SheHacks_KE

299 Followers About Follow

Windows ShellBags from an Offline Hive (Part 2)

SheHacks_KE Jul 21 · 5 min read

Introduction

This is a continuation of the Windows ShellBags article published in the SheHacks Medium. Let's do a quick recap, Microsoft Windows records the view preferences of folders and Desktop.

Therefore, when the folder is visited again, Windows can remember the location of the folder, view and positions of items. Microsoft Windows store the view preferences in the registry keys and values known as "ShellBags".

The properties of a folder contained within Windows Shellbags can be significant to a computer forensic investigation. It allows for an assessment on whether the content of the folder could have been viewed simply from the user accessing it. Also, it allows to determine whether the user has changed the default settings of the folder which can be compelling in a case where the folder contains documents or unlawful images for example.

Windows Shellbags can also provide evidence of access of external or removable devices that are no longer connected to the computer.

In this issue, we shall tackle offline registry analysis of the shellbags. In this scenario a digital forensics investigator only needs to confirm the connection of removable device from a suspect machine.

Case Study

How do you prove insider threat? How can one ascertain that a removable thumb drive was previously connected to a machine?

Now, let's clarify the difference between the windows registry and the hive. A hive is a logical group of keys, subkeys, and values in the registry that has a set of supporting files loaded into memory when the operating system is started or when a user logs in. Each time a new user logs on to a computer, a new hive is created for that user with a separate file for the user profile.

The Windows Registry is a hierarchical collection of databases that consist of operating system configurations. The Registry provides a centralized method of storing custom preferences for each Windows user, rather than storing them as individual .INI files.

Offline Analysis

For offline analysis, we first have to extract the shellbags file which is USRCLASS.DAT. Let's extract the ShellBag file using FTK imager. Download FTK imager from here.

Then add in the evidence, go to the add evidence item.

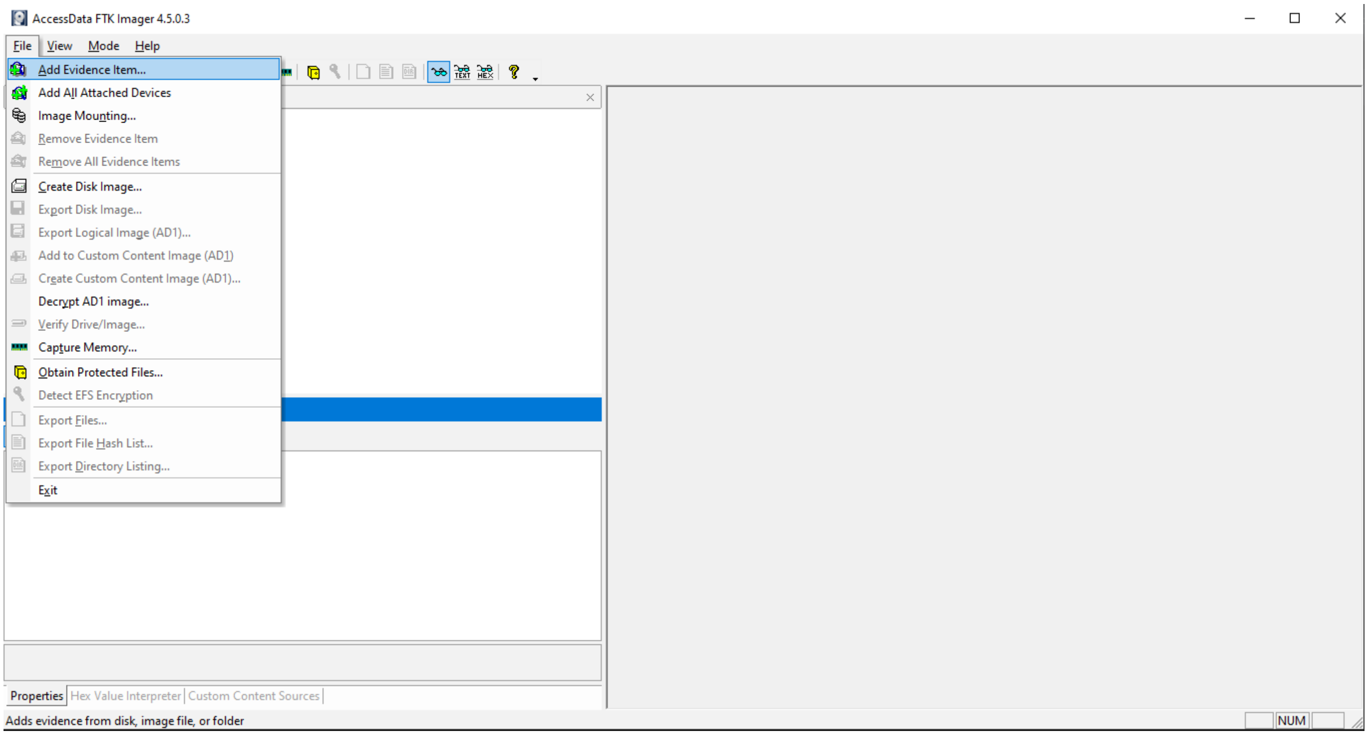
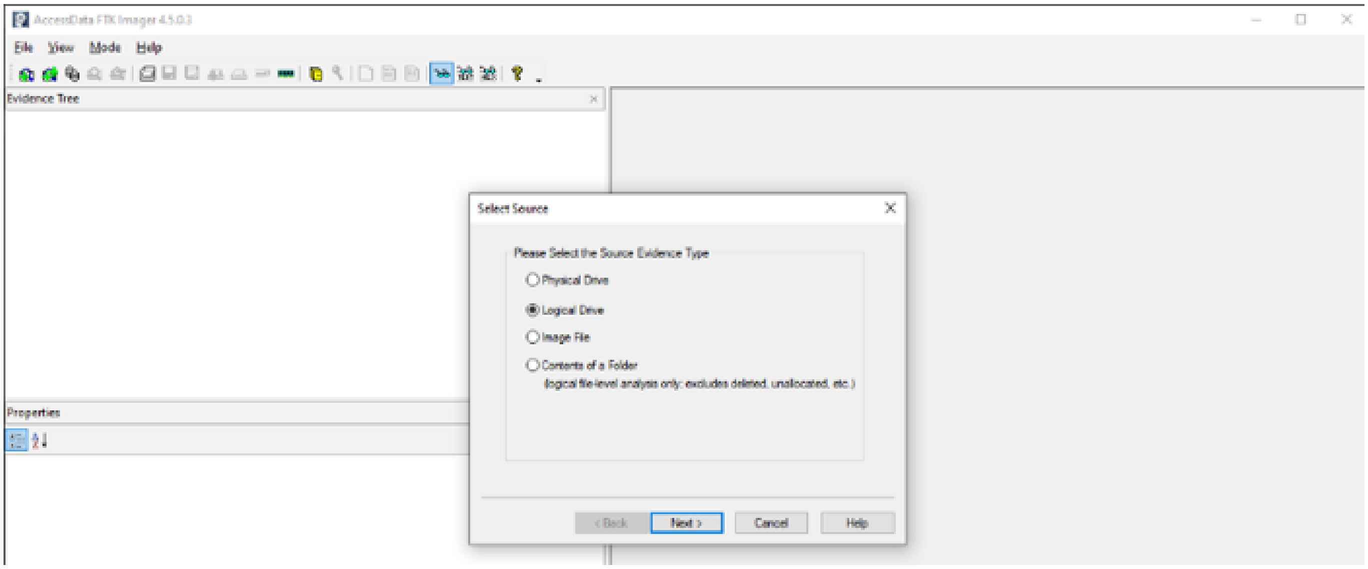


Figure 1 Extracting Offline Hive

Select the source for adding evidence as below, I have selected the logical drive as usrclass.dat is present in the C drive. Please note that we are extracting the offline hive from the suspect machine. This means that the FTK Imager software has to be installed in that computer from which the hive is being captured.



Please Select the Source Evidence Type

○ Physical Drive

● Logical Drive

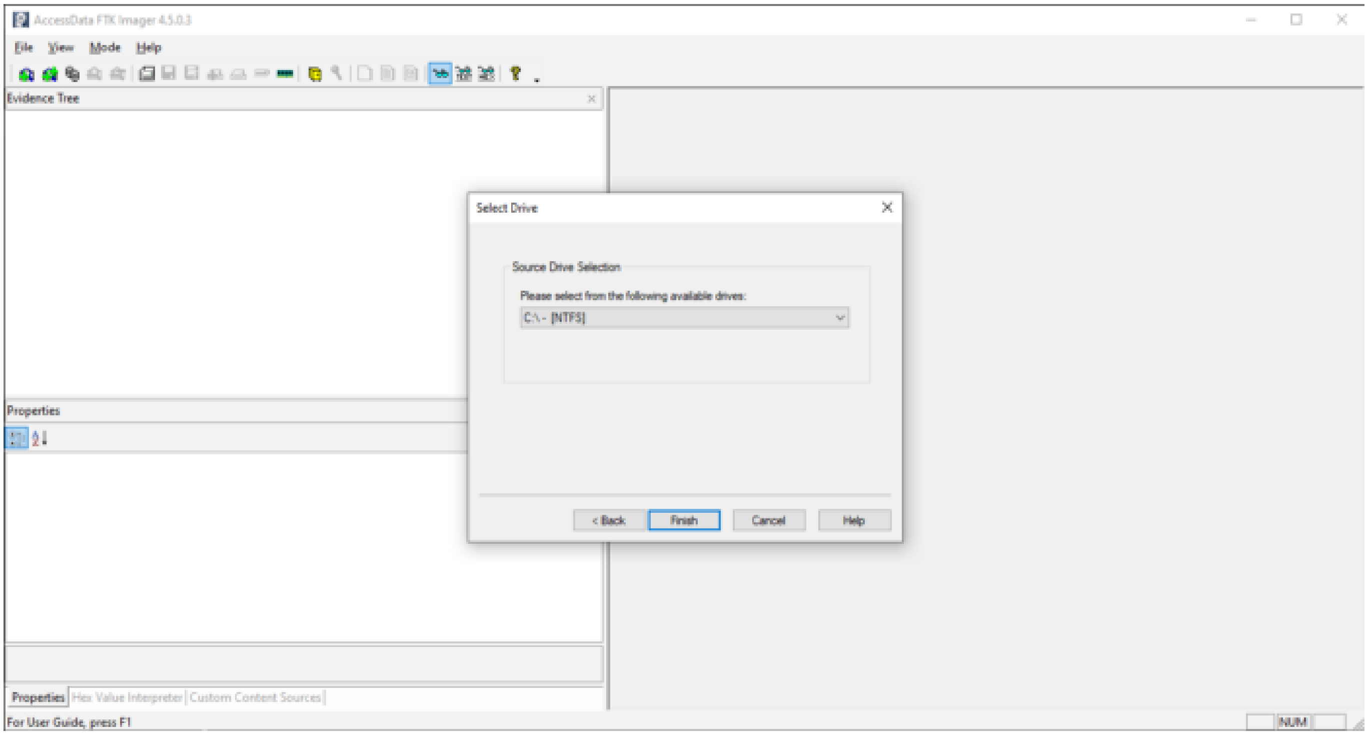
○ Image File

○ Contents of a Folder
(logical file-level analysis only; excludes deleted, unallocated, etc.)

Figure 2 usrclass.dat is a logical drive

A logical drive is a drive space that is logically created on top of a physical hard disk drive. A logical drive is a separate partition with its own parameters and functions, and it operates independently. It can also be called a logical drive partition or logical disk partition.

Next, select the desired user drive. Click Finish.



Source Drive Selection

Please select from the following available drives:

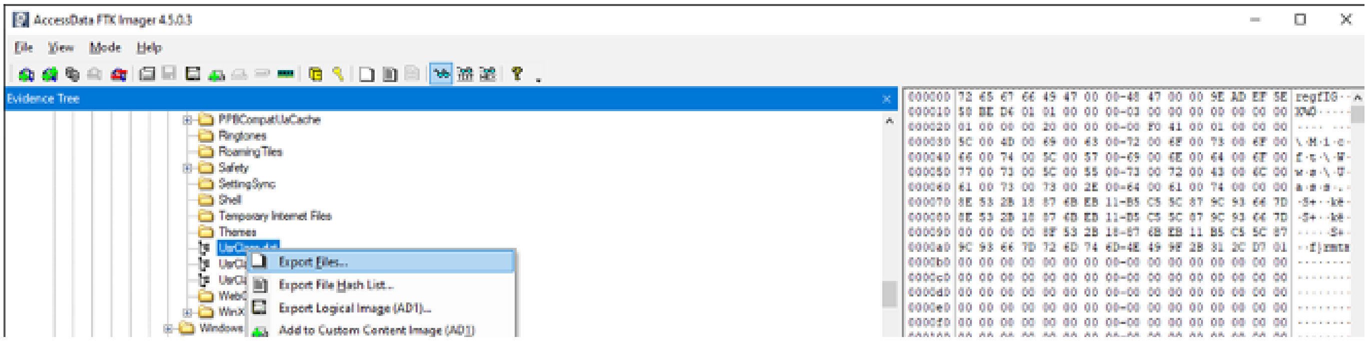
C:\ - [NTFS]

Figure 3 User Drive C:

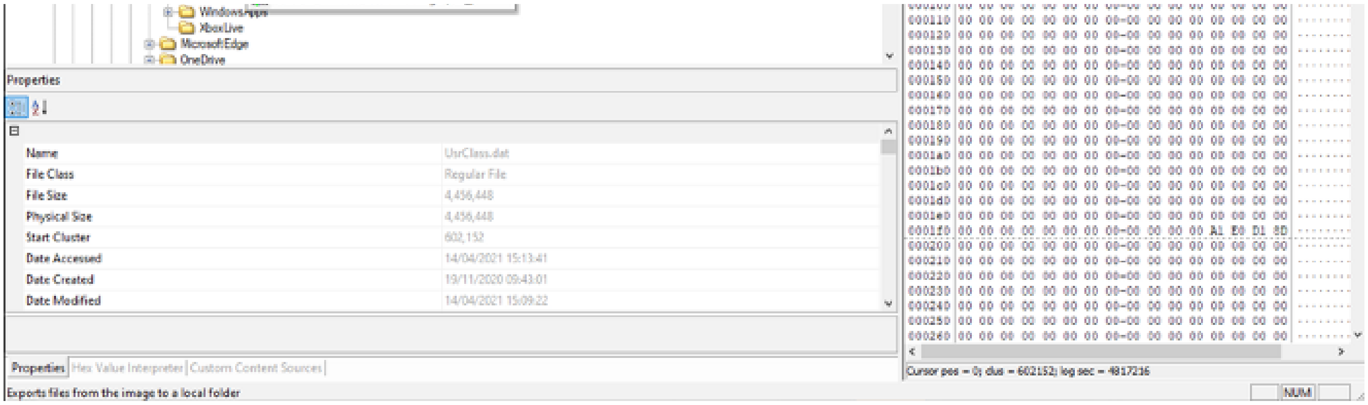
Expand the window to the location of the usrclass.dat. Select the user you want to investigate go to the following path to extract the UsrClass.dat.

root > users > administrator >Appdata>Local>Microsoft>windows

NB: in this case, I chose the user 'administrator'



```
000000 72 65 67 66 49 47 00 00-48 47 00 00 9E AD EF 5E regfIG··
000010 58 BE D6 01 01 00 00 00-03 00 00 00 00 00 00 00 XЖÖ·····
000020 01 00 00 00 20 00 00 00-00 F0 41 00 01 00 00 00 ····  ···
000030 5C 00 4D 00 69 00 63 00-72 00 6F 00 73 00 6F 00 \·M·i·c·
000040 66 00 74 00 5C 00 57 00-69 00 6E 00 64 00 6F 00 f·t·\·W·
000050 77 00 73 00 5C 00 55 00-73 00 72 00 43 00 6C 00 w·s·\·U·
000060 61 00 73 00 73 00 2E 00-64 00 61 00 74 00 00 00 a·s·s·.·
000070 8E 53 2B 18 87 6B EB 11-B5 C5 5C 87 9C 93 66 7D ·S+··kë·
000080 8E 53 2B 18 87 6B EB 11-B5 C5 5C 87 9C 93 66 7D ·S+··kë·
000090 00 00 00 00 8F 53 2B 18-87 6B EB 11 B5 C5 5C 87 ·····S+·
0000a0 9C 93 66 7D 72 6D 74 6D-4E 49 9F 2B 31 2C D7 01 ··f}rmtm
0000b0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ········
0000c0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ········
0000d0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ········
0000e0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ········
0000f0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ········
```



Name	UsrClass.dat
File Class	Regular File
File Size	4,456,448
Physical Size	4,456,448
Start Cluster	602,152
Date Accessed	14/04/2021 15:13:41
Date Created	19/11/2020 09:43:01
Date Modified	14/04/2021 15:09:22

```
000100 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000110 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000120 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000130 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000140 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000150 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000160 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000170 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000180 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000190 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
0001a0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
0001b0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
0001c0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
0001d0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
0001e0 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
0001f0 00 00 00 00 00 00 00 00-00 00 00 00 A1 E0 D1 8D ........
000200 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000210 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000220 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000230 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000240 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000250 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
000260 00 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 ........
```

Cursor pos = 0; clus = 602152; log sec = 4817216

Exports files from the image to a local folder

Figure 4 Exporting UsrClass.dat

We will be analysing the usrclass.dat extracted from the above step using shell bag explorer GUI version by Eric Zimmerman. Downloadable here. Ensure to run the tool as admin in order to get full feature capabilities.

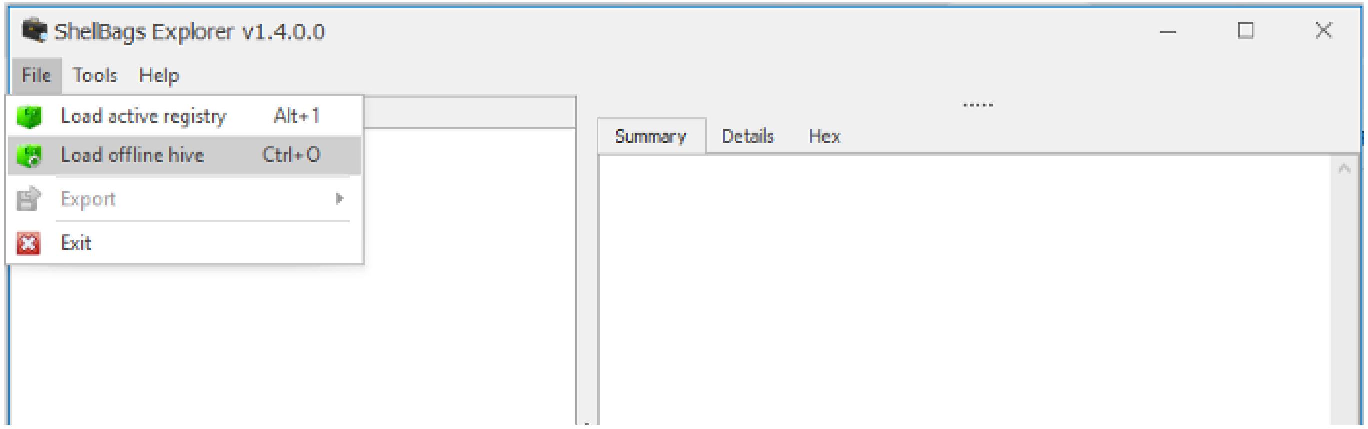
Transfer the *UsrClass.dat file* to your workstation/ where you will be doing your analysis. Remember the analysis will be done using the Shell Bag Explorer Tool.

As we have exported the registry hives we will choose "load offline hive" as seen below.

NB:

Ensure to run the Shell Bag Explorer Tool as an Administrator. This allows an analyst full feature access to the tool. As you upload the offline hive *(UsrClass.dat)*, long press the *SHIFT Button* to allow parsing even if the Hive is 'dirty'.

Data parsing is a method where one string of data gets converted into a different type of data. A parser will take the raw data and transform it into a more readable data format that can be easily read and understood.



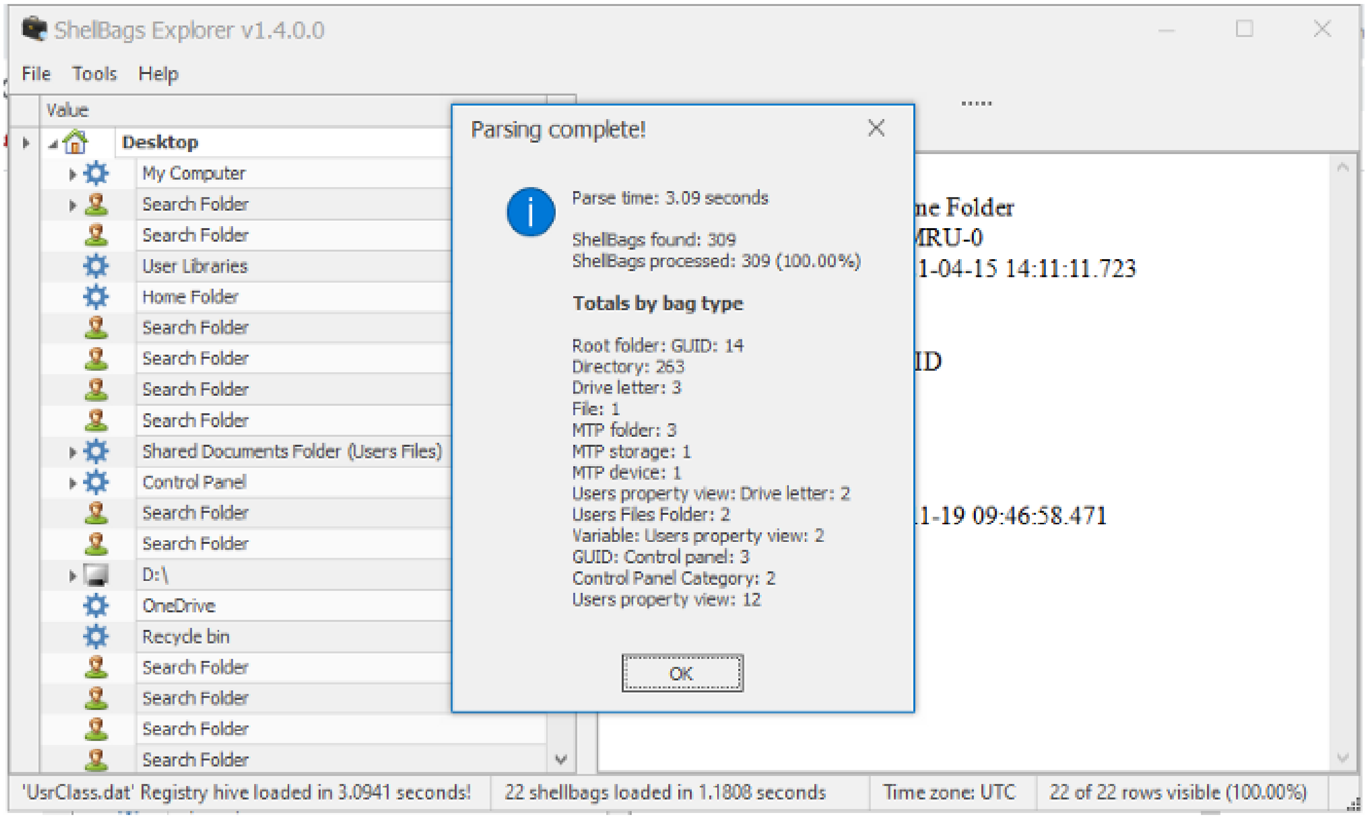
Active file: NA

Time zone: UTC

0 of 0 visible

Figure 5 Loading Offline Hive

After successful parsing of the extracted shellbags file, you will be able to see the entries for folders browsed, created, deleted, etc.



Parsing complete!

Parse time: 3.09 seconds

ShellBags found: 309
ShellBags processed: 309 (100.00%)

Totals by bag type

Root folder: GUID: 14
Directory: 263
Drive letter: 3
File: 1
MTP folder: 3
MTP storage: 1
MTP device: 1
Users property view: Drive letter: 2
Users Files Folder: 2
Variable: Users property view: 2
GUID: Control panel: 3
Control Panel Category: 2
Users property view: 12

'UsrClass.dat' Registry hive loaded in 3.0941 seconds!

22 shellbags loaded in 1.1808 seconds

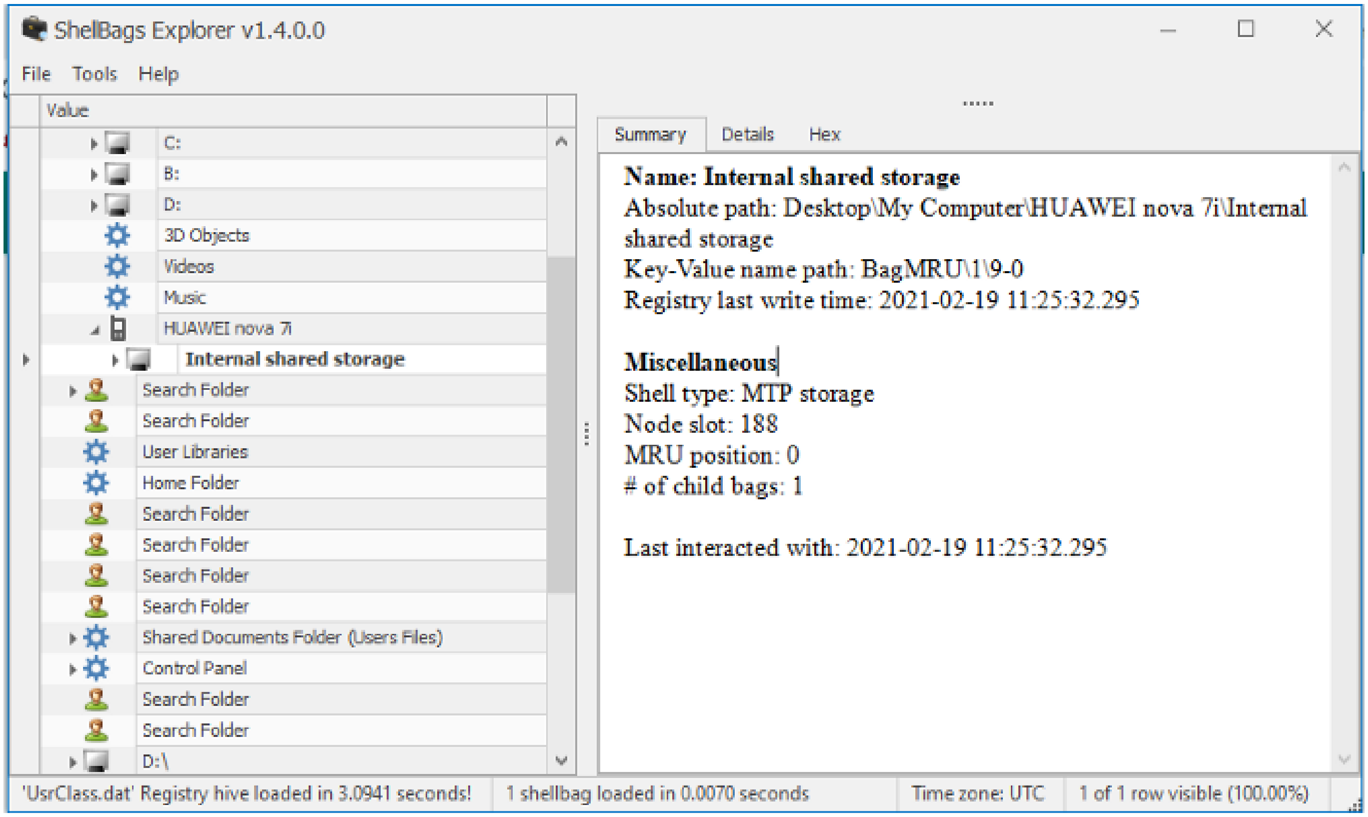
Time zone: UTC

22 of 22 rows visible (100.00%)

Figure 6 successful data parsing

Shellbags stores the entries of the directories accessed by the user, user preferences such as window size, icon size. Shellbags explorer parses the shellbags entries shows the absolute path of the directory accessed, creation time, file system, child bags. The tool classifies the folders accessed according to the location of the folder. Shellbags are created for compressed files (ZIP files), command prompt, search window, renaming, moving, and deleting a folder.

Below is an entry of a removable device. We can see a *HUAWEI nova 7i* was plugged into the machine on 19th Feb 2021. This crucial evidence is admissible in a court of law!



Name: Internal shared storage
Absolute path: Desktop\My Computer\HUAWEI nova 7i\Internal shared storage
Key-Value name path: BagMRU\1\9-0
Registry last write time: 2021-02-19 11:25:32.295

Miscellaneous
Shell type: MTP storage
Node slot: 188
MRU position: 0
# of child bags: 1

Last interacted with: 2021-02-19 11:25:32.295

'UsrClass.dat' Registry hive loaded in 3.0941 seconds! | 1 shellbag loaded in 0.0070 seconds | Time zone: UTC | 1 of 1 row visible (100.00%)

Figure 7 Previously connected removable device

With this information, an analyst can go further to investigate the actual data transferred between the removable device and the suspect machine using different techniques i.e., memory dump analysis.

This will save an analyst time searching for a needle in the haystack since it is known that a removable device was involved. Otherwise, full memory analysis can consume a lot of time and energy especially if one does not know what to look for.

Conclusion

We have seen how to promptly investigate insider threat instances i.e., using removable disk connections on suspect machines.

Starting out a digital forensic investigation by first analyzing Windows ShellBags, help to tell the story of: what and when was deleted, renamed, hidden, zipped, moved, transferred and removable drivers connected.

Keeping in mind the main reason for Microsoft creating Windows ShellBags was to improve the user's experience. It has turned out to be a favorite Windows Forensic Artifact as a hidden data source. One can also get timestamp data and do live triaging for Incidence Response Instances!

By Stella Magana

Twitter: @STLmagana

Shellbags Windows Cybersecurity Shehacks Hive