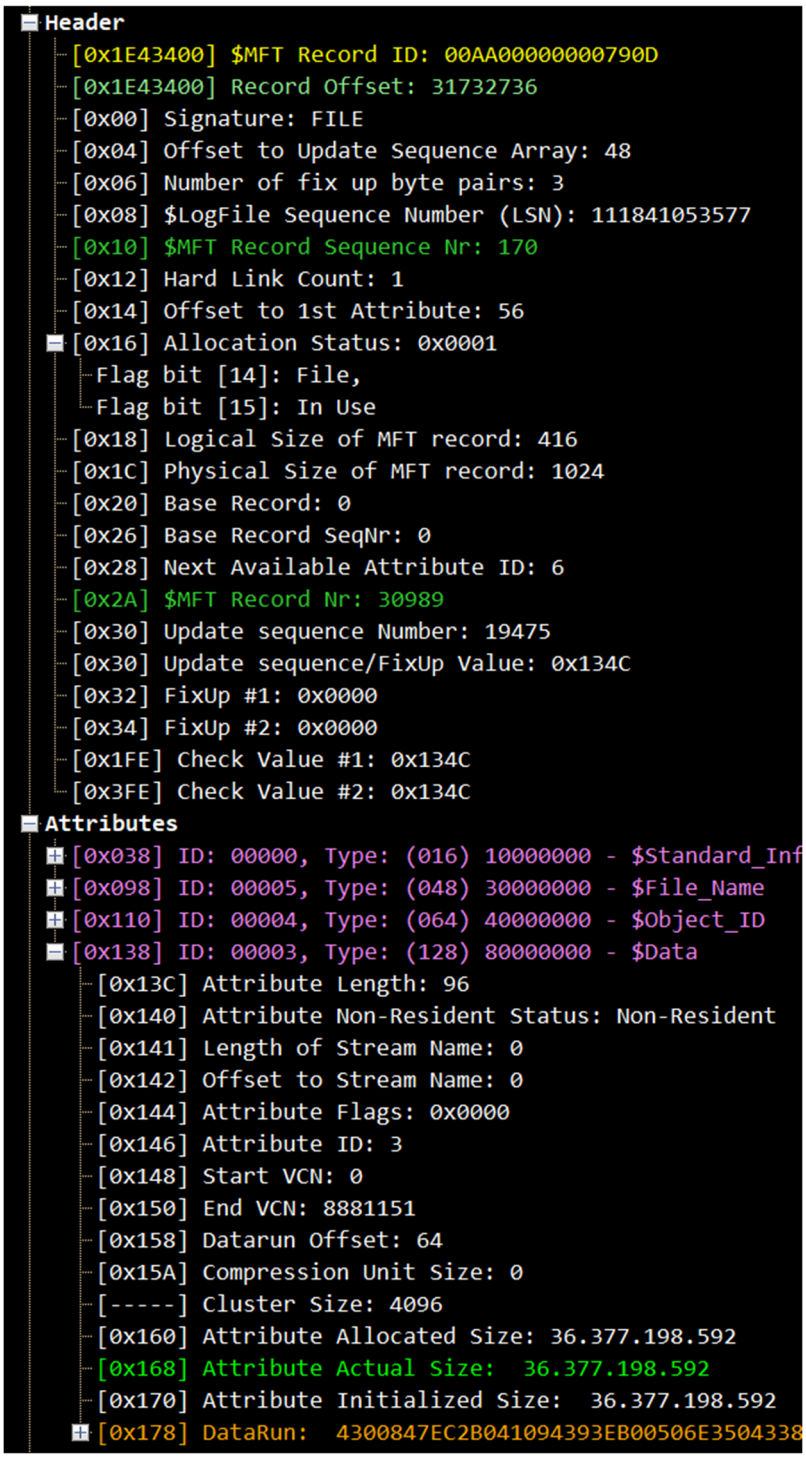
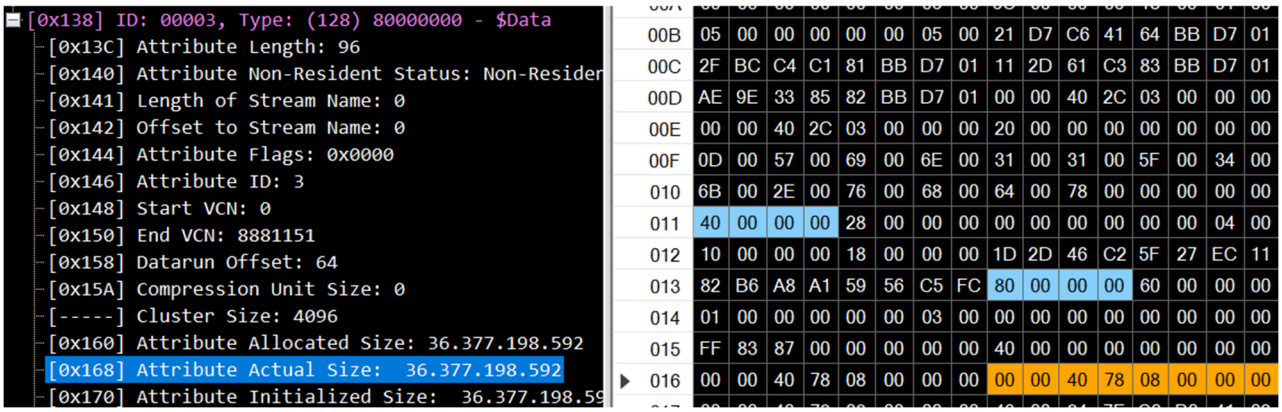
$MFT Record:


Header
[0x1E43400] $MFT Record ID: 00AA00000000790D
[0x1E43400] Record Offset: 31732736
[0x00] Signature: FILE
[0x04] Offset to Update Sequence Array: 48
[0x06] Number of fix up byte pairs: 3
[0x08] $LogFile Sequence Number (LSN): 111841053577
[0x10] $MFT Record Sequence Nr: 170
[0x12] Hard Link Count: 1
[0x14] Offset to 1st Attribute: 56
[0x16] Allocation Status: 0x0001
Flag bit [14]: File,
Flag bit [15]: In Use
[0x18] Logical Size of MFT record: 416
[0x1C] Physical Size of MFT record: 1024
[0x20] Base Record: 0
[0x26] Base Record SeqNr: 0
[0x28] Next Available Attribute ID: 6
[0x2A] $MFT Record Nr: 30989
[0x30] Update sequence Number: 19475
[0x30] Update sequence/FixUp Value: 0x134C
[0x32] FixUp #1: 0x0000
[0x34] FixUp #2: 0x0000
[0x1FE] Check Value #1: 0x134C
[0x3FE] Check Value #2: 0x134C
Attributes
[0x038] ID: 00000, Type: (016) 10000000 - $Standard_Inf
[0x098] ID: 00005, Type: (048) 30000000 - $File_Name
[0x110] ID: 00004, Type: (064) 40000000 - $Object_ID
[0x138] ID: 00003, Type: (128) 80000000 - $Data
[0x13C] Attribute Length: 96
[0x140] Attribute Non-Resident Status: Non-Resident
[0x141] Length of Stream Name: 0
[0x142] Offset to Stream Name: 0
[0x144] Attribute Flags: 0x0000
[0x146] Attribute ID: 3
[0x148] Start VCN: 0
[0x150] End VCN: 8881151
[0x158] Datarun Offset: 64
[0x15A] Compression Unit Size: 0
[-----] Cluster Size: 4096
[0x160] Attribute Allocated Size: 36.377.198.592
[0x168] Attribute Actual Size: 36.377.198.592
[0x170] Attribute Initialized Size: 36.377.198.592
[0x178] DataRun: 4300847EC2B041094393EB00506E3504338

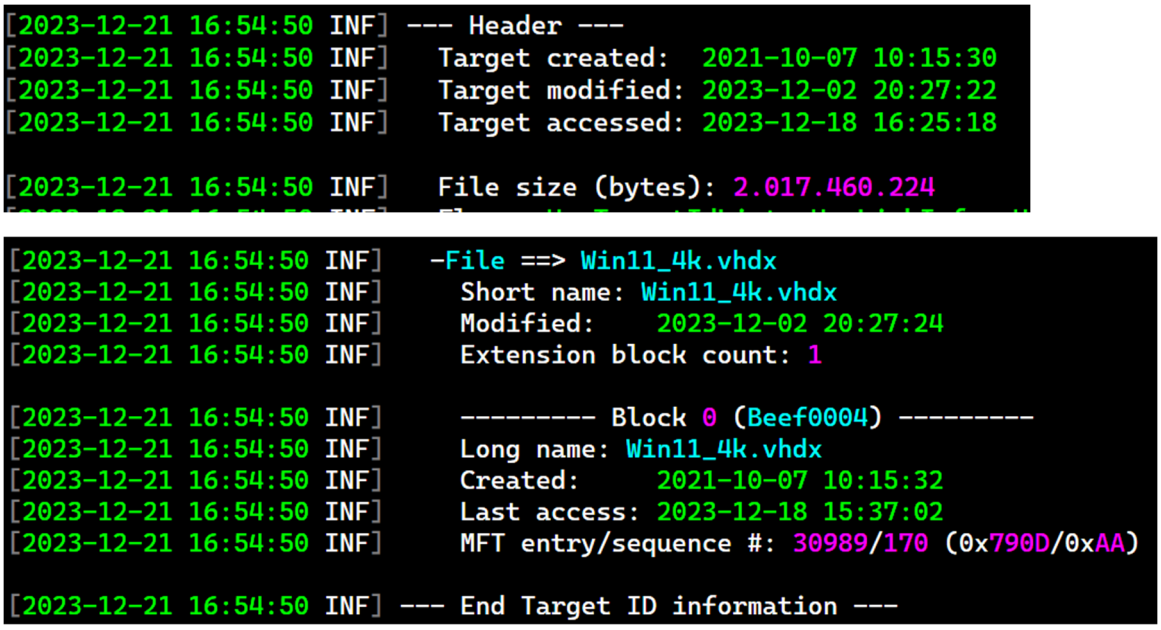
$Data attribute's File Size:



"0x 00 00 40 78 08 00 00 00"

Shell LNK pointing to the same file:

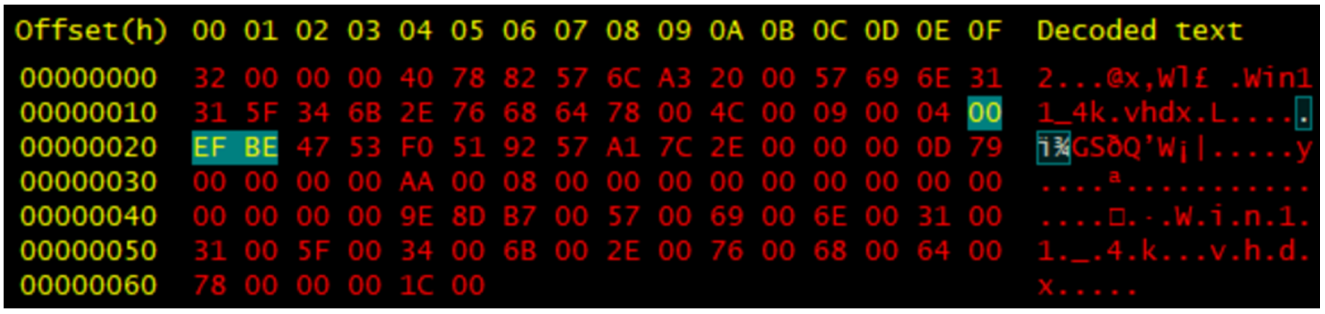
LECmd *(version 1.5.0.0)* reports wrong File Size:



So does Jumplist_Browser (up-to v.0.0.36.0):

```
IDList Entry #002 [32]
  ItemID Size: 102
  ItemID Type: 32
  File Size: 2017460224
  Attributes: Archive (0x0020)
  Ansi Name: Win11_4k.vhdx
  Modified: 02-Dec-2023 20:27:24.00
  Extension #0 Type: [BEEF0004]
    Extension Size: 76
    Extension Version: 9
    Host OS: Windows 8.1/10/11
    FS Hint: NTFS
    Unicode Name: Win11_4k.vhdx
    Created: 07-Oct-2021 10:15:32.00
    Accessed: 18-Dec-2023 15:37:02.00
    MFT Record Nr: 30989
    MFT Record Sequence Nr: 170
    Unknown value #1: 0x08000000
    Unknown value #3: 0x9E8DB700
    Extension Data
  ItemID Data
```

Hex view of ItemID:



Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	Decoded text
00000000	32	00	00	00	40	78	82	57	6C	A3	20	00	57	69	6E	31	2...@x,Wl£ .Win1
00000010	31	5F	34	6B	2E	76	68	64	78	00	4C	00	09	00	04	00	1_4k.vhdx.L.....▯
00000020	EF	BE	47	53	F0	51	92	57	A1	7C	2E	00	00	00	0D	79	ï¾GSðQ'W¡|.....y
00000030	00	00	00	00	AA	00	08	00	00	00	00	00	00	00	00	00	....ª..........
00000040	00	00	00	00	9E	8D	B7	00	57	00	69	00	6E	00	31	00	....□.·.W.i.n.1.
00000050	31	00	5F	00	34	00	6B	00	2E	00	76	00	68	00	64	00	1._.4.k...v.h.d.
00000060	78	00	00	00	1C	00											x.....

Offsets 2 -4 are the File size *(32bit with max value: 4.294.967.295*) – which in files with size within this limit works fine.

In this example it is: “0x 00 00 40 78” giving a File Size value of: 2.017.4460.224. But the real file size if greater than this value.

If we look at the ItemID’s Extension blocks (BE EF 00 xx) of the LNK, it has an Extension Type: [BEEF0004]:

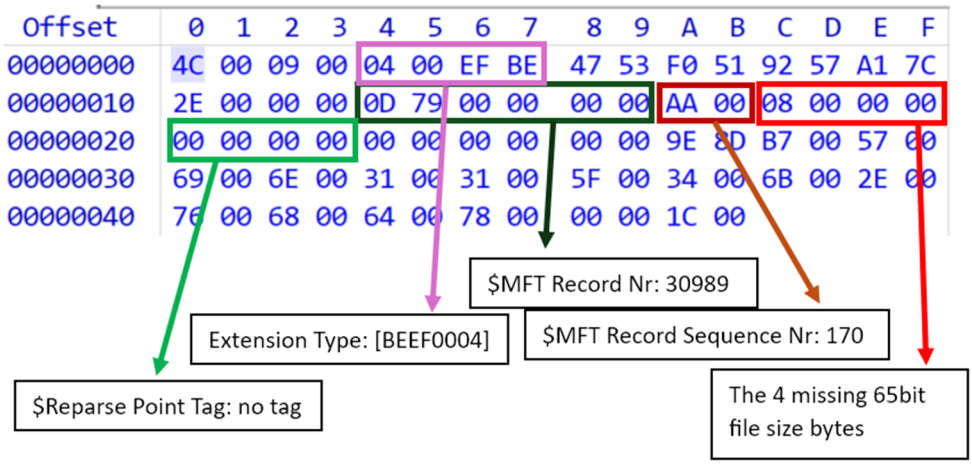
LECmd:

```
---------- Block 0 (Beef0004) ----------
Long name: Win11_4k.vhdx
Created:     2021-10-07 10:15:32
Last access: 2023-12-18 15:37:02
MFT entry/sequence #: 30989/170 (0x790D/0xAA)
```

Jumplist_Browser:

```
Extension #0 Type: [BEEF0004]
    Extension Size: 76
    Extension Version: 9
    Host OS: Windows 8.1/10/11
    FS Hint: NTFS
    Unicode Name: Win11_4k.vhdx
    Created: 07-Oct-2021 10:15:32.00
    Accessed: 18-Dec-2023 15:37:02.00
    MFT Record Nr: 30989
    MFT Record Sequence Nr: 170
    Unknown value #1: 0x08000000
```

Raw Hex:



So, if we add the 4 bytes (08 00 00 00”) after the $MFT information to the end of the 4 bytes in the ItemID above (“0x 00 00 40 78”), we get:

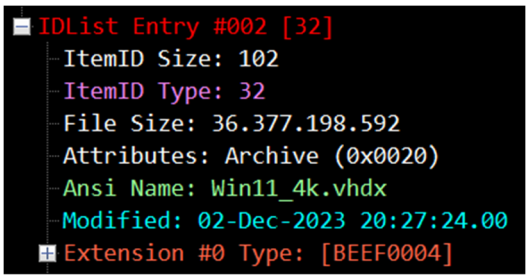
“0x 00 00 40 78” + “0x 08 00 00 00” => “0x 00 00 40 78 08 00 00 00” which in LE translates to

“0x 00 00 00 08 78 40 00 00” => 36.377.198.592:

HEX	8 7840 0000
DEC	36.377.198.592

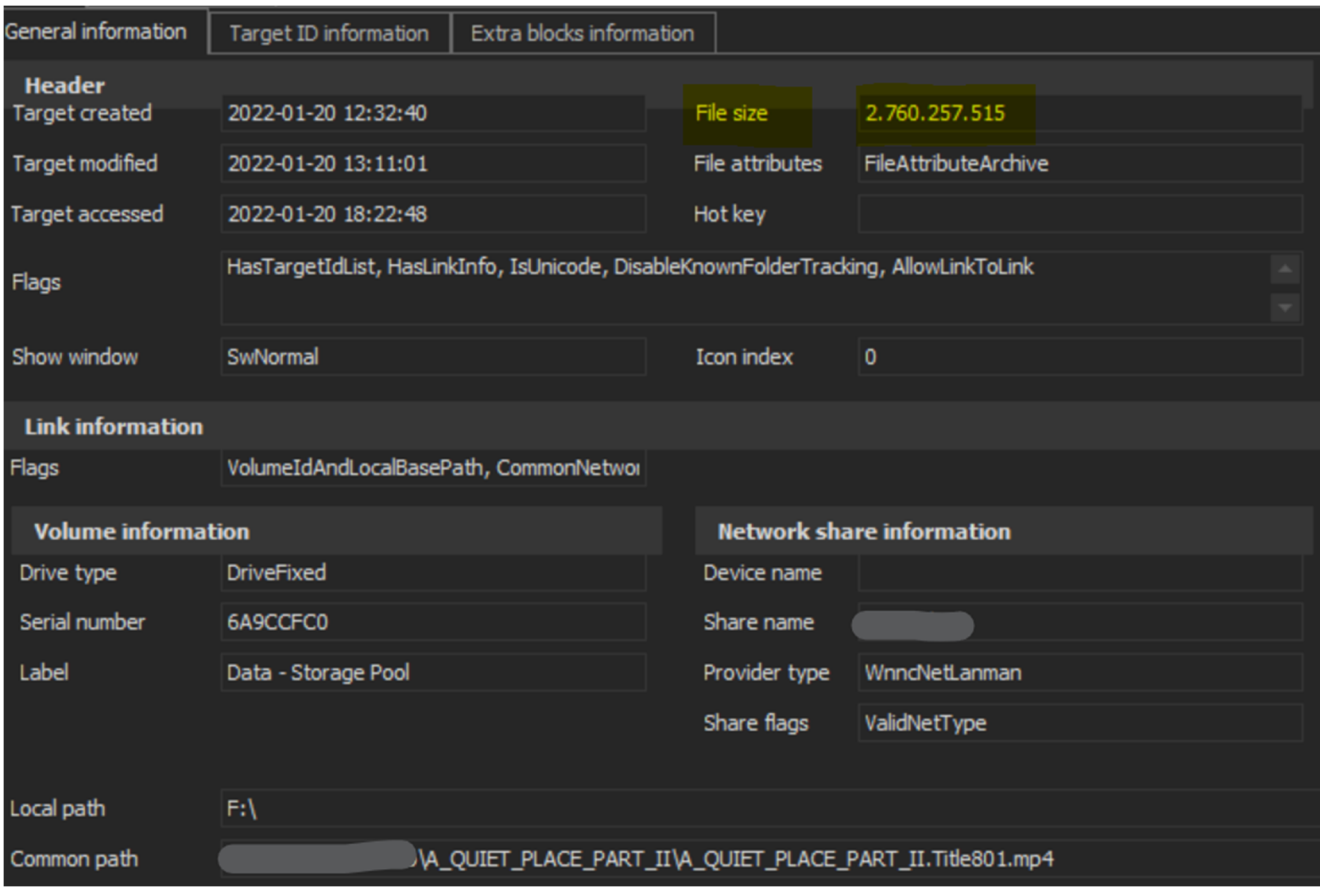
The same example with PowerShell:

```
PS5.1 C:\>$a = Hex-toByteArray -Hex "00004078"
$b = Hex-toByteArray -Hex "08000000"
$c = $a + $b
[System.BitConverter]::ToUInt64($c,0).ToString('N0')
36.377.198.592
```

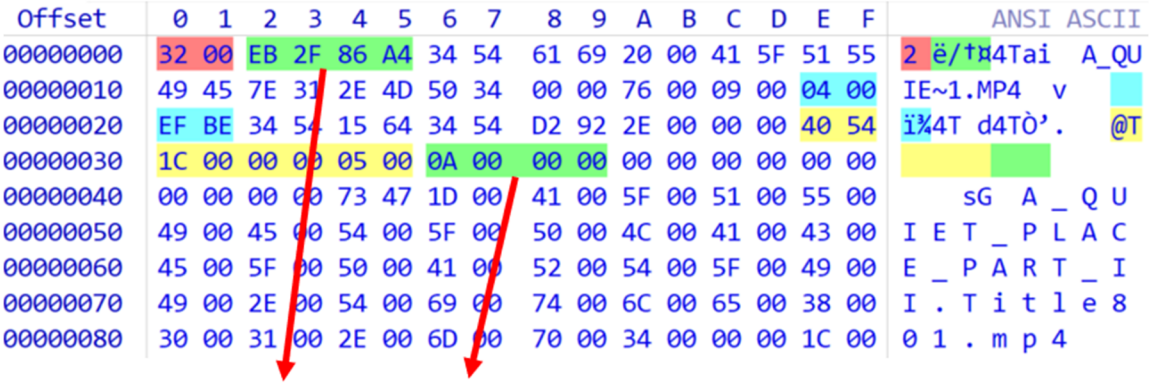


More examples:

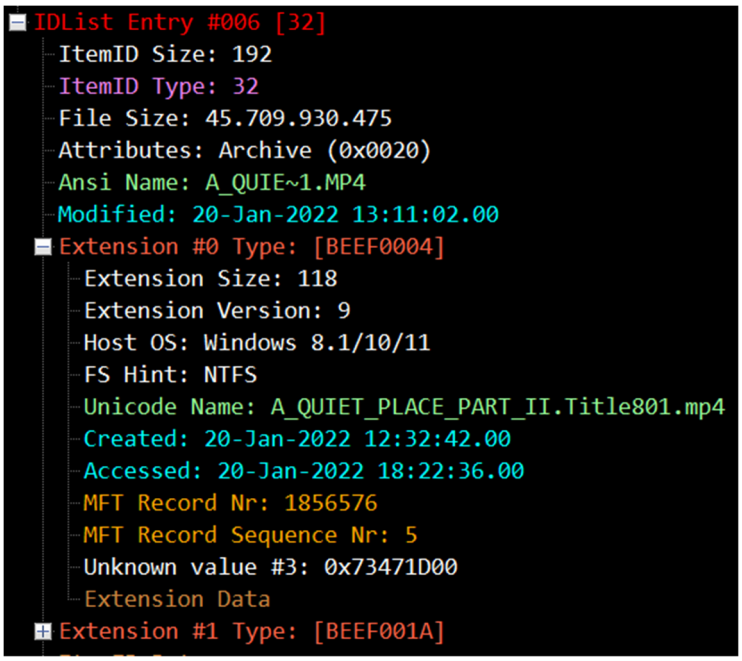
Automatic Destinations stream:
Jumplist Explorer v.2.0.0.0 => wrong File Size:



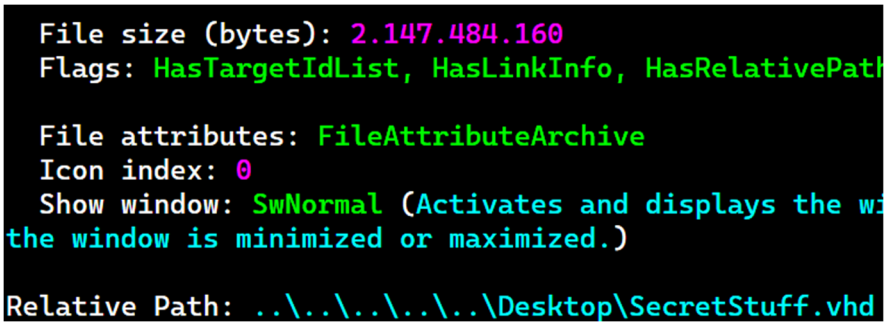
Raw Hex of ItemID:

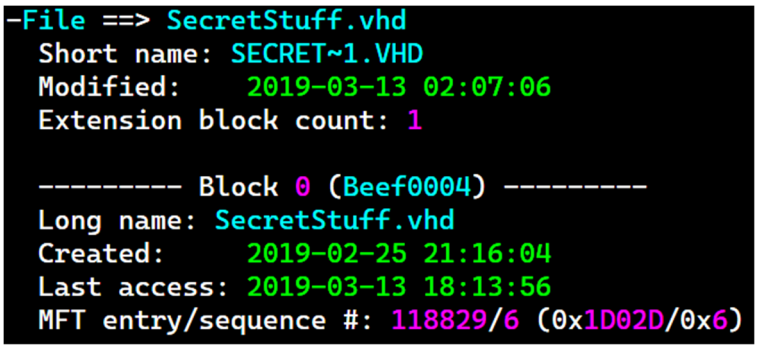


Actual 64bit File Size = 0xEB2F86A4 + 0x0A000000 => 45.709.930.475

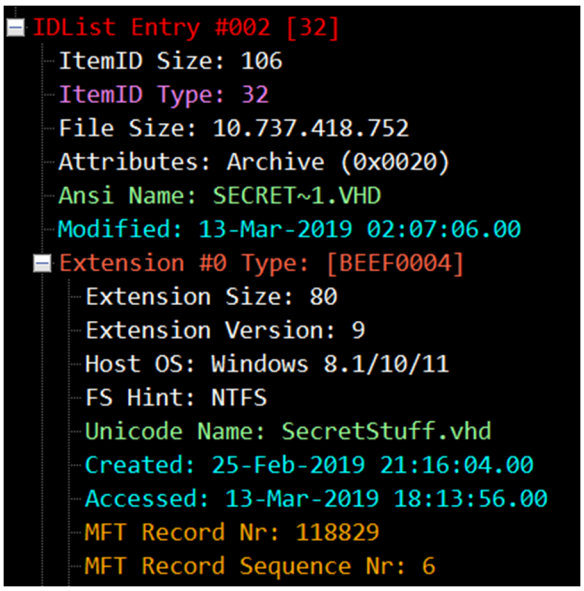


LNK Shotcut:





Actual File Size:


IDList Entry #002 [32]
ItemID Size: 106
ItemID Type: 32
File Size: 10.737.418.752
Attributes: Archive (0x0020)
Ansi Name: SECRET~1.VHD
Modified: 13-Mar-2019 02:07:06.00
Extension #0 Type: [BEEF0004]
Extension Size: 80
Extension Version: 9
Host OS: Windows 8.1/10/11
FS Hint: NTFS
Unicode Name: SecretStuff.vhd
Created: 25-Feb-2019 21:16:04.00
Accessed: 13-Mar-2019 18:13:56.00
MFT Record Nr: 118829
MFT Record Sequence Nr: 6