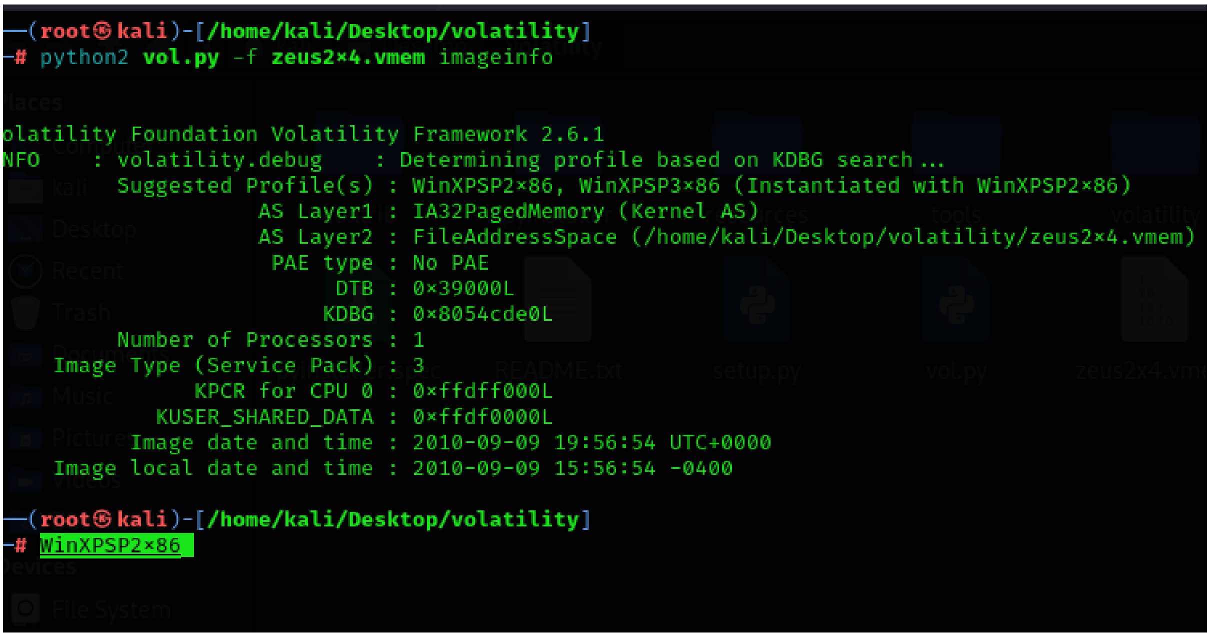
Determining the Profile

To begin, we'll determine the image profile by utilizing the imageinfo plugin.

python2 vol.py -f zeus2x4.vmem imageinfo



```
—(root㉿kali)-[/home/kali/Desktop/volatility]
-# python2 vol.py -f zeus2x4.vmem imageinfo

olatility Foundation Volatility Framework 2.6.1
NFO    : volatility.debug    : Determining profile based on KDBG search ...
         Suggested Profile(s) : WinXPSP2x86, WinXPSP3x86 (Instantiated with WinXPSP2x86)
                    AS Layer1 : IA32PagedMemory (Kernel AS)
                    AS Layer2 : FileAddressSpace (/home/kali/Desktop/volatility/zeus2x4.vmem)
                     PAE type : No PAE
                          DTB : 0x39000L
                         KDBG : 0x8054cde0L
         Number of Processors : 1
    Image Type (Service Pack) : 3
               KPCR for CPU 0 : 0xffdff000L
            KUSER_SHARED_DATA : 0xffdf0000L
          Image date and time : 2010-09-09 19:56:54 UTC+0000
    Image local date and time : 2010-09-09 15:56:54 -0400

—(root㉿kali)-[/home/kali/Desktop/volatility]
-# WinXPSP2x86
```

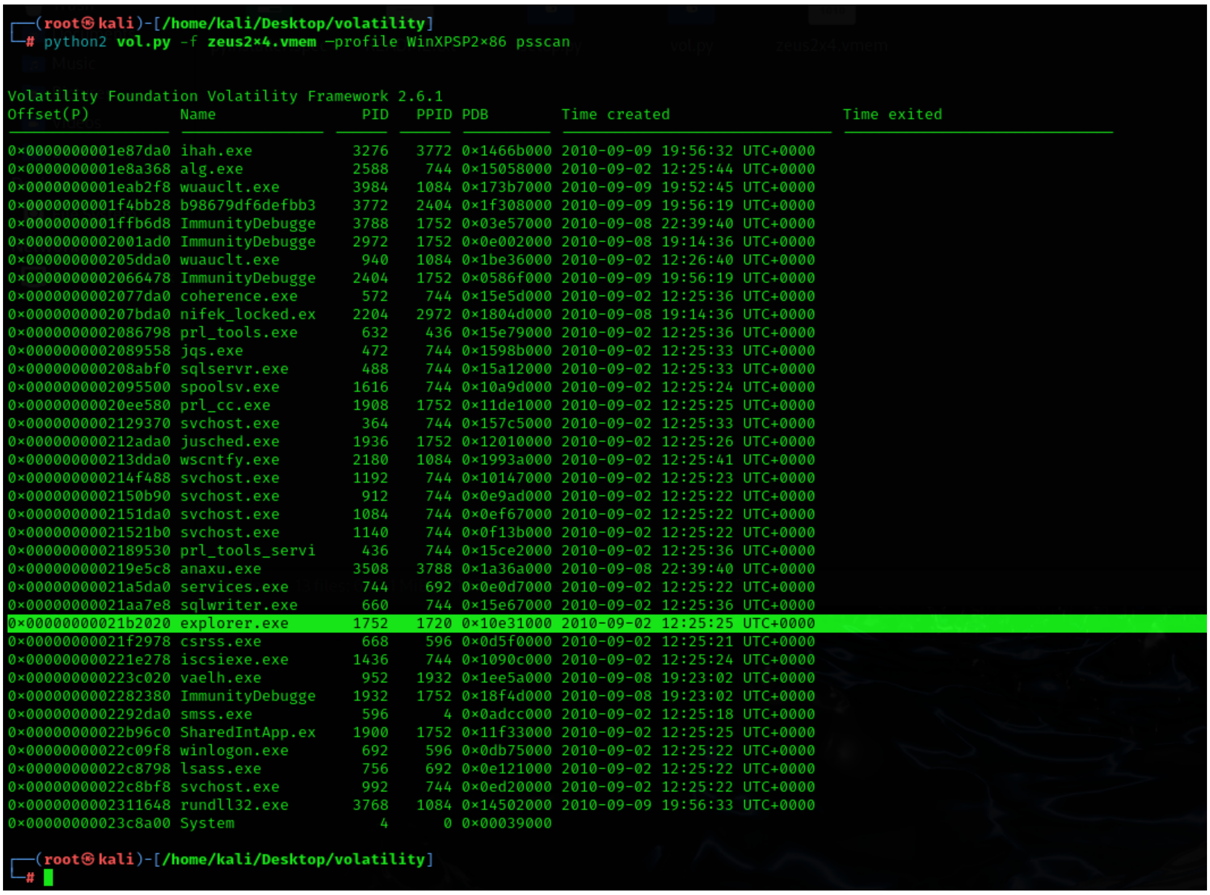
It is use windowsXP

From the recommended profiles, we'll choose "WinXPSP2x86" and proceed.

Examining Processes

After identifying the profile of the memory image, we can begin investigating for unusual activities by listing the processes using the psscan plugin:

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 psscan

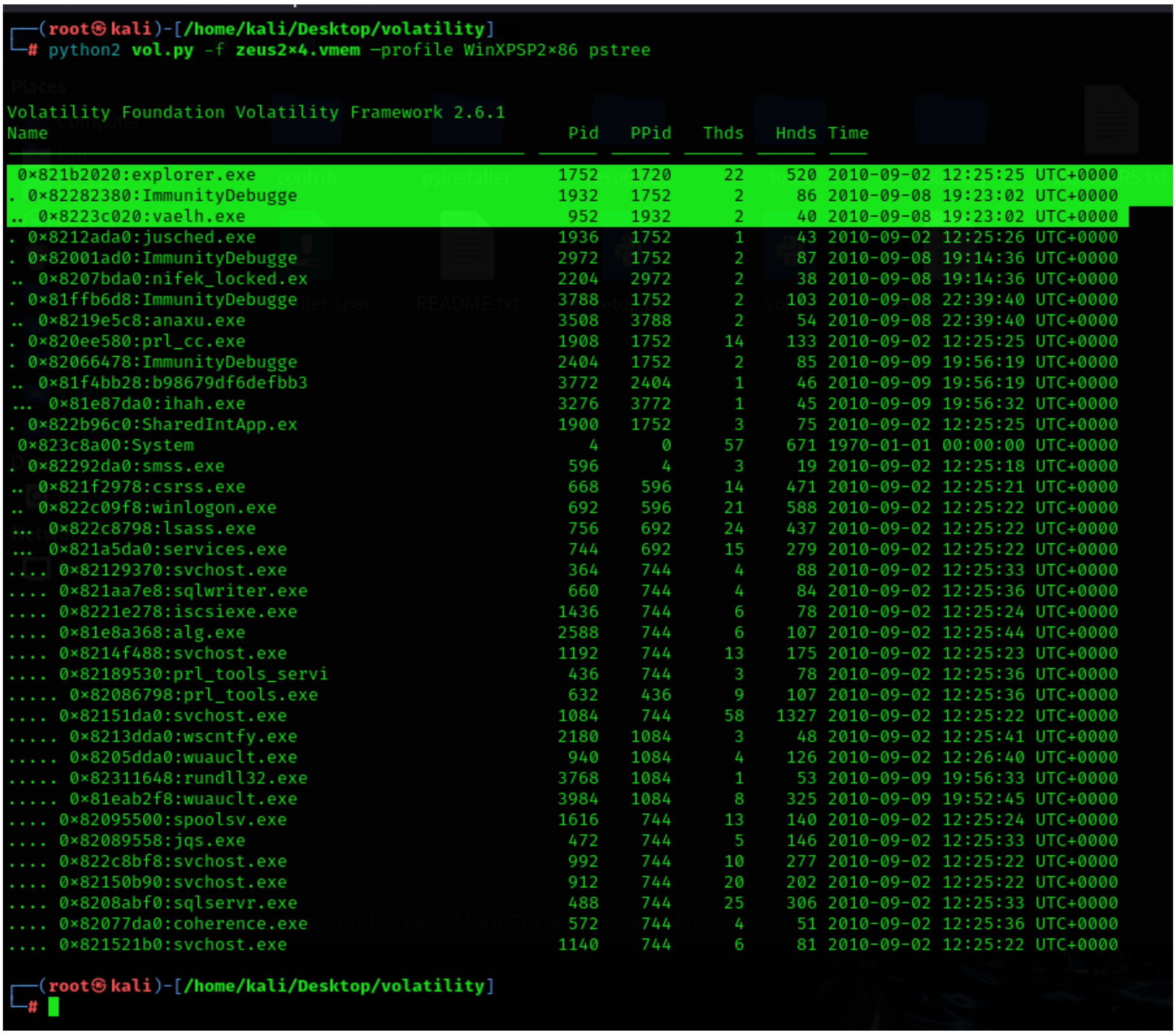


```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 psscan

Volatility Foundation Volatility Framework 2.6.1
Offset(P)          Name                PID   PPID PDB        Time created                   Time exited
------------------ ---------------- ------ ------ ---------- ------------------------------ ------------------------------
0×000000001e87da0 ihah.exe           3276   3772 0×1466b000 2010-09-09 19:56:32 UTC+0000
0×000000001e8a368 alg.exe            2588    744 0×15058000 2010-09-02 12:25:44 UTC+0000
0×000000001eab2f8 wuauclt.exe        3984   1084 0×173b7000 2010-09-09 19:52:45 UTC+0000
0×000000001f4bb28 b98679df6defbb3    3772   2404 0×1f308000 2010-09-09 19:56:19 UTC+0000
0×000000001ffb6d8 ImmunityDebugge    3788   1752 0×03e57000 2010-09-08 22:39:40 UTC+0000
0×00000000002001ad0 ImmunityDebugge    2972   1752 0×0e002000 2010-09-08 19:14:36 UTC+0000
0×000000000205dda0 wuauclt.exe         940   1084 0×1be36000 2010-09-02 12:26:40 UTC+0000
0×00000000002066478 ImmunityDebugge    2404   1752 0×0586f000 2010-09-09 19:56:19 UTC+0000
0×00000000002077da0 coherence.exe       572    744 0×15e5d000 2010-09-02 12:25:36 UTC+0000
0×000000000207bda0 nifek_locked.ex    2204   2972 0×1804d000 2010-09-08 19:14:36 UTC+0000
0×00000000002086798 prl_tools.exe       632    436 0×15e79000 2010-09-02 12:25:36 UTC+0000
0×00000000002089558 jqs.exe             472    744 0×1598b000 2010-09-02 12:25:33 UTC+0000
0×000000000208abf0 sqlservr.exe        488    744 0×15a12000 2010-09-02 12:25:33 UTC+0000
0×00000000002095500 spoolsv.exe        1616    744 0×10a9d000 2010-09-02 12:25:24 UTC+0000
0×0000000020ee580 prl_cc.exe         1908   1752 0×11de1000 2010-09-02 12:25:25 UTC+0000
0×00000000002129370 svchost.exe         364    744 0×157c5000 2010-09-02 12:25:33 UTC+0000
0×000000000212ada0 jusched.exe        1936   1752 0×12010000 2010-09-02 12:25:26 UTC+0000
0×000000000213dda0 wscntfy.exe        2180   1084 0×1993a000 2010-09-02 12:25:41 UTC+0000
0×000000000214f488 svchost.exe        1192    744 0×10147000 2010-09-02 12:25:23 UTC+0000
0×00000000002150b90 svchost.exe         912    744 0×0e9ad000 2010-09-02 12:25:22 UTC+0000
0×00000000002151da0 svchost.exe        1084    744 0×0ef67000 2010-09-02 12:25:22 UTC+0000
0×000000000021521b0 svchost.exe        1140    744 0×0f13b000 2010-09-02 12:25:22 UTC+0000
0×00000000002189530 prl_tools_servi     436    744 0×15ce2000 2010-09-02 12:25:36 UTC+0000
0×000000000219e5c8 anaxu.exe          3508   3788 0×1a36a000 2010-09-08 22:39:40 UTC+0000
0×000000000021a5da0 services.exe        744    692 0×0e0d7000 2010-09-02 12:25:22 UTC+0000
0×000000000021aa7e8 sqlwriter.exe       660    744 0×15e67000 2010-09-02 12:25:36 UTC+0000
0×000000000021b2020 explorer.exe       1752   1720 0×10e31000 2010-09-02 12:25:25 UTC+0000
0×000000000021f2978 csrss.exe           668    596 0×0d5f0000 2010-09-02 12:25:21 UTC+0000
0×00000000000221e278 iscsiexe.exe       1436    744 0×1090c000 2010-09-02 12:25:24 UTC+0000
0×000000000223c020 vaelh.exe           952   1932 0×1ee5a000 2010-09-08 19:23:02 UTC+0000
0×00000000002282380 ImmunityDebugge    1932   1752 0×18f4d000 2010-09-08 19:23:02 UTC+0000
0×00000000002292da0 smss.exe            596      4 0×0adcc000 2010-09-02 12:25:18 UTC+0000
0×000000000022b96c0 SharedIntApp.ex    1900   1752 0×11f33000 2010-09-02 12:25:25 UTC+0000
0×000000000022c09f8 winlogon.exe        692    596 0×0db75000 2010-09-02 12:25:22 UTC+0000
0×000000000022c8798 lsass.exe           756    692 0×0e121000 2010-09-02 12:25:22 UTC+0000
0×000000000022c8bf8 svchost.exe         992    744 0×0ed20000 2010-09-02 12:25:22 UTC+0000
0×00000000002311648 rundll32.exe       3768   1084 0×14502000 2010-09-09 19:56:33 UTC+0000
0×00000000023c8a00 System                4      0 0×00039000

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─#
```

At this point, we don't observe anything unusual, as both the number and names of the processes appear normal.

Next, we can use the pstree plugin to analyze the parent-child relationships of the processes:

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 pstree



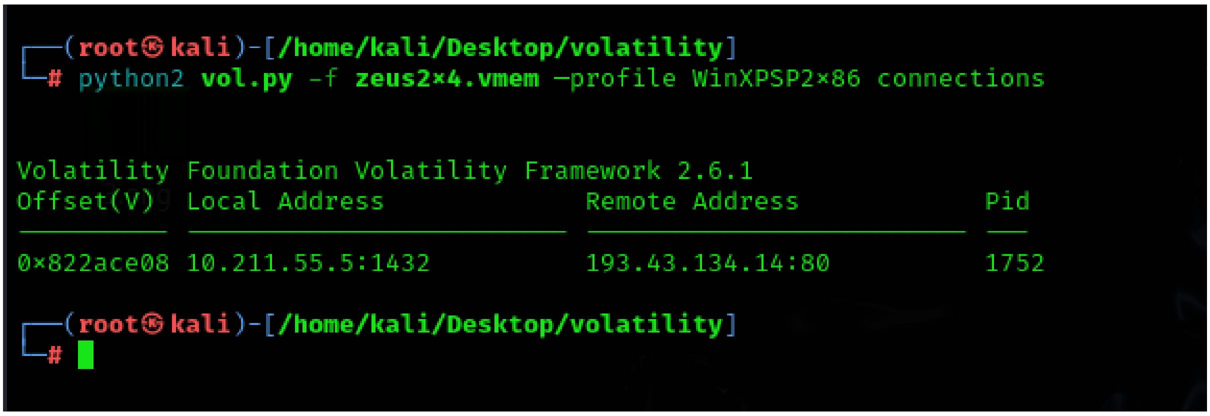
```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 pstree

Volatility Foundation Volatility Framework 2.6.1
Name                                                  Pid   PPid   Thds   Hnds Time
-------------------------------------------------- ------ ------ ------ ------ ----
 0×821b2020:explorer.exe                             1752   1720     22    520 2010-09-02 12:25:25 UTC+0000
. 0×82282380:ImmunityDebugge                         1932   1752      2     86 2010-09-08 19:23:02 UTC+0000
.. 0×8223c020:vaelh.exe                               952   1932      2     40 2010-09-08 19:23:02 UTC+0000
. 0×8212ada0:jusched.exe                             1936   1752      1     43 2010-09-02 12:25:26 UTC+0000
. 0×82001ad0:ImmunityDebugge                         2972   1752      2     87 2010-09-08 19:14:36 UTC+0000
.. 0×8207bda0:nifek_locked.ex                        2204   2972      2     38 2010-09-08 19:14:36 UTC+0000
. 0×81ffb6d8:ImmunityDebugge                         3788   1752      2    103 2010-09-08 22:39:40 UTC+0000
.. 0×8219e5c8:anaxu.exe                              3508   3788      2     54 2010-09-08 22:39:40 UTC+0000
. 0×820ee580:prl_cc.exe                              1908   1752     14    133 2010-09-02 12:25:25 UTC+0000
. 0×82066478:ImmunityDebugge                         2404   1752      2     85 2010-09-09 19:56:19 UTC+0000
.. 0×81f4bb28:b98679df6defbb3                        3772   2404      1     46 2010-09-09 19:56:19 UTC+0000
... 0×81e87da0:ihah.exe                              3276   3772      1     45 2010-09-09 19:56:32 UTC+0000
. 0×822b96c0:SharedIntApp.ex                         1900   1752      3     75 2010-09-02 12:25:25 UTC+0000
 0×823c8a00:System                                      4      0     57    671 1970-01-01 00:00:00 UTC+0000
. 0×82292da0:smss.exe                                 596      4      3     19 2010-09-02 12:25:18 UTC+0000
.. 0×821f2978:csrss.exe                               668    596     14    471 2010-09-02 12:25:21 UTC+0000
.. 0×822c09f8:winlogon.exe                            692    596     21    588 2010-09-02 12:25:22 UTC+0000
... 0×822c8798:lsass.exe                              756    692     24    437 2010-09-02 12:25:22 UTC+0000
... 0×821a5da0:services.exe                           744    692     15    279 2010-09-02 12:25:22 UTC+0000
.... 0×82129370:svchost.exe                           364    744      4     88 2010-09-02 12:25:33 UTC+0000
.... 0×821aa7e8:sqlwriter.exe                         660    744      4     84 2010-09-02 12:25:36 UTC+0000
.... 0×8221e278:iscsiexe.exe                         1436    744      6     78 2010-09-02 12:25:24 UTC+0000
.... 0×81e8a368:alg.exe                              2588    744      6    107 2010-09-02 12:25:44 UTC+0000
.... 0×8214f488:svchost.exe                          1192    744     13    175 2010-09-02 12:25:23 UTC+0000
.... 0×82189530:prl_tools_servi                       436    744      3     78 2010-09-02 12:25:36 UTC+0000
..... 0×82086798:prl_tools.exe                        632    436      9    107 2010-09-02 12:25:36 UTC+0000
.... 0×82151da0:svchost.exe                          1084    744     58   1327 2010-09-02 12:25:22 UTC+0000
..... 0×8213dda0:wscntfy.exe                         2180   1084      3     48 2010-09-02 12:25:41 UTC+0000
..... 0×8205dda0:wuauclt.exe                          940   1084      4    126 2010-09-02 12:26:40 UTC+0000
..... 0×82311648:rundll32.exe                        3768   1084      1     53 2010-09-09 19:56:33 UTC+0000
..... 0×81eab2f8:wuauclt.exe                         3984   1084      8    325 2010-09-09 19:52:45 UTC+0000
.... 0×82095500:spoolsv.exe                          1616    744     13    140 2010-09-02 12:25:24 UTC+0000
.... 0×82089558:jqs.exe                               472    744      5    146 2010-09-02 12:25:33 UTC+0000
.... 0×822c8bf8:svchost.exe                           992    744     10    277 2010-09-02 12:25:22 UTC+0000
.... 0×82150b90:svchost.exe                           912    744     20    202 2010-09-02 12:25:22 UTC+0000
.... 0×8208abf0:sqlservr.exe                          488    744     25    306 2010-09-02 12:25:33 UTC+0000
.... 0×82077da0:coherence.exe                         572    744      4     51 2010-09-02 12:25:36 UTC+0000
.... 0×821521b0:svchost.exe                          1140    744      6     81 2010-09-02 12:25:22 UTC+0000

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```

Again, no suspicious activity is detected. So far, the process names, counts, and parent-child relationships all seem legitimate.

Investigating Network Connections

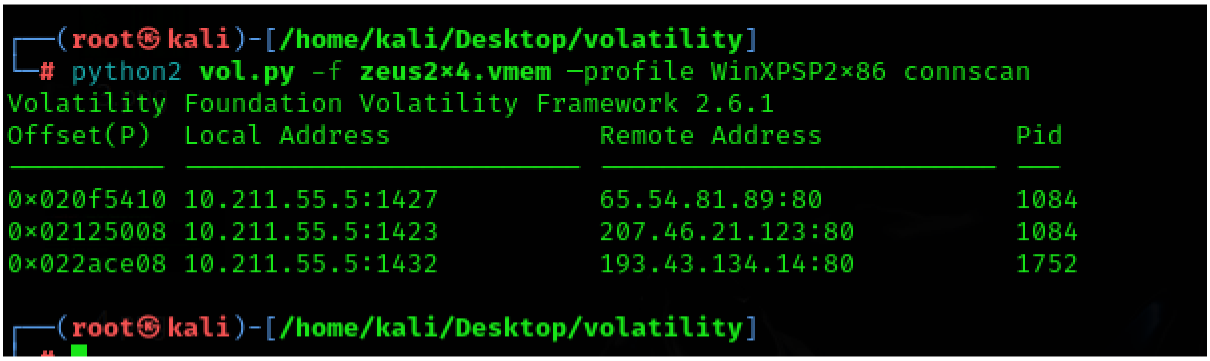
Another key area to investigate for suspicious activity is the network connections, which might reveal communication between the malicious process and its Command and Control (C2) servers.

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 connections



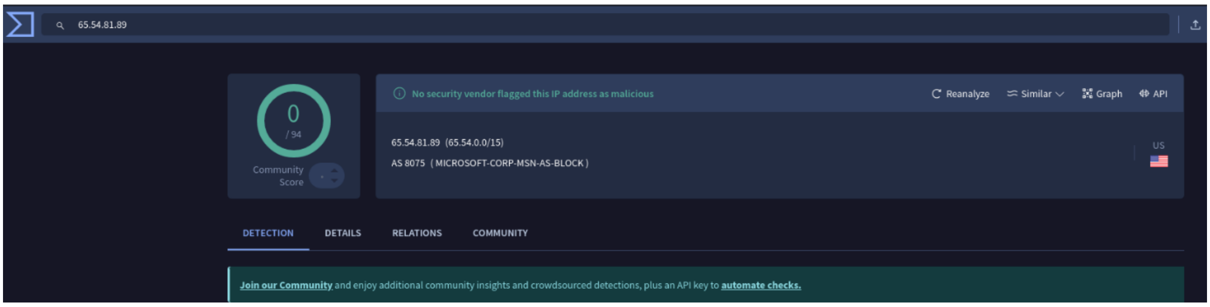
```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 connections

Volatility Foundation Volatility Framework 2.6.1
Offset(V)  Local Address             Remote Address            Pid
────────── ───────────────────────── ───────────────────────── ───
0×822ace08 10.211.55.5:1432          193.43.134.14:80          1752

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 connscan



```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 connscan
Volatility Foundation Volatility Framework 2.6.1
Offset(P)  Local Address             Remote Address            Pid
────────── ───────────────────────── ───────────────────────── ───
0×020f5410 10.211.55.5:1427          65.54.81.89:80            1084
0×02125008 10.211.55.5:1423          207.46.21.123:80          1084
0×022ace08 10.211.55.5:1432          193.43.134.14:80          1752

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
```

We can verify the reputation of the listed IP address using VirusTotal.

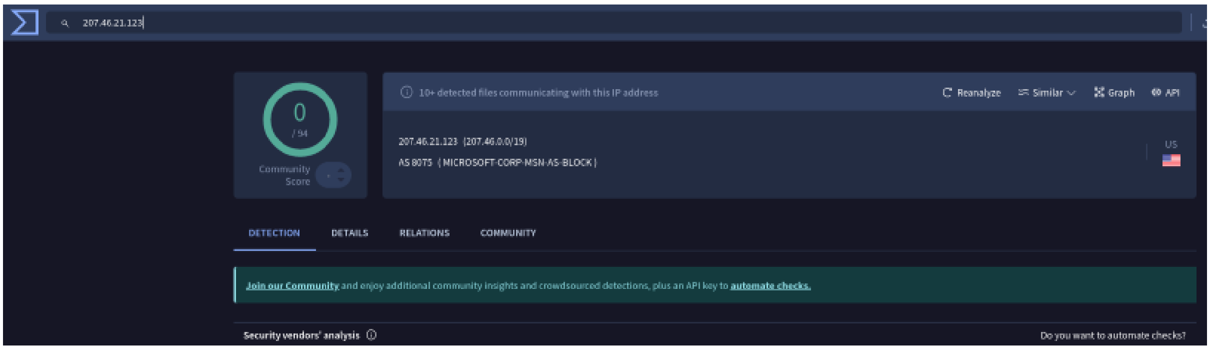


65.54.81.89

No security vendor flagged this IP address as malicious

65.54.81.89 (65.54.0.0/15)
AS 8075 (MICROSOFT-CORP-MSN-AS-BLOCK)

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

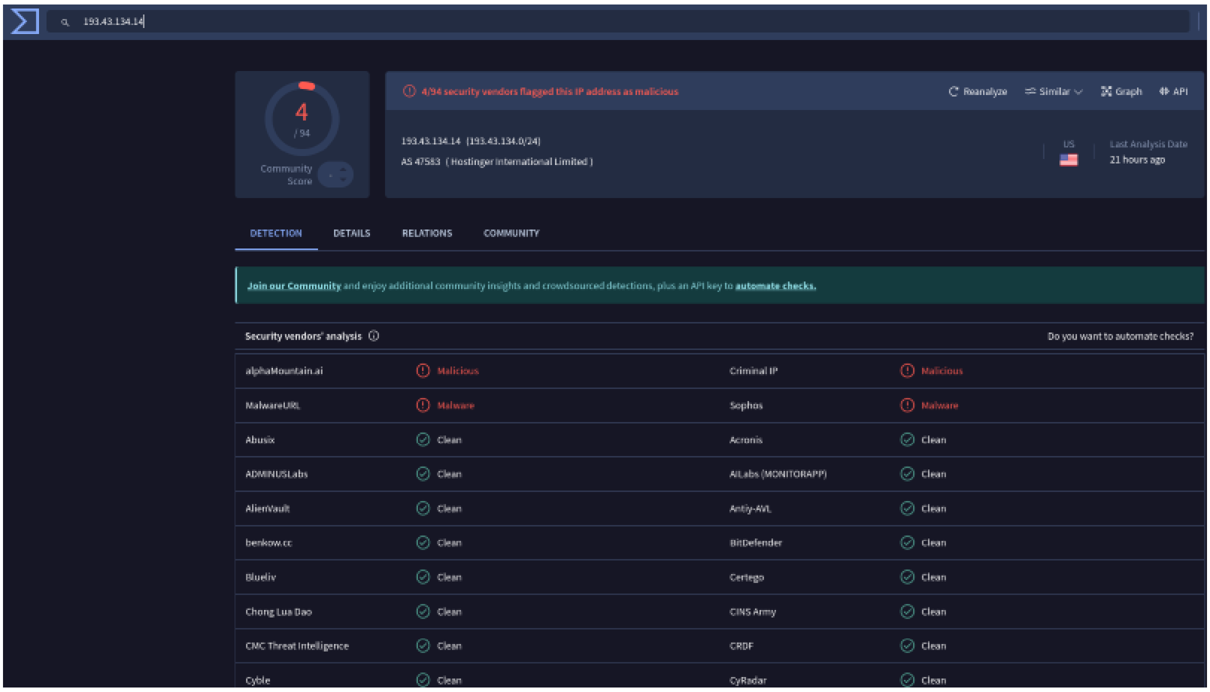


10+ detected files communicating with this IP address

207.46.21.123 (207.46.0.0/19)
AS 8075 (MICROSOFT-CORP-MSN-AS-BLOCK)

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis



4/94 security vendors flagged this IP address as malicious

193.43.134.14 (193.43.134.0/24)
AS 47583 (Hostinger International Limited)

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis

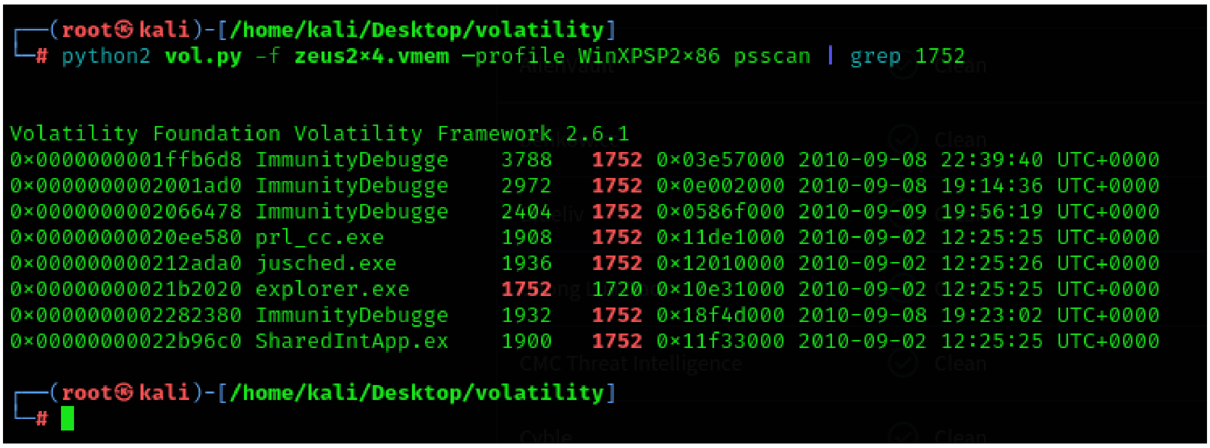
alphaMountain.ai	Malicious	Criminal IP	Malicious
MalwareURL	Malware	Sophos	Malware
Abusix	Clean	Acronis	Clean
ADMINUSLabs	Clean	AILabs (MONITORAPP)	Clean
AlienVault	Clean	Antiy-AVL	Clean
benkow.cc	Clean	BitDefender	Clean
Blueliv	Clean	Certego	Clean
Chong Lua Dao	Clean	CINS Army	Clean
CMC Threat Intelligence	Clean	CRDF	Clean
Cyble	Clean	CyRadar	Clean

The results appear suspicious, but we can’t make sure

Analyzing the Malicious Process

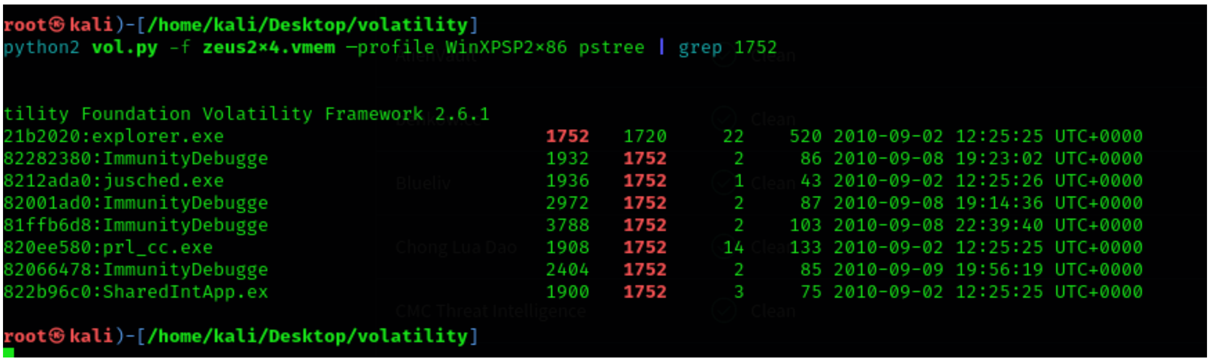
To identify the process communicating with the suspicious IP address, we can filter the output of psscan for the relevant PID:

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 psscan | grep 1752



```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 psscan | grep 1752


Volatility Foundation Volatility Framework 2.6.1
0×000000001ffb6d8 ImmunityDebugge    3788   1752 0×03e57000 2010-09-08 22:39:40 UTC+0000
0×0000000002001ad0 ImmunityDebugge    2972   1752 0×0e002000 2010-09-08 19:14:36 UTC+0000
0×0000000002066478 ImmunityDebugge    2404   1752 0×0586f000 2010-09-09 19:56:19 UTC+0000
0×00000000020ee580 prl_cc.exe         1908   1752 0×11de1000 2010-09-02 12:25:25 UTC+0000
0×000000000212ada0 jusched.exe        1936   1752 0×12010000 2010-09-02 12:25:26 UTC+0000
0×00000000021b2020 explorer.exe       1752   1720 0×10e31000 2010-09-02 12:25:25 UTC+0000
0×0000000002282380 ImmunityDebugge    1932   1752 0×18f4d000 2010-09-08 19:23:02 UTC+0000
0×00000000022b96c0 SharedIntApp.ex    1900   1752 0×11f33000 2010-09-02 12:25:25 UTC+0000

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```

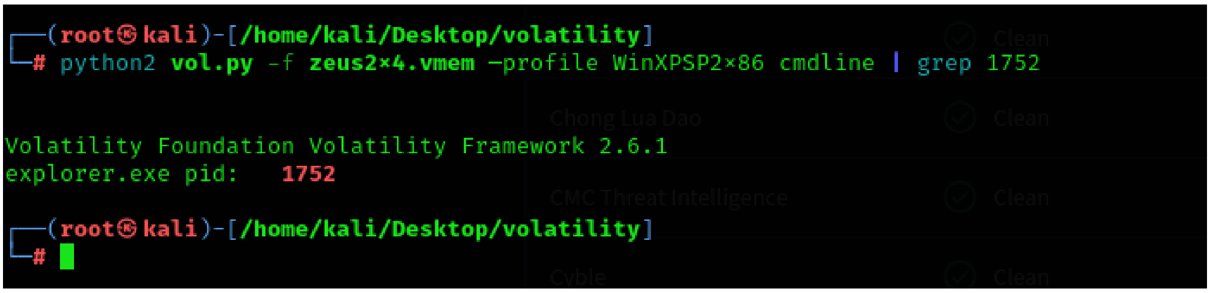
python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 pstree | grep 1752



```
root㉿kali)-[/home/kali/Desktop/volatility]
python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 pstree | grep 1752


tility Foundation Volatility Framework 2.6.1
21b2020:explorer.exe                            1752   1720     22    520 2010-09-02 12:25:25 UTC+0000
82282380:ImmunityDebugge                        1932   1752      2     86 2010-09-08 19:23:02 UTC+0000
8212ada0:jusched.exe                            1936   1752      1     43 2010-09-02 12:25:26 UTC+0000
82001ad0:ImmunityDebugge                        2972   1752      2     87 2010-09-08 19:14:36 UTC+0000
81ffb6d8:ImmunityDebugge                        3788   1752      2    103 2010-09-08 22:39:40 UTC+0000
820ee580:prl_cc.exe                             1908   1752     14    133 2010-09-02 12:25:25 UTC+0000
82066478:ImmunityDebugge                        2404   1752      2     85 2010-09-09 19:56:19 UTC+0000
822b96c0:SharedIntApp.ex                        1900   1752      3     75 2010-09-02 12:25:25 UTC+0000

root㉿kali)-[/home/kali/Desktop/volatility]
```

Next, we'll check the executable's path using the cmdline plugin:

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 cmdline | grep 1752



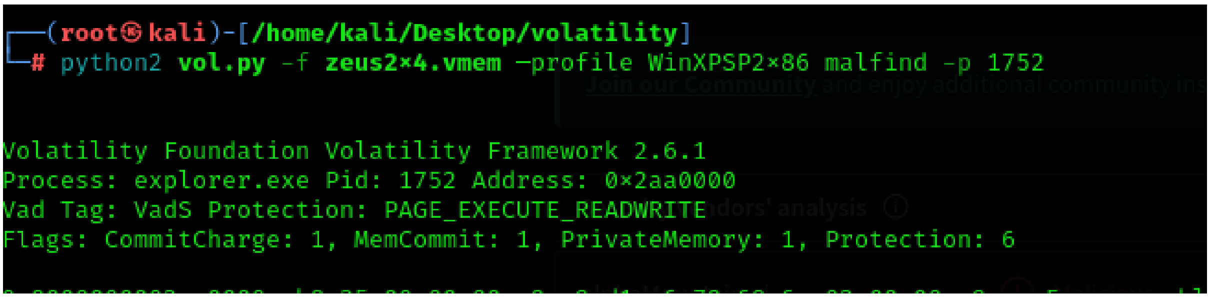
```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 cmdline | grep 1752


Volatility Foundation Volatility Framework 2.6.1
explorer.exe pid:   1752

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```

However, this doesn’t provide any useful insights, as the process path appears legitimate.

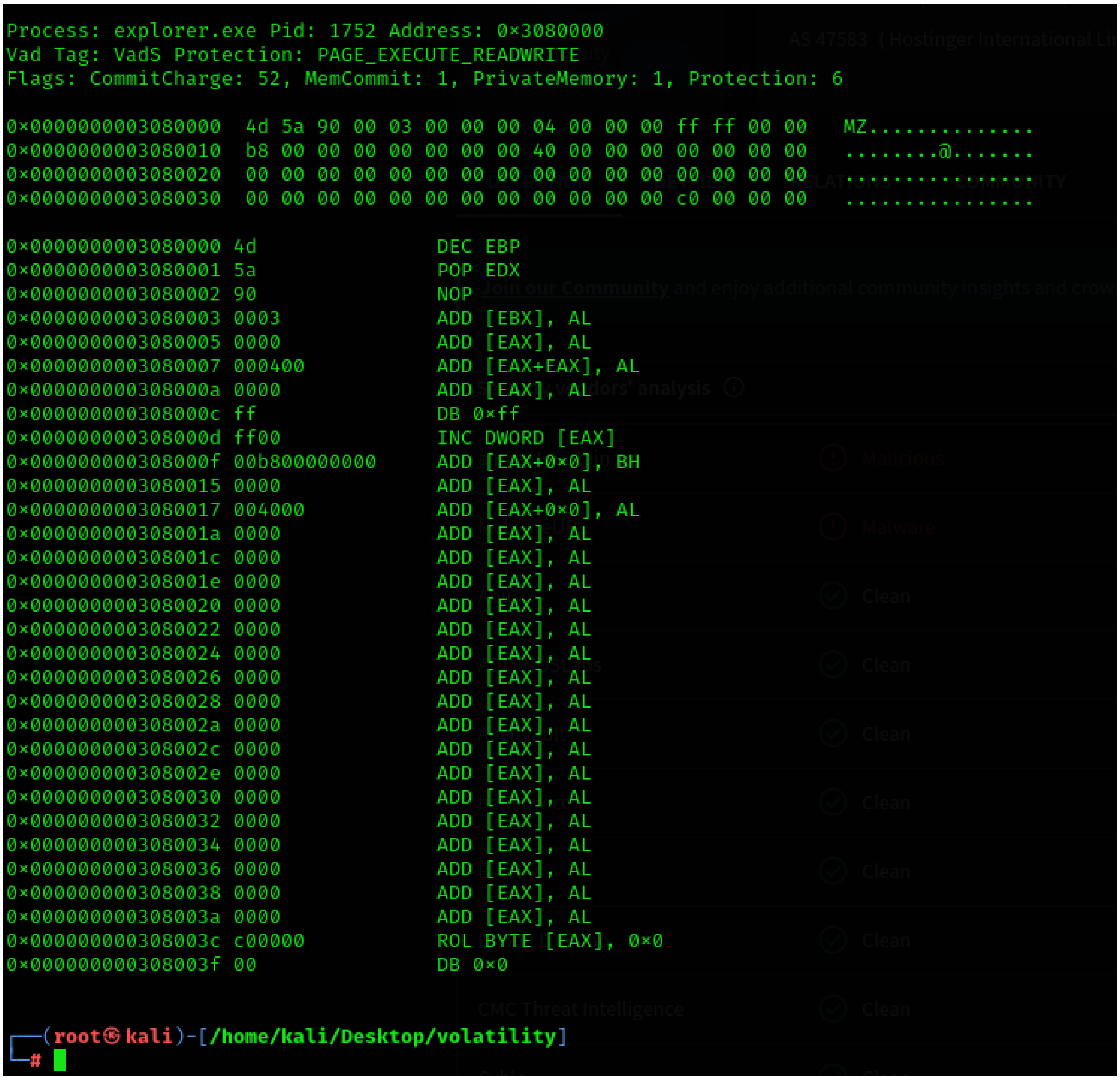
Investigating Code Injection

So far, nothing about the executable.1752.exe process appears suspicious, but it's possible that malicious code has been injected into it. To explore this possibility, we'll use the malfind plugin:

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 malfind -p 1752



```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 malfind -p 1752

Volatility Foundation Volatility Framework 2.6.1
Process: explorer.exe Pid: 1752 Address: 0x2aa0000
Vad Tag: VadS Protection: PAGE_EXECUTE_READWRITE
Flags: CommitCharge: 1, MemCommit: 1, PrivateMemory: 1, Protection: 6
```



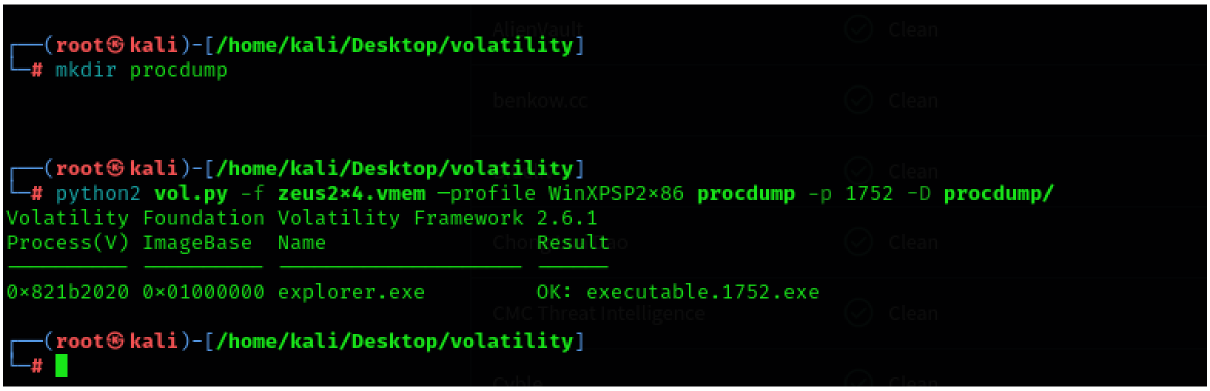
```
Process: explorer.exe Pid: 1752 Address: 0x3080000
Vad Tag: VadS Protection: PAGE_EXECUTE_READWRITE
Flags: CommitCharge: 52, MemCommit: 1, PrivateMemory: 1, Protection: 6

0x0000000003080000  4d 5a 90 00 03 00 00 00 04 00 00 00 ff ff 00 00   MZ..............
0x0000000003080010  b8 00 00 00 00 00 00 00 40 00 00 00 00 00 00 00   ........@.......
0x0000000003080020  00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
0x0000000003080030  00 00 00 00 00 00 00 00 00 00 00 00 c0 00 00 00   ................

0x0000000003080000 4d                DEC EBP
0x0000000003080001 5a                POP EDX
0x0000000003080002 90                NOP
0x0000000003080003 0003              ADD [EBX], AL
0x0000000003080005 0000              ADD [EAX], AL
0x0000000003080007 000400            ADD [EAX+EAX], AL
0x000000000308000a 0000              ADD [EAX], AL
0x000000000308000c ff                DB 0xff
0x000000000308000d ff00              INC DWORD [EAX]
0x000000000308000f 00b800000000      ADD [EAX+0x0], BH
0x0000000003080015 0000              ADD [EAX], AL
0x0000000003080017 004000            ADD [EAX+0x0], AL
0x000000000308001a 0000              ADD [EAX], AL
0x000000000308001c 0000              ADD [EAX], AL
0x000000000308001e 0000              ADD [EAX], AL
0x0000000003080020 0000              ADD [EAX], AL
0x0000000003080022 0000              ADD [EAX], AL
0x0000000003080024 0000              ADD [EAX], AL
0x0000000003080026 0000              ADD [EAX], AL
0x0000000003080028 0000              ADD [EAX], AL
0x000000000308002a 0000              ADD [EAX], AL
0x000000000308002c 0000              ADD [EAX], AL
0x000000000308002e 0000              ADD [EAX], AL
0x0000000003080030 0000              ADD [EAX], AL
0x0000000003080032 0000              ADD [EAX], AL
0x0000000003080034 0000              ADD [EAX], AL
0x0000000003080036 0000              ADD [EAX], AL
0x0000000003080038 0000              ADD [EAX], AL
0x000000000308003a 0000              ADD [EAX], AL
0x000000000308003c c00000            ROL BYTE [EAX], 0x0
0x000000000308003f 00                DB 0x0

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─#
```

The results show that the process has an MZ header and is marked with the protection PAGE_EXECUTE_READWRITE. This indicates that the memory region is both executable and writable, which is unusual and suspicious. It allows an attacker to execute code from this memory region and dynamically modify its content. Legitimate processes generally don’t have memory regions that are both executable and writable.

We’ll proceed by dumping the process to investigate further using procdump:

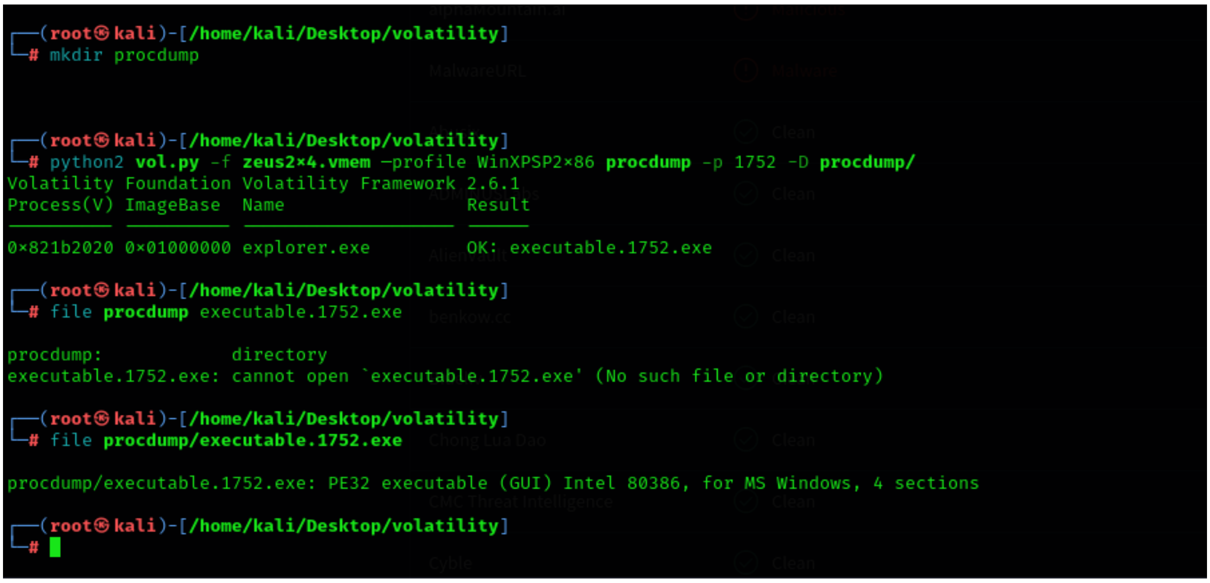
mkdir procdump

python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 procdump -p 1752 -D procdump/



```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# mkdir procdump


┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 procdump -p 1752 -D procdump/
Volatility Foundation Volatility Framework 2.6.1
Process(V) ImageBase  Name                 Result
────────── ────────── ──────────────────── ──────
0×821b2020 0×01000000 explorer.exe         OK: executable.1752.exe

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```

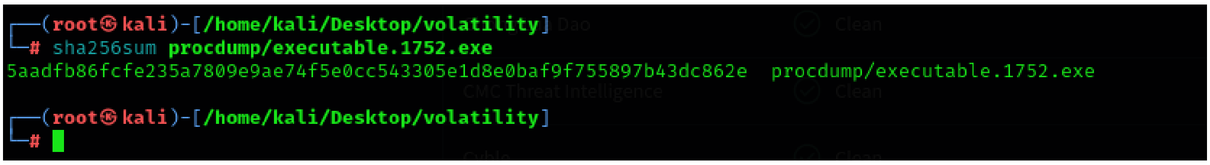
file procdump/executable.1752.exe



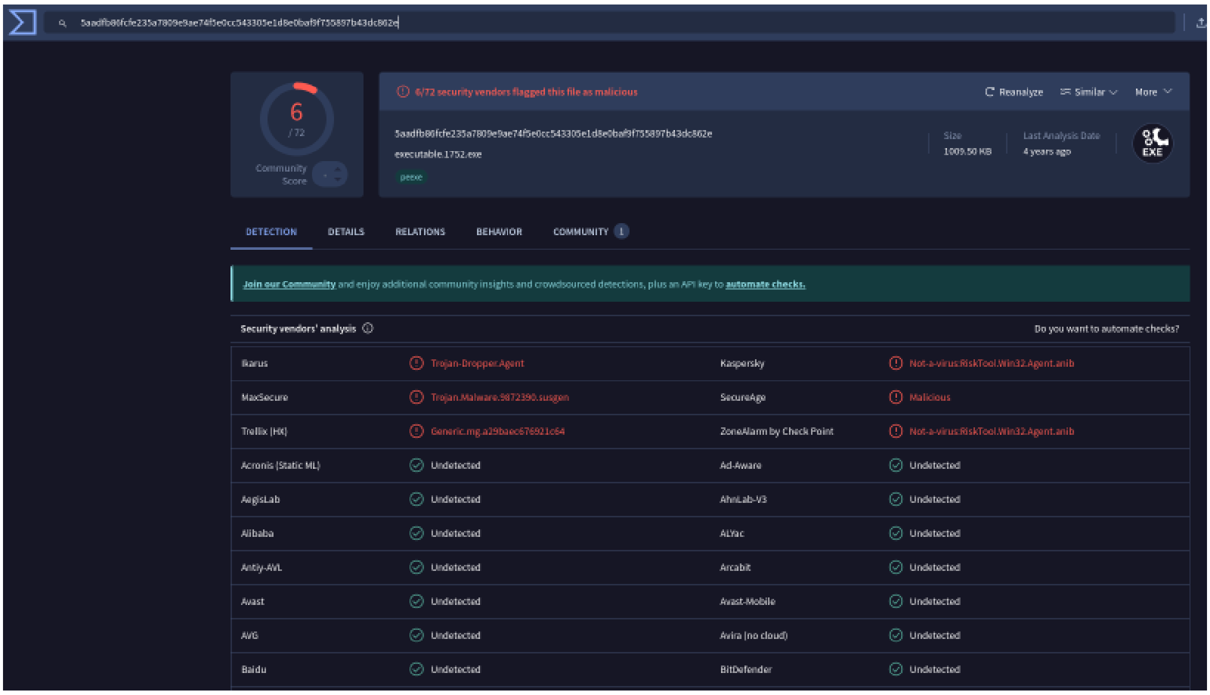
```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# mkdir procdump


┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 procdump -p 1752 -D procdump/
Volatility Foundation Volatility Framework 2.6.1
Process(V) ImageBase  Name                 Result
────────── ────────── ──────────────────── ──────
0×821b2020 0×01000000 explorer.exe         OK: executable.1752.exe

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# file procdump executable.1752.exe

procdump:             directory
executable.1752.exe: cannot open `executable.1752.exe' (No such file or directory)

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# file procdump/executable.1752.exe

procdump/executable.1752.exe: PE32 executable (GUI) Intel 80386, for MS Windows, 4 sections

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```

Then, we can compute the SHA256 hash of the dumped executable:

sha256sum procdump/executable.1752.exe



```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# sha256sum procdump/executable.1752.exe
5aadfb86fcfe235a7809e9ae74f5e0cc543305e1d8e0baf9f755897b43dc862e  procdump/executable.1752.exe

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# 
```



6/72 security vendors flagged this file as malicious

5aadfb86fcfe235a7809e9ae74f5e0cc543305e1d8e0baf9f755897b43dc862e
executable.1752.exe
peexe

Size 1009.50 KB
Last Analysis Date 4 years ago

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Security vendors' analysis

Do you want to automate checks?

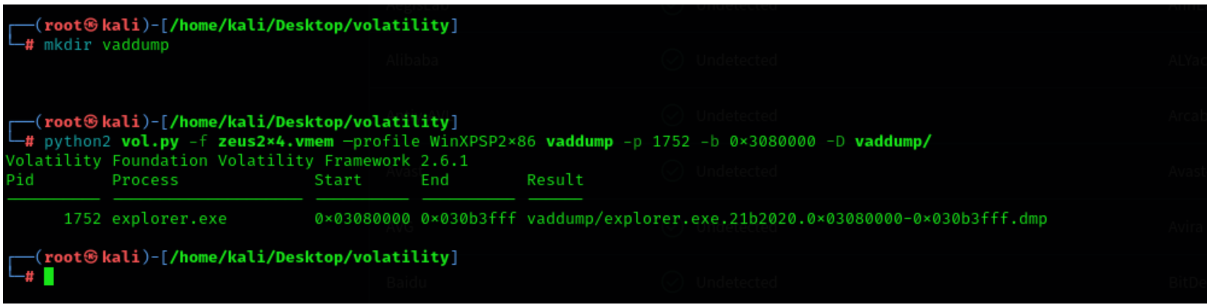
Ikarus	Trojan-Dropper.Agent	Kaspersky	Not-a-virus:RiskTool.Win32.Agent.anib
MaxSecure	Trojan.Malware.9872390.susgen	SecureAge	Malicious
Trellix (HX)	Generic.mg.a29baec676921c64	ZoneAlarm by Check Point	Not-a-virus:RiskTool.Win32.Agent.anib
Acronis (Static ML)	Undetected	Ad-Aware	Undetected
AegisLab	Undetected	AhnLab-V3	Undetected
Alibaba	Undetected	ALYac	Undetected
Antiy-AVL	Undetected	Arcabit	Undetected
Avast	Undetected	Avast-Mobile	Undetected
AVG	Undetected	Avira (no cloud)	Undetected
Baidu	Undetected	BitDefender	Undetected

Surprisingly, the file is not flagged as malicious. However, we shouldn't draw conclusions based on this alone, given the suspicious network activity and potential code injection.

Next, we will dump the specific memory region where we suspect the malicious code has been injected, using the vaddump plugin. We'll provide the PID and the base address (which we obtained from the malfind output):

mkdir vaddump

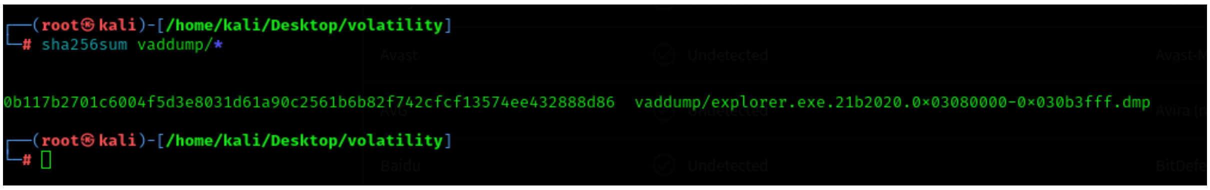
python2 vol.py -f zeus2x4.vmem —profile WinXPSP2x86 vaddump -p 1752 -b 0x3080000 -D vaddump/



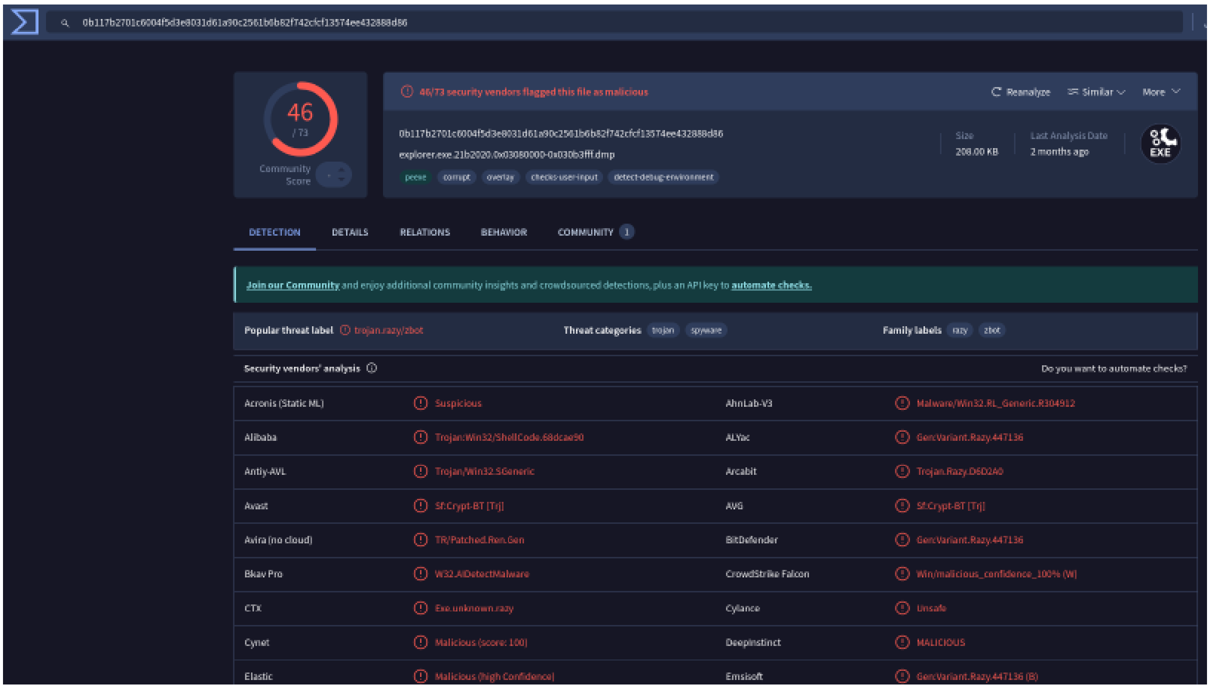
```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# mkdir vaddump


┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# python2 vol.py -f zeus2×4.vmem —profile WinXPSP2×86 vaddump -p 1752 -b 0×3080000 -D vaddump/
Volatility Foundation Volatility Framework 2.6.1
Pid         Process                    Start      End        Result
----------  -------------------------  ---------- ---------- ------
      1752  explorer.exe               0×03080000 0×030b3fff vaddump/explorer.exe.21b2020.0×03080000-0×030b3fff.dmp

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# █
```

Finally, let's check the SHA256 hash of the dumped memory region:

sha256sum vaddump/



```
┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# sha256sum vaddump/*

0b117b2701c6004f5d3e8031d61a90c2561b6b82f742cfcf13574ee432888d86  vaddump/explorer.exe.21b2020.0×03080000-0×030b3fff.dmp

┌──(root㉿kali)-[/home/kali/Desktop/volatility]
└─# ▯
```



0b117b2701c6004f5d3e8031d61a90c2561b6b82f742cfcf13574ee432888d86

46 / 73 Community Score

46/73 security vendors flagged this file as malicious

0b117b2701c6004f5d3e8031d61a90c2561b6b82f742cfcf13574ee432888d86
explorer.exe.21b2020.0x03080000-0x030b3fff.dmp

peexe corrupt overlay checks-user-input detect-debug-environment

Size 208.00 KB

Last Analysis Date 2 months ago

Join our Community and enjoy additional community insights and crowdsourced detections, plus an API key to automate checks.

Popular threat label trojan.razy/zbot   Threat categories trojan spyware   Family labels razy zbot

Security vendors' analysis   Do you want to automate checks?

Acronis (Static ML)	Suspicious	AhnLab-V3	Malware/Win32.RL_Generic.R304912
Alibaba	Trojan:Win32/ShellCode.68dcae90	ALYac	Gen:Variant.Razy.447136
Antiy-AVL	Trojan/Win32.SGeneric	Arcabit	Trojan.Razy.D6D2A0
Avast	Sf:Crypt-BT [Trj]	AVG	Sf:Crypt-BT [Trj]
Avira (no cloud)	TR/Patched.Ren.Gen	BitDefender	Gen:Variant.Razy.447136
Bkav Pro	W32.AIDetectMalware	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
CTX	Exe.unknown.razy	Cylance	Unsafe
Cynet	Malicious (score: 100)	DeepInstinct	MALICIOUS
Elastic	Malicious (high Confidence)	Emsisoft	Gen:Variant.Razy.447136 (B)

As expected, this memory region is indeed malicious, confirming that code injection has occurred.

At this point, we have a process with injected malicious code that is also communicating with a flagged IP address.

Findings

1. Active and Injected Processes The process 1752 was identified as suspicious based on its behavior and parent-child relationships in the process tree. Memory indicators confirmed injection techniques consistent with Zeus.

2. Network Connections Evidence of connections to external IPs associated with command-and-control (C2) servers, characteristic of Zeus activity. Network artifacts included traces of encrypted communication.

3. Malware Artifacts Dumped executable and memory regions contain signatures and characteristics aligning with Zeus variants. Hash analysis can facilitate further correlation with known malware databases.