University of Victoria

Department of Electrical and Computer Engineering

"Master of Engineering in Telecommunications & Information Security (MTIS)"

ECE 570 – "Computer Forensics Methodologies"

➢ Project Report: Investigating an Infected Machine

• **Name:** Nelson Gomez	• **Name:** Gabriel Naranjo
• **Student No.** V01072284	• **Student No.** V01052227
• **Email:** nelsongb@uvic.ca	• **Email:** gabrielnaranjoor@uvic.ca

June 09, 2025

- **TASK**
  Provide a report analyzing suspicious activities by answering the following questions:

1. Identify running processes, and determine which ones look suspicious and justify why [2.5%].

Solution:
Tools used: Volatility v2.6 and Bash shell.

1.1 Identifying the memory profile and exporting environment variables

To begin the forensic analysis of the memory dump, we first need to determine the correct operating system profile using the 'imageinfo' plugin.

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ vol.py -f ECE570-project-1-memory.dmp imageinfo
Volatility Foundation Volatility Framework 2.6
INFO    : volatility.debug    : Determining profile based on KDBG search...
          Suggested Profile(s) : Win7SP1x64, Win7SP0x64, Win2008R2SP0x64, Win2008R2SP1x64_23418, Win2008R2SP1x64, Win7SP1x64_23418
                     AS Layer1 : WindowsAMD64PagedMemory (Kernel AS)
                     AS Layer2 : VirtualBoxCoreDumpElf64 (Unnamed AS)
                     AS Layer3 : FileAddressSpace (/home/kali/Project_1_570/ECE570-project-1-memory/ECE570-project-1-memory.dmp)
                      PAE type : No PAE
                           DTB : 0x187000L
                          KDBG : 0xf80002a49070L
          Number of Processors : 1
     Image Type (Service Pack) : 0
                KPCR for CPU 0 : 0xfffff80002a4ad00L
             KUSER_SHARED_DATA : 0xfffff78000000000L
           Image date and time : 2018-04-13 01:49:35 UTC+0000
     Image local date and time : 2018-04-12 18:49:35 -0700
```

In this case, the `imageinfo` output shows that the Service Pack is 0, which indicates that the appropriate profile among the suggested ones is `Win7SP0x64`

Once the correct profile is identified, we export it as an environment variable along with the path to the memory image. This simplifies all subsequent Volatility commands by avoiding repetitive arguments.

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ export VOLATILITY_LOCATION=file:///home/kali/Project_1_570/ECE570-project-1-memory/ECE570-project-1-memory.dmp

┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ export VOLATILITY_PROFILE=Win7SP0x64
```

1.2 Listing running processes and identifying suspicious ones.

To identify potentially malicious processes, we used the pslist plugin in Volatility to visualize all active processes from the memory image, including names, PIDs, and parent PIDs.

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ vol.py pslist
Volatility Foundation Volatility Framework 2.6
Offset(V)          Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start                          Exit
------------------ -------------------- ------ ------ ------ -------- ------ ------ ------------------------------ ------------------------------
0xfffffa8000cb8040 System                    4      0     84      528 ------      0 2018-04-06 23:12:36 UTC+0000
0xfffffa8001d185f0 smss.exe                276      4      2       29 ------      0 2018-04-06 23:12:36 UTC+0000
0xfffffa80024e2800 csrss.exe               360    352      9      402      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa8002340060 wininit.exe             392    352      3       75      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa8001c9a4e0 csrss.exe               404    384      8      259      1      0 2018-04-06 23:12:43 UTC+0000
0xfffffa8002519060 winlogon.exe            444    384      3      108      1      0 2018-04-06 23:12:43 UTC+0000
0xfffffa800252c460 services.exe            488    392     11      207      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa800255ab30 lsass.exe               496    392      7      553      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa8002541b30 lsm.exe                 504    392     10      144      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa80025ae600 svchost.exe             612    488     10      345      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa80025d96b0 VBoxService.ex          672    488     12      114      0      0 2018-04-06 23:12:43 UTC+0000
0xfffffa80025ef060 svchost.exe             736    488      9      264      0      0 2018-04-06 23:12:44 UTC+0000
0xfffffa8002570b30 svchost.exe             824    488     18      457      0      0 2018-04-06 23:12:44 UTC+0000
0xfffffa80027d2740 svchost.exe             872    488     19      478      0      0 2018-04-06 23:12:44 UTC+0000
0xfffffa8002644b30 svchost.exe             904    488     39     1238      0      0 2018-04-06 23:12:44 UTC+0000
0xfffffa800285d4a0 svchost.exe             332    488     12      325      0      0 2018-04-06 23:12:45 UTC+0000
0xfffffa800288e920 svchost.exe             900    488     15      395      0      0 2018-04-06 23:12:46 UTC+0000
0xfffffa8002897b30 spoolsv.exe            1120    488     12      279      0      0 2018-04-06 23:12:47 UTC+0000
0xfffffa8002930060 svchost.exe            1152    488     17      319      0      0 2018-04-06 23:12:47 UTC+0000
0xfffffa80029c5b30 svchost.exe            1296    488     13      220      0      0 2018-04-06 23:12:48 UTC+0000
0xfffffa80029eab30 dwm.exe                1588    872      3       70      1      0 2018-04-06 23:12:50 UTC+0000
0xfffffa8002aa4b30 taskhost.exe           1596    488      7      141      1      0 2018-04-06 23:12:50 UTC+0000
0xfffffa8002ab6b30 explorer.exe           1652   1580     23      817      1      0 2018-04-06 23:12:51 UTC+0000
0xfffffa8002b53b30 svchost.exe            1948    488      5       96      0      0 2018-04-06 23:12:53 UTC+0000
0xfffffa8002afeb30 VBoxTray.exe           1376   1652     10      103      1      0 2018-04-06 23:12:56 UTC+0000
0xfffffa8002618060 python.exe             1208   1652      1       92      1      0 2018-04-06 23:12:59 UTC+0000
0xfffffa800288a5e0 conhost.exe            1916    404      2       53      1      0 2018-04-06 23:12:59 UTC+0000
0xfffffa8002cd3b30 acrotray.exe           2144   1484      2       60      1      1 2018-04-06 23:13:08 UTC+0000
0xfffffa80015e1b30 SearchIndexer.         2180    488     12      765      0      0 2018-04-06 23:13:10 UTC+0000
0xfffffa8002daab30 wmpnetwk.exe           2540    488     10      214      0      0 2018-04-06 23:13:22 UTC+0000
0xfffffa8002e82520 SndVol.exe             2568   1652      0 ------      1      0 2018-04-06 23:13:23 UTC+0000
0xfffffa8002a83950 python.exe             2176   1208     16      233      1      0 2018-04-13 01:28:34 UTC+0000
0xfffffa8000e9f060 7004af389d633b          512   2156      0 ------      1      0 2018-04-13 01:28:25 UTC+0000
0xfffffa8000e70060 mscorsvw.exe           2740    488      8      124      0      1 2018-04-13 01:28:34 UTC+0000
0xfffffa8000e2fb30 mscorsvw.exe           3056    488      7       86      0      0 2018-04-13 01:28:40 UTC+0000
0xfffffa8000eb4b30 svchost.exe            1404    488     13      355      0      0 2018-04-13 01:28:45 UTC+0000
0xfffffa8000e26790 7004af389d633b         1400    512      0 ------      1      0 2018-04-13 01:29:07 UTC+0000
0xfffffa8000e2cb30 aifkydk.exe            2652   1400      0 ------      1      0 2018-04-13 01:29:11 UTC+0000
0xfffffa8000efa480 cmd.exe                2920   1400      0 ------      1      0 2018-04-13 01:29:15 UTC+0000
0xfffffa8000edeb30 aifkydk.exe            1728   2652      0 ------      1      0 2018-04-13 01:29:54 UTC+0000
0xfffffa8002551550 bcdedit.exe             812   1728      0 ------      1      0 2018-04-13 01:30:08 UTC+0000
0xfffffa8002afe060 vssadmin.exe            580   1728      0 ------      1      0 2018-04-13 01:30:09 UTC+0000
0xfffffa8000e2bb30 bcdedit.exe            1752   1728      0 ------      1      0 2018-04-13 01:30:10 UTC+0000
0xfffffa8000dbfb30 bcdedit.exe            2208   1728      0 ------      1      0 2018-04-13 01:30:11 UTC+0000
0xfffffa8000f22920 bcdedit.exe            2768   1728      0 ------      1      0 2018-04-13 01:30:13 UTC+0000
0xfffffa8000f5c6a0 bcdedit.exe            2852   1728      0 ------      1      0 2018-04-13 01:30:14 UTC+0000
0xfffffa8000fe8930 notepad.exe             364   1728      1       68      1      1 2018-04-13 01:37:40 UTC+0000
0xfffffa800124e920 iexplore.exe           2800   1728     13      547      1      1 2018-04-13 01:37:41 UTC+0000
0xfffffa8000dbd250 dllhost.exe            1932    612      7      205      1      1 2018-04-13 01:37:46 UTC+0000
0xfffffa80011c9060 vssadmin.exe            400   1728      0 ------      1      0 2018-04-13 01:37:46 UTC+0000
0xfffffa8001153240 iexplore.exe           2120   2800      6      351      1      1 2018-04-13 01:37:47 UTC+0000
0xfffffa8000f22060 cmd.exe                1080   1728      0 ------      1      0 2018-04-13 01:37:57 UTC+0000
0xfffffa8002de6b30 VSSVC.exe              1564    488      9      184      0      0 2018-04-13 01:44:43 UTC+0000
0xfffffa8001050650 svchost.exe            2312    488      4       78      0      0 2018-04-13 01:44:47 UTC+0000
0xfffffa8000e39b30 mscorsvw.exe           2728   2740      8      145      0      1 2018-04-13 01:49:17 UTC+0000
```

To complement the pslist output, we used the pstree plugin to visualize parent-child relationships between processes and identify suspicious process origins.

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ vol.py pstree
Volatility Foundation Volatility Framework 2.6
Name                                                  Pid   PPid   Thds   Hnds Time
-------------------------------------------------- ------ ------ ------ ------ ----
 0xfffffa8002340060:wininit.exe                       392    352      3     75 2018-04-06 23:12:43 UTC+0000
. 0xfffffa800252c460:services.exe                     488    392     11    207 2018-04-06 23:12:43 UTC+0000
.. 0xfffffa8002930060:svchost.exe                    1152    488     17    319 2018-04-06 23:12:47 UTC+0000
.. 0xfffffa80029c5b30:svchost.exe                    1296    488     13    220 2018-04-06 23:12:48 UTC+0000
.. 0xfffffa800288e920:svchost.exe                     900    488     15    395 2018-04-06 23:12:46 UTC+0000
.. 0xfffffa8002b53b30:svchost.exe                    1948    488      5     96 2018-04-06 23:12:53 UTC+0000
.. 0xfffffa8000eb4b30:svchost.exe                    1404    488     13    355 2018-04-13 01:28:45 UTC+0000
.. 0xfffffa80025d96b0:VBoxService.ex                  672    488     12    114 2018-04-06 23:12:43 UTC+0000
.. 0xfffffa8000e2fb30:mscorsvw.exe                   3056    488      7     86 2018-04-13 01:28:40 UTC+0000
.. 0xfffffa8002de6b30:VSSVC.exe                      1564    488      9    184 2018-04-13 01:44:43 UTC+0000
.. 0xfffffa8002644b30:svchost.exe                     904    488     39   1238 2018-04-06 23:12:44 UTC+0000
.. 0xfffffa8002570b30:svchost.exe                     824    488     18    457 2018-04-06 23:12:44 UTC+0000
.. 0xfffffa8000e70060:mscorsvw.exe                   2740    488      8    124 2018-04-13 01:28:34 UTC+0000
... 0xfffffa8000e39b30:mscorsvw.exe                  2728   2740      8    145 2018-04-13 01:49:17 UTC+0000
.. 0xfffffa8001050650:svchost.exe                    2312    488      4     78 2018-04-13 01:44:47 UTC+0000
.. 0xfffffa8002897b30:spoolsv.exe                    1120    488     12    279 2018-04-06 23:12:47 UTC+0000
.. 0xfffffa80015e1b30:SearchIndexer.                 2180    488     12    765 2018-04-06 23:13:10 UTC+0000
.. 0xfffffa800285d4a0:svchost.exe                     332    488     12    325 2018-04-06 23:12:45 UTC+0000
.. 0xfffffa80027d2740:svchost.exe                     872    488     19    478 2018-04-06 23:12:44 UTC+0000
... 0xfffffa80029eab30:dwm.exe                       1588    872      3     70 2018-04-06 23:12:50 UTC+0000
.. 0xfffffa80025ef060:svchost.exe                     736    488      9    264 2018-04-06 23:12:44 UTC+0000
.. 0xfffffa80025ae600:svchost.exe                     612    488     10    345 2018-04-06 23:12:43 UTC+0000
... 0xfffffa8000dbd250:dllhost.exe                   1932    612      7    205 2018-04-13 01:37:46 UTC+0000
.. 0xfffffa8002aa4b30:taskhost.exe                   1596    488      7    141 2018-04-06 23:12:50 UTC+0000
.. 0xfffffa8002daab30:wmpnetwk.exe                   2540    488     10    214 2018-04-06 23:13:22 UTC+0000
. 0xfffffa800255ab30:lsass.exe                        496    392      7    553 2018-04-06 23:12:43 UTC+0000
. 0xfffffa8002541b30:lsm.exe                          504    392     10    144 2018-04-06 23:12:43 UTC+0000
 0xfffffa80024e2800:csrss.exe                         360    352      9    402 2018-04-06 23:12:43 UTC+0000
 0xfffffa8002ab6b30:explorer.exe                     1652   1580     23    817 2018-04-06 23:12:51 UTC+0000
. 0xfffffa8002afeb30:VBoxTray.exe                    1376   1652     10    103 2018-04-06 23:12:56 UTC+0000
. 0xfffffa8002618060:python.exe                      1208   1652      1     92 2018-04-06 23:12:59 UTC+0000
.. 0xfffffa8002a83950:python.exe                     2176   1208     16    233 2018-04-13 01:28:34 UTC+0000
. 0xfffffa8002e82520:SndVol.exe                      2568   1652      0 ------ 2018-04-06 23:13:23 UTC+0000
 0xfffffa8000cb8040:System                               4      0     84    528 2018-04-06 23:12:36 UTC+0000
. 0xfffffa8001d185f0:smss.exe                         276      4      2     29 2018-04-06 23:12:36 UTC+0000
 0xfffffa8002cd3b30:acrotray.exe                     2144   1484      2     60 2018-04-06 23:13:08 UTC+0000
 0xfffffa8000e9f060:7004af389d633b                    512   2156      0 ------ 2018-04-13 01:28:25 UTC+0000
. 0xfffffa8000e26790:7004af389d633b                  1400    512      0 ------ 2018-04-13 01:29:07 UTC+0000
.. 0xfffffa8000efa480:cmd.exe                        2920   1400      0 ------ 2018-04-13 01:29:15 UTC+0000
.. 0xfffffa8000e2cb30:aifkydk.exe                    2652   1400      0 ------ 2018-04-13 01:29:11 UTC+0000
... 0xfffffa8000edeb30:aifkydk.exe                   1728   2652      0 ------ 2018-04-13 01:29:54 UTC+0000
.... 0xfffffa8000dbfb30:bcdedit.exe                  2208   1728      0 ------ 2018-04-13 01:30:11 UTC+0000
.... 0xfffffa8000f22060:cmd.exe                      1080   1728      0 ------ 2018-04-13 01:37:57 UTC+0000
.... 0xfffffa8000fe8930:notepad.exe                   364   1728      1     68 2018-04-13 01:37:40 UTC+0000
.... 0xfffffa8000f5c6a0:bcdedit.exe                  2852   1728      0 ------ 2018-04-13 01:30:14 UTC+0000
.... 0xfffffa8002551550:bcdedit.exe                   812   1728      0 ------ 2018-04-13 01:30:08 UTC+0000
.... 0xfffffa8002afe060:vssadmin.exe                  580   1728      0 ------ 2018-04-13 01:30:09 UTC+0000
.... 0xfffffa8000f22920:bcdedit.exe                  2768   1728      0 ------ 2018-04-13 01:30:13 UTC+0000
.... 0xfffffa8000e2bb30:bcdedit.exe                  1752   1728      0 ------ 2018-04-13 01:30:10 UTC+0000
.... 0xfffffa80011c9060:vssadmin.exe                  400   1728      0 ------ 2018-04-13 01:37:46 UTC+0000
.... 0xfffffa800124e920:iexplore.exe                 2800   1728     13    547 2018-04-13 01:37:41 UTC+0000
..... 0xfffffa8001153240:iexplore.exe                2120   2800      6    351 2018-04-13 01:37:47 UTC+0000
 0xfffffa8002519060:winlogon.exe                      444    384      3    108 2018-04-06 23:12:43 UTC+0000
 0xfffffa8001c9a4e0:csrss.exe                         404    384      8    259 2018-04-06 23:12:43 UTC+0000
. 0xfffffa800288a5e0:conhost.exe                     1916    404      2     53 2018-04-06 23:12:59 UTC+0000
```

1.2 Suspicious Process Analysis

To identify potentially suspicious processes, we used pslist to enumerate all active processes and pstree to examine their structure in memory. We then compared the observed processes with known baseline processes for a clean Windows 7 system.

The following processes were found to deviate from expected behavior, naming, or activity patterns:

PID	Process Name	Why It Is Suspicious
512 1400	7004af389d633b	The process name is a non-standard, randomly generated string, which suggests an attempt to evade detection (obfuscation). It has 0 threads, no handles, and was active only for a brief moment. It is not part of any known legitimate Windows process baseline.
2652, 1728	aifkydk.exe	This executable name does not match any known system or third-party software. It appears twice in a fast sequence (2652: 2018-04-13 01:29:11 UTC, and 1728: 2018-04-13 01:29:54 UTC), both times with zero threads, no handles, and short runtimes.
2920, 1080	cmd.exe	These command shells appear to indicate no interaction with the system, as they have 0 threads and no handles and were initiated by suspicious processes (7004af389d633b and aifkydk.exe). This behaviour suggests it was likely triggered by a script or background process, rather than manually executed by the user.
812, 1752, 2208, 2768, 2852	bcdedit.exe	This is a legitimate Windows tool used to modify boot configuration settings. However, five instances of this process were launched in rapid succession, each with zero threads, no handles, and terminating almost immediately. This behaviour is highly unusual in a typical user environment. It indicates that the process was likely activated by a script or another program, rather than being run directly by the user.
580, 400	vssadmin.exe	This process is a legitimate Windows utility for managing Volume Shadow Copies. In this case, two instances were executed with zero threads, no handles, and terminated almost immediately. Both were launched by a suspicious parent process (aifkydk.exe), which is not a well-known system application. This unusual behavior increases the possibility of potential misuse to interfere with system backups.
2800, 2120	iexplore.exe	Although iexplore.exe is a legitimate browser process, its launch path is irregular. Instead of being started by the user, it was spawned from a suspicious chain of processes not usually associated with web activity. This deviates from expected behavior and may indicate the process was used in a non-standard or potentially malicious way.

2. Determine and explain the relationships (i.e., parent-child) between the suspicious processes identified above. Identify which process is most likely responsible for the initial exploit. [1.5%].

Solution:
Tools used: Volatility v2.6

Using the parent-child relationships extracted from the pstree plugin and confirmed with pslist, we can trace how the suspicious processes are related through their hierarchical structure. The diagram below summarizes this in a Suspicious Process Tree.

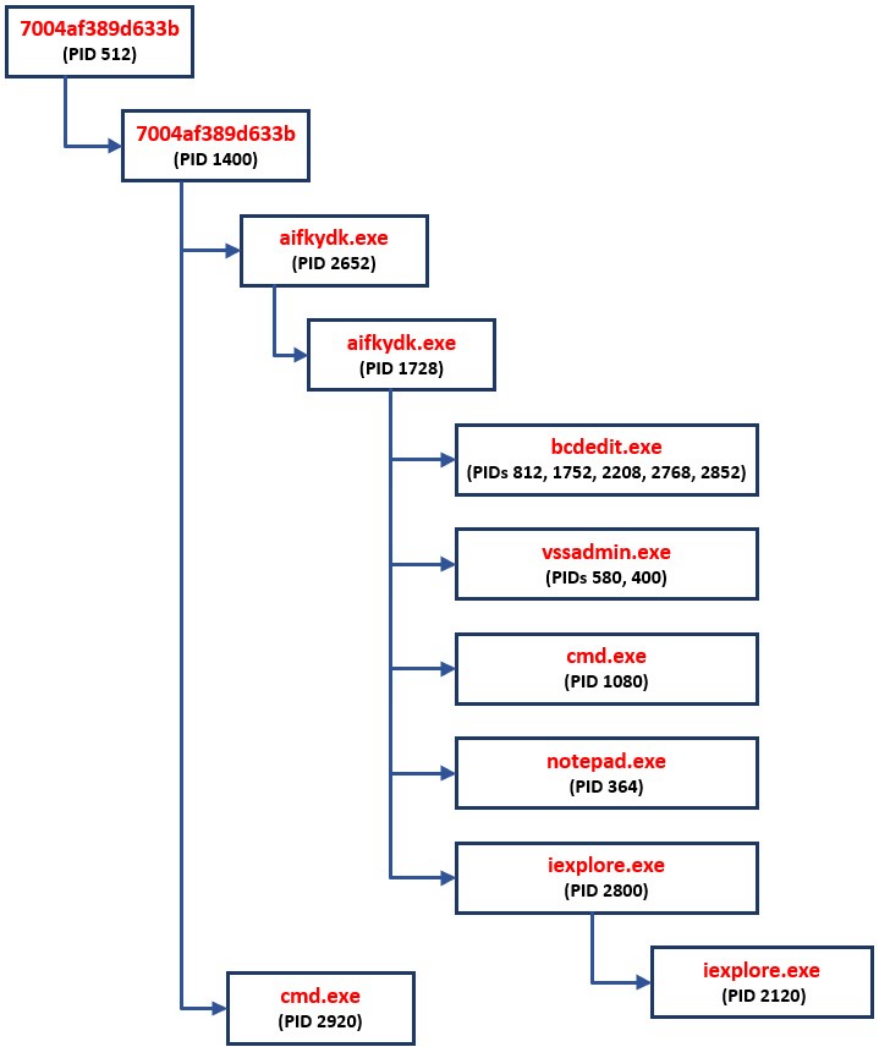


Figure: Suspicious Process Tree.

2.1 Relationships Explanation

7004af389d633b (PID 512) is detected as the first process unfamiliar with the common Windows process tree. Its name is non-standard, and it was launched by a process (PID 2156) not visible in the pslist output, potentially missing from the image. This process creates a second instance of itself (PID 1400), which then launches the rest of the suspicious activity.

From PID 1400, the first process to launch is aifkydk.exe (PID 2652), a process with no known legitimate function or origin, and aifkydk.exe runs again (PID 1728), and that second instance becomes the parent of:

- Five bcdedit.exe instances
- Two vssadmin.exe instances
- One cmd.exe (PID 1080)
- Notepad.exe (PID 364)
- Iexplore.exe (PIDs 2800 → 2120)

This structure reveals a scripted execution of multiple tools in quick sequence, which is uncommon in regular user activity or an indicator of post-exploitation.

2.2 Identification of the most likely process responsible for the initial exploit.

The process 7004af389d633b (PID 512) is probably the initial entry point. It is the first anomalous process, has no clear parent, uses a non-legitimate name, and immediately generates a clone (PID 1400), which launches several tools in sequence, indicating scripted or automated malicious behavior.

3. From the above list of suspicious processes, identify at least one process with hidden or injected code/DLLs, and identify corresponding hidden DLLs [0.5%].

Solution:
Tools used: Volatility v2.6 (plugins: malfind, procdump and ldrmodules), Virustotal.com

To investigate possible code injection or hidden DLLs, we analyzed iexplore.exe (PID 2800), a process previously flagged as suspicious in Questions 1 and 2.
The analysis was performed in three steps:

3.1. Suspicious memory detection using the malfind plugin.

We used the following command:

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ vol.py malfind -p 2800 > malfind_iexplore.txt

Volatility Foundation Volatility Framework 2.6
```

The output showed that iexplore.exe (PID 2800) has multiple memory regions marked with PAGE_EXECUTE_READWRITE permissions. These regions contain decoded assembly instructions (MOV, PUSH).

```
Process: iexplore.exe Pid: 2800 Address: 0x90000
Vad Tag: VadS Protection: PAGE_EXECUTE_READWRITE
Flags: CommitCharge: 1, MemCommit: 1, PrivateMemory: 1, Protection: 6

0x00090000  55 89 e5 83 ec 28 c7 45 f4 00 00 00 00 8b 45 08  U....(.E......E.
0x00090010  8b 10 8b 45 08 8d 48 08 8d 45 f0 89 44 24 0c 89  ...E..H..E..D$..
0x00090020  4c 24 08 c7 44 24 04 00 00 00 00 c7 04 24 00 00  L$..D$.......$..
0x00090030  00 00 ff d2 83 ec 10 85 c0 79 0b 8b 45 08 8b 40  .........y..E..@

0x00090000 55              PUSH EBP
0x00090001 89e5            MOV EBP, ESP
0x00090003 83ec28          SUB ESP, 0x28
```

One region even included part of a suspicious DLL path: \tmp\ifsubua\bin\monitor-x86.dll:

```
Process: iexplore.exe Pid: 2800 Address: 0x70000
Vad Tag: VadS Protection: PAGE_EXECUTE_READWRITE
Flags: CommitCharge: 1, MemCommit: 1, PrivateMemory: 1, Protection: 6

0x00070000  43 00 3a 00 5c 00 74 00 6d 00 70 00 69 00 66 00  C.:.\.t.m.p.i.f.
0x00070010  73 00 62 00 75 00 61 00 5c 00 62 00 69 00 6e 00  s.b.u.a.\.b.i.n.
0x00070020  5c 00 6d 00 6f 00 6e 00 69 00 74 00 6f 00 72 00  \.m.o.n.i.t.o.r.
0x00070030  2d 00 78 00 38 00 36 00 2e 00 64 00 6c 00 6c 00  -.x.8.6...d.l.l.
```

This, combined with the fact that this process was already identified as due to its unusual parent and launch sequence, suggests that iexplore.exe (PID 2800) contains injected or hidden executable code.

3.2. Memory dump and Virustotal verification

To validate this, we dumped the memory of the iexplore.exe process:

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ vol.py procdump -p 2800 -D ./dumps_ie
Volatility Foundation Volatility Framework 2.6
Process(V)         ImageBase          Name                 Result
------------------ ------------------ -------------------- ------
0xfffffa800124e920 0x0000000000d70000 iexplore.exe         OK: executable.2800.exe
```

The output file (executable.2800.exe) was submitted to VirusTotal, where 1 out of 72 antivirus engines identified it as suspicious (Win/malicious_confidence_60% (W) by CrowdStrike Falcon). While not conclusive, this supports the evidence of suspicious behavior.

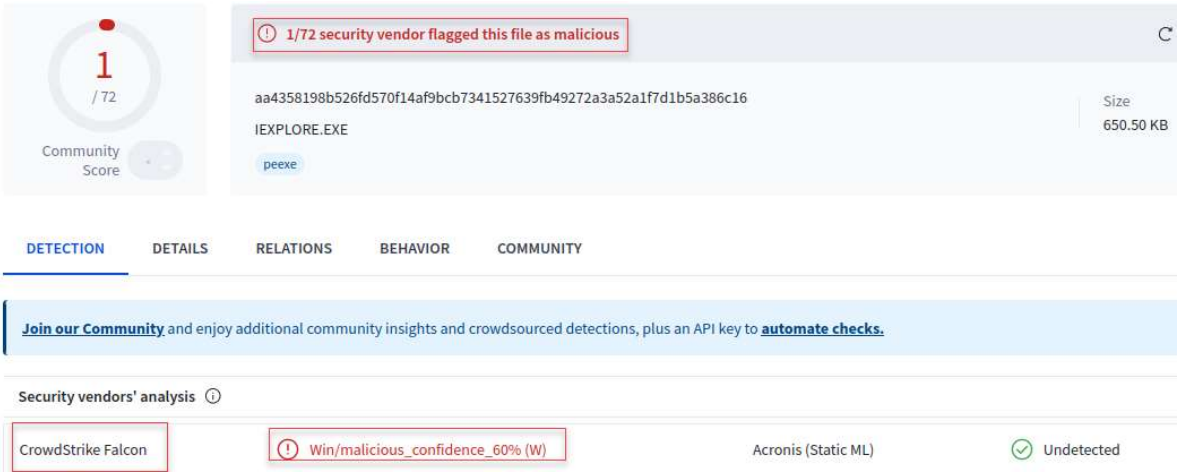


Figure. Virustotal.com verification for the iexplore.exe process.

3.3. Hidden DLLs modules with the ldrmodules plugin

Finally, we used the ldrmodules plugin to identify the corresponding hidden DLLs loaded by this process:

```
┌──(kali㉿kali)-[~/Project_1_570/ECE570-project-1-memory]
└─$ vol.py ldrmodules -p 2800
Volatility Foundation Volatility Framework 2.6
Pid      Process            Base               InLoad InInit InMem MappedPath
-------- ------------------ ------------------ ------ ------ ----- ----------
    2800 iexplore.exe       0x0000000000d70000 True   False  True  \Program Files (x86)\Internet Explorer\iexplore.exe
    2800 iexplore.exe       0x0000000000020000 False  False  False \Program Files (x86)\Internet Explorer\en-US\iexplo
    2800 iexplore.exe       0x0000000074340000 False  False  False \Windows\SysWOW64\WSHTCPIP.DLL
    2800 iexplore.exe       0x0000000002f60000 False  False  False \Windows\SysWOW64\en-US\urlmon.dll.mui
    2800 iexplore.exe       0x0000000074570000 False  False  False \Windows\SysWOW64\dnsapi.dll
    2800 iexplore.exe       0x0000000075fe0000 False  False  False \Windows\SysWOW64\clbcatq.dll
    2800 iexplore.exe       0x00000000745f0000 False  False  False \Windows\SysWOW64\rsaenh.dll
    2800 iexplore.exe       0x000000006bf80000 False  False  False \Windows\SysWOW64\dhcpcsvc.dll
    2800 iexplore.exe       0x0000000073a50000 False  False  False \Windows\SysWOW64\ieui.dll
    2800 iexplore.exe       0x0000000076670000 False  False  False \Windows\SysWOW64\ws2_32.dll
    2800 iexplore.exe       0x0000000075290000 False  False  False \Windows\SysWOW64\user32.dll
    2800 iexplore.exe       0x0000000004eb0000 False  False  False \Windows\SysWOW64\en-US\KernelBase.dll.mui
    2800 iexplore.exe       0x0000000074ad0000 False  False  False \Windows\SysWOW64\cryptbase.dll
    2800 iexplore.exe       0x0000000074540000 False  False  False \Windows\SysWOW64\winnsi.dll
    2800 iexplore.exe       0x0000000074390000 False  False  False \Windows\SysWOW64\rasapi32.dll
    2800 iexplore.exe       0x000000000769a0000 False  False  False \Windows\SysWOW64\kernel32.dll
    2800 iexplore.exe       0x00000000745c0000 False  False  False \Windows\SysWOW64\ntmarta.dll
    2800 iexplore.exe       0x00000000741e0000 False  False  False \Windows\SysWOW64\wship6.dll
    2800 iexplore.exe       0x0000000074410000 False  False  False \Windows\SysWOW64\mswsock.dll
    2800 iexplore.exe       0x0000000072690000 False  False  False \Windows\SysWOW64\xmllite.dll
    2800 iexplore.exe       0x0000000071a20000 False  False  False \Windows\SysWOW64\userenv.dll
    2800 iexplore.exe       0x000000006d030000 False  False  False \Windows\SysWOW64\dui70.dll
    2800 iexplore.exe       0x0000000074c50000 False  False  False \Windows\SysWOW64\crypt32.dll
```

This revealed several DLLs where all three flags: InLoad, InInit, and InMem were set to False. This means that, although the DLLs are mapped in memory, they are not fully loaded or initialized by the system, which is a known method used to hide injected or suspicious code.

4. Extract the executables for one of the suspicious processes identified above, and check whether at least one of these files is malicious using an online virus scanner [0.5%].

Solution:
Tools used: Volatility v2.6, Virustotal.com

The first method attempted to extract executable (.exe) files from a suspicious process (iexplore.exe – 2800 / 2120) was by using the memdump plugin in Volatility. After dumping the process memory, the corresponding folder labeled "exe" was examined. However, all the files retrieved displayed a file size of 0 bytes, suggesting that the memory regions where the executable was expected were either paged out or inaccessible at the time of acquisition. As a result, no valid executable files could be obtained through this method.

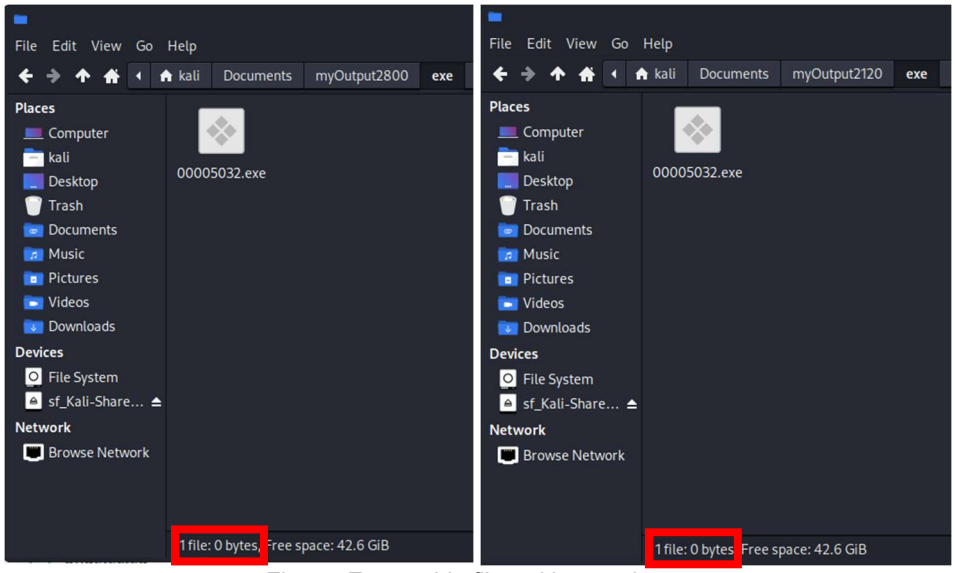


Figure. Executable files with memdump.

An attempt was made to scan the extracted executable file using VirusTotal. However, the analysis could not be completed successfully, as the platform failed to process the file. This may be due to file corruption, improper extraction, or the file being incomplete (e.g., 0 bytes in size). As a result, no malware classification could be confirmed through VirusTotal for this sample.

The second method used to extract the executable file was through the procdump Volatility plugin. This approach proved to be more effective than memdump, as it successfully generated a non-zero byte executable file for processes PID 2800 and 2120. The output is shown below:

```
┌──(root💀kali)-[/home/kali/Desktop/volatility-2.6]
└─# vol.py procdump -p 2800 --dump-dir=/home/kali/Documents/
Volatility Foundation Volatility Framework 2.6
Process(V)         ImageBase          Name                 Result
------------------ ------------------ -------------------- ------
0xfffffa800124e920 0x0000000000d70000 iexplore.exe         OK: executable.2800.exe
┌──(root💀kali)-[/home/kali/Desktop/volatility-2.6]
└─# vol.py procdump -p 2120 --dump-dir=/home/kali/Documents/
Volatility Foundation Volatility Framework 2.6
Process(V)         ImageBase          Name                 Result
------------------ ------------------ -------------------- ------
0xfffffa8001153240 0x0000000000d70000 iexplore.exe         OK: executable.2120.exe
```

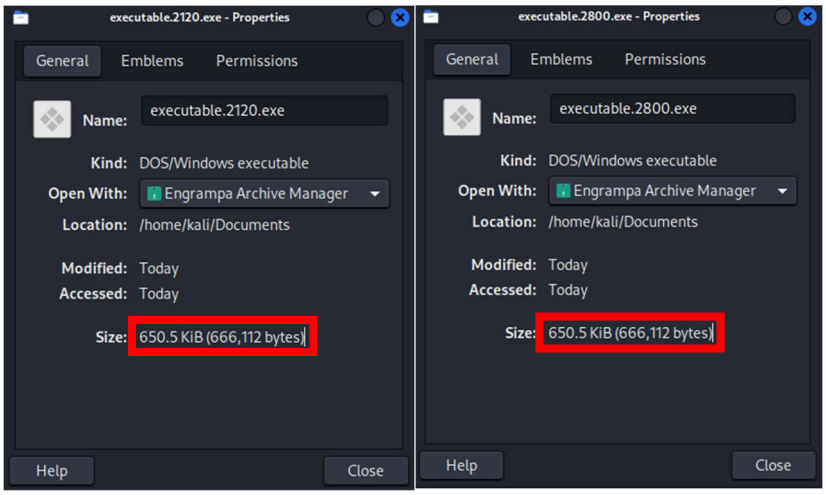


Figure. Executable file with procdump.

The analysis of the extracted executable files flagged them as malicious by CrowdStrike Falcon on VirusTotal. The results are shown below.

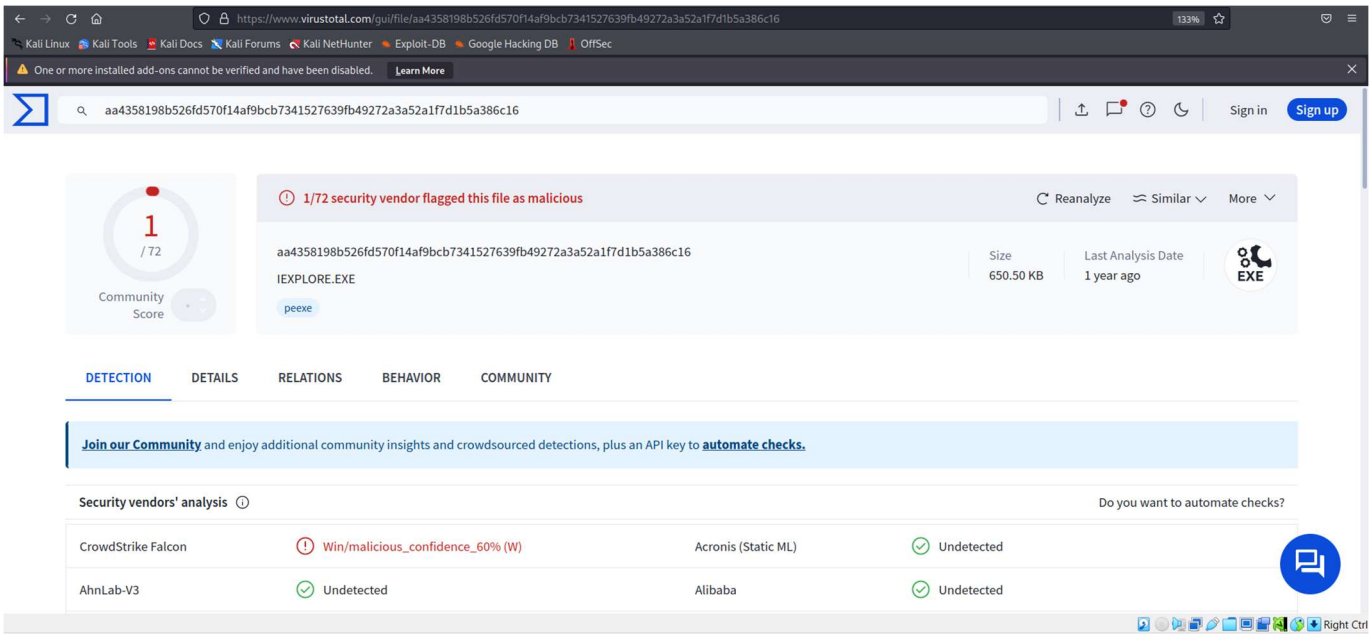


Figure. Virustotal result with the executable file.

5. Identify the URLs (and a corresponding IP address) for one of the possible remote command and control servers visited by the malware. Confirm that the selected URL is malicious using an online scanner. Note: You can limit the search to the initial (suspicious) process that triggered the exploit, or any other relevant process [1%].

Solution:
Tools used: Volatility v2.6, Virustotal.com

Based on the guidance provided in Tutorial #3, the strings tool was used to search for the pattern http:// to identify potentially malicious URLs. The analysis was focused on the process iexplore.exe (PIDs 2800 and 2120). The following URLs were identified:

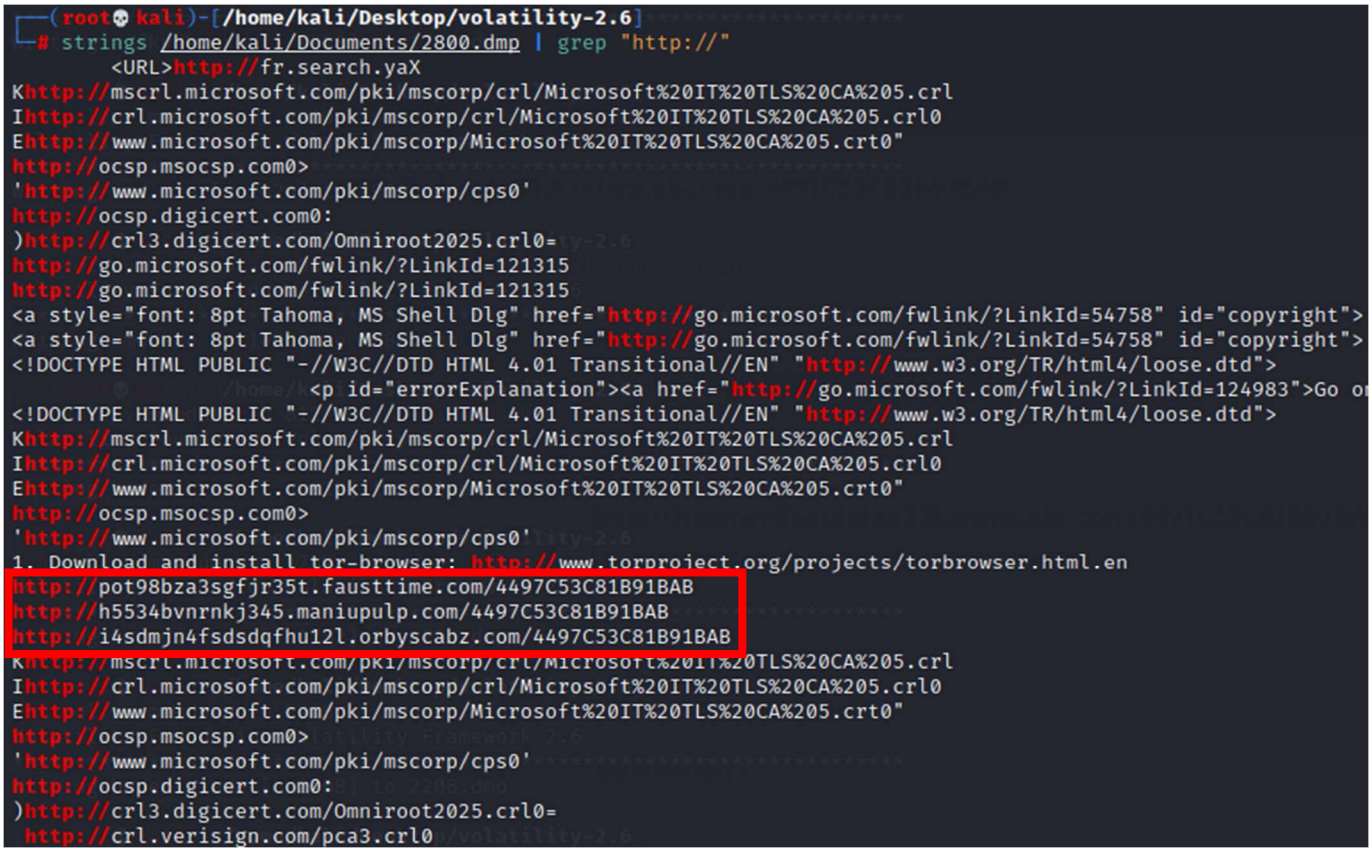


Figure. Malicious URL from p 2800.

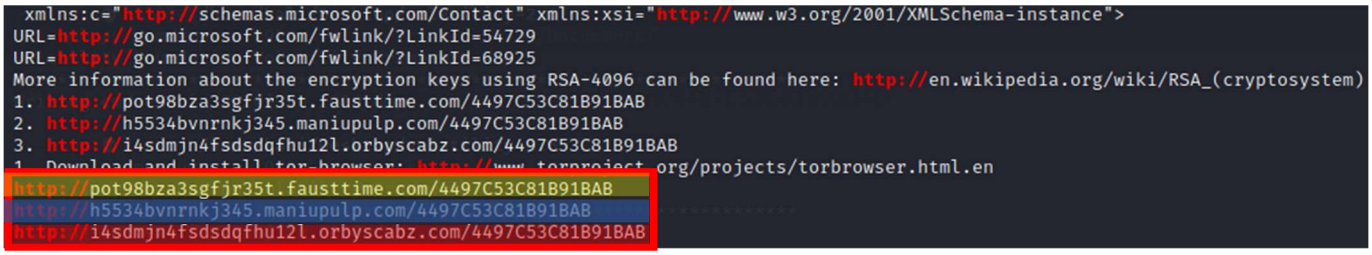


Figure. Malicious URL from p 2120.

VirusTotal includes a feature to scan suspicious URLs and determine whether they are flagged as potential threats. The summary of the results is as follows:

- First URL (highlighted in yellow): Identified as phishing by Avira and malicious by Sophos.
- Second URL (highlighted in blue): Detected as malware-related by Avira and Fortinet, and malicious by Sophos.
- Third URL (highlighted in red): Flagged as malicious by Sophos.

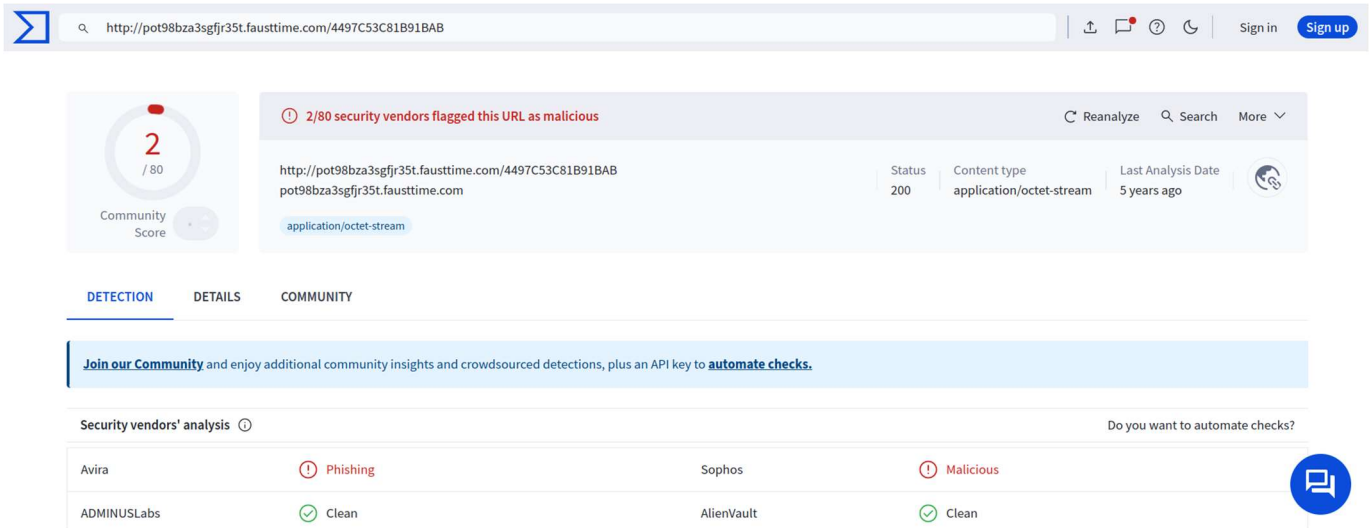


Figure. Evidence of Virustotal findings.

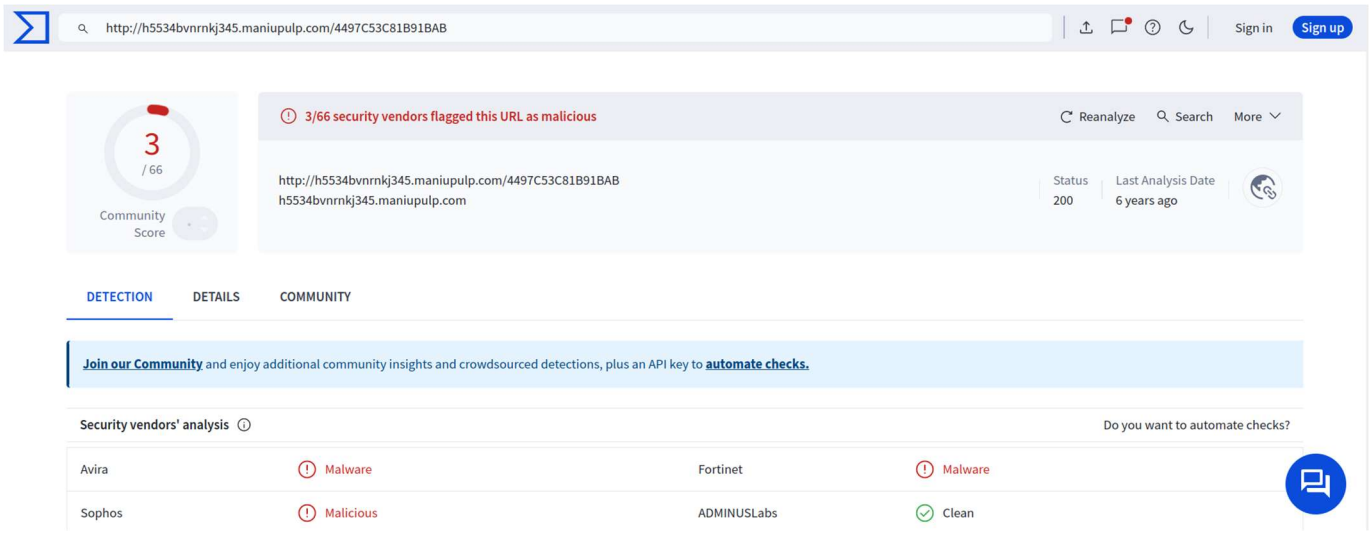


Figure. Evidence of Virustotal findings.

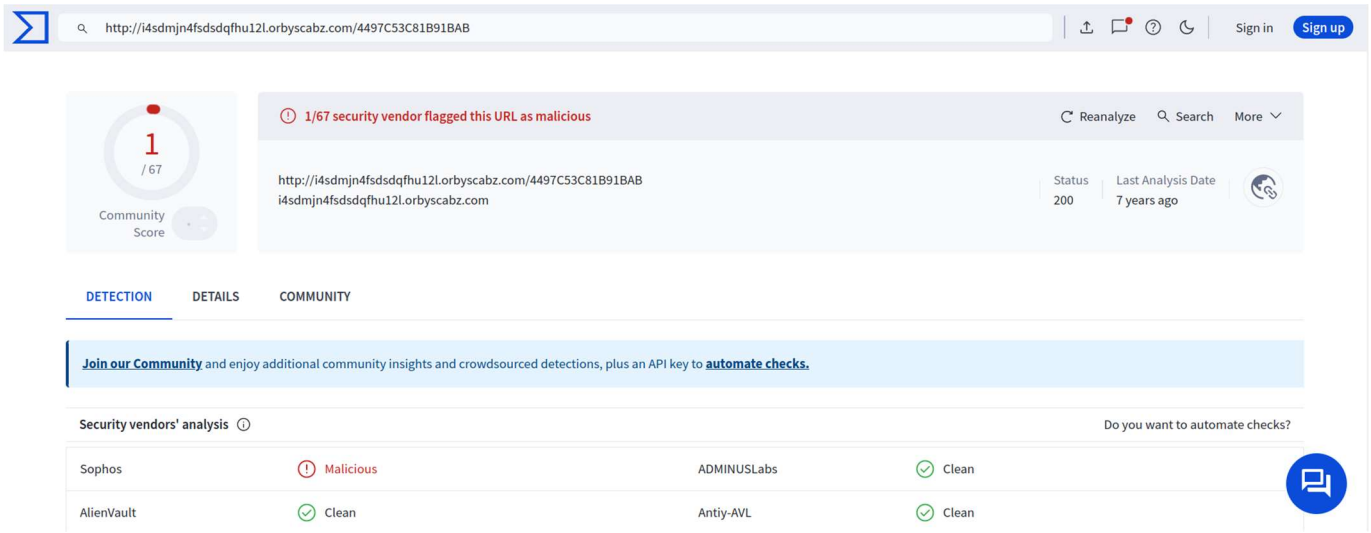
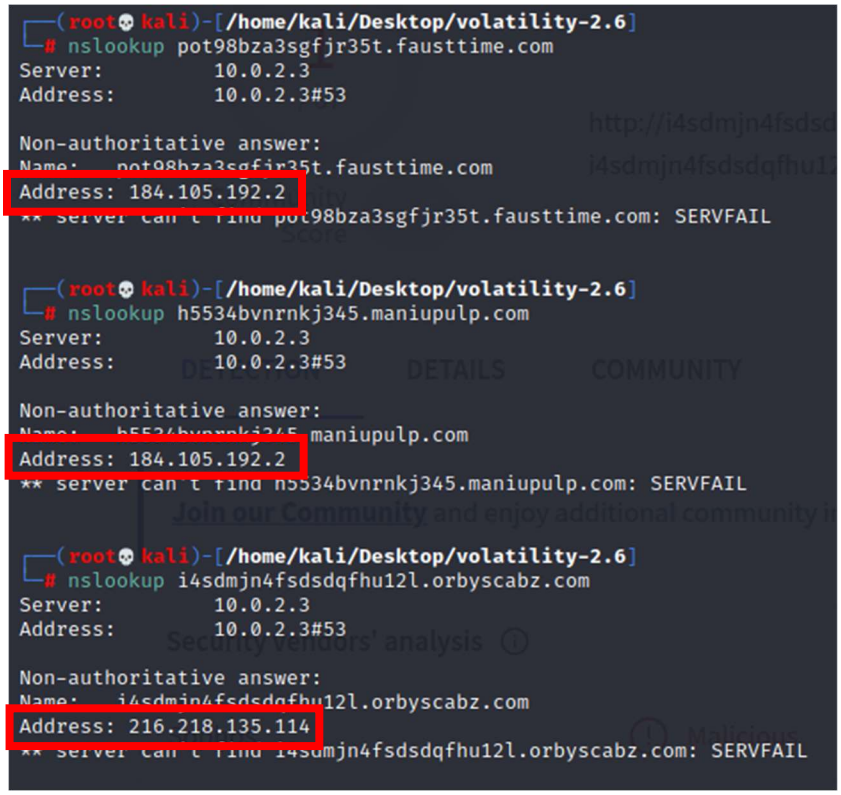


Figure. Evidence of Virustotal findings.

With the URLs identified as suspicious, the next step was to obtain their corresponding IP addresses. To accomplish this, the nslookup command was used. The following results were obtained:



While analyzing the suspicious URLs extracted from the malicious executables, the nslookup command was used to obtain their associated IP addresses. Although the IP addresses were successfully identified, all queries returned a "SERVFAIL" error, indicating that the domains are no longer active or have been taken down. This behavior is common in malware campaigns, where attackers use temporary servers to hide their infrastructure. Therefore, these results support the hypothesis that the URLs were linked to malicious activity.

6. List available registry hives and identify a potentially malicious hive from the list. Explain and justify why such hive could potentially be malicious [3%].

Solution:
Tools used: Volatility v2.6, crackstation.com

Hivelist was the Volatility tool selected to retrieve the list of registry hives for this project. This tool displays the virtual and physical memory addresses, as well as the names associated with each hive. The output obtained from this tool is shown below:

```
┌──(root💀kali)-[/home/kali]
└─# vol.py hivelist
Volatility Foundation Volatility Framework 2.6
Virtual            Physical           Name
------------------ ------------------ ----

0xfffff8a001a05010 0x000000001a3ac010 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0xfffff8a003465010 0x0000000024112010 \Device\HarddiskVolume1\Boot\BCD
0xfffff8a0034d4010 0x000000000eb9f010 \??\C:\Users\Win7\ntuser.dat
0xfffff8a003ec1010 0x0000000002223d010 \SystemRoot\System32\Config\DEFAULT
0xfffff8a003ed4010 0x0000000021ca2010 \SystemRoot\System32\Config\SAM
0xfffff8a006c98010 0x000000000eb53010 \??\C:\Users\Win7\AppData\Local\Microsoft\Windows\UsrClass.dat
0xfffff8a00000f010 0x000000000275ea010 [no name]
0xfffff8a000024010 0x000000000276b5010 \REGISTRY\MACHINE\SYSTEM
0xfffff8a00004e420 0x000000000276df420 \REGISTRY\MACHINE\HARDWARE
0xfffff8a000711420 0x000000000251e2420 \SystemRoot\System32\Config\SOFTWARE
0xfffff8a000727010 0x0000000002526b010 \SystemRoot\System32\Config\SECURITY
0xfffff8a000ce3010 0x000000000a12e010 \??\C:\System Volume Information\Syscache.hve
0xfffff8a001331010 0x0000000019945010 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
```

Figure. Hivelist evidence.

One of the first potentially suspicious hives identified is located at 0xfffff8a00000f010 0x000000000275ea010, labelled as [no name]. The absence of a defined image name or the presence of corrupted metadata may point to advanced techniques such as code injection, memory manipulation, or the use of stealthy loaders to evade detection. To further investigate, the hivedump plugin was executed, revealing the following details.

```
┌──(root💀kali)-[/home/kali]
└─# vol.py hivedump -o 0xfffff8a00000f010
Volatility Foundation Volatility Framework 2.6
Last Written         Key
2018-04-06 23:12:30 UTC+0000 \REGISTRY
2018-04-06 23:12:58 UTC+0000 \REGISTRY\A
2018-04-06 23:12:38 UTC+0000 \REGISTRY\MACHINE
2018-04-06 23:12:48 UTC+0000 \REGISTRY\USER
```

Figure. Hivedump result.

Additionally, the Regripper tool was used to analyze the hive, generating both a report and a log file. According to the log, a significant number of plugins failed with the message "Can't get method 'get_root_key'". Out of 102 plugins executed, many returned no output, suggesting that the registry hive may be incomplete, corrupted, or not fully accessible.

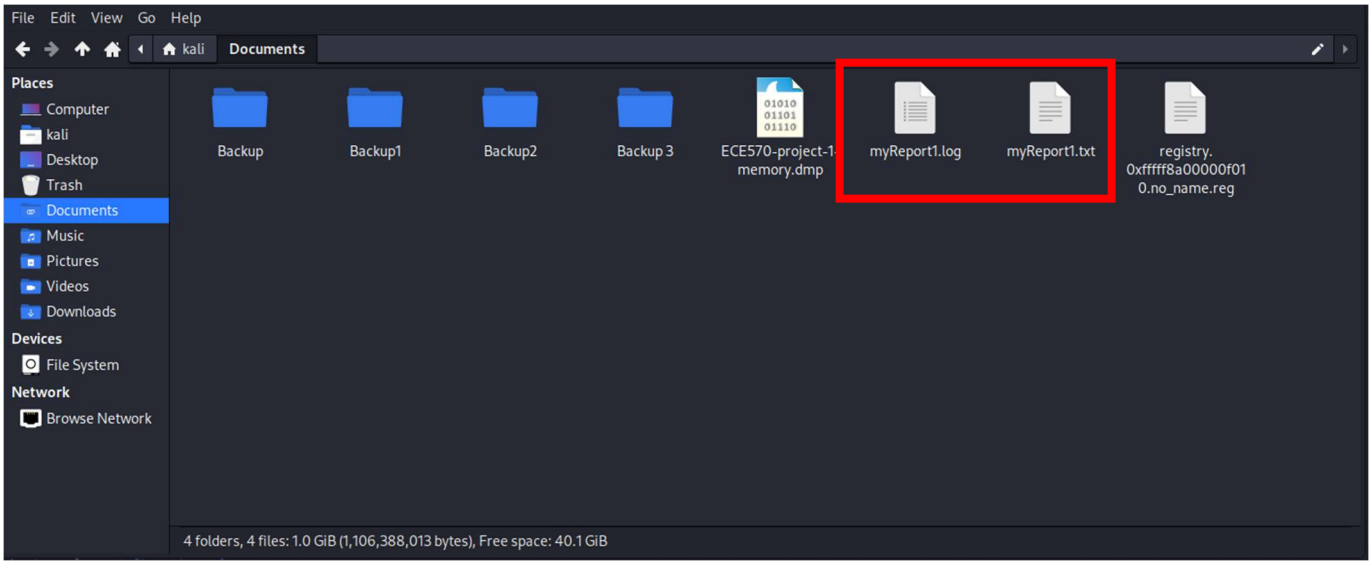


Figure. Regripper files .log and report.txt

The next set of suspicious hives includes 0xfffff8a003ed4010 0x0000000021ca2010 \SystemRoot\System32\Config\SAM and 0xfffff8a000024010 0x00000000276b5010 \REGISTRY\MACHINE\SYSTEM, as these contain the Windows password hashes for local user accounts.

Using the hashdump plugin, both hives were successfully parsed. However, the results were unusual — the NTLM hashes for all three user accounts (Administrator, Guest, and Win7) were identical. This may suggest that all accounts share the same password.

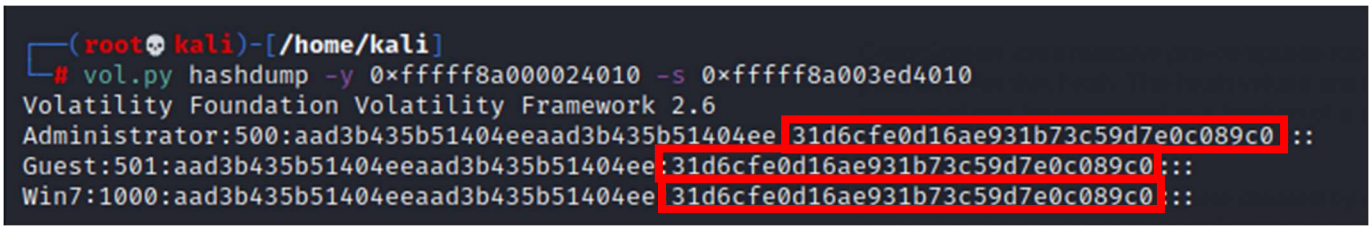


Figure. Hashdump results.

Using the CrackStation website to attempt password recovery from the extracted NTLM hash, the following result was obtained:

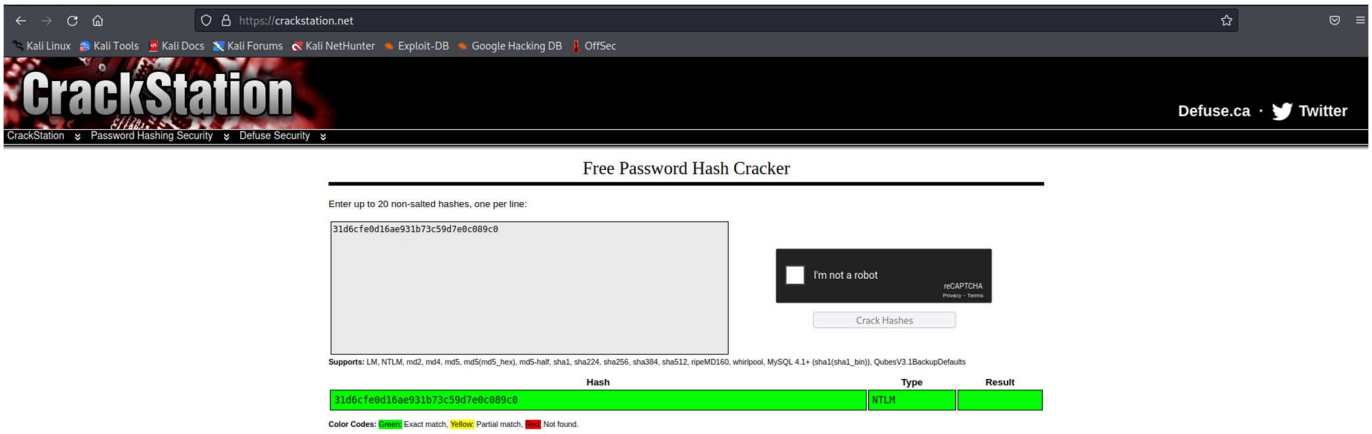


Figure. CrackStation Results.

However, the password could not be resolved, which may indicate it is either too complex, not present in the CrackStation database, or possibly blank.