Writeup :

Challenge Overview – Memory Forensics

Objective

The objective of this challenge is to analyze a Windows memory dump in order to recover sensitive user data belonging to a well-known environmental activist who lost access to his system due to an unknown error.

The investigation focuses on extracting:

- Environment variables
- Browser / password manager artifacts
- KeePass database credentials stored in memory

Q1 – Environment Variable Analysis

Investigation Context

During the memory analysis, it was noted that the victim frequently uses:

- Web browsers
- Password managers (KeePass)

Given the hint that the victim is an **environmental activist**, environment variables may contain **user-specific or suspicious values** that provide an initial clue.

Tool Used

- **Volatility 3 Framework (CLI , GUI)**
- **Windows memory dump (**`.raw`**)**

---

1) Initial Analysis – Process Enumeration

First Step in the Analysis

The first step in the memory forensic analysis was to enumerate all running processes in the memory dump.
This step is critical to understand what applications were active on the victim's system at the time the memory was captured.

```
"C:\Users\NVIDIA PLUS\Desktop\vol.exe" -f "C:\Users\NVIDIA PLUS\Desktop\New folder\MemoryDump_Lab2.raw" windows.pslist.PsList


Please wait, this may take a few minutes.
Volatility 3 Framework 2.26.2
PID     PPID    ImageFileName   Offset(V)       Threads Handles SessionId       Wow64   CreateTime      ExitTime        File output
4       0       System  0xfa8000ca0040  80      541     N/A     False   2019-12-14 10:35:21.000000 UTC  N/A     Disabled
248     4       smss.exe        0xfa80014976c0  3       37      N/A     False   2019-12-14 10:35:21.000000 UTC  N/A     Disabled
320     312     csrss.exe       0xfa80014fdb30  10      446     0       False   2019-12-14 10:35:27.000000 UTC  N/A     Disabled
368     360     csrss.exe       0xfa8001c40060  8       237     1       False   2019-12-14 10:35:28.000000 UTC  N/A     Disabled
376     248     psxss.exe       0xfa8000ca8840  18      786     0       False   2019-12-14 10:35:28.000000 UTC  N/A     Disabled
416     360     winlogon.exe    0xfa8001c5a700  6       112     1       False   2019-12-14 10:35:30.000000 UTC  N/A     Disabled
424     312     wininit.exe     0xfa8001c5b2b0  3       75      0       False   2019-12-14 10:35:30.000000 UTC  N/A     Disabled
484     424     services.exe    0xfa8001c95320  8       206     0       False   2019-12-14 10:35:31.000000 UTC  N/A     Disabled
492     424     lsass.exe       0xfa8001c9d910  8       546     0       False   2019-12-14 10:35:31.000000 UTC  N/A     Disabled
500     424     lsm.exe 0xfa8001c9e2d0  10      181     0       False   2019-12-14 10:35:31.000000 UTC  N/A     Disabled
588     484     svchost.exe     0xfa8001cec790  12      354     0       False   2019-12-14 10:35:35.000000 UTC  N/A     Disabled
652     484     VBoxService.ex  0xfa8001d13060  14      135     0       False   2019-12-14 10:35:36.000000 UTC  N/A     Disabled
720     484     svchost.exe     0xfa8001d4ab30  7       275     0       False   2019-12-14 10:35:37.000000 UTC  N/A     Disabled
812     484     svchost.exe     0xfa8001d76320  21      474     0       False   2019-12-14 10:35:38.000000 UTC  N/A     Disabled
```

2) Command Line Analysis

identifying the suspicious processes, the next step was to analyze how these processes were launched and whether any sensitive information was passed to them at runtime.

Volatility Plugin Used : `windows.cmdline.CmdLine`

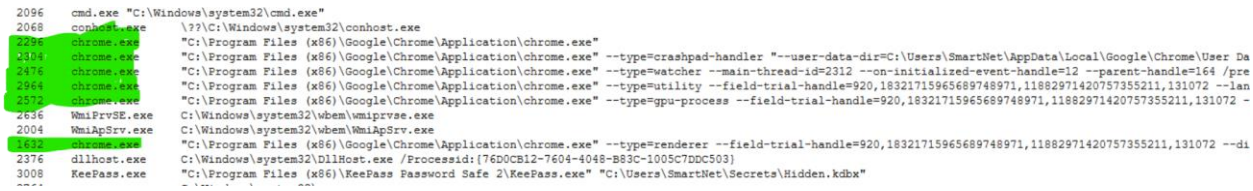
```
Time Stamp: Thu Dec 25 20:27:16 2025

"C:\Users\NVIDIA PLUS\Desktop\vol.exe" -f "C:\Users\NVIDIA PLUS\Desktop\New folder\MemoryDump_Lab2.raw" windows.cmdline.CmdLine


Please wait, this may take a few minutes.
Volatility 3 Framework 2.26.2
PID     Process Args
4       System  -
248     smss.exe        \SystemRoot\System32\smss.exe
320     csrss.exe       %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Windows=On SubSystemType=Windows ServerDll=ba
368     csrss.exe       %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Windows=On SubSystemType=Windows ServerDll=ba
376     psxss.exe       %SystemRoot%\system32\psxss.exe
416     winlogon.exe    winlogon.exe
424     wininit.exe     wininit.exe
484     services.exe    C:\Windows\system32\services.exe
492     lsass.exe       C:\Windows\system32\lsass.exe
```

Suspicious Processes Identified

The results revealed three suspicious processes that were of high forensic value. These processes were directly related to the challenge context (browsing activity, password management, and user data handling):

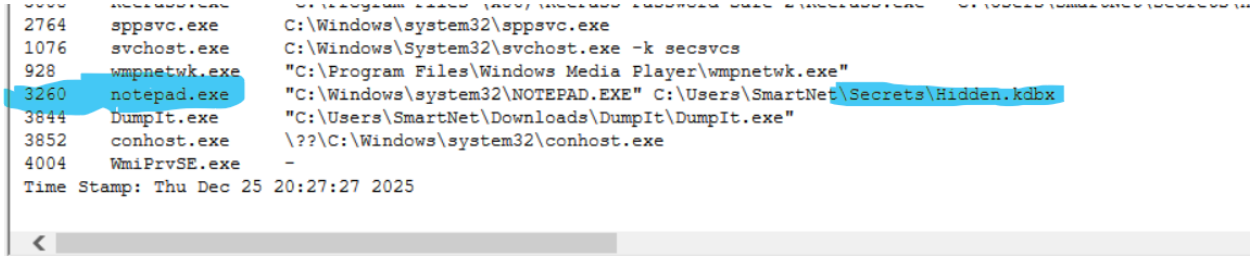
Google Chrome

- **Process Name:** `chrome.exe`
- **PID:** (2296 , 2304 , 2479 , 2964 , 2575)
- **Reason for Suspicion:**
    - Active web browser



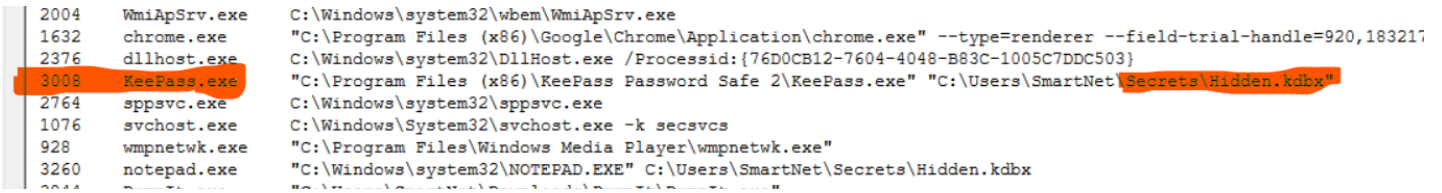
Notepad

- **Process Name:** `notepad.exe`
- **PID:** (3260)
- **Reason for Suspicion:**
    - Often used to temporarily store passwords, notes, or copied text



KeePass

- **Process Name:** `KeePass.exe`
- **PID: 3008**
- **Reason for Suspicion:**
    - Password manager
    - High-value target in memory for:
        - Database paths



3- Environment Variable Analysis

Since the challenge explicitly mentions that the victim is an “environmental activist”, this strongly suggests that environment variables may contain relevant clues or hidden information.

The `windows.envars.Envars` plugin was used because it directly aligns with the challenge hint (“environmental activist”) and provided critical insight into sensitive data stored in environment variables, leading to the next phase of the investigation.

```
"C:\Users\NVIDIA PLUS\Desktop\vol.exe" -f "C:\Users\NVIDIA PLUS\Desktop\New folder\MemoryDump_Lab2.raw" windows.envars.Envars

Please wait, this may take a few minutes.
Volatility 3 Framework 2.26.2
PID     Process Block   Variable        Value
248     smss.exe        0x2d1430        Path    C:\Windows\System32
248     smss.exe        0x2d1430        SystemDrive     C:
248     smss.exe        0x2d1430        SystemRoot      C:\Windows
320     csrss.exe       0x481970        ComSpec C:\Windows\system32\cmd.exe
320     csrss.exe       0x481970        FP_NO_HOST_CHECK        NO
320     csrss.exe       0x481970        NEW_TMP C:\Windows\ZmxhZ3t3M2xjMG0zX1QwXyRUNGczXyFfT2ZfTDRCXzJ9
320     csrss.exe       0x481970        NUMBER_OF_PROCESSORS    1
320     csrss.exe       0x481970        OS      Windows_NT
```

4- Encoded Data Discovery & Flag Extraction

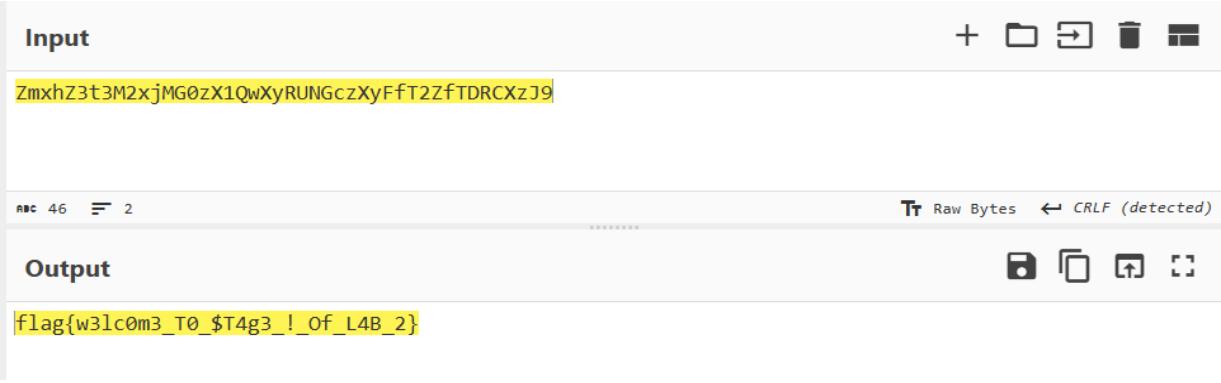
During further memory analysis, multiple Base64-encoded strings were identified across different memory regions and file paths.
The repetition of the same encoded content strongly indicated that it was intentionally stored**,** rather than being random memory noise.

- **Notepad Memory Dump Analysis**

Given that Notepad is commonly used to temporarily store sensitive information in plain text, its memory was dumped and analyzed.

- **Decoding Process**

The extracted Base64 string was then analyzed using CyberChef, a well-known forensic and decoding tool



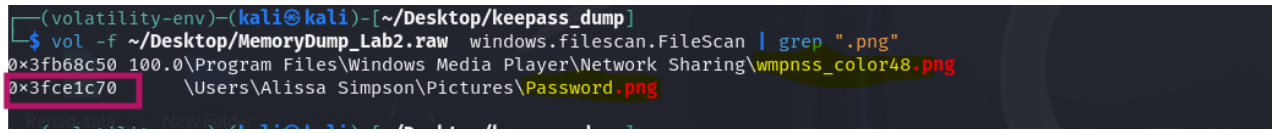
Q2- Overview – KeePass Database Recovery

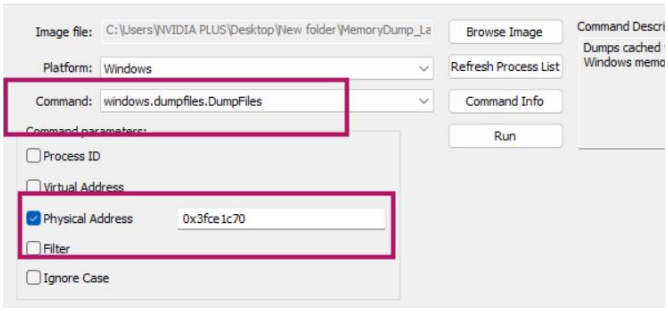
This task focuses on memory forensics to recover sensitive user data from a system memory dump.

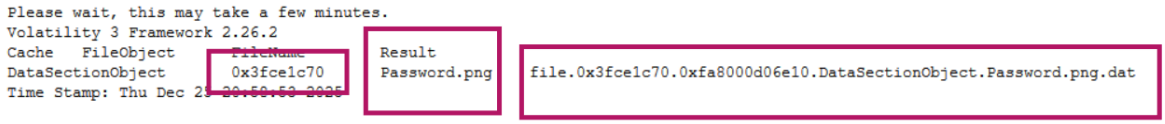
During the investigation, it was discovered that the victim relies heavily on:

- Web browsers
- Password managers (specifically KeePass)
- Simple applications for storing information (Notepad)

While analyzing the memory dump, a reference to an image file named **`Password.png`** was found.
This image contained a **password**, which was crucial because it was used as the **master password** to unlock a **KeePass database** that also existed in memor



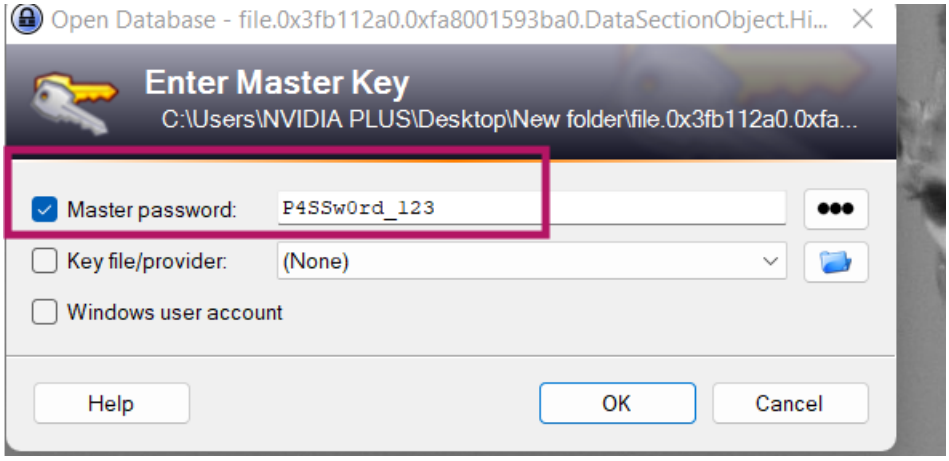


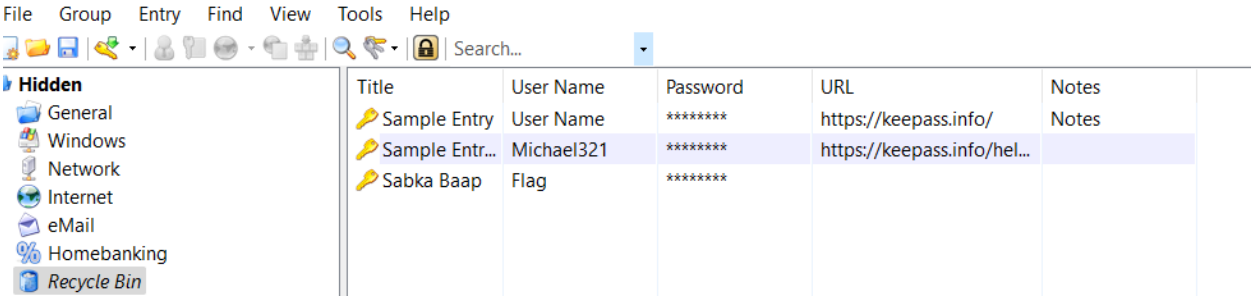


Using the Recovered Password to Unlock KeePass

After dumping the image file **Password.png** from memory and analyzing its contents, the password was successfully recovered.

The next step was to use this password as the **master password** for the **KeePass database** that was also found in the memory dump.





Right click on username “flag”