Challenge: Andromeda Bot Lab

Platform: CyberDefenders

Category: Endpoint Forensics

Difficulty: Medium

Tools Used: MemProcFS, EvtxECmd, Timeline Explorer, VirusTotal

Summary: This lab involves investigating a memory image from a compromised Windows host. Using tools like MemProcFS, EvtxECmd, and Timeline Explorer, it was determined that Andromeda malware was executed from a removeable USB storage device on the host. Further analysis revealed defence evasion techniques by disabling Windows Defender protections and dropped payloads including an executable and multiple DLLs.

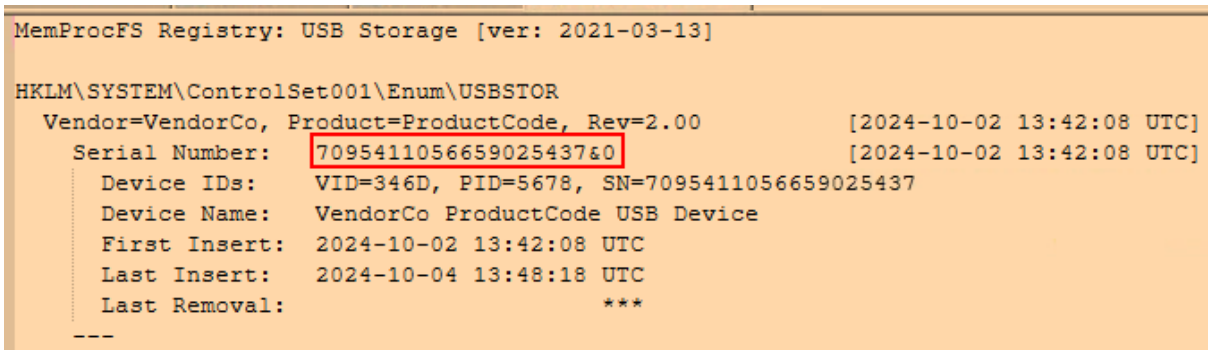
Scenario: As a member of the DFIR team at SecuTech, you're tasked with investigating a security breach affecting multiple endpoints across the organization. Alerts from different systems suggest the breach may have spread via removable devices. You've been provided with a memory image from one of the compromised machines. Your objective is to analyze the memory for signs of malware propagation, trace the infection's source, and identify suspicious activity to assess the full extent of the breach and inform the response strategy.

Tracking the serial number of the USB device is essential for identifying potentially unauthorized devices used in the incident, helping to trace their origin and narrow down your investigation. What is the serial number of the inserted USB device?

Within this lab we are provided with a memory dump of a Windows host. Let's start by parsing the memory dump using an incredible tool called MemProcFS, which enables you to view memory as files in a virtual file system:

- `memprocfs.exe -f "C:\Users\Administrator\Desktop\Start Here\Artifacts\memory.dmp" -forensic 3`

By default, it mounts to the M drive letter. To find USB artifacts, we can navigate to py\reg\usb. Here we can find a text file called usb_storage that contains information about connected usb storage devices:



Answer: 70954110566590254 37&0

Tracking USB device activity is essential for building an incident timeline, providing a starting point for your analysis. When was the last recorded time the USB was inserted into the system?

In the usb_storage.txt file explored previously, we can find the last recorded time the USB was inserted into the system:

```
HKLM\SYSTEM\ControlSet001\Enum\USBSTOR
  Vendor=VendorCo, Product=ProductCode, Rev=2.00          [2024-10-02 13:42:08 UTC]
    Serial Number:   709541105665902543 7&0               [2024-10-02 13:42:08 UTC]
      Device IDs:    VID=346D, PID=5678, SN=709541105665902543 7
      Device Name:   VendorCo ProductCode USB Device
      First Insert:  2024-10-02 13:42:08 UTC
      Last Insert:   2024-10-04 13:48:18 UTC
      Last Removal:                  ***
    ---
```

Alternatively, the last connected timestamp can be found directly in registry located at:

- `M:\registry\HKLM\SYSTEM\ControlSet001\Enum\USBSTOR\Disk&Ven_VendorCo&Prod_ProductCode&Rev_2.00\709541105665902543 7&0\Properties\{83da6326-97a6-4088-9453-a1923f573b29}`

```
ffffe200f728b000:00ab6690
REG_KEY
{83da6326-97a6-4088-9453-a1923f573b29}
2024-10-02 13:45:18 UTC
```

Answer: 2024-10-04 13:48

Identifying the full path of the executable provides crucial evidence for tracing the attack's origin and understanding how the malware was deployed. What is the full path of the executable that was run after the PowerShell commands disabled Windows Defender protections?

Let's start by using EvtxECmd to parse the event logs:

- `.\EvtxECmd.exe -d "M:\misc\eventlog\" --csv . --csvf logs_out.csv`

We can then view the output in Timeline Explorer. Filtering for Sysmon process creation logs (Event ID 1), at 2024-10-04 13:49:48 CMD was observed executing a PowerShell command which disables Windows Defender protections and executes a binary called "Trusted Installer.exe" from the E drive:

Cell contents

```
ParentCommandLine: "C:\Windows\System32\cmd.exe" /c powershell.exe -ExecutionPolicy
Bypass -Command "Set-MpPreference -DisableRealtimeMonitoring $true; Set-MpPreference
-DisableBehaviorMonitoring $true; Set-MpPreference -DisableIOAVProtection $true;
Set-MpPreference -DisableScriptScanning $true; Set-MpPreference -DisableBlockAtFirstSeen
$true; Set-MpPreference -DisableCloudProtection $true; Set-MpPreference
-DisableArchiveScanning $true; Set-MpPreference -SubmitSamplesConsent 2; sc stop
WinDefend; sc config WinDefend start= disabled; sc stop SecurityHealthService; sc config
SecurityHealthService start= disabled; Start-Process 'E:\hidden\Trusted Installer.exe'"
```

If we navigate to:

- `M:\registry\HKLM\SYSTEM\MountedDevices`

We can see that the USB storage device discovered previously, was assigned the drive letter E. Therefore, this binary was located in said storage device:

_DosDevices_E_.txt

```
ffffe200f728b000:00abafd0
REG_BINARY
0000    5f 00 3f 00 3f 00 5f 00  55 00 53 00 42 00 53 00   _.?.?._.U.S.B.S.
0010    54 00 4f 00 52 00 23 00  44 00 69 00 73 00 6b 00   T.O.R.#.D.i.s.k.
0020    26 00 56 00 65 00 6e 00  5f 00 56 00 65 00 6e 00   &.V.e.n._.V.e.n.
0030    64 00 6f 00 72 00 43 00  6f 00 26 00 50 00 72 00   d.o.r.C.o.&.P.r.
0040    6f 00 64 00 5f 00 50 00  72 00 6f 00 64 00 75 00   o.d._.P.r.o.d.u.
0050    63 00 74 00 43 00 6f 00  64 00 65 00 26 00 52 00   c.t.C.o.d.e.&.R.
0060    65 00 76 00 5f 00 32 00  2e 00 30 00 30 00 23 00   e.v._.2...0.0.#.
0070    37 00 30 00 39 00 35 00  34 00 31 00 31 00 30 00   7.0.9.5.4.1.1.0.
0080    35 00 36 00 36 00 35 00  39 00 30 00 32 00 35 00   5.6.6.5.9.0.2.5.
0090    34 00 33 00 37 00 26 00  30 00 23 00 7b 00 35 00   4.3.7.&.0.#.{.5.
00a0    33 00 66 00 35 00 36 00  33 00 30 00 37 00 2d 00   3.f.5.6.3.0.7.-.
00b0    62 00 36 00 62 00 66 00  2d 00 31 00 31 00 64 00   b.6.b.f.-.1.1.d.
00c0    30 00 2d 00 39 00 34 00  66 00 32 00 2d 00 30 00   0.-.9.4.f.2.-.0.
00d0    30 00 61 00 30 00 63 00  39 00 31 00 65 00 66 00   0.a.0.c.9.1.e.f.
00e0    62 00 38 00 62 00 7d 00                            b.8.b.}.
```

Answer: E:\hidden\Trusted Installer.exe

Identifying the bot malware's C&C infrastructure is key for detecting IOCs. According to threat intelligence reports, what URL does the bot use to download its C&C file?

If you take the hash associated with the command discovered previously and submit it to VirusTotal, we can see that it contacted multiple URLs:

Contacted URLs (9)

Scanned	Detections	Status	URL
2023-10-11	5 / 90	200	http://pe.suckmycocklameavindustry.in/cvspcvpfzpfvpcviczvpizzzcvppzsps
2023-10-11	5 / 90	200	http://pe.suckmycocklameavindustry.in/ecacccyyyrtrnlnlhhbxusqookieqfgs
2025-10-22	0 / 98	200	http://www.microsoft.com/pki/certs/MicCodSigPCA_08-31-2010.crt
2025-08-24	8 / 97	200	http://pe.suckmycocklameavindustry.in/efghjklnoprssuvxzbcdeghjkmoqrsst
2025-10-29	11 / 98	200	http://anam0rph.su/in.php
2025-11-18	0 / 98	200	http://crt.sectigo.com/SectigoPublicCodeSigningCAR36.crt
2025-02-09	11 / 96	200	http://xdqzpbcgrvkj.ru/in.php
2025-11-18	0 / 98	200	http://crt.sectigo.com/SectigoPublicCodeSigningRootR46.p7c
2025-10-22	0 / 98	200	http://www.microsoft.com/pki/certs/MicrosoftTimeStampPCA.crt

I cannot find any communications from this host to the discovered domain in Sysmon event logs.

Answer: http://anam0rph.su/in.php

Understanding the IOCs for files dropped by malware is essential for gaining insights into the various stages of the malware and its execution flow. What is the MD5 hash of the dropped .exe file?

If you filter for Sysmon file creation logs (Event ID 11), we can see that “Trusted Installer.exe” created 5 files, including an executable called “Sahofivizu.exe” within the Temp directory of Tomy:

Payload Data3	Payload Data4
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp__PSScriptPolicyTest_p2i40nfy.4ba.ps1
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\Gozekeneka.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\natigezeholi.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\Zojemilocan.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\xuxokuxoka.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\Sahofivizu.exe
Image: C:\Windows\SysWOW64\WerFault.exe	TargetFilename: C:\ProgramData\Microsoft\Windows\WER\Temp\WER1AFB.tmp.dmp

Filtering for process creation logs (Event ID 1) associated with this binary, we can find the MD5 hash:

Payload Data6	Executable Info
ParentCommandLine: "E:\hidden\Trusted Installer.exe"	"C:\Users\Tomy\AppData\Local\Temp\Sahofivizu.exe" "E:\hidden\Trusted Installer.exe"
ParentCommandLine: "C:\Users\Tomy\AppData\Local\Temp\Sahofivizu.exe" "E:\hidden\Trusted Installer.exe…	C:\Windows\SysWOW64\WerFault.exe -u -p 1040 -s 280

Cell contents

MD5=7FE00CC4EA8429629AC0AC610DB51993,

Answer: 7FE00CC4EA8429629AC0AC610DB51993

Having the full file paths allows for a more complete cleanup, ensuring that all malicious components are identified and removed from the impacted locations. What is the full path of the first DLL dropped by the malware sample?

Going back to file creation logs (Event ID 11), we can see that the first dropped DLL was:

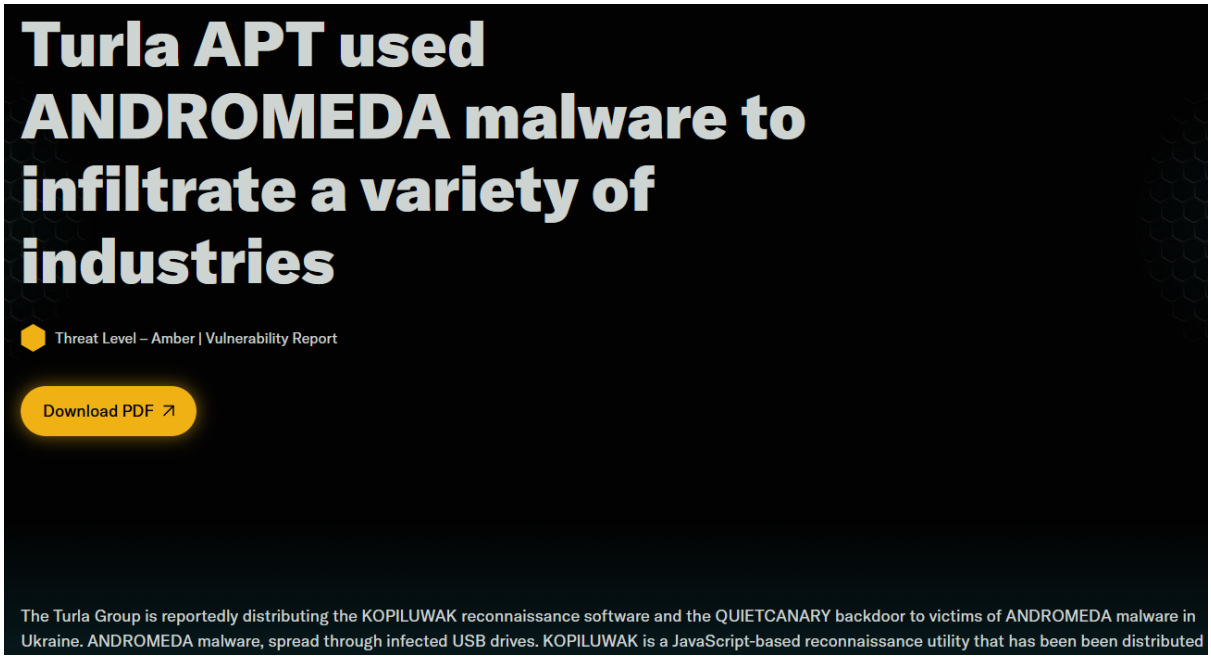
- `C:\Users\Tomy\AppData\Local\Temp\Gozekeneka.dll`

Payload Data3	Payload Data4
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp__PSScriptPolicyTest_p2i40nfy.4ba.ps1
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\Gozekeneka.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\natigezeholi.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\Zojemilocan.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\xuxokuxoka.dll
Image: E:\hidden\Trusted Installer.exe	TargetFilename: C:\Users\Tomy\AppData\Local\Temp\Sahofivizu.exe
Image: C:\Windows\SysWOW64\WerFault.exe	TargetFilename: C:\ProgramData\Microsoft\Windows\WER\Temp\WER1AFB.tmp.dmp

Answer: C:\Users\Tomy\AppData\Local\Temp\Gozekeneka.dll

Connecting malware to APT groups is crucial for uncovering an attack's broader strategy, motivations, and long-term goals. Based on IOCs and threat intelligence reports, which APT group reactivated this malware for use in its campaigns?

If you search the identified IOCs, including dropped DLLs, file hashes, etc, we can determine that this is Andromeda Malware, a trojan first discovered in 2011. After searching “Andromeda Malware APT groups” I came across a post regarding Turla using Andromeda in their campaigns:



Answer: Turla