LetsDefend: Memory Analysis

The following writeup covers the Memory Analysis room hosted on LetsDefend. This room is entirely concerned with memory forensics using volatility 2 or 3.

Scenario: A Windows Endpoint was recently compromised. Thanks to our cutting-edge EDR/IDS solution we immediately noticed it. The alert was escalated to Tier 2 (Incident Responders) for further investigation. As our Forensics guy, you were given the memory dump of the compromised host. You should continue to investigate.

What was the date and time when Memory from the compromised endpoint was acquired?

We can use the windows.info.Info plugin to determine when the memory dump was acquired:

```
vol -f dump.mem windows.info.Info
```

```
Kernel Base     0xf80047ea7000
DTB     0x1aa000
Symbols file:///usr/local/lib/python3.8/dist-pack
-1.json.xz
Is64Bit True
IsPAE   False
layer_name      0 WindowsIntel32e
memory_layer    1 FileLayer
KdVersionBlock  0xf800482a9dc0
Major/Minor     15.17763
MachineType     34404
KeNumberProcessors      2
SystemTime      2022-07-26 18:16:32
NtSystemRoot    C:\Windows
NtProductType   NtProductWinNt
NtMajorVersion  10
NtMinorVersion  0
PE MajorOperatingSystemVersion  10
PE MinorOperatingSystemVersion  0
PE Machine      34404
PE TimeDateStamp        Thu Oct 28 12:04:50 2060
```

2022-07-26 18:16:32

What was the suspicious process running on the system? (Format: name.extension)

I started off by using the windows.pslist plugin to list all running processes:

```
vol -f dump.mem windows.pslist
```

Nothing immediately stood out, so I started to look for possible network connections using the netscan plugin but found nothing. Therefore I went back to the process list and looked for

anything imitating legitimate Windows processes. After scrolling for a while, I see lsass which in and of itself isnt suspicious:

```
lsass.exe       0xdf0e86d86580  14      -       1       False   2022-07-26 18:09:33.000000
```

However, if we use the pstree plugin for more information, we can see that its running in the wrong session, its in the wrong location, and its parent process is explorer.exe:

```
vol -f dump.mem windows.pstree | grep "lsass"
```

```
* 640ess500100.0lsass.exe       0xdf0e8394f080  7       -       0       False   2022-07-26 18:01:15.000000      N/A     \Device
\HarddiskVolume1\Windows\System32\lsass.exe     C:\Windows\system32\lsass.exe   C:\Windows\system32\lsass.exe
*** 7592        3996    lsass.exe       0xdf0e86d86580  14      -       1       False   2022-07-26 18:09:33.000000      N/A   \
Device\HarddiskVolume1\Windows\System\lsass.exe "C:\Windows\System\lsass.exe"   C:\Windows\System\lsass.exe
```

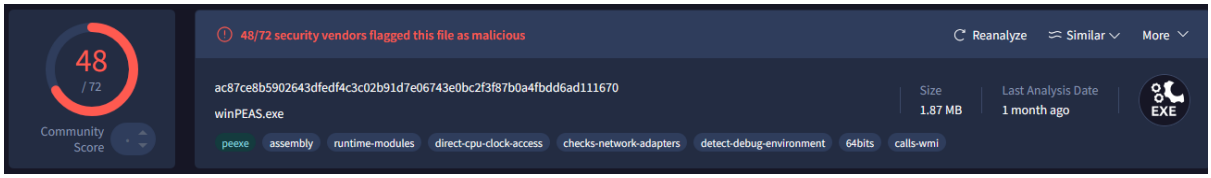
Analyse and find the malicious tool running on the system by the attacker (Format name.extension)

We already know that lsass.exe is very suspicious (PID 7952). So let's dump this process, hash it, and search it on VirusTotal:

```
vol -f dump.mem windows.pslist --pid 7592 --dump
```

```
sha256sum 7592.lsass.exe.0x2238edc0000.dmp
```

```
ac87ce8b5902643dfedf4c3c02b91d7e06743e0bc2f3f87b0a4fbdd6ad111670  7592.lsass.exe.0x2238edc0000.dmp
```



As we can see, the malicious tool is winPEAS.exe, aka a tool that looks for privilege escalation vectors in Windows environments.

Which User Account was compromised? Format (DomainName/USERNAME)

We can use the windows.getsids plugin to determine the user account that was compromised:

```
vol -f dump.mem windows.getsids | grep 7952
```

```
7952    MicrosoftEdgeC  S-1-5-21-321011808-3761883066-353627080-1003    CyberJunkie
7952    MicrosoftEdgeC  S-1-5-21-321011808-3761883066-353627080-513     Domain Users
7952    MicrosoftEdgeC  S-1-1-0 Everyone
7952    MicrosoftEdgeC  S-1-5-114       Local Account (Member of Administrators)
7952    MicrosoftEdgeC  S-1-5-32-544    Administrators
7952    MicrosoftEdgeC  S-1-5-32-545    Users
7952    MicrosoftEdgeC  S-1-5-4 Interactive
7952    MicrosoftEdgeC  S-1-2-1 Console Logon (Users who are logged onto the physical console)
7952    MicrosoftEdgeC  S-1-5-11        Authenticated Users
7952    MicrosoftEdgeC  S-1-5-15        This Organization
7952    MicrosoftEdgeC  S-1-5-113       Local Account
7952    MicrosoftEdgeC  S-1-5-5-0-299342        Logon Session
7952    MicrosoftEdgeC  S-1-2-0 Local (Users with the ability to log in locally)
7952    MicrosoftEdgeC  S-1-5-64-10     NTLM Authentication
7952    MicrosoftEdgeC  S-1-16-4096     Low Mandatory Level
```

We can see that the user account is CyberJunkie. We now need to use the envars plugin to display a processes environment variables:

```
vol -f dump.mem windows.envars.Envars | grep 7592
```

```
7592    lsass.exe       0x2238f203590   PUBLIC  C:\Users\Public
7592    lsass.exe       0x2238f203590   SESSIONNAME     Console
7592    lsass.exe       0x2238f203590   SystemDrive     C:
7592    lsass.exe       0x2238f203590   SystemRoot      C:\Windows
7592    lsass.exe       0x2238f203590   TEMP    C:\Users\CYBERJ~1\AppData\Local\Temp
7592    lsass.exe       0x2238f203590   TMP     C:\Users\CYBERJ~1\AppData\Local\Temp
7592    lsass.exe       0x2238f203590   USERDOMAIN      MSEDGEWIN10
7592    lsass.exe       0x2238f203590   USERDOMAIN_ROAMINGPROFILE       MSEDGEWIN10
7592    lsass.exe       0x2238f203590   USERNAME        CyberJunkie
7592    lsass.exe       0x2238f203590   USERPROFILE     C:\Users\CyberJunkie
7592    lsass.exe       0x2238f203590   windir  C:\Windows
```

The domain is MSEDGEWIN10. Therefore, the answer is MSEDGEWIN10/CyberJunkie.

What is the compromised user password?

To find the users password, use the windows.hashdump plugin like as follows:

```
vol -f dump.mem windows.hashdump
```

```
User    rid     lmhash  nthash

Administrator   500     aad3b435b51404eeaad3b435b51404ee        fc525c9683e8fe067095ba2ddc971889
Guest   501     aad3b435b51404eeaad3b435b51404ee        31d6cfe0d16ae931b73c59d7e0c089c0
DefaultAccount  503     aad3b435b51404eeaad3b435b51404ee        31d6cfe0d16ae931b73c59d7e0c089c0
WDAGUtilityAccount      504     aad3b435b51404eeaad3b435b51404ee        20ff0389f84bdbf9ce6fc36af6993b63
IEUser  1000    aad3b435b51404eeaad3b435b51404ee        fc525c9683e8fe067095ba2ddc971889
sshd    1002    aad3b435b51404eeaad3b435b51404ee        42760776cade85fd98103a0f44437800
CyberJunkie     1003    aad3b435b51404eeaad3b435b51404ee        a9fdfa038c4b75ebc76dc855dd74f0da
```

Copy the nthash and enter it into a tool like Crackstation (or use john the ripper or hashcat):

This was a really fun room and happens to be my first experience using LetsDefend. This was also the first time I have come across a challenge where there is a malicious process masquerading as a legitimate Windows process. This of course is a good thing as its commonly seen in the wild (unlikely to see evil.exe in the process list).