(WIP) Using Zimmerman Tools for Windows Forensics (WIP)

Logo

Description automatically generated

What are Zimmerman Tools? A suite of digital forensics tools developed by Eric Zimmerman. They are used by forensic examiners and investigators to analyze digital evidence. These tools are used for extracting and analyzing various types of data from Windows systems.

Download tools here:

<https://ericzimmerman.github.io/#!index.md>

To download all of his most recent tools, go to the above website and download his powershell script, Get-ZimmermanTools. Run this on the host you want to download the toolset to (you will need internet, also make sure your powershell is up to date and run as admin)



Once the script is run a folder will populate with the toolset (where the powershell script is ran, unless specified). Each tool has its own purpose and is used to examine certain windows artifacts.

Graphical user interface, text

Description automatically generated

In the following pages we will go over commonly used Artifacts on a Windows host and how to use Zimmerman tools to analyze them.

You will need .net 6 to be installed on the host to run any GUI zimmerman tool! If not, you will need to go the the below link to download it.

<https://dotnet.microsoft.com/en-us/download/dotnet/6.0>

A person with a beard

Description automatically generated with medium confidence

ARTIFACTS

Artifact: $MFT

What is it: The $MFT (Master File Table) is a special file used by the NTFS (New Technology File System) in Windows operating systems. It is a crucial component of the NTFS file system and serves as a central database that contains information about every file and directory on an NTFS volume.

For every file on the NTFS volume there are these dates

* (M) Data Content Change Time- his timestamp shows when the content of the file was last modified. It changes every time the file's data is altered.
* (A) Data Last Accessed Time- This timestamp indicates the last time the file was read or accessed. Simply viewing a file's contents will update this timestamp.
* (C) Metadata Change Time- In the NTFS file system, this is referred to as the MFT record modification timestamp. It indicates when the file's metadata or properties were last changed. Changes could include modifications to the file's name, attributes, or location (not the content itself).
* (B) Birth - Often also referred to as the creation timestamp, this records when the file was created on the file system.

Every time one of these events takes place an entry and timestamp will be placed in the $MFT

Parsing the $MFT is tremendously effective at creating a timeline of events that took place on a host.

**Windows 10 location**: C:\$MFT (you will not be able to view it using file explore, but it’s there)

**Tool:** MFTECmd.exe

**How to:** We must run MFTECmd.exe to parse the $MFT into a csv file

Text

Description automatically generated

Now we can view the MFT using TimelineExplorer, lets try to search for a file (potential IOC):

Open Timeline Explorer à File à Open à navigate to the parsed MFT is CSV format

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

You can search for a file name

Graphical user interface, text, application, email

Description automatically generated

View NTFS time stamps, can pivot off the timestamps to see surrounding activity

Graphical user interface, text, application, email

Description automatically generated

Artifact: Windows Eventlogs

What is it: used to record detailed information about significant hardware, software, and system events on a computer. These logs are essential for system diagnostics, troubleshooting, security monitoring, and forensic investigations.

Types of logs: Application, System, Security, PowerShell, Sysmon (if installed)

**Windows 10 location:** C:\Windows\System32\winevt\Logs\

**Tool:** Windows Event Viewer

**How to:** take .evtx file and open with Windows event viewer

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

You will need to understand the type of logs you are reviewing and their corresponding Event IDs. I usually find myself in the Security Logs as a great place to start (Sysmon even better if that is installed).

Below is a link that lists Windows Security Log event IDs and what they mean (not official Microsoft documentation, but useful)

<https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/>

Also, note that in the directory C:\Windows\System32\winevt\Logs\ contains Windows Defender logs. This could provide valuable information to see if HBSS successfully defeated a malicious program. If 3rd party antivirus is used you’ll have to do research to see where the location for those logs are.



Artifact: Prefetch

What is it: Prefetch files are files created by the Windows operating system to speed up the loading time of applications. They are part of the Windows Prefetcher, a component of the memory management system introduced in Windows XP and present in later versions. They are a key artifact to show potential execution of a program.

**Windows 10 location:** C:\Windows\Prefetch

**Tool:** PECmd

**How to:** You can view the directory to see all the prefetch files created, these should line up with a corresponding executable. Even better you can note a specific prefetch file and use the Zimmerman tool PECmd to provide detailed information about the execution of the application, including the files and directories it accessed, execution timestamps, and usage statistics

Graphical user interface, text

Description automatically generated



Text

Description automatically generated

Files referenced (could be used to see an injected process, is it accessing any strange dlls?)

Text

Description automatically generated

You can also run <PECmd.exe -d “C:\Windows\Prefetch”> To output the entire directory

Artifact: Internet history files

What is it: Each browser will store a history file in it’s default location that contains a user’s browsing history. These are usually stored in an SQLite database.

**Windows 10 location**:

Chrome: %USERPROFILE%\AppData\Local\Google\Chrome\User Data\Default\History

Edge: %USERPROFILE%\AppData\Local\Microsoft\Edge\User Data\Default\History

Firefox: %USERPROFILE%\AppData\Roaming\Mozilla\Firefox\Profiles\places.sqlite

**Tool:** SQLite viewer. As of my knowledge Zimmerman doesn’t have a tool to view brower history. An easy solution is to use SQLite viewer, just drag and drop the history file and export it to a csv (note, be cautious on what you upload. The github claims no file will be uploaded as it uses JavaScript HTML5 File Reader ).

How to: upload one of the SQL lite databases to [https://inloop.github.io/sqlite-viewer/#](https://inloop.github.io/sqlite-viewer/)

Graphical user interface, text, application, email

Description automatically generated

Export to csv

Graphical user interface, text, application, chat or text message

Description automatically generated

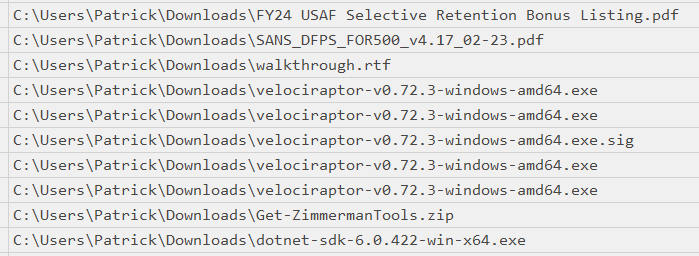
View in the tables you have downloaded, this is the parsed out history file

Graphical user interface, text

Description automatically generated

I think the most useful tables are downloads and urls

Lets open the downloads up (of course we’ll use Timeline explorer)



As you can see it stores what was downloaded from chrome. I was surprised how far the database goes back.

Lets take a look at the urls table. A column of urls visited will appear. Using timeline explorer we can filter for whatever we are searching for.

Text, application

Description automatically generated

Artifact: Amcache.hve

What is it: The Amcache.hve file in Windows is a part of the Windows Registry that stores important information related to the execution of applications.

**Windows 10 location**: C:\Windows\AppCompat\Programs\Amcache.hve

**Tool:** AmcacheParser

**How to:** parse hive with the amcacheparser



Text

Description automatically generated

Use timeline explorer to view the csv. We’ll check out the unassociatedFileEntries csv output, most closely associated with program execution.

Graphical user interface, text, application, email

Description automatically generated

Artifact: NTUSER.dat

What is it: is a registry file associated with a user profile on Windows. It contains user-specific registry settings that are loaded when a user logs into their Windows account. These settings help configure the user's environment according to their preferences and previous configurations.

**Windows 10 location:** %USERPROFILE%\NTUSER.dat

**Tool:** Registry Explorer

**How to:** open registry Explorer à load hive and select the artifact NTUSER.dat

Graphical user interface, table

Description automatically generated

There is a plethora of information you can get about a user from this file

Example:

UserAssist – records GUI-based program execution

NTUSER.dat location –

NTUSER.DAT\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist\{GUID}\Count

Graphical user interface, table

Description automatically generated

Also gives last execution time

Graphical user interface, table

Description automatically generated

Artifact: Registry Hive

What is it: A registry hive is a group of keys, subkeys, and values in the Windows Registry that is stored in a respective file. Each hive contains a specific portion of the registry data and is loaded during system startup or user login. The hives represent different aspects of the configuration and are vital for the operation of the operating system and applications.

**Windows 10 location:** C:\Windows\System32\Config\

Text

Description automatically generated

For forensics I generally will try to pull the Software hive, but the others may be useful as well.

**Tool:** registry explorer or reg editor

**How to:** pull one of the hives and open in either reg explorer (Zimmerman tool) or windows reg editor.

Graphical user interface, application

Description automatically generated with medium confidence Graphical user interface, text, application, email

Description automatically generated

I opened the software hive and brought up the Run key. There are numerous artifacts in the registry that are useful. I’ll list a few

===========================================================================

**HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\UserAssist**

We’ve talked about this before, it shows applications executed by gui

**HKEY\_CURRENT\_USER\Software\Microsoft\Windows\Shell\BagMRU**

Stores information about folder settings, such as view preferences and size, which can reveal whether a user accessed a specific folder, even if the folder has been deleted.

**HKEY\_LOCAL\_MACHINE\SYSTEM\MountedDevices**

Contains information on all devices that have ever been mounted, providing insights into external drives, USB sticks, and other storage devices that were connected to the system.

**HKEY\_CURRENT\_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs**

Lists recently opened documents, which can be used to determine recent user activities.

**HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Uninstall**

Contains details on installed software and any software that has been uninstalled, useful for identifying potentially malicious installations.

Artifact: System Resource Usage Monitor (SRUM)

What is it: The System Resource Usage Monitor (SRUM) is a feature in Windows that collects system data related to resource usage, which can be useful in forensic analysis. SRUM tracks detailed information about resource usage by applications and users on a Windows machine, such as CPU usage, network usage, disk activity, and more, over time

**Windows 10 location:** C:\Windows\System32\sru\SRUDB.dat

**Tool:** ScumECmd

**How to:** first parse the data into a CSV



Text

Description automatically generated

7 csv’s are created

Text

Description automatically generated

Now we can view the csv with Timeline explorer. Lets look at the AppResourceUseInfo, this potentially can be used as evidence of file execution.

A picture containing calendar

Description automatically generated

Artifact: Windows 10 Timeline

What is it: This is a newer artifact in windows 10. Windows 10 records recently used applications and files in a “timeline” database in SQLite format.

**Location:** %USERPROFILE%\AppData\Local\ConnectedDevicesPlatform\<account-ID>\ActivitiesCache.db

Tool: SQLECmd.exe

How to: parse the db file with SQLECmd.exe.



3 CSVs should be produced. I found the most useful to be the Windows\_ActivityOperation

Graphical user interface, text, website

Description automatically generated

Lets use timeline explorer to open the CSV and see the timeline windows 10 has in the database. Looking through the results I see a familiar list of recent applications I have used. The database went back about two months.

Calendar

Description automatically generated with medium confidence