The Windows Forensics Journey

Version 1.0
April-2024

By Dr. Shlomi Boutnaru

Created using Craiyon, AI Image Generator

Table of Contents

Introduction

When using a workstation/server running a Microsoft Windows based operating system there are different forensics artifacts which are created. I have decided to write a series of short writeups aimed at providing the basic understanding on the different forensics artifacts created by Windows.

Overall, I wanted to create something that will improve the overall knowledge of digital forensics in Windows with writeups that can be read in 1-3 mins. I hope you are going to enjoy the ride.

Lastly, you can follow me on twitter - @boutnaru (https://twitter.com/boutnaru). Also, you can read my other writeups on medium - https://medium.com/@boutnaru. Lastly, You can find my free eBooks at https://TheLearningJourneyEbooks.com.

Lets GO!!!!!!

LNK Files (Shortcut Files)

Overall, users/the OS can create shortcuts to files/directories. We can think of a shortcut as a file which contains information used for accessing another file/folder. By default, Windows' shortcut files have a "*.lnk" extension (cause they are link files). Windows creates LNK files automatically when users open non-executable files, we can think about documents and images for example[1].

Moreover, LNK files contain different types of attributes (not all of that is displayed in the GUI of Windows) - as shown in the screenshot below. Among the information we can find: the size of the target file, timestamps (both for the LNK file and the target file), the system name, volume serial number, MAC address, indication if the target file is stored local/remote and attributes of the target file (readonly/hidden/etc). There is a great tool by Eric Zimmerman called LECmd[2] which parses LNK files - as shown in the screenshot below in an XML output (it shows more information than the GUI). Lastly, LNK files is based on the "Shell Link Binary File Format"[3].

[1] https://dfir.pubpub.org/pub/wfuxlu9v/release/1
[2] https://ericzimmerman.github.io/#!index.md
[3] https://learn.microsoft.com/en-us/openspecs/windows_protocols/ms-shllink/16cb4ca1-9339-4d0c-a68d-bf1d6cc0f943

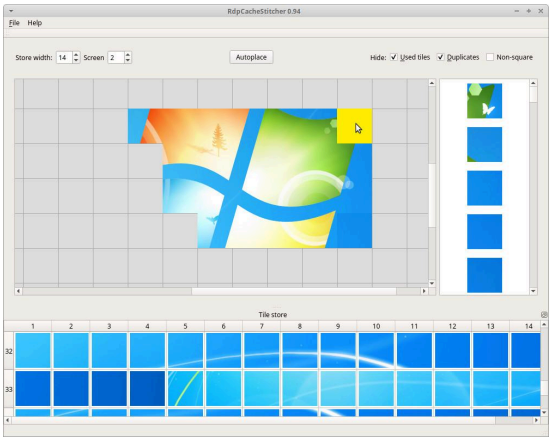
RDP Bitmap Cache (Remote Desktop Protocol Bitmap Cache)

When using "mstsc.exe"[4] for connecting remotely to Windows systems (workstation/server) the client leverages an RDP caching mechanism. It is used to reduce the amount of data to be sent by the server. The caching is done by caching those parts of the screen that have not changed since the display was last refreshed[5].

Thus, when enabled the RDP bitmap caching allows the session to use data already in the local cache files to provide better experience and reduce network bandwidth. Each bitmap cache entry stores bitmap data and metadata (color depth, key and dimensions). It is important to understand that this cache is persistent even after the RDP session has been closed[6].

Moreover, the cache files store raw bitmaps in the forms of tiles. Although the tile size can vary, the most common size is 64x64 pixels. The location of the RDP bitmap cache is "%localappdata%\Microsoft\Terminal Server Client\Cache" (as a reminder "Terminal Server" is the old RDP name). There we can have two type of files "bcacheX.bmc" (where X is 2/22/24 which represent the quality) and "CacheXYZW.bin" (where XYZW are numbers that are generated on each session), we can use their timestamp to correlate with other log files[7].

Lastly, we can use the open source "BMC-Tools" (which is written in Python) in order to parse the RDP bitmap cache[8]. Also, we can use the perl script in order to try and rebuild some of the screenshots automatically[9] after they are extracted by using "BMC-Tools". There is also an option of trying to stitch the bitmaps using a UI tool called "RdpCacheStitcher"[10] - shown below.



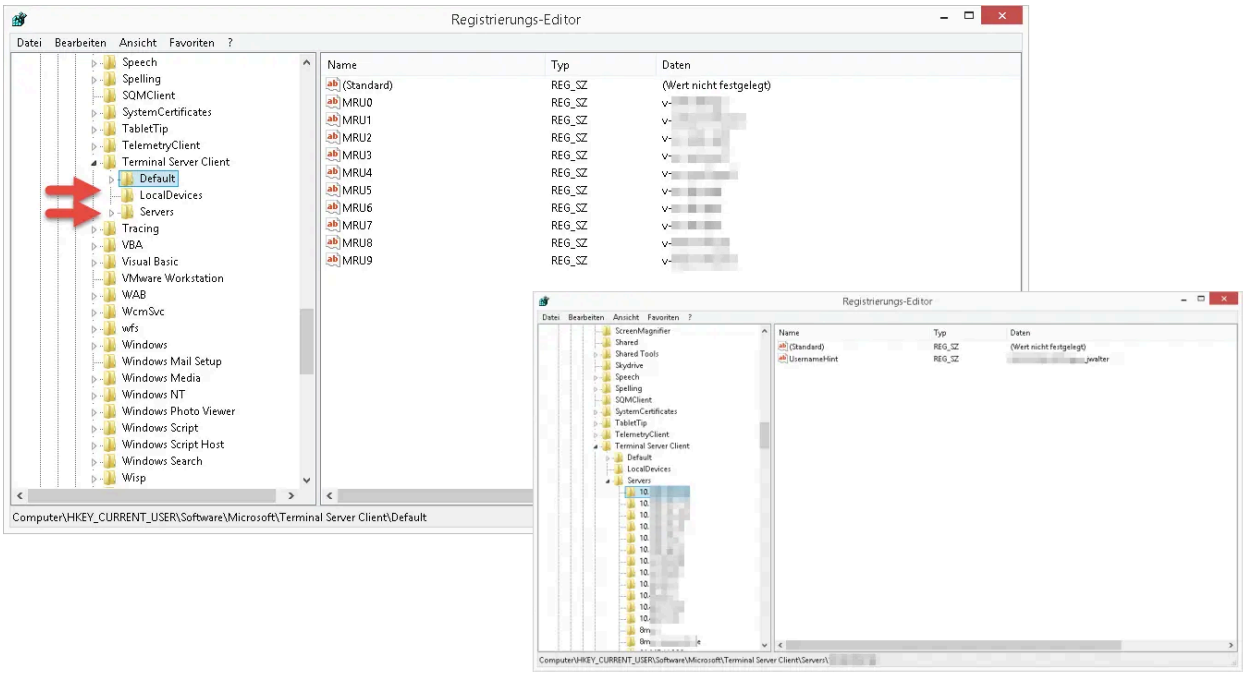
[4] https://medium.com/@boutnaru/the-windows-process-journey-mstsc-exe-remote-desktop-connection-981bae774bac
[5] https://security.opentext.com/appDetails/RDP-Cached-Bitmap-Extractor
[6] https://www.paloaltonetworks.com/blog/security-operations/playbook-of-the-week-uncover-your-rdp-secrets/
[7] https://www.linkedin.com/pulse/blind-forensics-rdp-bitmap-cache-ronald-craft
[8] https://github.com/ANSSI-FR/bmc-tools
[9] https://github.com/brimorlabs/rdpieces
[10] https://github.com/BSI-Bund/RdpCacheStitcher

RDP Connection History (Remote Desktop Protocol Connection History)

When using “mstsc.exe”[11] for initiating an RDP connection, every successful connection causes the connection details to be logged (IP/hostname information). This information is saved for each user in the following registry branch: “HKCU\SOFTWARE\Microsoft\Terminal Server Client”. There are two relevant registry keys: “Default” and “Servers”[12].

Moreover, “Default” holds the history of the last 10 RDP connections. While “Servers” contains a list of all RDP connections that have ever been created from the local machine by the user. An example of both is shown in the screenshots below. By the way, MRU shown in the screenshots stands for “Most Recently Used”[13].

Lastly, when using “mstsc.exe” a hidden file named “Default.rdp” is created in the home directory of the user, the full path is “%homepath%\Documents\Default.rdp”[14].



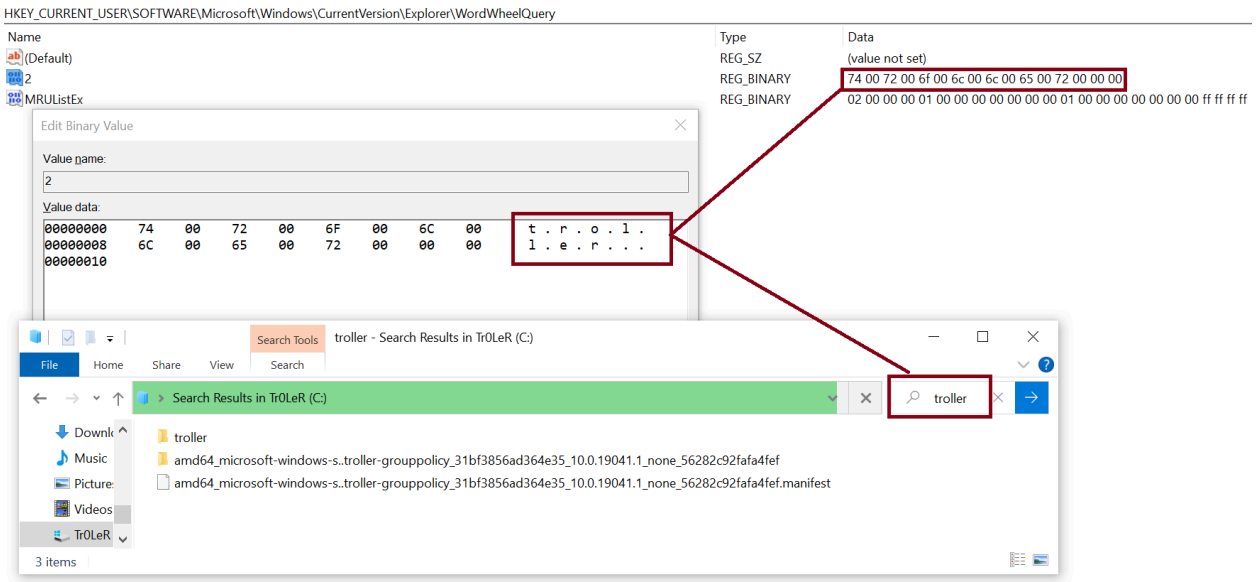
[11] https://medium.com/@boutnaru/the-windows-process-journey-mstsc-exe-remote-desktop-connection-981bae774bac
[12] https://www.tachytelic.net/2019/01/clear-rdp-cache/
[13] https://www.fity.club/lists/suggestions/hkey-current-user-software-microsoft-windows/
[14] https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/mstsc

Word Wheel Query (File Explorer Searches)

In case users are using the builtin search feature in "File Explorer"[15] we can extract the searched items for the "WordWheelQuery" registry key - as shown in the screenshot below. This is relevant for different versions of Windows such as 7/8/10/11[16].

Overall, we can read the "WordWheelQuery" registry key from the following location: "HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\WordWheelQuery" in case of an online analysis (when the operating system is running). In case of an offline analysis we can extract the information from the NTUSER.DAT file (which holds the information/configurations of a specific local/domain Windows user).

Lastly, by removing the data values in the location mentioned above we basically erase/clear the search history in "File Explorer"[17].



[15] https://medium.com/@boutnaru/the-windows-concept-journey-file-explorer-previously-windows-explorer-e48077b135a0
[16] https://forensafe.com/blogs/searchedstrings.html
[17] https://www.windowscentral.com/how-clear-search-history-file-explorer-windows-10

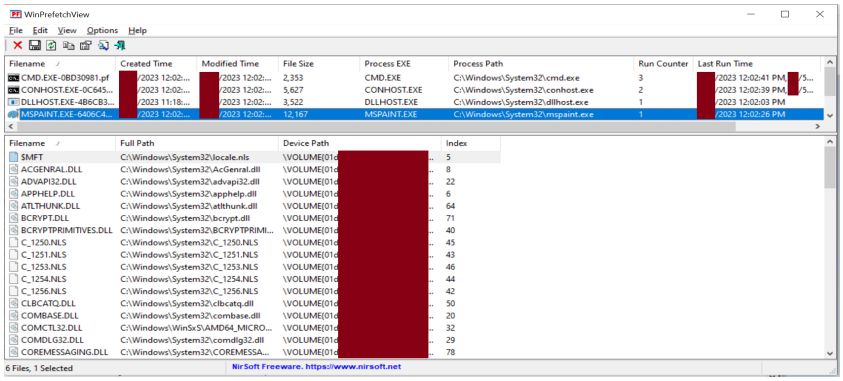
Prefetch

Since Windows XP there is a component called “Prefetcher” which is part of the Memory Manager. Its goal is to speed up the Windows boot process and reduce the time it takes to start programs. This is done by caching to RAM files that are needed while the program is launched (based on information collected from previous executions). Since Windows Vista this mechanism was extended by “SuperPrefetch” and “ReadyBoost”[18].

Overall, for every process execution there is a creation/modification of a “*.pf” file in the “%systemroot%\Prefetch” directory. It is important to know that those files are not user-specific and have a global scope. Due to that, there is no user information as part of the artifact. The existence of a “*.pf” file states that a certain executable was launched on the system[19].

Moreover, from prefetch files we can extract the following information: file size, the binary name, the number of times the binary was executed, the path to the binary, first execution time, last execution time (up to the last 8) and a list of referenced files (like “*.dll” files that have been loaded by the process). We can use “WinPrefetchView” by Nirsoft[20] for parsing the information of “*.pf” file - as shown in the screenshot below.

Lastly, the patten of the “*.pf” files’ names is “[ORIGINAL_BINARY_NAME]-[HASH_OF_APP_PATH].pf”, an example of that is “MSPAINT.EXE-6406C4A1.pf”[21]. For disabling prefetch we need to set the value name “EnablePrefetcher” to the value “0” in the following registry key “HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager\Memory Management\PrefetchParameters”[22] By the way, the prefetch technology is based on a patent from Microsoft[23].



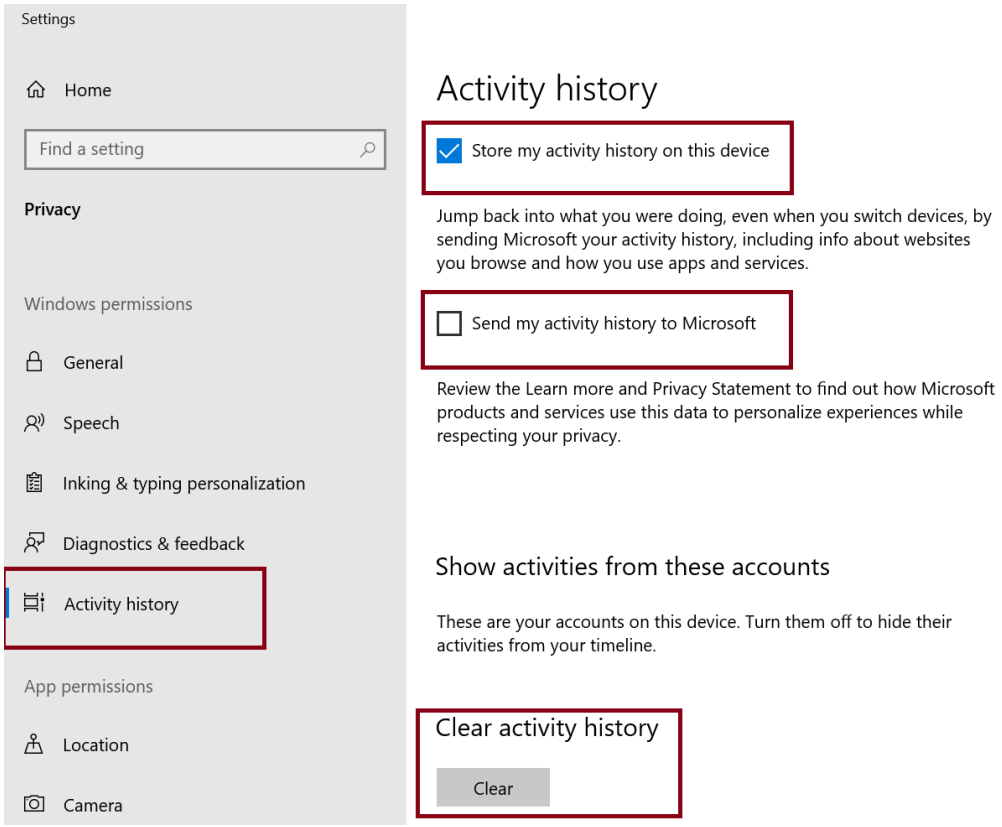
[18] https://en.wikipedia.org/wiki/Prefetcher
[19] https://www.hackthebox.com/blog/how-to-detect-psexec-and-lateral-movements
[20] https://www.nirsoft.net/utils/win_prefetch_view.html
[21] https://docs.velociraptor.app/docs/forensic/evidence_of_execution/
[22] https://4n6shetty.com/How-Windows-Artifact-Prefetch-Can-Help-in-Digital-Forensics-Investigations-in-Windows-11-Machine
[23] https://patents.google.com/patent/US6317818B1/en

Activity History

The goal of "Activity History" is to keep track of the thing the user is doing on a specific device (applications/services in use, files opened, website browsed). This information can be used to personalize the experience while using Windows. Examples for that are ordering the user activities based on duration of use or anticipating the user needs based on their activities[24].

By default, the "Activity History" is stored locally, however if we give permissions and logon with a school/work account Windows can send the information collected to Microsoft - as shown in the settings' screen in the screenshot below (Settings->Privacy->Activity History). By sending the information to Microsoft the user can jump back into activities that have been done in different devices (this is not configured by default).

Lastly, "Activity History" is used by different Windows features (Timeline and Microsoft Edge - more on that in future writeups). Also, Beside disabling the ability to store "Activity History" we can also "Clear Activity History" - as shown in the screenshot below.



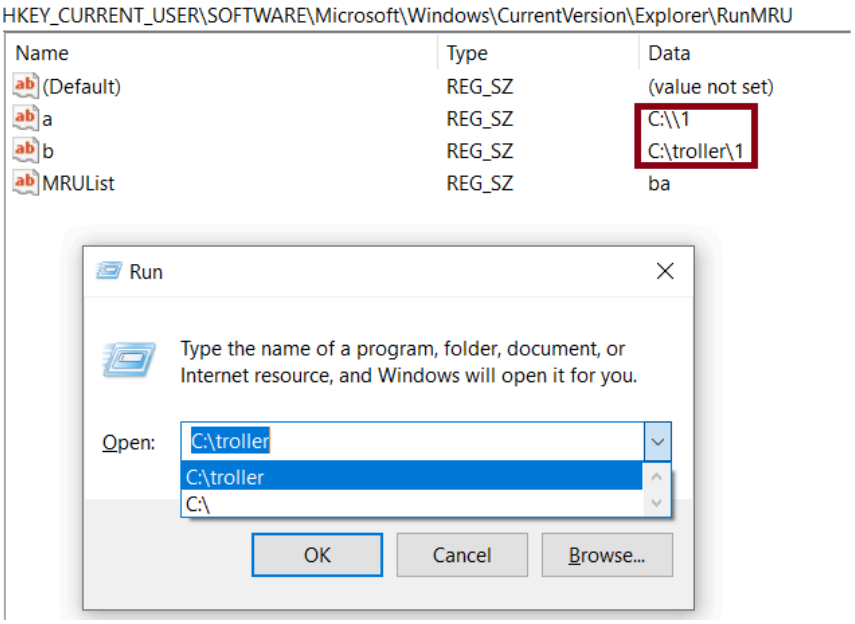
[24] https://support.microsoft.com/en-us/windows/-windows-activity-history-and-your-privacy-2b279964-44ec-8c2f-e0c2-6779b07d2cbd

Run MRU (Run Dialog Box Most Recently Used)

When using the "Run" command box ("Winkey+R") users can directly launch programs or open files/folders. "Run" includes a dropdown list of the last commands executed - as shown in the screenshot below. Those commands are saved in the registry under the "RunMRU" key[25] MRU in that case stands for "Most Recently Used.

Overall, "RunMRU" is saved separately for each Windows user (local/domain) in the following registry location: "HKCU\Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU" which we can access while the operating system is running (online analysis). For an offline analysis we can read the information for the NTUSER.DAT file ("Software\Microsoft\Windows\CurrentVersion\Explorer\RunMRU").

Moreover, each command is saved in a different value and the "MRUList" contains a list of all the commands to show and in what order. Also, each command is saved with a suffix with "\1" - as shown in the screenshot below. We can also clear the "RunMRU" history by removing the keys and values detailed above[26]. Lastly, "RunMRU" is not the MRU list in Windows there are others like "Microsoft Office MRU".



[25] https://forensafe.com/blogs/runmrukey.html
[26] https://www.thewindowsclub.com/clear-most-recently-used-mru-list?expand_article=1

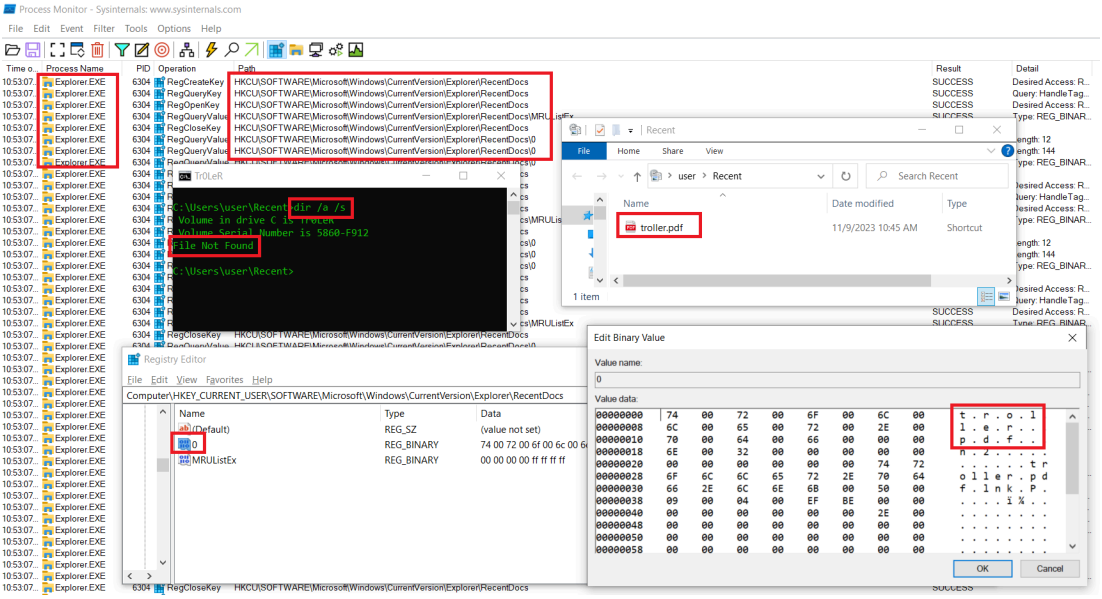
Recent Docs (Recently Used Documents)

Overall, “RecentDocs” is a key in the registry located at the following location: “HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs”. We can see there a list of recently accessed files (documents/images/presentations/links/etc).

Thus, this key holds a list of accessed files using “File Explorer”[27] by the currently logged on user. Those are basically the entries we see if we open the directory “C:\users\%username%\Recent” using “File Explorer”[28]. By the way, we can also use “%userprofile%\recent”.

However, if we open the “C:\users%username%\Recent” folder using “cmd.exe”[29] and list for files (including hidden files) we won’t see any file there - as shown in the screenshot below. By the way, the RecentDocs key also has an MRU list (MRUListEx) of type REG_BINARY, which gives the access order of the files[30].

Moreover, opening the same directory with “File Explorer” will show up the entries. Using “Process Monitor” we can verify that “explorer.exe”[31] parses the “RecentDocs” registry key in order to show files accessed - as shown in the screenshot below. Lastly, in “RecentDocs” has some correlation with the “%appdata%\Microsoft\Windows\Recent” - more on that in a separate writeup.



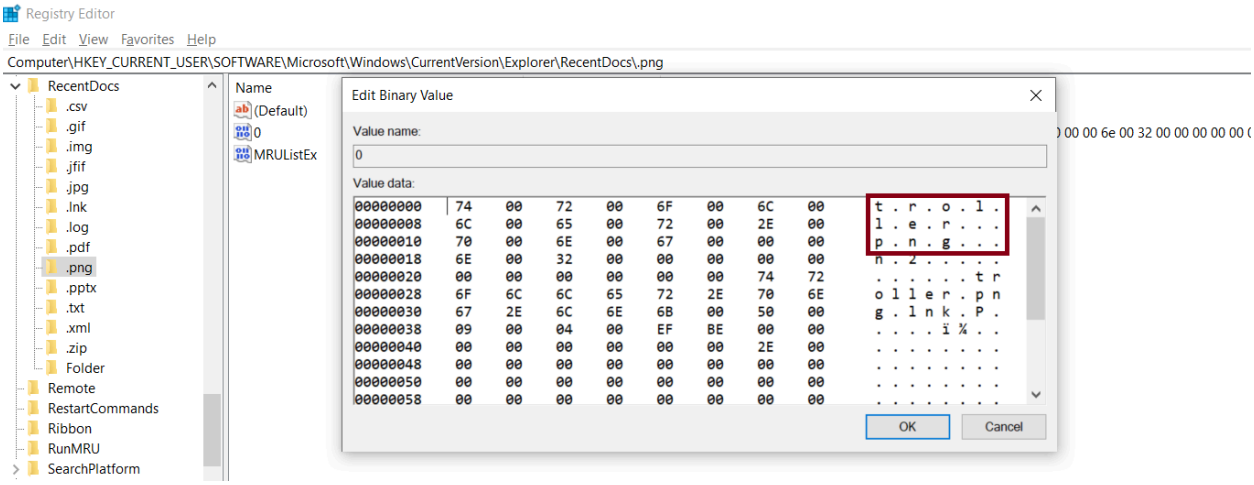
[27] https://medium.com/@boutnaru/the-windows-process-journey-explorer-exe-windows-explorer-9a96bc79e183
[28] https://digitalf0rensics.wordpress.com/2014/01/17/windows-registry-and-forensics-part2/
[29] https://medium.com/@boutnaru/the-windows-process-journey-cmd-exe-windows-command-processor-501be17ba81b
[30] https://forensic4cast.com/2019/03/the-recentdocs-key-in-windows-10/
[31] https://medium.com/@boutnaru/the-windows-process-journey-explorer-exe-windows-explorer-9a96bc79e183

Recent Docs by Extension (Recently Used Documents by Extension)

The “RecentDocs”[32] registry key also has subkeys which are per extensions (such as “csv”/”gif”/”jpg”/”lnk”/”log”/”zip”/”xml”/”txt”/”pdf”/”ppt”/”pptx”/etc) and also for folders (on folders I am going to elaborate as part of a separate writeup) - as shown in the screenshot below.

Moreover, each of the extension's subkeys has its own “MRUListEx” with information regarding files from the same extension. Thus, we basically have duplicate data. For example if we open a “*.png” file it will appear both in “Computer\HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs” and in “HKCU\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs\.png”[33] - as shown in the screenshot below.

Lastly, besides getting the list of files we can also get a quick view of types of files that have been accessed by a specific user (remember that the information is stored in HKCU). Alos, it is an easy way to understand what is the last file accessed based on a specific extension[34].



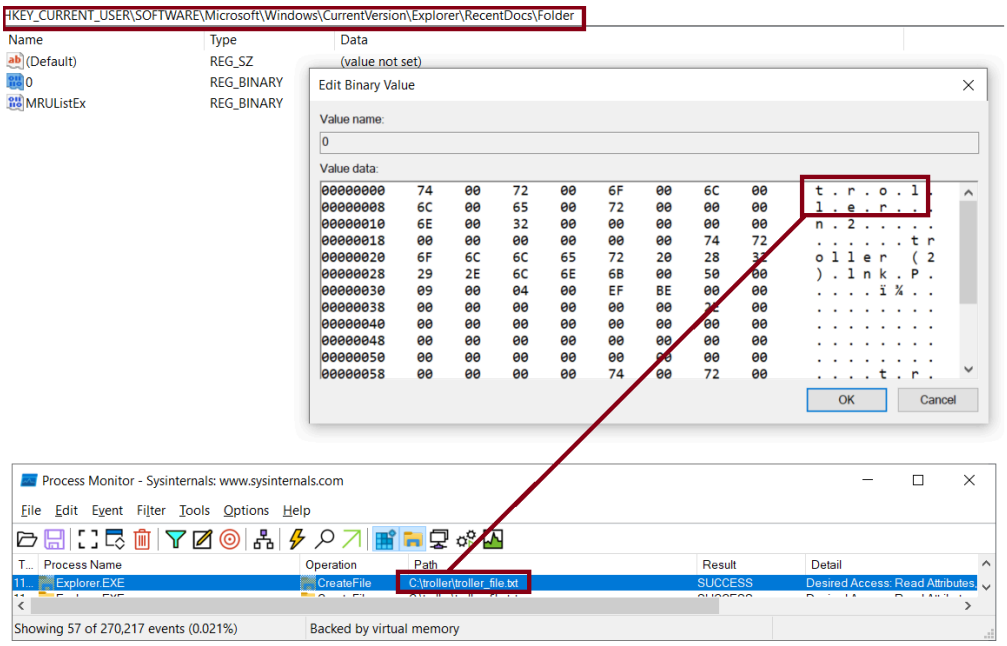
[32] https://medium.com/@boutnaru/the-windows-forensic-journey-recent-docs-recently-used-documents-a6d092d945ce
[33] https://forensic4cast.com/2019/03/the-recentdocs-key-in-windows-10/
[34] https://forensafe.com/blogs/recentdocs.html

Folder of RecentDocs (Folder/s of Recently Used Documents)

The "RecentDocs"[35] registry key has subkeys for file extensions[36] and in conjunction with a subkey for folder/s. The key is located in the following "registry path"- "HKCU\SOFTWARE\Microsoft\Windows\Current Version\Explorer\RecentDocs\Folder".

Overall, the "Folder" subkey contains the folder of recently opened files. However, the folder is included without a drive letter and the part folder. Thus, opening a folder is not enough to trigger the collection of the information, we need to specifically open a file from the directory[37] - as shown in the screenshot below.

Lastly, we can use this indication regarding folder/s from which files were opened from even if it has been deleted since. Moreover, due to the fact the information is contained in HKCU we know which user has opened files for the specific folder/s.



[35] https://medium.com/@boutnaru/the-windows-forensic-journey-recent-docs-recently-used-documents-a6d092d945ce

[36] https://medium.com/@boutnaru/the-windows-forensic-journey-recent-docs-by-extension-recently-used-documents-by-extension-ff6deb94e880

[37] https://www.forensicfocus.com/articles/forensic-analysis-of-the-windows-registry/