Forensics LAB

LAB 10

Josiah Smythe

April 22, 2019

Dr. Lehrfeld -Information Security and Assurance

East Tennessee State University

1. Purpose

This Lab was completed to demonstrate tools and procedures that allow detailed forensics investigation of the most volatile data on a computer system. This process was implemented to show how a real-life situation concerning computer forensics could be implemented.

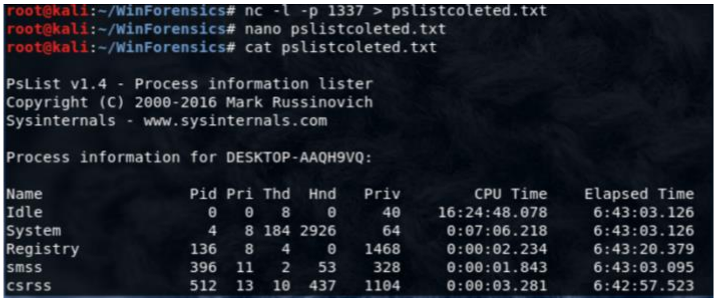
2. Materials

This lab was completed within the three virtual machine environments Kali Linux Rolling, Windows 7 and Windows 10 Consumer. These of three systems were running in the virtual machine application VMware Fusion Professional, Version 11.0.2. VMware Fusion was being supported by the operation system Mac OS Mojave on a MacBook Pro 2015 15" with 16 GB of DDR3 Ram and an Intel i7-4770HQ running 2.2GHz. The research for this project was completed according to the Acquire Live Memory - Lab.dox instructions provided by the instructor and was also supported by personal class notes.

3. Procedures and Results

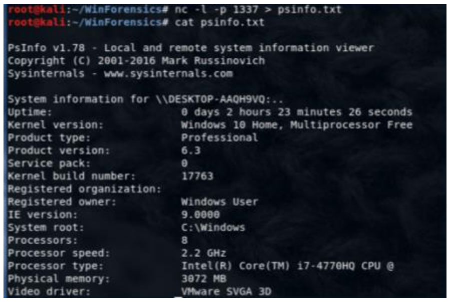
This process was begun by launching the Windows 10 and Kali Linux virtual systems running on VMware Fusion. After launching both machines the connectivity between the two systems was ensured by sending Ping requests from each system to the other. The Kali Linux system ipaddress is 192.168.241.149 and the ipaddress for the Windows system is 192.168.241.136. When the connection had been confirmed a new directory was added to the root of the Windows system named "forensics". Now in that folder the packages pstools from Microsoft, Dumpit from the D2L Dropbox and netcat from the web address https://eternallybored.org/misc/netcat/.

With these files downloaded the Kali Linux system was navigated to for the purpose of creating a new directory called "WinForensics". This directory was created via the command line to store all of the gathered text files that will be collected during the Kali Linux system forensics investigation. Now within the WinForensics directory, a host of commands were executed to listen for information being sent from the Windows 10 Machine. The first command is shown in image #1 and was implemented on the Kali Linux machine to create a listener that saves the collected information to the file pslistcoleted.txt. This was done by utilizing the command line application netcat. Similarly, the command in image #2 was executed to create another listener for the psinfo command.



```
root@kali:~/WinForensics# nc -l -p 1337 > pslistcoleted.txt
root@kali:~/WinForensics# nano pslistcoleted.txt
root@kali:~/WinForensics# cat pslistcoleted.txt

PsList v1.4 - Process information lister
Copyright (C) 2000-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

Process information for DESKTOP-AAQH9VQ:

Name                Pid Pri Thd  Hnd   Priv        CPU Time    Elapsed Time
Idle                  0   0   8    0     40    16:24:48.078     6:43:03.126
System                4   8 184 2926     64     0:07:06.218     6:43:03.126
Registry            136   8   4    0   1468     0:00:02.234     6:43:20.379
smss                396  11   2   53    328     0:00:01.843     6:43:03.095
csrss               512  13  10  437   1104     0:00:03.281     6:42:57.523
```

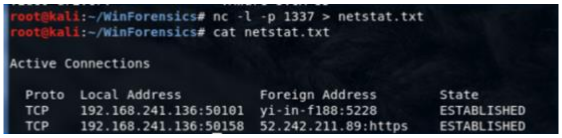
pslist listener Kali, image #1



```
root@kali:~/WinForensics# nc -l -p 1337 > psinfo.txt
root@kali:~/WinForensics# cat psinfo.txt

PsInfo v1.78 - Local and remote system information viewer
Copyright (C) 2001-2016 Mark Russinovich
Sysinternals - www.sysinternals.com

System information for \\DESKTOP-AAQH9VQ:..
Uptime:                    0 days 2 hours 23 minutes 26 seconds
Kernel version:            Windows 10 Home, Multiprocessor Free
Product type:              Professional
Product version:           6.3
Service pack:              0
Kernel build number:       17763
Registered organization:
Registered owner:          Windows User
IE version:                9.0000
System root:               C:\Windows
Processors:                8
Processor speed:           2.2 GHz
Processor type:            Intel(R) Core(TM) i7-4770HQ CPU @
Physical memory:           3072 MB
Video driver:              VMware SVGA 3D
```

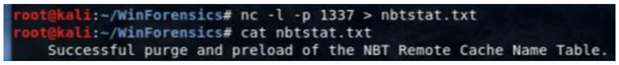
psinfo listener Kali, image #2

These two listeners were waiting for information to be sent on through port 1337 to collect. That information was sent from the Windows 10 system by executing the command “pslist.exe | nc.exe 192.168.241.149 1337 -w 3” in the administrative command line. This process of commands was done one at a time as in the listener was opened and then the corresponding Windows sender command was implemented. This process was continued for the collection commands netstat.exe, nbtstat.exe, and ipconfig /all. The results that were gathered on Kali Linux are shown in the screenshots below.



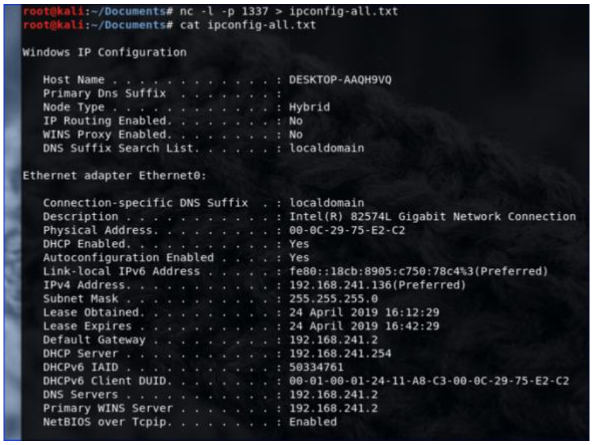
```
root@kali:~/WinForensics# nc -l -p 1337 > netstat.txt
root@kali:~/WinForensics# cat netstat.txt

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    192.168.241.136:50101  yi-in-f188:5228        ESTABLISHED
  TCP    192.168.241.136:50158  52.242.211.89:https    ESTABLISHED
```

Kali Linux listener netstat, image #3



```
root@kali:~/WinForensics# nc -l -p 1337 > nbtstat.txt
root@kali:~/WinForensics# cat nbtstat.txt
    Successful purge and preload of the NBT Remote Cache Name Table.
```

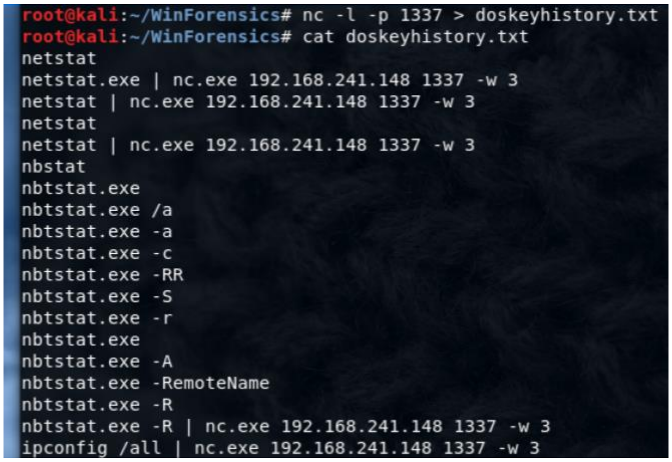
Kali Linux listener nbtstat, image #4



```
root@kali:~/Documents# nc -l -p 1337 > ipconfig-all.txt
root@kali:~/Documents# cat ipconfig-all.txt

Windows IP Configuration

   Host Name . . . . . . . . . . . . : DESKTOP-AAQH9VQ
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : localdomain

Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . : localdomain
   Description . . . . . . . . . . . : Intel(R) 82574L Gigabit Network Connection
   Physical Address. . . . . . . . . : 00-0C-29-75-E2-C2
   DHCP Enabled. . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::18cb:8905:c750:78c4%3(Preferred)
   IPv4 Address. . . . . . . . . . . : 192.168.241.136(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Lease Obtained. . . . . . . . . . : 24 April 2019 16:12:29
   Lease Expires . . . . . . . . . . : 24 April 2019 16:42:29
   Default Gateway . . . . . . . . . : 192.168.241.2
   DHCP Server . . . . . . . . . . . : 192.168.241.254
   DHCPv6 IAID . . . . . . . . . . . : 50334761
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-24-11-A8-C3-00-0C-29-75-E2-C2
   DNS Servers . . . . . . . . . . . : 192.168.241.2
   Primary WINS Server . . . . . . . : 192.168.241.2
   NetBIOS over Tcpip. . . . . . . . : Enabled
```

Kali Linux listener ipconfig /all, image #5

After storing all this information on the Kali system, a final text-based data collection was performed. This differed from the previous commands because it collected keystrokes by utilizing the doskey executable file. This was implemented in the same way as the previous steps and the storing of the information is shown in image #6.

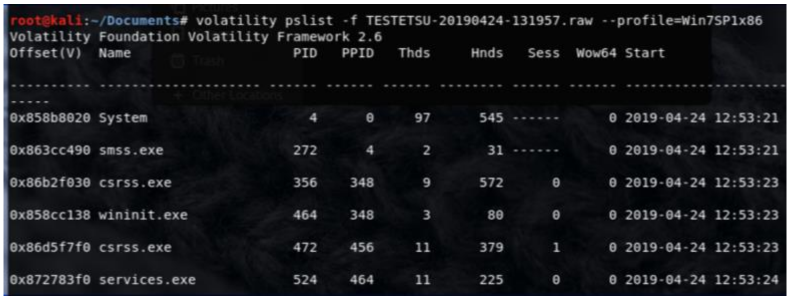


```
root@kali:~/WinForensics# nc -l -p 1337 > doskeyhistory.txt
root@kali:~/WinForensics# cat doskeyhistory.txt
netstat
netstat.exe | nc.exe 192.168.241.148 1337 -w 3
netstat | nc.exe 192.168.241.148 1337 -w 3
netstat
netstat | nc.exe 192.168.241.148 1337 -w 3
nbstat
nbtstat.exe
nbtstat.exe /a
nbtstat.exe -a
nbtstat.exe -c
nbtstat.exe -RR
nbtstat.exe -S
nbtstat.exe -r
nbtstat.exe
nbtstat.exe -A
nbtstat.exe -RemoteName
nbtstat.exe -R
nbtstat.exe -R | nc.exe 192.168.241.148 1337 -w 3
ipconfig /all | nc.exe 192.168.241.148 1337 -w 3
```

Kali Linux listener/result doskey, image #5

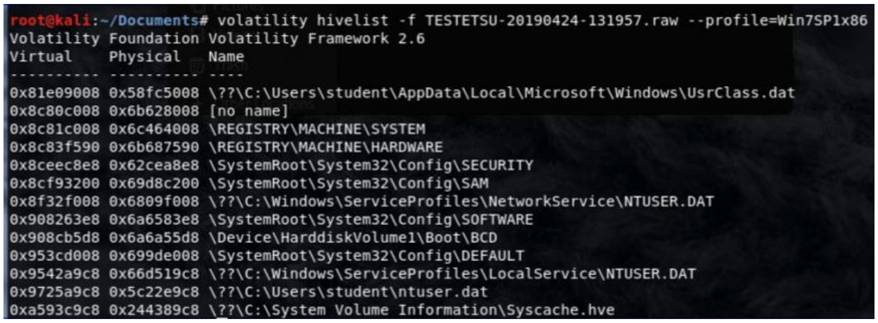
With these processes of collecting the basic volatile information from this system, the next step was to collect an image of the system ram for forensic investigation. This was done by using Dumpit and creating a raw image of the current system ram from the Windows 7 command line. This portion of the lab examination was done in the Windows 7 system instead of the Windows 10 system because the configuration of the forensics tool volatility didn’t support the profile needed for analysis the ram file. The file was created by issuing the simple command “Dumpit.exe” from the C:/ directory.

After this file was created it was dragged from the Windows 7 system to the documents folder in the Kali Linux system for investigation. The tool used for this investigation of the ram was Volatility and was controlled from the terminal on the Linux box. The initial command that was used against the ram image was “volatility pslist -f TESTETSU-20190424-131957.raw--profile=Win7SP1x86”. This command parsed ram file to show the process list. The results of this first command are shown by image #6 below. For this portion of the lab, five different commands were executed against the ram image including the first pslist. The additional four commands were, “havelist”, “svcscan”, “hasdump” and “shellbags”. Each of these commands can be seen in an image below each is labeled accordingly.



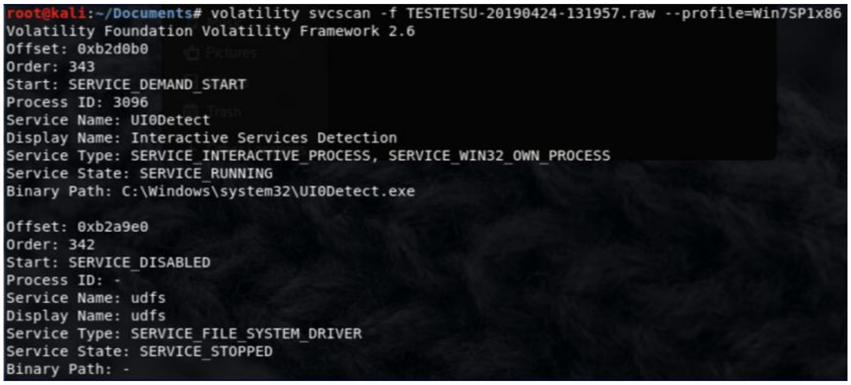
```
root@kali:~/Documents# volatility pslist -f TESTETSU-20190424-131957.raw --profile=Win7SP1x86
Volatility Foundation Volatility Framework 2.6
Offset(V)  Name                    PID   PPID   Thds     Hnds   Sess  Wow64 Start

---------- -------------------- ------ ------ ------ -------- ------ ------ --------------------
-----
0x858b8020 System                    4      0     97      545 ------      0 2019-04-24 12:53:21

0x863cc490 smss.exe                272      4      2       31 ------      0 2019-04-24 12:53:21

0x86b2f030 csrss.exe               356    348      9      572      0      0 2019-04-24 12:53:23

0x858cc138 wininit.exe             464    348      3       80      0      0 2019-04-24 12:53:23

0x86d5f7f0 csrss.exe               472    456     11      379      1      0 2019-04-24 12:53:23

0x872783f0 services.exe            524    464     11      225      0      0 2019-04-24 12:53:24
```

Command pslist against the ram.raw file, image #6



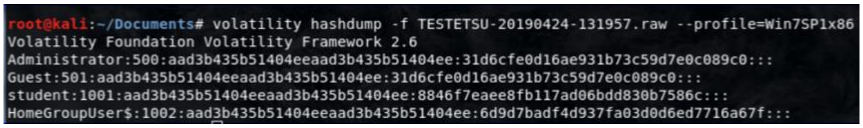
```
root@kali:~/Documents# volatility hivelist -f TESTETSU-20190424-131957.raw --profile=Win7SP1x86
Volatility Foundation Volatility Framework 2.6
Virtual    Physical   Name
---------- ---------- ----
0x81e09008 0x58fc5008 \??\C:\Users\student\AppData\Local\Microsoft\Windows\UsrClass.dat
0x8c80c008 0x6b628008 [no name]
0x8c81c008 0x6c464008 \REGISTRY\MACHINE\SYSTEM
0x8c83f590 0x6b687590 \REGISTRY\MACHINE\HARDWARE
0x8ceec8e8 0x62cea8e8 \SystemRoot\System32\Config\SECURITY
0x8cf93200 0x69d8c200 \SystemRoot\System32\Config\SAM
0x8f32f008 0x6809f008 \??\C:\Windows\ServiceProfiles\NetworkService\NTUSER.DAT
0x908263e8 0x6a6583e8 \SystemRoot\System32\Config\SOFTWARE
0x908cb5d8 0x6a6a55d8 \Device\HarddiskVolume1\Boot\BCD
0x953cd008 0x699de008 \SystemRoot\System32\Config\DEFAULT
0x9542a9c8 0x66d519c8 \??\C:\Windows\ServiceProfiles\LocalService\NTUSER.DAT
0x9725a9c8 0x5c22e9c8 \??\C:\Users\student\ntuser.dat
0xa593c9c8 0x244389c8 \??\C:\System Volume Information\Syscache.hve
```

Command hivelist against the ram.raw file, image #7



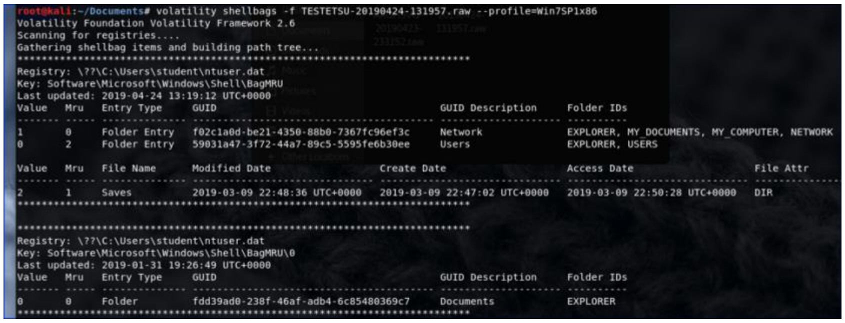
```
root@kali:~/Documents# volatility svcscan -f TESTETSU-20190424-131957.raw --profile=Win7SP1x86
Volatility Foundation Volatility Framework 2.6
Offset: 0xb2d0b0
Order: 343
Start: SERVICE_DEMAND_START
Process ID: 3096
Service Name: UI0Detect
Display Name: Interactive Services Detection
Service Type: SERVICE_INTERACTIVE_PROCESS, SERVICE_WIN32_OWN_PROCESS
Service State: SERVICE_RUNNING
Binary Path: C:\Windows\system32\UI0Detect.exe

Offset: 0xb2a9e0
Order: 342
Start: SERVICE_DISABLED
Process ID: -
Service Name: udfs
Display Name: udfs
Service Type: SERVICE_FILE_SYSTEM_DRIVER
Service State: SERVICE_STOPPED
Binary Path: -
```

Command svcscan against the ram.raw file, image #8



```
root@kali:~/Documents# volatility hashdump -f TESTETSU-20190424-131957.raw --profile=Win7SP1x86
Volatility Foundation Volatility Framework 2.6
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
student:1001:aad3b435b51404eeaad3b435b51404ee:8846f7eaee8fb117ad06bdd830b7586c:::
HomeGroupUser$:1002:aad3b435b51404eeaad3b435b51404ee:6d9d7badf4d937fa03d0d6ed7716a67f:::
```

Command hashdump against the ram.raw file, image #9



```
root@kali:~/Documents# volatility shellbags -f TESTETSU-20190424-131957.raw --profile=Win7SP1x86
Volatility Foundation Volatility Framework 2.6
Scanning for registries....
Gathering shellbag items and building path tree...
***************************************************************************
Registry: \??\C:\Users\student\ntuser.dat
Key: Software\Microsoft\Windows\Shell\BagMRU
Last updated: 2019-04-24 13:19:12 UTC+0000
Value   Mru   Entry Type     GUID                                     GUID Description     Folder IDs
------- ----- -------------- ---------------------------------------- -------------------- ----------
1       0     Folder Entry   f02c1a0d-be21-4350-88b0-7367fc96ef3c     Network              EXPLORER, MY_DOCUMENTS, MY_COMPUTER, NETWORK
0       2     Folder Entry   59031a47-3f72-44a7-89c5-5595fe6b30ee     Users                EXPLORER, USERS

Value   Mru   File Name      Modified Date                  Create Date                    Access Date                    File Attr
------- ----- -------------- ------------------------------ ------------------------------ ------------------------------ --------------
2       1     Saves          2019-03-09 22:48:36 UTC+0000   2019-03-09 22:47:02 UTC+0000   2019-03-09 22:50:28 UTC+0000   DIR
***************************************************************************

***************************************************************************
Registry: \??\C:\Users\student\ntuser.dat
Key: Software\Microsoft\Windows\Shell\BagMRU\0
Last updated: 2019-01-31 19:26:49 UTC+0000
Value   Mru   Entry Type     GUID                                     GUID Description     Folder IDs
------- ----- -------------- ---------------------------------------- -------------------- ----------
0       0     Folder         fdd39ad0-238f-46af-adb4-6c85480369c7     Documents            EXPLORER
***************************************************************************
```

Command shellbags against the ram.raw file, image #10

As shown, each of these commands gathered varying amounts of system information that would be beneficial for a forensics investigation. Here end the processes implemented in this lab, the Observations area will address the knowledge gained through this exercise as well as possible benefits therein.

4. Observations

This process of doing this forensics examination provided a robust platform for the increase of understanding the investigation of a computer system. There were several aspects of this report that provided specific opportunities for learning as well as facilitating the need for external research.

The practice of passing information from the Windows 10 system to the listener on Kali provided a unique understanding of the ability to transfer the gathered information over the internet. This functionality allowed for a nonintrusive way to gather the information from the system without possibly compromising the host system. This ability would allow a virtual investigation to be a feasible option for a system that was physically inaccessible.

To complete the first set of analytics using the Kali listener and the Windows machine sending the information there was a significant amount of research required. To figure out that some of the required executable files are stored Windows32 directory made online research a necessity. Similarly, understanding the usability of volatility against the Windows 10 ram image took some serious looking around on the internet. And after extensive research, it was discovered that the profile must be changed from the given “win7ps1x86” to a preset volatility profile starting with “win10”. After figuring out how to gather all the available profiles from the volitivity application all of them were implemented against the Windows 10 ram image to no avail. This provided the need to revert the final stages of this lab to be done within the Windows 7 system, and then implementing the given volatility profile against it.

Altogether this lab provided an exhalant introduction to forensics while also providing stimulating processes that create a desire to grow in computer science knowledge. By this lab, an individual will learn the basics of dealing with volatile computer information in an organized and common-sense way.