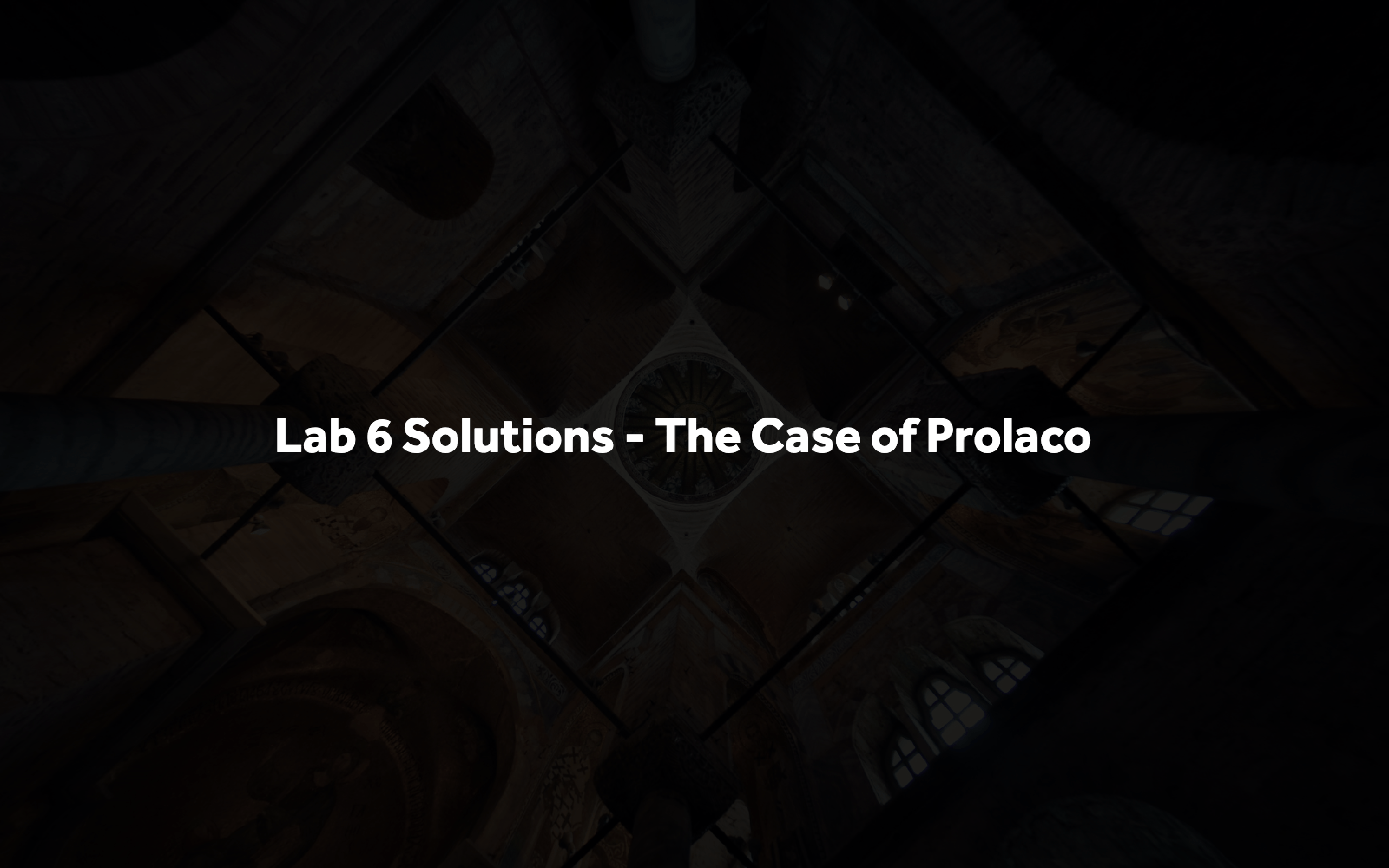
Lab 6 Solutions - The Case of Prolaco



Lab 6 - The Case of Prolaco

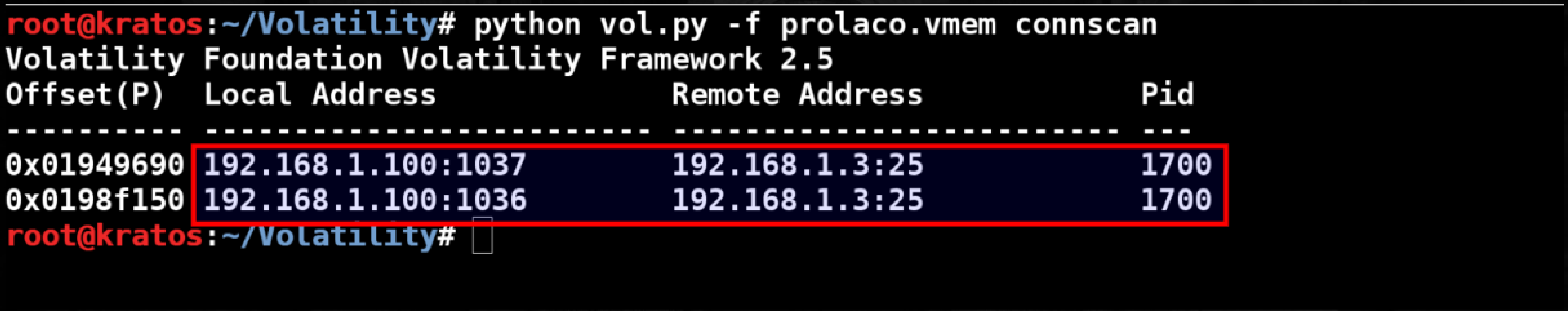
While monitoring for security events, you determine that one of the host (192.168.1.100) is sending a spam email, you suspect this host to be infected with a spambot, you collect the memory image (prolaco.vmem) from the host. use memory image prolaco.vmem to answer below questions:

- Does the network connection show any indication of host sending the spam?
- Can you determine the malicious process id that is responsible for spam activity?
- Can you determine the name of the malicious process that is sending the spam?
- Can you dump the malicious process from the memory?
- Can you confirm, if the dumped process is malicious?
- Can you determine any unique indicator associated with this malware?
- Is there any other process that is related to the malicious process?

Answers

01. Does the network connection show any indication of the host sending the spam?

Running the connscan plugin shows a process with process id ***1700*** communicating on the SMTP port ***25***.



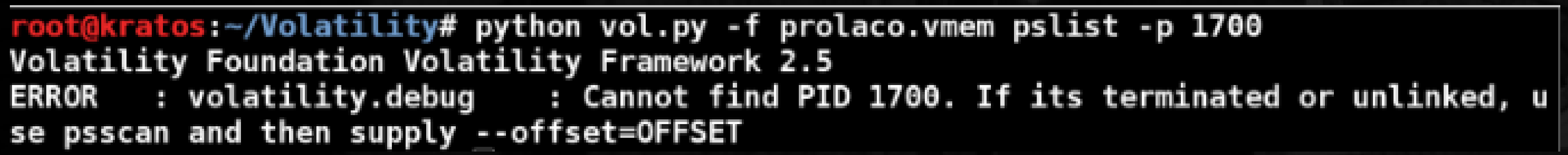
```
root@kratos:~/Volatility# python vol.py -f prolaco.vmem connscan
Volatility Foundation Volatility Framework 2.5
Offset(P)  Local Address             Remote Address            Pid
---------- ------------------------- ------------------------- ---
0x01949690 192.168.1.100:1037        192.168.1.3:25            1700
0x0198f150 192.168.1.100:1036        192.168.1.3:25            1700
root@kratos:~/Volatility# 
```

02. Can you determine the malicious process id that is responsible for spam activity?

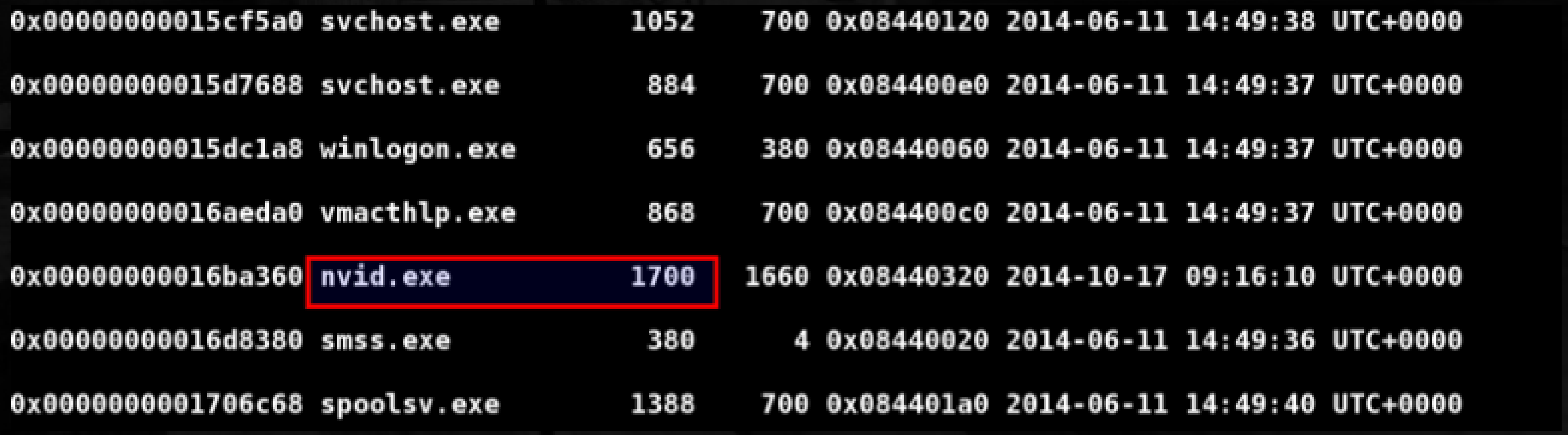
Process id responsible for sending the spam activity is ***1700*** as shown in the screenshot

03. Can you determine the name of the malicious process that is sending the spam?

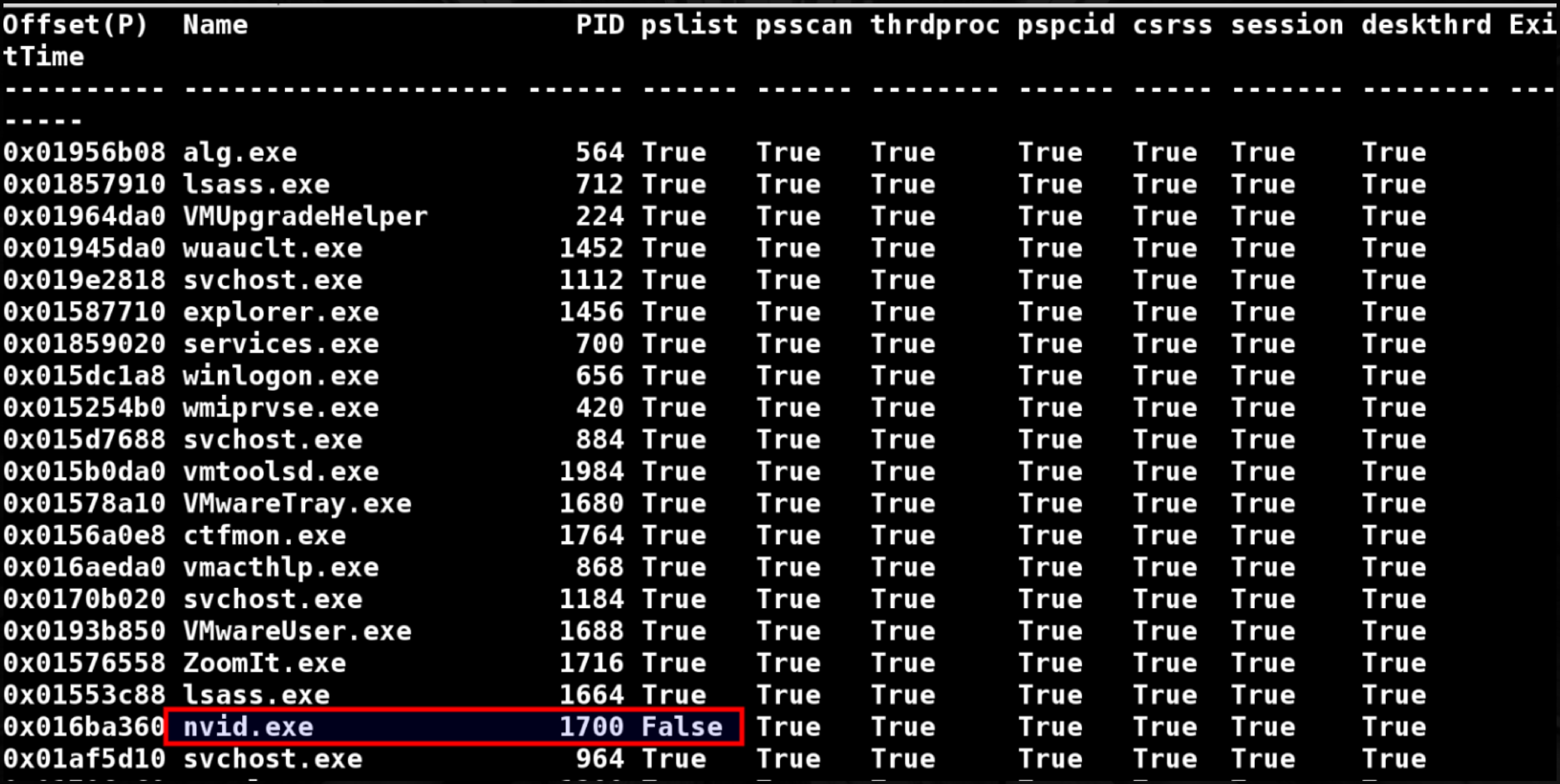
The name of the malicious process is "***nvid.exe***", the ***pslist*** plugin does not show the presence of the process with pid ***1700***, whereas ***psscan*** and ***psxview*** plugin shows its presence indicating that the attackers unlinked this process from the double-linked list used by the operating system to keep track of active processes.



```
root@kratos:~/Volatility# python vol.py -f prolaco.vmem pslist -p 1700
Volatility Foundation Volatility Framework 2.5
ERROR   : volatility.debug    : Cannot find PID 1700. If its terminated or unlinked, u
se psscan and then supply --offset=OFFSET
```



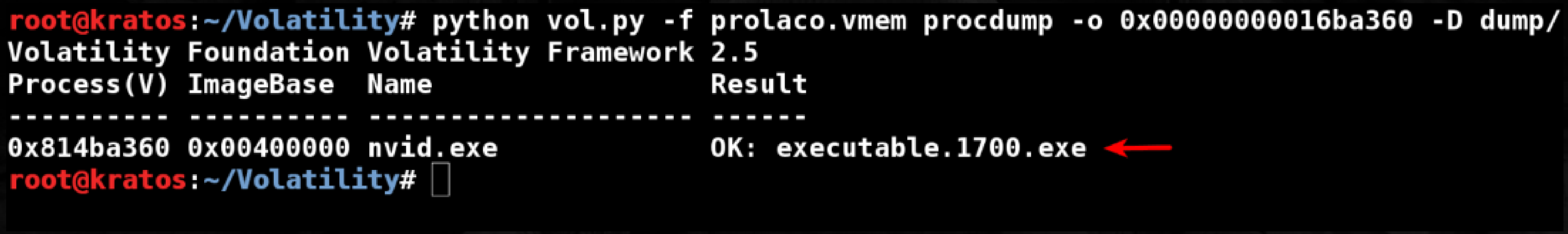
```
0x00000000015cf5a0 svchost.exe        1052    700 0x08440120 2014-06-11 14:49:38 UTC+0000
0x00000000015d7688 svchost.exe         884    700 0x084400e0 2014-06-11 14:49:37 UTC+0000
0x00000000015dc1a8 winlogon.exe        656    380 0x08440060 2014-06-11 14:49:37 UTC+0000
0x00000000016aeda0 vmacthlp.exe        868    700 0x084400c0 2014-06-11 14:49:37 UTC+0000
0x00000000016ba360 nvid.exe           1700   1660 0x08440320 2014-10-17 09:16:10 UTC+0000
0x00000000016d8380 smss.exe            380      4 0x08440020 2014-06-11 14:49:36 UTC+0000
0x0000000001706c68 spoolsv.exe        1388    700 0x084401a0 2014-06-11 14:49:40 UTC+0000
```



```
Offset(P)  Name                    PID pslist psscan thrdproc pspcid csrss session deskthrd Exi
tTime
---------- -------------------- ------ ------ ------ -------- ------ ----- ------- -------- ---
-----
0x01956b08 alg.exe                 564 True   True   True     True   True  True    True
0x01857910 lsass.exe               712 True   True   True     True   True  True    True
0x01964da0 VMUpgradeHelper         224 True   True   True     True   True  True    True
0x01945da0 wuauclt.exe            1452 True   True   True     True   True  True    True
0x019e2818 svchost.exe            1112 True   True   True     True   True  True    True
0x01587710 explorer.exe           1456 True   True   True     True   True  True    True
0x01859020 services.exe            700 True   True   True     True   True  True    True
0x015dc1a8 winlogon.exe            656 True   True   True     True   True  True    True
0x015254b0 wmiprvse.exe            420 True   True   True     True   True  True    True
0x015d7688 svchost.exe             884 True   True   True     True   True  True    True
0x015b0da0 vmtoolsd.exe           1984 True   True   True     True   True  True    True
0x01578a10 VMwareTray.exe         1680 True   True   True     True   True  True    True
0x0156a0e8 ctfmon.exe             1764 True   True   True     True   True  True    True
0x016aeda0 vmacthlp.exe            868 True   True   True     True   True  True    True
0x0170b020 svchost.exe            1184 True   True   True     True   True  True    True
0x0193b850 VMwareUser.exe         1688 True   True   True     True   True  True    True
0x01576558 ZoomIt.exe             1716 True   True   True     True   True  True    True
0x01553c88 lsass.exe              1664 True   True   True     True   True  True    True
0x016ba360 nvid.exe               1700 False  True   True     True   True  True    True
0x01af5d10 svchost.exe             964 True   True   True     True   True  True    True
```

04. Can you dump the malicious process from the memory?

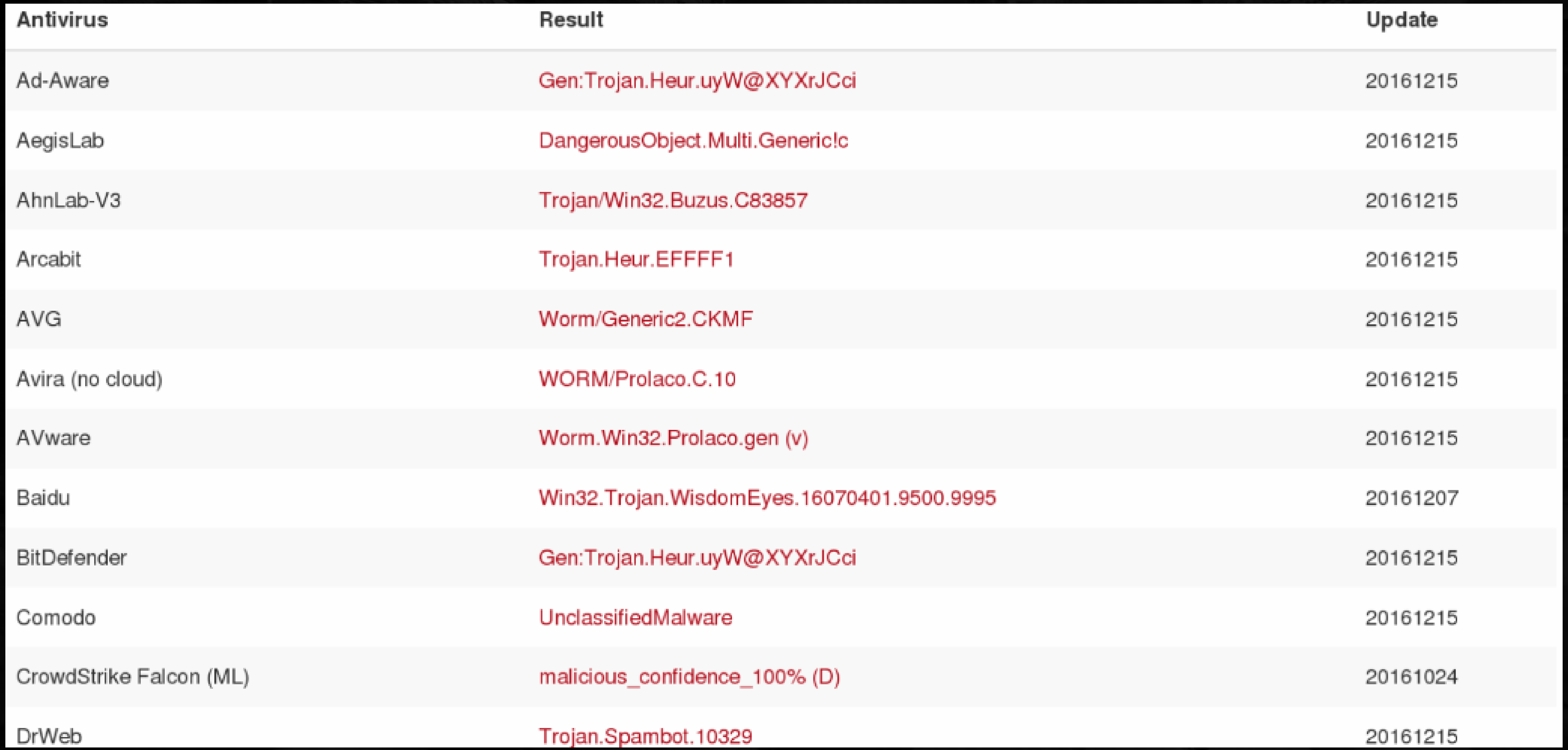
The malicious process cannot be dumped by giving the ***-p*** option to the ***procdump*** plugin as this process is hidden. To dump the malicious process we can use the physical offset (determined from the ***psscan*** or ***psxview*** output) and then use the ***-o*** option as shown in the below screenshot.



```
root@kratos:~/Volatility# python vol.py -f prolaco.vmem procdump -o 0x00000000016ba360 -D dump/
Volatility Foundation Volatility Framework 2.5
Process(V) ImageBase  Name                 Result
---------- ---------- -------------------- ------
0x814ba360 0x00400000 nvid.exe             OK: executable.1700.exe
root@kratos:~/Volatility# 
```

05. Can you confirm, if the dumped process is malicious?

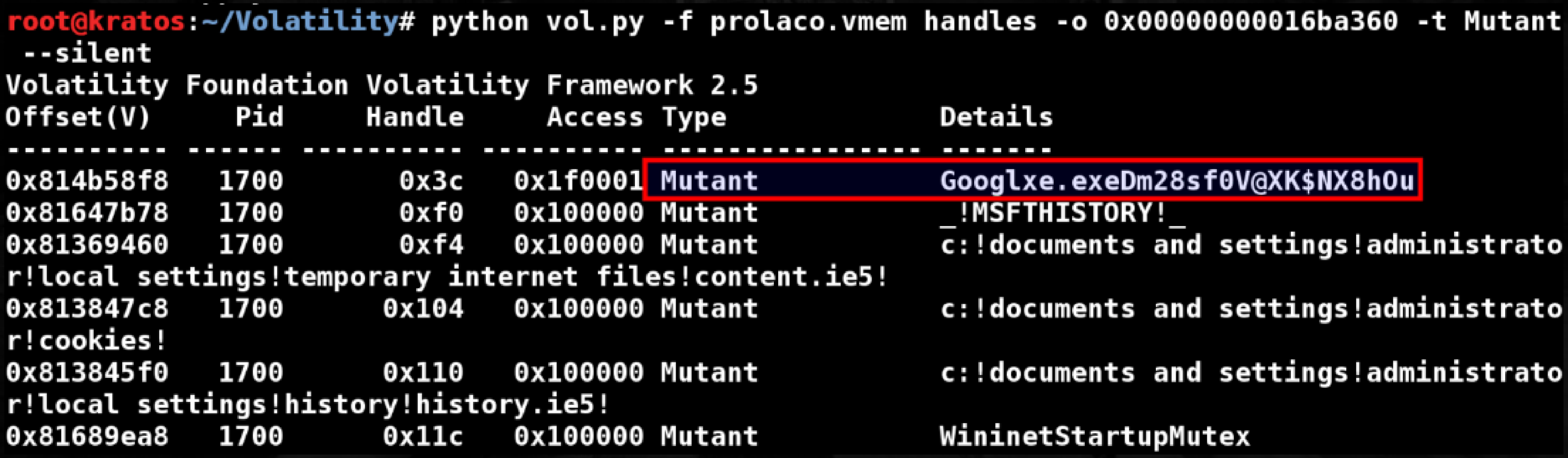
Submitting the dumped process to ***VirusTotal*** confirms it to be malicious as shown in the screenshot



Antivirus	Result	Update
Ad-Aware	Gen:Trojan.Heur.uyW@XYXrJCci	20161215
AegisLab	DangerousObject.Multi.Generic!c	20161215
AhnLab-V3	Trojan/Win32.Buzus.C83857	20161215
Arcabit	Trojan.Heur.EFFFF1	20161215
AVG	Worm/Generic2.CKMF	20161215
Avira (no cloud)	WORM/Prolaco.C.10	20161215
AVware	Worm.Win32.Prolaco.gen (v)	20161215
Baidu	Win32.Trojan.WisdomEyes.16070401.9500.9995	20161207
BitDefender	Gen:Trojan.Heur.uyW@XYXrJCci	20161215
Comodo	UnclassifiedMalware	20161215
CrowdStrike Falcon (ML)	malicious_confidence_100% (D)	20161024
DrWeb	Trojan.Spambot.10329	20161215

06. Can you determine any unique indicator associated with this malware?

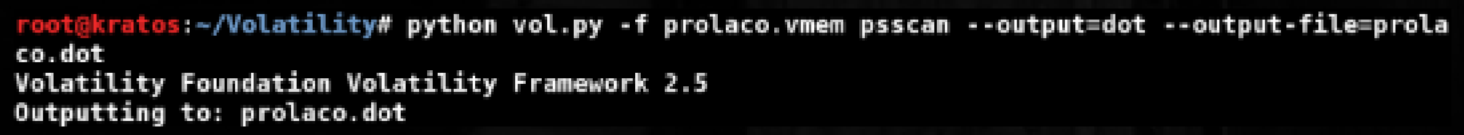
Inspecting the handles of the malicious process using its physical offset shows a Mutex created by the malware. This can be used as a unique indicator.



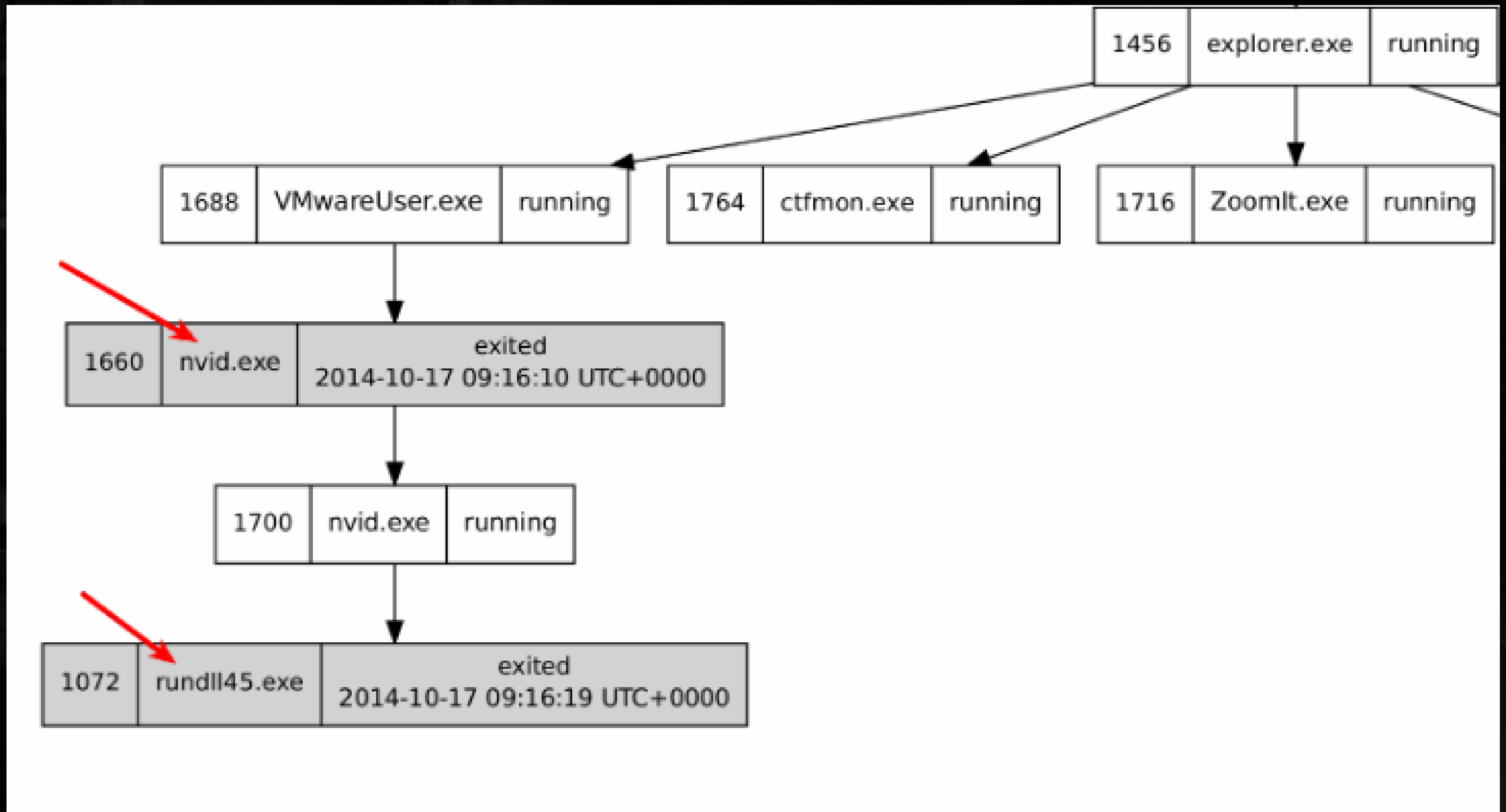
```
root@kratos:~/Volatility# python vol.py -f prolaco.vmem handles -o 0x00000000016ba360 -t Mutant
 --silent
Volatility Foundation Volatility Framework 2.5
Offset(V)     Pid     Handle     Access Type             Details
---------- ------ ---------- ---------- ---------------- -------
0x814b58f8   1700       0x3c   0x1f0001 Mutant           Googlxe.exeDm28sf0V@XK$NX8hOu
0x81647b78   1700       0xf0   0x100000 Mutant           _!MSFTHISTORY!_
0x81369460   1700       0xf4   0x100000 Mutant           c:!documents and settings!administrato
r!local settings!temporary internet files!content.ie5!
0x813847c8   1700      0x104   0x100000 Mutant           c:!documents and settings!administrato
r!cookies!
0x813845f0   1700      0x110   0x100000 Mutant           c:!documents and settings!administrato
r!local settings!history!history.ie5!
0x81689ea8   1700      0x11c   0x100000 Mutant           WininetStartupMutex
```

07. Is there any other process that is related to the malicious process?

psscan plugin can be used to get the parent-child relationship; this can be done by dumping it in dot format and opening it a dot viewer. From the below screenshot it can be seen that malicious process ***nvid.exe*** (***pid 1700***) was created by a process ***nvid.exe*** (***pid 1660***) and malicious process ***nvid.exe*** (***1700***) in turn created the rundll45.exe process



```
root@kratos:~/Volatility# python vol.py -f prolaco.vmem psscan --output=dot --output-file=prola
co.dot
Volatility Foundation Volatility Framework 2.5
Outputting to: prolaco.dot
```


1456 | explorer.exe | running
1688 | VMwareUser.exe | running
1764 | ctfmon.exe | running
1716 | ZoomIt.exe | running
1660 | nvid.exe | exited 2014-10-17 09:16:10 UTC+0000
1700 | nvid.exe | running
1072 | rundll45.exe | exited 2014-10-17 09:16:19 UTC+0000