Investigating the use of VHD files by cybercriminals

By Christiaan Beek

Published: 2020-12-03 · Archived: 2026-04-07 02:02:13 UTC

Investigating the use of VHD files by cybercriminals

In recent investigations, I observed an adversary making use of a VHD attachment to a spear-phishing email being sent. VHD files are 'Virtual Hard-Disk' files. Originally the file format was introduced with Connectix Virtual PC and it can store the contents of a hard disk drive. Windows 7 and newer systems include the ability to manually mount VHD files. From Windows 8 and onwards, a user can mount a VHD by simply double-clicking on the file. Once mounted, a VHD disk image appears to Windows as a normal hard disk physically connected to the system.

Why would an adversary use a VHD file? To launch a document. Starting from Microsoft Office 10, when a document is tagged as "Mark of the Web" (MotW), the file will be opened in Protected view. (Mark of the Web is a technology used by MS to tag files with the Internet Security zone info from where they originated). In Windows 10, the SmartScreen technology will block files that are downloaded from the Internet from being executed.

The interesting thing with a VHD file is that files inside will not carry the MotW tag and can be executed without having to deal with the restrictions that belong to such files. That makes it attractive for adversaries to attempt using this technique.

In our case we have a VHD file that contains two files:

content of the VHD file

As I observe, two files are in the VHD, including a PDF document that, once opened, will launch the executable file. Analysis reveals that the executable is a trojan belonging to the Sednit family, often used by the Turla group. I will not go into details on the malware analysis since I want to focus more on the VHD file format and forensics.

$MFT

Since this is a sort of hard disk with a filesystem, there are several approaches to investigate it from a forensic perspective. Mounting the disk, I can carve for deleted files or extract files like the Master-File-Table (MFT) to inspect the file behaviors and interactions on the disk.

Once I have extracted the MFT, I can start to extract the information and put the output in a CSV format.

MFT analysis of VHD

In the above screenshot, I showcase a small part of the data. There are, from a forensic and incident-response & intelligence perspective, a few interesting things to observe. The files I discussed are copied and I can see the creation date and modification date. The VHD volume seems to have been created on October 21, 2020, while the files appear to have been copied and modified around November 10 and 11, 2020. Secondly, I observe the SIDs that were used. The Security Identifier in this case indicates a specific domain user.

S-1–5–21–635164469–325223577–1075005921–1001

S — Indicates it is a SID string.

1 — The version of the SID structure. Windows NT and later starts with 1.

5 — Identifier Authority. 5 = NT Authority.

21–635164469–325223577–1075005921– Domain identifier.

1001 — RID. Identifies the particular account or group.

More Details About the VHD?

Mounting the VHD file in a VM, I launched Windows PowerShell ISE in admin mode and installed the module 'PowerForensics'

Using this module, the filesystem can be queried for several types of information. In my screenshot below I firstly queried for the presence of deleted files and secondly the volume information:

Example of PowerForensics output

The BytesPerCluster value, for example, is interesting if you discover a deleted file and want to restore it manually from the disk.

Many commands in the module are very useful for my inspection:

Volume Name

Footer analysis of VHD file

Digging deeper into the file-system and specifics, I discovered this explanation of the footer structure of a VHD file. The specification was written by Joachim Metz who I had the privilege to work with and learned a lot from with regards to filesystem forensics. Using the table of the footer and our VHD's footer information, let's see if we can discover more.

Footer of VHD file

The first 8 bytes are the signature (cookie), with the value of ‘conectix’, an indicator this is a vhd file. The next 4 bytes describe which features are enabled. In this case, I observe the value 0x00000002 which translates to “reserved”. The 8 bytes with value 0xFFFFFFFFFFFFFFFF indicate that this is a fixed disk.

The next 4 bytes are the format version followed by 8 bytes of the next offset. The next 4 bytes in this footer have the value of 0x27227A65 — this is the number of seconds since January 1, 2000, and points to the modification time. In this case “656570981 seconds”, which is the modification time of “2020–10–22 04:49:41 UTC”

The next 4 bytes are indicating the ‘Creator Application’ used to create the VHD file. In this case, this is the value “zewin”. I am not currently aware of an application that this value refers to. There are multiple tools that can create VHD files.

Skipping the creator version, I look at the 4 bytes that identify the creator’s operating system. Here it has the value “wi2k” which refers to Microsoft Windows.

The disk-size value in bytes is 12582912 bytes (0x0000000000c00000), aka 12.58 Megabytes.

Flipping a few more bytes I go to the section that indicates the disk type. In this case, the value is 0x00000002, indicating it is a ‘fixed hard disk’.

The last bit I look at is the 16 bytes that contain a big Endian GUID value, in this case,”6D CA 9A 42 A4 6E 55 DA C8 F7 96 DB”.

Summary

Adversaries will always look for new techniques to bypass security controls. Where we as an industry mostly stop our investigations with malicious files, it can be worth digging deeper with a forensics mindset to find more information about the actor and then adding the discovered information to the profile.
