

# **Digital Forensics Lab file**

Submitted by

**Student Name: OJASVI SINGH CHAUHAN  
Roll No: R134218111  
SAP ID: 500068394**



**SCHOOL OF COMPUTER SCIENCE  
UNIVERSITY OF PETROLEUM & ENERGY STUDIES  
Bidholi Campus, Energy Acres, Dehradun – 248007.**

**April - 2020**

## INDEX

<b>Experiment No.</b>	<b>Experiment Name</b>	<b>Description</b>
<b>1</b>	<b><i>Creating a Forensic Image using FTK Imager/Encase Imager</i></b>	<ul style="list-style-type: none"> <li>• Creating Forensic Image</li> <li>• Check Integrity of Data</li> <li>• Analyze Forensic Image</li> </ul>
<b>2</b>	<b><i>Creating RAM Dump using FTK Imager and Volatility</i></b>	<ul style="list-style-type: none"> <li>• Create RAM Dump of Windows XP/7</li> <li>• Analyze using Volatility</li> </ul>
<b>3</b>	<b><i>Forensics Case Study</i></b>	<ul style="list-style-type: none"> <li>• Solve the Case study (image file) provide in lab using Volatility tool</li> <li>• Use of Virus Total</li> </ul>
<b>4</b>	<b><i>Capturing and analyzing network packets using Wireshark (Fundamentals)</i></b>	<ul style="list-style-type: none"> <li>• Identification the live network</li> <li>• Capture Packets</li> <li>• Analyze the captured packets</li> <li>• Nmap using Wireshark</li> </ul>
<b>5</b>	<b><i>Analyze the packets provided in lab and solve the questions using Wireshark</i></b>	<ul style="list-style-type: none"> <li>• Analysis of various pcap files provided in lab using Wireshark</li> </ul>
<b>6</b>	<b><i>Using Sysinternals tools for Network Tracking and Process Monitoring</i></b>	<ul style="list-style-type: none"> <li>• Check Sysinternals tools</li> <li>• Monitor Live Processes</li> <li>• Capture RAM</li> <li>• Capture TCP/UDP packets</li> <li>• Monitor Hard Disk</li> <li>• Monitor Virtual Memory</li> <li>• Monitor Cache Memory</li> </ul>
<b>7</b>	<b><i>Recovering and Inspecting deleted files</i></b>	<ul style="list-style-type: none"> <li>• Check for Deleted Files</li> <li>• Recover the Deleted Files</li> <li>• Analyzing and Inspecting the recovered files</li> <li>• Use \$Iparse to see the metadata</li> <li>• See metadata through command-line</li> <li>• Perform this using recovery option in ENCASE</li> </ul>
<b>8</b>	<b><i>Email Forensics</i></b>	<ul style="list-style-type: none"> <li>• Mail Service Providers</li> <li>• Email protocols</li> <li>• Recovering emails</li> <li>• Analyzing email header</li> </ul>
<b>9</b>	<b><i>Web Browser Forensics</i></b>	<ul style="list-style-type: none"> <li>• Web Browser working</li> <li>• Forensics activities on browser</li> <li>• Cache / Cookies analysis</li> <li>• Last Internet activity</li> </ul>

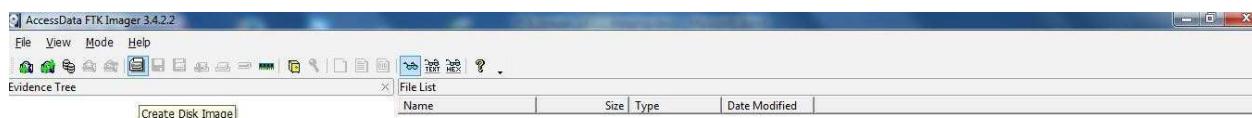
# EXPERIMENT 1

## *Creating a Forensic Image using FTK Imager/Encase Imager*

Use FTK Imager to create the image of files in pen drive and verify the hash values.

### TASK I: Imaging

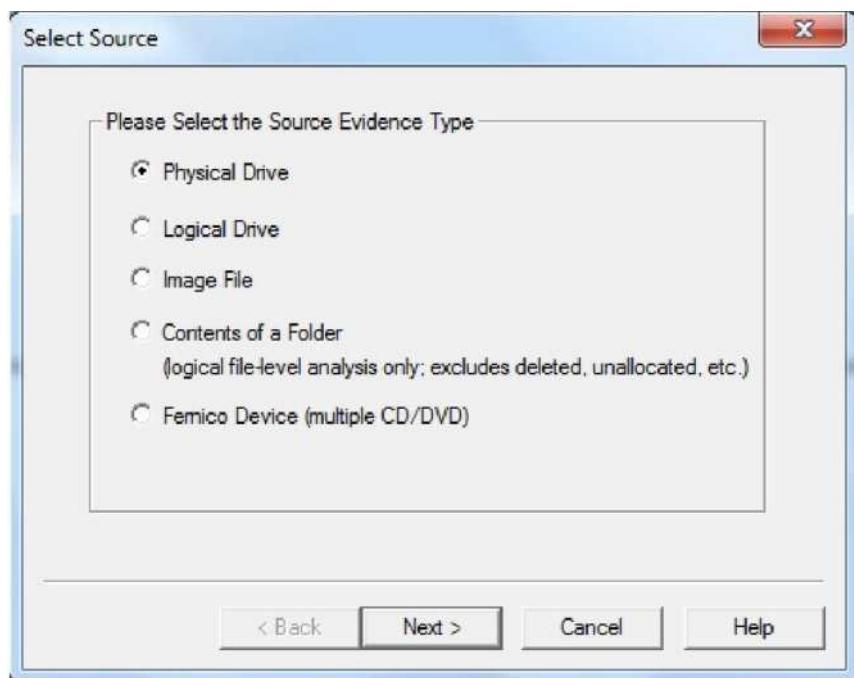
1. To run the application, select the application, right click on it and run as an "Administrator".
2. The application will be opened as shown below.



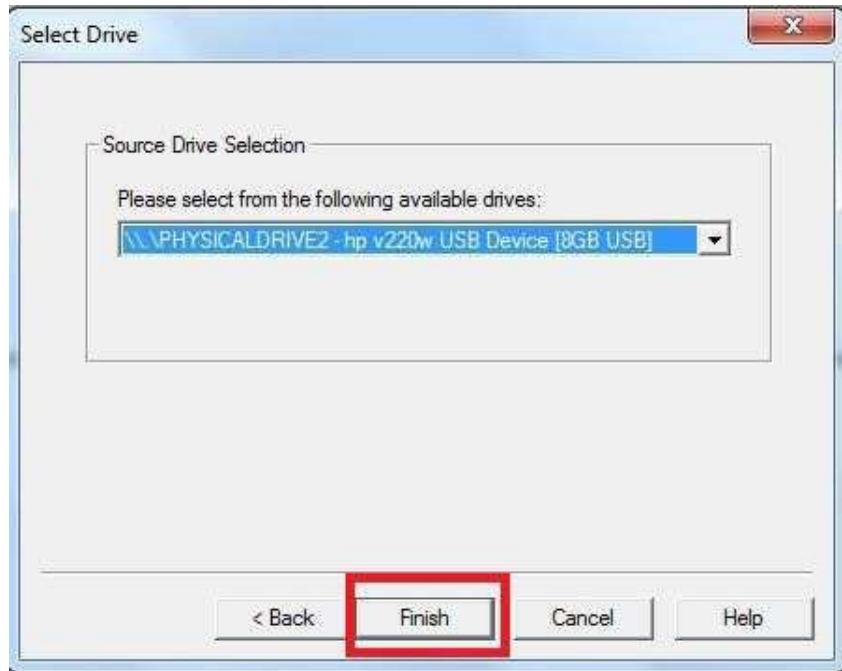
Now to create an Image, we need to select the icon as shown in the above image.

**Note:** In this experiment we have used Sandisk 15 GB pendrive to create an Image.

3. After clicking on the icon, the page will be opened as shown below:



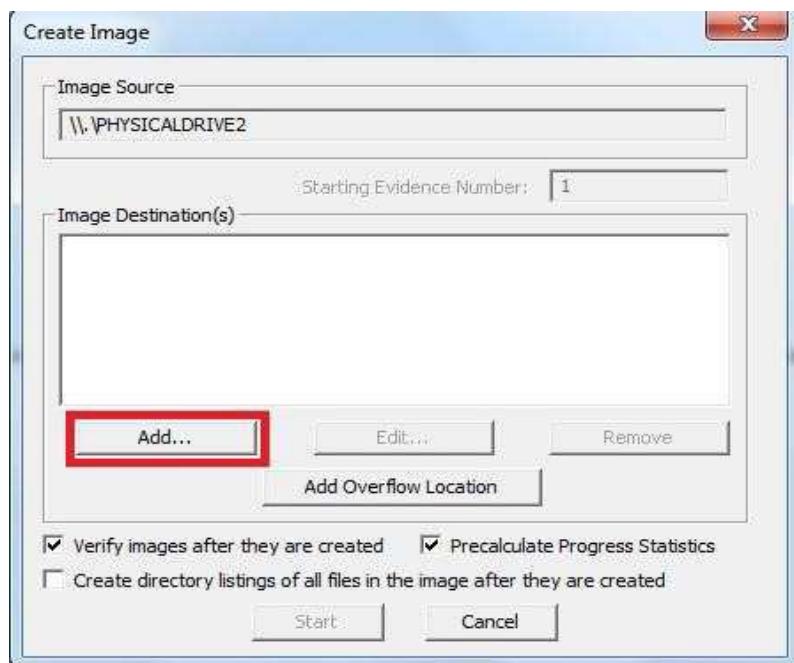
4. After selecting the device type, click on "Next" button to proceed, the page will be opened as shown below:



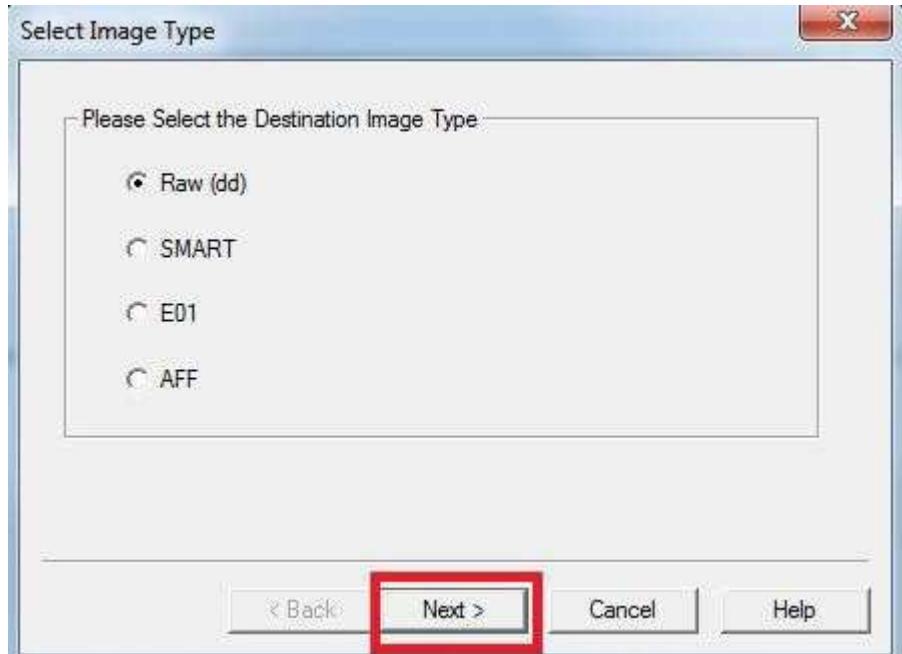
Select the drive to which you want to create an Image as shown above:

After selecting the drive, click on "Finish" button, the page will be opened as shown.

5. Click on "Add" button as shown above to add the destination location to save the image:-



6. After clicking on Add button, the page will be opened as shown above. Select the image type you want to create and click on "Next" button as shown below:-



7. After clicking on "Next" button, the page will be displayed as shown below
- 8.

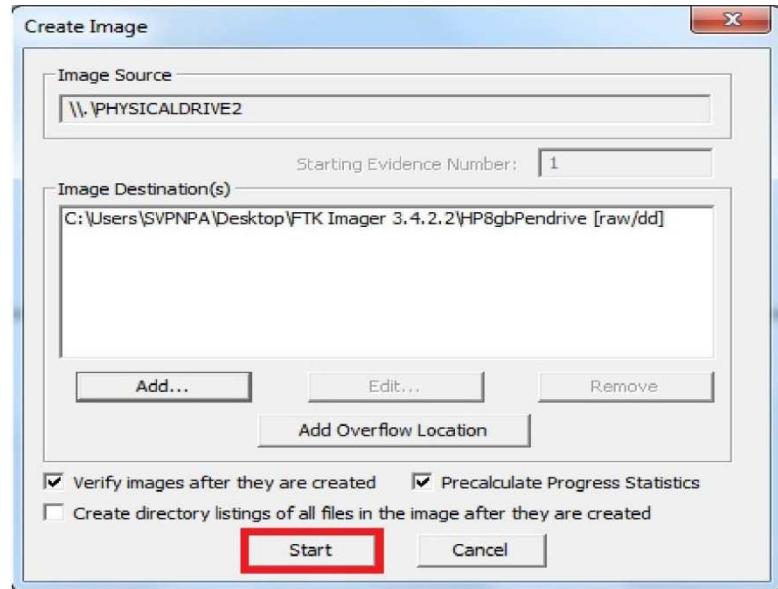
A screenshot of a Windows-style dialog box titled "Evidence Item Information". It contains five text input fields:

- Case Number: 1
- Evidence Number: 1
- Unique Description: 8 gb HP pendrive data Image
- Examiner: NPA
- Notes: Imaging 8 gb HP pendrive data

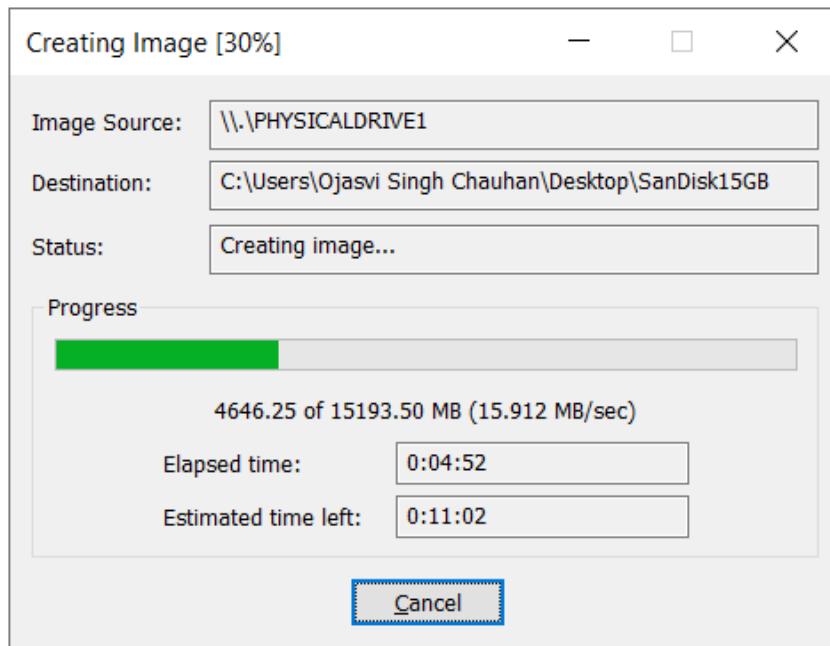
At the bottom are buttons for "< Back", "Next >" (which is highlighted with a red box), "Cancel", and "Help".

Fill the evidence item information and click on "Next" button as shown above.

9. Fill the image destination folder location details and Image File name details and Click on "Finish" button.
10. After clicking on Finish button, all the details will be added as shown below



Click on "Start" button as shown above to start imaging process. This can be observed from the below image.



11. After completion of imaging, the hash value of the image will be calculated using MD5 and SHA1 algorithm and will be displayed as shown below.

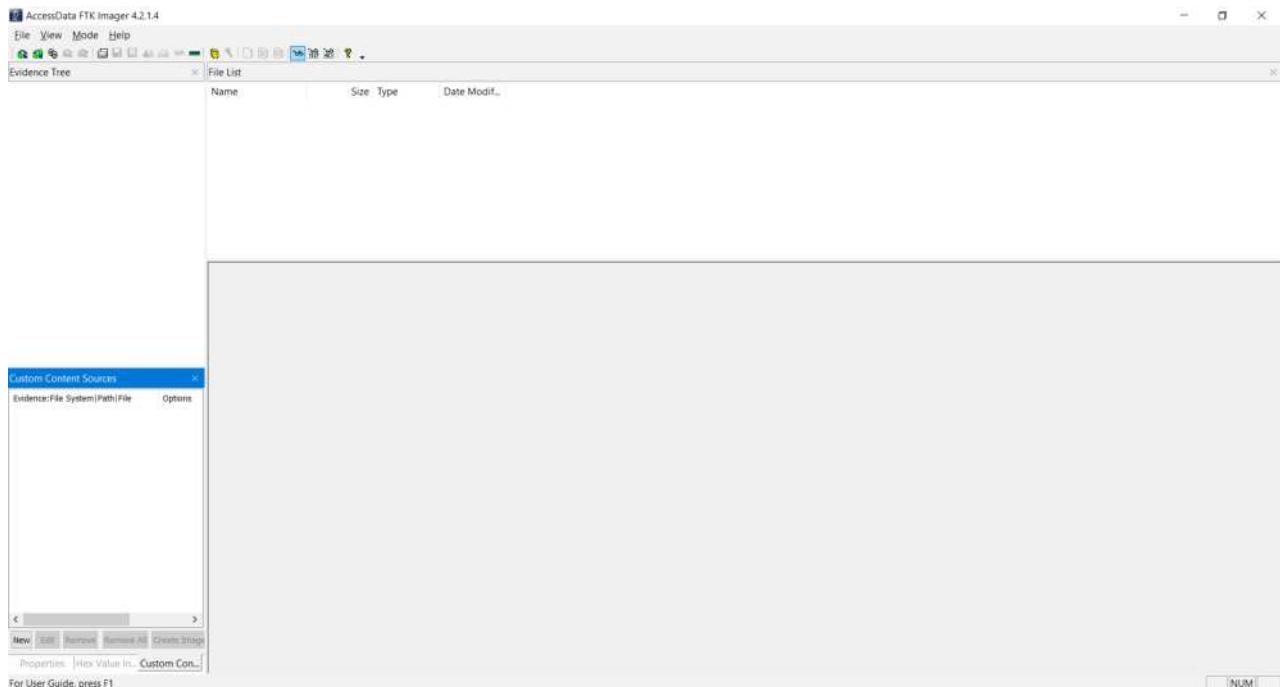
Drive/Image Verify Results	
Name	Sandisk15gb.001
Sector count	31116288
<b>MD5 Hash</b>	
Computed hash	765d507c5fd3ac97689dbb4da1ccbd7b
Report Hash	765d507c5fd3ac97689dbb4da1ccbd7b
Verify result	Match
<b>SHA1 Hash</b>	
Computed hash	34a71db24e281480a757d0446f19b1b6adb6beb3
Report Hash	34a71db24e281480a757d0446f19b1b6adb6beb3
Verify result	Match
<b>Bad Blocks List</b>	
Bad block(s) in image	No bad blocks found in image

[Close](#)

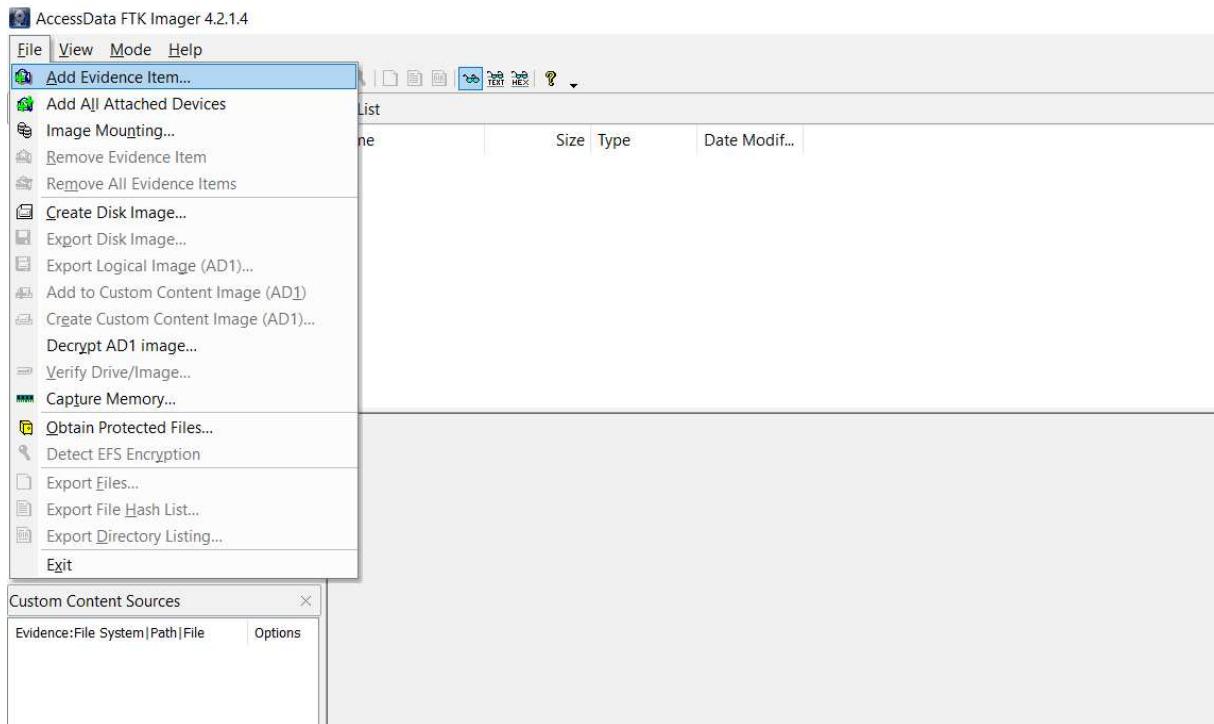
## TASK II: Hashing

Verify the MD5 SHA1 hash value of an image Using FTK Imager version 4.2

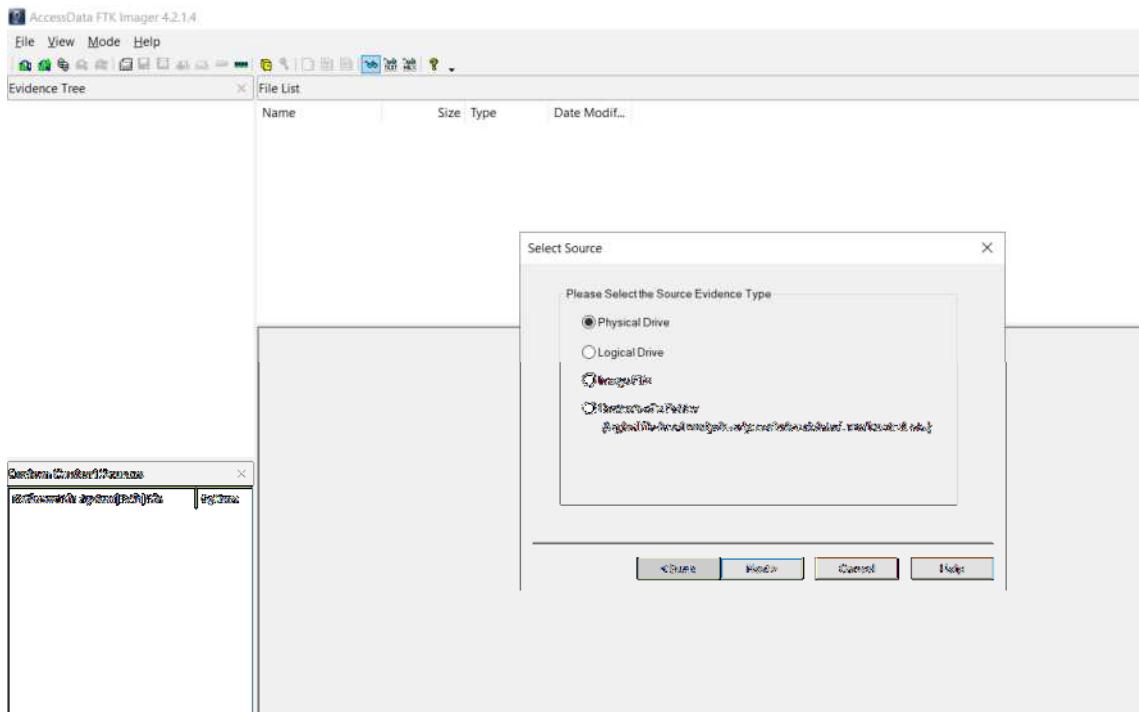
1. Launch FTK Imager



2. Select File > Add Evidence Item



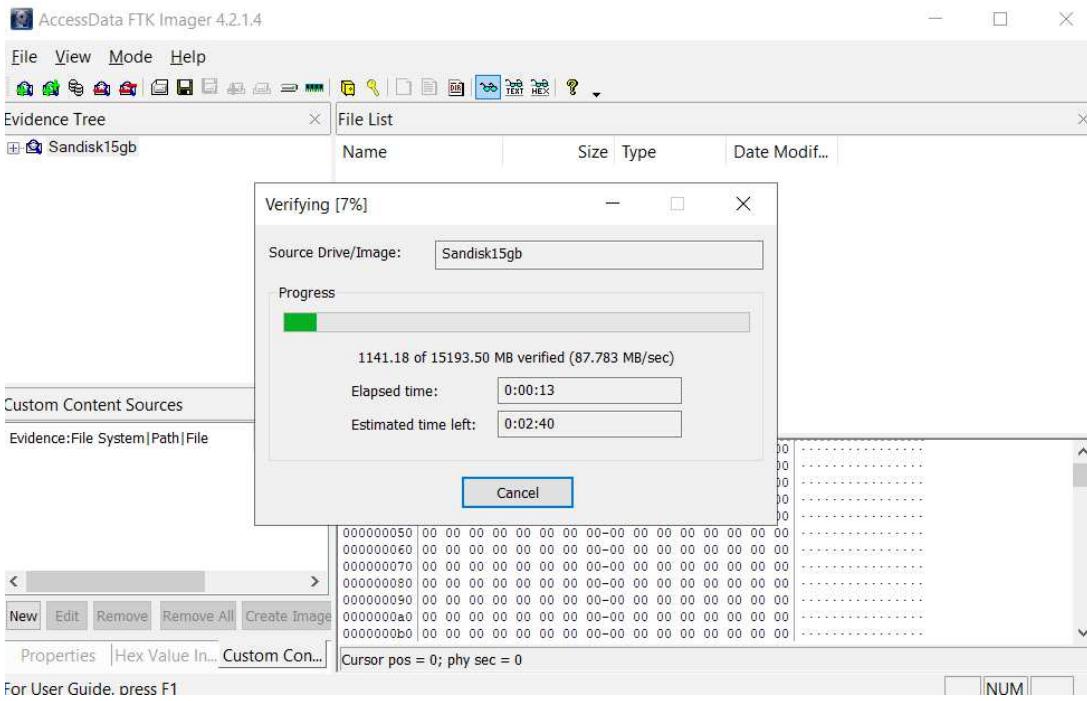
### 3. Select "Image File" and proceed to add the image



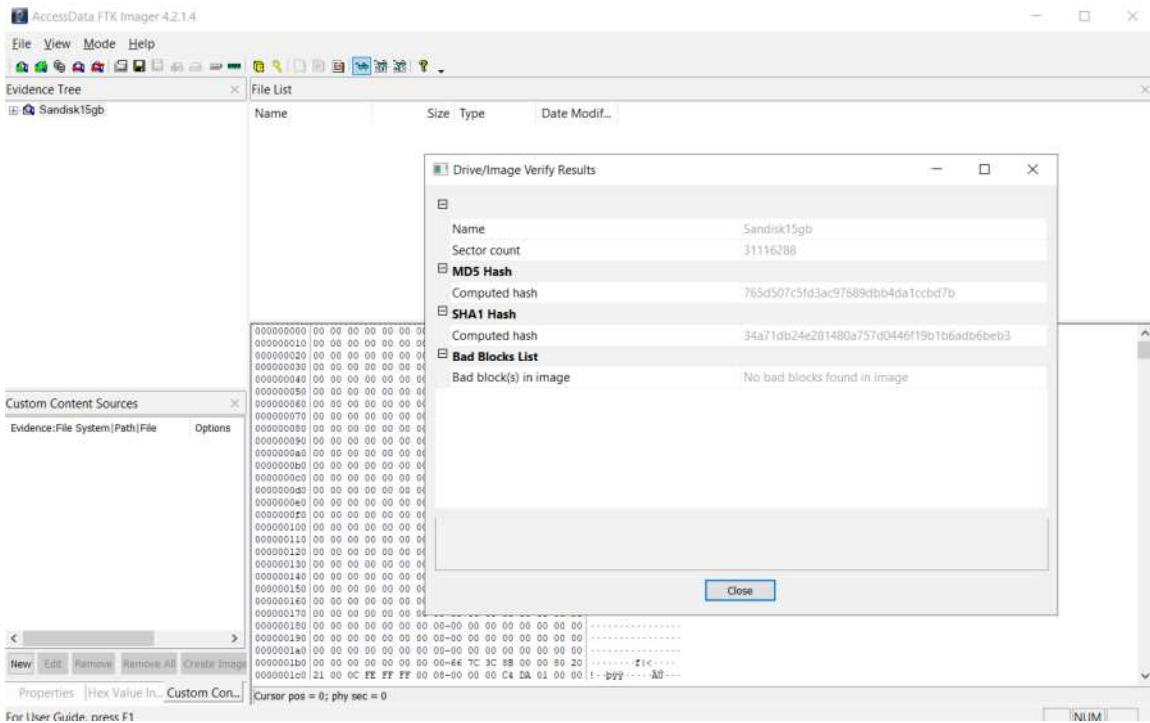
The screenshot shows the AccessData FTK Imager interface. The Evidence Tree pane on the left displays a single item: "Sandisk15gb". The File List pane on the right is currently empty. Below the panes, a "Custom Content Sources" dialog is open, showing a list of file paths under "Evidence:File System|Path|File". The bottom status bar indicates "Listed: 0 Selected: 0 Sandisk15gb".

4. Under the "Evidence Tree", right click your image and select Verify Drive/Image.

The screenshot shows the AccessData FTK Imager interface. The Evidence Tree pane on the left has "Sandisk15gb" selected, and a context menu is open with "Verify Drive/Image..." highlighted. The File List pane on the right is empty. Below the panes, a "Custom Content Sources" dialog is open, showing a list of file paths under "Evidence:File System|Path|File". The bottom status bar indicates "Hashes the selected drive or image and compares the results to stored verification hashes, if any".



## 5. Compare the hash value calculated to the known hash value.



## **EXPERIMENT – 2**

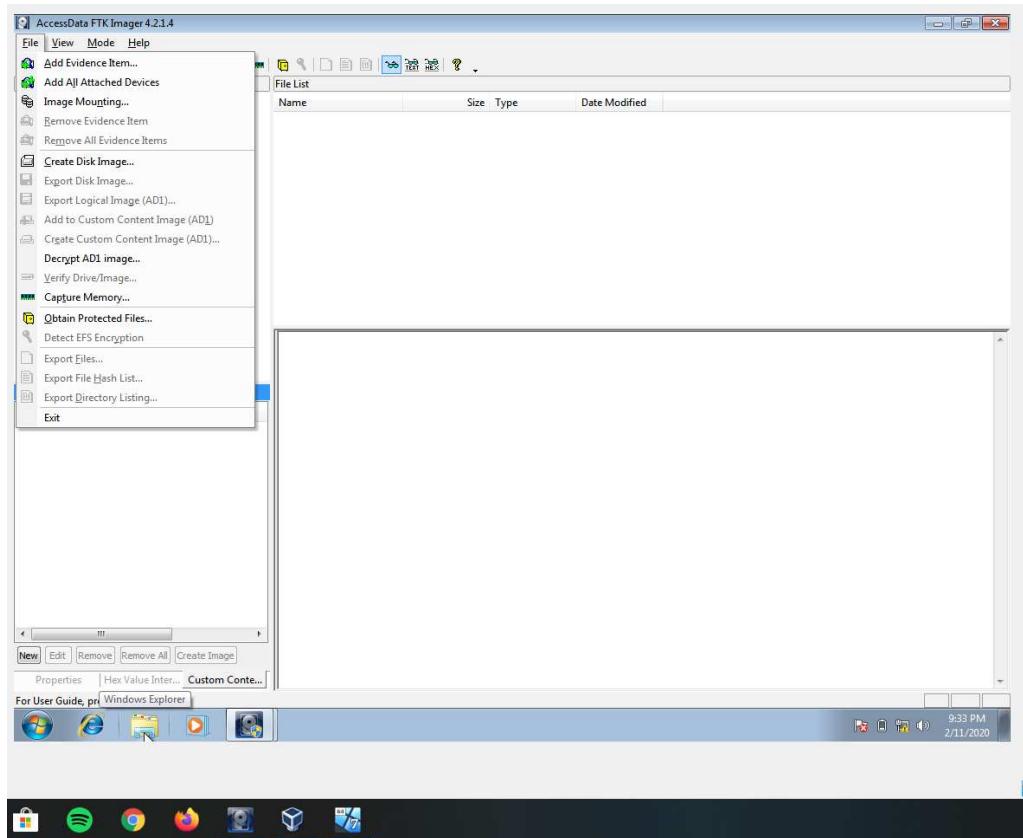
## ***Creating RAM Dump using FTK Imager and Volatility***

## **VOLATILITY: RAM DUMP**

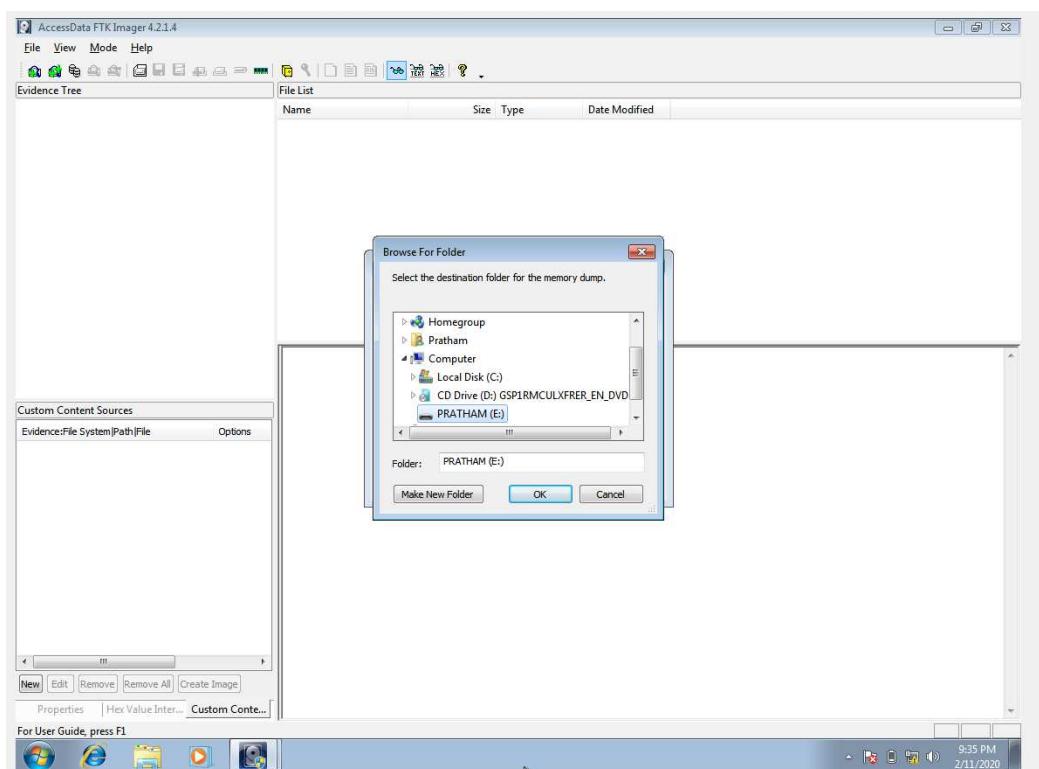
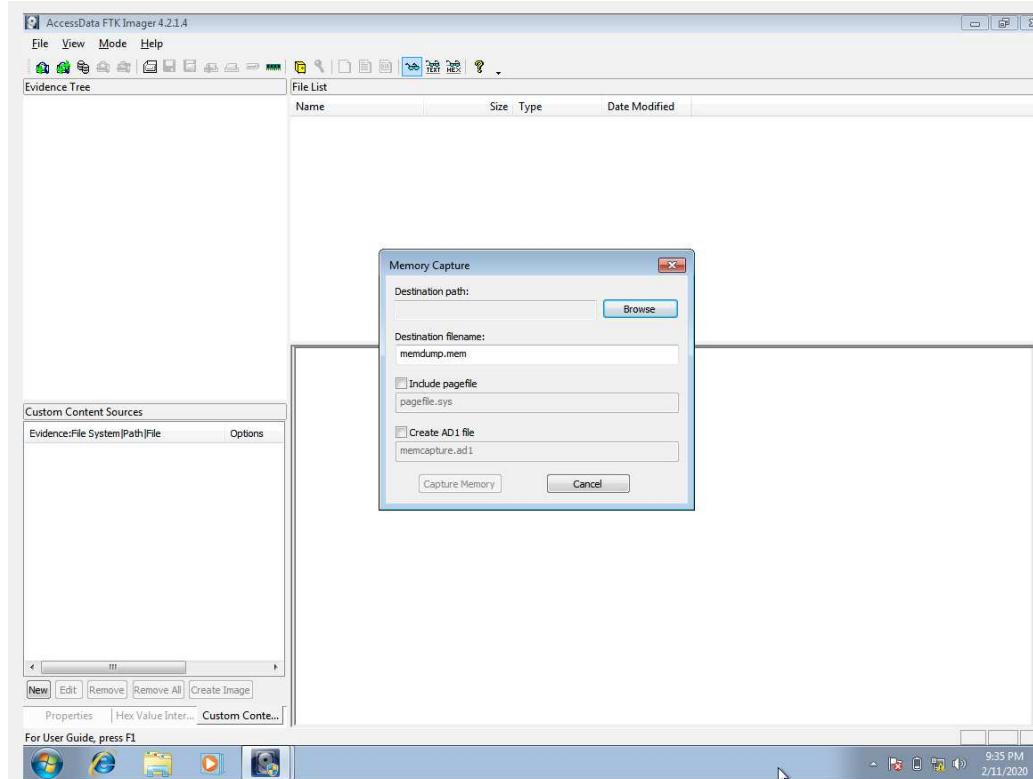
**Aim:** Creating a Ram dump using FTK Imager & analysing it using volatility.

## **Procedure:**

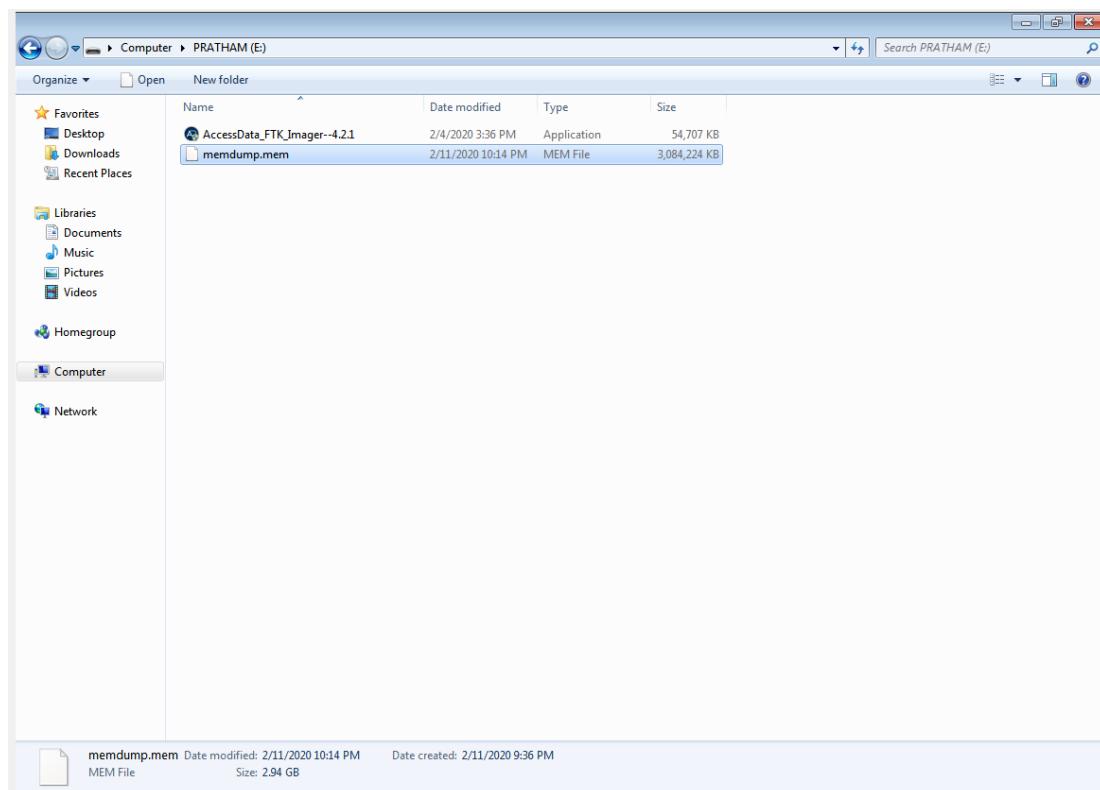
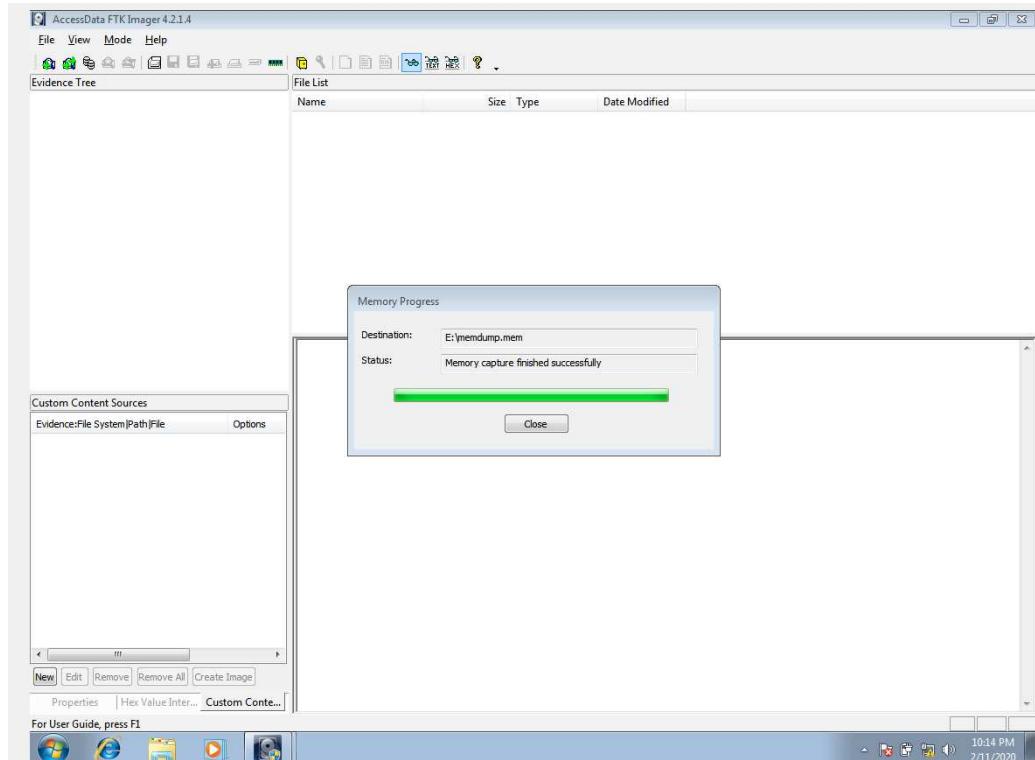
- Open FTK Imager and click on file.



- Select capture memory and select the location where it is to be saved.



- A .mem file will be saved in the target folder.



- Copy the file to Kali partition.

- Run the file on volatility by setting the profile to Windows 7 and listing the processes by pslist.

```

2.6 Win Profiles · volatility ... root@kali: ~/Downloads
root@kali:~/Downloads

File Actions Edit View Help
root@kali:~/Downloads x

root@kali:~# cd Downloads
root@kali:~/Downloads# volatility -f memdump.mem --profile=Win7SP1x64_23418 pslist
Volatility Foundation Volatility Framework 2.6
Offset(V) Name PID PPID Thds Hnds Sess Wow64 Start
----- -----
0xfffffa8002409040 System 4 0 82 538 ----- 0 2020-02-11 15:57:17 UTC+0000
0xfffffa8003509740 smss.exe 216 4 2 29 ----- 0 2020-02-11 15:57:18 UTC+0000
0xfffffa80035a73b0 csrss.exe 288 280 9 409 0 0 2020-02-11 15:57:22 UTC+0000
0xfffffa8002410060 wininit.exe 336 280 3 74 0 0 2020-02-11 15:57:24 UTC+0000
0xfffffa8002411b30 csrss.exe 344 328 8 197 1 0 2020-02-11 15:57:24 UTC+0000
0xfffffa8003d04810 winlogon.exe 372 328 3 113 1 0 2020-02-11 15:57:24 UTC+0000
0xfffffa8003d341a0 services.exe 432 336 9 196 0 0 2020-02-11 15:57:25 UTC+0000
0xfffffa8003d45b30 lsass.exe 440 336 8 697 0 0 2020-02-11 15:57:26 UTC+0000
0xfffffa8003d48b30 lsm.exe 448 336 10 138 0 0 2020-02-11 15:57:26 UTC+0000
0xfffffa8003da35a0 svchost.exe 540 432 10 356 0 0 2020-02-11 15:57:28 UTC+0000
0xfffffa8003dcc500 svchost.exe 616 432 7 250 0 0 2020-02-11 15:57:29 UTC+0000
0xfffffa8003e09b30 svchost.exe 704 432 22 569 0 0 2020-02-11 15:57:30 UTC+0000
0xfffffa8003e1f460 svchost.exe 748 432 31 651 0 0 2020-02-11 15:57:30 UTC+0000
0xfffffa8003e27b30 svchost.exe 772 432 30 928 0 0 2020-02-11 15:57:30 UTC+0000
0xfffffa8003e9c060 svchost.exe 924 432 21 495 0 0 2020-02-11 15:57:32 UTC+0000
0xfffffa8003ec3890 svchost.exe 1012 432 15 460 0 0 2020-02-11 15:57:33 UTC+0000
0xfffffa8003f20550 spoolsv.exe 820 432 13 283 0 0 2020-02-11 15:57:35 UTC+0000
0xfffffa8003f575e0 svchost.exe 844 432 19 314 0 0 2020-02-11 15:57:36 UTC+0000
0xfffffa8003fcf650 taskhost.exe 1184 432 7 144 1 0 2020-02-11 15:57:37 UTC+0000
0xfffffa8003fd7890 svchost.exe 1208 432 23 304 0 0 2020-02-11 15:57:37 UTC+0000
0xfffffa800407f3e0 dwm.exe 1332 748 3 69 1 0 2020-02-11 15:57:38 UTC+0000
0xfffffa80040ae060 explorer.exe 1360 1292 25 753 1 0 2020-02-11 15:57:39 UTC+0000
0xfffffa80041a8060 SearchIndexer. 2044 432 11 614 0 0 2020-02-11 15:57:51 UTC+0000
0xfffffa80042c7b30 wmpnetwk.exe 1480 432 23 525 0 0 2020-02-11 15:57:53 UTC+0000
0xfffffa80042c68d0 svchost.exe 1344 432 9 345 0 0 2020-02-11 15:57:57 UTC+0000
0xfffffa80025bbb30 sppsvc.exe 1844 432 4 141 0 0 2020-02-11 15:59:48 UTC+0000
0xfffffa8002641060 svchost.exe 2436 432 12 316 0 0 2020-02-11 15:59:48 UTC+0000
0xfffffa800259a060 FTK Imager.exe 2140 1360 7 322 1 0 2020-02-11 16:02:46 UTC+0000
0xfffffa8002b38b30 WUDFHost.exe 2780 748 8 216 0 0 2020-02-11 16:04:38 UTC+0000
root@kali:~/Downloads# volatility -f memdump.mem --profile=Win7SP1x64_23418 cmdline
Volatility Foundation Volatility Framework 2.6
*****
System pid: 4
*****
smss.exe pid: 216
Command line : \SystemRoot\System32\smss.exe
*****
csrss.exe pid: 288
Command line : %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Windows=On SubSystemType=alization,2 ServerDl=sxssrv,4 ProfileControl=Off MaxRequestThreads=16
*****
wininit.exe pid: 336
Command line : wininit.exe
*****
csrss.exe pid: 344
Command line : %SystemRoot%\system32\csrss.exe ObjectDirectory=\Windows SharedSection=1024,20480,768 Windows=On SubSystemType=alization,2 ServerDl=sxssrv,4 ProfileControl=Off MaxRequestThreads=16
*****
winlogon.exe pid: 372
Command line : winlogon.exe
*****
services.exe pid: 432
Command line : C:\Windows\system32\services.exe
*****
```

# EXPERIMENT – 3

## *Forensics Case Study*

### Analysis of the Memory dump using Volatility Forensics tool

**Command 1:** Volatility -f stuxnet.vmem pslist

```
root@kali:~# cd Documents
root@kali:~/Documents# volatility -f stuxnet.vmem pslist
Volatility Foundation Volatility Framework 2.6
Offset(V) Name PID PPID Thds Hnds Sess Wow64 Start Exit
0x823c8830 System 4 0 59 403 ----- 0
0x820df020 smss.exe 376 4 3 19 ----- 0 2010-10-29 17:08:53 UTC+0000
0x821a2d00 csrss.exe 600 376 11 395 0 0 2010-10-29 17:08:54 UTC+0000
0x81d56550 winlogon.exe 624 376 19 579 0 0 2010-10-29 17:08:54 UTC+0000
0x82073020 services.exe 668 624 21 431 0 0 2010-10-29 17:08:54 UTC+0000
0x81e70020 lsass.exe 680 624 19 342 0 0 2010-10-29 17:08:54 UTC+0000
0x823315d8 vmauthip.exe 844 668 1 25 0 0 2010-10-29 17:08:55 UTC+0000
0x81db8d00 svchost.exe 856 668 17 193 0 0 2010-10-29 17:08:55 UTC+0000
0x81e61da0 svchost.exe 940 668 13 312 0 0 2010-10-29 17:08:55 UTC+0000
0x822843e0 svchost.exe 1032 668 61 1169 0 0 2010-10-29 17:08:55 UTC+0000
0x81e18b20 svchost.exe 1080 668 5 89 0 0 2010-10-29 17:08:55 UTC+0000
0x81ff7f20 svchost.exe 1200 668 14 197 0 0 2010-10-29 17:08:55 UTC+0000
0x81fee6b0 spoolsv.exe 1412 668 10 118 0 0 2010-10-29 17:08:56 UTC+0000
0x81fe0edaa qmgr.exe 1580 668 5 148 0 0 2010-10-29 17:09:05 UTC+0000
0x81fe52d0 vmtoolsd.exe 1664 668 5 284 0 0 2010-10-29 17:09:05 UTC+0000
0x821a0568 VMUpgradeHelper 1816 668 3 96 0 0 2010-10-29 17:09:06 UTC+0000
0x82059adad alg.exe 188 668 6 107 0 0 2010-10-29 17:09:09 UTC+0000
0x829ec7e0 explorer.exe 1196 1728 16 582 0 0 2010-10-29 17:11:49 UTC+0000
0x8206cc10 wsctrlfy.exe 2040 1032 1 28 0 0 2010-10-29 17:11:49 UTC+0000
0x81fe6978 TSNCache.exe 324 1196 7 54 0 0 2010-10-29 17:11:49 UTC+0000
0x81fc5da0 VMwareTray.exe 1912 1196 1 50 0 0 2010-10-29 17:11:50 UTC+0000
0x81febb60 VMwareUser.exe 1356 1196 9 251 0 0 2010-10-29 17:11:50 UTC+0000
0x821b0d478 jushed.exe 1712 1196 1 26 0 0 2010-10-29 17:11:50 UTC+0000
0x82279998 imapi.exe 756 668 4 116 0 0 2010-10-29 17:11:54 UTC+0000
0x822b9a10 vuault.exe 976 1032 3 133 0 0 2010-10-29 17:12:03 UTC+0000
0x81c543a0 Procmon.exe 660 1196 13 189 0 0 2011-06-03 04:25:56 UTC+0000
0x81fa5390 vmprvse.exe 1872 856 5 134 0 0 2011-06-03 04:25:58 UTC+0000
0x81c498c8 lsass.exe 868 668 2 23 0 0 2011-06-03 04:26:55 UTC+0000
0x81c47c00 lsass.exe 1928 668 4 65 0 0 2011-06-03 04:26:55 UTC+0000
0x81c0cda0 cmd.exe 968 1664 0 ----- 0 0 2011-06-03 04:31:35 UTC+0000 2011-06-03 04:31:36 UTC+0000
0x81f14938 ipconfig.exe 304 968 0 ----- 0 0 2011-06-03 04:31:35 UTC+0000 2011-06-03 04:31:36 UTC+0000
root@kali:~/Documents# volatility -f stuxnet.vmem image info
Volatility Foundation Volatility Framework 2.6
ERROR : volatility.debug : You must specify something to do (try -h)
root@kali:~/Documents# volatility -f stuxnet.vmem malfind -p 1928 --dump-dir Documents
Volatility Foundation Volatility Framework 2.6
ERROR : volatility.debug : Documents is not a directory
```

Then, we find connections and then we find any process in which we can find any malware.

"**lsass.exe**" is the Local Security Authentication Server. It verifies the validity of user logons to your PC or server. **Lsass** generates the **process** responsible for authenticating users for the Winlogon service. This is performed by using authentication packages such as the default, Msgina.

## Command 2: volatility -f stuxnet.vmem malfind -p 668 –dump-dir dump

```
root@kali:~/Documents# volatility -f stuxnet.vmem malfind -p 668 --dump-dir Dump
Volatility Foundation Volatility Framework 2.6
Process: services.exe Pid: 668 Address: 0x940000
Vad Tag: Vad Protection: PAGE_EXECUTE_READWRITE
Flags: Protection: 6

0x00940000 90 06 94 00 c6 07 94 00 24 00 94 00 a5 04 00 00 .....$.....
0x00940010 f2 04 94 00 48 06 00 00 c9 04 94 00 29 00 00 .....H.....
0x00940020 00 00 c5 00 e8 13 00 00 00 5a 77 4d 61 70 56 69 .....ZwMapVi
0x00940030 65 77 4f 66 53 65 63 74 69 6f 6e 00 5a 51 81 c1 ewOfSection.ZO.

0x00940080 NOP
0x00940081 PUSH ES
0x00940082 XCHG ESP, EAX
0x00940083 ADD DH, AL
0x00940085 POP ES
0x00940086 XCHG ESP, EAX
0x00940087 ADD [EAX+EAX], AH
0x00940088 XCHG ESP, EAX
0x00940089 ADD [EBP-0xdffffc], AH
0x0094008a ADD AL, 0x94
0x0094008b ADD [EAX-0x61], CL
0x0094008c ADD [EAX], AL
0x0094008d LEAVE
0x0094008e ADD AL, 0x94
0x0094008f ADD AL, 0x94
0x00940090 ADD [ECX], CH
0x00940091 ADD [EAX], AL
0x00940092 ADD CH, AL
0x00940093 ADD AL, CH
0x00940094 ADC EAX, [EAX]
0x00940095 ADD [EAX], AL
0x00940096 POP EDX
0x00940097 IA_0x9400979
0x00940098 POPA
0x00940099 JO 0x940085
0x0094009a 6965774f665365 INUL ESP, [EBP+0x77], 0x6553664f
0x0094009b 6374698f AMPL [ECX+BP+2+0x6f], SI
0x0094009c DUTS DX, BYTE [E51]
0x0094009d ADD [DX+0x51], BL
0x0094009e DB 0x81
0x0094009f DB 0xc1

Process: services.exe Pid: 668 Address: 0x13f0000
Vad Tag: Vad Protection: PAGE_EXECUTE_READWRITE
```

Then upload dump file on virustotal.com to check.

DETECTION	DETAILS	COMMUNITY	
AegisLab	0 suspicious	Ad-Aware	Gen:Variant.Gootkit.Eicar.17940
ALYac	0 generic	Antivirus	0 generic
Arcabit	0 generic	Avast	0 generic
AVG	0 generic	BIDefender	0 generic
CAT-QuickHeal	0 generic	Comodo	0 generic
CynetStrike Falcon	0 generic	Cyber	0 generic
DrWeb	0 generic	eGentle	0 generic
Emsisoft	0 generic	Endgame	0 generic
eScan	0 generic	FirEye	0 generic
GDax	0 generic	IKAN	0 generic
K7AntiVirus	0 generic	Know	0 generic
MAX	0 generic	McAfee	0 generic
McAfee-GW-Edition	0 generic	Microsoft	0 generic

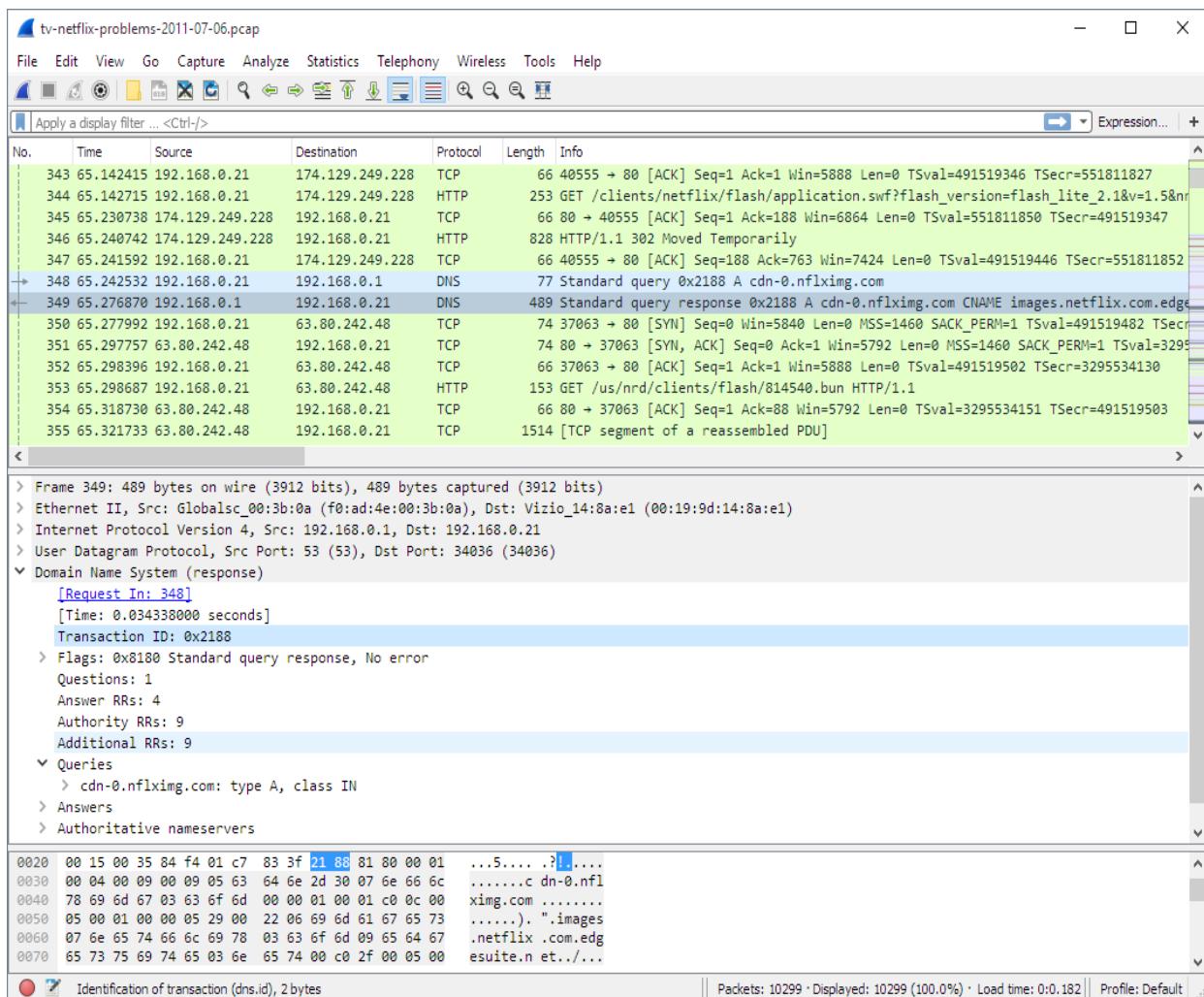
# EXPERIMENT 4: WIRESHARK FUNDAMENTALS

## 1. What is Wireshark?

Wireshark is a network packet analyzer. A network packet analyzer presents captured packet data in as much detail as possible. We could think of a network packet analyzer as a measuring device for examining what's happening inside a network cable, just like an electrician uses a voltmeter for examining what's happening inside an electric cable (but at a higher level, of course).

## 2. User Interface

- a) **The Main Window:** Let's look at Wireshark's user interface. The Main window shows Wireshark as you would usually see it after some packets are captured or loaded.



Wireshark's main window consists of parts that are commonly known from many other GUI Programs.

1. The **menu** is used to start actions.
2. The **main toolbar** provides quick access to frequently used items from the menu.
3. The **filter toolbar** allows users to set *display filters* to filter which packets are displayed.
4. The **packet list pane** displays a summary of each packet captured. By clicking on packets in this pane you control what is displayed in the other two panes.
5. The **packet details pane** displays the packet selected in the packet list pane in more detail.
6. The **packet bytes pane** displays the data from the packet selected in the packet list pane, and highlights the field selected in the packet details pane.
7. The **statusbar** shows some detailed information about the current program state and the captured data.

- b) The “Main” Toolbar:** The main toolbar provides quick access to frequently used items from the menu. This toolbar cannot be customized by the user, but it can be hidden using the View menu if the space on the screen is needed to show more packet data.



Main tool bar items:

Toolbar Icon	Toolbar Item	Menu Item	Description
	[ Start ]	Capture > Start	Starts capturing packets with the same options as the last capture or the default options if none were set ( <a href="#">Start Capturing</a> ).
	[ Stop ]	Capture > Stop	Stops the currently running capture ( <a href="#">Start Capturing</a> ).
	[ Restart ]	Capture > Restart	Restarts the current capture session.
	[ Options... ]	Capture > Options...	Opens the “Capture Options” dialog box. See <a href="#">Start Capturing</a> for details.

- c) Packet List pane:** The packet list pane displays all the packets in the current capture file. Each line in the packet list corresponds to one packet in the capture file. If you

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.0.21	192.168.0.1	DNS	84	Standard query 0x403d A moviecontrol.netflix.com
2	0.055888	192.168.0.1	192.168.0.21	DNS	479	Standard query response 0x403d A moviecontrol.netflix.com CNAME nccp-moviecontrol-fro
3	0.057694	192.168.0.21	50.17.249.22	TCP	74	37314->443 [SYN] Seq=0 Win=5840 Len=0 MSS=1468 SACK_PERM=1 TStamp=491454310 TSectr=0 WSE
4	0.154716	50.17.249.22	192.168.0.21	TCP	74	443->37314 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1 TSval=2102931926 TSectr=0 WSE
5	0.155962	192.168.0.21	50.17.249.22	TCP	66	37314->443 [ACK] Seq=1 Ack=1 Win=5888 Len=0 TStamp=491454408 TSectr=2102931926
6	0.163162	192.168.0.21	50.17.249.22	TLSv1	187	Client Hello
7	0.250734	50.17.249.22	192.168.0.21	TCP	66	443->37314 [ACK] Seq=1 Ack=122 Win=5792 Len=0 TStamp=2102931950 TSectr=491454416
8	0.252716	50.17.249.22	192.168.0.21	TLSv1	1514	Server Hello
9	0.253828	192.168.0.21	50.17.249.22	TCP	66	37314->443 [ACK] Seq=122 Ack=1449 Win=8768 Len=0 TStamp=491454507 TSectr=2102931950
10	0.254730	50.17.249.22	192.168.0.21	TCP	1514	[TCP segment of a reassembled PDU]
11	0.254778	50.17.249.22	192.168.0.21	TLSv1	349	Certificate
12	0.255853	192.168.0.21	50.17.249.22	TCP	66	37314->443 [ACK] Seq=122 Ack=2897 Win=11648 Len=0 TStamp=491454509 TSectr=2102931950
13	0.256102	192.168.0.21	50.17.249.22	TCP	66	37314->443 [ACK] Seq=122 Ack=3180 Win=14528 Len=0 TStamp=491454509 TSectr=2102931950
14	0.319870	192.168.0.21	50.17.249.22	TLSv1	264	Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message
15	0.411795	50.17.249.22	192.168.0.21	TLSv1	125	Change Cipher Spec, Encrypted Handshake Message

select a line in this pane, more details will be displayed in the “Packet Details” and “Packet Bytes” panes.

- d) **Packet Details Pane:** The packet details pane shows the current packet (selected in the “Packet List” pane) in a more detailed form.

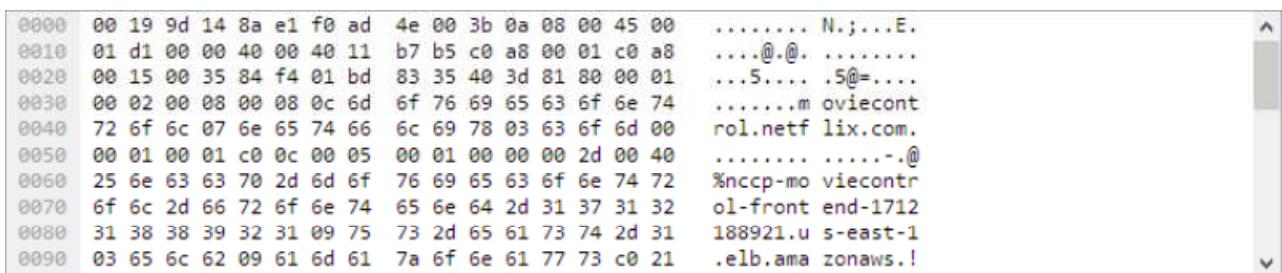


The screenshot shows the NetworkMiner interface with the "Packet Details" pane open. The selected packet is a DNS response (opcode 0x8180). The tree view shows the following structure:

- > Ethernet II, Src: Globalsc\_00:3b:0a (f0:ad:4e:00:3b:0a), Dst: Vizio\_14:8a:e1 (00:19:9d:14:8a:e1)
- > Internet Protocol Version 4, Src: 192.168.0.1, Dst: 192.168.0.21
- > User Datagram Protocol, Src Port: 53 (53), Dst Port: 34036 (34036)
- ▼ Domain Name System (response)
  - [Request In: 11]
    - [Time: 0.055880000 seconds]
    - Transaction ID: 0x403d
  - > Flags: 0x8180 Standard query response, No error
  - Questions: 1
  - Answer RRs: 2
  - Authority RRs: 8
  - Additional RRs: 8
  - > Queries
  - > Answers
  - > Authoritative nameservers
  - > Additional records

This pane shows the protocols and protocol fields of the packet selected in the “Packet List” pane. The protocols and fields of the packet are shown in a tree which can be expanded and collapsed.

- e) **Packet Bytes Pane:** The packet bytes pane shows the data of the current packet (selected in the “Packet List” pane) in a hexdump style.



0000	00 19 9d 14 8a e1 f0 ad	4e 00 3b 0a 08 00 45 00	..... N.;...E.
0010	01 d1 00 00 40 00 40 11	b7 b5 c0 a8 00 01 c0 a8	....@. @. ....
0020	00 15 00 35 84 f4 01 bd	83 35 40 3d 81 80 00 01	...5.... .5@=....
0030	00 02 00 08 00 08 0c 6d	6f 76 69 65 63 6f 6e 74	.....m oviecont
0040	72 6f 6c 07 6e 65 74 66	6c 69 78 03 63 6f 6d 00	rol.netf lix.com.
0050	00 01 00 01 c0 0c 00 05	00 01 00 00 00 2d 00 40	..... .....-@
0060	25 6e 63 63 70 2d 6d 6f	76 69 65 63 6f 6e 74 72	%nccp-mo viecontr
0070	6f 6c 2d 66 72 6f 6e 74	65 6e 64 2d 31 37 31 32	ol-front end-1712
0080	31 38 38 39 32 31 09 75	73 2d 65 61 73 74 2d 31	188921.u s-east-1
0090	03 65 6c 62 09 61 6d 61	7a 6f 6e 61 77 73 c0 21	.elb.ama zonaws.!.

The “Packet Bytes” pane shows a canonical [hex dump](#) of the packet data. Each line contains the data offset, sixteen hexadecimal bytes, and sixteen ASCII bytes. Non-printable bytes are replaced with a period (“.”).

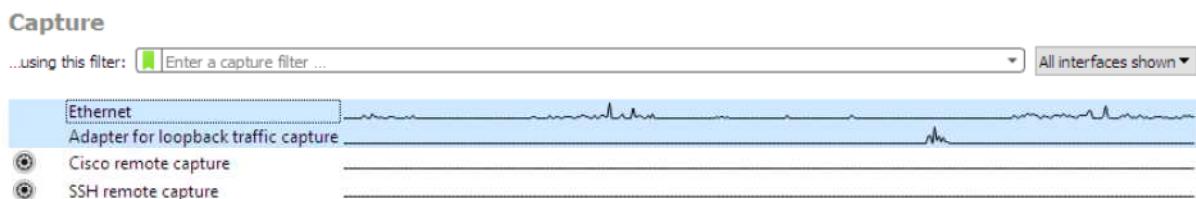
### 3. Capturing Live network Data

- a. **Introduction:** Capturing live network data is one of the major features of Wireshark.  
The Wireshark capture engine provides the following features:

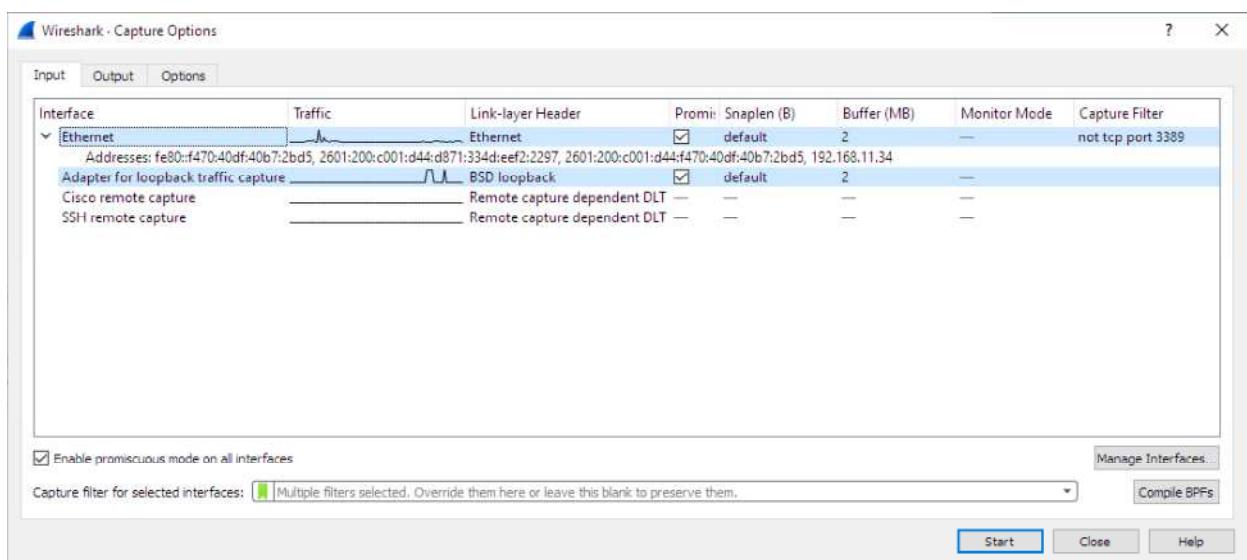
- Capture from different kinds of network hardware such as Ethernet or 802.11.
- Simultaneously capture from multiple network interfaces.
- Stop the capture on different triggers such as the amount of captured data, elapsed time, or the number of packets.
- Simultaneously show decoded packets while Wireshark is capturing.
- Filter packets, reducing the amount of data to be captured.
- Save packets in multiple files while doing a long term capture, optionally rotating through a fixed number of files (a “ringbuffer”).

- b. **Capture types:** Capture can be performed by two ways:

- i. **The “Capture” Section Of The Welcome Screen:** When you open Wireshark without starting a capture or opening a capture file it will display the “Welcome Screen,” which lists any recently opened capture files and available capture interfaces. Network activity for each interface will be shown in a sparkline next to the interface name. It is possible to select more than one interface and capture from them simultaneously.

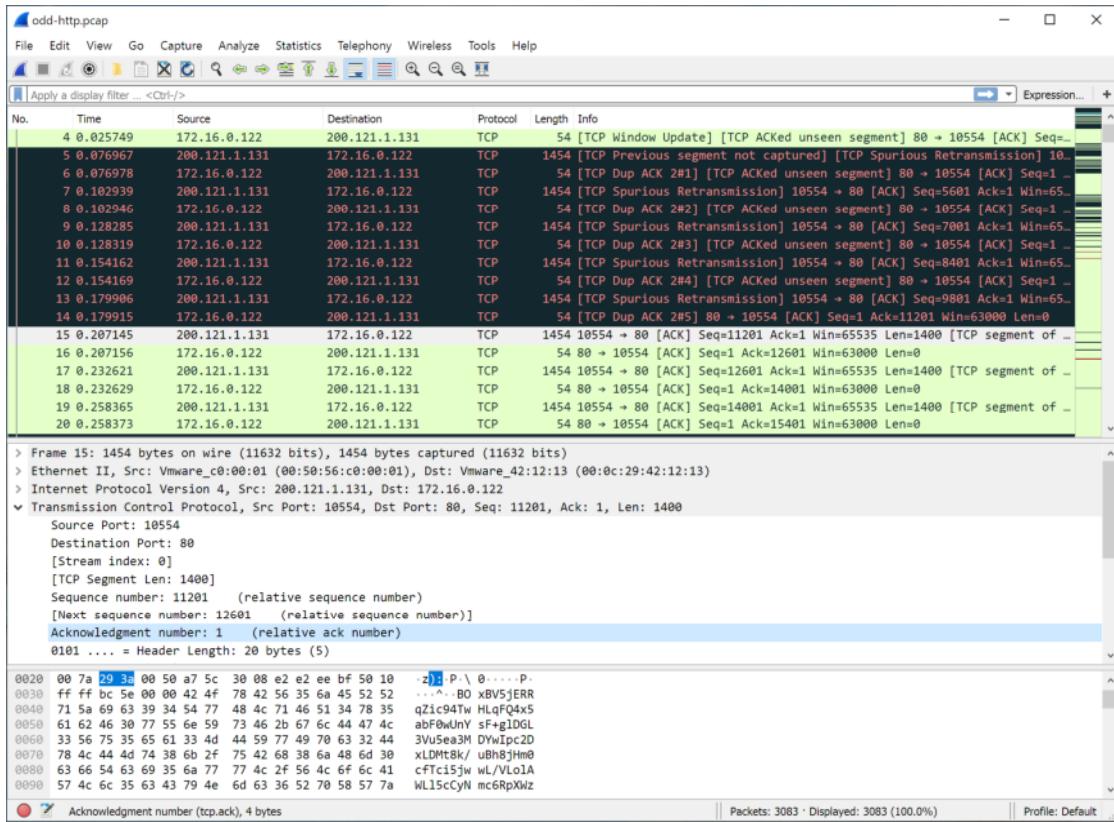


- ii. **The “Capture Options” Dialog Box:** When you select **Capture > Options...** (or use the corresponding item in the main toolbar), Wireshark pops up the “Capture Options” dialog box as shown in [The “Capture Options” input tab](#).

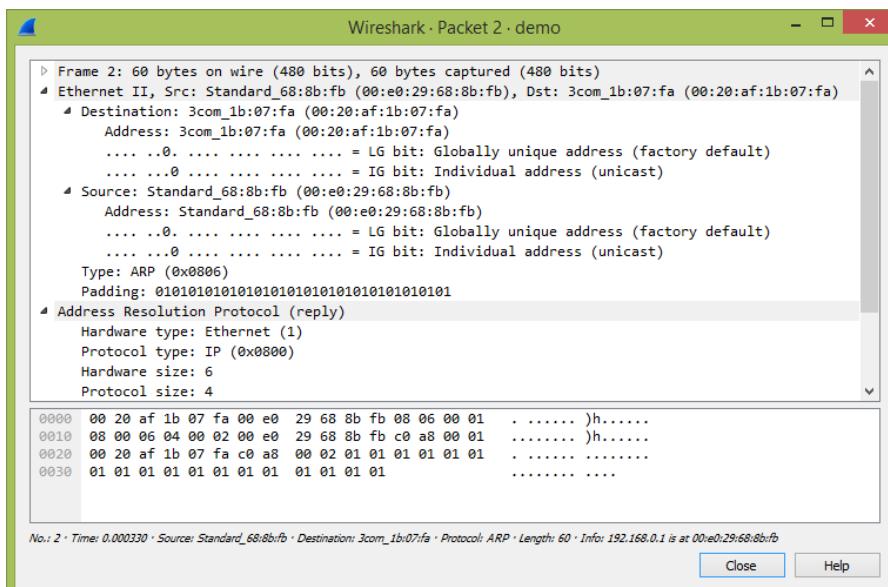


## 4. Working with Captured Packets

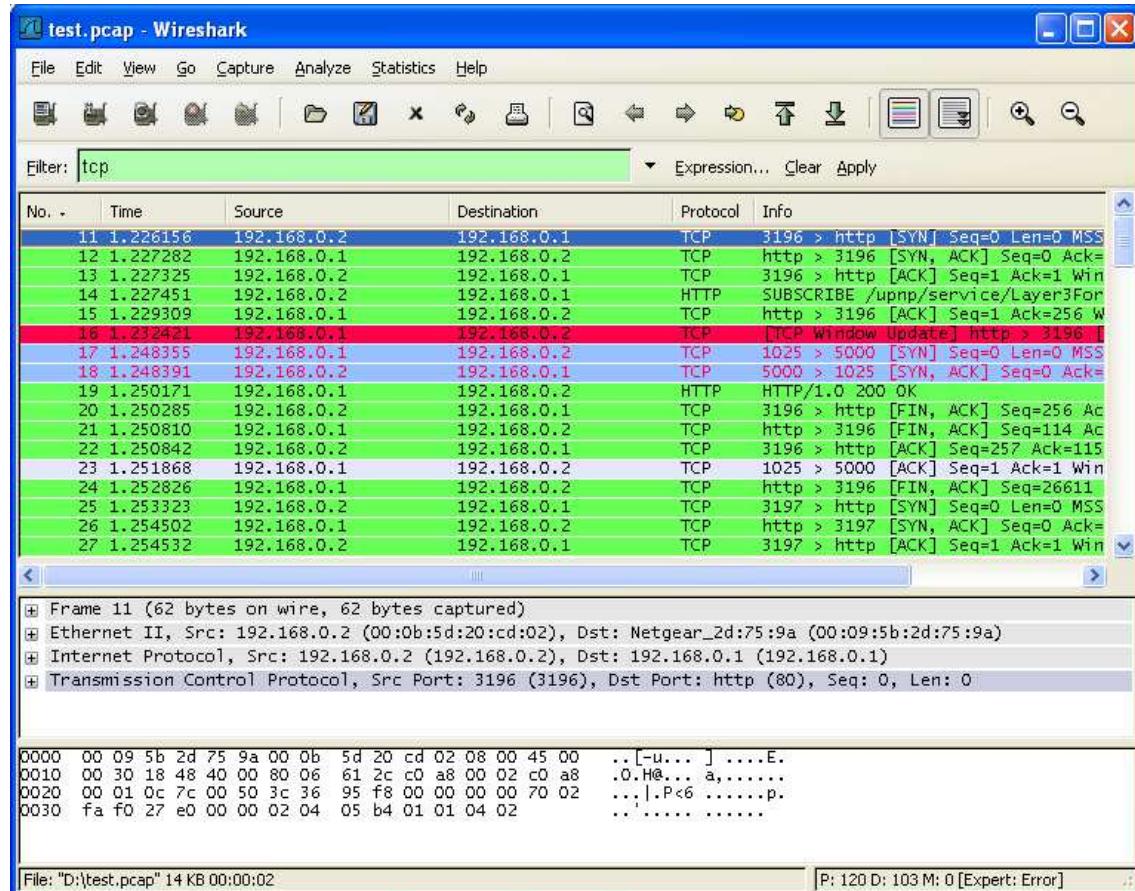
- a. **Viewing Packets we've captured:** Once you have captured some packets or you have opened a previously saved capture file, you can view the packets that are displayed in the packet list pane by simply clicking on a packet in the packet list pane, which will bring up the selected packet in the tree view and byte view panes.



In addition we can view individual packets in a separate window as shown in viewing a packet in a separate window by selecting the packet in which you are interested in the packet list pane and selecting **View > Show Packet in New Window**. This allows us to easily compare two or more packets, even across multiple files.

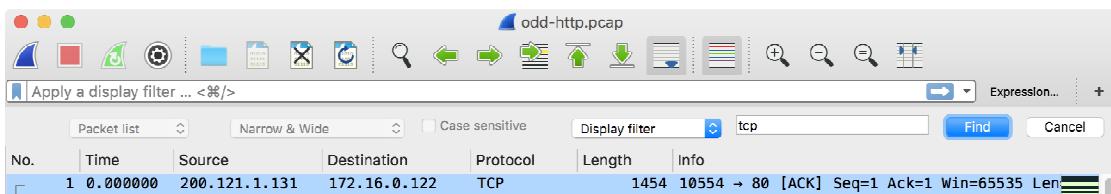


- b. Filtering Packets while viewing:** Wireshark has two filtering languages: *capture filters* and *display filters*. *Capture filters* are used for filtering when capturing packets and are discussed in Filtering while capturing.



- c. Finding Packets:** We can easily find packets once we have captured some packets or have read in a previously saved capture file. Simply select **Edit > Find Packet...** in the main menu. Wireshark will open a toolbar between the main toolbar and the packet list shown in The “Find Packet” toolbar.

#### The “Find Packet” Toolbar



We can search using the following criteria:

#### Display filter

Enter a display filter string into the text entry field and click the **[ Find ]** button. + For example, to find the three way handshake for a connection from host 192.168.0.1, use the following filter string:

ip. src==192.168.0.1 and tcp. flags. syn==1

The value to be found will be syntax checked while you type it in. If the syntax check of your value succeeds, the background of the entry field will turn green, if it fails, it will turn red. For more details see [Filtering Packets While Viewing](#)

### Hexadecimal Value

Search for a specific byte sequence in the packet data. For example, use “ef:bb:bf” to find the next packet that contains the [UTF-8 byte order mark](#).

### String

Find a string in the packet data, with various options.

## 5. Following Protocol Streams

It can be very helpful to see a protocol in the way that the application layer sees it. Perhaps we are looking for passwords in a Telnet stream, or we are trying to make sense of a data stream. Maybe we just need a display filter to show only the packets in a TLS or SSL stream. If so, Wireshark’s ability to follow protocol streams will be useful to us.

To filter to a particular stream, select a TCP, UDP, TLS, or HTTP packet in the packet list of the stream/connection we are interested in and then select the menu item **Analyze > Follow TCP Stream** (or use the context menu in the packet list). Wireshark will set an appropriate display filter and display a dialog box with the data from the stream laid out, as shown in The “Follow TCP Stream” dialog box.

The screenshot shows the Wireshark interface with a title bar "Wireshark · Follow TCP Stream (tcp.stream eq 0) · test.cap". The main window displays a text-based conversation between two hosts. The client host (192.168.0.1) sends a POST request to the server host (192.168.0.2) to subscribe to a UPnP service. The server responds with a 200 OK status, indicating successful subscription. The interface includes standard Wireshark controls at the bottom: "Entire conversation (368 bytes)", "Show and save data as ASCII", "Stream 0", "Find:" input field, "Find Next" button, and links for "Help", "Filter Out This Stream", "Print", "Save as...", "Back", and "Close".

```
SUBSCRIBE /upnp/service/Layer3Forwarding HTTP/1.1
NT: upnp:event
Callback: <http://192.168.0.2:5000/notify>
Timeout: Second-1800
User-Agent: Mozilla/4.0 (compatible; UPnP/1.0; Windows NT/5.1)
Host: 192.168.0.1
Content-Length: 0
Pragma: no-cache

HTTP/1.0 200 OK
Connection: close
Server: UPnP/1.0 UPnP-Device-Host/1.0
Timeout: Second-1800
SID: uuid:cf

3 client pkts, 4 server pkts, 3 turns.
```

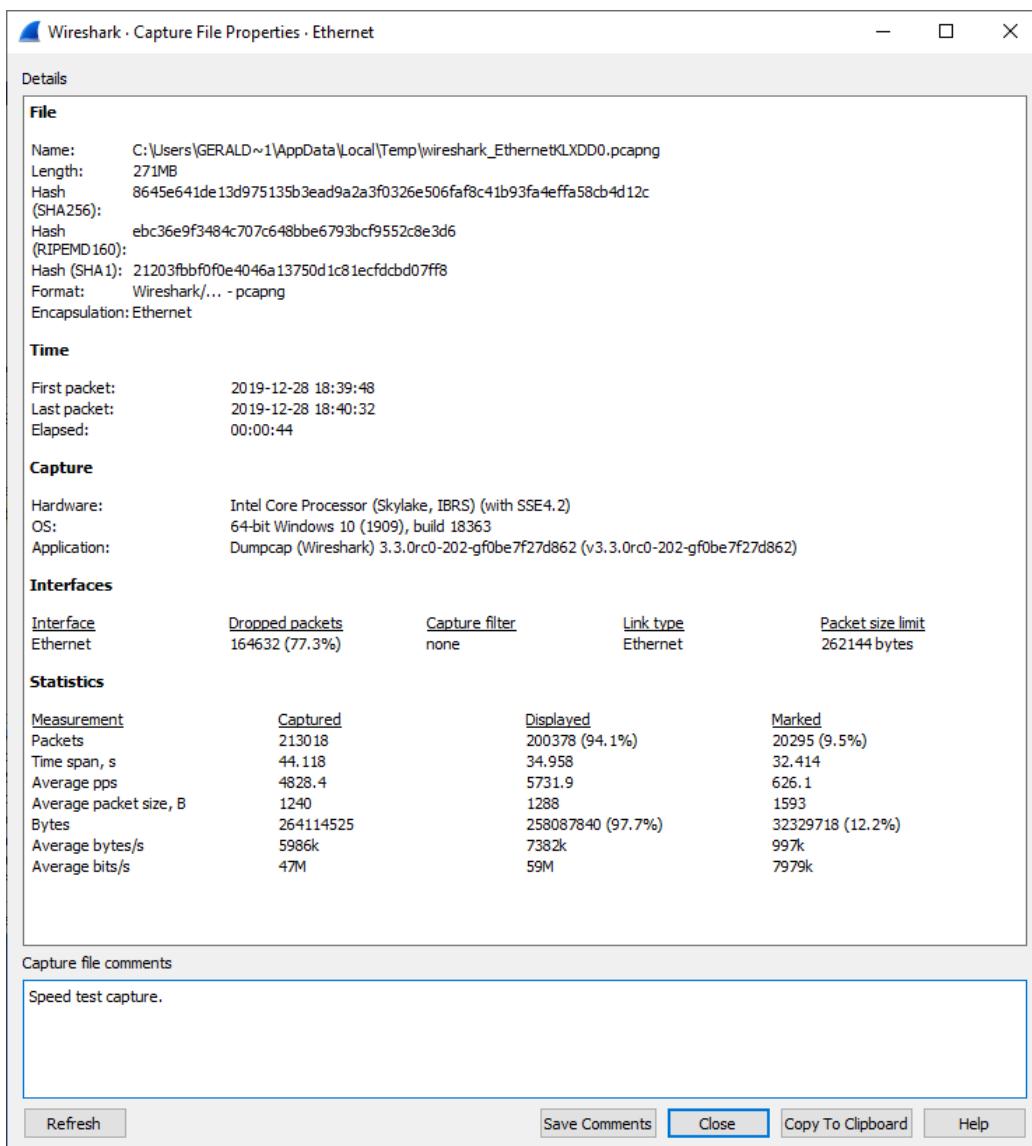
## 6. Statistics

- a. **Introduction:** Wireshark provides a wide range of network statistics which can be accessed via the Statistics menu. These statistics range from general information about the loaded capture file (like the number of captured packets), to statistics about specific protocols (e.g. statistics about the number of HTTP requests and responses captured).

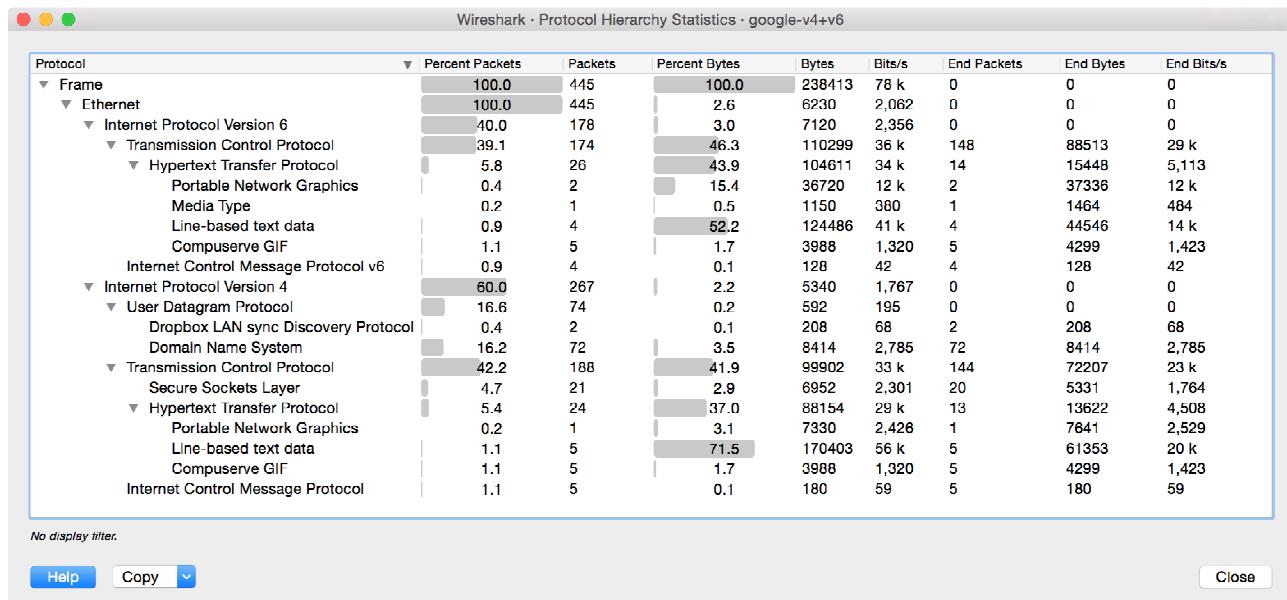
### *General statistics*

- **Capture File Properties** about the capturefile.
- **Protocol Hierarchy** of the captured packets.
- **Conversations** e.g. traffic between specific IP addresses.
- **Endpoints** e.g. traffic to and from an IP addresses.
- **I/O Graphs** visualizing the number of packets (or similar) in time.

- b. **The “Capture File Properties” Dialog:** General information about the current capture file.



c. The “Protocol Hierarchy” Window: The protocol hierarchy of the captured packets.

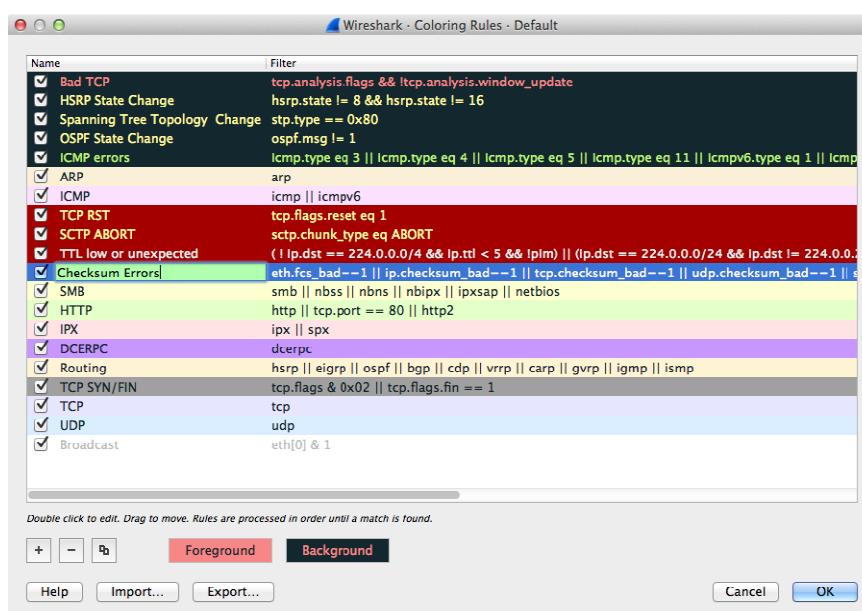


This is a tree of all the protocols in the capture. Each row contains the statistical values of one protocol. Two of the columns (*Percent Packets* and *Percent Bytes*) serve double duty as bar graphs. If a display filter is set, it will be shown at the bottom.

## 7. Customizing Wireshark

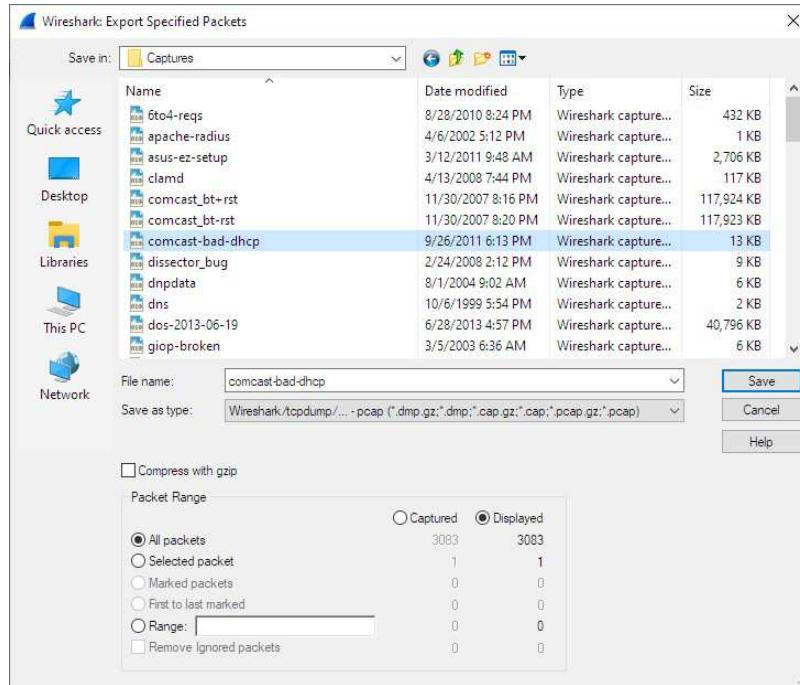
a. **Packet Colorization:** A very useful mechanism available in Wireshark is packet colorization. You can set up Wireshark so that it will colorize packets according to a display filter. This allows you to emphasize the packets you might be interested in.

To permanently colorize packets, select **View > Coloring Rules....** Wireshark will display the “Coloring Rules” dialog box as shown in the “Coloring Rules” dialog box.



**b. Exporting Data:** Wireshark provides a variety of options for exporting packet data.

The “Export Specified Packets” Dialog Box



This is similar to the “[Save](#)” dialog box, but it lets you save specific packets. This can be useful for trimming irrelevant or unwanted packets from a capture file.

# EXPERIMENT 5

## WIRESHARK CASE STUDIES

### ACCESS POINT FORENSICS

1) Joe's WAP is beaconing. Based on the contents of the packet capture, What is:

- a. The SSID of his access point?

➤ SSID = **MentOrNet**

**Steps:** Open the access point forensics.pcap in the Wireshark and capture the packets and find for string **Beacon** in packet details and then analyze the Cisco packet.

Time	Source	Destination	Protocol	Length	Ethernet	Info
1 0.000000	Cisco-Li_61:00:d0	Broadcast	802.11	105		Beacon frame, SN=3583, FN=0, Flags=....., BI=100, SSID=MentOrNet

- b. The BSSID of his access point?

➤ BSSID = **Cisco-Li\_61:00:d0 (00:23:69:61:00:d0)**

**Steps:** in the captured packets, inside the Frame section:-

Go to IEEE 802.11, by traversing down, we will get the BSSID:

No.	Time	Source	Destination	Protocol	Length	Ethernet	Info
1 0.000000	Cisco-Li_61:00:d0	Broadcast	802.11	105			Beacon frame, SN=3583, FN=0, Flags=....., BI=100, SSID=MentOrNet
2 0.007098		BelkinIn_63:83:26	(.. 802.11	10			Acknowledgement, Flags=.....
3 0.011195		Senaoint_33:a9:55	(.. 802.11	10			Acknowledgement, Flags=.....
4 0.059323		Senaoint_33:a9:55	(.. 802.11	10			Acknowledgement, Flags=.....
5 0.064957		Senaoint_33:a9:55	(.. 802.11	10			Acknowledgement, Flags=.....
6 0.068024		SonicWAL_53:71:15	(.. 802.11	10			Acknowledgement, Flags=.....

```

> Frame 1: 105 bytes on wire (840 bits), 105 bytes captured (840 bits)
> IEEE 802.11 Beacon frame, Flags: .....
  Type/Subtype: Beacon frame (0x0008)
  <Frame Control Field: 0x8000
    ... ..00 = Version: 0
    ... 00.. = Type: Management frame (0)
    1000 .... = Subtype: 8
  > Flags: 0x00
    .000 0000 0000 0000 = Duration: 0 microseconds
    Receiver address: Broadcast (ff:ff:ff:ff:ff:ff)
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    Transmitter address: Cisco-Li_61:00:d0 (00:23:69:61:00:d0)
    Source address: Cisco-Li_61:00:d0 (00:23:69:61:00:d0)
    BSS Id: Cisco-Li_61:00:d0 (00:23:69:61:00:d0)
    .... ..0000 = Fragment number: 0
    1101 1111 1111 .... = Sequence number: 3583
  <IEEE 802.11 Wireless Management
    > Fixed parameters (12 bytes)
    > Tagged parameters (69 bytes)
  
```

2) How long is the packet capture, from beginning to end (in SECONDS – please round to the nearest full second)?

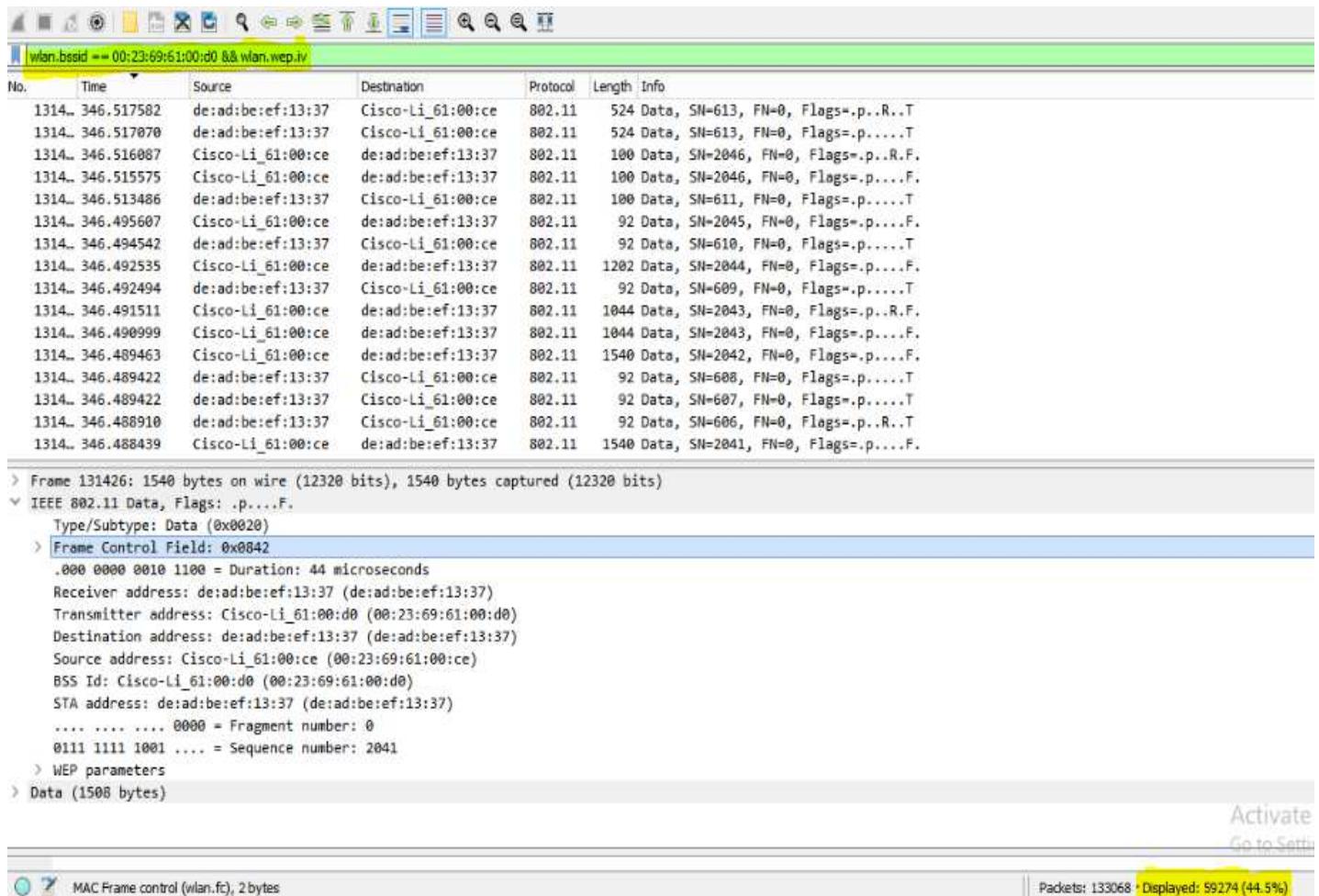
The length of package captured was 133,068 with a maximum capture time in the last packet showing 413,576954. If rounded, the capture time is **414 seconds**

### 3) How many WEP-encrypted data frames are there total in the packet capture?

- Total WEP - encrypted data frames is **59274**

**Steps:** In the Wireshark, in the apply capture filter box, write the command:

**wlan.bssid == 00:23:69:61:00:d0 && wlan.wep.iv** and in the bar present at the bottom of the Wireshark, it will be displayed:



### 4) How many \*unique\* WEP initialization vectors (IVs) are there TOTAL in the packet capture relating to Joe's access point?

- **29719**

```

root@kali:~/media/sf_Shared_Folder# tshark -r accesspointforensics.pcap -Y 'wlan.bssid==00:23:69:61:00:d0 and wlan.wep.iv' -T fields -e wlan.wep.iv | sort -u | wc -l
Running as user "root" and group "root". This could be dangerous.
20719
root@kali:~/media/sf_Shared_Folder# 

```

**Steps:** using terminal in the kali, write the commands:

**tshark -r accesspointforensics.pcap -Y 'wlan.bssid == 00:23:69:61:00:d0 and wlan.wep.iv' -T fields -e wlan.wep.iv | sort -u | wc -l**

## 5) What was the MAC address of the station executing the Layer 2 attacks?

➤ **1c:4b:d6:69:cd:07**

The one with the biggest value and as it is mentioned in the passage that Joey' s connection is dropping consistently.

**Steps:** In the statistics option, go to endpoint and select the Ethernet type to get the Mac address of the station executing the Layer 2 attacks.

No.	Time	Source	Destination	Protocol	Length	Info
1314..	346.517582	de:ad:be:ef:13				
1314..	346.517070	de:ad:be:ef:13				
1314..	346.516087	Cisco-Li_61:00				
1314..	346.515575	Cisco-Li_61:00				
1314..	346.513486	de:ad:be:ef:13				
1314..	346.495607	Cisco-Li_61:00				
1314..	346.494542	de:ad:be:ef:13				
1314..	346.492535	Cisco-Li_61:00				
1314..	346.492494	de:ad:be:ef:13				
1314..	346.491511	Cisco-Li_61:00				
1314..	346.490999	Cisco-Li_61:00				
1314..	346.489463	Cisco-Li_61:00				
1314..	346.489422	de:ad:be:ef:13				
1314..	346.489422	de:ad:be:ef:13				
1314..	346.488910	de:ad:be:ef:13				
1314..	346.488439	Cisco-Li_61:00				
> Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						
IEEE 802.11 Data, Flags: .p....F.						
Type/Subtype: Data (0x0020)						
Frame 131446: 92 bytes on wire (73 bytes captured) (100.00% of 131446)						

6) How many \*unique\* IVs were generated (relating to Joe's access point):

a) By the attacker station?

14133

Steps: in the terminal, write the following commands:

```
tshark -r wireshark.1.pcapng -Y 'wlan.bssid == 00:23:69:61:00:d0 and wlan.wep.iv and wlan.sa== 1c:4b:d6:69:cd:07' -T fields -e wlan.wep.iv |sort -u| wc-l
```

b) By all \*other\* stations combined?:

15587

Steps: In the terminal write the commands as:

```
tshark -r wireshark.1.pcapng -Y 'wlan.bssid == 00:23:69:61:00:d0 and wlan.wep.iv and wlan.sa!= 1c:4b:d6:69:cd:07' -T fields -e wlan.wep.iv |sort -u| wc-l
```

```
root@kali:~/media/sf_Shared_Folder# tshark -r accesspointforensics.pcap -Y 'wlan.bssid==00:23:69:61:00:d0 and wlan.wep.iv' -T fields -e wlan.wep.iv |sort -u| wc-l
Running as user "root" and group "root". This could be dangerous.
29719
root@kali:~/media/sf_Shared_Folder# tshark -r accesspointforensics.pcap -Y 'wlan.bssid==00:23:69:61:00:d0 and wlan.wep.iv and wlan.sa==1c:4b:d6:69:cd:07' -T fields -e wlan.wep.iv |sort -u| wc-l
Running as user "root" and group "root". This could be dangerous.
14133
root@kali:~/media/sf_Shared_Folder# tshark -r accesspointforensics.pcap -Y 'wlan.bssid==00:23:69:61:00:d0 and wlan.wep.iv and wlan.sa !=1c:4b:d6:69:cd:07' -T fields -e wlan.wep.iv |sort -u| wc-l
Running as user "root" and group "root". This could be dangerous.
tshark: "1c:4b:d6:69:cd:07" was unexpected in this context.
0
root@kali:~/media/sf_Shared_Folder# tshark -r accesspointforensics.pcap -Y 'wlan.bssid==00:23:69:61:00:d0 and wlan.wep.iv and wlan.sa !=1c:4b:d6:69:cd:07' -T fields -e wlan.wep.iv |sort -u| wc-l
Running as user "root" and group "root". This could be dangerous.
15587
root@kali:~/media/sf_Shared_Folder# ]
```

## 7) What was the WEP key of Joe's WAP?

➤ D0:E5:9E:B9:04

**Steps:** write the command :

*aircrack-ng accesspointforensics.pcap*

```
Aircrack-ng 1.2 beta1

[00:00:02] Tested 938 keys (got 26805 IVs)

KB    depth   byte(vote)
0    3/  4    D0(33536) 1F(33024) 27(33024) BC(33024) 2F(31744) 7B(31744) F
1    0/  1    E5(38656) 82(33024) 0C(32256) 3C(32000) EB(31744) 42(31488) 3
2    0/  6    9E(34048) 27(33792) 7A(32768) E9(32512) 8B(31744) 0E(31744) 2
3    0/  4    B9(35328) D4(35072) 2E(34048) B9(33024) 00(32768) 06(32512) C
4    8/ 10    6D(31488) 10(31232) B9(31232) 7A(30976) 95(30976) A5(30976) 0

KEY FOUND! [ D0:E5:9E:B9:04 ]
Decrypted correctly: 100%
```

# ANN'S BED MACHINE

## **1. What is the name of Ann's IM buddy?**

Buddy: Sec558user1

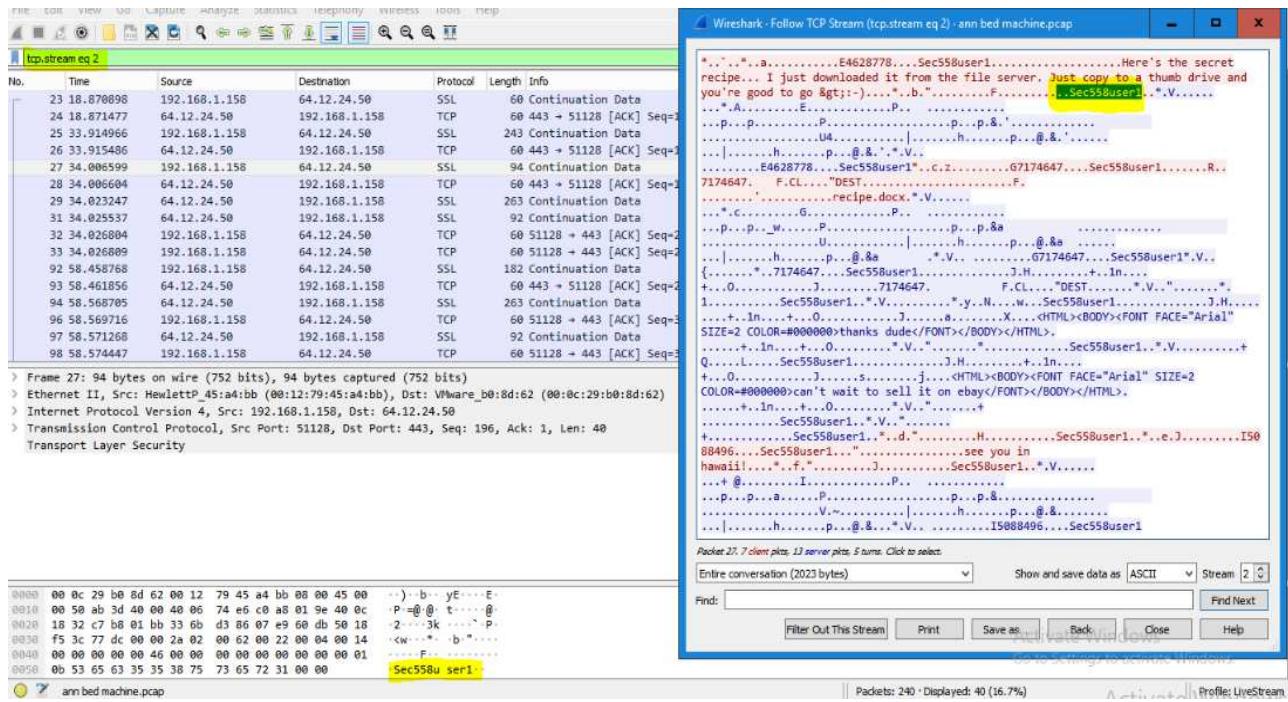
## STEPS:

- In the apply filter option, apply filter *p-cap file with tcp.stream eq 2.*

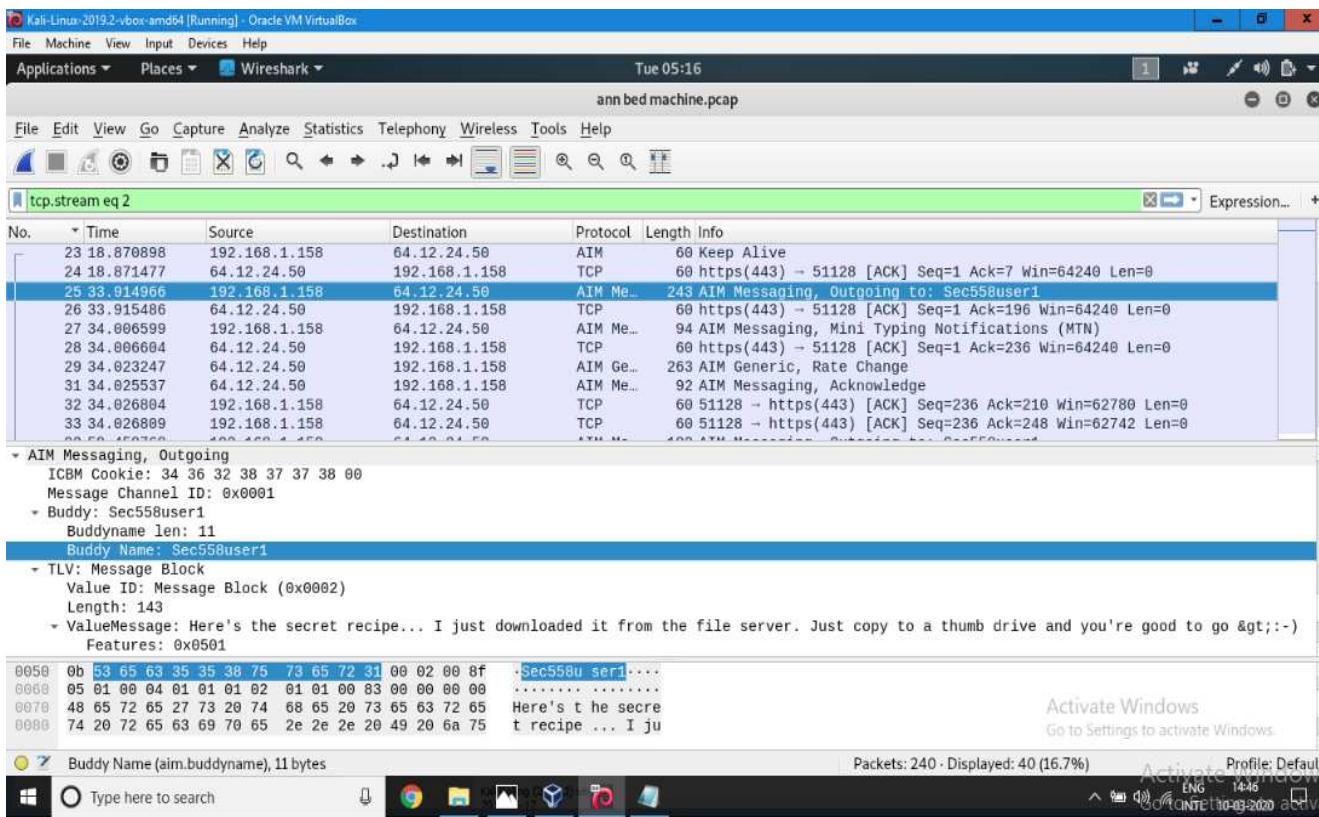
It shows the packets in a session, and then we can sequentially analyze them as we increase the numbers.

- Then we will see in the packet no. 21, 23, 27, 92 and few more showing the continuation of data.
  - Now we will analyze any of them by clicking on follow stream and selecting the tcp stream.

The tcp stream will show the text as shown in screenshot:



- Now we will download and save the message of tcp stream in the kali.
  - Then in the packet detail section, after selecting a particular packet,we will find the message in the **AIM MESSAGING** section and then we will be able to get the name of buddy name as Sec558user1.



## 2. What was the first comment in the captured IM conversation?

**Value Message: Here's the secret recipe... I just downloaded it from the file server. Just copy to a thumb drive and you're good to go &gt;:-)**

STEPS:

- When we analyzed the packet 21 and opened the tcp stream, we were able to see the message as shown in the screenshot .Now we will download the message and again open the message in the wireshark.

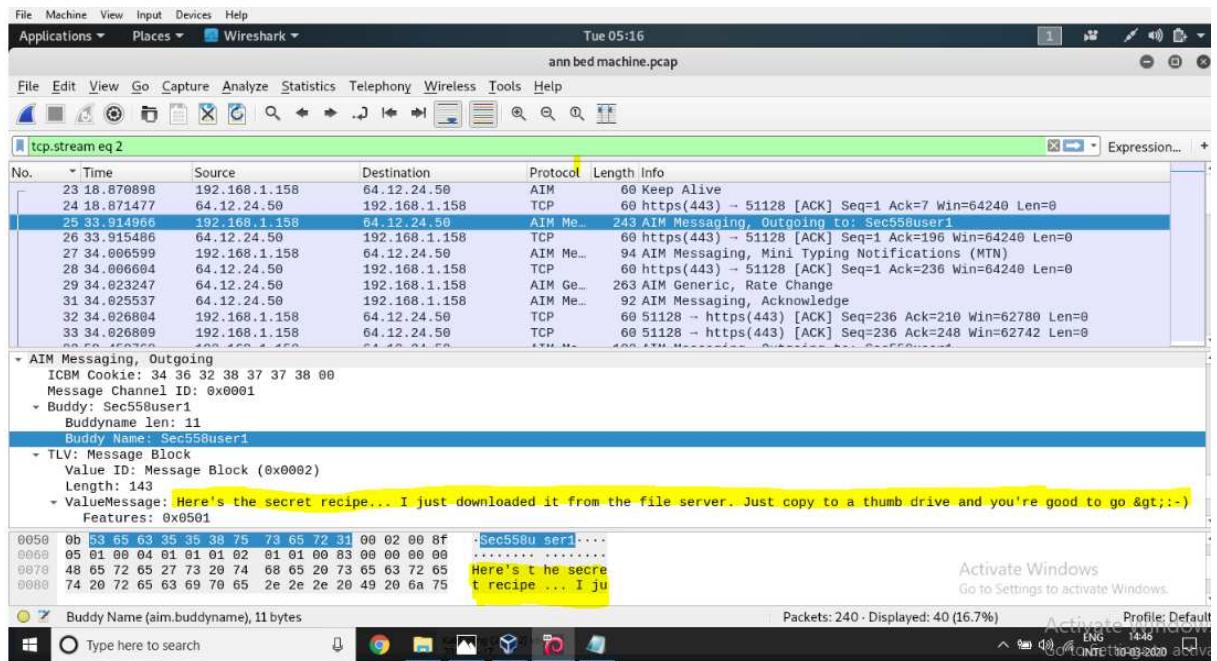
The screenshot shows a Google Docs document titled "recipe". The message "Here's the secret recipe... I just downloaded it from the file server. Just copy to a thumb drive and you're good to go &gt;:-)" is highlighted in yellow. The document content is:

```

* * a E4628778 Sec558user1 Here's the secret recipe... I just
downloaded it from the file server. Just copy to a thumb drive and you're good to go
&gt;:- b F Sec558user1 V
A E P p p P p p &
U4 h p @ & * V
E4628778 Sec558user1 c G7174647 Sec558user1 R 7174647
FCL_DEST F
recipe.docx V
c G P p p w P p p & a
U l h p @ & a
h p @ & * V
G7174647 Sec558user1 V J. 7174647 Sec558user1 J.H. + In +
O J 7174647
FCL_DEST V ^ 1 Sec558user1 V y.N...w Sec558user1 J
H + In + O J a X <HTML><BODY><FONT FACE="Arial" SIZE=2
COLOR=#000000>thanks dude</FONT></BODY></HTML>
+ In + O V Sec558user1 V FONT FACE="Arial" SIZE=2
COLOR=#000000>can't wait to sell it on ebay</FONT></BODY></HTML>
+ In + O V Sec558user1 V d H Sec558user1 e J
15088496 Sec558user1 see you in
hawaii! f J Sec558user1 V
@ I P p p a P p p &
V~ h p @ & V 15088496 Sec558user1

```

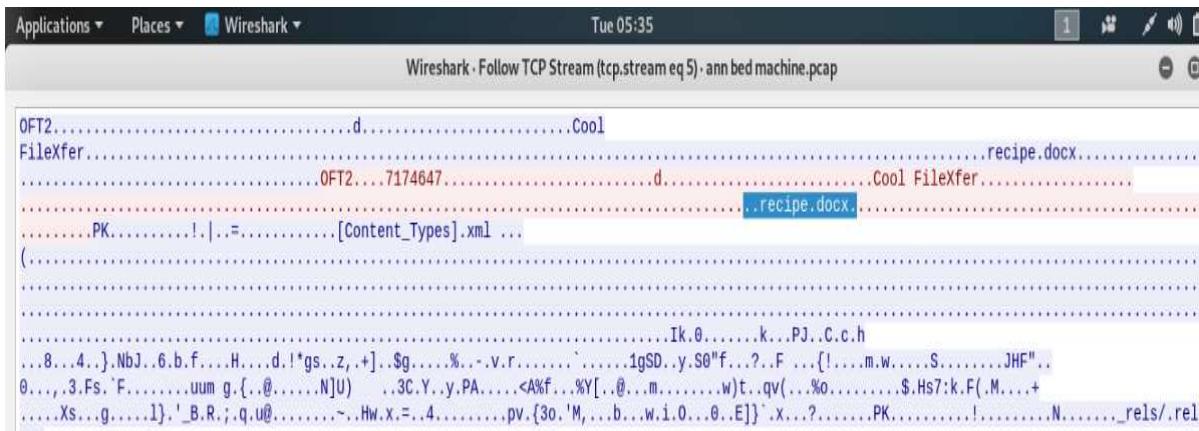
2. Opening the p-cap in the Wireshark. Here in the screenshot I have analyzed the packet 25, in the packet detail section, the highlighted part shows the value message which is the captured in conversation.



### 3. What is the name of the file Ann transferred?

#### ➤ RECEIPE.DOC

Steps: To know the file name transferred, we will go in the statistics option and then in the conversation option we will select the TCP and various tcp packets transferred and port received and delivered. Information is present then we will analyse each packet by clicking on the follow stream and then we will get the name of file transferred. Then by knowing the port no and bytes option in the apply filter option we will apply `tcp.port==5190` and then by searching in the string bytes, we will find the name as `recipe.docx`.



In the packet bytes panel section, we can see the name of the file transferred as shown in the screenshot.

#### 4. What was the MD5sum of the file?

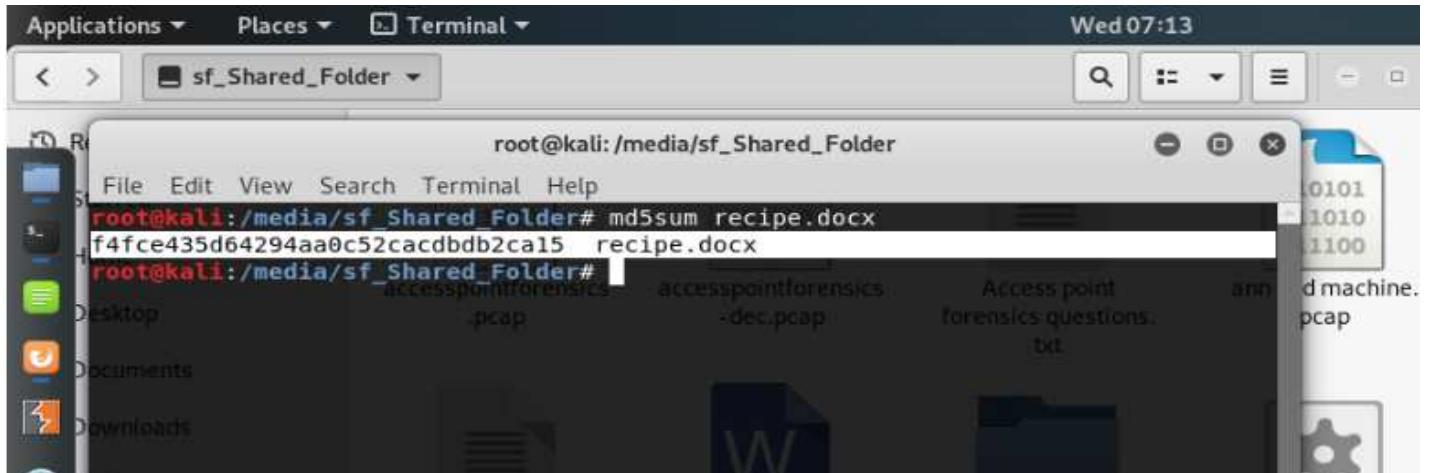
```
TCP payload (256 bytes)
TCP segment data (256 bytes)

00d0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .
00e0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .
00f0 00 00 00 00 00 72 65 63 69 70 65 2e 64 6f 63 ..... re cipe.doc
0100 78 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 x..... .
0110 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .
0120 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .
0130 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ..... .
```

The MD5sum of the file is shown:

Steps: We will download and save the recipe.docx and then in the terminal, write the commands :

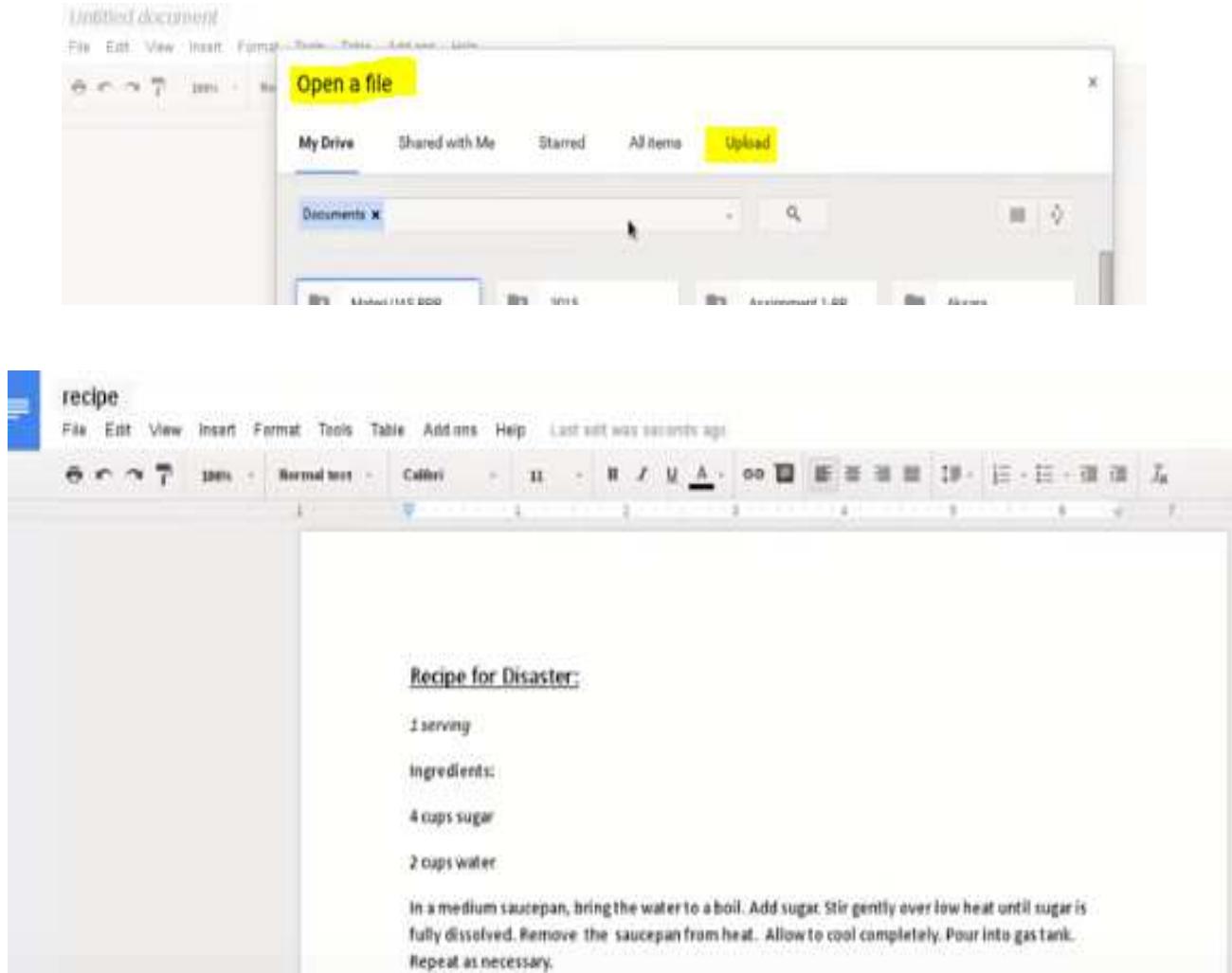
**Md5sum recipe.docx**  
**"f4fce435d64294aa0c52cacdbdb2ca15"**



```
root@kali:/media/sf_Shared_Folder# md5sum recipe.docx
f4fce435d64294aa0c52cacdbdb2ca15  recipe.docx
root@kali:/media/sf_Shared_Folder#
```

#### 5. What is the secret recipe?

To get the secret recipe, open the recipe.docx in the untitled document and upload it:  
The result will be as shown in the figure:



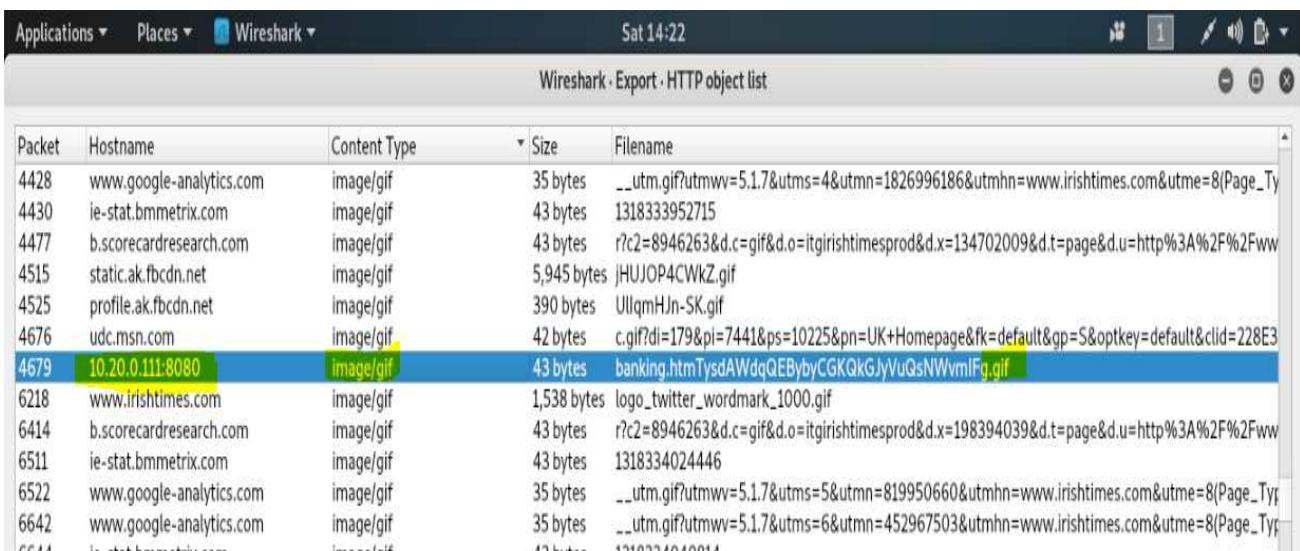
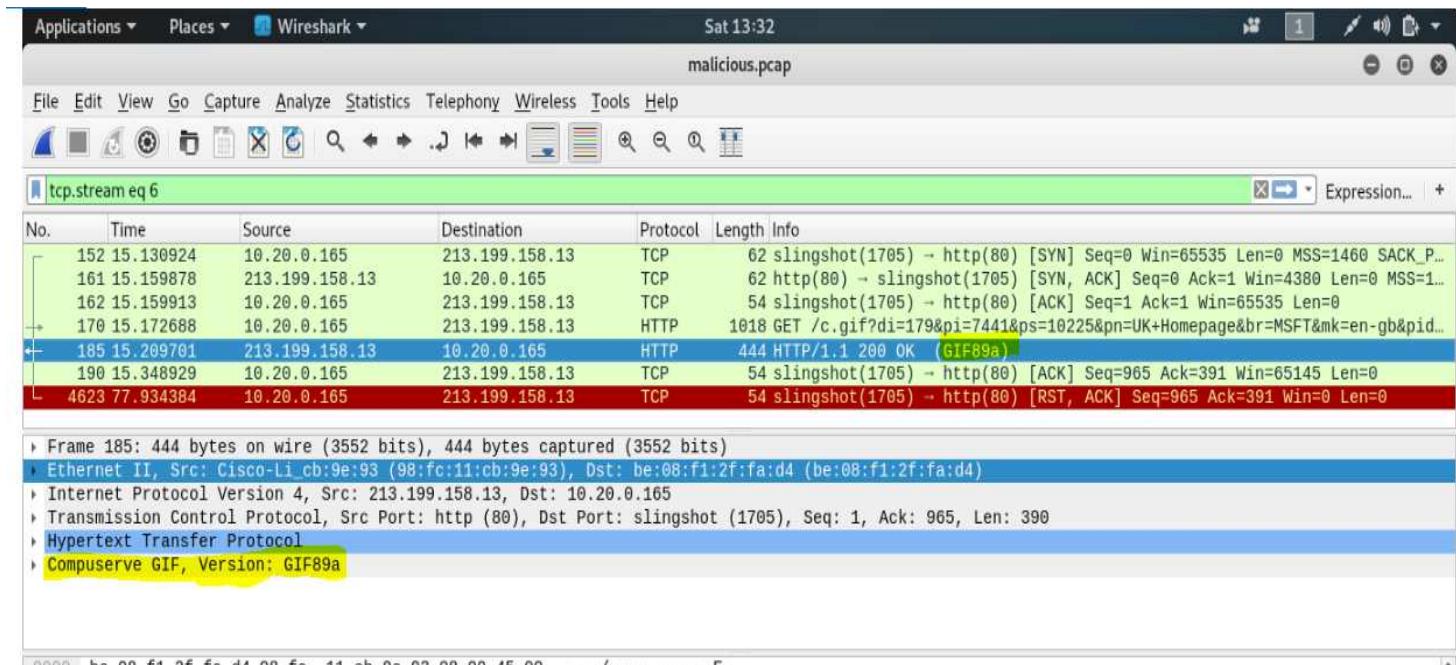
# Malicious.pcap

1. What was the complete URI of the original web request that led to the client being compromised?

➤ **http://10.20.0.111:8080/banking.htm**

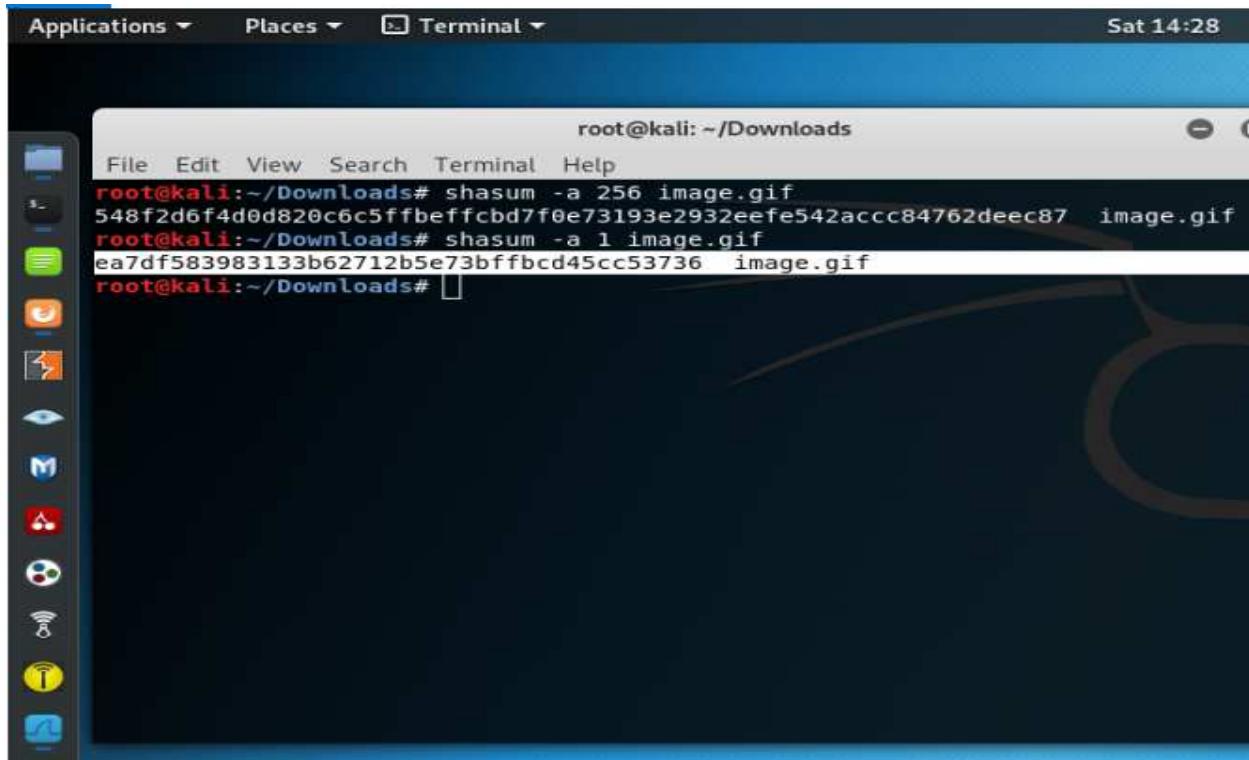
2. What file type was requested in the final web request to the malicious server?

➤ **GIF**



### 3. What is the sha1 hash of the afore-mentioned file?

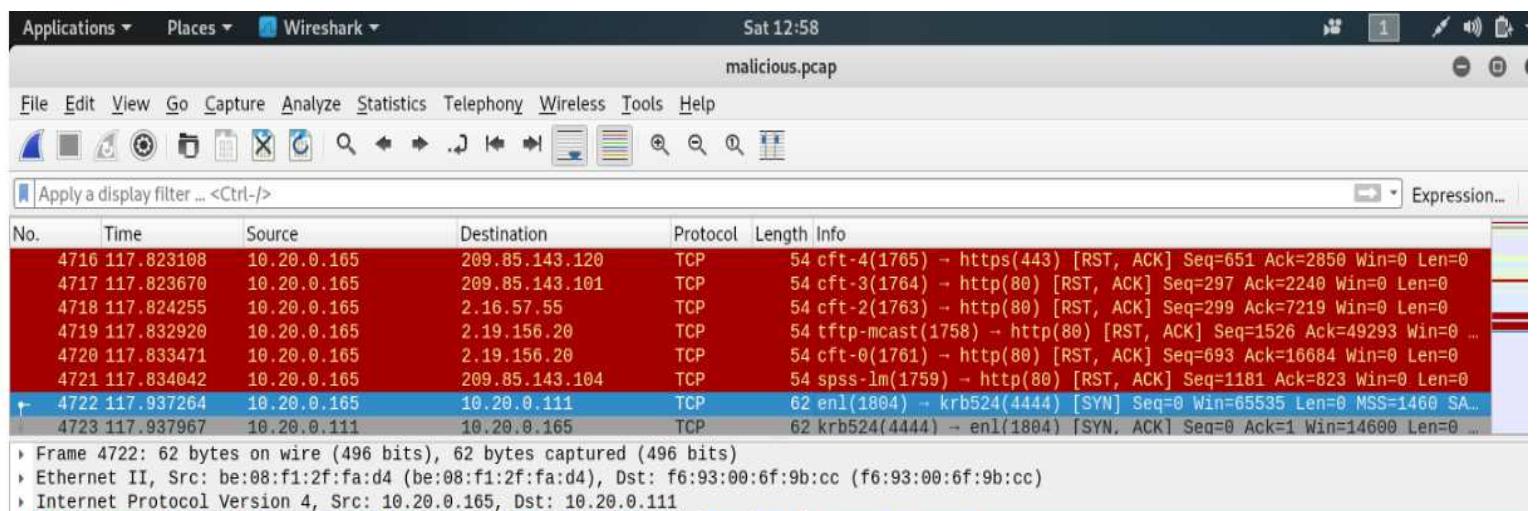
➤ ea7df583983133b62712b5e73bffd45cc53736



```
root@kali: ~/Downloads
File Edit View Search Terminal Help
root@kali:~/Downloads# shasum -a 256 image.gif
548f2d6f4d0d820c6c5ffbeffcbd7f0e73193e2932eefe542accc84762deec87 image.gif
root@kali:~/Downloads# shasum -a 1 image.gif
ea7df583983133b62712b5e73bffd45cc53736 image.gif
root@kali:~/Downloads#
```

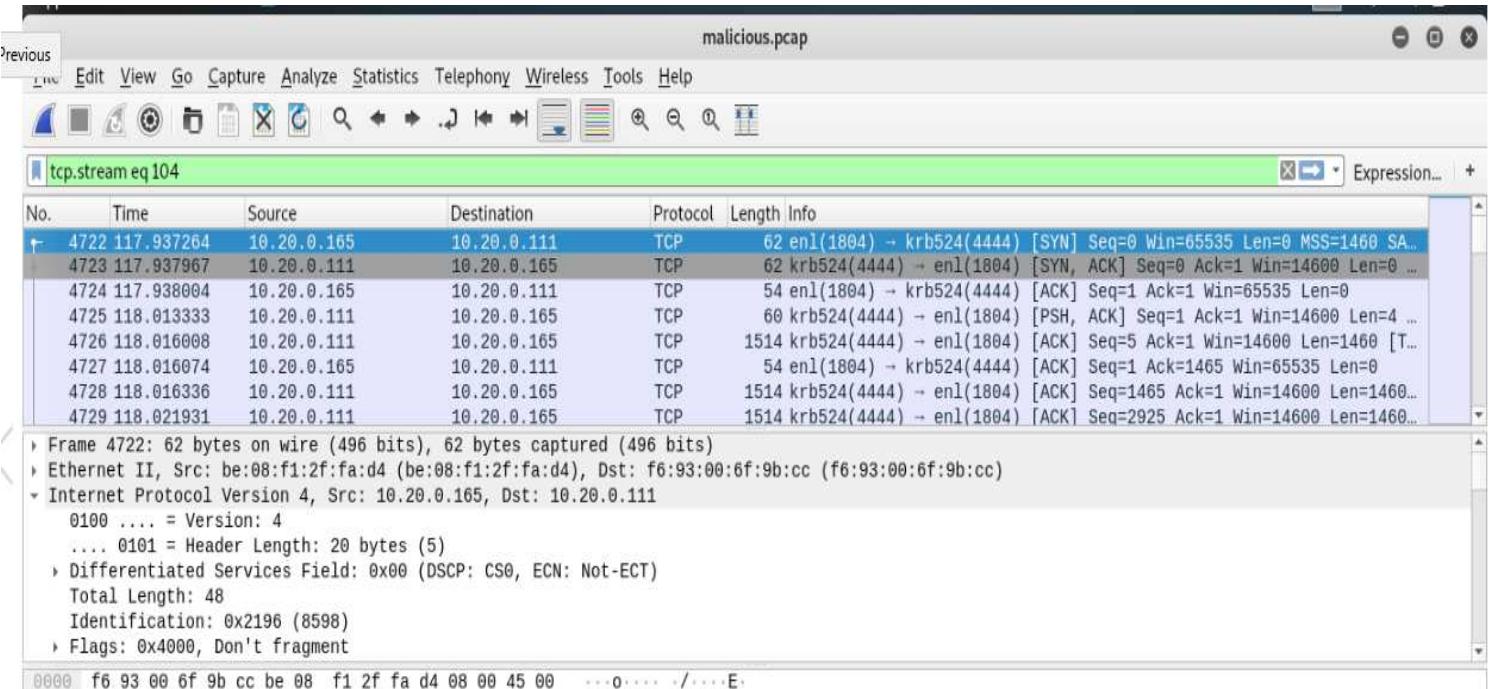
### 4. What is the number of the first frame that indicates that the client has been compromised?

➤ 4722



No.	Time	Source	Destination	Protocol	Length	Info
4716	117.823108	10.20.0.165	209.85.143.120	TCP	54	cft-4(1765) → https(443) [RST, ACK] Seq=651 Ack=2850 Win=0 Len=0
4717	117.823670	10.20.0.165	209.85.143.101	TCP	54	cft-3(1764) → http(80) [RST, ACK] Seq=297 Ack=2240 Win=0 Len=0
4718	117.824255	10.20.0.165	2.16.57.55	TCP	54	cft-2(1763) → http(80) [RST, ACK] Seq=299 Ack=7219 Win=0 Len=0
4719	117.832920	10.20.0.165	2.19.156.20	TCP	54	tftp-mcast(1758) → http(80) [RST, ACK] Seq=1526 Ack=49293 Win=0 Len=0
4720	117.833471	10.20.0.165	2.19.156.20	TCP	54	cft-0(1761) → http(80) [RST, ACK] Seq=693 Ack=16684 Win=0 Len=0
4721	117.834042	10.20.0.165	209.85.143.104	TCP	54	spss-lm(1759) → http(80) [RST, ACK] Seq=1181 Ack=823 Win=0 Len=0
4722	117.937264	10.20.0.165	10.20.0.111	TCP	62	enl(1804) → krb524(4444) [SYN] Seq=0 Win=65535 Len=0 MSS=1460 SA...
4723	117.937967	10.20.0.111	10.20.0.165	TCP	62	krb524(4444) → enl(1804) [SYN, ACK] Seq=0 Ack=1 Win=14600 Len=0

Frame 4722: 62 bytes on wire (496 bits), 62 bytes captured (496 bits)  
Ethernet II, Src: be:08:f1:2f:fa:d4 (be:08:f1:2f:fa:d4), Dst: f6:93:00:6f:9b:cc (f6:93:00:6f:9b:cc)  
Internet Protocol Version 4, Src: 10.20.0.165, Dst: 10.20.0.111



## PORt 4444 – Information

- Port Number: 4444
- TCP / UDP: TCP
- Delivery: Yes
- Protocol / Name: [Malware known as CrackDown]  
krb524.nv-video.eggdrop
- Port Description: [malware info: CrackDown]  
Common for eggdrop bot

Kbr524 (KNet Web Server) is vulnerable to a buffer overflow. By sending a specially-crafted request to **TCP port 4444**, containing an overly long string argument, a remote attacker could overflow a buffer and execute arbitrary code on the system or cause the server to crash. And in the screenshot we can see it very clearly that the packet above that packet i.e. 4721 is http port (80), thus this packet no 4722 is the first frame that indicates that the client has been compromised.

## 5. At one point, the malicious server sends a malicious file to the client. What type of file is it?

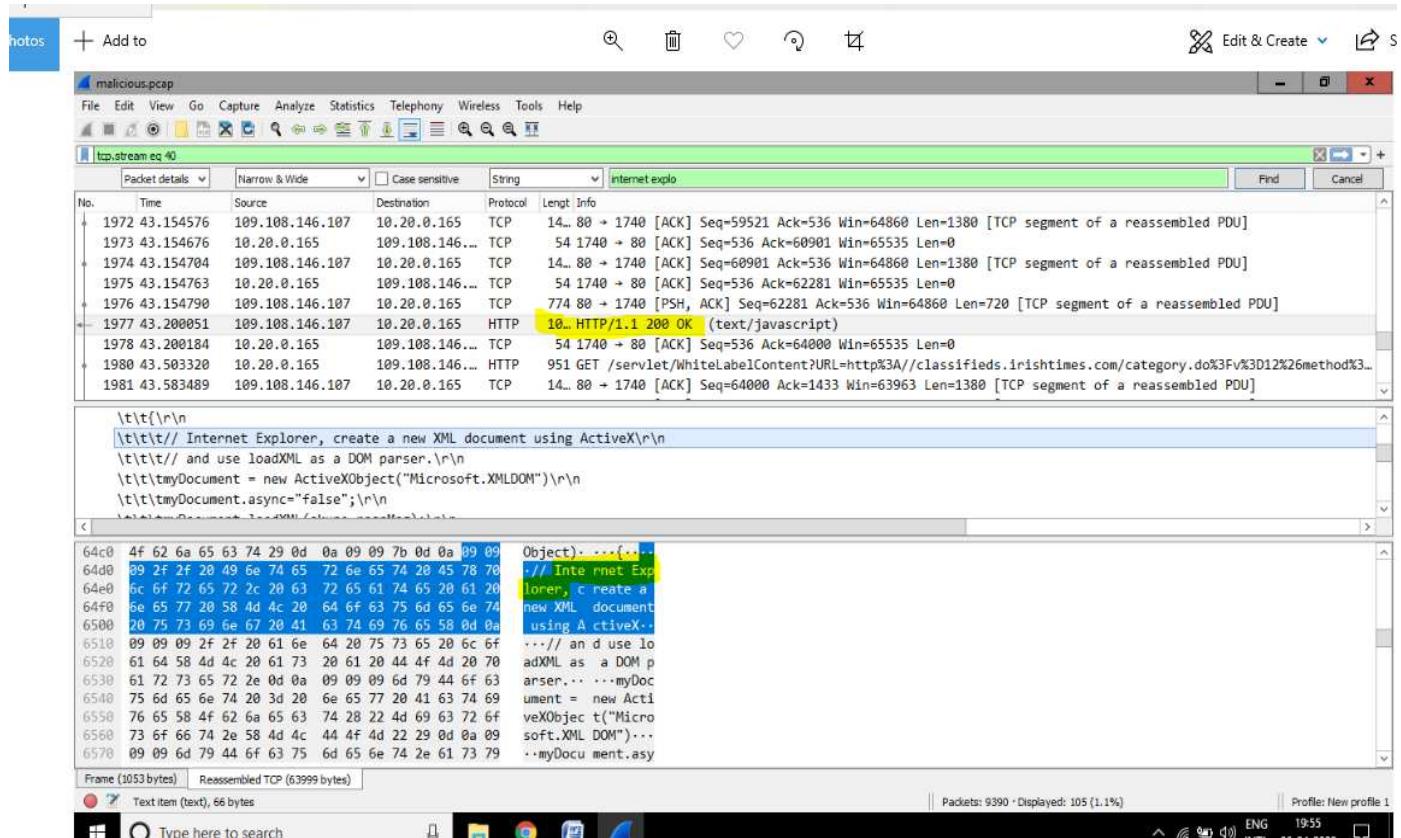
- Window exe file with written as MZ

## 6. What is the sha1 hash of the malicious file?

- **94adf100411a80076192766a214e0ff92da13ab7**

## 7. What vulnerable software is exploited?

- **Internet explorer(ie6)**



## 8. Can you give the corresponding CVE security bulletin that covers the vulnerability here that was exploited (answer in form of CVE-\$year-\$number).

- **CVE-2010-0249**

**CVE-2010-0249 : Use-after-free vulnerability in Microsoft Internet Explorer 6, 6 SP1, 7, and 8 on Windows 2000 SP4; Windows XP SP2.**

- Number Of Affected Versions By Product

Vendor	Product	Vulnerable Versions
Microsoft	Internet Explorer	4

- References For CVE-2010-0249

<https://exchange.xforce.ibmcloud.com/vulnerabilities/55642>

XF ie-freed-object-code-execution(55642)

<http://www.kb.cert.org/vuls/id/492515>

CERT-VN VU#492515

<https://docs.microsoft.com/en-us/security-updates/securitybulletins/2010/ms10-002>

MS MS10-002

<http://technet.microsoft.com/en-us/security/advisory/979352>

Microsoft Advisory <http://www.microsoft.com/technet/security/advisory/979352.mspx> Microsoft Security Advisory (979352

9. From the capture, it is clear that the attacker gets a certain form of access (i.e. the interface), what (type of) access does the attacker "get" on the client?

➤ Shell

An *attacker* can use a web *shell* to issue commands, perform privilege escalation on the web server, and the ability to upload, delete, download and execute files.

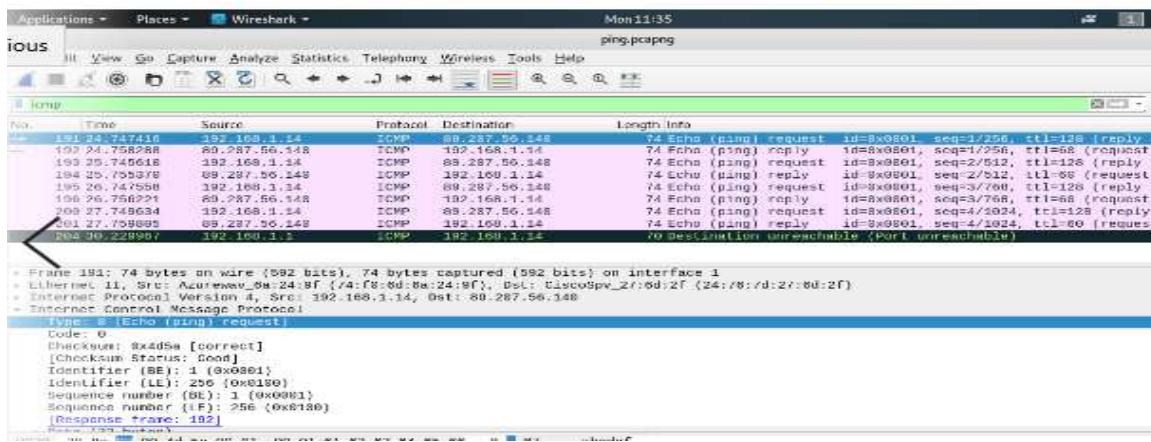
The screenshots shows the list of interfaces which the attacker get access and uses them to attack the client using shell.

# Ping.pcap

## 1. What type of ICMP traffic is shown in this capture?

- Echo Request/Reply

The packet capture displays the type of packet (echo request/reply) in the 'info' column. I looked at the header information and determined the type of ICMP message from the first field 'Type'.

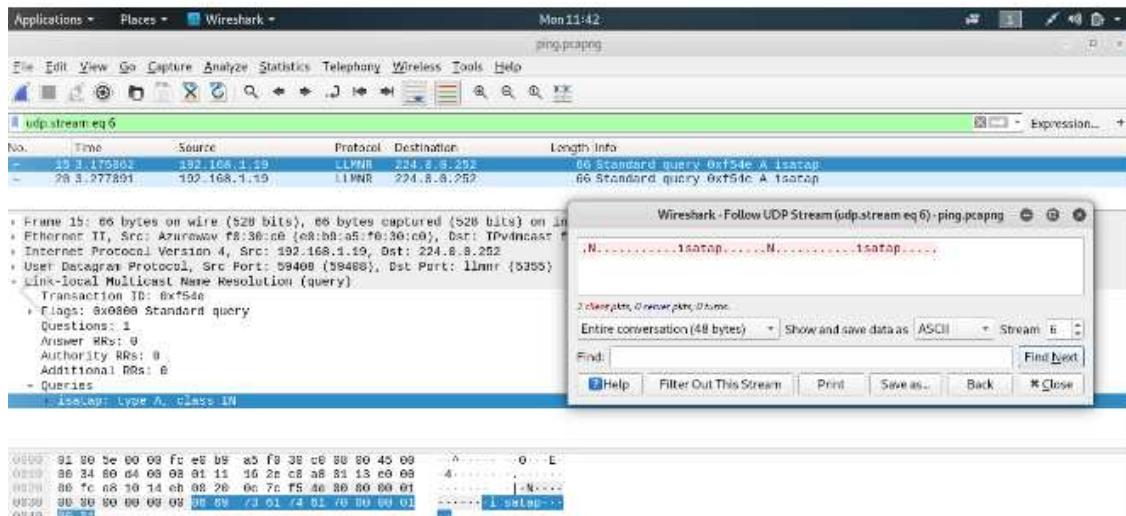


## 2. What is the number of the first frame that indicates that something funny might be going on?

- 15 NO FRAME

*Frame no. 15 is the first frame that suggests something funny might be going on.*

Explanation: As I was looking for the hidden protocol, I noticed from the hex view that frame no. 15 was the first frame that indicated the beginning of an SSH connection. The host 192.168.5.217 kept sending ICMP reply packets every 1 or 2 seconds till frame 26 without waiting for a request which seemed strange.



**15 (Unprompted reply with suspicious content - SSH-2.0-OpenSSH\_5.3p1 Debian-3ubuntu6...)**

### 3. What is the application layer protocol that is hidden within the ICMP traffic?

- The *SSH protocol* (also referred to as Secure Shell) is a method for secure remote login from one computer to another. It provides several alternative options for strong authentication, and it protects the communications security and integrity with strong encryption.

### 4. What tool most likely generated this 'malicious' traffic?

#### ➤ **ICMP Tunnel**

*The tool that most likely generated this traffic could be Shell over ICMP or Ping Tunnel.*

Explanation: The tool that generated this traffic must support SSH traffic hidden within ICMP traffic also known as ICMP tunneling. I searched 'ssh within icmp' on Google. It allows a user to connect to a remote shell daemon, by using ICMP protocol instead of classical TCP. Another tool I came across is **Ping Tunnel** that holds a BSD license. It is used for sending TCP traffic over ICMP using the echo request/reply pings.

No.	Time	Source	Protocol	Destination	Length Info
181	24.747416	192.168.1.14	ICMP	89.287.56.148	74 Echo (ping) request id=8x0801, seq=1/256, ttl=128 (re)
192	24.758288	89.287.56.148	ICMP	192.168.1.14	74 Echo (ping) reply id=8x0801, seq=1/256, ttl=68 (re)
193	25.745616	192.168.1.14	ICMP	89.287.56.148	74 Echo (ping) request id=8x0801, seq=2/512, ttl=128 (re)
194	25.755376	89.287.56.148	ICMP	192.168.1.14	74 Echo (ping) reply id=8x0801, seq=2/512, ttl=68 (re)
195	26.747558	192.168.1.14	ICMP	89.287.56.148	74 Echo (ping) request id=8x0801, seq=3/768, ttl=128 (re)
196	26.756221	89.287.56.148	ICMP	192.168.1.14	74 Echo (ping) reply id=8x0801, seq=3/768, ttl=68 (re)
200	27.749634	192.168.1.14	ICMP	89.287.56.148	74 Echo (ping) request id=8x0801, seq=4/1024, ttl=128 (re)
201	27.758095	89.287.56.148	ICMP	192.168.1.14	74 Echo (ping) reply id=8x0801, seq=4/1024, ttl=68 (re)
204	30.228967	192.168.1.1	ICMP	192.168.1.14	70 Destination unreachable (Port unreachable)

Frame 181: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on interface 1  
 Ethernet II, Src: Azuraway\_5a:24:b1 (74:f8:6d:24:b1), Dst: CiscoSpu\_27:6d:21 (24:76:7d:27:6d:21)

5. What is the 'true' destination of the ICMP traffic generated from 192.168.5.208?

➤ **172.16.15.138**

6. What is the session identifier for each packet? (answer in hex, 2 bytes

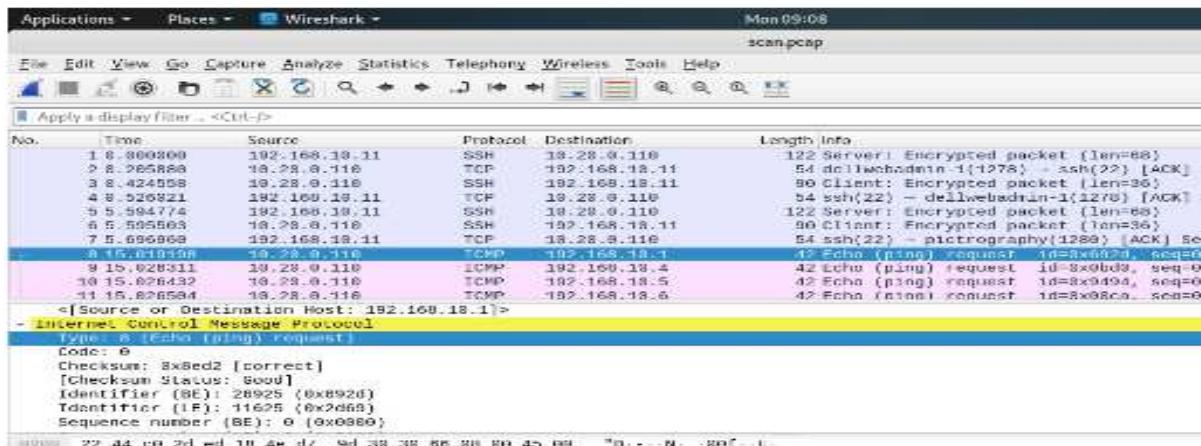
➤ **e59c**

The session identifiers could be easily determined by looking at the 'info' field. These identifiers are displayed in ICMP header information as both big endian and little endian formats.

Frame Range	Session Identifier (big endian representation)
1 - 12	0x0754
13 - 34	0xe59c
35 - 78	0xc7cc
79 - 306	0x8f5b

# Scan.pcap

## 1. What tool is generating this traffic?



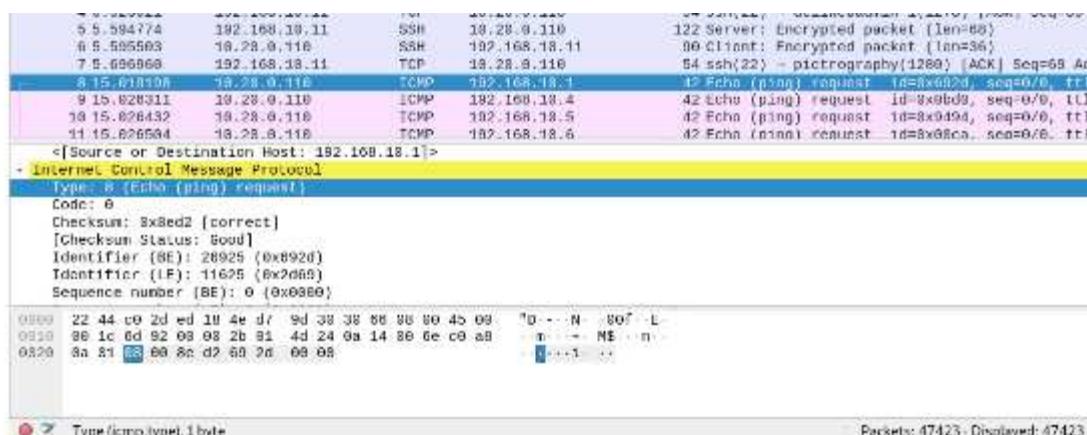
➤ ICMP tool

## 2. What is the frame that indicates something strange might be going on?

➤ Frame 8

Frame no. 8 indicates something strange might be going on.

**Explanation:** I identified the malicious host as 10.20.0.110 from NetworkMiner. The first few frames were TCP and SSH request and replies. The 8th frame however was an ICMP ping request from the host 10.20.0.110 to host 198.168.10.1. The subsequent frames showed that the malicious host had sent ICMP ping requests to multiple ip addresses on the same network without receiving replies from every address.

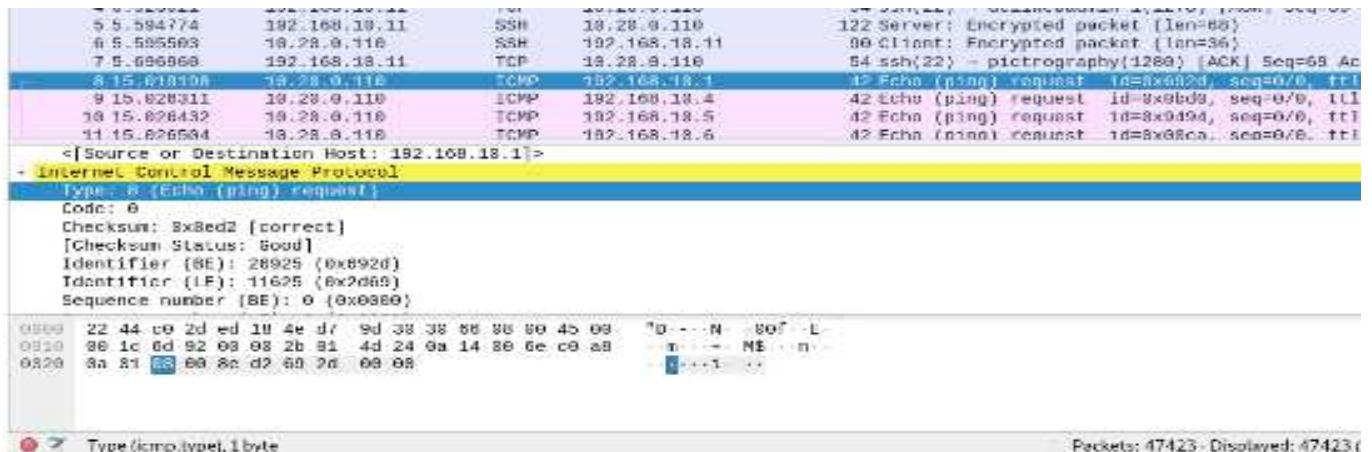


Packets: 47423 - Displayed: 47423 (

### 3. What does this frame constitute the beginning of? (What type of Scan?)

➤ Ping Scan

**Explanation:** According to nmap.org, this is one of the methods for Host Discovery to identify active or interesting hosts on the network. Therefore, the attacking host 10.20.0.110 was clearly trying to identify the active hosts by sending ICMP ping requests or doing the ping scan.



### 4. The 'miscreant' then runs two scans beginning just after six minutes and 24 minutes into the trace, however, these traces weren't to his/her liking as they were too slow. On the following scans, a switch was removed from the command, what was this switch (just the letters, case-sensitive)?

➤ Su is the name of the switch

I applied the filter 'ip.addr == 10.20.0.110' and examined the pcap file in Wireshark. The two scans began after frame no. 16719 (time = 360 secs approx.) and frame no. 25519 (time = 1440 secs approx.). The packets captured after 360 secs showed that the miscreant had sent UDP packets and therefore it was performing a UDP scan this time. I was not clear what a command switch were so I searched 'namp command switches' on google. I found out that they were parameters used to perform specific type of scans [2]. The option listed for UDP scan was –sU. In order to find out more about the option, I visited the nmap.org and looked for UDP scan.

## 5. What switch was added to the final scan (case-sensitive)?

➤ sS

**Explanation:** I applied the filter 'ip.addr == 10.20.0.110' and examined the pcap file in Wireshark. On examining the packets after 1440 secs, I noticed that the protocol used was TCP. The first two steps of the TCP handshake were performed between the miscreant 10.20.0.110 and many other hosts. For instance, Frame no. 34051 shows a SYN packet sent from host 10.20.0.110 to 192.168.10.10. The victim machine replies with a SYN/ACK packet according to frame no. 34062. But, instead of sending an ACK packet, the miscreant sends a RST packet to terminate the connection before it is fully established, as it now knows that the port is open. This process is followed in the SYN Stealth scan by nmap according to the same document that I looked up for the previous question. Therefore, the miscreant performs the final scan as the SYN stealth scan .This confirmed that the switch added was -sS.

Frame ID	Source IP	Destination IP	Protocol	Details
47210	1850.972425	192.168.10.11	10.20.0.110	SSH
47211	1850.972787	10.20.0.110	192.168.10.11	SSH
47212	1851.074305	192.168.10.11	10.20.0.110	TCP
47421	1865.975777	192.168.10.11	10.20.0.110	SSH
47422	1865.976041	10.20.0.110	192.168.10.11	SSH
47423	1866.077780	192.168.10.11	10.20.0.110	TCP

.... 0.... = Congestion Window Reduced (CWR): Not set  
.... .0.... = ECN-Echo: Not set  
.... ..0.... = Urgent: Not set  
.... ...1.... = Acknowledgment: Set  
.... ....0... = Push: Not set  
.... ....0... = Reset: Not set  
.... ....0. = Syn: Not set  
.... ....0.= Fin: Not set  
[TCP Flags: ....A....]  
Window size value: 65535  
[Calculated window size: 65535]  
[Window size scaling factor: -1 (unknown)]

```
0000: 4e d7 9d 30 30 66 22 44 c0 2d ed 18 08 00 45 10 N--00f"0 -----E-
0010: 00 28 fa 80 40 00 3f 06 6c 0a c0 a8 0a 0b 0a 14 -(.-@?-. 1-----
0020: 00 6e 00 16 05 00 df 54 32 dd c4 60 c1 e1 50 10 n-----T 2...*..p.
0030: ff ff 3d 15 00 00 .....
```

## 2018 – 02 – 13 Traffic Analysis "OFFICE WORK"

Review the pcap, and document the following:

- Date and time of the malicious activity in UTC (GMT).
- IP address of the affected Windows host.
- Mac address of the affected Windows host.
- Host name of the affected Windows host.
- User account name on the affected Windows host.
- What malware might be involved.

### **ANSWERS:**

- Date/Time: **2018-02-13 at approximately 05:06 UTC**
- IP address: **10.23.1.205**
- Mac address: **00:16:17:f9:42:e5 (Msi\_f9:42:e5)**
- Host name: **REGINALD-PC**
- User account name: **reginald.farnsworth**
- What malware might be involved: **DarkComet RAT**

### **DETAILS**

---

User account **reginald.farnsworth** logged into his Windows client **REGINALD-PC** through a domain controller for **moondustries.com**. The associated IP addresses are:

- Windows client (REGINALD-PC): 10.23.1.205
- Domain controller for moondustries.com: 10.23.1.7
- Broadcast address for this LAN segment: 10.23.1.255
- Gateway for this LAN segment: 10.23.1.1

For Reginald's IP address, Mac address, and host name, filter on ***nbns*** and we'll find our answers as shown in the image below:

Time	Src	port	Dst	port	Info
2018-02-13 05:06:14	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<00>
2018-02-13 05:06:14	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<20>
2018-02-13 05:06:14	10.23.1.205	137	10.23.1.255	137	Registration NB MOONDUSTRIES<00>
2018-02-13 05:06:14	10.23.1.205	137	10.23.1.255	137	Registration NB MOONDUSTRIES<00>
2018-02-13 05:06:14	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<20>
2018-02-13 05:06:14	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<00>
2018-02-13 05:06:15	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<00>
2018-02-13 05:06:15	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<20>
2018-02-13 05:06:15	10.23.1.205	137	10.23.1.255	137	Registration NB MOONDUSTRIES<00>
2018-02-13 05:06:16	10.23.1.205	137	10.23.1.255	137	Registration NB MOONDUSTRIES<00>
2018-02-13 05:06:16	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<20>
2018-02-13 05:06:16	10.23.1.205	137	10.23.1.255	137	Registration NB REGINALD-PC<00>

```

Frame 158: 110 bytes on wire (880 bits), 110 bytes captured (880 bits)
Ethernet II, Src: MSI-f9:42:e5 (00:16:17:f9:42:e5), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Internet Protocol Version 4, Src: 10.23.1.205, Dst: 10.23.1.255
User Datagram Protocol, Src Port: 137, Dst Port: 137
NetBIOS Name Service
    Transaction ID: 0x85e7
    Flags: 0x2910, Opcode: Registration, Recursion desired, Broadcast
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
    Queries
    Additional records
        REGINALD-PC<00> - type NB, class IN
            Name: REGINALD-PC<00> (Workstation/Redirector)
            Type: NB (32)
            Class: IN (1)
            Time to live: 3 days, 11 hours, 20 minutes
            Data length: 6
        Name flags: 0x4000, QNT: Unknown (M-node, unique)
            Addr: 10.23.1.205

```

The user account name can be found through Kerberos traffic generated when Reginald logged into his Windows client. To find the user account name, we use the following Wireshark filter:

**kerberos.CNameString and !(kerberos.CNameString contains \$)**

In the results, work our way down to the ***cname*** field and find the user account name as shown below:

kerberos.CNameString and !(kerberos.CNameString contains \$)

Time	Src	port	Dst	port	Info
2018-02-13 05:06:50	10.23.1.205	49181	10.23.1.7	88	AS-REQ
2018-02-13 05:06:50	10.23.1.205	49192	10.23.1.7	88	AS-REQ
2018-02-13 05:06:50	10.23.1.7	88	10.23.1.205	49182	AS-REP
2018-02-13 05:06:50	10.23.1.7	88	10.23.1.205	49183	TGS-REP
2018-02-13 05:06:50	10.23.1.7	88	10.23.1.205	49185	TGS-REP
2018-02-13 05:06:51	10.23.1.7	88	10.23.1.205	49187	TGS-REP
2018-02-13 05:06:51	10.23.1.7	88	10.23.1.205	49190	TGS-REP
2018-02-13 05:06:51	10.23.1.7	88	10.23.1.205	49191	TGS-REP

Frame 367: 297 bytes on wire (2376 bits), 297 bytes captured (2376 bits)  
 ▶ Ethernet II, Src: Msi\_f9:42:e5 (00:16:17:f9:42:e5), Dst: Dell\_53:d4:1b  
 ▶ Internet Protocol Version 4, Src: 10.23.1.205, Dst: 10.23.1.7  
 ▶ Transmission Control Protocol, Src Port: 49181, Dst Port: 88, Seq 1, A  
 ▶ Kerberos  
 ▶ Record Mark: 239 bytes  
 ▶ as-req  
 ▷ ptype: 5  
 ▷ msg-type: krb-as-req (10)  
 ▶ padata: 1 item  
 ▶ req-body  
 ▷ Padding: @  
 ▷ kdc-options: 40810010 (forwardable, renewable, canonicalize, render)  
 ▷ cname  
 ▷ name-type: KRB5-NT-PRINCIPAL (1)  
 ▷ cname-string: 1 item  
 ▷ CNameString: reginald.farnsworth  
 ▷ realm: MOONDUSTRIES  
 ▶ sname

How can we find out the alerts? We can check the pcap on VirusTotal and PacketTotal. Both show alerts for the DarkComet RAT.

In the VirusTotal analysis of the pcap, we'll find alerts for DarkComet RAT under both the Snort and the Suricata alerts in the "File detail" section.

- <https://www.virustotal.com/en/file/88413b71e5e2836f8686b3390c2d802d1a0c3de33b510bcfd1adc2b18ff07eb3/analysis/>

PacketTotal analysis of the pcap also shows several alerts for DarkComet on traffic to 185.86.151.37 over TCP port 2200:

- <https://packettotal.com/app/analysis?id=6f4a5f6d7b3c4af88577fa79f8aa105d>

 VirusTotal

virustotal.com/gui/file/88413b71e5e2836f8686b3390c2d802d1a0c3de33b510bcfd1adc2b18ff07eb3/details

**Snort Alerts**

- + Potentially Bad Traffic
- + Executable code was detected
- + Potential Corporate Privacy Violation
- + Attempted User Privilege Gain
- A Network Trojan was detected

MALWARE-CNC Win.Trojan.Darkkomet variant inbound connection [25229]  
 MALWARE-CNC Win.Trojan.Darkkomet variant outbound connection [25230]  
 MALWARE-CNC Win.Trojan.Darkcomet outbound keepalive signal sent [31814]

**Suricata Alerts**

- + Not Suspicious Traffic
- + Potentially Bad Traffic
- A Network Trojan was Detected

ET TROJAN Backdoor.Win32.DarkComet Screenshot Upload Successful [2021996]  
 ETPRO TROJAN DarkComet-RAT init connection 2 [2806577]  
 ETPRO TROJAN DarkComet-RAT server join acknowledgement 2 [2806578]  
 ETPRO TROJAN DarkComet-RAT activity [2807821]  
 ETPRO TROJAN Backdoor.Win32.DarkKomet Keep-Alive [2809530]

## Snort Alert

- **MALWARE-CNC Win.Trojan.Darkkomet variant inbound connection [25229]**
- **MALWARE-CNC Win.Trojan.Darkkomet variant outbound connection [25230]**
- **MALWARE-CNC Win.Trojan.Darkcomet outbound keepalive signal sent [31814]**

## Suricata Alert

- **ET TROJAN Backdoor.Win32.DarkComet Screenshot Upload Successful [2021996]**
- **ETPRO TROJAN DarkComet-RAT init connection 2 [2806577]**
- **ETPRO TROJAN DarkComet-RAT server join acknowledgement 2 [2806578]**
- **ETPRO TROJAN DarkComet-RAT activity [2807821]**
- **ETPRO TROJAN Backdoor.Win32.DarkKomet Keep-Alive [2809530]**

tcp.flags eq 0x0002 and p.adcr eq 185.86.151.37						Expression..	+
Time	Src	port	Dst	port	Info		
2018-02-13 05:07:43	10 23.1.205	49212	185.86.151.37	2200	49212 → 2200		
2018-02-13 05:08:14	10 23.1.205	49213	185.86.151.37	2200	49213 → 2200		
2018-02-13 05:08:16	10 23.1.205	49214	185.86.151.37	2200	49214 → 2200		

*Shown above: You'll find three TCP streams using the Wireshark filter `tcp.flags eq 0x0002` and `ip.addr eq 185.86.151.37`*

Follow any one of the TCP streams shown in the above image, and we'll see what DarkComet traffic looks like.

The screenshot shows a Wireshark window with the following details:

- Filter Bar:** Shows the filter `tcp.stream eq 54`.
- Table Headers:** Time, Src, port, Dst, port, Info.
- Table Data:** Five rows of network traffic. The first row is expanded to show the raw hex and ASCII data.
- Panel:** The bottom panel displays the raw hex and ASCII representation of the selected packet's payload. The ASCII output is filled with numerous red characters, indicating encrypted or heavily obfuscated data.

Time	Src	port	Dst	port	Info
2018-02-13 05:07:43	10.23.1.205	49212	185.86.151.37	2200	49212 → .
2018-02-13 05:07:43	185.86.151..	2200	10.23.1.205	49212	2200 → .
2018-02-13 05:07:43	10.23.1.205	49212	185.86.151.37	2200	49212 → .
2018-02-13 05:07:43	185.86.151..	2200	10.23.1.205	49212	2200 → .
2018-02-13 05:07:43	10.23.1.205	49212	185.86.151.37	2200	49212 → .

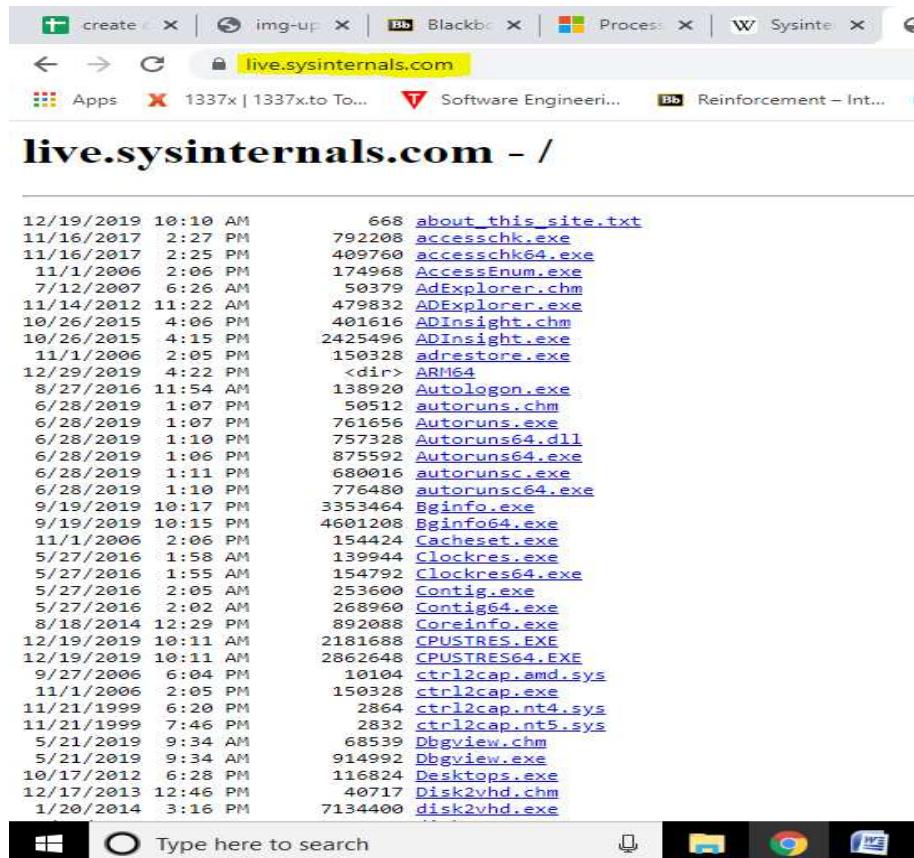
*Shown above: A TCP stream of DarkComent RAT traffic from this infection.*

## EXPERIMENT 6

### **Using Sys-internals tools for Network Tracking and Process Monitoring**

**Windows Sysinternals** is a website which offers technical resources and utilities to manage, diagnose, troubleshoot, and monitor a Microsoft Windows environment. The tools include utilities such as Process Explorer, which is a lot like Task Manager with a plethora of extra features, or Process Monitor, which monitors your PC for file-system, registry, or even network activity from almost any process on your system. Most of these tools are going to require administrator access on your computer, so you'd be wise to test them out in a virtual machine or a test computer if you aren't sure what you are doing — these are some heavy duty tools.

- **Check Sysinternals tools :** To check sysinternals tool, browse on the google and go to the website:  
<https://live.sysinternals.com/>, there we can see all the tools:



The screenshot shows a web browser window with the URL 'live.sysinternals.com' highlighted in the address bar. Below the address bar, the browser's toolbar includes icons for back, forward, search, and refresh. The main content area displays a table of system files:

Date	File Size	File Name
12/19/2019 10:10 AM	668	<a href="#">about_this_site.txt</a>
11/16/2017 2:27 PM	792208	<a href="#">accesschk.exe</a>
11/16/2017 2:25 PM	409760	<a href="#">accesschk64.exe</a>
11/1/2006 2:06 PM	174968	<a href="#">AccessEnum.exe</a>
7/12/2007 6:26 AM	50379	<a href="#">AdExplorer.chm</a>
11/14/2012 11:22 AM	479832	<a href="#">ADExplorer.exe</a>
10/26/2015 4:06 PM	401616	<a href="#">ADInsight.chm</a>
10/26/2015 4:15 PM	2425496	<a href="#">ADInsight.exe</a>
11/1/2006 2:05 PM	150328	<a href="#">adrestore.exe</a>
12/29/2019 4:22 PM	<dir>	<a href="#">ARM64</a>
8/27/2016 11:54 AM	138920	<a href="#">Autologon.exe</a>
6/28/2019 1:07 PM	50512	<a href="#">autoruns.chm</a>
6/28/2019 1:07 PM	761656	<a href="#">Autoruns.exe</a>
6/28/2019 1:10 PM	757328	<a href="#">Autoruns64.dll</a>
6/28/2019 1:06 PM	875592	<a href="#">Autoruns64.exe</a>
6/28/2019 1:11 PM	680016	<a href="#">autorunsc.exe</a>
6/28/2019 1:10 PM	776480	<a href="#">autorunsc64.exe</a>
9/19/2019 10:17 PM	3353464	<a href="#">Bginfo.exe</a>
9/19/2019 10:15 PM	4601208	<a href="#">Bginfo64.exe</a>
11/1/2006 2:06 PM	154424	<a href="#">Cacheset.exe</a>
5/27/2016 1:58 AM	139944	<a href="#">Clockres.exe</a>
5/27/2016 1:55 AM	154792	<a href="#">Clockres64.exe</a>
5/27/2016 2:05 AM	253600	<a href="#">Contig.exe</a>
5/27/2016 2:02 AM	268960	<a href="#">Contig64.exe</a>
8/18/2014 12:29 PM	892088	<a href="#">Coreinfo.exe</a>
12/19/2019 10:11 AM	2181688	<a href="#">CPUSTRES.EXE</a>
12/19/2019 10:11 AM	2862648	<a href="#">CPUSTRES64.EXE</a>
9/27/2006 6:04 PM	10104	<a href="#">ctrl2cap.amd.sys</a>
11/1/2006 2:05 PM	150328	<a href="#">ctrl2cap.exe</a>
11/21/1999 6:20 PM	2864	<a href="#">ctrl2cap.nt4.sys</a>
11/21/1999 7:46 PM	2832	<a href="#">ctrl2cap.nt5.sys</a>
5/21/2019 9:34 AM	68539	<a href="#">Dbgview.chm</a>
5/21/2019 9:34 AM	914992	<a href="#">Dbgview.exe</a>
10/17/2012 6:28 PM	116824	<a href="#">Desktops.exe</a>
12/17/2013 12:46 PM	40717	<a href="#">Disk2vhd.chm</a>
1/20/2014 3:16 PM	7134400	<a href="#">disk2vhd.exe</a>

Now we can check the details about the tools by right click on the link as :-

The screenshot shows a web browser window with multiple tabs open at the top. The active tab is for 'live.sysinternals.com'. A context menu is displayed over a list of links, with the option 'Open link in new tab' highlighted. The list includes various Sysinternals tool names and their file paths, such as 'about\_th', 'accessch', 'AccessEn', 'AdExelor', 'ADInsigh', 'adrestor', 'ARM64', 'Autologo', 'autoruns', 'Autoruns.exe', 'Autoruns64.dll', 'Autoruns64.exe', 'autorunsc.exe', and 'autorunsc64.exe'. The date and time of each entry are listed to the left.

The details about the about this site is shown below :

The screenshot shows a web browser window displaying the contents of 'about\_this\_site.txt'. The page contains text explaining the purpose of the file share, how it allows running Sysinternals tools from the Internet, and instructions for users who are unfamiliar with Sysinternals. It also provides contact information for questions or comments. The Microsoft Windows Sysinternals Team is mentioned at the bottom.

What is this?  
This is a file share allowing access to all Sysinternals utilities. We have developed this to test an alternate distribution mechanism for our utilities.  
This will allow you to run these tools from any computer connected to the Internet without having to navigate to a webpage, download and extract the zip file.  
If you are unfamiliar with Microsoft Windows Sysinternals, it is highly recommended that you visit the website at <http://technet.microsoft.com/sysinternals> before using these tools.  
If you have any questions or comments on this file share, please email [syssite@microsoft.com](mailto:syssite@microsoft.com)  
Regards,  
The Microsoft Windows Sysinternals Team

**Here is the list of all the sysinternal tools with their description of their job:**

### [Process Explorer](#)

*v16.21 (May 16, 2017)*

Find out what files, registry keys and other objects processes have open, which DLLs they have loaded, and more. This uniquely powerful utility will even show you who owns each process.

### [Process Monitor](#)

*v3.50 (February 13, 2018)*

Monitor file system, Registry, process, thread and DLL activity in real-time.

### [PsExec](#)

*v2.2 (June 29, 2016)*

Execute processes on remote systems.

### [PsFile](#)

*v1.03 (June 29, 2016)*

See what files are opened remotely.

### [AccessChk](#)

*v6.20 (November 19, 2017)*

AccessChk is a command-line tool for viewing the effective permissions on files, registry keys, services, processes, kernel objects, and more.

### [AccessEnum](#)

*v1.32 (November 1, 2006)*

This simple yet powerful security tool shows you who has what access to directories, files and Registry keys on your systems. Use it to find holes in your permissions.

### [AdRestore](#)

*v1.1 (November 1, 2006)*

Undelete Server 2003 Active Directory objects.

### [Autologon](#)

*v3.10 (August 29, 2016)*

Bypass password screen during logon.

### [Autoruns](#)

*v13.95 (June 11, 2019)*

See what programs are configured to startup automatically when your system boots and you login. Autoruns also shows you the full list of Registry and file locations where applications can configure auto-start settings.

### [CacheSet](#)

*v1.0 (November 1, 2006)*

CacheSet is a program that allows you to control the Cache Manager's working set size using functions provided by NT. It's compatible with all versions of NT.

[ClockRes](#)

v2.1 (July 4, 2016)

View the resolution of the system clock, which is also the maximum timer resolution.

[Hex2dec](#)

v1.1 (July 4, 2016)

Convert hex numbers to decimal and vice versa.

[Junction](#)

v1.07 (July 4, 2016)

Create Win2K NTFS symbolic links.

[LMDDump](#)

v1.02 (November 1, 2006)

Dump the contents of the Logical Disk Manager's on-disk database, which describes the partitioning of Windows 2000 Dynamic disks.

[ListDLLs](#)

v3.2 (July 4, 2016)

List all the DLLs that are currently loaded, including where they are loaded and their version numbers.

[ProcDump](#)

v9.0 (May 16, 2017)

This command-line utility is aimed at capturing process dumps of otherwise difficult to isolate and reproduce CPU spikes.

- **Monitor Live Processes:** *Process Monitor* is an advanced monitoring tool for Windows that shows real-time file system, Registry and process/thread activity. It combines the features of two legacy Sysinternals utilities, *Filemon* and *Regmon*, and adds an extensive list of enhancements including rich and non-destructive filtering, comprehensive event properties such session IDs and user names, reliable process information, full thread stacks with integrated symbol support for each operation, simultaneous logging to a file, and much more. Its uniquely powerful features will make Process Monitor a core utility in your system troubleshooting and malware hunting toolkit.

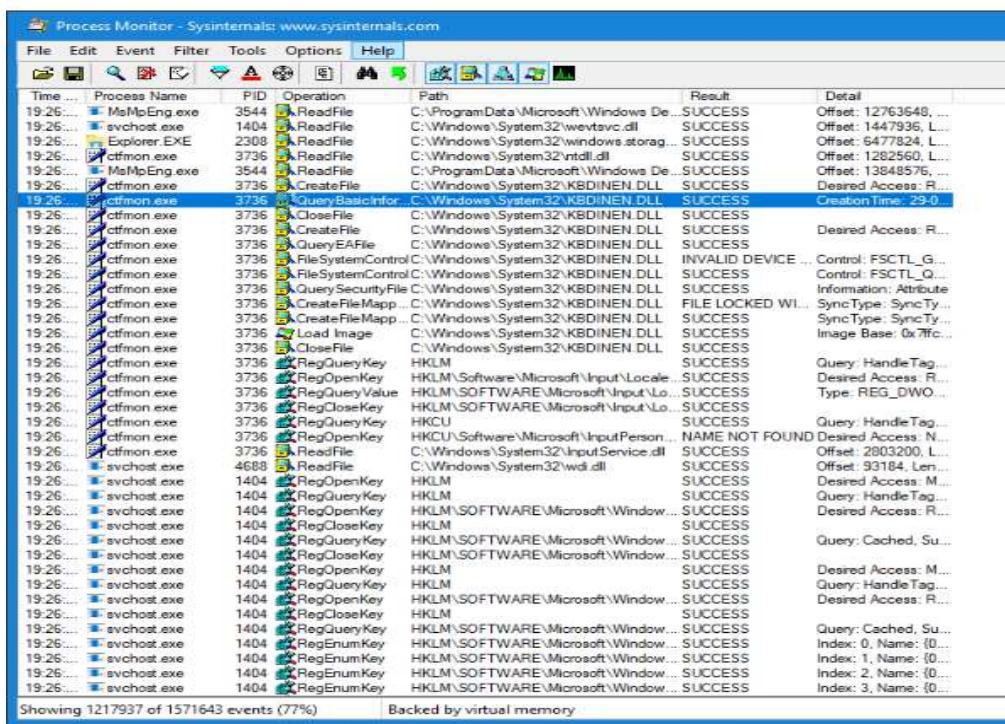
**Process Monitor includes powerful monitoring and filtering capabilities, including:**

- More data captured for operation input and output parameters
- Non-destructive filters allow you to set filters without losing data

- Capture of thread stacks for each operation make it possible in many cases to identify the root cause of an operation
- Reliable capture of process details, including image path, command line, user and session ID
- Configurable and moveable columns for any event property
- Filters can be set for any data field, including fields not configured as columns
- Advanced logging architecture scales to tens of millions of captured events and gigabytes of log data.

I installed the process monitor sysinternal tool, and then opened it on my system, the process monitor working is shown:

## The Process Monitor Interface:

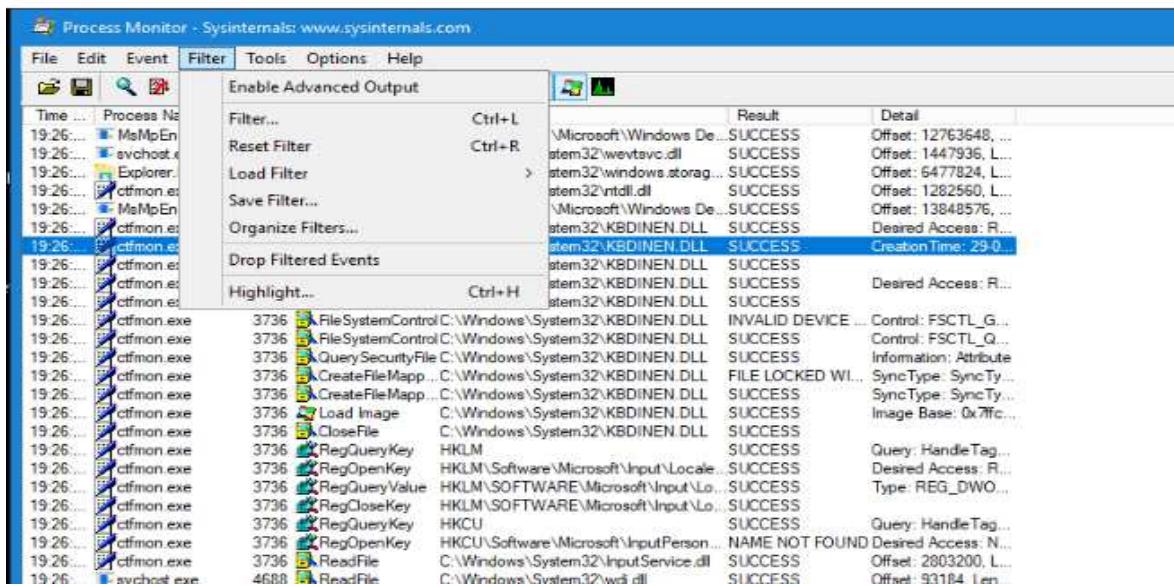


The screenshot shows the Process Monitor application window. The title bar reads "Process Monitor - Sysinternals: www.sysinternals.com". The menu bar includes File, Edit, Event, Filter, Tools, Options, and Help. Below the menu is a toolbar with various icons. The main area is a grid table with the following columns: Time, Process Name, PID, Operation, Path, Result, and Detail. The table contains numerous rows of data, mostly from the "ctfmon.exe" process, showing various file operations like ReadFile, CreateFile, CloseFile, and Registry operations like RegOpenKey and RegQueryKey. The "Result" column shows mostly "SUCCESS" with some "INVALID DEVICE" and "FILE LOCKED" entries. The "Detail" column provides more specific information about the operations. At the bottom of the table, a status bar indicates "Showing 1217937 of 15171643 events (77%) Backed by virtual memory".

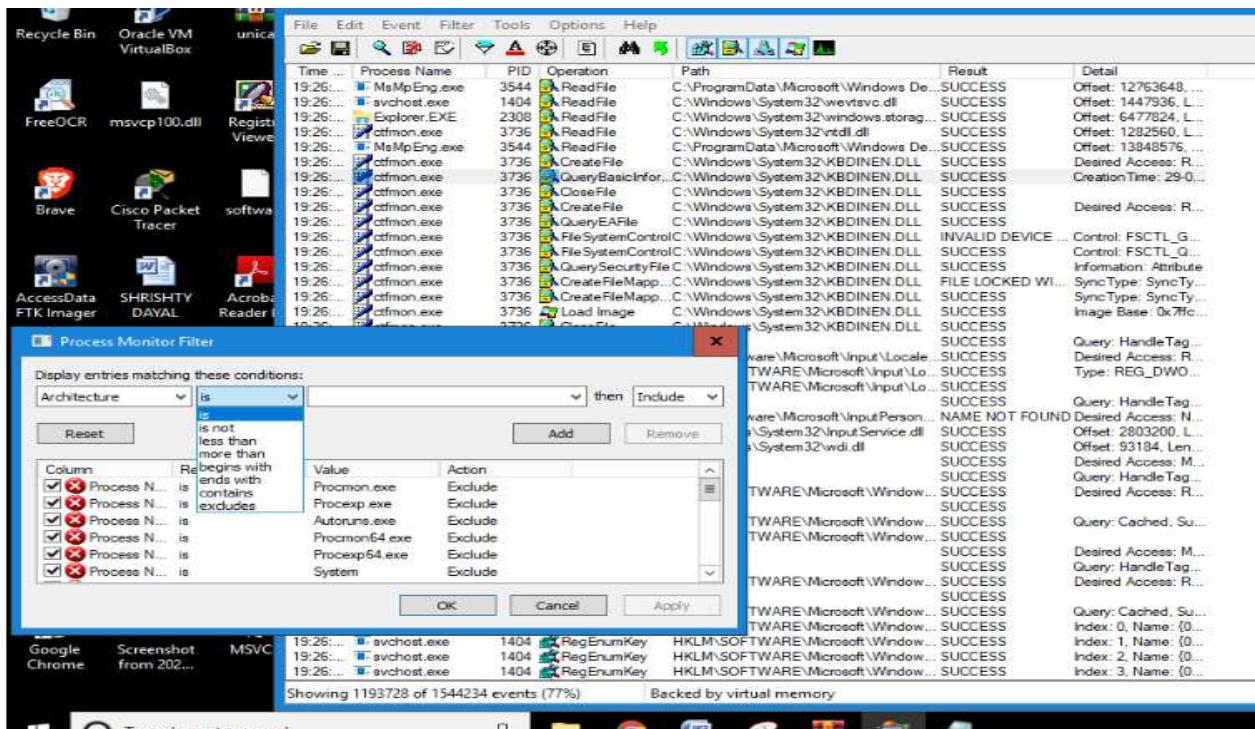
Time	Process Name	PID	Operation	Path	Result	Detail
19:26...	MsMpEng.exe	3544	ReadFile	C:\ProgramData\Microsoft\Windows\De...	SUCCESS	Offset: 12763648, ...
19:26...	svchost.exe	1404	ReadFile	C:\Windows\System32\wevtvc.dll	SUCCESS	Offset: 1447936, L...
19:26...	Explorer.EXE	2308	ReadFile	C:\Windows\System32\windowstorag...	SUCCESS	Offset: 6477824, L...
19:26...	ctfmon.exe	3736	ReadFile	C:\Windows\System32\vtdl.dll	SUCCESS	Offset: 1282560, L...
19:26...	MsMpEng.exe	3544	ReadFile	C:\ProgramData\Microsoft\Windows\De...	SUCCESS	Offset: 13648576, ...
19:26...	ctfmon.exe	3736	CreateFile	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	Desired Access: R...
19:26...	ctfmon.exe	3736	QueryBasicInfor...	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	Creation Time: 29-0...
19:26...	ctfmon.exe	3736	CloseFile	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	
19:26...	ctfmon.exe	3736	CreateFile	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	Desired Access: R...
19:26...	ctfmon.exe	3736	QueryEAFile	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	
19:26...	ctfmon.exe	3736	FileSystemControl	C:\Windows\System32\KBDDINEN.DLL	INVALID DEVICE ...	Control: FSCTL_G...
19:26...	ctfmon.exe	3736	FileSystemControl	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	Control: FSCTL_Q...
19:26...	ctfmon.exe	3736	QuerySecurityFile	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	Information: Attribute
19:26...	ctfmon.exe	3736	CreateFileMapp...	C:\Windows\System32\KBDDINEN.DLL	FILE LOCKED WI...	SyncType: SyncTy...
19:26...	ctfmon.exe	3736	CreateFileMapp...	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	SyncType: SyncTy...
19:26...	ctfmon.exe	3736	LoadImage	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	Image Base: 0x7fc...
19:26...	ctfmon.exe	3736	CloseFile	C:\Windows\System32\KBDDINEN.DLL	SUCCESS	
19:26...	ctfmon.exe	3736	RegQueryKey	HKLM	SUCCESS	Query: HandleTag...
19:26...	ctfmon.exe	3736	RegOpenKey	HKLM\Software\Microsoft\Input\Locale...	SUCCESS	Desired Access: R...
19:26...	ctfmon.exe	3736	RegQueryValue	HKLM\SOFTWARE\Microsoft\Input\Lo...	SUCCESS	Type: REG_DWO...
19:26...	ctfmon.exe	3736	RegCloseKey	HKLM\SOFTWARE\Microsoft\Input\Lo...	SUCCESS	
19:26...	ctfmon.exe	3736	RegQueryKey	HKCU	SUCCESS	Query: HandleTag...
19:26...	ctfmon.exe	3736	RegOpenKey	HKCU\Software\Microsoft\InputPerson...	NAME NOT FOUND	Desired Access: N...
19:26...	ctfmon.exe	3736	RegQueryValue	C:\Windows\System32\InputService.dll	SUCCESS	Offset: 2803200, L...
19:26...	svchost.exe	4688	ReadFile	C:\Windows\System32\wdl.dll	SUCCESS	Offset: 93184, Len...
19:26...	svchost.exe	1404	RegOpenKey	HKLM	SUCCESS	Desired Access: M...
19:26...	svchost.exe	1404	RegQueryKey	HKLM	SUCCESS	Query: HandleTag...
19:26...	svchost.exe	1404	RegOpenKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Desired Access: R...
19:26...	svchost.exe	1404	RegCloseKey	HKLM	SUCCESS	Query: Cached, Su...
19:26...	svchost.exe	1404	RegQueryKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Desired Access: M...
19:26...	svchost.exe	1404	RegOpenKey	HKLM	SUCCESS	Query: HandleTag...
19:26...	svchost.exe	1404	RegQueryKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Desired Access: R...
19:26...	svchost.exe	1404	RegCloseKey	HKLM	SUCCESS	Query: Cached, Su...
19:26...	svchost.exe	1404	RegQueryKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Desired Access: M...
19:26...	svchost.exe	1404	RegOpenKey	HKLM	SUCCESS	Query: HandleTag...
19:26...	svchost.exe	1404	RegQueryKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Desired Access: R...
19:26...	svchost.exe	1404	RegCloseKey	HKLM	SUCCESS	Query: Cached, Su...
19:26...	svchost.exe	1404	RegQueryKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Index: 0, Name: (0...
19:26...	svchost.exe	1404	RegEnumKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Index: 1, Name: (0...
19:26...	svchost.exe	1404	RegEnumKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Index: 2, Name: (0...
19:26...	svchost.exe	1404	RegEnumKey	HKLM\SOFTWARE\Microsoft\Window...	SUCCESS	Index: 3, Name: (0...

**Description about the process monitor interface:** When I first load up the Process Monitor interface, it was presented with an enormous number of rows of data, with more data flying in quickly. The first thing that I noticed was that to try filter those rows down to the much smaller subset of data.

- Similar to wireshark, in this tool also we can apply filters as well:

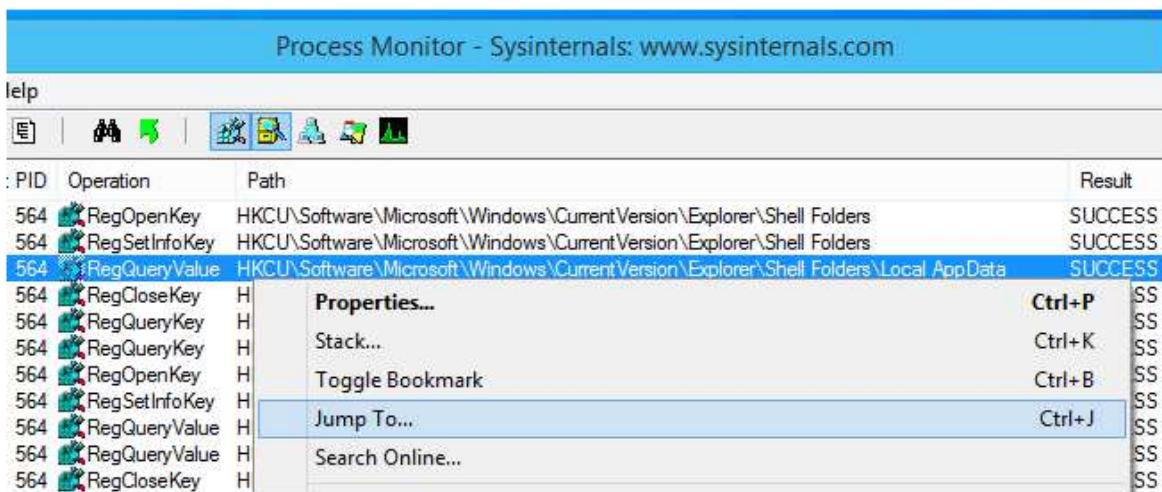


- We can also remove or edit filters by selecting them in the list and then modifying or removing them :



## ○ Jumping to an Event Data Path

All of this information is really great. We can right-click on the Path field for an item and use the Jump To option to quickly access that data to see what it contains and try to figure out why the application is requesting that data in the first place. We can also use the Search Online feature to quickly search for the name of the process, the registry path, or any other field, which can be really useful when you don't understand what something is used for



We can also add some additional columns to the default display:

This screenshot shows the 'Process Monitor Column Selection' dialog box. It allows users to choose which columns appear in the main Process Monitor window. The columns are grouped into three sections: Application Details, Event Details, and Process Management.

Select columns to appear in the Process Monitor window:	
<b>Application Details</b>	
<input checked="" type="checkbox"/> Process Name	<input type="checkbox"/> Description
<input checked="" type="checkbox"/> Image Path	<input type="checkbox"/> Version
<input checked="" type="checkbox"/> Command Line	<input type="checkbox"/> Architecture
<input checked="" type="checkbox"/> Company Name	
<b>Event Details</b>	
<input type="checkbox"/> Sequence Number	<input checked="" type="checkbox"/> Path
<input type="checkbox"/> Event Class	<input checked="" type="checkbox"/> Detail
<input checked="" type="checkbox"/> Operation	<input checked="" type="checkbox"/> Result
<input type="checkbox"/> Date &Time	<input type="checkbox"/> Relative Time
<input checked="" type="checkbox"/> Time of Day	<input type="checkbox"/> Duration
<input type="checkbox"/> Category	<input type="checkbox"/> Completion
<b>Process Management</b>	
<input type="checkbox"/> User Name	<input checked="" type="checkbox"/> Process ID
<input type="checkbox"/> Session ID	<input type="checkbox"/> Thread ID
<input type="checkbox"/> Authentication ID	<input checked="" type="checkbox"/> Parent PID
<input type="checkbox"/> Integrity	<input checked="" type="checkbox"/> Virtualized

Go to Options -> Select Columns.

One of the reasons for adding additional columns to the display is so we can very quickly filter by those events without being overwhelmed with data. Here are a few of the extra columns that we use:

### Command Line

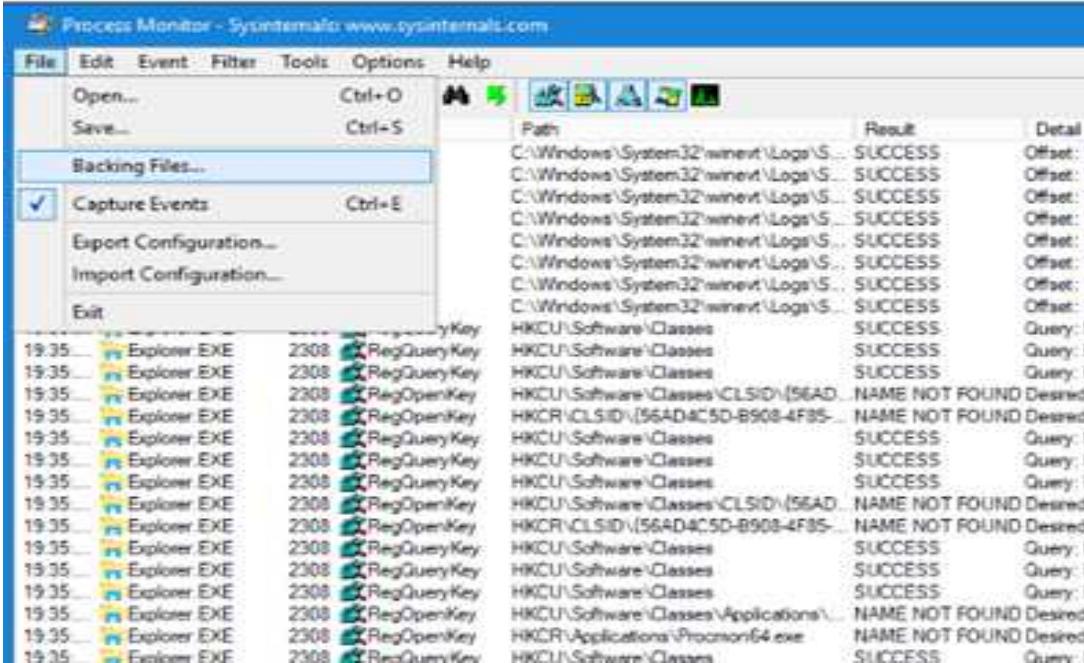
### Company Name

### Parent PID

#### o Capture RAM

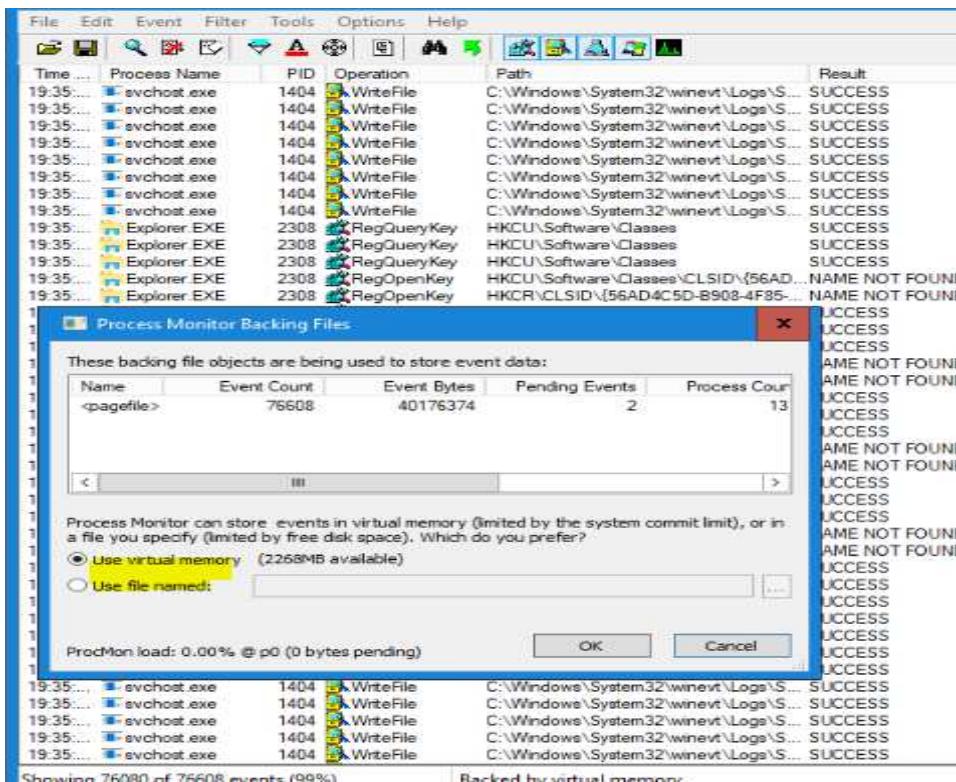
We can choose to store Process Monitor data in a file on disk instead of virtual memory

#### 1. Go to File -> Backing files:



Time	Process	Event ID	Action	Path	Result	Detail
19.35	Explorer.EXE	2308	RegQueryKey	C:\Windows\System32\winevent\Log\\$...	SUCCESS	Offset: 1
19.35	Explorer.EXE	2308	RegQueryKey	C:\Windows\System32\winevent\Log\\$...	SUCCESS	Offset: 1
19.35	Explorer.EXE	2308	RegOpenKey	C:\Windows\System32\winevent\Log\\$...	SUCCESS	Offset: 1
19.35	Explorer.EXE	2308	RegOpenKey	HKCR\CLSID\{56AD4C5D-B908-4F85...	NAME NOT FOUND	Desired
19.35	Explorer.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *
19.35	Explorer.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *
19.35	Explorer.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *
19.35	Explorer.EXE	2308	RegOpenKey	HKCU\Software\Classes\CLSID\{56AD...	NAME NOT FOUND	Desired
19.35	Explorer.EXE	2308	RegOpenKey	HKCR\CLSID\{56AD4C5D-B908-4F85...	NAME NOT FOUND	Desired
19.35	Explorer.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *
19.35	Explorer.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *
19.35	Explorer.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *
19.35	Explorer.EXE	2308	RegOpenKey	HKCU\Software\Classes\Applications\...	NAME NOT FOUND	Desired
19.35	Explorer.EXE	2308	RegOpenKey	HKCR\Applications\Procmon64.exe	NAME NOT FOUND	Desired
19.35	Filemon.EXE	2308	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: *

## 2. Specify the file where you want event data to be stored



### MONITOR TCP/UDP PACKETS

Open TCPView from the suite tools. TCPView is a Windows program that will show you detailed listings of all TCP and UDP endpoints on your system, including the local and remote addresses and state of TCP connections.

When you start TCPView it will enumerate all active TCP and UDP endpoints, resolving all IP addresses to their domain name versions. You can use a toolbar button or menu item to toggle the display of resolved names. You can also

TCPView - SystemInfo: www.sysinternals.com											
Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port	SINR	Sent Packets	Sent Bytes	Recv Packets	Rcvd I
System Print...	8	TCP	desktop-0f9e0	51781	172.217.167.42	Http	TIME_WAIT				
System Print...	8	TCP	desktop-0f9e0	51754	218.58.196.195	Http	TIME_WAIT				
System Print...	8	TCP	desktop-0f9e0	51787	573.194.152.126	Http	TIME_WAIT				
System Print...	8	TCP	(2482.0.90.298)	51781	24.06.27.0.1281...	Http	TIME_WAIT				
System Print...	8	TCP	(2482.0.90.298)	51782	24.06.27.0.1281...	Http	TIME_WAIT				
System Print...	8	TCP	(2482.0.90.298)	51777	24.06.27.0.1281...	Http	TIME_WAIT				
System Print...	8	TCP	(2482.0.90.298)	51778	24.06.27.0.1281...	Http	TIME_WAIT	1	28		
firefox.exe	4056	TCP	DESKTOP-ETL8U00	49709	localhost	49803	ESTABLISHED				74
firefox.exe	4056	TCP	DESKTOP-ETL8U00	49800	localhost	49879	ESTABLISHED				74
firefox.exe	4238	TCP	DESKTOP-ETL8U00	49893	localhost	49884	ESTABLISHED				74
firefox.exe	4036	TCP	DESKTOP-ETL8U00	49894	localhost	49883	ESTABLISHED				74
firefox.exe	9163	TCP	DESKTOP-ETL8U00	49709	localhost	49715	ESTABLISHED				74
firefox.exe	9163	TCP	DESKTOP-ETL8U00	49710	localhost	49759	ESTABLISHED				74
firefox.exe	3144	TCP	DESKTOP-ETL8U00	49848	localhost	49849	ESTABLISHED				74
firefox.exe	3144	TCP	DESKTOP-ETL8U00	49849	localhost	49848	ESTABLISHED				74
firefox.exe	1688	TCP	DESKTOP-ETL8U00	49862	localhost	49863	ESTABLISHED				74
firefox.exe	1968	TCP	DESKTOP-ETL8U00	49863	localhost	49882	ESTABLISHED				74
firefox.exe	7576	TCP	DESKTOP-ETL8U00	49884	localhost	49885	ESTABLISHED				74
firefox.exe	7576	TCP	DESKTOP-ETL8U00	49885	localhost	49884	ESTABLISHED				74
firefox.exe	6329	TCP	DESKTOP-ETL8U00	49966	localhost	49967	ESTABLISHED				74
firefox.exe	6329	TCP	DESKTOP-ETL8U00	49967	localhost	49958	ESTABLISHED				74
firefox.exe	4956	TCP	(69.84.99.9)	51723	103.52.0.94[45.45.45.45]	Http	ESTABLISHED				214
firefox.exe	7356	TCP	DESKTOP-ETL8U00	51452	localhost	51433	ESTABLISHED				214
firefox.exe	7356	TCP	DESKTOP-ETL8U00	51453	localhost	51432	ESTABLISHED				214
firefox.exe	4056	TCP	(69.84.99.9)	51733	218.58.266.174	Http	ESTABLISHED				214
firefox.exe	4056	TCP	(69.84.99.9)	51738	172.217.161.14	Http	ESTABLISHED				214
firefox.exe	4228	TCP	desktop-0f9e0	51737	172.217.161.101	Http	ESTABLISHED				214
firefox.exe	4228	TCP	desktop-0f9e0	51736	172.217.161.225	Http	ESTABLISHED				214
firefox.exe	4256	TCP	desktop-0f9e0	51741	391.98.351.54	Http	ESTABLISHED	1	18		214

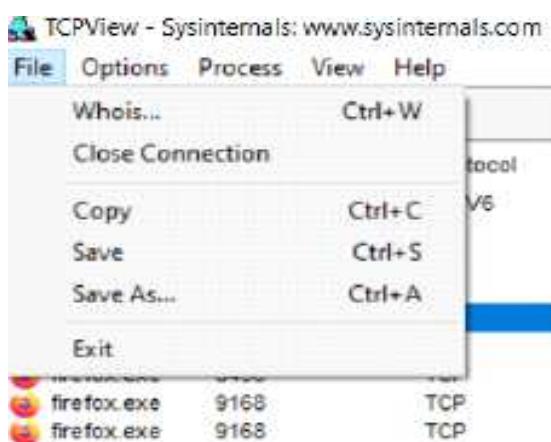
filter data in any column.

By default, TCPView updates every second, but you can use the Options>Refresh Rate menu item to change the rate. Endpoints that change state from one update to the next are highlighted in yellow; those that are deleted are shown in red, and new endpoints are shown in green.

Process	PID	Protocol	Local Address	Local Port	Remote Address	Remote Port
chrome.exe	1936	TCP	192.168.1.15	57525	93.184.221.200	443
chrome.exe	1936	TCP	192.168.1.15	57513	104.83.216.94	443
chrome.exe	1936	TCP	192.168.1.15	57516	104.83.201.19	443
chrome.exe	1936	TCP	192.168.1.15	57515	93.184.221.200	443
chrome.exe	1936	TCP	192.168.1.15	57522	134.170.180.146	443
svchost.exe	1616	UDP	127.0.0.1	1900	*	*
chrome.exe	1936	TCP	192.168.1.15	57514	104.83.216.94	443
chrome.exe	1936	TCP	192.168.1.15	57512	104.83.216.94	443
System	4	UDP	192.168.1.15	138	*	*
chrome.exe	1936	TCP	192.168.1.15	57511	104.83.216.94	443
System	4	UDP	192.168.1.15	137	*	*
chrome.exe	1936	TCP	192.168.1.15	57508	216.58.212.68	443



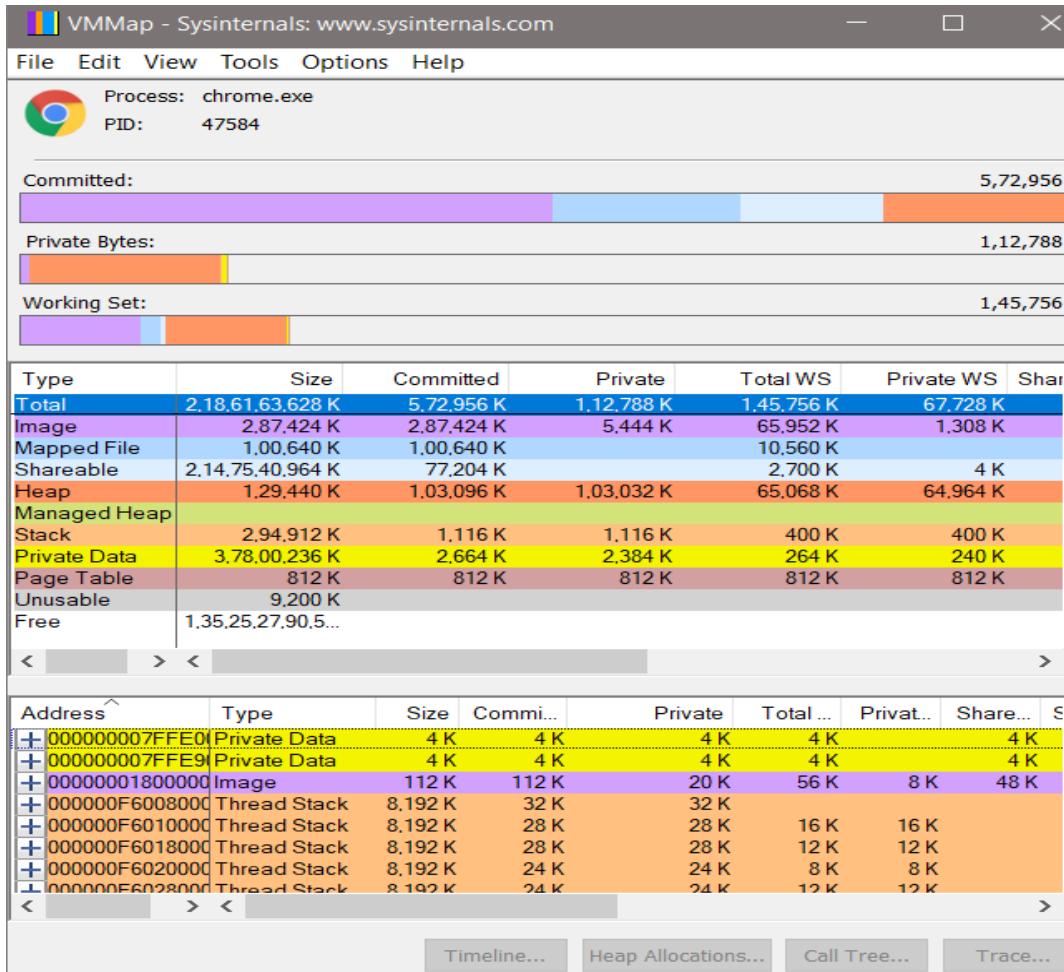
We can close established TCP/IP connections (those labeled with a state of ESTABLISHED) by selecting File>Close Connections, or by right-clicking on a connection and choosing Close Connections from the resulting context menu.



○

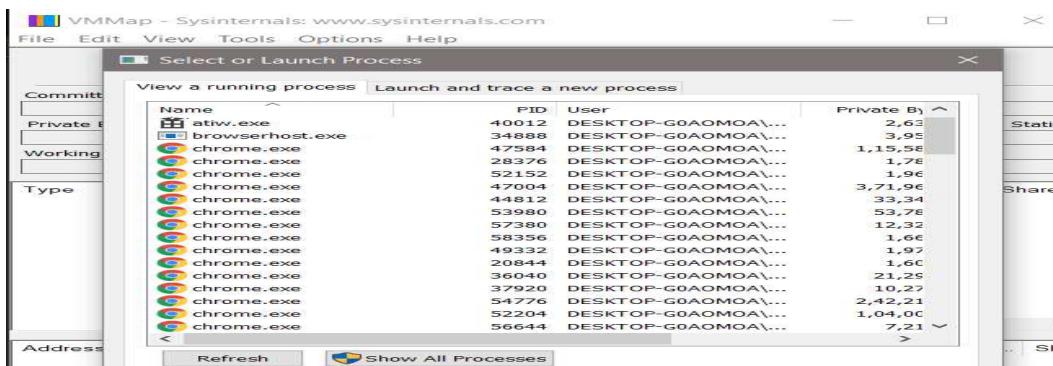
### ○ Monitor Virtual Memory

To Monitor Virtual Memory using a Sys-Internals tool, we have to use VM-Map. We can find this in the SysInternals Suite package. Once we double click on the application we will be greeted by the major live processes of the system as we can see in the picture above and to take a look about their memory, we need to



select a process and click 'Okay'.

Click the Start menu on your VPS' desktop, then using the 'Search programs and files' box simply type in resmon. Hit enter to start Resource Monitor. When the

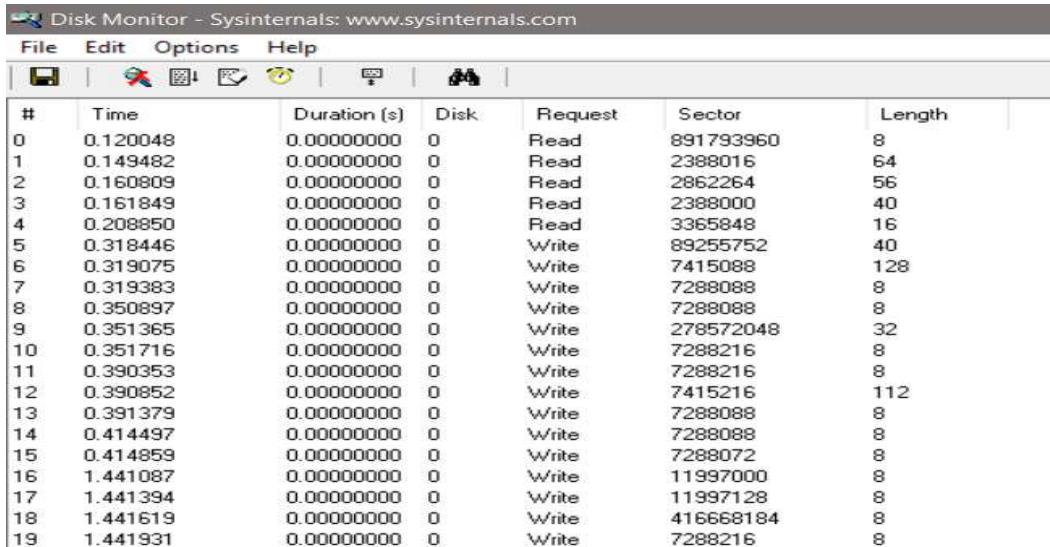


Resource **Monitor** window opens, click the **Memory** tab. In the upper section here we will see a list of running processes and how much **memory** they are using.

- **Monitor Hard Disk**

To monitor the activity inside the Hard Disk we need to use an application in SysInternals Suite package known as Disk Monitor. We can find it as Diskmon in the package. We need to open this application as an Administrator because a guest cannot access such sensitive information.

DiskMon is an application that logs and displays all hard disk activity on a Windows system. We can also minimize DiskMon to your system tray where it acts as a disk light, presenting a green icon when there is disk-read activity and a red icon when there is disk-write activity.

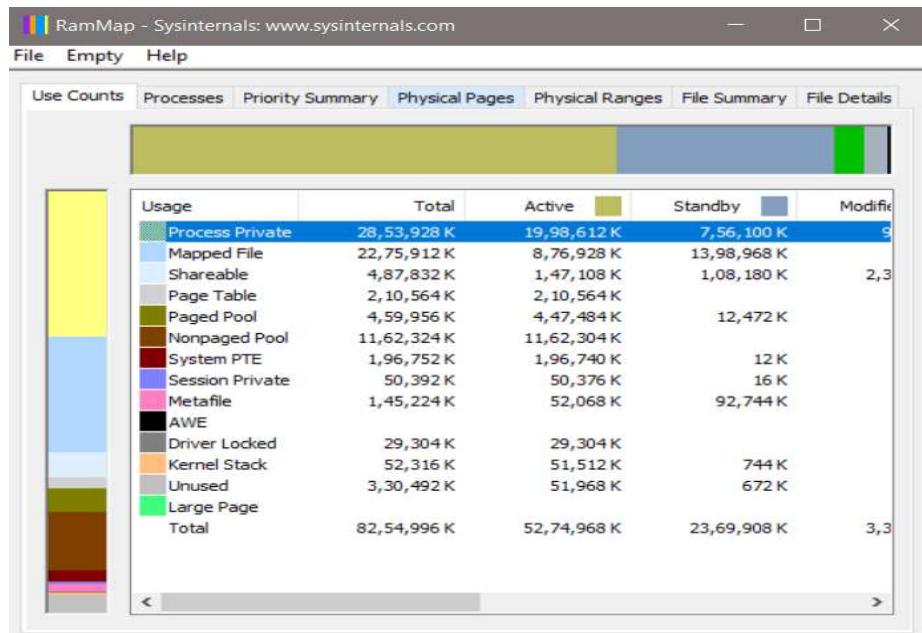


#	Time	Duration (s)	Disk.	Request	Sector	Length
0	0.120048	0.00000000	0	Read	891793960	8
1	0.149482	0.00000000	0	Read	2388016	64
2	0.160809	0.00000000	0	Read	2862264	56
3	0.161849	0.00000000	0	Read	23880000	40
4	0.208850	0.00000000	0	Read	3365848	16
5	0.318446	0.00000000	0	Write	89255752	40
6	0.319075	0.00000000	0	Write	7415088	128
7	0.319383	0.00000000	0	Write	7288088	8
8	0.350897	0.00000000	0	Write	7288088	8
9	0.351365	0.00000000	0	Write	278572048	32
10	0.351716	0.00000000	0	Write	7288216	8
11	0.390353	0.00000000	0	Write	7288216	8
12	0.390852	0.00000000	0	Write	7415216	112
13	0.391379	0.00000000	0	Write	7288088	8
14	0.414497	0.00000000	0	Write	7288088	8
15	0.414859	0.00000000	0	Write	7288072	8
16	1.441087	0.00000000	0	Write	11997000	8
17	1.441394	0.00000000	0	Write	11997128	8
18	1.441619	0.00000000	0	Write	416668184	8
19	1.441931	0.00000000	0	Write	7288216	8

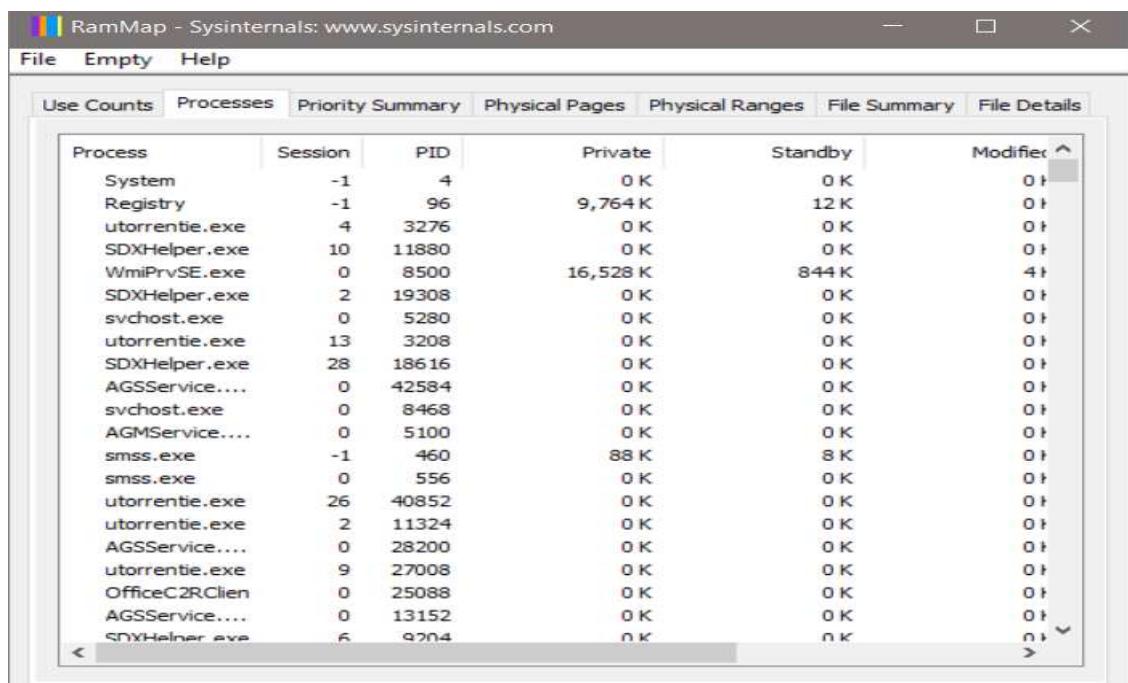
Therefore, as we can see in the image above these are some of the process that are running in the Hard Disk.

- **Monitor Cache Memory**

We know that cache memory is stored in RAM. Therefore, we need to use RAM-Map to monitor Cache Memory. We can find this in the Sys-Internals Suite package.



The above image shows the different portions of the RAM. To access the Cache processes we need to select the 'Processes' tab.



The picture above shows the processes stored in the RAM.

## **EXPERIMENT – 7.A**

### **Recovering deleted files on Windows**

In this experiment, what we do is we recover deleted files from pendrive.

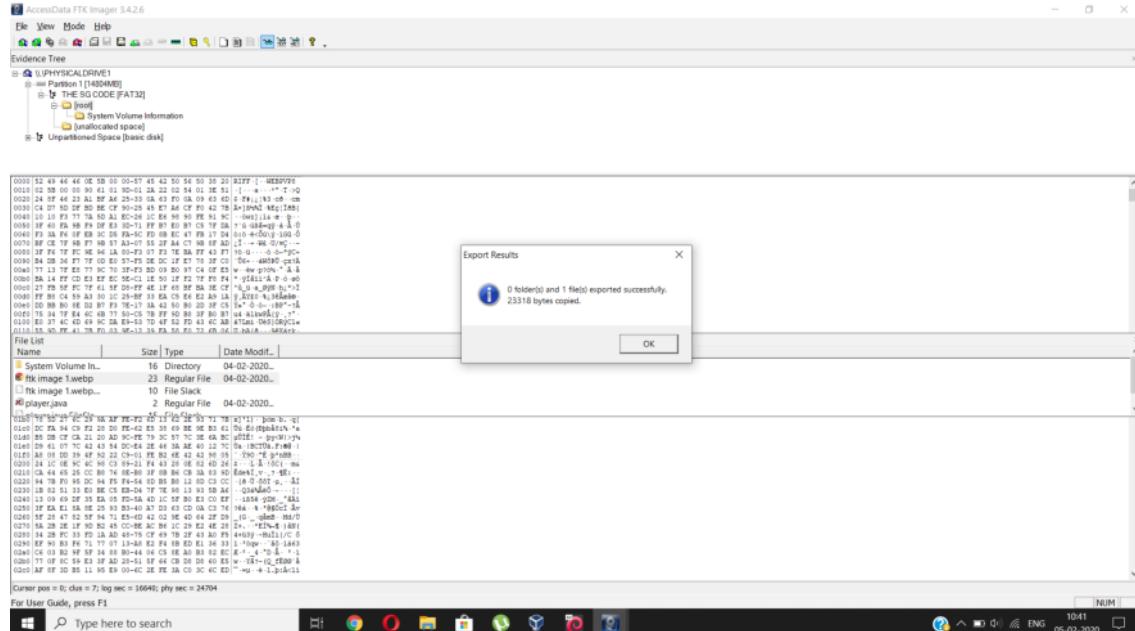
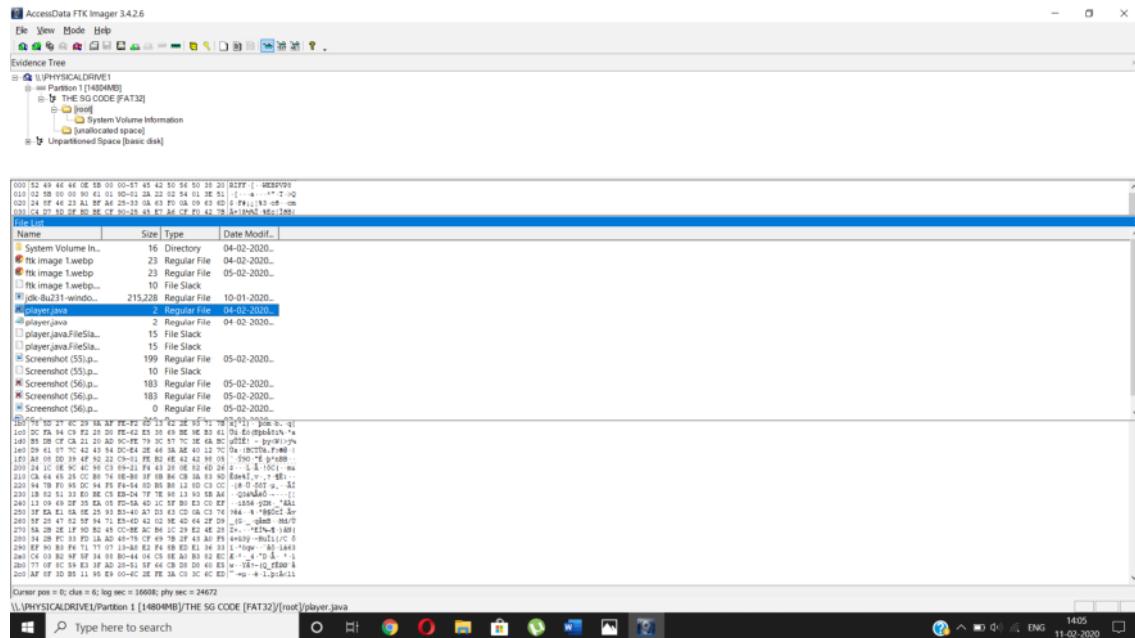
We achieve this by using FTK imager tool by deleting some files from the pen drive and then finding the deleted items on it.

We do this for both linux and Windows.

In this we first add evidence item on the tool ADD EVIDENCE ITEM.

1. Launch FTK Imager
2. Select File > Add Evidence Item
3. Select "physical drive" which is pen drive here and proceed to add the file
4. Under the "Evidence Tree", right click your file and select Verify Drive.
5. Then here we right click and click on root option.
6. Then we find for the folder from we have deleted some files.
7. Here in front of the files where we have a cross mark it means it has been deleted from the pen drive very previously and we have to recover it.
8. For recovering it, we can right click on that picture of text file and can select the option of exporting it.
9. After this we can choose destination as of where to recover those deleted files or pic.

# Snapshot:



## **RECOVERING DELETED FILES ON LINUX SYSTEM**

In this, we use tools such as TESTDISK which we downloaded from browser.

Test Disk is a free and open-source data recovery utility. It is primarily designed to help recover lost data storage partitions and/or make non-booting disks bootable again when these symptoms are caused by faulty software, certain types of viruses or human error.

Then we perform following commands to recover files on linux.

```
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

TestDisk is free data recovery software designed to help recover lost
partitions and/or make non-booting disks bootable again when these symptoms
are caused by faulty software, certain types of viruses or human error.
It can also be used to repair some filesystem errors.

Information gathered during TestDisk use can be recorded for later
review. If you choose to create the text file, testdisk.log , it
will contain TestDisk options, technical information and various
outputs; including any folder/file names TestDisk was used to find and
list onscreen.

Use arrow keys to select, then press Enter key:
>[ Create ] Create a new log file
  [ Append ] Append information to log file
  [ No Log ] Don't record anything
```

Applications ▾ Places ▾ Terminal ▾

```
File Edit View Search Terminal Help
TestDisk 7.1, Data Recovery Utility, July 2019
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

TestDisk is free software, and
comes with ABSOLUTELY NO WARRANTY.

Select a media (use Arrow keys, then press Enter):
>Disk /dev/sda - 85 GB / 80 GiB - VBOX HARDDISK
```

```
Please select where to store the file image.dd (2145 MB), an image of the
partition
Keys: Arrow keys to select another directory
      C when the destination is correct
      Q to quit
Directory /root
>drwxr-xr-x    0    0    4096 11-Feb-2020 12:16 .
drwxr-xr-x    0    0    36864 17-May-2019 09:25 ..
drwxr-xr-x    0    0    4096  5-Feb-2020 00:13 Desktop
drwxr-xr-x    0    0    4096 17-May-2019 09:07 Documents
drwxr-xr-x    0    0    4096  4-Feb-2020 23:54 Downloads
drwxr-xr-x    0    0    4096 17-May-2019 09:07 Music
drwxr-xr-x    0    0    4096 11-Feb-2020 12:51 Pictures
drwxr-xr-x    0    0    4096 17-May-2019 09:07 Public
drwxr-xr-x    0    0    4096 17-May-2019 09:07 Templates
drwxr-xr-x    0    0    4096 17-May-2019 09:07 Videos
-rw-r--r--    0    0     376 11-Feb-2020 12:40 testdisk.log
```

```
Christophe GRENIER <grenier@cgsecurity.org>
https://www.cgsecurity.org

Disk /dev/sda - 85 GB / 80 GiB - CHS 10443 255 63

      Partition            Start            End    Size in sectors
> 1 * Linux                  0  32 33 10182  63 57  163575808
  2 E extended                10182  96 25 10443  52 41   4190210
  5 L Linux Swap               10182  96 27 10443  52 41   4190208
```

```
TestDisk 7.1, Data Recovery Utility, April 2018
Christophe GRENIER <grenier@cgsecurity.org>
http://www.cgsecurity.org
 1 P FAT32 LBA          3  60  1 14807  63 32  30318720 [THE SG CODE]
Directory /

>drwxr-xr-X  0  0        0 4-Feb-2020 19:12 System Volume Information
-rw xr-xr-X  0  0     1340 4-Feb-2020 10:23 player.java
-rw xr-xr-X  0  0    23318 4-Feb-2020 18:59 ftk image 1.webp
-rw xr-xr-X  0  0    23318 5-Feb-2020 00:29 ftk image 1.webp
-rw xr-xr-X  0  0     1340 4-Feb-2020 15:53 player.java
-rw xr-xr-X  0  0   203368 5-Feb-2020 10:41 Screenshot (55).png
-rw xr-xr-X  0  0   187007 5-Feb-2020 10:41 Screenshot (56).png
-rw xr-xr-X  0  0   187007 5-Feb-2020 02:41 Screenshot (56).png
-rw xr-xr-X  0  0        0 5-Feb-2020 02:41 Screenshot (56).png
-rw xr-xr-X  0  0   224123 7-Feb-2020 00:51 SS.docx
-rw xr-xr-X  0  0  220392992 10-Jan-2020 16:15 jdk-8u231-windows-x64.exe
-rw xr-xr-X  0  0   39731395 11-Feb-2020 15:54 Mantooth32 (1) (1).E01
-rw xr-xr-X  0  0   88178223 11-Feb-2020 15:51 CLAMPET12(2).E01
-rw xr-xr-X  0  0   118460 11-Feb-2020 16:04 Clampet and Mantooth Case Stu

Next
Use Right to change directory, h to hide deleted files
  q to quit, : to select the current file, a to select all files
  C to copy the selected files, c to copy the current file
```

## EXPERIMENT – 7.B

### GEORGE AND MARTHA CASE

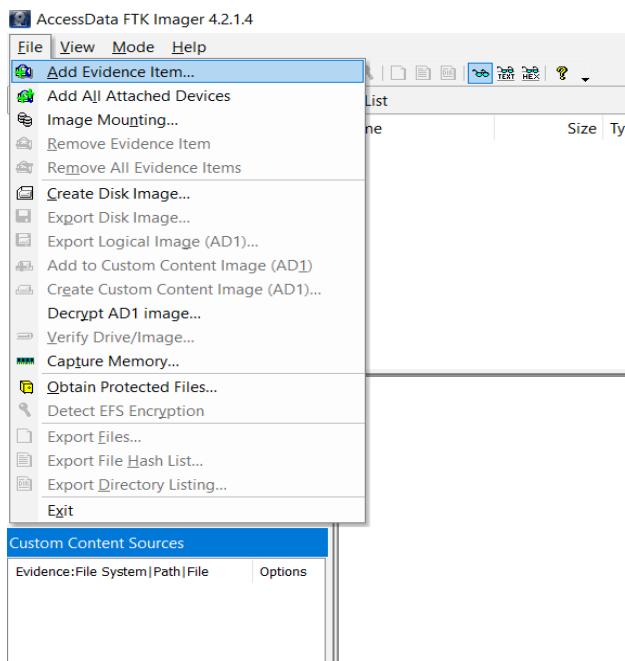
We have given a case of 2 people working in a company George and Martha.

George Montgomery has worked at a firm for several years and is now missing. Another employee, Martha, is also missing. No one knows where they are or has seen them in over a week, so Steve (George's supervisor) asks the IT Department to confiscate George's harddrive and all storage media in his work area.

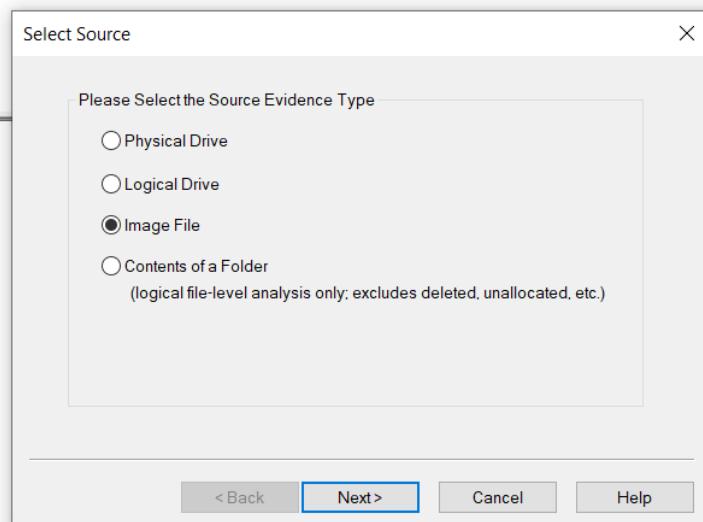
Investigate their scenario of what policies they broke and how they performed a theft of large amount of money.

#### **Procedure:**

- Open the evidence file on FTK Imager using add evidence.



- Select image file.



- File's content will be loaded in the evidence tree. Click on the image to reveal folders.

The screenshot shows the FTK Imager application window. The top menu bar includes File, View, Mode, Help, and a toolbar with various icons for file operations. The left pane, titled "Evidence Tree", displays a hierarchical tree structure of a FAT12 image named "ftk-demo1-image.1". The root directory "CHAPTER 5" contains sub-directories "[root]" and "[unallocated space]". The right pane, titled "File List", shows a table of files found in the root directory:

Name	Size	Type	Date Modif...
account	1	Directory	15-02-2003...
personal	1	Directory	15-02-2003...
work	1	Directory	15-02-2003...
flowers.jpg	432	Regular File	15-02-2003...
mt_rainier2.jpg	29	Regular File	15-02-2003...
y.exe	22	Regular File	15-02-2003...

The bottom left pane, "Custom Content Sources", contains a dropdown menu for "Evidence: File System | Path | File" and an "Options" button. The bottom right pane shows a detailed hex dump of the file "y.exe" starting at offset 0000, displaying ASCII characters and their corresponding hex values.

- Check the personal>messages folder

Evidence Tree		File List			
		Name	Size	Type	Date Modif...
flk-demo1-image.1					
CHAPTER 5 [FAT12]					
[root]					
account					
data					
personal					
Messages					
work					
[unallocated space]					
		g-021218.msg	1	Regular File	15-02-2003...
		g-021229.msg	1	Regular File	15-02-2003...
		m-021220.msg	1	Regular File	15-02-2003...
		m-021230.msg	1	Regular File	15-02-2003...
		msg5.txt	1	Regular File	15-02-2003...
		msg7.txt	1	Regular File	15-02-2003...

- Upon reading the messages it was found that George was planning something illegal to get a lot of money.

From: Jones, George [mailto:georgej@widgets\_intl.com]  
 Sent: 18 December 2001 18:37  
 To: James, Martha [marthaj@widgets\_intl.com]  
 Subject: A plan

Martha,

I have a plan to pay for our vaction next Spring. I'll tell you about it later.

George

From: Jones, George [mailto:georgej@widgets\_intl.com]  
 Sent: 29 December 2001 10:52  
 To: James, Martha [marthaj@widgets\_intl.com]  
 Subject: Re: A plan

I can't talk about it now, it will have to wait until we can talk in private. Maybe Year's Eve.

George

-----Original Message-----  
 From: Jones, George [mailto:georgej@widgets\_intl.com]  
 Sent: 26 December 2001 08:02  
 To: James, Martha [marthaj@widgets\_intl.com]  
 Subject: A plan

Martha,

I have a plan to pay for our vaction next Spring. I'll tell you about it later.

George

```

George,
What are you talking about, what's the big deal?
Martha
-----Original Message-----
From: Jones, George [mailto:georgej@widgets_intl.com]
Sent: 29 December 2001 10:52
To: James, Martha [marthaj@widgets_intl.com]
Subject: Re: A plan

I can't talk about it now, it will have to wait until we can talk in private. Maybe Year's Eve.

George

```

- Some deleted texts were revealed too.

ear Mart ,

I've got a plan to get more cash. It involves rerouting invoices here at work to the field offices. It will take six months to get back here and we can be in Brazil enjoying the fruits of our labo

Mrg  
Dr.

deposits. I encrypted this file so as to prevent anyone from reading them. If I have to flee when the auditor finds the missing money, you can meet me in Zurich % on the 19 of January. yours devotedly, Ge + orge ,

- Check the work folder which was deleted.

Evidence Tree		File List			
		Name	Size	Type	Date Modif...
↳	fd-disk1-image 1	AF6.JPG	1	Regular File	15-02-2003...
↳	CHAPTER 5[FAT12]	LY.EXE	2	Regular File	15-02-2003...
↳	[root]	cat.jpg	274	Regular File	15-02-2003...
↳	account	daffodil.jpg	36	Regular File	15-02-2003...
↳	data	flowers.jpg	432	Regular File	15-02-2003...
↳	personal	flower2.gif	79	Regular File	15-02-2003...
↳	Messages	msg4.txt	1	Regular File	15-02-2003...
↳	work				
↳	[unallocated space]				

Dear Martha, Here it is June I have almost deposited a million dollars. If I can get away with this I'll be able to have two million in the Isle of Man bank /

Dear Martha, Here it is June I have almost deposited a million dollars. If I can get away with this I'll be able to have two million in the Isle of Man bank account by xmas time. more later, devotedly yours, George

Name	Size	Type	Date Modified
X !AF6.JPG	1	Regular File	15-02-2003...
X !_Y.EXE	2	Regular File	15-02-2003...
X cat.jpg	274	Regular File	15-02-2003...
X daffodil.jpg	36	Regular File	15-02-2003...
X flowers.jpg	432	Regular File	15-02-2003...
X flowers2.gif	79	Regular File	15-02-2003...
X msg4.txt	1	Regular File	15-02-2003...

- Check the account>data folder. A web link reveals the password of a file to be “couch”.

**Evidence Tree**

- flk-demo1-image.1
  - CHAPTER 5 [FAT12]
    - [root]
    - account
    - data
    - personal
    - work
    - [unallocated space]

**File List**

Name	Size	Type	Date Modif...
mt_bank_secrecy...	3	Regular File	15-02-2003...
X.ZIP	7	Regular File	15-02-2003...

Mr. Jones,

The password for your account is: couch

Please let us know if you need anything else.

Regards,

Sigor Krautfletz

Isle of Man Saving & Loan

**Custom Content Sources**

Evidence:File System|Path|File      Options

- A zip file revealed some excel sheet which was opened using the password couch. It was a transaction sheet which showed illegal transactions made by George.

**Evidence Tree**

- flk-demo1-image.1
  - CHAPTER 5 [FAT12]
    - [root]
    - account
    - data
    - personal
    - Messages
    - work
    - [unallocated space]

**File List**

Name	Size	Type	Date Modif...
mt_bank_secrecy...	3	Regular File	15-02-2003...
X.ZIP	7	Regular File	15-02-2003...

**Custom Content Sources**

Evidence:File System|Path|File      Options

**File List**

Name	Type	Compress...	Passw...	Size	Ratio	Date modif...
SWISS.CSV						
SWISS.TXT						
SWISS.XLS						

**Password needed**

File 'SWISS.XLS' is password protected.  
Please enter the password in the box below.

OK      Skip File      Cancel

Password:  \*\*\*\*\*

Janvier 29, 2002				
A	B	C	D	E
4				
5 Account Number:	9882111			
6 Les montants ont énumérés en des dollars des Etats-Unis				
7 Quantité de dépôt	Argent Total Courant	Intérêt gagné à 6,533 pour cent		Date de dépôt
8				
9 \$1,524.00	\$1,623.56	\$99.56		Janvier 29, 2002
10 \$15,888.00	\$18,655.59	\$1,037.96		Février 14, 2002
11 \$10,056.00	\$30,587.32	\$656.96		Mars 12, 2002
12 \$1,547.00	\$34,233.66	\$101.07		Avril 13, 2002
13 \$22,014.00	\$59,922.32	\$1,438.17		Mai 13, 2002
14 \$2,554.00	\$66,557.89	\$166.85		Juin 10, 2002
15 \$24,450.00	\$96,953.44	\$1,597.32		Juillet 6, 2002
16 \$2,412.00	\$1,05,856.98	\$157.58		
17 \$24,186.00	\$1,38,538.69	\$1,580.07		Septembre 12, 2002
18 \$2,541.00	\$1,50,296.43	\$166.00		Octobre 13, 2002
19 \$5,321.00	\$1,65,783.92	\$347.62		Novembre 12, 2002
20 \$24,632.00	\$2,02,855.79	\$1,609.21		Décembre 2, 2002
21 \$2,12,588.00	\$4,42,584.73	\$13,888.37		Janvier 24, 2003
22 \$24,553.00	\$4,97,655.84	\$1,604.05		Février 12, 2003
23 \$9,823.00	\$5,40,632.43	\$641.74		Mars 22, 2003
24 \$7,892.00	\$5,84,359.53	\$515.58		Avril 4, 2003
25 \$2,353.00	\$6,25,042.46	\$153.72		Mai 22, 2003
26 \$22,145.00	\$6,89,468.22	\$1,446.73		Juin 15, 2003
27 \$58,200.00	\$7,96,513.38	\$3,802.21		Juillet 3, 2003
28 \$59,311.00	\$9,11,735.39	\$3,874.79		Août 23, 2003
29 \$6,548.00	\$9,78,274.84	\$427.78		Septembre 24, 2003
30 \$54,156.00	\$10,99,879.55	\$3,538.01		Octobre 11, 2003
31 \$2,144.00	\$11,74,018.75	\$140.07		Novembre 2, 2003
32 \$47,872.00	\$13,01,716.87	\$3,127.48		Décembre 3, 2003
33 \$36,548.00	\$14,25,693.71	\$2,387.68		Janvier 20, 2004
34 \$2,31,455.00	\$17,65,410.24	\$15,120.96		Février 13, 2004
35 \$2,486.00	\$18,83,392.90	\$162.41		Mars 2, 2004
36 \$24,863.00	\$20,32,922.26	\$1,624.30		Avril 14, 2004
37 \$98,765.00	\$22,70,950.39	\$6,452.32		Mai 3, 2004
38 \$17,893.00	\$24,38,373.53	\$1,168.95		Juin 12, 2004
39 \$34,795.00	\$26,34,740.63	\$2,273.16		Juillet 4, 2004
40 \$54,892.00	\$28,65,346.33	\$3,586.09		Août 21, 2004
41 \$45,789.00	\$31,01,319.80	\$2,991.40		Septembre 22, 2004
42 \$34,447.00	\$33,40,626.44	\$2,250.42		Octobre 10, 2004
43 \$29,833.00	\$35,90,651.56	\$1,948.99		Novembre 3, 2004
44 \$68,945.00	\$38,98,678.00	\$4,504.18		Décembre 4, 2004

- The unallocated folder revealed some further deleted files which upon recovery showed following:

Dear Mart,

I've got a plan to get more cash. It involves rerouting invoices here at work to the field offices. It will take six months to get back here and we can be in Brazil enjoying the fruits of our labo

I'll tell you more about it when I get it started.

Yours always, George

**Conclusion:** George did some rerouting with his company's money to transfer it to his own account while sharing details of the same with Martha through emails and messages. Files for the same were found in the personal folder of George. An excel sheet of transactions of his account further proved that such transactions were made.

This proved that George and Martha were involved in the forgery.

## **EXPERIMENT 8.a**

### **EMAIL FORENSICS**

Email forensics refers to analyzing the source and content of emails as evidence. Investigation of email related crimes and incident involves various approaches.

#### **Analyzing Email Header**

- View the Message Header in Google Mail (GMail) Webmail:**

Login to your account on the webpage and open the message (click on it). Click on the "down-arrow" on the top-right of the message and select "Show Original". Now you will see the complete message source.

- View the Message Header in Yahoo! Mail Webmail:**

Login to your account on the webpage and open the message (click on it).  
Click on "Actions" and select "View Full Header".

- View the Message Header in Hotmail Webmail:**

Login to your account on the webpage and go to the message list.  
Right-click on the message and select "View Message Source".

- View the Message Header in MS Outlook:**

Open the message in MS Outlook. Now go to "View" and select "Message Options" - or "File" -> "Info" -> "Properties".

Look at "Internet Headers"

- **View the Message Header in Thunderbird:**

Open the message, then click on "View" and select "Message Source".

- **View the Message Header in MS Windows Mail (and MS Outlook Express):**

Select the message in the list, right-click on it and select "Properties" and go to "Details".

- **Standard Email Header Fields**

- Return Path: The email address which should be used for bounces.  
The mail server will send a message to the specified email address if the message cannot be delivered.
- Delivery-date: The date the message was delivered.
- Date: The date the message was sent.
- Message-ID: The ID of the message.
- X-Mailer: The mail client (mail program) used to send the message
- From: The message sender in the format: "Friendly Name" <email@address.tld>
- To: The message recipient in the format: "Friendly Name" <email@address.tld>
- Subject: The message subject

### ***Recovering Emails:***

1. Open Outlook.
2. Select the "Deleted Items" folder.
3. Go to the "Tools >> Recover Deleted Items from server"
4. Select the email(s) that you would like to recover.
5. Click the "Recover Selected Items" button (the icon is an email message with an arrow).
6. The email will go back to the "Deleted Items" folder it was in. (You may need to select another folder and then reselect this folder for it to appear.)

Similar methods can be used for all mailboxes.

### ***Email Protocols***

Email protocol is a standard method for exchanging information between email clients like Thunderbird, Apple Mail, or Mailbird and email provider's servers like Gmail, Outlook, Yahoo, and vice versa.

Following are some common Email Protocols:

**SMTP:** As the name suggests, Simple Message Transfer Protocol is responsible for email transfers between email clients (Windows Mail, Thunderbird, etc.) and email providers' servers (Gmail, Outlook, Yahoo). Companies use their SMTP server for email marketing and for sending automated transactional emails.

**POP3:** Email clients use Post Office Protocol 3 for retrieving messages from email servers. Email clients that use POP3 store messages on the user's computer, deleting them from the email server. People using email clients with POP3 also have the option of keeping their emails on the server after download.

**IMAP:** Internet Message Access Protocol is similar to POP3, but unlike it, IMAP allows multiple users to send emails at a time. This is a helpful feature for business owners, who assign communication with customers to different team members — especially when they need to have access to one email address at one time.

# EXPERIMENT 8.b

## Case Study: Wes Mantooh

### 1. Under the email category find the following details:

- a. Appointments: 11
- b. Contacts: 8
- c. Mails: 22
- d. Notes: 4
- e. Tasks: 5

The screenshot shows a digital forensics tool interface with two main panes: 'Evidence Tree' and 'File List'.

**Evidence Tree:** Shows a tree view of evidence items under 'Deleted Items', 'Drafts', and 'Junk Email'. Some file names are visible, such as '0C130270-0000000D.eml', '10AB12E1-00000012.eml', etc.

**File List:** A table showing a list of files with columns for Name, Size, Type, and Date Modified.

Name	Size	Type	Date Modified
\$130	4	NTFS Index All...	04-08-2007 16:08:36
0C130270-0000000D.eml	4	Regular File	24-07-2007 21:02:29
10AB12E1-00000012.eml	3	Regular File	25-07-2007 23:50:03
1A3A3A70-00000013.eml	19	Regular File	25-07-2007 23:49:49
1B5C05E6-00000007.eml	16	Regular File	20-06-2007 17:47:44
25BB4381-00000005.eml	3	Regular File	13-04-2007 00:28:26

**Message Content:** A large text block showing the details of the selected email message (0C130270-0000000D.eml). It includes headers, body content, and attachments.

**Custom Content Sources:** A pane showing evidence file system paths and options.

**Properties:** A status bar at the bottom showing 'Listed: 16 Selected: 1 Mantooth32 (1).E01/Partition 1 [109MB]/MANTOOTH [NTFS]/[root]/Users/Wes Mantooh/AppData/Local/Microsoft/Windows Mail/Local Folders/Inbox/0C130270-0000000D.eml'.

### 2. Look and bookmark 10 email messages?

0	0C130270-00000...	4	Regular File	24-07-2007...	
0	10AB12E1-00000...	3	Regular File	25-07-2007...	
0	1A3A3A70-00000...	19	Regular File	25-07-2007...	
0	1B5C05E6-00000...	16	Regular File	20-06-2007...	
0	25BB4381-00000...	3	Regular File	13-04-2007...	
0	26FC5471-00000...	9	Regular File	12-04-2007...	
0	2A29541D-00000...	14	Regular File	24-07-2007...	
0	31D0562C-00000...	24	Regular File	27-02-2007...	
0	3376666D-00000...	768	Regular File	13-07-2007...	
0	40A511AF-00000...	768	Regular File	12-07-2007...	

**3. An email was received by Wes Mantooth regarding a contact in Bujumbura, Africa. Locate the email and answer following:**

**a. When was the email sent**

08/01/2007

**b. List all the parties associated with the email and note down their email.**

RascoBadguy, chkwasher@comcast.com, dollarhyde86@comcast.com,  
skimmerman27@hotmail.com, molarman420@gmail.com

**c. Are there any attachment to the email? If so, what are the file names?**

Yes, there are attachments to the mail:

Confidential Business Letter.doc

**d. What is the originating email server ip address?**

66.196.96.95

**e. In which email box is this email located?**

In Outlook.pst file, inside Inbox.

**4. Use an Index search, attempt to identify the real names associated with discovered email addresses?**

Email Address	Name
chkwasher@comcast.com	John Washer
dollarhyde86@comcast.com	Wes Mantooth
molarman420@hotmail.com	Wes Mantooth
skimmerman27@hotmail.com	David Thomas
ckidd@swbell.net	RascoBadguy

**5. Wes Mantooth sent an email to his mom. What is his mom's email?**

toothfairy@mentaldental.com

---

```
From: "Wes Mantooth" <dollarhyde86@comcast.net>
To: <toothfairy@mentaldental.com>
Subject: Hey Mom
Date: Thu, 12 Jul 2007 17:36:36 -0600
MIME-Version: 1.0
Content-Type: multipart/mixed;
    boundary="===== _NextPart_000_0011_01C7C4AB.32251050"
X-Priority: 3
X-MSMail-Priority: Normal
X-Mailer: Microsoft Windows Mail 6.0.6000.16480
X-MimeOLE: Produced By Microsoft MimeOLE V6.0.6000.16480
```

This is a multi-part message in MIME format.

## 6. What event was he discussed with his mother?

He discussed about a Wedding Event with his mother. He mentioned that her mother needed his wedding pic which he got to know by his dad.

```
Hey there mom. How is it going?

Dad said that you needed a pic of me for the weding announclment?

Here is a good one.

Thanks for all your help with that. I am so busy with school, I don't =
know how I would have planned it!

Love ya!
```

## 7. Wes Mantooth has a friend by name Joan Acetone. Where does he work?

Yes, He had a friend by name Joan Acetone. He worked at "Arbys, Inc."

## 8. Wes Mantooth dad had been away for a while. Wes mentioned him in a letter to someone called "sweetie". Where was his dad?

Mantooth's dad was just released from jail. (Dear Sweetie.doc)

**9. Wes Mantooth has a list of names of possible associates, two of which are john washer and Simple Simon. What is the name of the file where this information is located?**

~arl730.xar

**10.The word is that Wes is talking to Rasco about ragging ATM Machine card readers. Use the case evidence to verify this rumor and identify who else was involved. What is the accomplice real name?**

David Thomas – skimmerman27@hotmail.com

A:\originalizing.txt. 1608.192.27.162  
From: David Thomas <skimmerman27@hotmail.com>  
To: <dollarhyde@comcast.net>  
Subject: Re: [REDACTED]  
Date: Mon, 23 Jul 2007 18:59:26 -0400  
Importance: Normal  
MIME-Version: 1.0  
Return-Path: skimmerman27@hotmail.com  
X-OriginalArrivalTime: 23 Jul 2007 22:59:26.0335 (UTC) FILETIME=[1E1FB8F0:01C7CD7D]  
--\_#abbcc-e0fe-4ffa-a15-a528c59f4b81\_=  
Content-Type: multipart/alternative;  
boundary="c7e19817-36b4-415b-ac3-6d031906a227"  
--\_c7e19817-36b4-415b-ac3-6d031906a227\_  
Content-Type: text/plain; charset="Windows-1252"  
Content-Transfer-Encoding: quoted-printable  
  
Dude....Rasco said to contact you....I picked up a thing and need to get it open and he said you had experience with ATM machines. Got one in my living room. You know a guy who may know a guy?  
=20  
Skimmerman  
  
PC Magazine=92s 2007 editors=92 choice for best web mail=97award-winning Windows Live Hotmail.  
http://imagine-windowslive.com/hotmail/?locale=3Den-us&cid=3DTXT\_TAGS&mag=ration\_HMW\_mini4\_pcmag\_0707=20  
--\_c7e19817-36b4-415b-ac3-6d031906a227\_=  
Content-Type: text/html; charset="Windows-1252"  
Content-Transfer-Encoding: quoted-printable  
  
<html>  
<head>  
<title>

**11.What is the washer's AIM username?**

Washergonebad

**Conclusion:**

1. After a thorough investigation into Wes Mantooth's flash drive, I believe there is sufficient evidence to support illegal activity in many areas. Wes Mantooth has supported this evidence with emails between John Washer, Rasco Badguy, and David Thomas.
2. In these emails, Wes Mantooth discusses check, ATM, credit card, and prescription fraud. Also David Thomas produced evidence that showed he stole an ATM machine and was trying to break into it. There are plenty of pictures sent in these emails to show specific activities in these fraudulent areas.
3. He also found on the flash drive are illegal drug activities which include the production and sales of methamphetamines and marijuana. On this flash drive, there were two partitions set up that include both an EXT2 and NTFS files system. The NTFS files system included the bulk of the space used on the flash drive, and almost all of the evidence. Typically Windows is unable to see a second partition on a flash drive and it is a great way to hide things, since most people tend to use windows based machines and not Linux.
4. There were also 22 encrypted files stored on this flash drive. Three of these files are included in the report but the third file was only a picture of the front and back on a 1 million dollar bill.

## **EXPERIMENT 9.a**

### **WEB BROWSER FORENSICS**

Browser forensic is mainly used for analyzing things like browsing history and general web activity of a PC to check for suspicious usage or content that has been accessed. This also refers to monitoring traffic on a webpage and analysis of LOG files from server to get actual information about targeted machine.

#### *Working of Web Browser*

A Web browser is actually a software application that runs on your Internet-connected computer. It allows you to view Web pages, as well as use other content and technologies such as video, graphics files, and digital certificates, to name a few.

Browsers are able to display Web pages largely in part to an underlying Web protocol called HyperText Transfer Protocol (HTTP). HTTP defines how messages are formatted and transmitted, and what actions Web servers and browsers should take in response to various commands. It is what allows Web clients and Web servers to communicate with each other. When you enter a Web address (URL) in your browser, this actually sends an HTTP command to the Web server directing it to fetch and transmit the requested Web page and display the information in your browser. All Web servers serving Web sites and pages support the HTTP protocol.

#### *Web browser forensic artifacts*

Of course, each web browser leaves its own individual artifacts in the operating system. Types of artifacts from the web browser can vary depending on the version of the web browser. Typically, when researching artifacts of web browsers, you can extract the following types of artifacts:

- History
- Cache
- Cookies
- Typed URLs

- Sessions
- Most visited sites

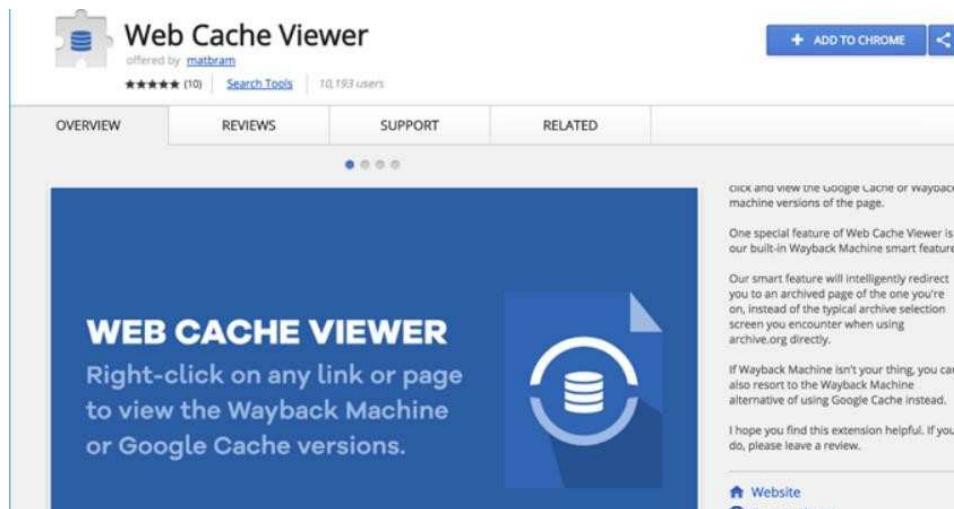
## History and Favourites

### View Cache Files Using Chrome Extension

1. Open your Chrome browser and go to the Chrome Web Store.

2. Search for "Web Cache Viewer" and add it on your browser.

After installation, you can right-click on any web link in Chrome and choose "Web Cache Viewer". You can choose to view the cached page from either Wayback Machine or Google archive.



## Cookie Analysis

Name	Value	Domain	S...	P...	Expires (GMT)	H...	S...
urlgen	"\\"2409:4053:18b:3cd8:847:d933:19d9:...	instag...	9...	/	Session	T...	T...
ur...	PRN	instag...	6 B	/	Session	T...	T...
shbts	1588094300.3197896	instag...	2...	/	Tue May 05 ...	T...	T...
shbid	1152	instag...	9 B	/	Tue May 05 ...	T...	T...
sessionid	1061472660%3AKtNCX4H2LMpTtL%3...	instag...	4...	/	Wed Apr 28 ...	T...	T...
ds_user_id	1061472660	instag...	2...	/	Mon Jul 27 ...		T...
csrfToken	w2YBFtrjPi1qg6kcm4Hcqhv2rX02PhT	instag...	4...	/	Tue Apr 27 ...		T...
fbm_12402...	base_domain=.instagram.com	instag...	4...	/	Sun Apr 11 ...		T...
ig_did	68A11E5C-E631-464E-BC47-A642DBB...	instag...	4...	/	Tue Apr 09 ...	T...	T...
mid	WzugyAALAAGFhiqqx9bFPj7i5iTn	instag...	3...	/	Mon Jun 28 ...		

# EXPERIMENT 9.b

## Case Study: Granny Clampet

### **1. Export all the registry files from the evidence?**

For exporting config registry files go to following path :

/root]\WINDOWS\system32\config\software

Name	Size	Type	Date Modif...
systemprofile	1	Directory	03-07-2009...
\$130	4	NTFS Index ...	04-07-2009...
\$130	4	Regular File	30-06-2009...
AppEvent.Evt	192	Regular File	04-07-2009...
default	256	Regular File	04-07-2009...
default.LOG	8	Regular File	04-07-2009...
default.sav	92	Regular File	29-06-2009...
SAM	256	Regular File	04-07-2009...
SAM.LOG	1	Regular File	04-07-2009...
SecEvent.Evt	64	Regular File	29-06-2009...
SECURITY	256	Regular File	04-07-2009...
SECURITY.LOG	1	Regular File	04-07-2009...
software	15,616	Regular File	04-07-2009...
software.LOG	12	Regular File	04-07-2009...
software.sav	644	Regular File	29-06-2009...
SysEvent.Evt	256	Regular File	04-07-2009...
system	4,096	Regular File	04-07-2009...
system.LOG	1	Regular File	04-07-2009...
system.sav	892	Regular File	29-06-2009...
TempKey.LOG	1	Regular File	29-06-2009...
userdiff	256	Regular File	29-06-2009...
userdiff.LOG	1	Regular File	29-06-2009...

For exporting NTUSER.DAT file go to following path:

/root]\Documents and Settings\Granny\NTUSER.DAT

Name	Size	Type	Date Modif...
Application Data	1	Directory	03-07-2009...
Cookies	1	Directory	04-07-2009...
Desktop	1	Directory	04-07-2009...
Favorites	1	Directory	04-07-2009...
Local Settings	1	Directory	03-07-2009...
My Documents	1	Directory	04-07-2009...
NetHood	1	Directory	03-07-2009...
PrintHood	1	Directory	03-07-2009...
Recent	1	Directory	04-07-2009...
SendTo	1	Directory	03-07-2009...
Start Menu	1	Directory	03-07-2009...
Templates	1	Directory	03-07-2009...
UserData	1	Directory	04-07-2009...
\$130	4	NTFS Index ...	04-07-2009...
NTUSER.DAT	1,024	Regular File	03-07-2009...
ntuser.dat.LOG	1	Regular File	04-07-2009...
ntuser.ini	1	Regular File	03-07-2009...

**2. Police suspect that Internet activity played a role, what internet sites did the suspect last visit?**

ab] url1	REG_SZ	http://www.google.com/
ab] url2	REG_SZ	http://www.live.com/
ab] url3	REG_SZ	http://www.yahoo.com/
ab] url4	REG_SZ	http://www.dogpile.com/
ab] url5	REG_SZ	http://www.microsoft.com/isapi/redir.dll?prd=ie&p...

*NTUSER.DAT[3548].tmp\Software\Microsoft\Internet Explorer\TypedURLs*

**3. What was Granny's Internet Explorer home page?**

<http://www.dogpile.com/>



*NTUSER.DAT[3548].tmp\Software\Microsoft\Internet Explorer>Main*

**4. A. What printer was Granny using?**

SnagIt 7

Name	Type	Data
DeviceOld	REG_SZ	SnagIt 7,winspool,Ne00:

*NTUSER.DAT[3548].tmp\Printers*

**B. What was Buddy's default printer?**

SnagIt 7

Name	Type	Data
DeviceOld	REG_SZ	SnagIt 7,winspool,Ne00:

*NTUSER.DAT[2269].tmp\Printers*

**C. What was Jethro's default printer?**

Microsoft Office Document Image Writer

Name	Type	Data
DeviceOld	REG_SZ	Microsoft Office Document Image Writer,winspool,N...

NTUSER.DAT[2242].tmp\Printers

## 5. What was the last location that Granny downloaded something from using Internet Explorer?

Save Directory REG\_SZ C:\Documents and Settings\Granny\My Documents\  
 NTUSER.DAT[3548].tmp\Software\Microsoft\Internet Explorer>Main

## 6. Generate a report based on Granny's NTUSER.DAT file

Granny's NTUSER.DAT file

Page 1 of 11

### Registry Information

Registry Viewer Report 

### Granny's NTUSER.DAT file

#### Software\Microsoft\Internet Explorer\TypedURLs

Last Written Time 7/3/2009 22:34:22 UTC

Name	Type	Data
url1	REG_SZ	http://www.google.com/
url2	REG_SZ	http://www.live.com/
url3	REG_SZ	http://www.yahoo.com/
url4	REG_SZ	http://www.dogpile.com/
url5	REG_SZ	http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnhome

#### Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

**7. Jethro maintained an email account. See if you can provide evidence of this account from the Inetcomm server Passwords area of protected storage system provider?**

Jethro email account

[smurferator@gmail.com](mailto:smurferator@gmail.com)

*NTUSER.DAT[2242].tmp\Software\Microsoft\Protected Storage System Provider\S-1-5-21-1409082233-2049760794-839522115-1003\Internet Explorer\Internet Explorer\email:StringData*

**8. A. Who is the registered owner and what is the registered organization?**

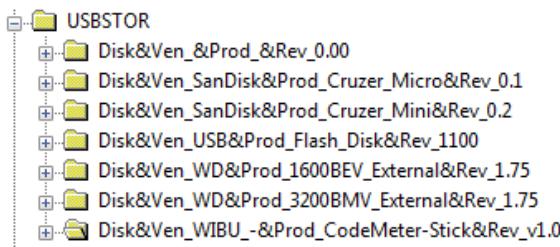
Jed, Clampett Industries

 RegisteredOrganization	REG_SZ	Clampett Industries
 RegisteredOwner	REG_SZ	Jed

*software[2314].tmp\Microsoft\Windows NT\CurrentVersion*

**B. Investigators found some portable devices in the parlor? Can you give them any that can help them determine if any external or portable devices have been connected to the computer?**

Yes, Investigators found some portable devices. They are listed as follows :



*system[2310].tmp\ControlSet001\Enum\USBSTOR*

**C. Document when Granny last logged on to this machine**

7/3/2009 23:10:23 UTC

*SAM[2320].tmp\SAM\Domains\Account\Users\000003EE*

Key Properties	
Last Written Time	04-07-2009 03:38:24 UTC
SID unique identifier	1006
User Name	Granny
Logon Count	13
Last Logon Time	03-07-2009 23:10:23 UTC
Last Password Change Time	10-10-2009 04:24:39 UTC
Expiration Time	Never
Invalid Logon Count	2
Last Failed Login Time	04-07-2009 03:38:24 UTC
Account Disabled	false

SAM\SAM\Domains\Account\Users\000003EE

#### D. Generate a Report based on the System File.

Registry Information Registry Viewer Report 

#### Granny's NTUSER.DAT file

##### Software\Microsoft\Internet Explorer\TypedURLs

Last Written Time	7/3/2009 22:34:22 UTC
<hr/>	
Name	Type
ur11	REG_SZ
ur12	REG_SZ
ur13	REG_SZ
ur14	REG_SZ
ur15	REG_SZ

http://www.google.com/  
http://www.live.com/  
http://www.yahoo.com/  
http://www.dogpile.com/  
http://www.microsoft.com/isapi/redir.dll?prd=ie&pver=6&ar=msnhome

##### Software\Microsoft\Windows\CurrentVersion\Explorer\RecentDocs

Last Written Time	7/3/2009 23:31:27 UTC
<hr/>	
Name	Type
MRUListEx	REG_BINARY
27	REG_BINARY
26	REG_BINARY
25	REG_BINARY
24	REG_BINARY
0	REG_BINARY

MRU ordered list : 27, 26, 25, 24, 0, 23, 22, 16, 17, 21, 20, 1, 2, 19, 18, 15, 10, 14, 13, 12, 11, 9, 8, 7, 6, 5, 4, 3  
 Shortcut Target Name : Customers.xls  
 Shortcut Name (ASCII) : Customers.lnk  
 Shortcut Name (Unicode) : Customers.lnk  
 Shortcut Target Name : New Microsoft Excel Worksheet.xls  
 Shortcut Name (ASCII) : New Microsoft Excel Worksheet.lnk  
 Shortcut Name (Unicode) : New Microsoft Excel Worksheet.lnk  
 Shortcut Target Name : Apology number 2.doc  
 Shortcut Name (ASCII) : Apology number 2.lnk  
 Shortcut Name (Unicode) : Apology number 2.lnk  
 Shortcut Target Name : Apology.doc  
 Shortcut Name (ASCII) : Apology.lnk  
 Shortcut Name (Unicode) : Apology.lnk  
 Shortcut Target Name : Thank You Letter.doc  
 Shortcut Name (ASCII) : Thank You Letter.lnk  
 Shortcut Name (Unicode) : Thank You Letter.lnk