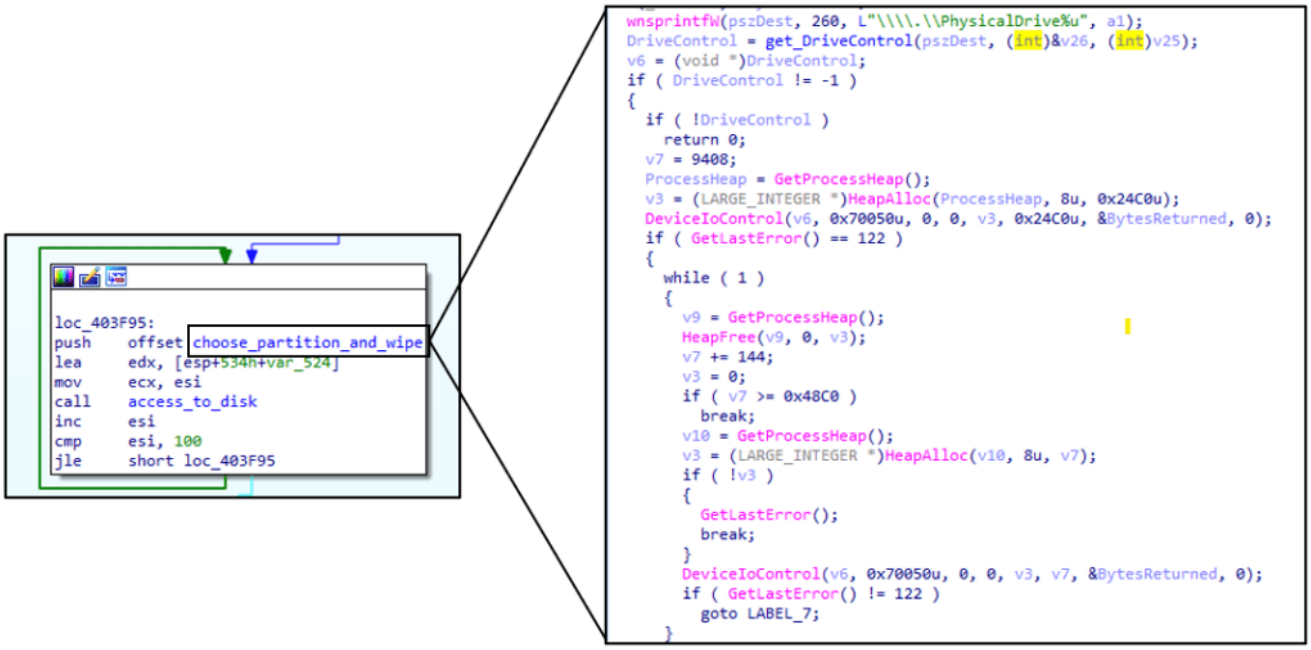
DiskKill/HermeticWiper and NotPetya (Dis)similarities

marcoramilli.com/2022/03/01/diskkill-hermeticwiper-and-notpetya-dissimilarities/

View all posts by marcoramilli

March 1, 2022



Many security researchers, professional cybersecurity analysts and cybsec organizations realized great analyses on DiskKill (HermeticWiper), some of my favorite are HERE, HERE and HERE. Today what I'd like to do, is to focus on specific HermeticWiper characteristics and looking for similarities (or differences) to another similar (and well known) cyber attack happened in Ukraine few years ago: NotPetya. But let's start to highlights some interesting characteristics that DiskKill has implemented.

The Crash Dump Handling

Applications have bugs and eventually they might crash. The Crash Dump capability in Microsoft Windows is quite useful to understand the type of crash and what stimulated (or, in simple cases, caused) the application crash. In this way developers can investigate memory dumps to increase security and stability of the developed system (or application). Starting with Windows Server 2008 and Windows Vista with Service Pack 1 (SP1), Windows Error Reporting (WER) can be configured so that full user-mode dumps are collected and stored locally after a user-mode application crashes. On the other hand crash bumps let to cybersecurity analysts the chance to figure out how the system (application) works or, in specific scenarios, if some vulnerabilities could be exploited to take control over the malicious code. Threat Actors behind DiskKill decided to disable this functionality avoiding to leak such information. As described in Yoroi report (HERE) threat actor behind Hermetic

Wiper modified the following RegKey `HKLM\SYSTEM\CurrentControlSet\Control\CrashControl` in order to disable crash dumps on the target system avoiding memory and stacks investigations.

```
if ( !RegOpenKeyW(HKEY_LOCAL_MACHINE, L"SYSTEM\\CurrentControlSet\\Control\\CrashControl", &phkResult) )
{
  *(_DWORD *)Data = 0;
  RegSetValueExW(phkResult, L"CrashDumpEnabled", 0, 4u, Data, 4u);
  RegCloseKey(phkResult);
}
```

CrashDump disabling

100 Is The Number !

It happens that DiskKill erases “only” the first 100 hard drive available on the target system. So, if you have really important files you can emulate 100 “empty” HD and store your files on the 101 ;). I believe this is another interesting characteristic of this specific wiper. The following image shows the loop to 100.

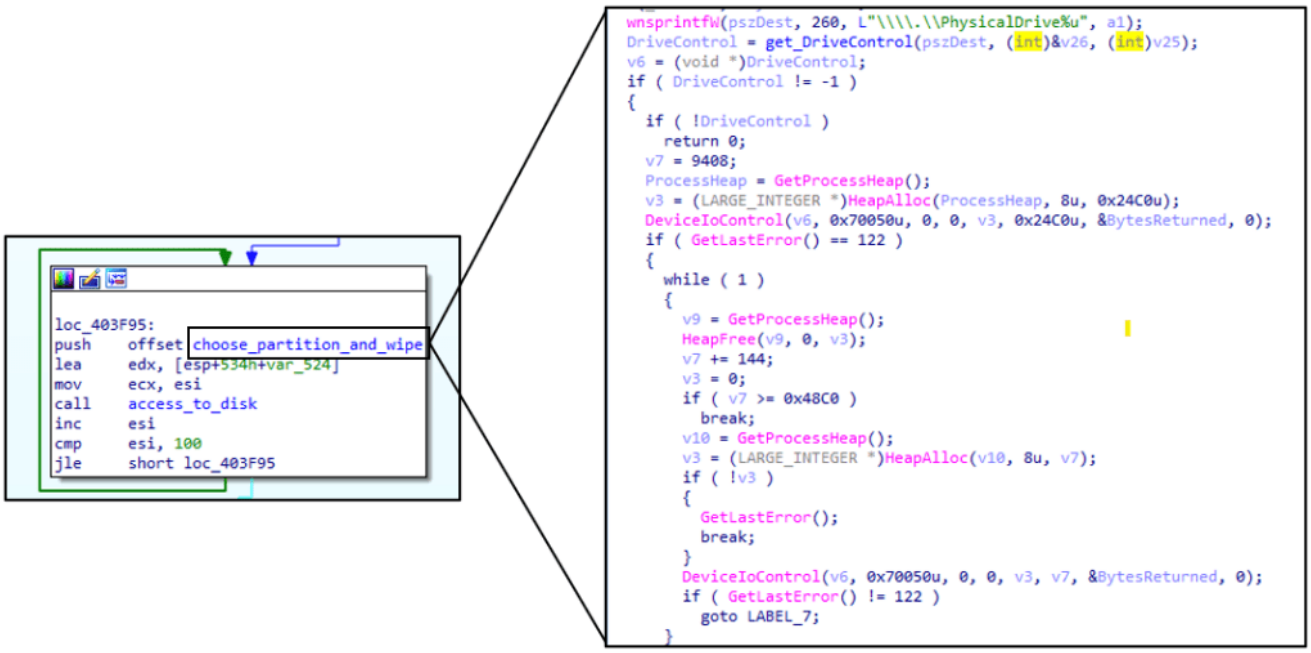


Image from HERE

Living Off The Land Binaries

A quite used techniques from many threat actors is to load legit independent binaries to use their functionalities in order to accomplish their target. In this specific case threat actors behind DiskKill (ab)used a legit driver from EaseUS Partition Master to read and to overwrite specific disk area. I found the usage of that specific `EaseUS` driver (not the use of LOLBas) quite characteristic and specific for this wiper so far.

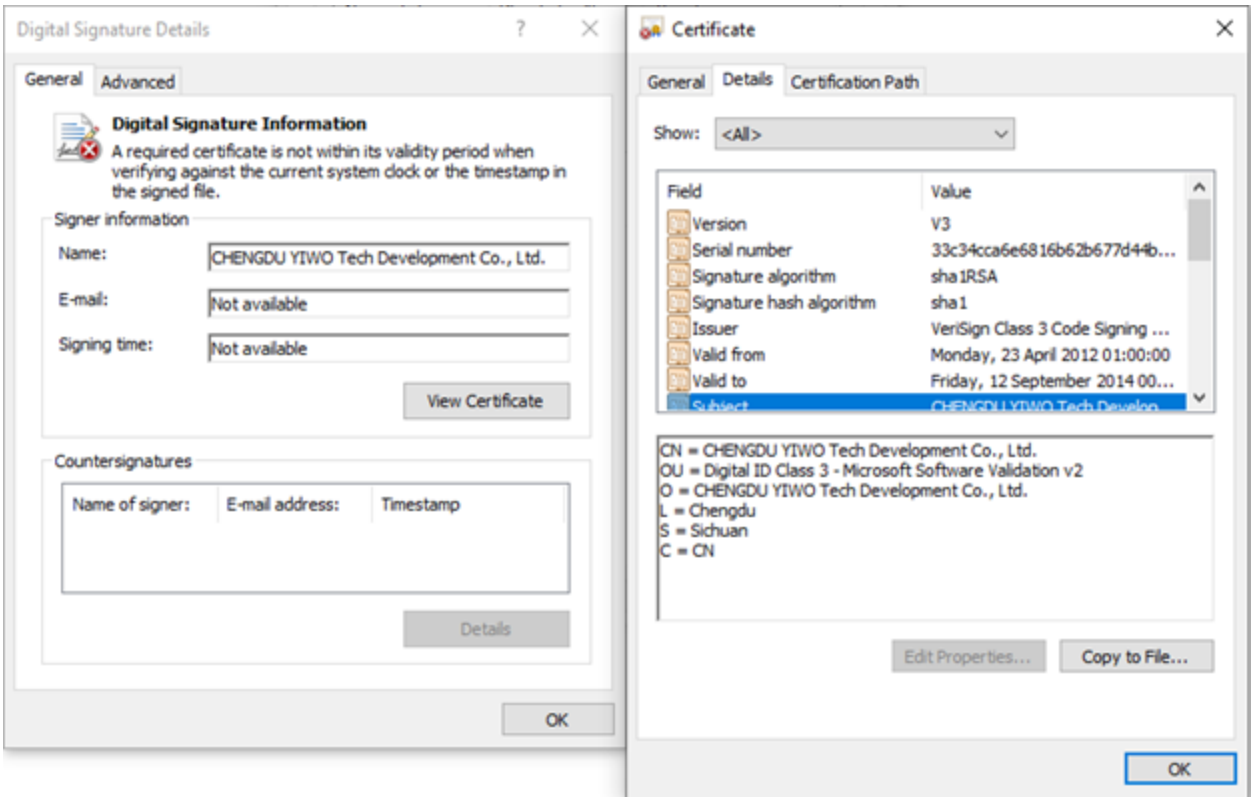


Image From ESET tweets

HermeticWiper VS NotPetya

Before getting into details about similarities between DiskKill and NotPetya, two important cyber attacks against Ukraine, it would be interesting to remind a little bit of history:

“In June 2017, a new variant of Petya was used for a global cyberattack, primarily targeting Ukraine. The new variant propagates via the EternalBlue exploit, which is generally believed to have been developed by the U.S. National Security Agency (NSA), and was used earlier in the year by the WannaCry ransomware. Kaspersky Lab referred to this new version as NotPetya to distinguish it from the 2016 variants, due to these differences in operation” (from Wikipedia).

Both of the Malware have destructive capabilities and both of them targeted the state of Ukraine, but beside that what do they have in common ? Let’s take a closer look by comparing the two malwares in the following table.

Functionality	HermeticWiper / DiskKill	NotPetya
Process Hashes and Process Privilege Checks	NO	Yes. Checks for specific security vendors 1. 0x6403527E 2. 0x23214B44 3. 0x651B3005

Credential Theft	NO	Yes. 1. As an argument to the DLL 2. Communication via a named pipe from the credential theft module
Token Impersonation	NO	Yes. `OpenProcessToken` and `GetTokenInformation` is used to grab the `TokenSessionId` for terminal service sessions.
Malware Propagation (Trojanized)	No Trojanized	Yes. 1. Network node enumeration 2. SMB copy and remote execution 3. SMB exploitation via EternalBlue
Remote Execution	No	Yes. `C:\Windows\dllhost.dat \ -accepteula -s -d C:\Windows\System32\rundll32.exe "C:\Windows\perfc.dat",#1 18 """`
MBR Ransomware	Yes. By looping (to 100) the `"\.\PhysicalDrive"` and overwriting the first 512 bytes of each disk through mounted `LOLBas` Using `"CryptAcquireContextW"` and `"CryptGenRandom"` Windows syscalls.	Yes. By focusing on `\.\PhysicalDrive0` (boot one). It uses `XOR` encoding with key `0x7`. MFT encryption on its own bootloader
Delete ShadowCoipes	Yes. throush `ServicesActive` `vss`	No
NTFS and FAT Specific routines	Yes	No

Multi Thread	Yes. Used to perform quick MBR destruction	Yes. Used for functionalities, communications and propagation

comparative table

Conclusion

To my understanding DiskKill samples are quite different from NotPetya Ransomware. NotPetya looks a more complex and well structured software while HermeticWiper looks like more simple ad “slapdashed” with a single intent: to wipe Master Boot Records. On the other hand DiskKill takes care about wiping speed (multi thread structure) while NotPetya is most interested in lateral movements (through CVEs), C2 communications and modularity. It’s hard to tell if the threat actor is the same or not, but the two families looks like quite different to me.