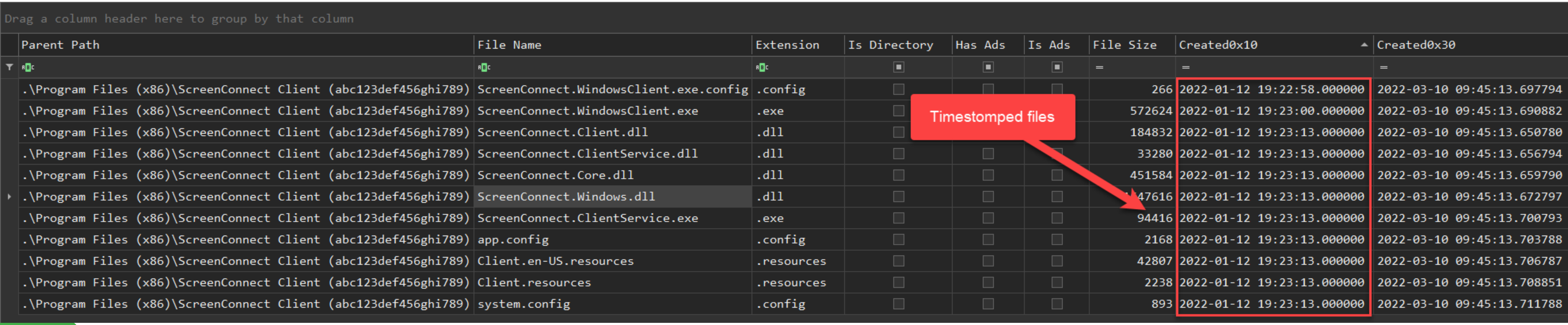
☑ Extension = .exe

C:\temp\advportscanneronly.csv

Real-Life Examples of Timestomping

ScreenConnect timestomped by threat actors

- Again, context matters! ScreenConnect isn’t a malicious tool, but it can be used by a malicious actor
- Client stated they never have used ScreenConnect, yet these files appear during the timeframe of interest
- Analysis will direct us to look around 2022-03-10 at 0945 hours to see what else what going on at the time these unauthorized files appeared on disk….likely more malicious activity!



Drag a column header here to group by that column

Parent Path	File Name	Extension	Is Directory	Has Ads	Is Ads	File Size	Created0x10	Created0x30
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.WindowsClient.exe.config	.config	☐	☐	☐	266	2022-01-12 19:22:58.000000	2022-03-10 09:45:13.697794
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.WindowsClient.exe	.exe	☐	☐	☐	572624	2022-01-12 19:23:00.000000	2022-03-10 09:45:13.690882
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.Client.dll	.dll	☐	☐	☐	184832	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.650780
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.ClientService.dll	.dll	☐	☐	☐	33280	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.656794
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.Core.dll	.dll	☐	☐	☐	451584	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.659790
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.Windows.dll	.dll	☐	☐	☐	[illegible]47616	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.672797
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	ScreenConnect.ClientService.exe	.exe	☐	☐	☐	94416	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.700793
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	app.config	.config	☐	☐	☐	2168	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.703788
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	Client.en-US.resources	.resources	☐	☐	☐	42807	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.706787
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	Client.resources	.resources	☐	☐	☐	2238	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.708851
.\Program Files (x86)\ScreenConnect Client (abc123def456ghi789)	system.config	.config	☐	☐	☐	893	2022-01-12 19:23:13.000000	2022-03-10 09:45:13.711788

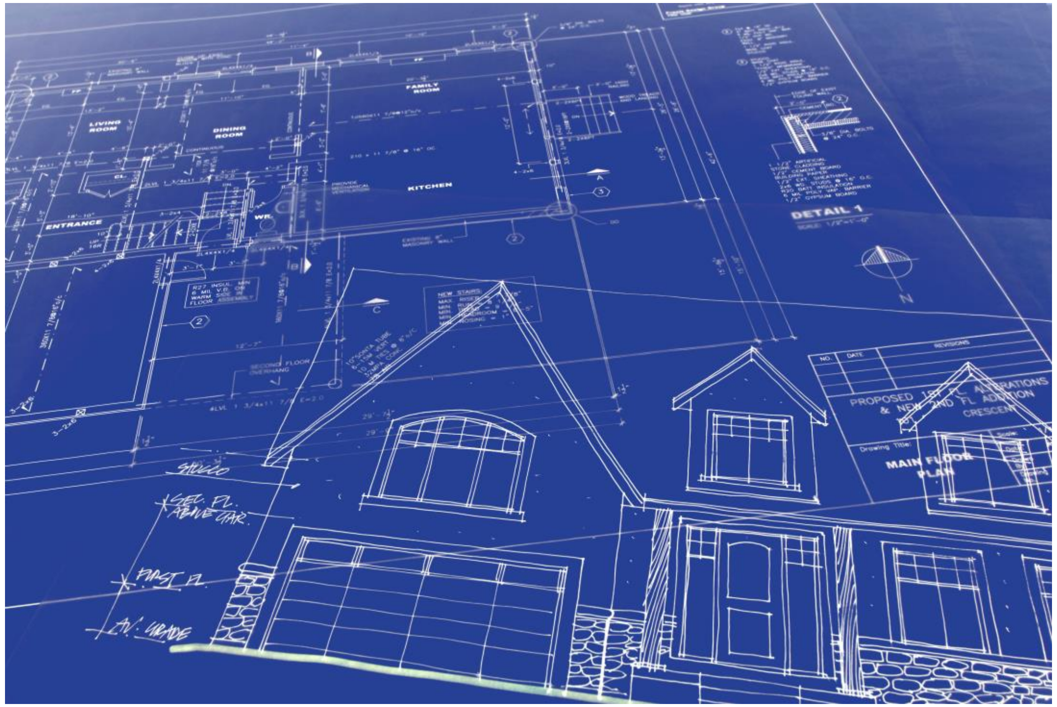
Timestomped files

Advanced Timestomping Methods/Tools

Advanced Timestomping Methods

Special thanks to Mark Spencer of Arsenal Recon for these ideas

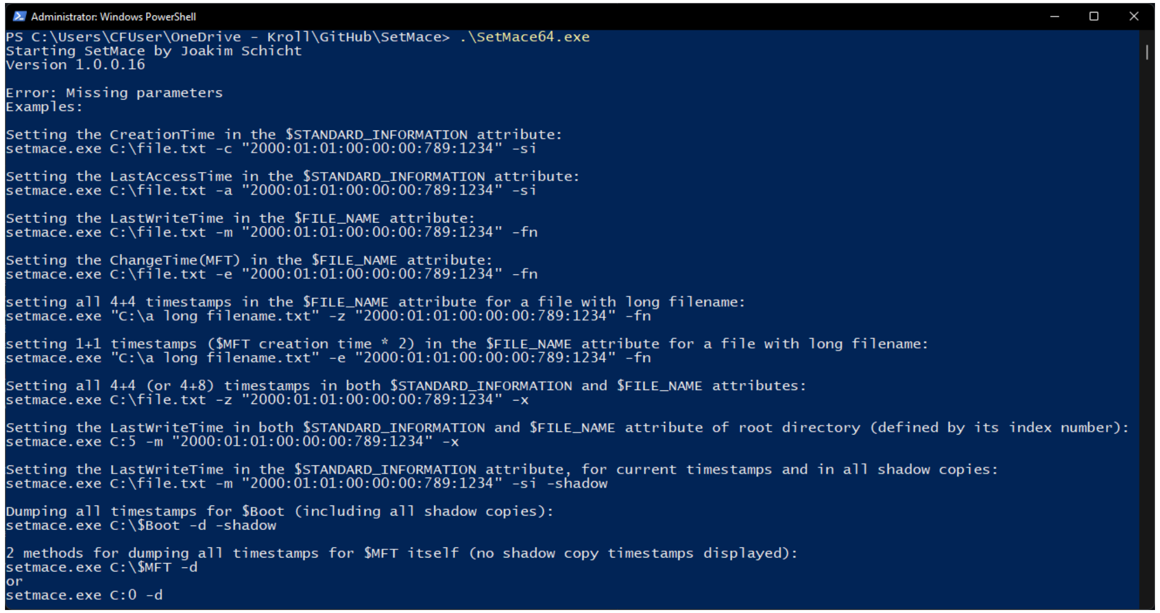
- Copying files to a drive containing one or more NTFS volumes by physically attaching it to an already backdated workstation and extracting ZIP/RAR contents whose timestamps have been previously forged
- Delivering files using malware that contains an embedded NTFS driver
- Using SetMace to alter the **0x30** timestamps using a kernel mode driver (next slide)



SetMace

Created by Joachim Schicht of Arsenal Recon

- SetMace is an advanced timestamp manipulation tool
- SetMace rebuilds the filesystem internally and writes new timestamps directly to the physical disk
- Uses a kernel driver and completely bypasses the filesystem/OS (Windows API)
- Can change the 0x10 **AND** 0x30 timestamps
- More advanced than something like NewFileTime, Total Commander, etc.
- Threat actor(s) can get away with simpler ways of timestomping. Likely only to be used by highly motivated actors in highly sensitive environments
- [jschicht/SetMace: Manipulate timestamps on NTFS (github.com)]

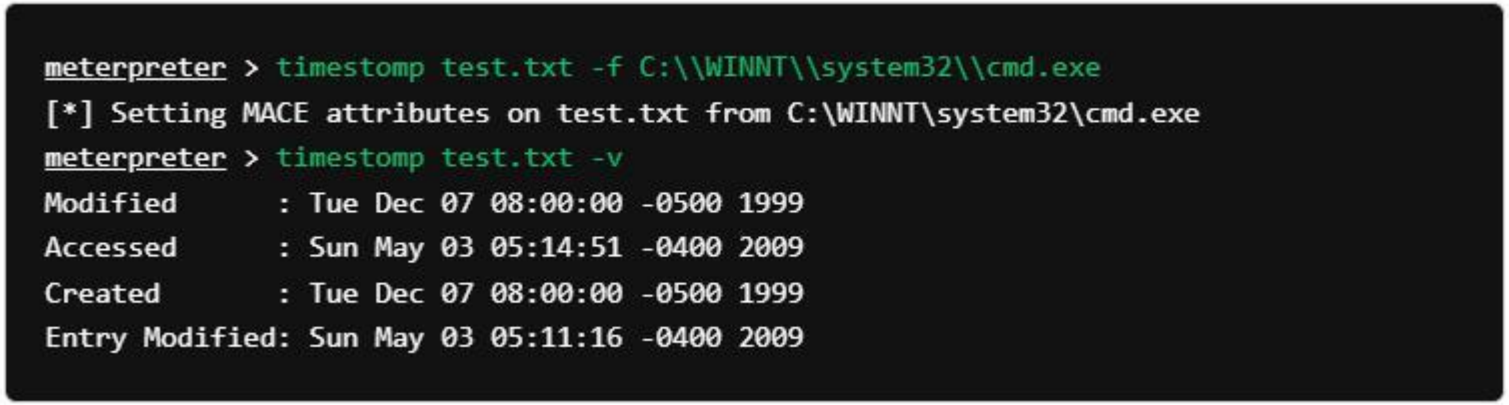


```
PS C:\Users\CFUser\OneDrive - Kroll\GitHub\SetMace> .\SetMace64.exe
Starting SetMace by Joakim Schicht
Version 1.0.0.16

Error: Missing parameters
Examples:

Setting the CreationTime in the $STANDARD_INFORMATION attribute:
setmace.exe C:\file.txt -c "2000:01:01:00:00:00:789:1234" -si

Setting the LastAccessTime in the $STANDARD_INFORMATION attribute:
setmace.exe C:\file.txt -a "2000:01:01:00:00:00:789:1234" -si

Setting the LastWriteTime in the $FILE_NAME attribute:
setmace.exe C:\file.txt -m "2000:01:01:00:00:00:789:1234" -fn

Setting the ChangeTime(MFT) in the $FILE_NAME attribute:
setmace.exe C:\file.txt -e "2000:01:01:00:00:00:789:1234" -fn

setting all 4+4 timestamps in the $FILE_NAME attribute for a file with long filename:
setmace.exe "C:\a long filename.txt" -z "2000:01:01:00:00:00:789:1234" -fn

setting 1+1 timestamps ($MFT creation time * 2) in the $FILE_NAME attribute for a file with long filename:
setmace.exe "C:\a long filename.txt" -e "2000:01:01:00:00:00:789:1234" -fn

Setting all 4+4 (or 4+8) timestamps in both $STANDARD_INFORMATION and $FILE_NAME attributes:
setmace.exe C:\file.txt -z "2000:01:01:00:00:00:789:1234" -x

Setting the LastWriteTime in both $STANDARD_INFORMATION and $FILE_NAME attribute of root directory (defined by its index number):
setmace.exe C:5 -m "2000:01:01:00:00:00:789:1234" -x

Setting the LastWriteTime in the $STANDARD_INFORMATION attribute, for current timestamps and in all shadow copies:
setmace.exe C:\file.txt -m "2000:01:01:00:00:00:789:1234" -si -shadow

Dumping all timestamps for $Boot (including all shadow copies):
setmace.exe C:\$Boot -d -shadow

2 methods for dumping all timestamps for $MFT itself (no shadow copy timestamps displayed):
setmace.exe C:\$MFT -d
or
setmace.exe C:0 -d
```

Other Tools with Timestomping Capabilities

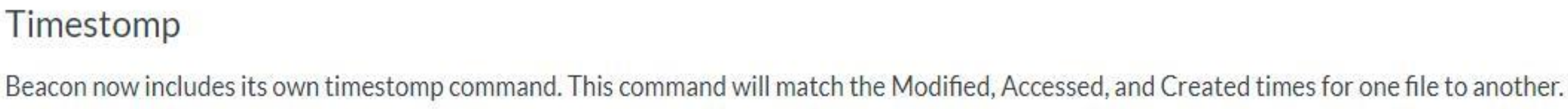
Commonly seen in IR engagements

- Metasploit
  - Timestomp module



```
meterpreter > timestomp test.txt -f C:\\WINNT\\system32\\cmd.exe
[*] Setting MACE attributes on test.txt from C:\WINNT\system32\cmd.exe
meterpreter > timestomp test.txt -v
Modified      : Tue Dec 07 08:00:00 -0500 1999
Accessed      : Sun May 03 05:14:51 -0400 2009
Created       : Tue Dec 07 08:00:00 -0500 1999
Entry Modified: Sun May 03 05:11:16 -0400 2009
```

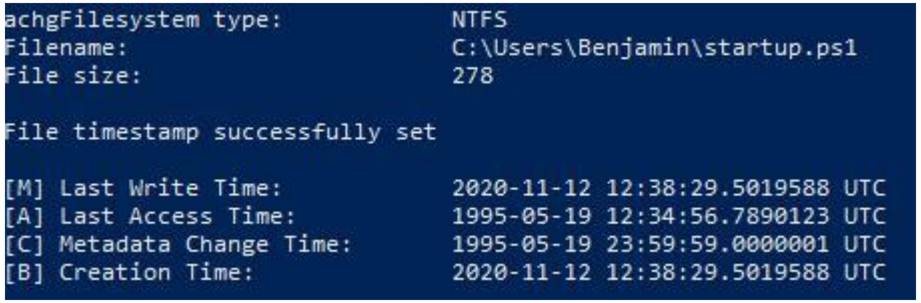
- Cobalt Strike
  - Timestomp beacon added in 2014



Timestomp

Beacon now includes its own timestomp command. This command will match the Modified, Accessed, and Created times for one file to another.

- nTimetools
  - [limbenjamin/nTimetools: Timestomper and Timestamp checker with nanosecond accuracy for NTFS volumes (github.com)]



```
achgFilesystem type:            NTFS
Filename:                       C:\Users\Benjamin\startup.ps1
File size:                      278

File timestamp successfully set

[M] Last Write Time:            2020-11-12 12:38:29.5019588 UTC
[A] Last Access Time:           1995-05-19 12:34:56.7890123 UTC
[C] Metadata Change Time:       1995-05-19 23:59:59.0000001 UTC
[B] Creation Time:              2020-11-12 12:38:29.5019588 UTC
```

Thank You

Upcoming KAPE Intensive Training and Certification Sessions

- Virtual live sessions
- Max 25 students

Full Calendar Available here:
https://bit.ly/KAPE2022

SCHEDULE	INSTRUCTORS
June 23, 2022 10:00 a.m. - 7:00 p.m. ET	Eric Zimmerman Sean Straw Scott Zuberbuehler Andrew Rathbun
September 27, 2022 10:00 a.m. - 7:00 p.m. ET	Eric Zimmerman Sean Straw Scott Zuberbuehler Andrew Rathbun
October 4, 2022 8:00 a.m. - 5:00 p.m. BST	James Thoburn Paul Wells Guillermo Roman