SANS

www.sans.org

FORENSICS 508

ADVANCED DIGITAL FORENSICS AND INCIDENT RESPONSE

SRL Intrusion – Incident Response and Forensics Exercise Workbook

The right security training for your staff, at the right time, in the right location.

For508_Wkbk_A04_03

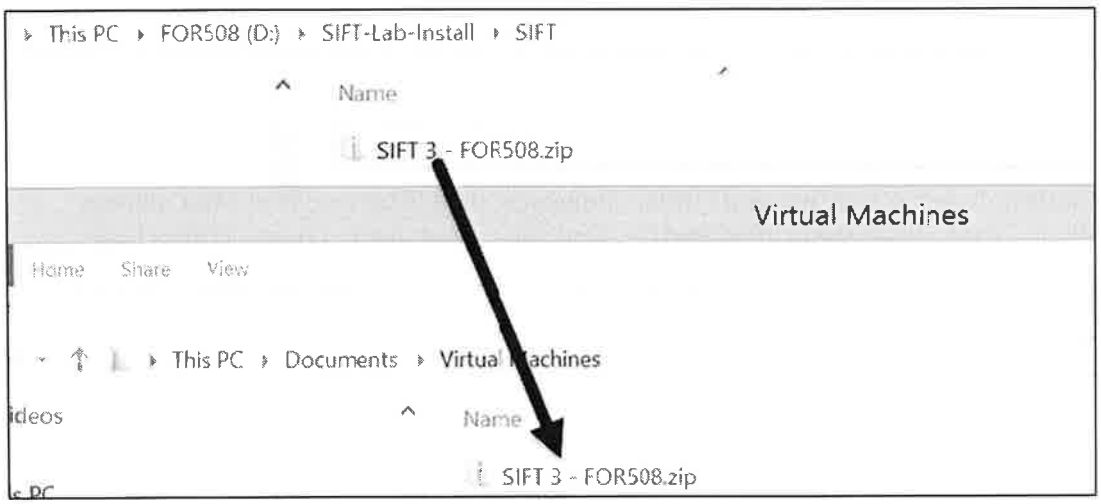
Exercise 0 – SIFT Lab Installation

Objectives

- Install and prepare your lab workstation for digital forensic analysis for the week.
- Move the evidence files gathered during IR evidence collection phase to our analysis platform for further scrutiny.

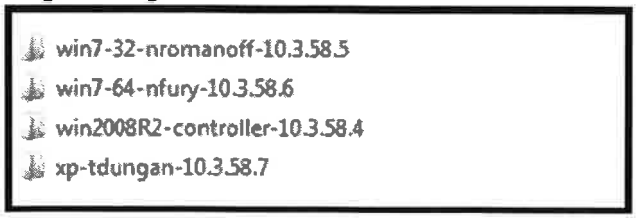
Exercise Preparation

1. *Install VMware Workstation/Fusion or VMware Player from vmware.com*
   - https://www.vmware.com/products/workstation
   - https://www.vmware.com/products/player
   - http://www.vmware.com/products/fusion
2. *Install RedLine on your Windows Host or VM*
   - Located on your USB under `\SIFT-Lab-Install\redline`
   - You might need to install .NET before the installation. The installed for the FULL version of .NET is found in `\SIFT-Lab-Install\redline`
   - It is CRITICAL that you install and make sure RedLine functions before the beginning of Section 2 in the course. Do not simply wait to install it minutes prior to needing it for the RedLine exercises
   - It is also highly recommended that you install RedLine on your host system instead of your virtual machine to increase processing power during memory analysis
   - You can check to see if Redline works correctly by simply executing it and getting to the "Main Screen"
3. *If Windows Host, Install 7zip*
   - Located on your USB under `\SIFT-Lab-Install\7zip`
4. *Copy and Install the SIFT Workstation from:*
   - Located on your USB under `\SIFT-Lab-Install\SIFT\SIFT 3 - FOR508.zip`

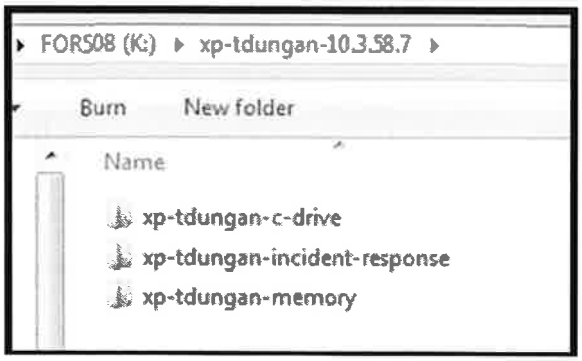


What is on the USB?

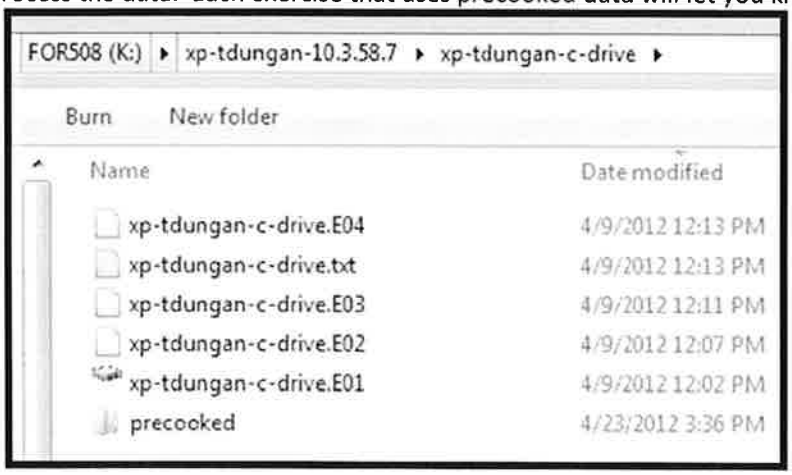
1. The USB Contains the data from the 4 systems potentially involved in the intrusion breach.
2. The name of each of the directories is listed by:
    - Operating System
    - System Owner
    - IP Address
    - **`OS-User-IP`**
        - Example: **`xp-tdungan-10.3.58.7`**



3. In each of the directories you will find 3 additional directories.

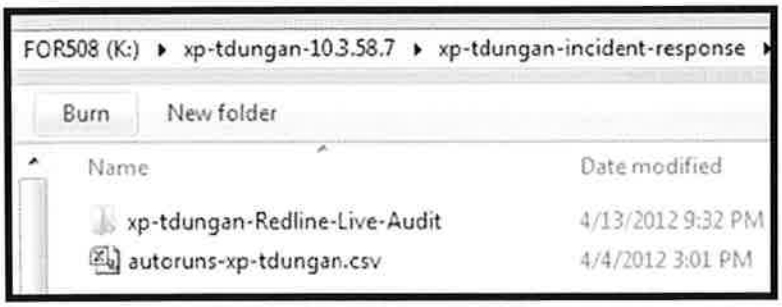


a. <u>c-drive directory</u>
    - This is the location of the collected hard drive images.
    - On the **`xp-tdungan-c-drive`** directory you will also see a **`precooked`** directory.
    - The precooked is where we have some data preformatted in case certain tools take too long to process the data. Each exercise that uses precooked data will let you know.



b. Incident Response directory

- This is the location of data collected during Incident Response on each system.



c. Memory directory

- This is the location of the memory image collected during Incident Response.

Exercise - Setting up your Lab - Preparation for the Week

1. Tweak the settings of your VMware configuration to allow for more memory and processor power as your system will allow. Note: Do not ever increase it beyond the maximum recommended settings.
2. Launch VMWare Player, Fusion, or Workstation
    1. File -> Open ->
    2. **`\Path-To\Virtual Machines\SIFT 3 - FOR508\SIFT 3 - FOR508.vmx`**

3. Upgrade your virtual machine if you can -> VM -> Upgrade or Change Version (Follow Wizard).

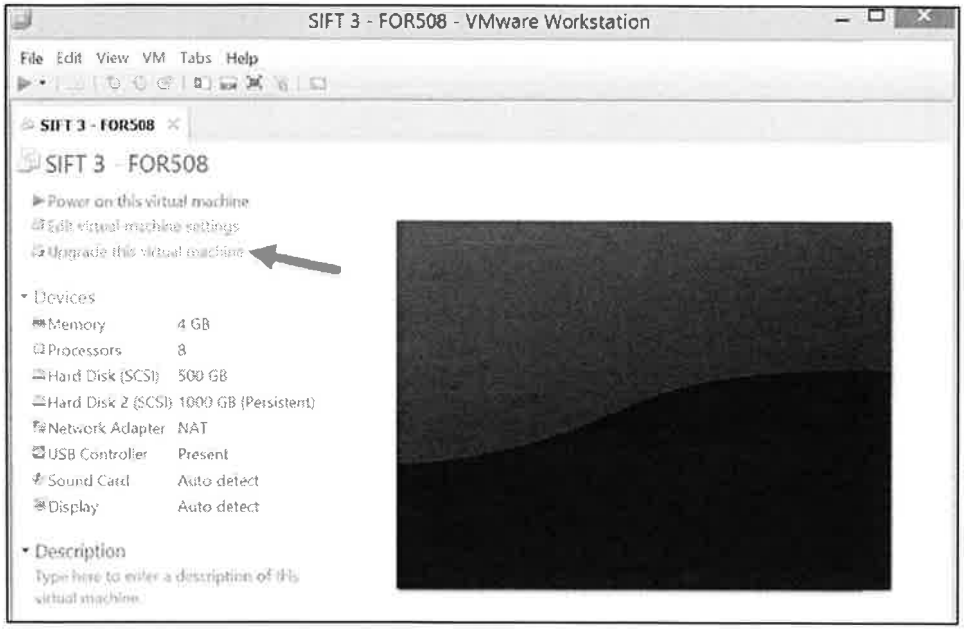


Figure 1 - Update Version to Latest

4. Adjust Memory -> Select "Edit Virtual Machine Settings" -> Select Memory

    NOTE: Do not give your VM more than ½ of your machines memory. If your machine slows down as a result, reduce the amount of memory allocated to the VM.

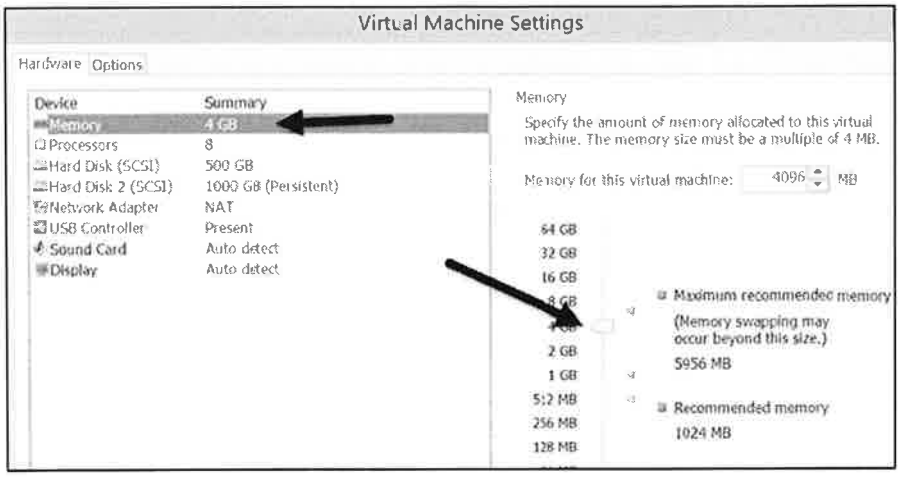
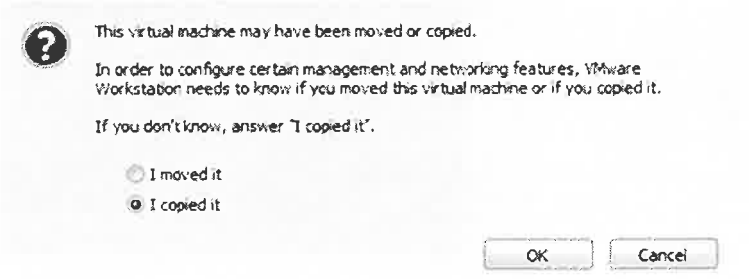
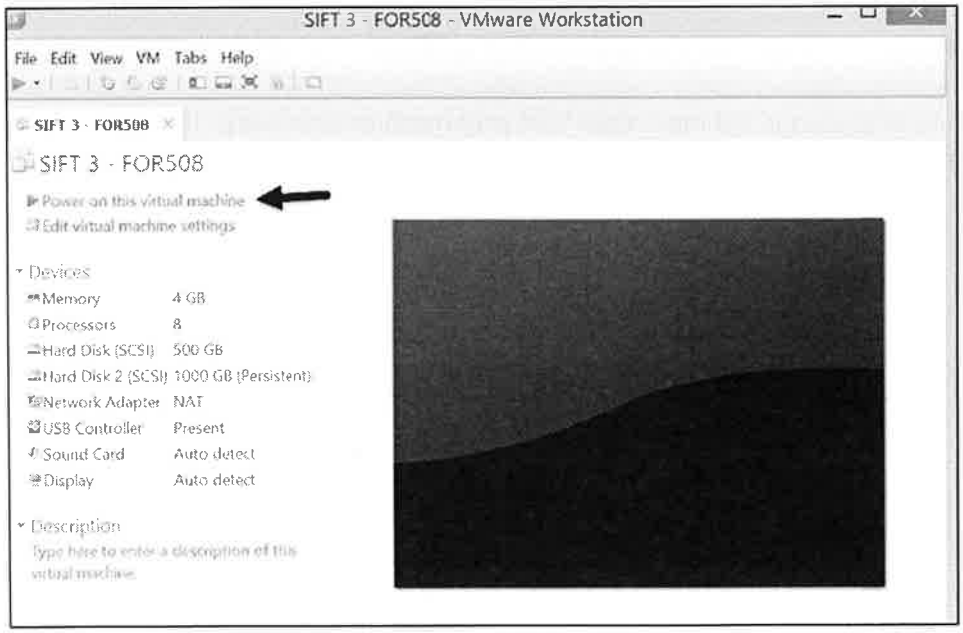


Figure 2 - Edit Memory Configuration

5. Power on Your Virtual Machine.
   1. Press "Power on virtual machine" and select "I copied it".

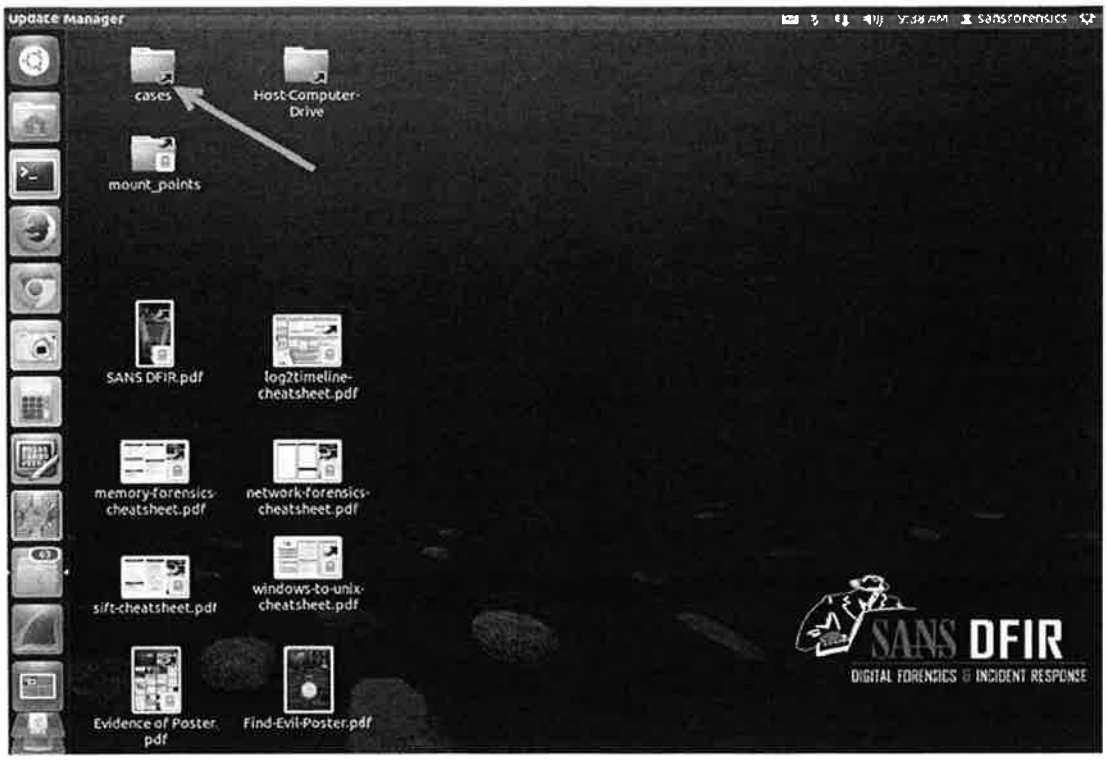


6. Start your SIFT VMware Workstation in VMware Workstation.



7. Login the VMware machine.
   - LOGIN = **`sansforensics`**
   - PASSWORD = **`forensics`**

8. Double Click on the SIFT Workstation icon and browse to the **`/cases`** directory (Click on the Filesystem Icon ->**`/cases`**).

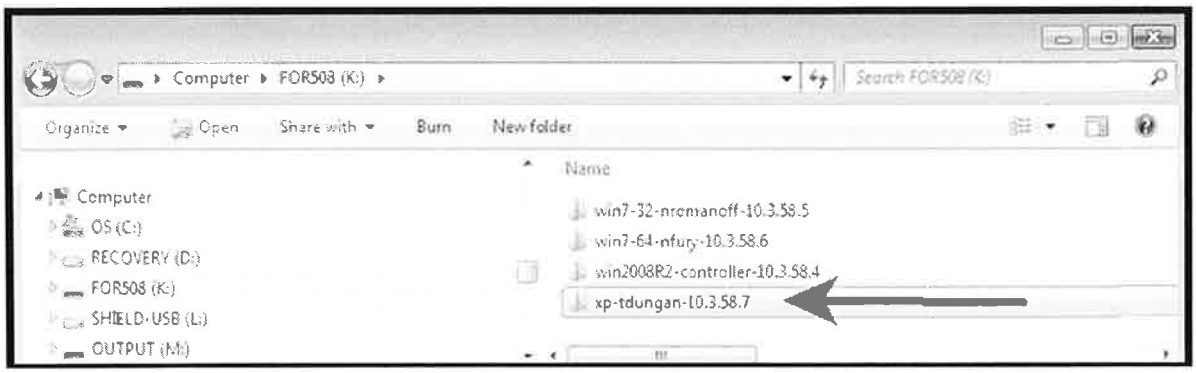


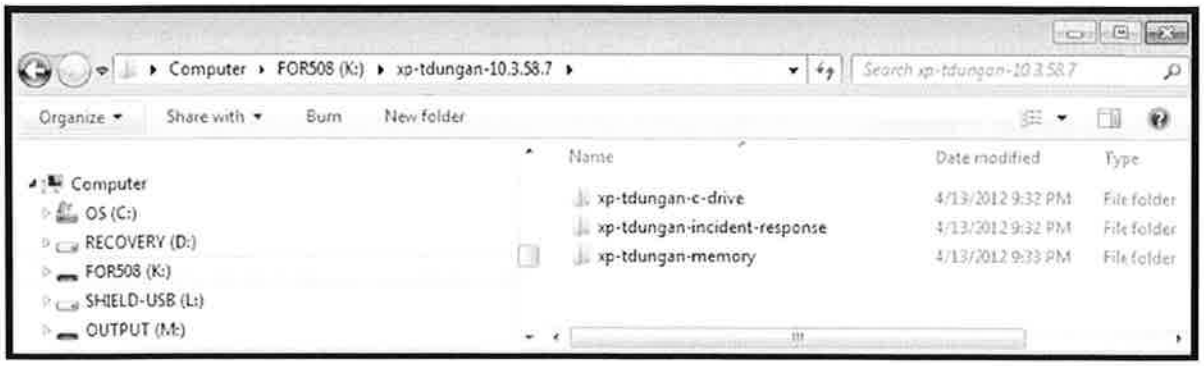
9. Insert the FOR508 Course USB into your Host System. Note if you know how to connect the USB directly to the SIFT, the following can be done inside the SIFT itself or by using shared directories. If you are not familiar with those techniques I have found for those new to VMware simply showing files can be dragged into the VM is quite useful.

10. Begin copying files to the SIFT workstation we will use for the core exercises for the course. We copy them into the SIFT to increase speed and processing. While the copying process can be slow, it will save hours of agony later in the course.

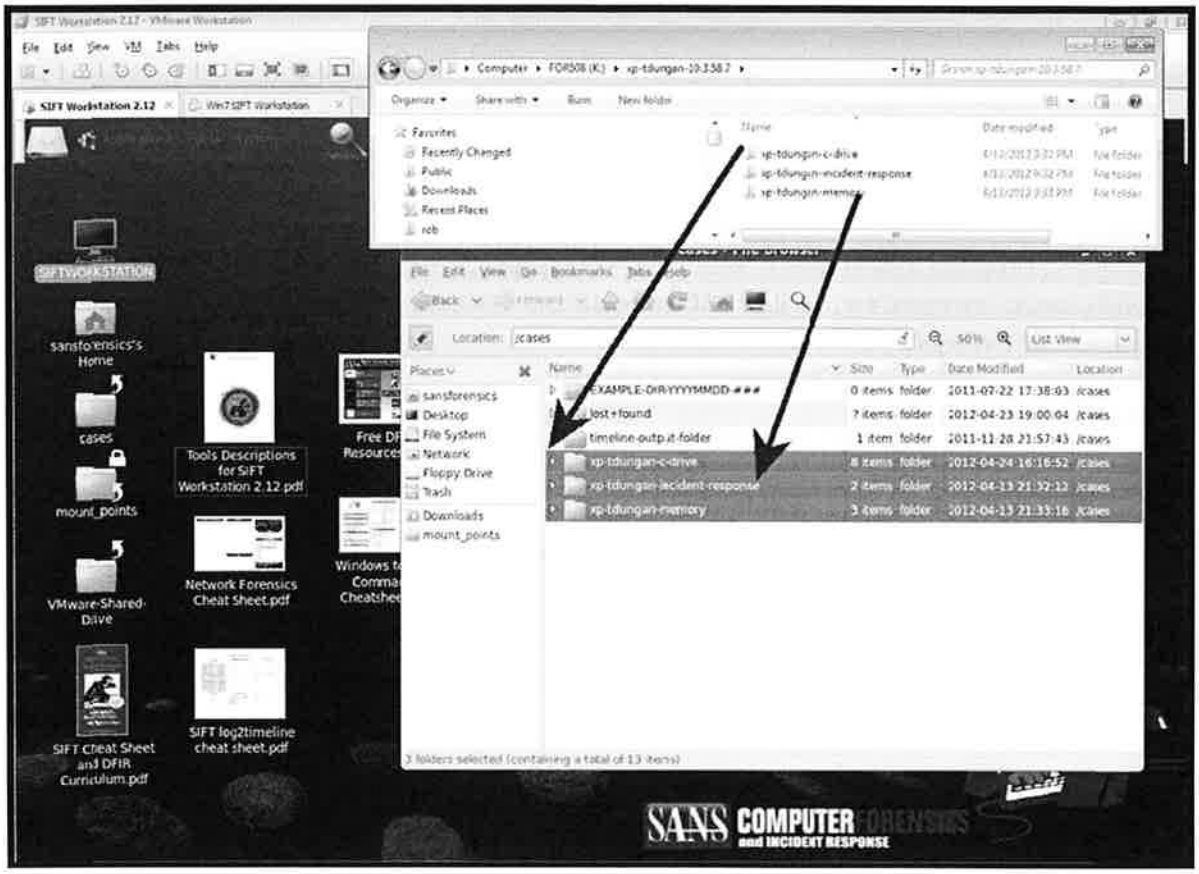
11. *On the FOR508 USB, browse to the directory and open it:*

- **`DriveLetter:>\xp-tdungan-10.3.58.7`**

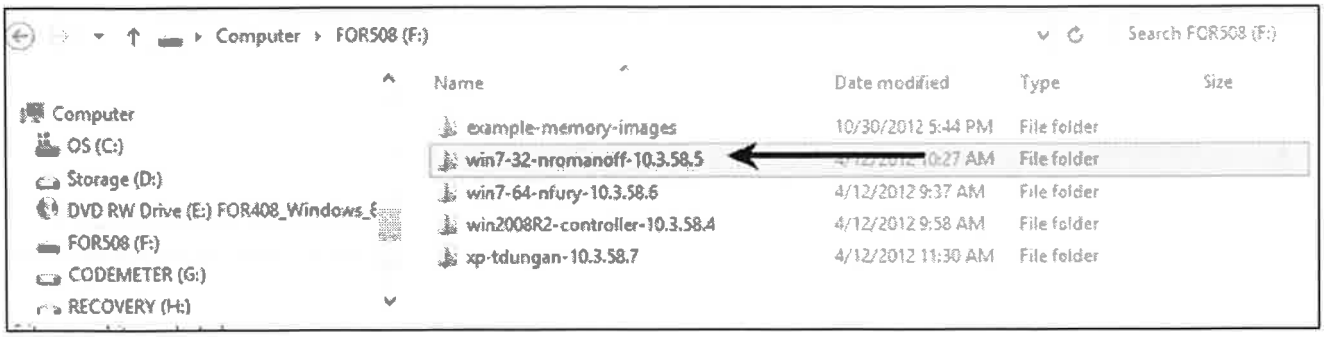


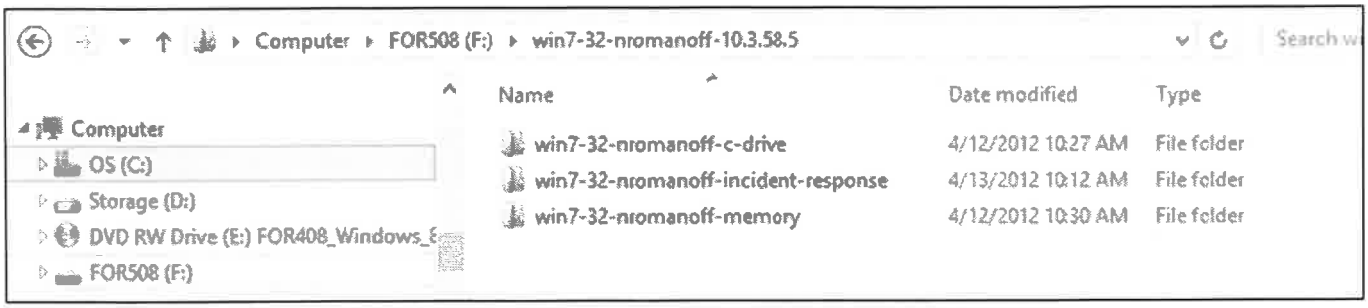


12. Drag and Drop the 3 directories "**`xp-tdungan-c-drive, xp-tdungan-incident-response, xp-tdungan-memory`**" to your **`/cases/`** folder into your Ubuntu SIFT Workstation Virtual Machine.

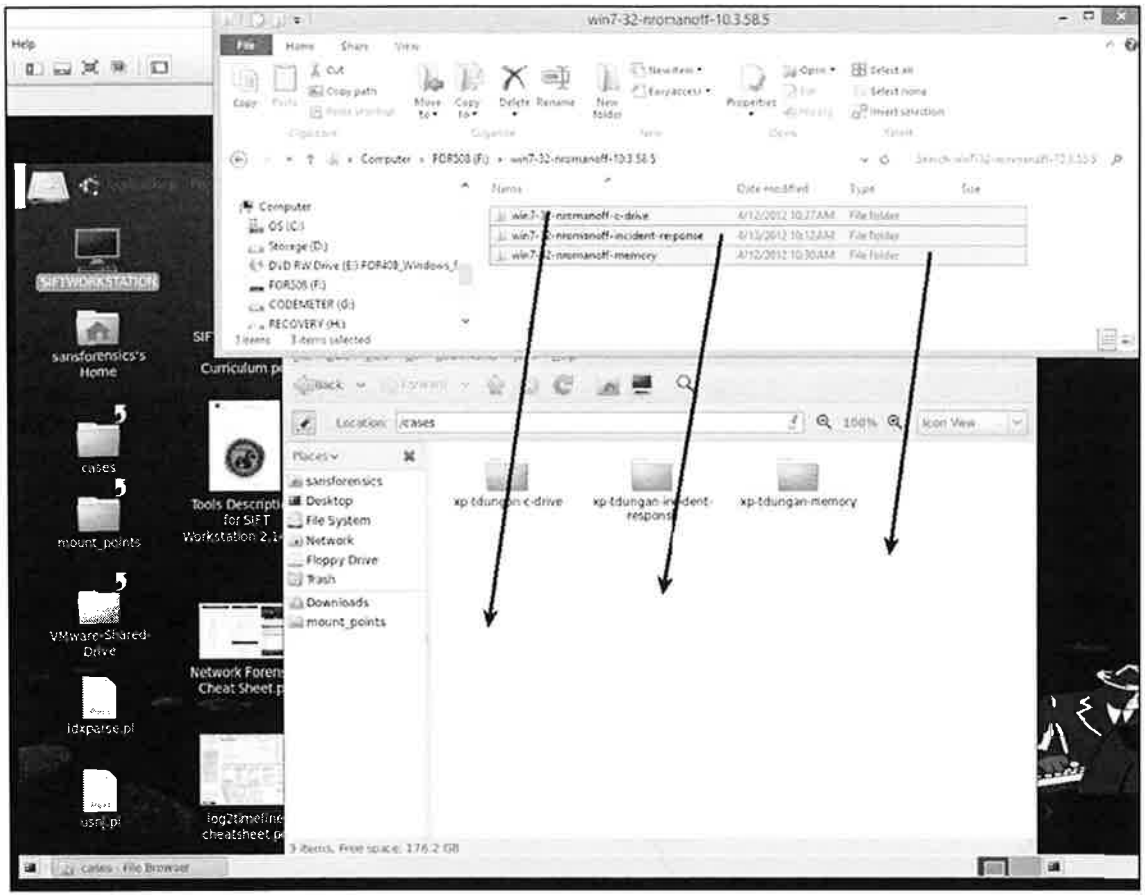


13. Change directories to the **`DriveLetter:>\win7-32-nromanoff-10.3.58.5`**

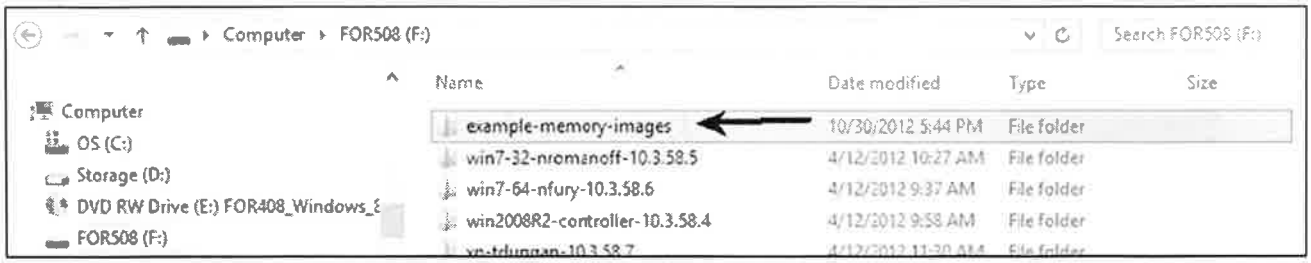


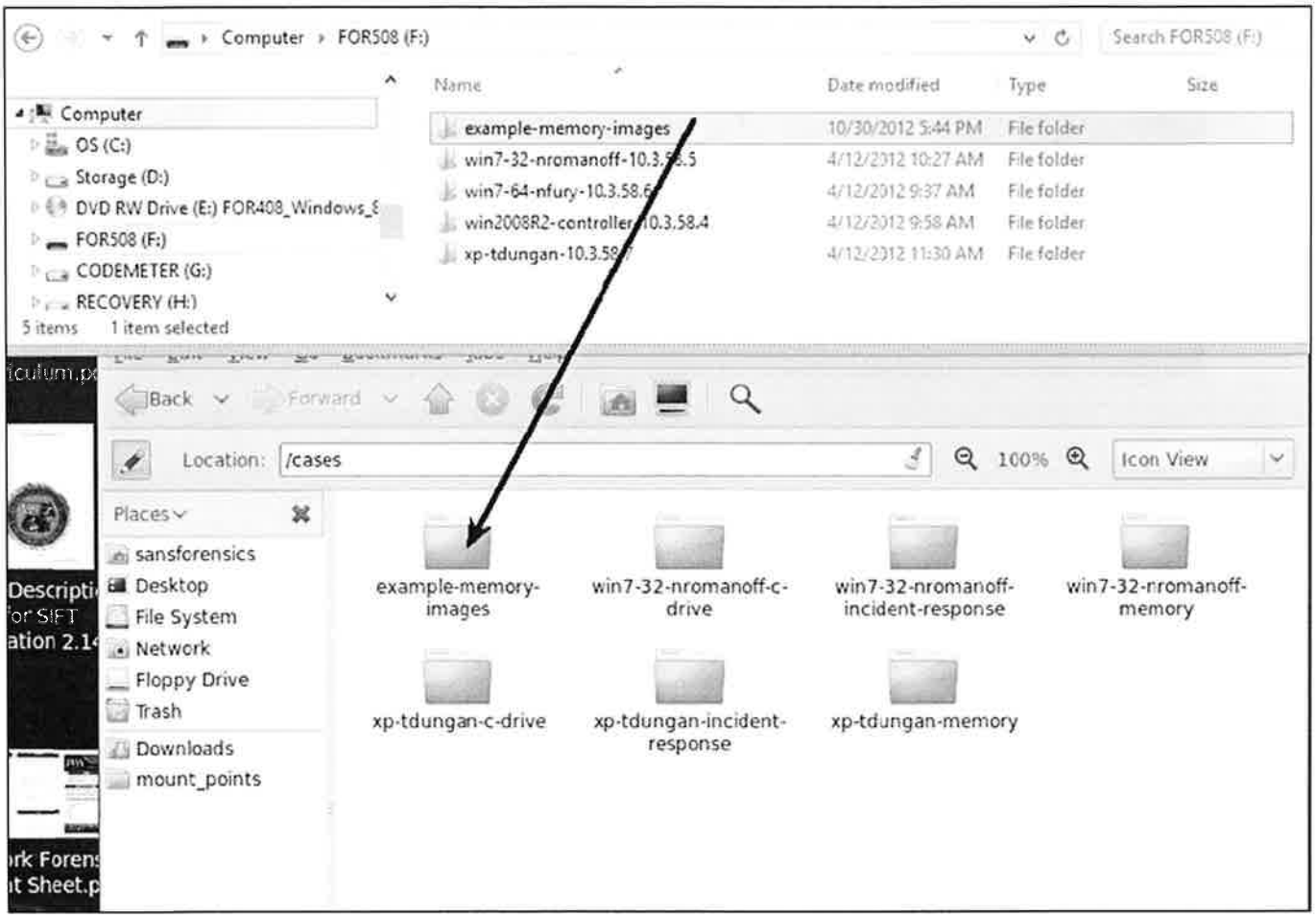


14. Drag and Drop the 3 directories "**`win7-32-nromanoff-c-drive, win7-32-nromanoff-incident-response, win7-32-nromanoff-memory`**" to your **`/cases/`** folder into your Ubuntu SIFT Workstation Virtual Machine.

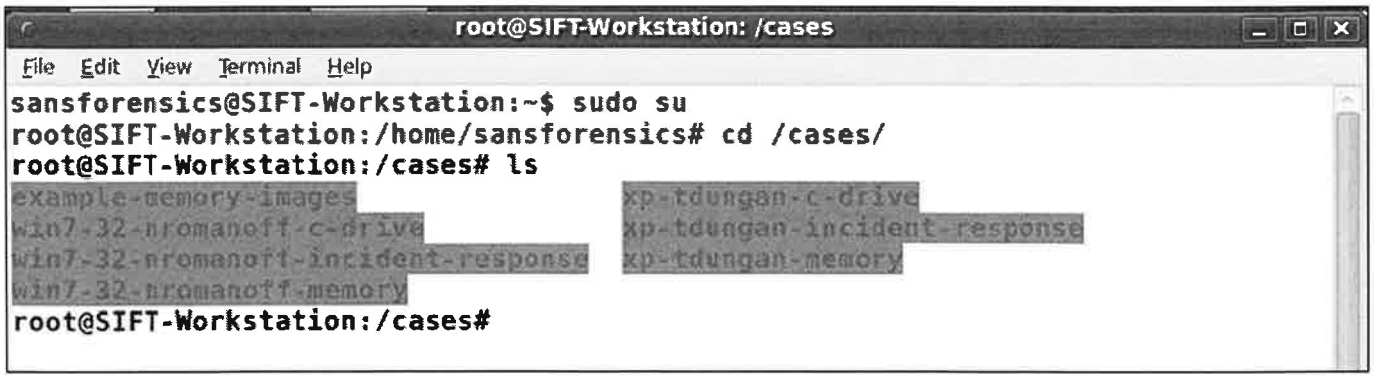

win7-32-nromanoff-10.3.58.5
SIFTWORKSTATION
sansforensics's Home
cases
mount_points
VMware Shared Drive
idxparse.pl
usnj.pl
Curriculum p
Tools Descripti for SIFT Workstation 2.1
Network Foren Cheat Sheet p
log2timeline cheatsheet p
Computer
OS (C:)
Storage (D:)
DVD RW Drive (E:) FOR408_Windows_f
FOR508 (F:)
CODEMETER (G:)
RECOVERY (H:)
win7-32-nromanoff-c-drive
win7-32-nromanoff-incident-response
win7-32-nromanoff-memory
Location: /cases
sansforensics
Desktop
File System
Network
Floppy Drive
Trash
Downloads
mount_points
xp-tdungan-c-drive
xp-tdungan-incident-respons
xp-tdungan-memory

15. Change directories to the **`DriveLetter:>\example-memory-images`** and copy the directory to the **`/cases`** directory.





16. Check your work! When completed, change directories to the **/cases** directory in your terminal and run the file listing command "**ls**". You should see all the directories here that you just copied.



17. **You have successfully prepped your SIFT Workstation for FOR508 – Advanced Forensics Analysis and Incident Response for the first part of the week!!**

STARK RESEARCH LABS

ADVANCED ADVERSARY INTRUSION SCENARIO

Scenario Designed and Created by Rob Lee

APT Team - John Strand and Timothy Tomes

A 3-4 LETTER GOVERNMENT AGENCY CONTACTS YOUR COMPANY BY PHONE

"WE HAVE SEEN A FEW HUNDRED MEGABYTES OF SENSITIVE DATA LEAVE YOUR NETWORK BOUND FOR A FOREIGN COUNTRY."

"DON'T ASK HOW WE KNOW, BUT YOU MIGHT WANT TO CHECK **10.3.58.7** ON YOUR NETWORK"

COMPANY INFORMATION:

Stark Research Labs is a government sponsored laboratory that researches specialized metal alloys and bio engineering capabilities. Lately, SRL has been tasked to find the secret and once lost alloy formula for VIBRANIUM-Alloy. The lead researcher, Timothy Dungan, has been making progress and it looks like with the help of others he was finally able to replicate the formula

M. Hill is the current SRL-LABS Director and Administrator appointed by the President.

ONE WEEK AGO – MARCH 31, 2012 @ 16:11:16

```
From "Timothy Dungan" Sat Mar 31 16:11:16 2012
From: "Timothy Dungan" <SMTP:TDUNGAN@STARK-RESEARCH-LABS.COM>
Subject: Test Success - Alloy Combination
To: 'mhill.shield@yahoo.com'
Cc: 'nromanoff@stark-research-labs.com'
Date: Sat, 31 Mar 2012 16:11:16 +0000

We have finally replicated the long lost secret of Vibranium-Iron alloy.

After months of testing we have found that the secret to combing the alloy was
accidentally discovered when a lab student (Peter of Horizon Labs) stopped by.  See
attached report.  Next Steps?

 -Timothy
```

EXCITED, TIMOTHY DUNGAN POSTS PUBLICALLY ABOUT THE FIND!

PUBLIC TWEET ON 31 MAR 2012

THE VERY NEXT DAY, MARIA HILL RESPONDS – ARPIL 1, 2012

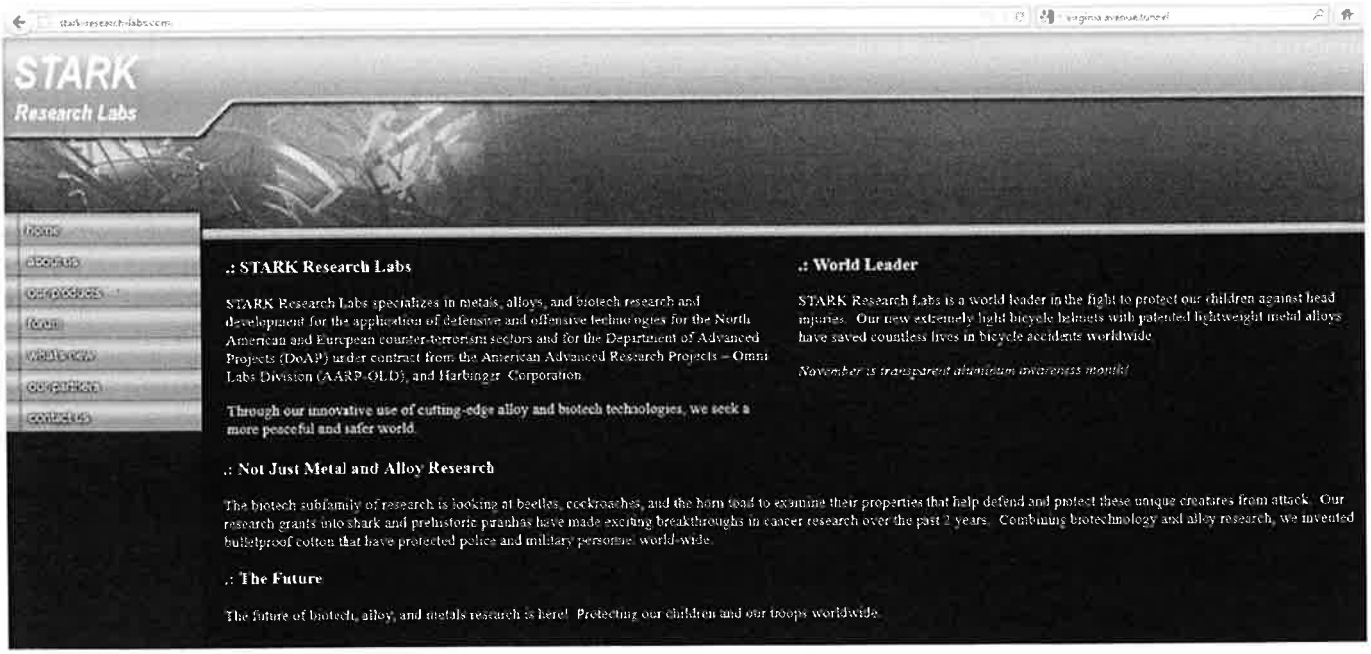
```
Date: Sun, 1 Apr 2012 07:08:48 -0700 (PDT)
From: Maria Hill <mhill.shield@yahoo.com>
Reply-To: Maria Hill <mhill.shield@yahoo.com>
Subject: Re: Test Success - Alloy Combination
To: Timothy Dungan <tdungan@stark-research-labs.com>
In-Reply-To: <1331844046.4518.YahooMailNeo@web122203.mail.ne1.yahoo.com>
---2079400718-714555717-1333289328=:84815

Content-type: text/plain

Timothy,

This is wonderful news.  Incredible.  Peter from Horizon labs should be
commended.  We will immediately move to phase II.  Hopefully HYDRA has no real idea
of our plans.  Hydra has been behind specific APT attacks originating from Madripoor
in Asia.  That has been their base of operations for quite some time.  The nation
was taken over by HYDRA with Madame Hydra as de facto ruler, using the nation to
finance terrorist plots against the world.  HYDRA is a criminal organization
dedicated to the achievement of world domination through terrorist and subversive
activities on various fronts, resulting in a fascist New World Order. Since the
takeover, this Asian nation has started to infiltrate many sectors in almost every
industry stealing economic information, intellectual property, and targeting key
executives with spear phishing attacks.

Be on the lookout for anything suspicious as Im sure word will get out that we made
a breakthrough.  Im positive they have spies inside our operations at Stark Research
Labs.

At all costs, please make sure you secure any information regarding the Star Fury
projects, our undercover agent lists, our backstopped credit card accounts, and any
information regarding our new base in DC.  Finally, you have to protect the key
information to the secret formula at all costs.  Having HYDRA find and recover any
of this information could be devastating to our operations.

Thanks again for this wonderful news and stay in contact.  Looking forward to
hearing more updates.

-MH
```

FAST FORWARD – APRIL 6, 2012 @5PM

The SRL IRT is called into action. Your mission, if you choose to accept it, is to answer the following questions. → High level question, not Tech specific one

1. Did a breach occur?
2. What systems were compromised?
3. What was taken?
4. How did they breach our network?
5. Where did they go?
6. Who compromised us?
7. What malware was used?
8. What should we do?

COMPANY WEBSITE: WWW.STARK-RESEARCH-LABS.COM



COMPANY NETWORK

Enterprise network with over 1,000 users and systems. A few key systems in SRL subnet network are as follows.

- 10.3.58.4 Win2008R2 Domain Controller
- 10.3.58.5 Win7-SP1-32bit N. Romanoff Workstation
- 10.3.58.6 Win7-SP1-64bit N. Fury Workstation
- 10.3.58.7 WinXP-SP3-32bit Timothy Dungan Workstation

Internal Networks

10.3.58.X – Workstation and Lab Stations

10.3.16.X – Workstation and Lab Stations

FalconIII – VPN Concentrator

SRL-LABS BASE DOMAIN INFORMATION

- Full auditing turned on per recommended guidelines.

- Users are restricted to being users (cannot even install a program if they wanted to).
- Windows 2008R2SP1 Domain Controller
- Systems installed and have real software on it that is used (Office, Adobe, Skype, Tweetdeck, E-mail, Dropbox, Firefox, Chrome).
- Fully patched (Patches are automatically installed).
- Enterprise Incident Response agents (F-Response Enterprise).
- Enterprise A/V and On-Scan capability (McAfee Endpoint Protection —Anti-virus, Anti-spyware, Safe surfing, Anti-spam, Device Control, Onsite Management, Host Intrusion Prevention (HIPS)).
- Network Using HBSS (Host Based Security System – Per DOD Recommendations)
- Firewall only allowed inbound 25,80 and outbound 25, 80, 443 only.
- Users have been "using" this network for over a year prior to the attack. That way it has the look and feel of something real. These users have setup social media (yes they are on twitter... you might be friends with them), e-mail, Skype, etc. Each character user has a backstory and a reason to be there working.
- Rsydow is the Domain Administrator
- SRL-HELPDESK Account is local admin account w/shared password

ADDITIONAL INFORMATION REGARDING SRL DOMAIN

- Local Admin User (SRL-Helpdesk) found on each system w/same password.
- Not every user has migrated to Win7 and Win2008. We do still have some legacy WinXP systems.
- Most of the employees telecommute from home to the lab. The VPN concentrator is located on the \\FALCONIII system. Most users RDP into their systems from the VPN.

USER'S POTENTIALLY COMPROMISED:

TIMOTHY DUNGAN – 10.3.58.7

SOCIAL MEDIA ACCOUNTS

- LINKEDIN:
  - http://www.linkedin.com/pub/timothy-dungan/3a/153/678
- Twitter: @TimothyDungan
  - https://twitter.com/#!/TimothyDungan
- Google+:
  - https://plus.google.com/104381120365479308959/posts
- Facebook User

- Gmail Account

SENSITIVE DATA ON 10.3.58.7

Metal and Alloy Research

This page intentionally left blank.

Exercise 2 – Mounting Evidence Using SIFT

Objectives

- Mount an acquired disk image into the SIFT workstation so you can see the files and folders and begin analysis of the system.
- Learn the process for examining and manipulating compressed images, such as split E01s, using the SIFT workstation.

Exercise Preparation

1. *Start your SIFT VMware Workstation in VMware Workstation.*
2. *Login the VMware machine.*
   - *LOGIN =* **`sansforensics`**
   - *PASSWORD =* **`forensics`**

Exercise – Step-by-Step Guide

1. Elevate your privileges to root.

```
$ sudo su -
```

2. Change directories to the **`/cases/xp-tdungan-c-drive/`**. When you run the "ls" command below, take note of how many and the sizes of the "E01, E02, E03" files in the directory.

```
# cd /cases/xp-tdungan-c-drive/

# ls -lh
```

```
-rwxrwxrwx 1 sansforensics sansforensics 2.0G Apr  9  2012 xp-tdungan-c-drive.E01
-rwxrwxrwx 1 sansforensics sansforensics 2.0G Apr  9  2012 xp-tdungan-c-drive.E02
-rwxrwxrwx 1 sansforensics sansforensics 2.0G Apr  9  2012 xp-tdungan-c-drive.E03
-rwxrwxrwx 1 sansforensics sansforensics 703M Apr  9  2012 xp-tdungan-c-drive.E04
-rwxrwxrwx 1 sansforensics sansforensics 1.7K Apr  9  2012 xp-tdungan-c-drive.txt
```

3. Mount the **`xp-tdungan-c-drive.E01`** image files in the **`/mnt/ewf_mount`** directory. Note the result of this command will combine the compressed expert witness images into a single raw image. The **`ewfmount`** command will always name this file "ewf1." The **`ewfmount`** command will combine the (E01, E02, E03, etc.) files together and uncompress them via the running mount process. When you run the "ls" command below, take note of the file size of the uncompressed raw image.

E01 to Raw

Mount E01 using `ewfmount`

- `# ewfmount image-name.E01 /mnt/ewf/`
- `/mnt/ewf/` Directory will now contain a raw dd/raw image

```
# ewfmount xp-tdungan-c-drive.E01 /mnt/ewf_mount/

# cd /mnt/ewf_mount

# ls -lh
```

```
root@siftworkstation:/# cd /cases/xp-tdungan-c-drive/
root@siftworkstation:/cases/xp-tdungan-c-drive# ewfmount xp-tdungan-c-drive.E01 /mnt/ewf_mount
ewfmount 20140608
root@siftworkstation:/cases/xp-tdungan-c-drive# cd /mnt/ewf_mount
root@siftworkstation:/mnt/ewf_mount# ls -lh
total 0
-r--r--r-- 1 root root 16G Oct 28 20:48 ewf1
```

4. Mount the xp-tdungan-c-drive raw image found in the `/mnt/ewf_mount` directory on the `/mnt/windows_mount` directory.

`mount -o[options] device directory`

[Useful Options]

`-o`	Use the following options separated by commas -- no spaces
`ro,`	mount as read only
`loop,`	mount on a loop device (used for image files)
`noexec,`	do not execute files from mounted partitions
`offset=byte_num,`	if mounting a physical drive you can use the offset option to have it mount at `x bytes` into the physical disk
`show_sys_files,`	show NTFS volume metafiles
`streams_interface=windows`	use alternate data streams

Figure 1 Mount Command Options and Explanations

```
# mount -o ro,loop,show_sys_files,streams_interface=windows ewf1
/mnt/windows_mount/

# cd /mnt/windows_mount

# ls
```

Ewf
↓ ewfmount
Disk image
↓ mount
Linux

```
root@SIFT-Workstation:/mnt/windows_mount# ls
$AttrDef        Documents and Settings  MSOCache            System Volume Information
AUTOEXEC.BAT    $Extend                 NTDETECT.COM        Temp
$BadClus        hiberfil.sys            ntldr               $UpCase
$Bitmap         IO.SYS                  pagefile.sys        $Volume
$Boot           $LogFile                [redacted]          [redacted]
boot.ini        $MFTMirr                [redacted]
CONFIG.SYS      MSDOS.SYS               $Secure
```

This page intentionally left blank.

Exercise 3 – CTI - Indicator Creation and Examination

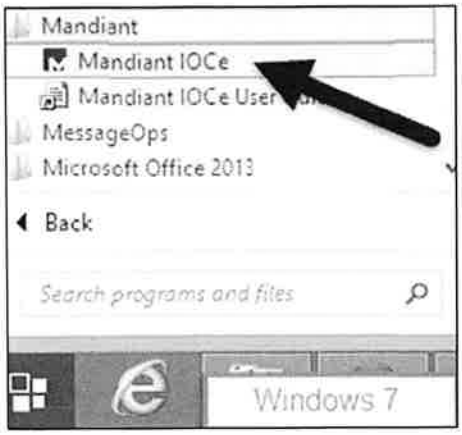
Objectives

- Install and Launch IOC Editor on your Windows Workstation
- Examine IOC Editor's interface and examine default APT1 "Comment Crew" indicators
- Build your own IOC based on the information provided by the Government Agency that contacted you
- Use Boolean logic and different operators to ensure your IOC is properly configured

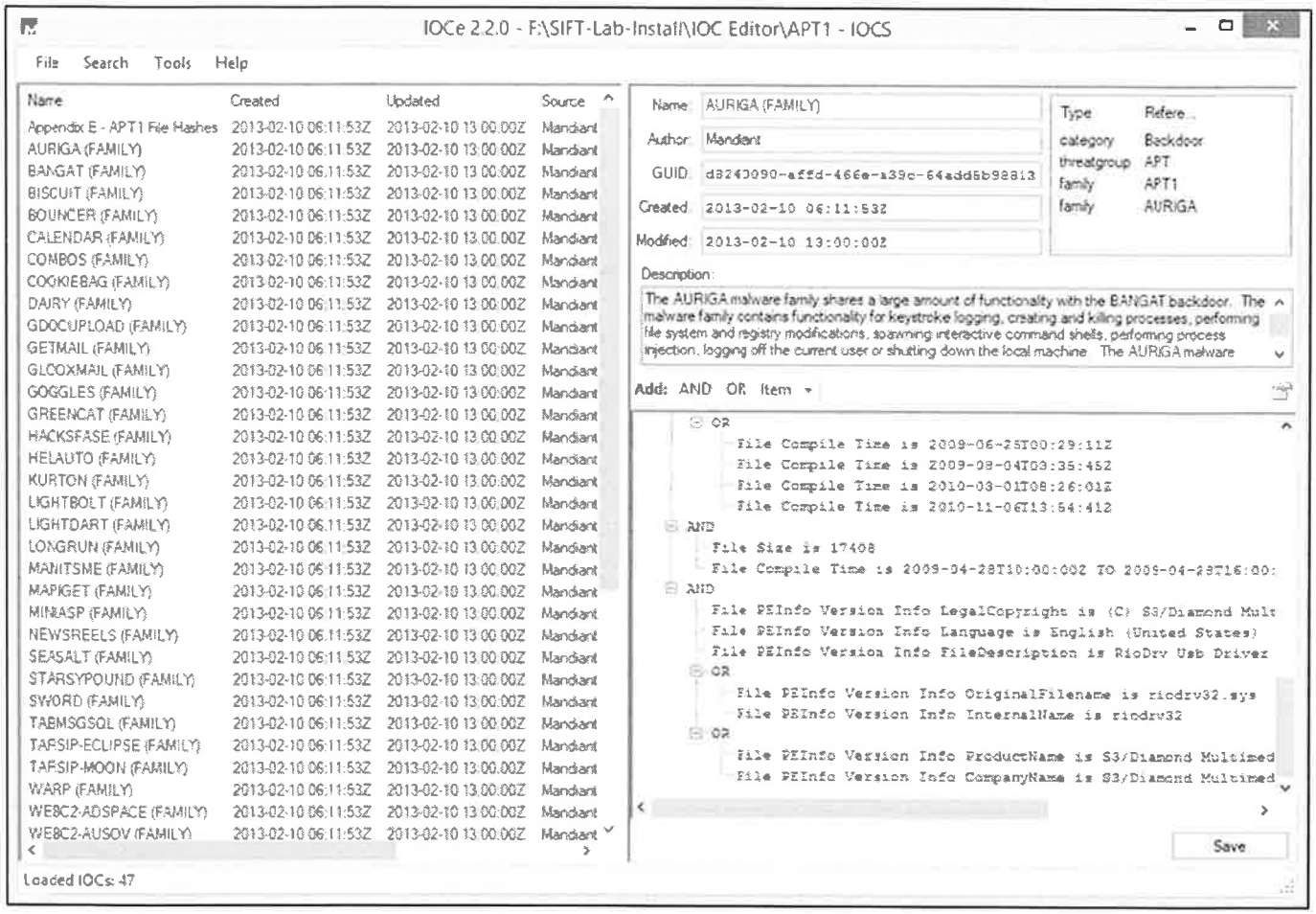
Exercise Preparation

➢ *Install Mandiant IOC Editor from your USB on your Windows Host (or VM) in the* **`\SIFT-Lab-Install\IOC Editor`** *folder*

➢ *Launch Mandiant IOC Editor (Win 7 and Win 8 shown below)*

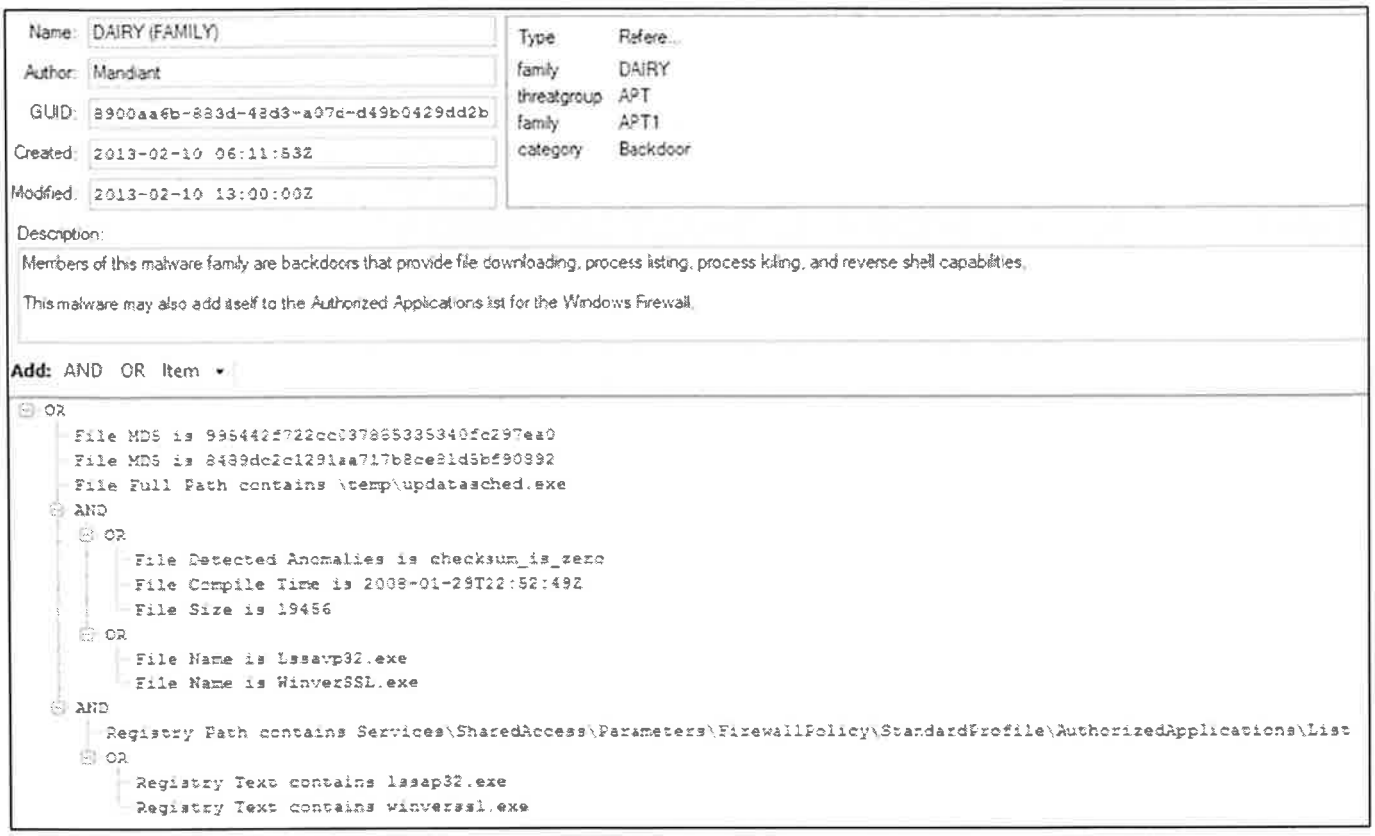


➢ *Open "IOC Directory"* **`\SIFT-Lab-Install\IOC Editor\APT1 – IOCS\`**



Exercise – IOC Familiarization

1. Spend about 5 minutes browsing the various APT1 indicators and the IOCs that have been created already. The main objective is to identify a specific "level of detail" and familiarization with the Boolean expressions of the various malware families
2. Select DAIRY (FAMILY) of malware and step through the IOC ruleset in the lower right hand corner



3. Detail the Boolean Expression IOC language by stepping through the logic. Again all we would like you to do here is mainly become familiar with the language.

Exercise – Create an IOC based on Threat Intelligence provided by Government Agency

As a part of the notification, the government agency that notified you also gave you a basic indicator of the APT group HYDRA that have compromised other locations. Look at the provided information and create an IOC for the indicator

- **Family –HTTPPUMP**
- **Threatgroup – APT**
- **Family – HYDRA**
- **Category - Backdoor**

Note this information is found in a textile (so you can copy the text) called **`HTTPPUMP.txt`** *in the* **`\SIFT-Lab-Install\IOC Editor folder`**

Description: Members of this family are full featured backdoors that communicates with a Web-based Command & Control (C2) server over HTTP port 80 using XMLRPC over HTTP for encoding. Features include interactive shell, gathering system info, uploading and downloading files, and creating and killing processes, Malware in this family usually communicates with a hard-coded domain using XMLRPC over HTTP on port 80.

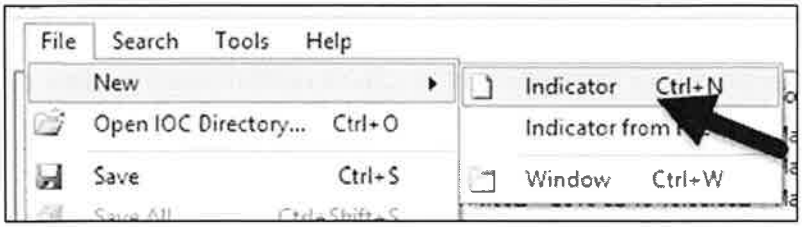
Some members of this family rely on launchers to establish persistence mechanism for them. Several variants use %USER%\Local Settings\Temp or %USER%\AppData\Local\Temp as working directories, additional malware artifacts may be found there.

IOC Provided --

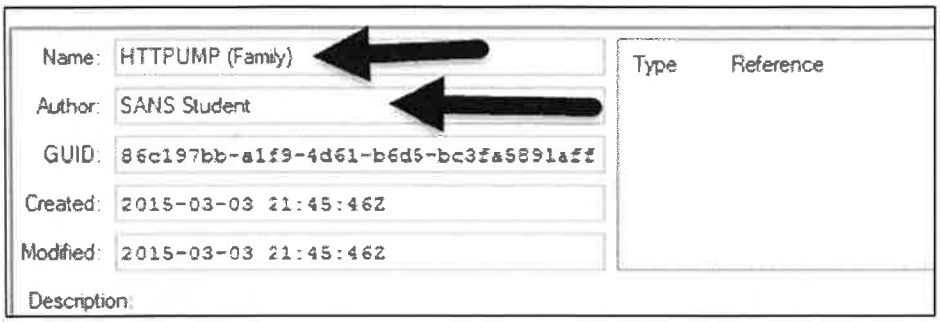
- (MD5 = **c4b0458c04abdaa773348c2668212b45** or Filename = **a.exe**, **b.exe**, **c.exe**)
    - AND
        - (Compile Time = **2011-10-13 04:19:53** or **2011-10-19 02:39:12**)
        - File Size = **9216** or **9245**
        - AND
            - Directory location = **\Temp**
            - Strings inside malware = **httppump**
            - **\CurrentControlSet\Services\Netman\domain**

Exercise – Build the IOC Template Family

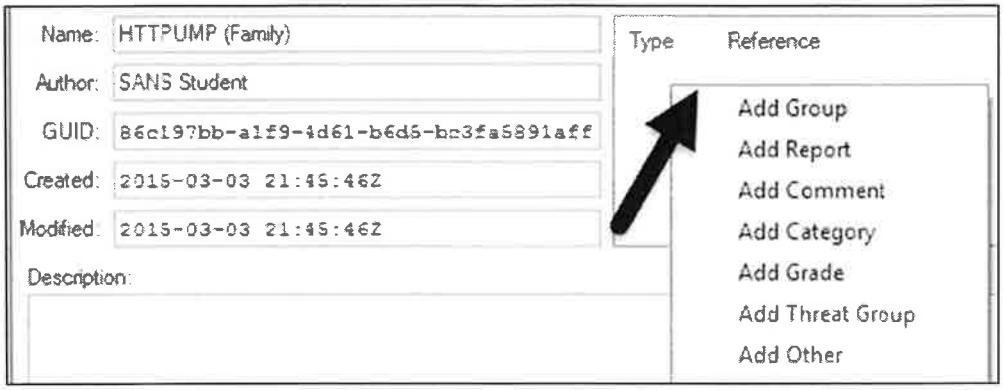
1. Select File -> New -> Indicator



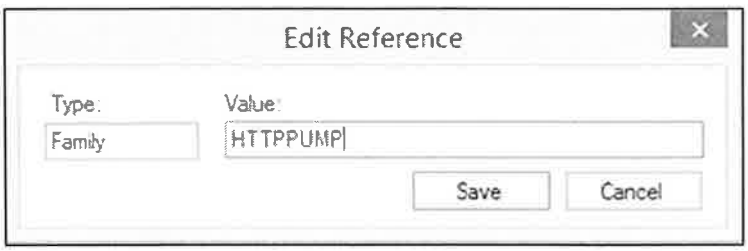
2. Set Name = **`HTTPPUMP (Family)`**
3. Set Author = **`SANS Student`**



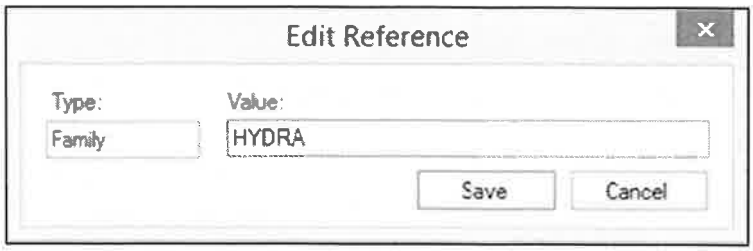
4. Right Click Area to the Right of the Name and Author Location (It should say "**`Type/Reference`**")



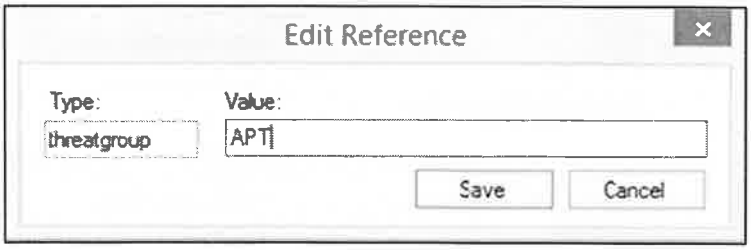
5. Right Click -> Add Other -> Type = **`Family`**, Value = **`HTTPPUMP`**



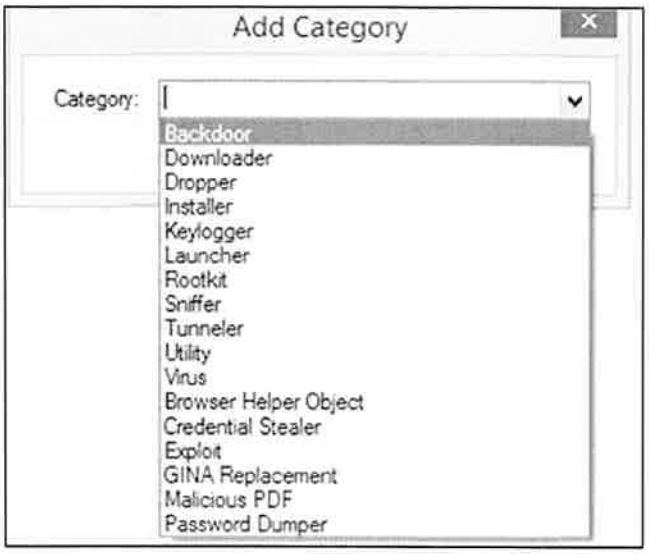
6. Right Click -> Add Other -> Type = **`Family`**, Value = **`HYDRA`**



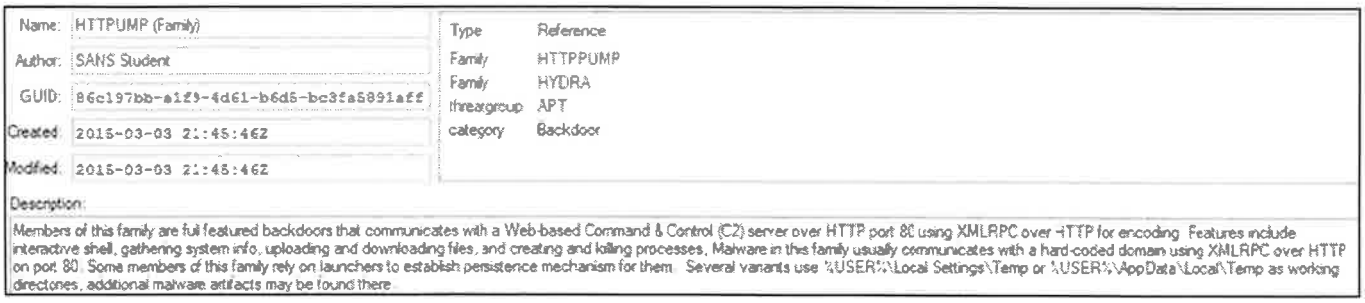
7. Right Click -> Add Threat Group -> Type "**APT**"



8. Right-Click -> Add Category -> **Backdoor**

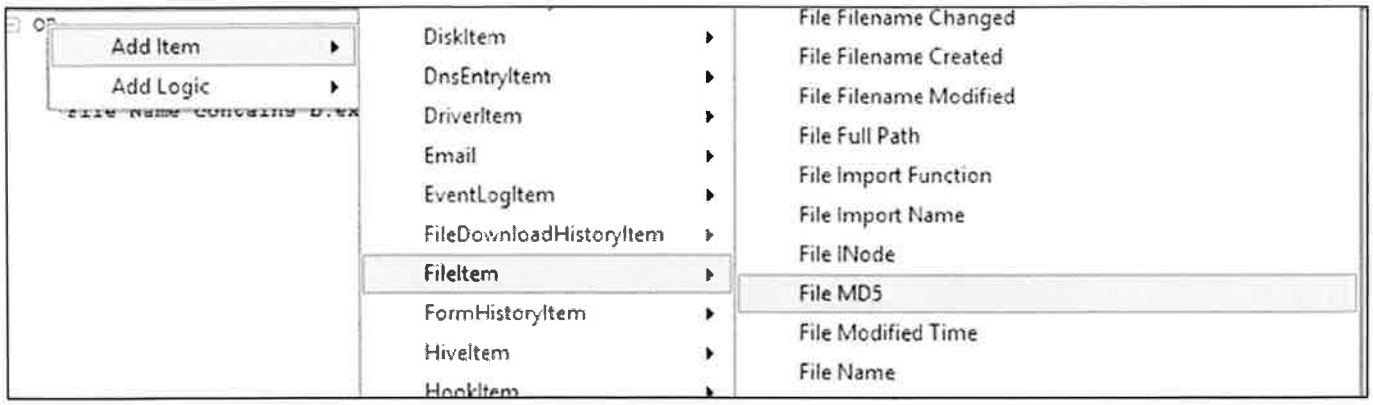


9. Copy and paste the description for the malware from the textfile on your USB.

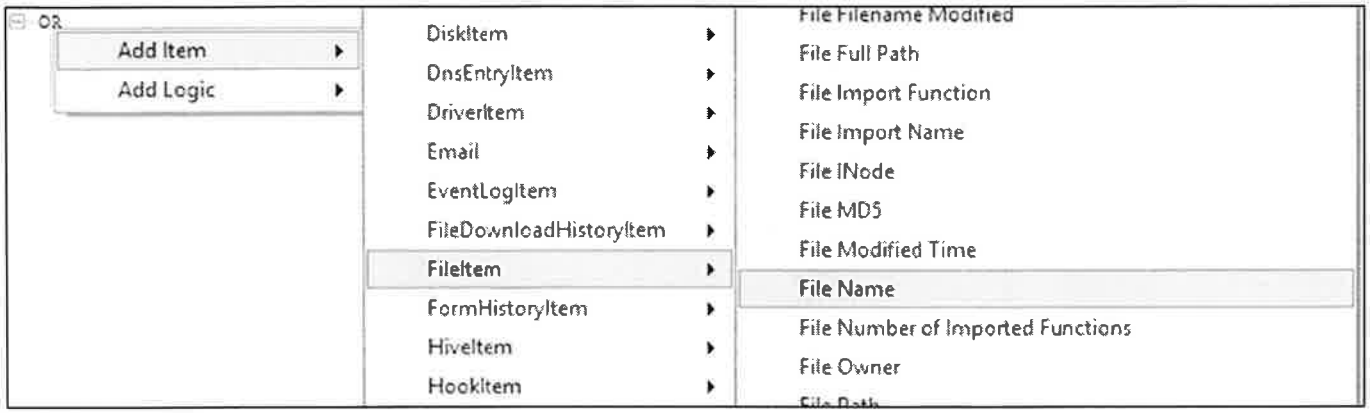


Exercise – Build the IOC using the provided information

1. Right Click on 1st "OR" -> Add Item -> File Item -> File MD5 -> Type "**c4b0458c04abdaa773348c2668212b45**"



2. Right Click on 1st "OR" -> Add Item -> File Item -> File Name -> Type "**a.exe**"



3. Right Click on 1st "**OR**" -> Add Item -> File Item -> File Name -> Type "**b.exe**"
4. Right Click on 1st "**OR**" -> Add Item -> File Item -> File Name -> Type "**c.exe**"

5. **CHECK YOUR WORK** -> YOUR IOC SHOULD NOW LOOK LIKE THIS

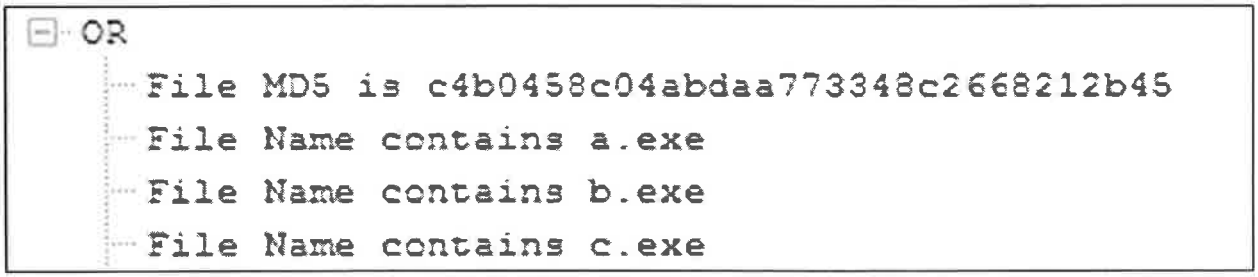
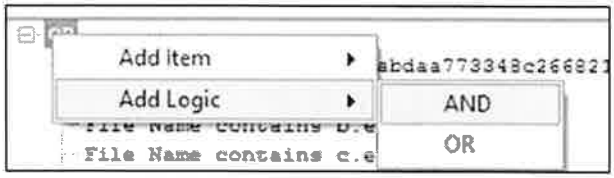
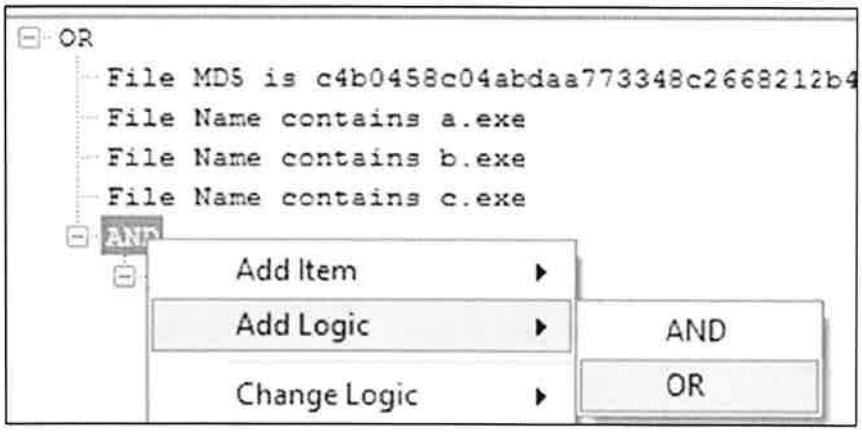


Figure 1 Current IOC at End of Step 4

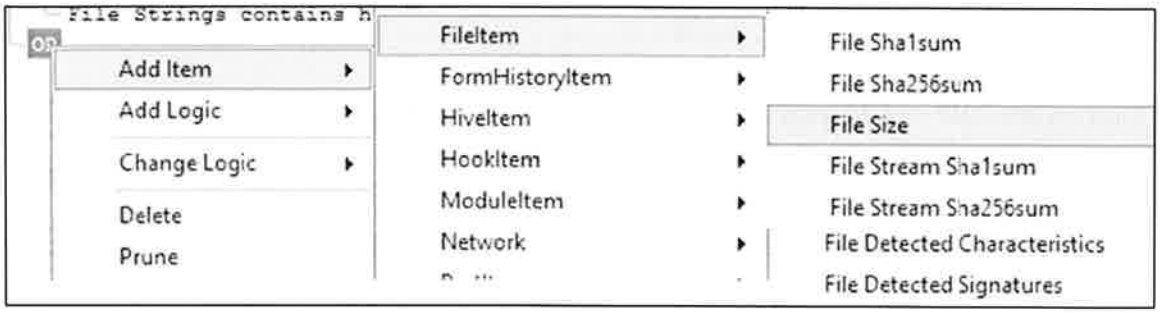
6. Right Click on "**OR**" -> Add Logic -> Select "**AND**"



7. Right Click on 1st "**AND**" -> Add Logic -> Select "**OR**"



8. Right Click on 2nd "**OR**" -> Add Item -> File Item -> File Size -> Type "**9216**"



9. Right Click on 2nd "**OR**" -> Add Item -> File Item -> File Size -> Type "**9245**"
10. **CHECK YOUR WORK** -> YOUR IOC SHOULD NOW LOOK LIKE THIS

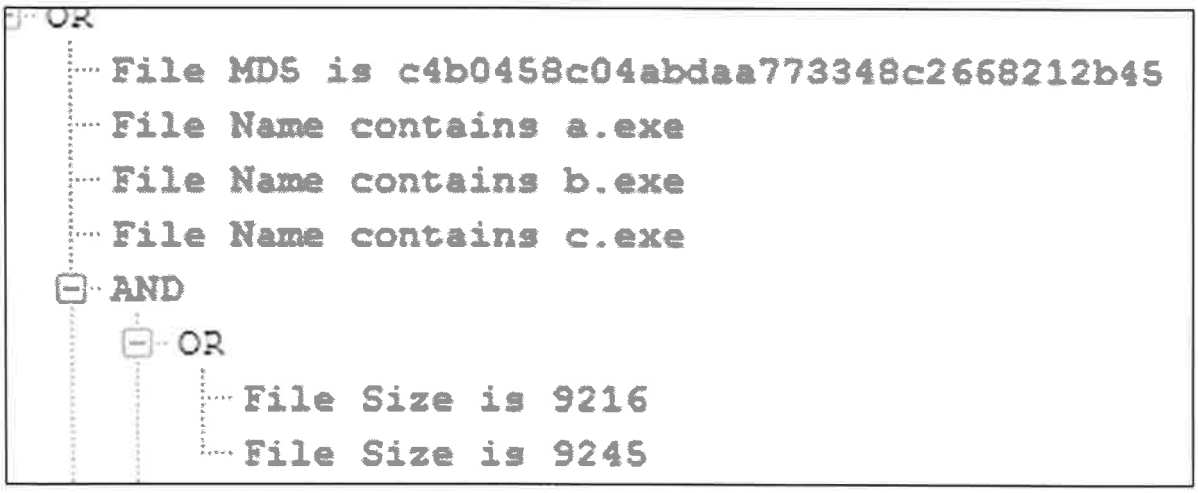
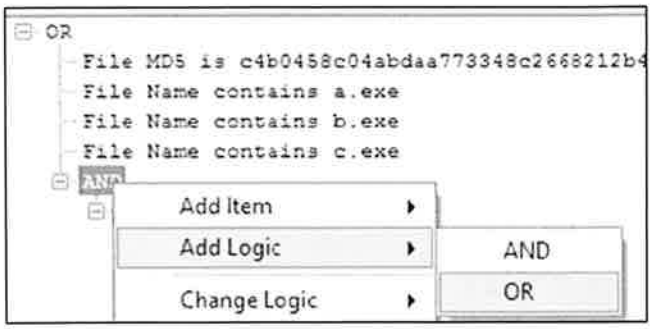
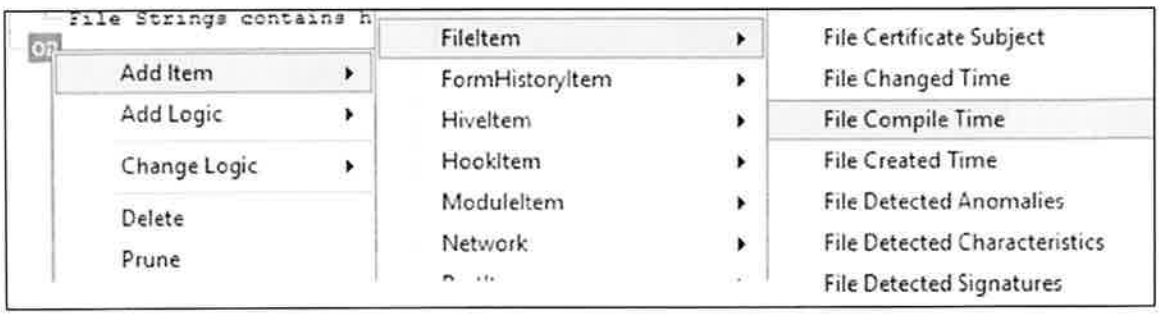


Figure 2 Current IOC at End of Step 9

11. Right Click on 1st "**AND**" -> Add Logic -> Select "**OR**"



12. Right Click on 3rd "**OR**" -> Add Item -> File Item -> File Compile Time -> Type "**2011-10-13T04:19:53Z**"



13. Right Click on 3rd "OR" -> Add Item -> File Item -> File Compile Time -> Type "**2011-10-19T02:39:12Z**"

14. **CHECK YOUR WORK** -> YOUR IOC SHOULD NOW LOOK LIKE THIS

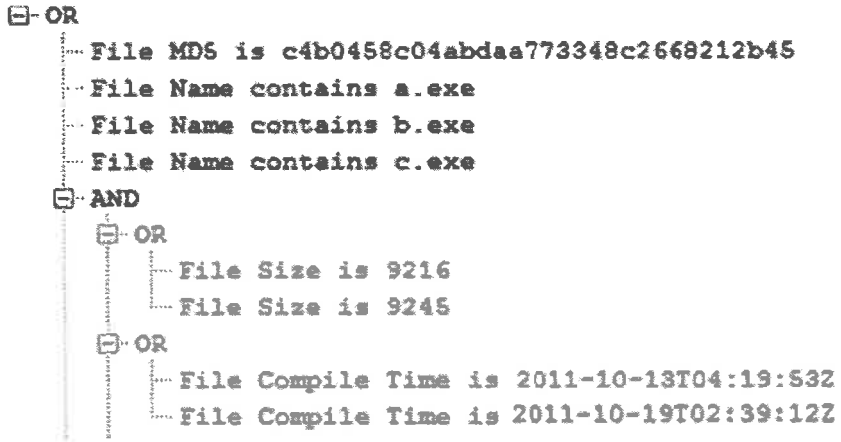
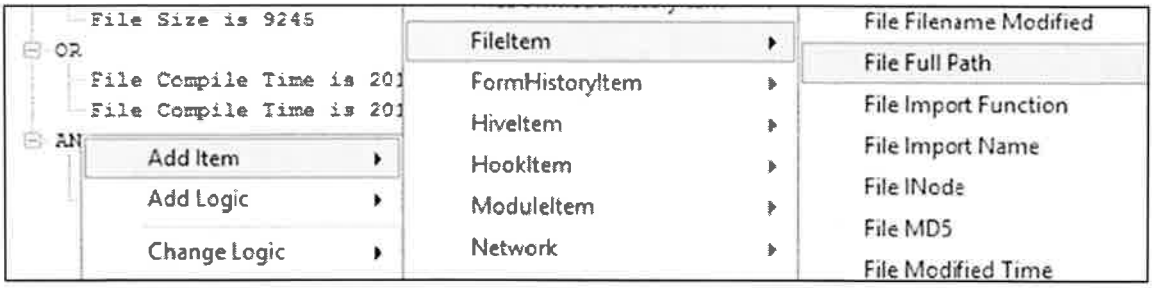


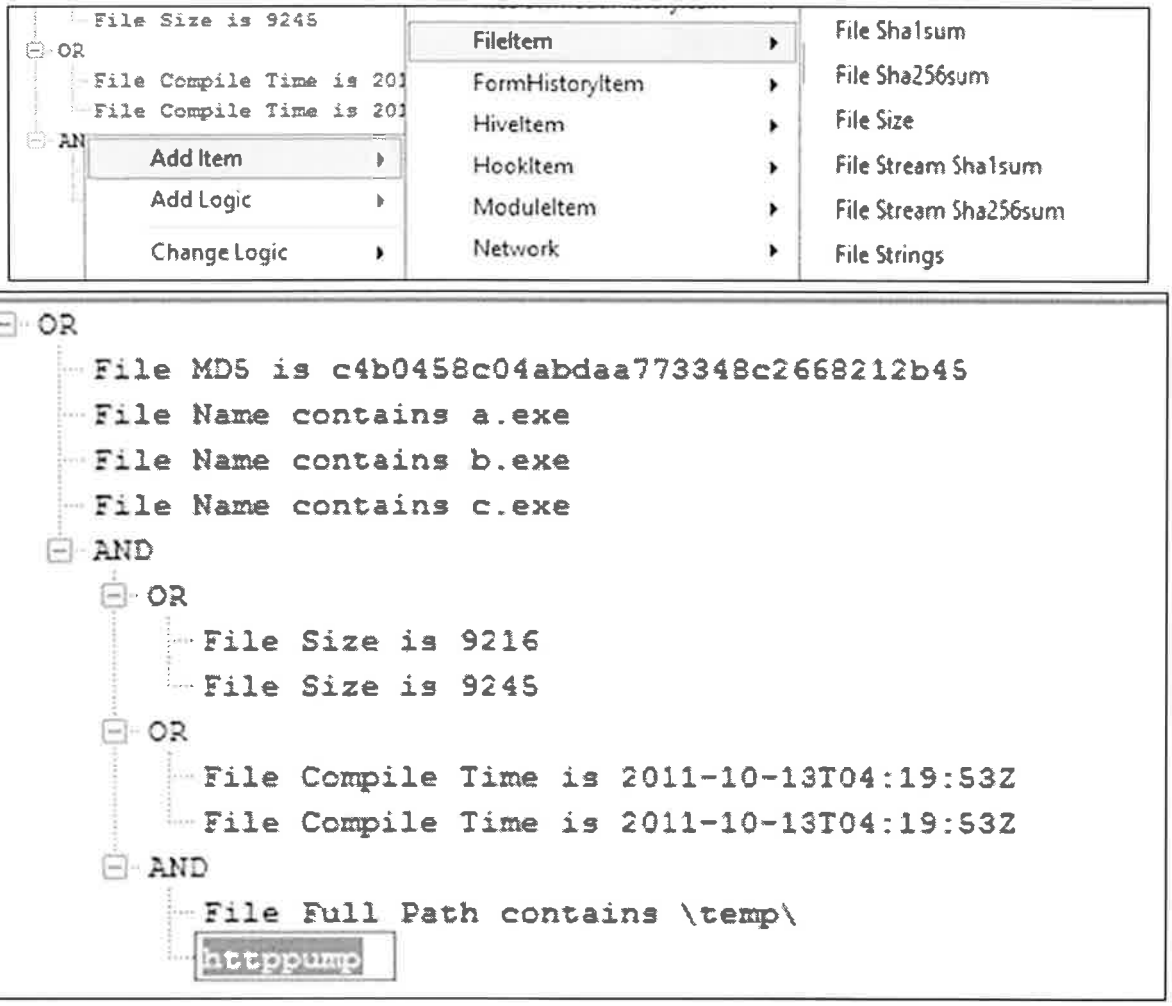
Figure 3 Current IOC at End of Step 13

15. Right Click on 1st "**AND**" -> Add Logic -> Select "**AND**"

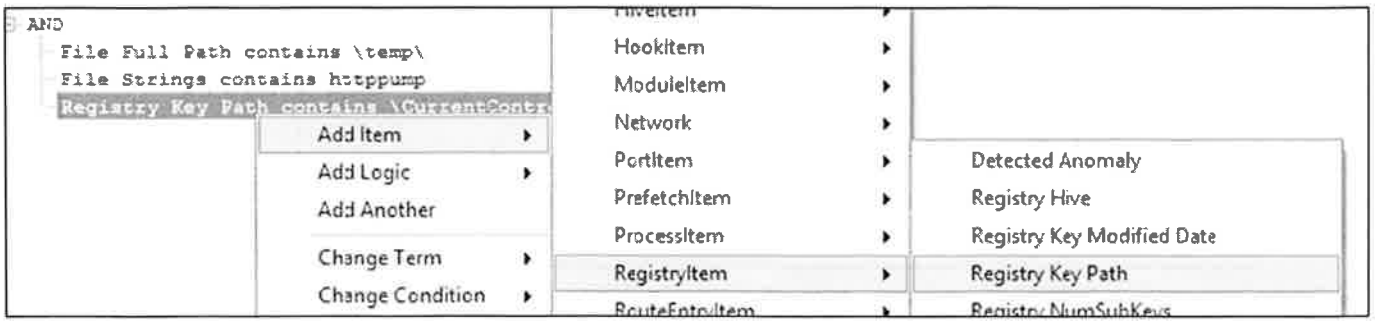
16. Right Click on 2nd "AND" -> Add Item -> File Item -> File Full Path -> Type "\temp\"



17. Right Click on 2nd "AND" -> Add Item -> File Item -> File Strings -> Type "httppump"



18. Right Click on 2nd "AND" -> Add Item -> RegistryItem -> Registry Key Path -> Type "**\CurrentControlSet\Services\Netman\domain**"



19. Congrats!! You are now complete with your first IOC that you have added.
It should look like this in the end.

```
OR
    File MD5 is c4b0458c04abdaa773348c2668212b45
    File Name contains a.exe
    File Name contains b.exe
    File Name contains c.exe
  AND
    OR
        File Size is 9216
        File Size is 9245
    OR
        File Compile Time is 2011-10-13T04:19:53Z
        File Compile Time is 2011-10-13T04:19:53Z
    AND
        File Full Path contains \temp\
        File Strings contains httppump
        Registry Key Path contains \CurrentControlSet\Services\Netman\domain
```

Figure 4 Current IOC at End of Step 11

Exercise Takeaways

Building a good IOC is sometimes very tedious, but an analyst that is armed with a decent IOC list is extremely powerful. There are many indicator languages out there including STIX, YARA, and OpenIOC. Threat indicators and management is extremely powerful during incident response.

Both Redline and Volatility have OpenIOC support built into their capabilities. And using capabilities such as the Live Analysis with Redline, it is feasible to scan a single target with both host and process based indicators.

In the end, understanding how an indicator can be built and how it can be used to look for malware variants is important for incident response and hunting in the world today. The best information your malware analysis team can provide is a decent threat indicator for the host and the network scanning teams. Learning the basics of how to build a single signature is useful.

The more you discover and the more you can add will help make your threat indicator scanning much more useful and practical.

Exercise 4A – Conference/Onsite Version

Description

This version of the lab is intended to demonstrate the capabilities of remote forensic and incident response analysis by connecting to a system that the instructor has set up. This typically would be in classes where we are live and have limited access to additional systems for this exercise.

When you get back to your own lab, work/home network, or if you are attending the class via On-Demand, vLive, or Simulcast remotely, please accomplish Exercise 4B.

In Exercise 4B, you will not only configure the Enterprise License server, you will also create your own agent, deploy it in your network, and connect to another host in your environment. Not only is it a bit more complex, it usually takes about twice as long to complete. In a classroom environment, we will demonstrate remote forensics by having the students connect to a preexisting deployed agent scenario to demonstrate the features of remote incident response.

If attending in a live conference or onsite course, the instructor will set up a demonstration system (or two) to have you connect to. Please obtain that IP address from the instructor.

If you are an On-Demand, vLive, or Simulcast student, please accomplish Exercise 4B.

Exercise Preparation

Receive any IP Addresses from your instructor that you can connect to:

IP ADDRESS: _____

Objectives

- Connect/Disconnect to remote system hard drives and memory using the SIFT workstation

Exercise – Install and Configure F-Response Enterprise

1. *Start your SIFT VMware Workstation in VMware*

2. *Login the SIFT VMware machine.*
   - *LOGIN* = `sansforensics`
   - *PASSWORD* = `forensics`

3. **IMPORTANT!!** Open a terminal inside the SIFT workstation and make sure you are a root user by running the command `sudo su -`

```
$ sudo su -
```

Exercise – USE SIFT Workstation to connect to remote system(s) you have deployed the agent to and started the service

1. Switch to your SIFT Workstation

2. Login to see the available nodes (note in this EXAMPLE, we are using the server IP which on my system is 192.168.112.1, but on your system, this should match the IP address that your instructor provided to you.) **`IP_ADDRESS` = Instructor Provided IP address**

```
$ sudo su -

# f-response-accel-lin -n sansforensics -p forensics1234 -s IP_ADDRESS
```

```
root@siftworkstation:/# f-response-accel-lin -n sansforensics -p forensics1234 -s 192.168.112.1
F-Response Acclerator - Linux Version 5.0.3
F-Response Acclerator for Linux requires Open-iSCSI.
Checking for Open-iSCSI utils now..
Open-iSCSI (iscsiadm) found.
Connecting to F-Response Target 192.168.112.1:3260...
Discovery Results.
F-Response Target = iqn.2008-02.com.f-response.jotumheim:disk-0
F-Response Target = iqn.2008-02.com.f-response.jotumheim:vol-c
F-Response Target = iqn.2008-02.com.f-response.jotumheim:pmem
Populating Open-iSCSI with node details..
Node information complete, adding authentication details.
```

3. Login to Remote Hard Drive Target - Usually Disk-0. Note in the above example, my system name was **`jotumheim`**. The exact name of the system you are logging into will change for each system you are logging into. Please note that it is a lowercase "**`L`**" in the **`f-response-accel-lin`** command. ?????? = instructor's system name – this will likely change from class to class. It is generally best practice to "copy and paste" the name of the system into the command below.

```
# f-response-accel-lin -l iqn.2008-02.com.f-response.??????:disk-0
```

```
root@siftworkstation:/# f-response-accel-lin -l iqn.2008-02.com.f-response.jotumheim:disk-0
F-Response Acclerator - Linux Version 5.0.3
F-Response Acclerator for Linux requires Open-iSCSI.
Logging in to [iface: default, target: iqn.2008-02.com.f-response.jotumheim:disk-0, portal: 192.168.112.1,3260]
Login to [iface: default, target: iqn.2008-02.com.f-response.jotumheim:disk-0, portal: 192.168.112.1,3260]: successful
IQN:iqn.2008-02.com.f-response.jotumheim:disk-0 attached as /dev/sdc
```

4. Run **fdisk -l** and examine output locating new drive that is attached at /dev/sdc

```
# fdisk -l
```

```
Disk /dev/sdc: 214.7 GB, 214748364800 bytes
255 heads, 63 sectors/track, 26108 cylinders, total 419430400 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x47856f08

   Device Boot      Start         End      Blocks   Id  System
/dev/sdc1   *        2048   419428351   209713152    7  HPFS/NTFS/exFAT
```

5. Mount larger partition via the SIFT Workstation using the mount command. (Note: **/dev/sdc2** is what it is on the example, your actual results will vary depending on your system). You can determine the largest partition by examining the partition with the largest number of blocks. Also, usually on Win7 and later systems the 2nd partition is the C:\ of the system

```
# imageMounter.py /dev/sdc /mnt/windows_mount2
```

```
root@siftworkstation:/home/sansforensics# imageMounter.py /dev/sdc /mnt/windows_
mount2/
[+] Creating Temp Mount Point at /mnt/windows_mount2/0
[+] Attempting to Mount Partition 0 at /mnt/windows_mount2/0
   [-] Mounted /dev/sdc at /mnt/windows_mount2/0
   [-] To unmount run 'sudo umount /mnt/windows_mount2/0'
```

6. Change Directories to the **/mnt/windows_mount2** directory and examine files. You should now be able to see the files from the remote system with the F-Response Agent installed. Below is a generic example.

```
# cd /mnt/windows_mount2/0

# ls
```

```
root@siftworkstation:/home/sansforensics# cd /mnt/windows_mount2/0/
root@siftworkstation:/mnt/windows_mount2/0# ls
$AttrDef                    IEFCrashLog.txt             Python32
$BadClus                    $LogFile                    Recovery
$Bitmap                     $MFTMirr                    $Recycle.Bin
Boot                        msdia80.dll                 $Secure
$Boot                       pagefile.sys                swapfile.sys
bootmgr                     PerfLogs                    System Volume Information
BOOTNXT                     Perl64                      $UpCase
BOOTSECT.BAK                pgData91                    Users
cases                       ProgramData                 $Volume
Documents and Settings      Program Files               Windows
$Extend                     Program Files (x86)
Forensic Program Files      Python27
```

7. Change Directories to the **Windows/System32/config directory and extract the registry contents of the SAM and SYSTEM hives. Please note that due to "lower/upper case conversions" your exact directory path might vary on your own system. You might need to use "tab complete" to help you complete the directory path.**

```
# cd /mnt/windows_mount2/0

# cd Windows/System32/config

# rip.pl -r SAM -f sam > /cases/sam.txt

# less /cases/sam.txt
```

```
root@siftworkstation:/mnt/windows_mount2/1# cd Windows/System32/config/
root@siftworkstation:/mnt/windows_mount2/1/Windows/System32/config# rip.pl -r SAM -f sam > /cases
/sam.txt
Parsed Plugins file.
Launching samparse v.20120722
samparse complete.
root@siftworkstation:/mnt/windows_mount2/1/Windows/System32/config# less sam.txt
```

```
# rip.pl -r SYSTEM -f system > /cases/system.txt

# less /cases/system.txt
```

iscsiadm -m node

Exercise – Attach Remote Memory and Perform Memory Acquisition

1. Login to Appropriate Target - Usually PMEM. Please note that it is a lowercase "**L**" in the **f-response-accel-lin** command. ?????? = your target system name.

Example (**Memory**) = **iqn.2008-02.com.f-response.?????:pmem**

```
# f-response-accel-lin -l iqn.2008-02.com.f-response.?????????:pmem
```

```
root@siftworkstation:/# f-response-accel-lin -l iqn.2008-02.com.f-response.jotunheim:pmem
F-Response Acclerator - Linux Version 5.0.3
F-Response Acclerator for Linux requires Open-iSCSI.
Logging in to [iface: default, target: iqn.2008-02.con.f-response.jotunheim:pmem, portal: 192.168.112.1,3260]
Login to [iface: default, target: iqn.2008-02.com.f-response.jotunheim:pmem, portal: 192.168.112.1,3260]: successful
IQN:iqn.2008-02.com.f-response.jotunheim:pmem attached as /dev/sdd
```

2. Run **fdisk -l** and examine output locating the memory image.
   a. It should not show a valid partition table at **/dev/sdd**

```
# fdisk -l
```

```
Disk /dev/sdd: 9636 MB, 9636413440 bytes
64 heads, 32 sectors/track, 1148 cylinders, total 2352640 sectors
Units = sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disk identifier: 0x00000000

Disk /dev/sdd doesn't contain a valid partition table
```

In the next section we will learn how to use Volatility and Redline during memory analysis.

Exercise – Detach Remote F-Response Targets

1. Detach the remote target system memory and hard drives.

```
# f-response-accel-lin -u iqn.2008-02.com.f-response.?????????:pmem
# cd /
# umount /mnt/windows_mount2/0
# f-response-accel-lin -u iqn.2008-02.com.f-response.?????????:disk-0
```

Exercise Takeaways

Being able to connect to 100s of system in your environment to perform incident response is critical. Using a deployable agent via group policy will allow you to connect to almost any IP address in your environment.

The next step is analysis and then making it automated -- using a script to connect and pull specific data from 100s of systems in your enterprise.

The days of pulling a hard drive each time you want to analyze a system are over. The time to be able to perform remote analysis is here and using tools such as F-Response, Mandiant MIR, Encase Enterprise, and many others gives the ability to scale from a single system to thousands.

This page intentionally left blank.

Exercise 4B – Enterprise Forensics (OnDemand/vLive/Simulcast) Version

Description

This version of the lab is intended to demonstrate the capabilities of remote forensic and incident response analysis by connecting to a system that the instructor and faculty has set up. This typically would be in classes where we are live and have limited access to additional systems for the setup. This exercise will not only show you how to connect to remote systems, but how to set up, configure and deploy F-Response Agents on your own networks.

In this exercise you will not only configure the F-Response-Enterprise License server, you will also create your own agent, deploy it in your network, and connect to another host in your environment.

Please note – this follow on exercise is meant to be completed on your own networks and not in live classroom environments such as SANS conferences or Onsites. To demonstrate F-Response, the instructor will set up a system for students to connect to. If attending in a live conference or onsite course, the instructor will set up a demonstration system (or two) to have you connect to and please follow Exercise 4A. Please obtain that IP address from the instructor.

If you are an On-Demand, vLive, or Simulcast student, please continue Exercise 4B.

PLEASE NOTE – If you are having trouble with the F-Response Enterprise Lab Setup, there is a troubleshooting guide at the end of this exercise.

Objectives

- Install and configure an enterprise incident response and forensics capability
- Create a deployable agent
- Deploy agents on to target systems in your environment
- Connect/Disconnect to remote system hard drives and memory using the SIFT workstation

Exercise – Install and Configure F-Response Enterprise

1. *Start your SIFT VMware Workstation in VMware*

2. *Login the SIFT VMware machine.*
   - *LOGIN =* **`sansforensics`**
   - *PASSWORD =* **`forensics`**

3. Open a terminal inside the SIFT workstation and make sure you are a root user by running the command **`sudo su -`**

```
$ sudo su -
```

4. SIFT WORKSTATION: Check IP Address of SIFT Workstation using "**ifconfig**"

```
$ ifconfig
```

Note you are looking for output similar to the following where the IP address of the example system is **192.168.112.138**.

```
sansforensics@siftworkstation:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 00:0c:29:61:54:df
          inet addr:192.168.112.138  Bcast:192.168.112.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe61:54df/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:838 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2503 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:562499 (562.4 KB)  TX bytes:244357 (244.3 KB)
```

Read the "**inet addr**" of your output and write down the IP Address of your SIFT workstation.

SIFT IP ADDRESS = _____

5. Locate and insert your F-Response Enterprise Dongle into your Windows System

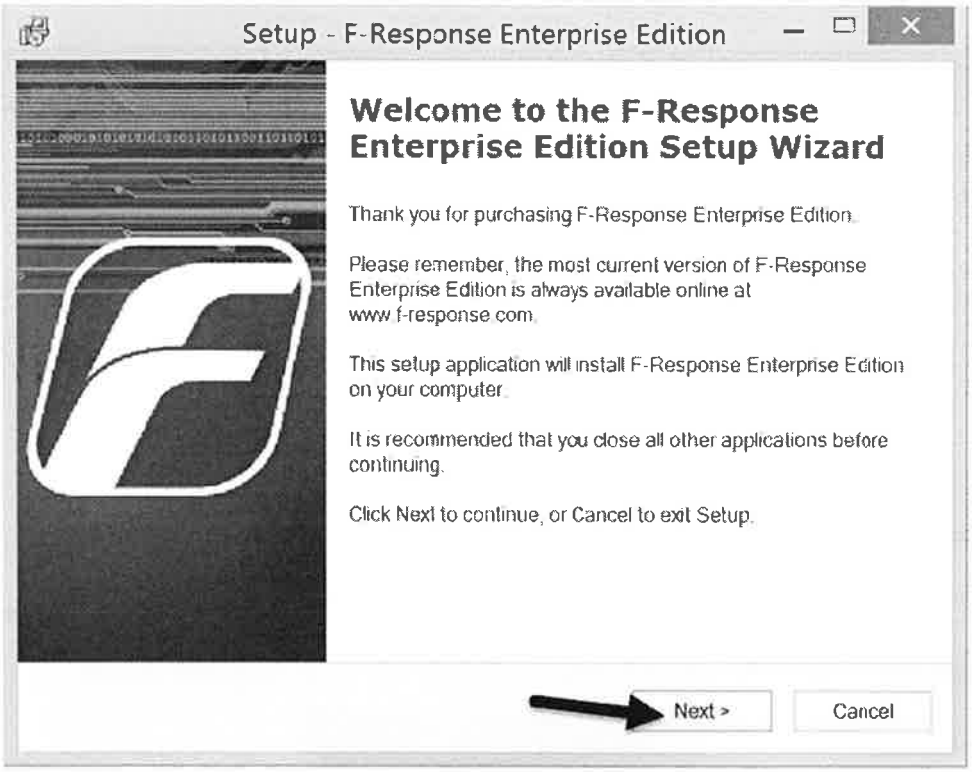
6. **Windows System: Turn off** your windows system firewall or open up ports on your enterprise network to allow for the following ports to communicate over TCP (Control Panel -> System and Security -> Windows Firewall -> Turn Firewall On or Off)

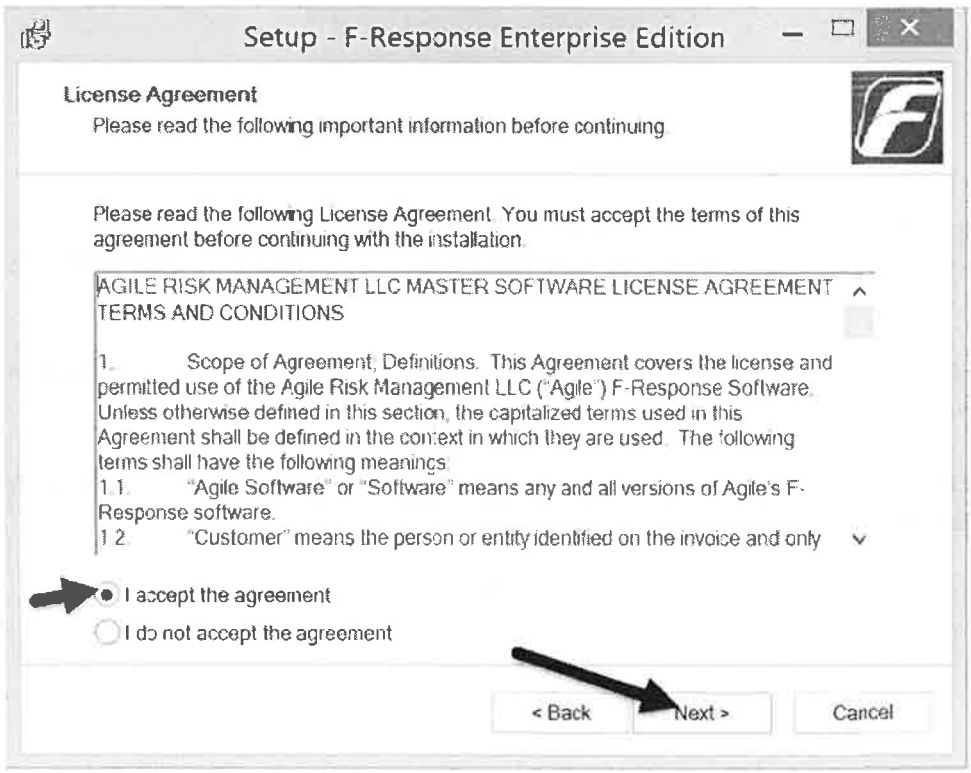
   1. **3260** – This is the main port F-Response uses to communicate with the target machine's resources (iSCSI TCP).

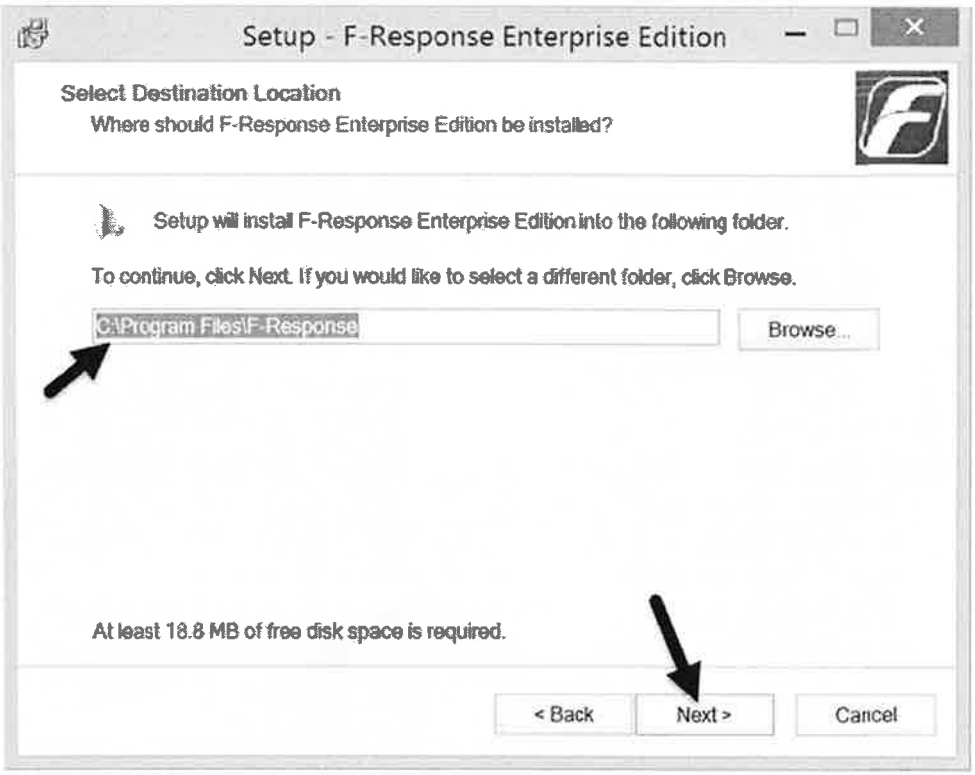
   2. **5681** – This port is used to communicate with the F-Response Licensing Manager. (TCP is used for most editions of F-Response, UDP is used for F-Response Tactical Edition).

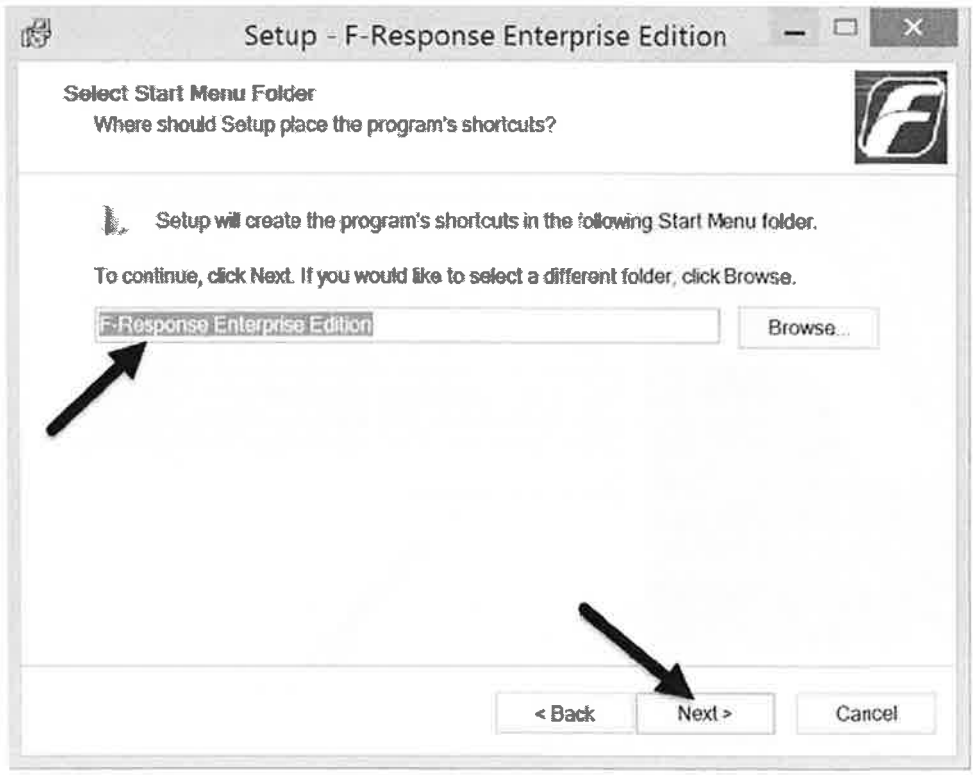
7. Insert your **COURSE USB with DFIR** on the side into your Windows System.
   - Navigate to `\SIFT-Lab-Install\F-Response-Enterprise`

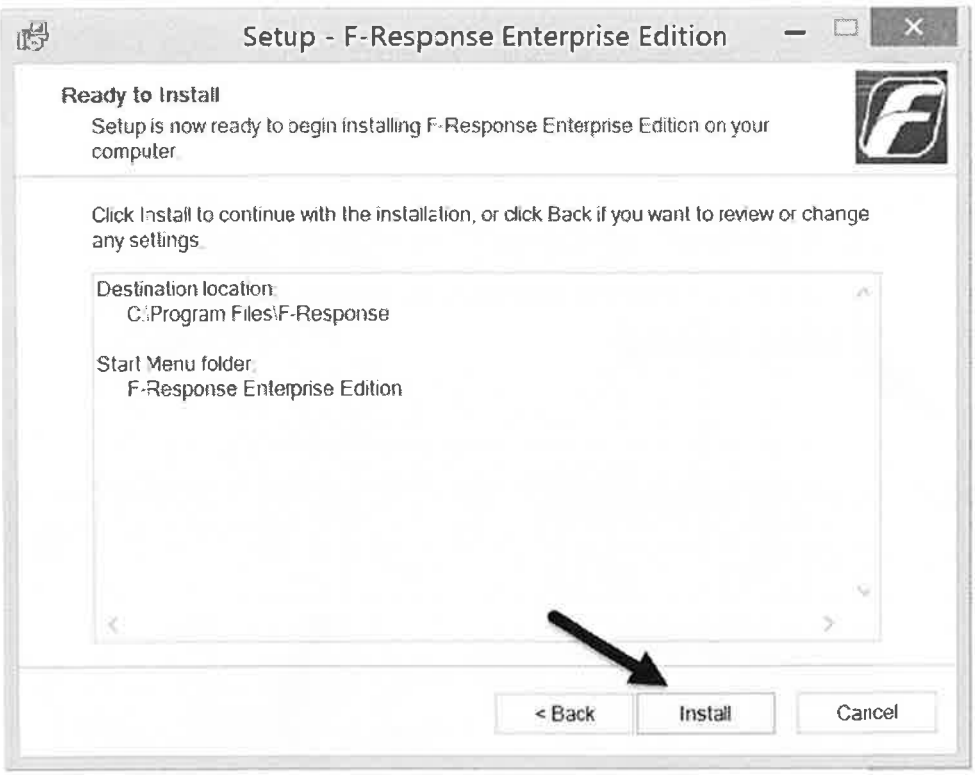
8. Install "**`F-Response Enterprise`**" software (**`F-ResponseEnterprise.exe`**) on your Windows System. All steps require you to simply select "**`Next`**" and Finally "**`Install`**" and "**`Finish`**"






Setup - F-Response Enterprise Edition
Select Destination Location
Where should F-Response Enterprise Edition be installed?
Setup will install F-Response Enterprise Edition into the following folder.
To continue, click Next. If you would like to select a different folder, click Browse.
C:\Program Files\F-Response
Browse...
At least 18.8 MB of free disk space is required.
< Back
Next >
Cancel


Setup - F-Response Enterprise Edition
Select Start Menu Folder
Where should Setup place the program's shortcuts?
Setup will create the program's shortcuts in the following Start Menu folder.
To continue, click Next. If you would like to select a different folder, click Browse.
F-Response Enterprise Edition
Browse...
< Back
Next >
Cancel

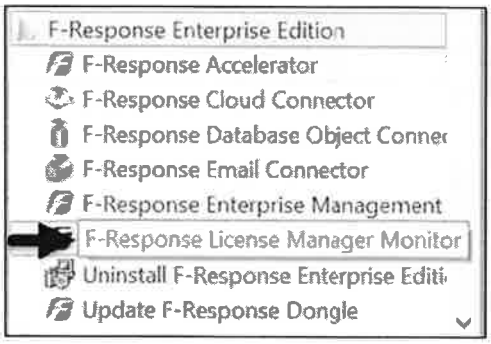

Setup - F-Response Enterprise Edition
Ready to Install
Setup is now ready to begin installing F-Response Enterprise Edition on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Destination location:
C:\Program Files\F-Response
Start Menu folder:
F-Response Enterprise Edition
< Back
Install
Cancel


Setup - F-Response Enterprise Edition
Completing the F-Response Enterprise Edition Setup Wizard
Setup has finished installing F-Response Enterprise Edition on your computer.
Don't forget to insert your F-Response License dongle before installing and starting the F-Response License Manager Server.
In addition, F-Response requires the Microsoft iSCSI Initiator (Windows XP). If you have not already installed the Microsoft iSCSI Initiator and are using Windows XP please use the link below. Windows Vista, 2008, and 7 users already have iSCSI installed and will see no link. U.S. Pat. No. 7,899,822 8,171,108.
Click Finish to exit Setup.
Finish

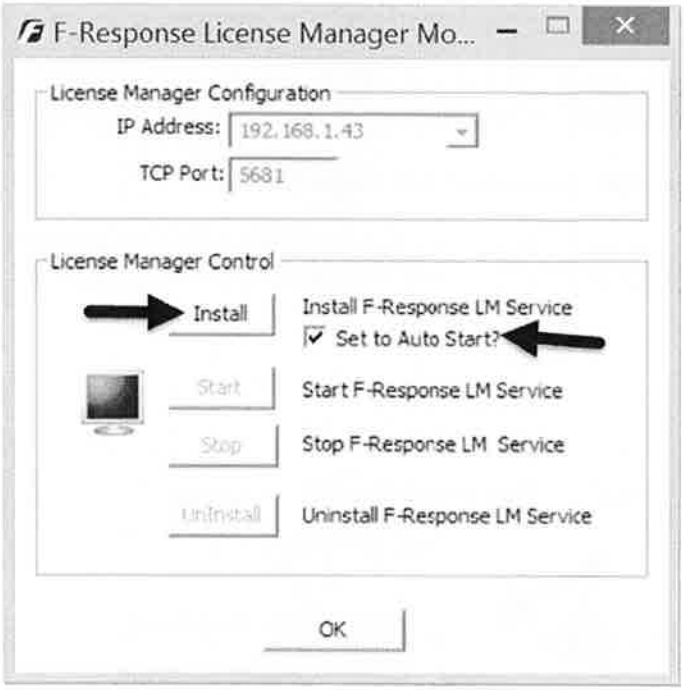
Exercise – Set up the F-Response Enterprise License Manager

1. Insert F-Response Enterprise Dongle into a Windows operating system. The F-Response Enterprise software you just installed allows you to create remote agents. It will also serve as your license manager for remote workstations connecting to other system targets in your environment.

2. Launch **LICENSE MANAGER MONITOR**



3. Install **F-Response LM Service** and check **"`Set to Auto Start`"**

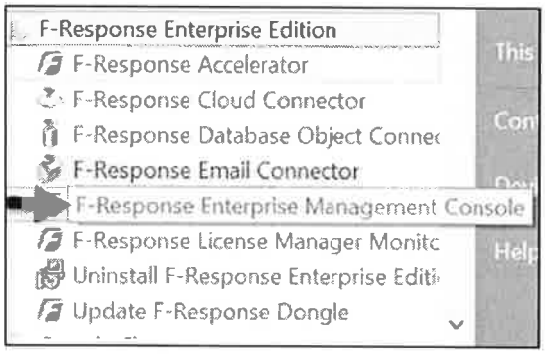


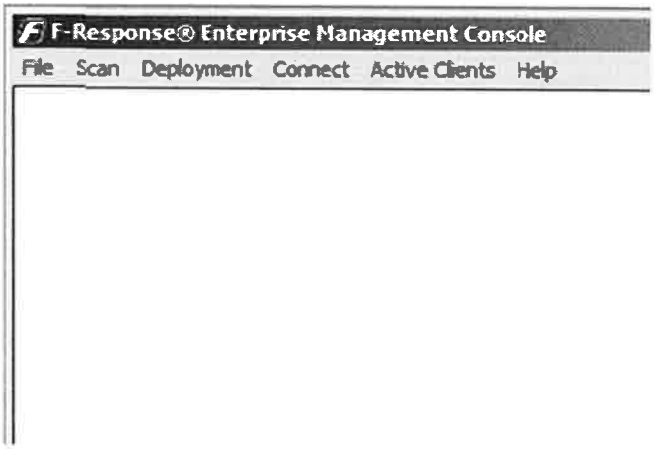
4. From the "IP Address:" dropdown select the IP Address that matches the same subnet of your SIFT Workstation IP Address documented in step #4 during your Exercise Preparation above. NOTE: In the example listed here, our SIFT IP Address was **"192.168.112.138"** thus the IP Address of the License Manager will need to be on the same subnet **"192.168.112.1"**.

5. Start the **"F-Response LM Service"** and select **"OK"**

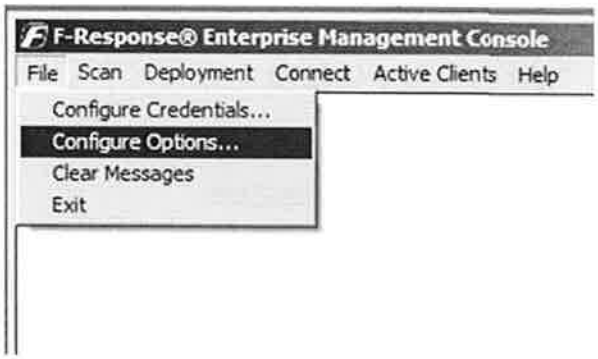
Exercise – Create a Deployable Remote Forensic Agent

1. Launch **`F-Response Enterprise Management Console`**





2. **Select `File -> Configure Options`**

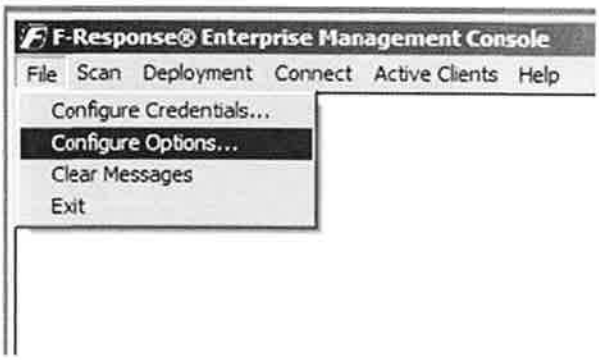


You will need to configure the following:

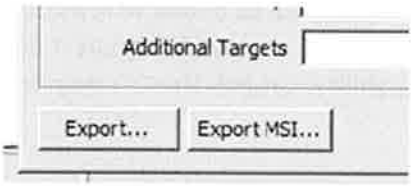
- License Manager IP -> **DO NOT CHANGE**
- License Manager Port -> **5681 – DO NOT CHANGE**
- Host Configuration-> **Physical Memory-> CHECKED**
- Username -> `sansforensics`
- Password -> `forensics1234`
- Service Name -> `F-Response Agent Service`
- Service Desc-> `F-Response Enterprise Agent Service`
- Executable -> `C:\program files\f-response\f-response enterprise edition\f-response-ent.exe`

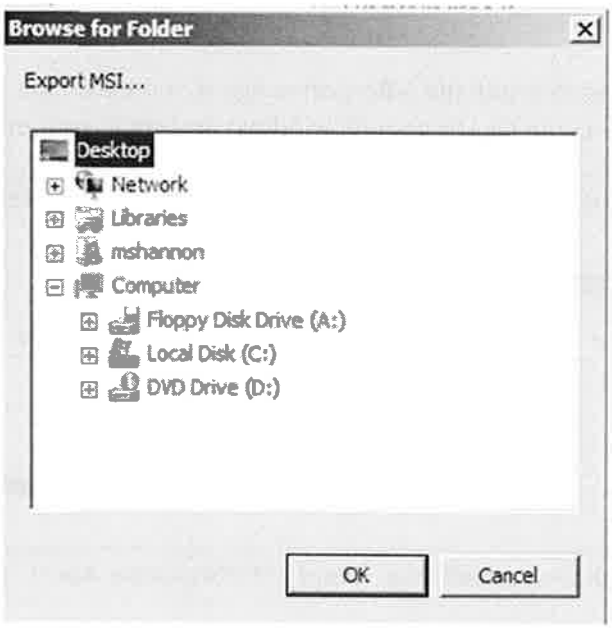
Press **Ok** to save changes.

3. "Select `File -> Configure Options`" again



4. Click the button at the bottom of the window to Export the deployable **F-ResponseAgent.MSI.** Once created, this .msi file can be distributed to thousands of systems in your environment.





5. **Deploy Agent(s) to systems you would like to examine remotely. We recommend another system on your network or lab. You can also use your own host, but it doesn't show the true capabilities of the enterprise incident response capability. Launch the "`fresponse.msi`" program on your local system by double clicking on it.**

Typically, we would deploy the agent to other systems in our environment via Group Policy or the F-Response Enterprise Management Console.

For this example, we are going to install the F-Response Agent Service on the same system we are using as the license server. However, this could be any generic windows system in your entire enterprise.

Launch the "F-Response Agent.MSI" to install the service locally on your system

6. **Start the `F-Response Agent Service` using your command prompt**

Launch an **`Administrator Command Prompt`** and use "sc" to start the service


```
C:\WINDOWS\system32>sc start "F-Response Agent Service"
```

Your output should look similar to the following:

```
C:\WINDOWS\system32>sc start "F-Response Agent Service"

SERVICE_NAME: F-Response Agent Service
        TYPE               : 10  WIN32_OWN_PROCESS
        STATE              : 2  START_PENDING
                                (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)
        WIN32_EXIT_CODE    : 0  (0x0)
        SERVICE_EXIT_CODE  : 0  (0x0)
        CHECKPOINT         : 0x0
        WAIT_HINT          : 0x7d0
        PID                : 3064
        FLAGS              :
```

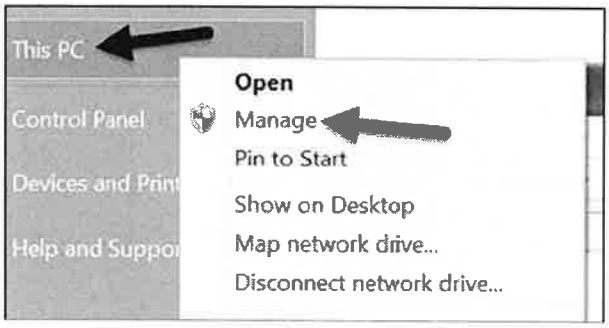
7. **Write down the IP address of the system you deployed the AGENT to by typing "ipconfig"**

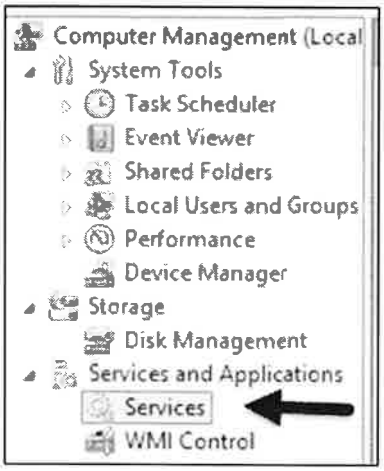
   AGENT IP ADDRESS = _____

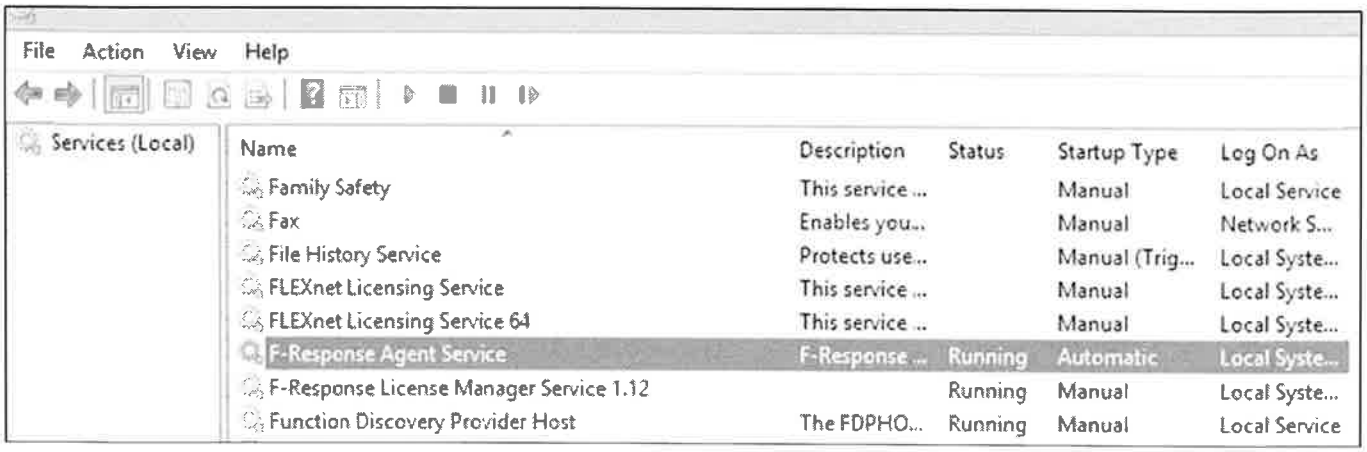
   In this example, we are likely only using VMNET8

8. **OPTIONAL -> Using the Services GUI, start the "F-Response Agent Service"**

Right click on "Your System" and Select Manage







Right-click, choose Start

Exercise – USE SIFT Workstation to connect to remote system(s) you have deployed the agent to and started the service

1. Switch to your SIFT Workstation

2. Login to see the available nodes (note in this EXAMPLE, we are using the server IP which on my system is 192.168.112.1, but on your system, this should match the IP address that you deployed the agent to.)

```
$ sudo su -

# f-response-accel-lin -n sansforensics -p forensics1234 -s
192.168.XXX.YYY
```

```
root@siftworkstation:/# f-response-accel-lin -n sansforensics -p forensics1234 -s 192.168.112.1
F-Response Acclerator - Linux Version 5.0.3
F-Response Acclerator for Linux requires Open-iSCSI.
Checking for Open-iSCSI utils now..
Open-iSCSI (iscsiadm) found.
Connecting to F-Response Target 192.168.112.1:3260...
Discovery Results.
F-Response Target = iqn.2008-02.com.f-response.jotumheim:disk-0
F-Response Target = iqn.2008-02.com.f-response.jotumheim:vol-c
F-Response Target = iqn.2008-02.com.f-response.jotumheim:pmem
Populating Open-iSCSI with node details..
Node information complete, adding authentication details.
```

3. Login to Remote Hard Drive Target - Usually Disk-0. Note in the above example, my system name was **jotumheim**. The exact name of the system you are logging into will change for each system you are logging into. Please note that it is a lowercase "**L**" in the **f-response-accel-lin** command. ?????? = your system name.

```
# f-response-accel-lin -l iqn.2008-02.com.f-response.??????:disk-0
```

```
root@siftworkstation:/# f-response-accel-lin -l iqn.2008-02.com.f-response.jotumheim:disk-0
F-Response Acclerator - Linux Version 5.0.3
F-Response Acclerator for Linux requires Open-iSCSI.
Logging in to [iface: default, target: iqn.2008-02.com.f-response.jotumheim:disk-0, portal: 192.168.112.1,3260]
Login to [iface: default, target: iqn.2008-02.com.f-response.jotumheim:disk-0, portal: 192.168.112.1,3260]: successful
IQN:iqn.2008-02.com.f-response.jotumheim:disk-0 attached as /dev/sdc
```

4. Run **fdisk -l** and examine output locating new drive that is attached at /dev/sdc

```
# fdisk -l
```

```
Disk /dev/sdc: 512.1 GB, 512110190592 bytes
255 heads, 63 sectors/track, 62260 cylinders, total 1000215216 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x9ef8684e

   Device Boot      Start         End      Blocks   Id  System
/dev/sdc1   *        2048      718847      358400    7  HPFS/NTFS/exFAT
/dev/sdc2          718848  1000212479   499746816    7  HPFS/NTFS/exFAT
```

5. Mount larger partition via the SIFT Workstation using the mount command. (Note: **/dev/sdc2** is what it is on the example, your actual results will vary depending on your system). You can determine the largest partition by examining the partition with the largest number of blocks. Also, usually on Win7 and later systems the 2nd partition is the C:\ of the system

```
# imageMounter.py /dev/sdc /mnt/windows_mount2
```

```
root@siftworkstation:/# imageMounter.py /dev/sdc /mnt/windows_mount2
[+] Creating Temp Mount Point at /mnt/windows_mount2/0
[+] Attempting to Mount Partition 0 at /mnt/windows_mount2/0
   [-] Mounted /dev/sdc at /mnt/windows_mount2/0
   [-] To unmount run 'sudo umount /mnt/windows_mount2/0'
[+] Creating Temp Mount Point at /mnt/windows_mount2/1
[+] Attempting to Mount Partition 1 at /mnt/windows_mount2/1
   [-] Mounted /dev/sdc at /mnt/windows_mount2/1
   [-] To unmount run 'sudo umount /mnt/windows_mount2/1'
```

6. Change Directories to the **/mnt/windows_mount2** directory and examine files. You should now be able to see the files from the remote system with the F-Response Agent installed.

```
# cd /mnt/windows_mount2/1

# ls
```

```
root@siftworkstation:/# cd /mnt/windows_mount2/1/
root@siftworkstation:/mnt/windows_mount2/1# ls
Apps                        END                 OneDriveTemp                System Volume Information
BOOTNXT                     ESD                 PerfLogs                    temp
Config.Msi                  hiberfil.sys        ProgramData                 Users
dell                        inetpub             Program Files               Windows
dell.sdr                    Intel               Program Files (x86)
Documents and Settings      logFileUI.txt       $Recycle.Bin
Drivers                     MSOCache            swapfile.sys
```

7. Change Directories to the **Windows/System32/config directory and extract the registry contents of the SAM and SYSTEM hives. Please note that due to "lower/upper case conversions" your exact directory path might vary on your own system. You might need to use "tab complete" to help you complete the directory path.**

```
# cd /mnt/windows_mount2/1

# cd Windows/System32/config

# rip.pl -r SAM -f sam > /cases/sam.txt

# less /cases/sam.txt
```

```
root@siftworkstation:/mnt/windows_mount2/1# cd Windows/System32/config/
root@siftworkstation:/mnt/windows_mount2/1/Windows/System32/config# rip.pl -r SAM -f sam > /cases
/sam.txt
Parsed Plugins file.
Launching samparse v.20120722
samparse complete.
root@siftworkstation:/mnt/windows_mount2/1/Windows/System32/config# less sam.txt
```

```
# rip.pl -r SYSTEM -f system > /cases/system.txt

# less /cases/system.txt
```

Exercise – Attach Remote Memory and Perform Memory Acquisition

1. Login to Appropriate Target - Usually PMEM. Please note that it is a lowercase "**L**" in the **f-response-accel-lin** command. ?????? = your target system name.

Example (**Memory**) = **iqn.2008-02.com.f-response.?????:pmem**

```
# f-response-accel-lin -l iqn.2008-02.com.f-response.?????????:pmem
```

```
root@siftworkstation:/# f-response-accel-lin -l iqn.2008-02.com.f-response.jotunheim:pmem
F-Response Acclerator - Linux Version 5.0.3
F-Response Acclerator for Linux requires Open-iSCSI.
Logging in to [iface: default, target: iqn.2008-02.con.f-response.jotunheim:pmem, portal: 192.168.112.1,3260]
Login to [iface: default, target: iqn.2008-02.com.f-response.jotunheim:pmem, portal: 192.168.112.1,3260]: successful
IQN:iqn.2008-02.com.f-response.jotunheim:pmem attached as /dev/sdd
```

2. Run **fdisk -l** and examine output locating the memory image.
   a. It should not show a valid partition table at **/dev/sdd**

```
# fdisk -l
```

```
Disk /dev/sdd: 9636 MB, 9636413440 bytes
64 heads, 32 sectors/track, 1148 cylinders, total 2352640 sectors
Units = sectors of 1 * 4096 = 4096 bytes
Sector size (logical/physical): 4096 bytes / 4096 bytes
I/O size (minimum/optimal): 4096 bytes / 4096 bytes
Disk identifier: 0x00000000

Disk /dev/sdd doesn't contain a valid partition table
```

3. Image Remote System Memory using dc3dd

Example (**Memory**) = **iqn.2008-02.com.f-response.?????:pmem** = **/dev/sdd** (note on your system /dev/sdd could be different)

```
# dc3dd if=/dev/sdd of=/cases/remote-system-memory.img hash=md5
hlog=/cases/remote-system-memory.md5
```

```
root@siftworkstation:/# dc3dd if=/dev/sdd of=/cases/remote-system-memory.img hash=md5 hlog=/cases/remote-system-memory.md5

dc3dd 7.1.614 started at 2014-05-23 17:00:48 +0100
compiled options:
command line: dc3dd if=/dev/sdd of=/cases/remote-system-memory.img hash=md5 hlog=/cases/remote-system-memory.md5
device size: 2352640 sectors (probed)
sector size: 4096 bytes (probed)
392626176 bytes (374 M) copied ( 4%), 10.0541 s, 37 M/s
```

In the next section we will learn how to use Volatility and Redline during memory analysis

Exercise – Detach Remote F-Response Targets

1. Detach the remote target system memory and hard drives.

```
# f-response-accel-lin -u iqn.2008-02.com.f-response.?????????:pmem
# cd /
# umount /mnt/windows_mount2/1
# umount /mnt/windows_mount2/0
# f-response-accel-lin -u iqn.2008-02.com.f-response.?????????:disk-0
```

Troubleshooting F-Response Enterprise

If you are having any problems using F-Response Enterprise you might need to check this troubleshooting guide.

1. F-Response Agent Service not starting correctly?

   a. Ensure your F-Response Enterprise Dongle is inserted and you have started the F-Response Enterprise License Manager

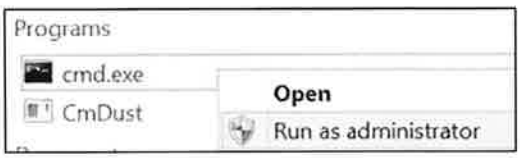
   b. Check your firewall. The firewall should be disabled or allow TCP ports 3260 and 5681 through.

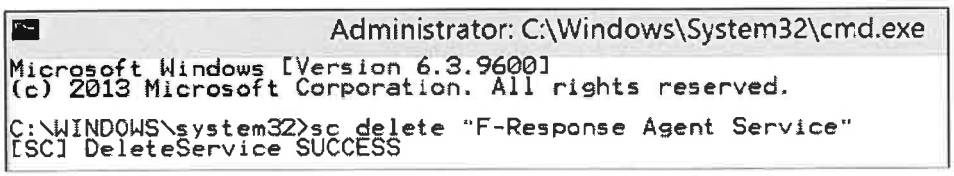
   c. Check the configuration options set in Step 2 of "Creating a Remote Deployable Agent" portion of this exercise above. Recreate the **`.msi`** file if necessary.

2. Misconfigured the F-ResponseAgent.MSI and need to reconfigure/reinstall it?

   a. Right click on F-ResponseAgent.msi file and click "**`uninstall`**"

   b. Launch an Administrator Level "command prompt" and type the following: **`sc delete "F-Response Agent Service"`**





   c. Reboot your system before starting over and installing a new .MSI file.

Exercise Takeaways

Being able to connect to 100s of system in your environment to perform incident response is critical. Using a deployable agent via group policy will allow you to connect to almost any IP address in your environment.

The next step is analysis and then making it automated -- using a script to connect and pull specific data from 100s of systems in your enterprise.

The days of pulling a hard drive each time you want to analyze a system are over. The time to be able to perform remote analysis is here and using tools such as F-Response, Mandiant MIR, Encase Enterprise, and many others gives the ability to scale from a single system to thousands.

Exercise 5A – Autostart Persistence Analysis

Objectives

- Examine output from **`autorunsc.exe`**
- Attempt to initially identify any suspicious processes

Exercise – Is there a suspicious process in the Autostart Programs on 10.3.58.7?

On the 10.3.58.7 system the `autorunsc.exe` was executed during Live Response on the *system*

```
C:\> autorunsc.exe -accepteula -a * -s -h -c > autoruns-xp-tdungan.csv
```

In order to speed up the examination process, your live response team went ahead and filtered the data for you using the output from AutoRuns found on the next page in this exercise.

1. Removed "Verified" and Trusted programs from Microsoft, Apple, Google, McAfee, and Paragon.
2. Eliminated the following columns to make it easier to read initially
   a. Publisher – only included untrusted, unknown, or blank programs
      i. There are some examples of well-known trusted publishers also having their code signing certificates stolen. While not impossible, it is extremely rare that malware will be using a trusted certificate from Microsoft to sign their code.
   b. Description – most were blank and most didn't have an adequate description to help you determine which one is likely malware
   c. MD5, SHA1, SHA256

For this exercise:

1. Examine the autorunsc.exe output on the next page.
2. Attempt to determine the most likely candidate that is a suspicious process. "List specifically what is wrong or odd about each suspicious process (you may not need all of the blank lines provided)
   a. \dllhost\svchost.exe
   b. _____
   c. _____
   d. _____
3. **Please note – answers will be provided in the next exercise – please do not work ahead.**

Entry Location	Entry	Category	Image Path	Launch String
HKLM\System\CurrentControlSet\Services	fresponse	Services	c:\windows\system32\f-response-ent.exe	f-response-ent.exe
HKLM\System\CurrentControlSet\Services	Changer	Drivers	File not found: C:\WINDOWS\System32\Drivers\Changer.sys	Changer
HKLM\System\CurrentControlSet\Services	i2omgmt	Drivers	File not found: C:\WINDOWS\System32\Drivers\i2omgmt.sys	i2omgmt
HKLM\System\CurrentControlSet\Services	lbrtfdc	Drivers	File not found: C:\WINDOWS\System32\Drivers\lbrtfdc.sys	lbrtfdc
HKLM\System\CurrentControlSet\Services	mfeavfk01	Drivers	File not found: C:\WINDOWS\System32\Drivers\mfeavfk01.sys	mfeavfk01
HKLM\System\CurrentControlSet\Services	mfefirek01	Drivers	File not found: C:\WINDOWS\System32\Drivers\mfefirek01.sys	mfefirek01
HKLM\System\CurrentControlSet\Services	Mnemosyne	Drivers	c:\windows\system32\mnemosyneI386.sys	\??\C:\WINDOWS\system32\Mnemosyne\I386.sys
HKLM\System\CurrentControlSet\Services	PCIDump	Drivers	File not found: C:\WINDOWS\System32\Drivers\PCIDump.sys	PCIDump
HKLM\System\CurrentControlSet\Services	PDCOMP	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDCOMP.sys	PDCOMP
HKLM\System\CurrentControlSet\Services	PDFRAME	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDFRAME.sys	PDFRAME
HKLM\System\CurrentControlSet\Services	PDRELI	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDRELI.sys	PDRELI
HKLM\System\CurrentControlSet\Services	PDRFRAME	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDRFRAME.sys	PDRFRAME
HKLM\System\CurrentControlSet\Services	vmscsi	Drivers	c:\windows\system32\drivers\vmscsi.sys	System32\DRIVERS\vmscsi.sys
HKLM\System\CurrentControlSet\Services	WDICA	Drivers	File not found: C:\WINDOWS\System32\Drivers\WDICA.sys	WDICA
HKLM\SYSTEM\CurrentControlSet\Control\Print\Monitors	ThinPrint	Print	c:\windows\system32\tpvmmon.dll	TPVMMon.dll
HKLM\System\CurrentControlSet\Services\WinSock2	12	Network	c:\program files\vmware\vsocklib.dll	C:\Program Files\VMware\vsocklib.dll
HKLM\System\CurrentControlSet\Services\WinSock2	13	Network	c:\program files\vmware\vsocklib.dll	C:\Program Files\VMware\vsocklib.dll
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	QuickTime	Logon	c:\program files\quicktime\qttask.exe	"C:\Program Files\QuickTime\QTTask.exe" -atboottime
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	svchost	Logon	c:\windows\system32\dllhost\svchost.exe	c:\windows\system32\dllhost\svchost.exe
HKLM\Software\Classes*\ShellEx\ContextMenuHandlers	7-Zip	Explorer	c:\program files\7-zip\7-zip.dll	HKCR\CLSID\{23170F69-40C1-278A-1000-000100020000}
HKLM\Software\Classes\Directory\ContextMenuHandlers	7-Zip	Explorer	c:\program files\7-zip\7-zip.dll	HKCR\CLSID\{23170F69-40C1-278A-1000-000100020000}
HKLM\Software\Classes\Directory\Shellex\DragDropHandlers	7-Zip	Explorer	c:\program files\7-zip\7-zip.dll	HKCR\CLSID\{23170F69-40C1-278A-1000-000100020000}
HKLM\Software\Classes\CLSID\{083863F1-70DE-11d0-BD40-00A0	Media	Codecs	c:\program files\movie maker\wmmfilt.dll	HKCR\CLSID\{F689C580-F60F-479F-886D-A01D091756 73}
HKLM\Software\Classes\CLSID\{083863F1-70DE-11d0-BD40-00A0	ShotBoundary	Codecs	c:\program files\movie maker\wmmfilt.dll	HKCR\CLSID\{FE487D4D-DC92-4F10-91DE-C676E25624CF}

Exercise Takeaways

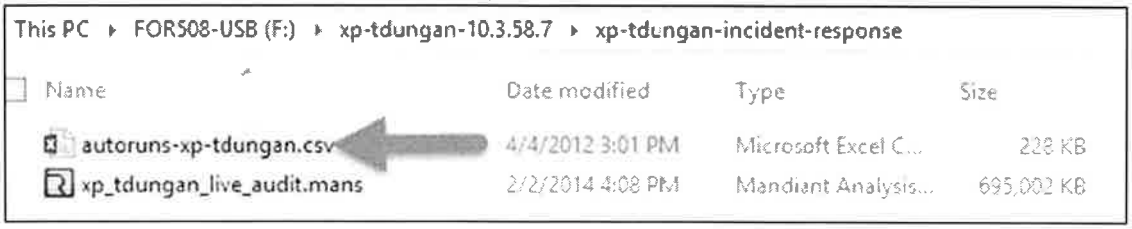
Without additional information, unless you already know and understand Windows OS processes, it is fairly difficult to identify suspicious processes via a variety of tool output. While having a big collection of output from various incident response tools is nice, it is not very useful if you cannot quickly and easily identify anomalies.

In the next section, we will do this exercise again, but this time we will cover some of the core Windows OS processes that you should be aware of. With this additional knowledge, doing this exercise again will make easy work of identifying anomalies in the output of a tool such as autorunsc.exe.

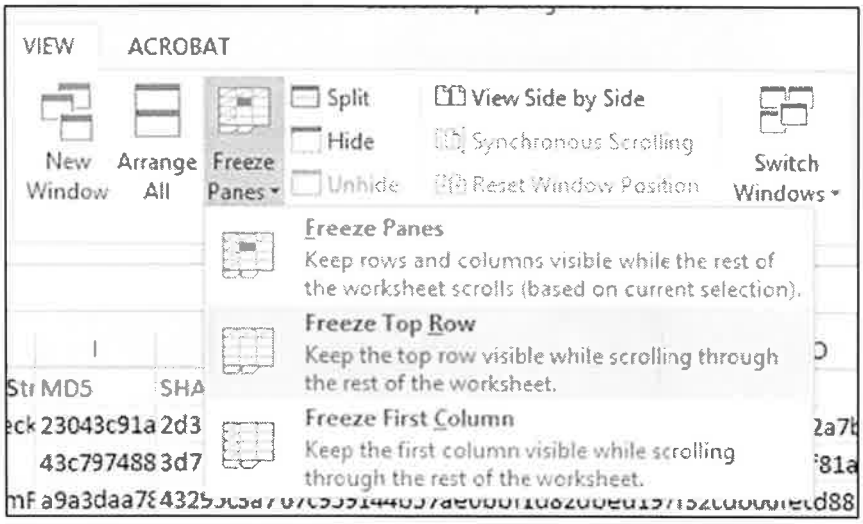
Following this takeaway is a step by step guide in how to directly examine autorunsc.exe output via the csv file that it outputs. We welcome you to examine the csv file directly as it is good practice, but for this exercise we would prefer you quickly just look at the output provided.

Out-Of-Class Exercise – How to Examine and Filter `autoruns-xp-tdungan.csv`

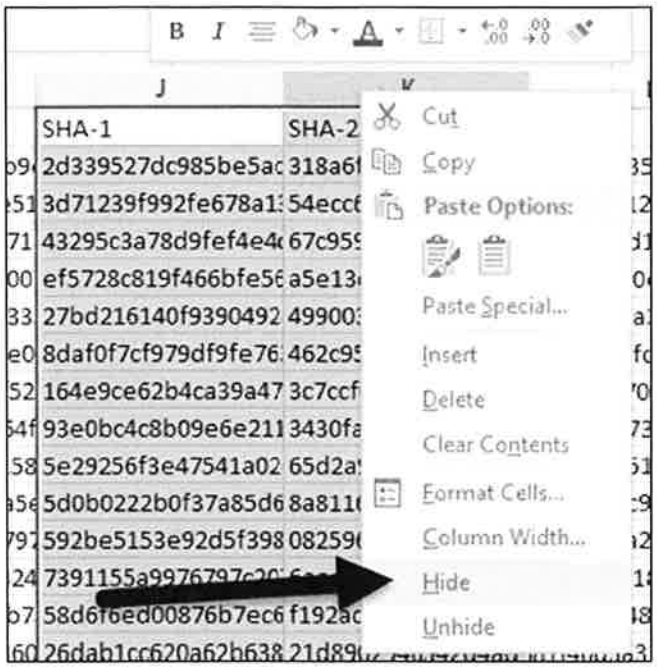
1. Open the **`autoruns-xp-tdugnan.csv`** file found on your USB under **`/xp-tdungan-10.3.58.7/xp-tdungan-incident-response`** using Microsoft Excel – simply "double clicking" autoruns-xp-tdungan.csv usually works



2. Select View -> Freeze Panes -> Freeze Top Row



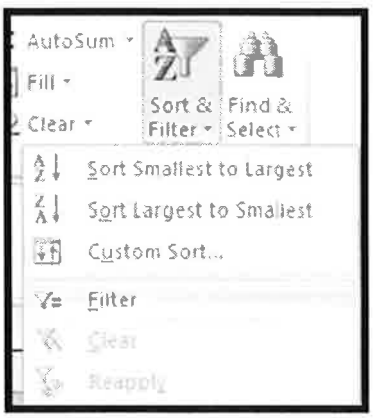
3. Hide Columns "SHA-1" and "SHA-256"



4. Select HOME Ribbon.

5. Select all Cells "CTRL-A".

6. In Home Ribbon -> Sort and Filter – Filter and you will be ready to begin analysis.

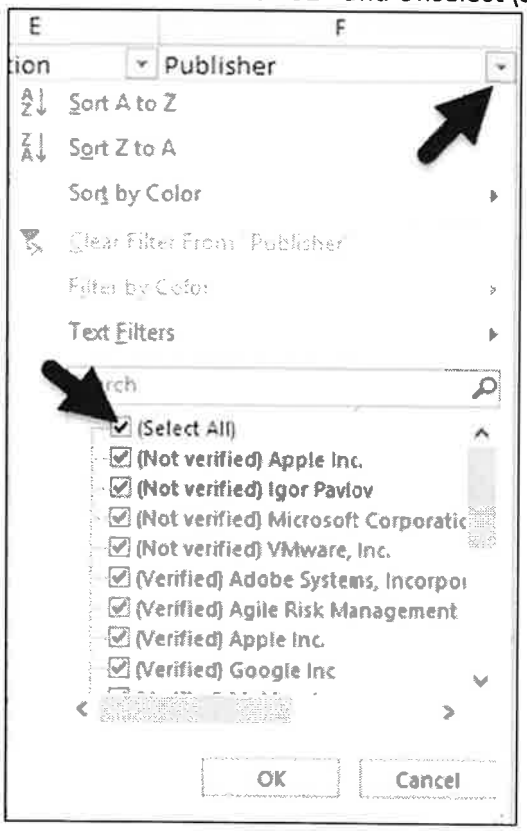


Out-Of-Class Exercise – Autorunsc Output Analysis

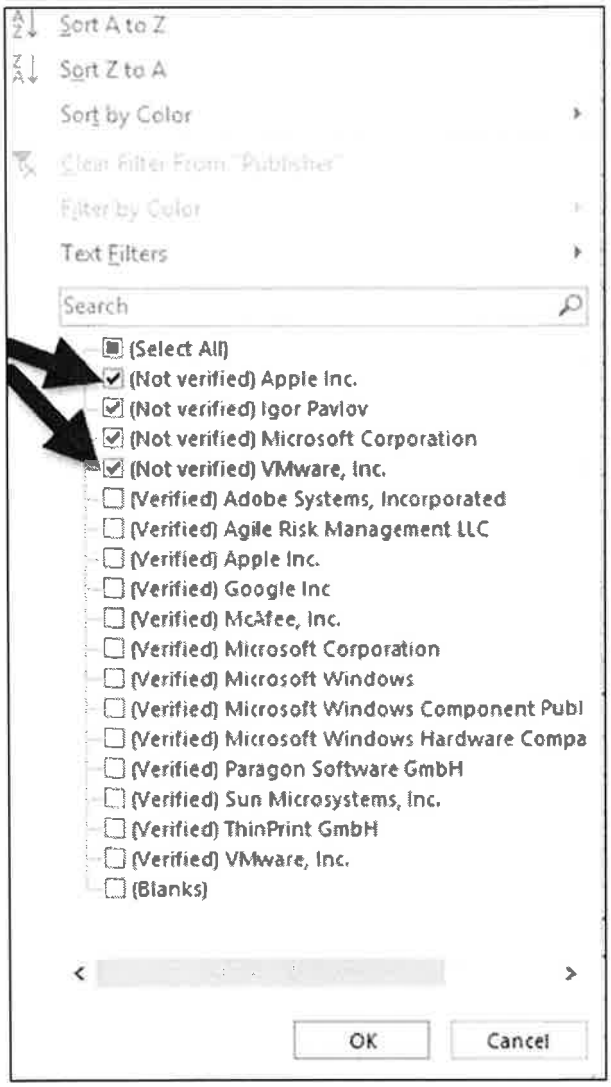
1. Initially – Filter out all signed and trusted (verified) startup locations from being viewed. Select "Publisher" column and select only those publishers that we are not sure about. While this is by far a perfect way of eliminating trusted publishers from being examiner, it is initially good enough.

There are some examples of well-known trusted publishers also having their code signing certificates stolen. However, if you happen to discover a new one, especially a stolen certificate from Microsoft, Apple, Google, or McAfee – this would be nationwide news and you should prepare for your 10 minutes of fame as you will be heavily interviewed on how you discovered it. In a nutshell, initially eliminate well known trusted publishers from your examination. While not impossible, it is extremely rare that malware will be using a trusted certificate from Microsoft to sign their code.

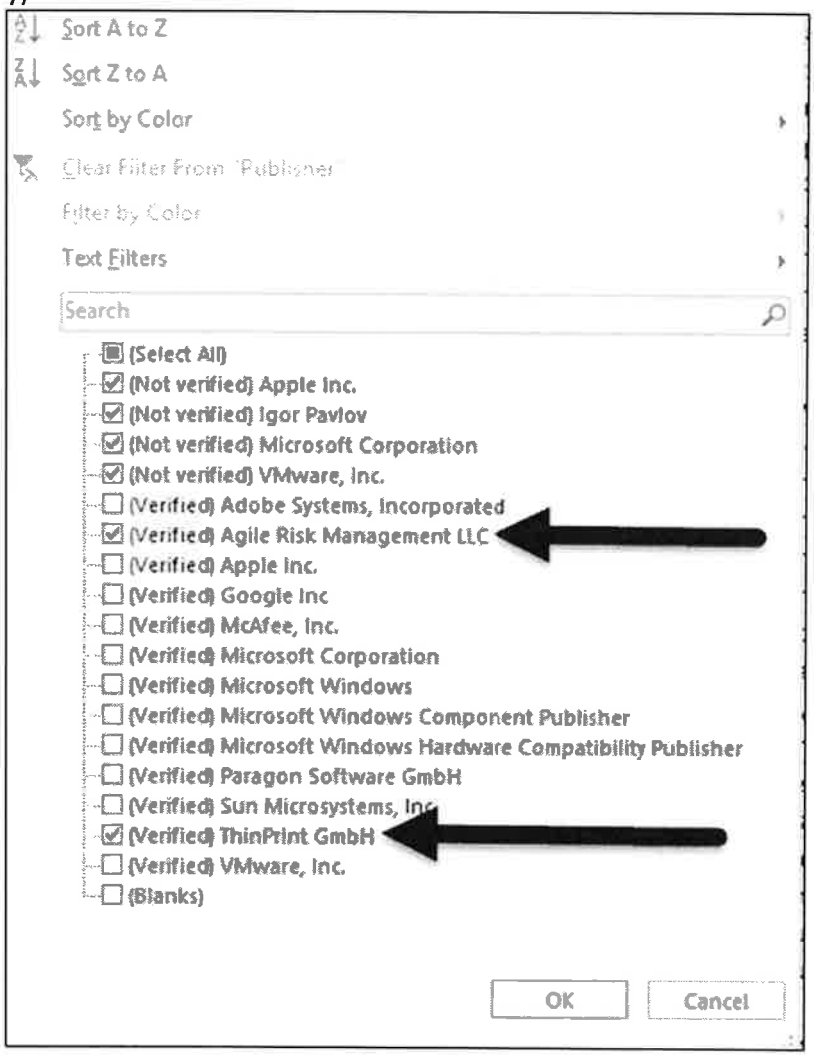
a. Select Dropdown Filter next to "**`Publisher`**" and Unselect (Select All)



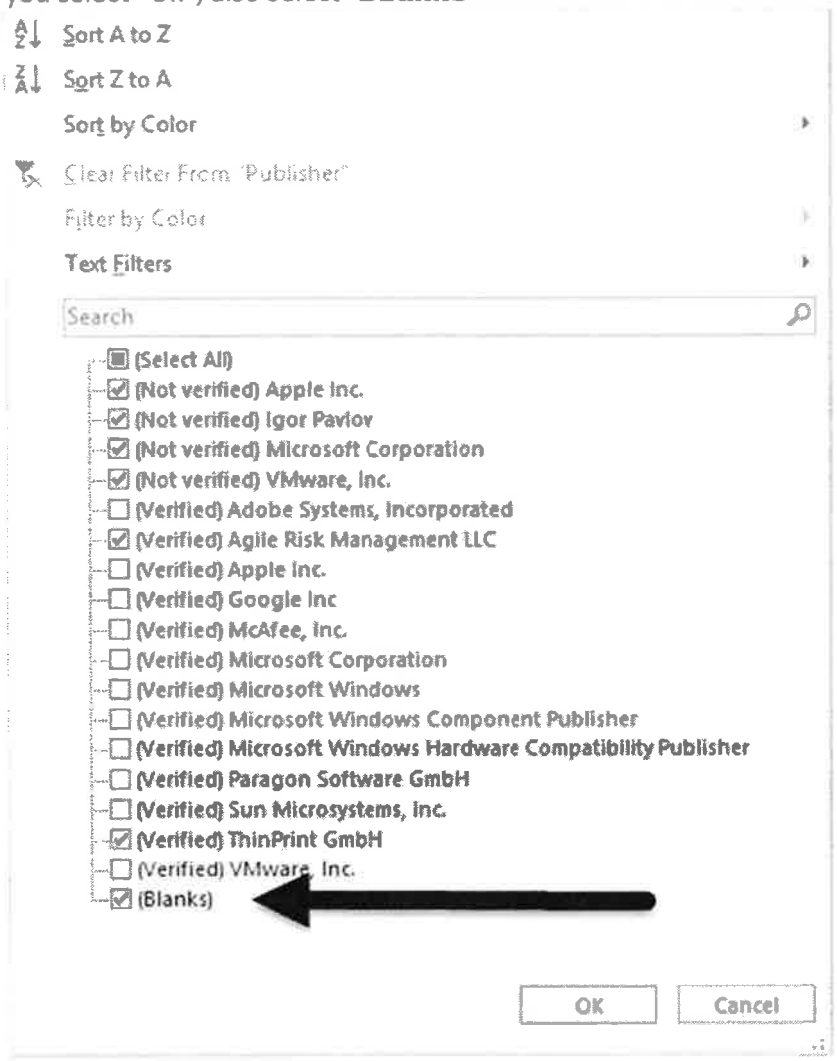
b. Select all publishers that are "**`Not Verified`**"



c. Before you select "**Ok**", also select any publishers not immediately recognized as a core software publisher (e.g. something you might need to "google search" to verify they are a real company)



d. Before you select "Ok", also select "`blanks`"



This page intentionally left blank.

Exercise5B– Autostart Persistence Analysis

Objectives

- Examine output from **`autorunsc.exe`**
- Attempt to initially identify any suspicious processes using the "What is Normal Poster?"
- Using our understanding of core windows operating system processes and services, it is likely that any malware that does not share characteristics with expected usage might be considered suspicious or an anomaly.

Exercise – Is there a suspicious process in the Autostart Programs collected from 10.3.58.7?

On the 10.3.58.7 system the **`autorunsc.exe`** was executed during Live Response on the system

```
C:\> autorunsc.exe -accepteula -a * -s -h -c > autoruns-xp-tdungan.csv
```

In order to speed up the examination process, your live response team went ahead and filtered the data for you using the output from AutoRuns found on the next page in this exercise.

1. Removed "Verified" and Trusted programs from Microsoft, Apple, Google, McAfee, and Paragon.
2. Eliminated the following columns to make it easier to read initially
   a. Publisher – only included untrusted, unknown, or blank programs
      i. There are some examples of well-known trusted publishers also having their code signing certificates stolen. While not impossible, it is extremely rare that malware will be using a trusted certificate from Microsoft to sign their code.
   b. Description – most were blank and most didn't have an adequate description to help you determine which one is likely malware
   c. MD5, SHA1, SHA256

For this exercise:

1. Examine the autorunsc.exe output on the next page.
2. **Look for processes identified by autoruns that are present on the "What is Normal" Poster. What is the most likely candidate that is a suspicious process?**
3. List specifically what is wrong with odd about the selected suspicious process.
   a. _____
   b. _____
   c. _____
   d. _____
   e. _____

```
C:\> autorunsc.exe -accepteula -a * -s -h -c > autoruns-xp-tdungan.csv
```

Entry Location	Entry	Category	Image Path	Launch String
HKLM\System\CurrentControlSet\Services	fresponse	Services	c:\windows\system32\f-response-ent.exe	f-response-ent.exe
HKLM\System\CurrentControlSet\Services	Changer	Drivers	File not found: C:\WINDOWS\System32\Drivers\Changer.sys	Changer
HKLM\System\CurrentControlSet\Services	i2omgmt	Drivers	File not found: C:\WINDOWS\System32\Drivers\i2omgmt.sys	i2omgmt
HKLM\System\CurrentControlSet\Services	lbrtfdc	Drivers	File not found: C:\WINDOWS\System32\Drivers\lbrtfdc.sys	lbrtfdc
HKLM\System\CurrentControlSet\Services	mfeavfk01	Drivers	File not found: C:\WINDOWS\System32\Drivers\mfeavfk01.sys	mfeavfk01
HKLM\System\CurrentControlSet\Services	mfefirek01	Drivers	File not found: C:\WINDOWS\System32\Drivers\mfefirek01.sys	mfefirek01
HKLM\System\CurrentControlSet\Services	Mnemosyne	Drivers	c:\windows\system32\mnemosynei386.sys	\??\C:\WINDOWS\system32\Mnemosynei386.sys
HKLM\System\CurrentControlSet\Services	PCIDump	Drivers	File not found: C:\WINDOWS\System32\Drivers\PCIDump.sys	PCIDump
HKLM\System\CurrentControlSet\Services	PDCOMP	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDCOMP.sys	PDCOMP
HKLM\System\CurrentControlSet\Services	PDFRAME	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDFRAME.sys	PDFRAME
HKLM\System\CurrentControlSet\Services	PDRELI	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDRELI.sys	PDRELI
HKLM\System\CurrentControlSet\Services	PDRFRAME	Drivers	File not found: C:\WINDOWS\System32\Drivers\PDRFRAME.sys	PDRFRAME
HKLM\System\CurrentControlSet\Services	vmscsi	Drivers	c:\windows\system32\drivers\vmscsi.sys	System32\DRIVERS\vmscsi.sys
HKLM\System\CurrentControlSet\Services	WDICA	Drivers	File not found: C:\WINDOWS\System32\Drivers\WDICA.sys	WDICA
HKLM\SYSTEM\CurrentControlSet\Control\Print\Monitors	ThinPrint	Print	c:\windows\system32\tpvmmon.dll	TPVMMon.dll
HKLM\System\CurrentControlSet\Services\WinSock2	12	Network	c:\program files\vmware\vsocklib.dll	C:\Program Files\VMware\vsocklib.dll
HKLM\System\CurrentControlSet\Services\WinSock2	13	Network	c:\program files\vmware\vsocklib.dll	C:\Program Files\VMware\vsocklib.dll
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	QuickTime	Logon	c:\program files\quicktime\qttask.exe	"C:\Program Files\QuickTime\QTTask.exe" -atboottime
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run	svchost	Logon	c:\windows\system32\dllhost\svchost.exe	c:\windows\system32\dllhost\svchost.exe
HKLM\Software\Classes*\ShellEx\ContextMenuHandlers	7-Zip	Explorer	c:\program files\7-zip\7-zip.dll	HKCR\CLSID\{23170F69-40C1-278A-1000-000100020000}
HKLM\Software\Classes\Directory\ContextMenuHandlers	7-Zip	Explorer	c:\program files\7-zip\7-zip.dll	HKCR\CLSID\{23170F69-40C1-278A-1000-000100020000}
HKLM\Software\Classes\Directory\Shellex\DragDropHandlers	7-Zip	Explorer	c:\program files\7-zip\7-zip.dll	HKCR\CLSID\{23170F69-40C1-278A-1000-000100020000}
HKLM\Software\Classes\CLSID\{083863F1-70DE-11d0-BD40-00A0	Media	Codecs	c:\program files\movie maker\wmmfilt.dll	HKCR\CLSID\{F6B9C580-F60F-479F-885D-A01D09175673}
HKLM\Software\Classes\CLSID\{083863F1-70DE-11d0-BD40-00A0	ShotBoundary	Codecs	c:\program files\movie maker\wmmfilt.dll	HKCR\CLSID\{FE48704D-DC92-4F10-91DE-C676E25624CF}

Exercise – Autorunsc Output Analysis Step-By-Step

Using the "What is Normal Poster?" to help us understand core windows processes, look at specifically `svchost.exe`

`svchost.exe`

Image Path: `%SystemRoot%\System32\svchost.exe`

Parent Process: services.exe

Number of Instances: Five or more

User Account: Varies depending on svchost instance, though it typically will be Local System, Network Service, or Local Service accounts. Instances running under any other account should be investigated.

Start Time: Typically within seconds of boot time. However, services can be started after boot, which might result in new instances of svchost.exe well after boot time.

Description: The generic host process for Windows Services. It is used for running service DLLs. Windows will run multiple instances of svchost.exe, each using a unique "-k" parameter for grouping similar services. Typical "-k" parameters include BTsvcs, DcomLaunch, RPCSS, LocalServiceNetworkRestricted, netsvcs, LocalService, NetworkService, LocalServiceNoNetwork, secsvcs, and LocalServiceAndNoImpersonation. Malware authors often take advantage of the ubiquitous nature of svchost.exe and use it either directly or indirectly to hide their malware. They use it directly by installing the malware as a service in a legitimate instance of svchost.exe.

Alternatively, they use it indirectly by trying to blend in with legitimate instances of svchost.exe, either by slightly misspelling the name (e.g., scvhost.exe) or spelling it correctly but placing it in a directory other than System32. Keep in mind that a legitimate svchost.exe should always run from %SystemRoot%\System32, should have services.exe as its parent, and should host at least one service. Also, on default installations of Windows 7, all service executables and all service DLLs are signed by Microsoft.

3. List specifically what is wrong with odd about the selected suspicious process.

`C:\Windows\System32\dllhost\svchost.exe` **is the suspicious process**

a. **Image Path is C:\Windows\System32\dllhost. It should be C:\Windows\System32**
b. **Entry Location shows svchost.exe being referenced by HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run, but the poster identifies the owning parent of svchost to be Services.exe (the Entry Location should be HKLM\System\CurrentControlSet\Services)**

c. **Since svchost.exe is present in a "Run" registry key, its start time will be when a user logs onto the system, not at boot as identified on the poster**
d. **The Launch String (command line) should include a "-k" parameter with it as mentioned in the poster Description field for svchost.exe.**
e. **The code is not signed by Microsoft (remember at the beginning of this exercise we listed how the team pre-filtered the data, including removing any signed and verified programs from Microsoft). Svchost.exe is a core windows process and should be signed and verified by Microsoft similar to the one that exists in the `C:\Windows\System32` folder.**

Finding any one of these would be a good enough reason to look deeper at this suspicious process. If you identified all of them from the output provided, you are a ninja!

Exercise Takeaways

In an intrusion case, spotting the difference between abnormal and normal is often the difference between success and failure. Your mission is to quickly identify suspicious artifacts in order to verify potential intrusions.

Even if you were able to identify the suspicious process in the first part of this exercise, it is likely you were not able to identify all the parts of it that were anomalous.

Knowing what is normal on a Windows host helps cut through the noise to quickly locate potential malware

As a result of your examination of the **`autorunsc.exe`** output from the **`10.3.58.7`** we have identified a suspicious process called **`svchost`** that is in an incorrect directory **`c:\windows\system32`** that starts automatically when a user logs into the system.

The last part of the exercise is intended for you to help facilitate memorization of a key windows process, **`svchost.exe`**, and the properties that **`svchost.exe`** should exhibit on a windows operating system.

In the next section, we will begin memory analysis and attempt to identify if **`svchost`** is running and if it has any indicators that might be even more suspicious.

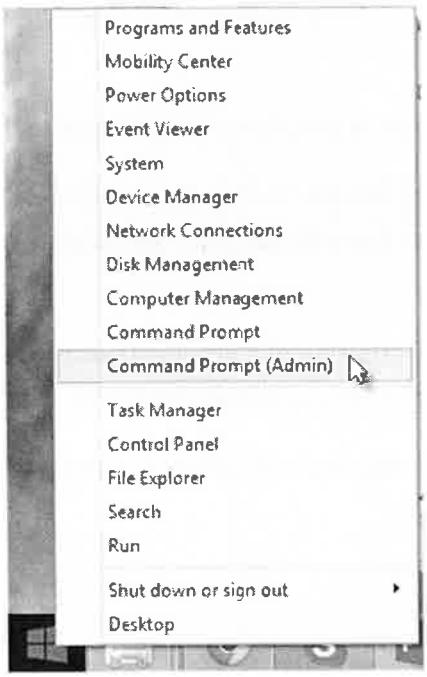
Optional Exercise, Exercise 6– WMI, PowerShell, and Kansa

Objectives

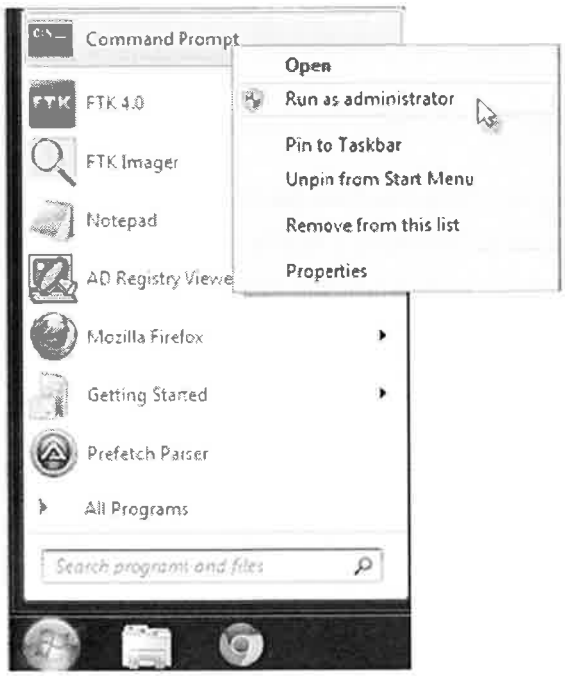
- Review output from WMI commands
- Use PowerShell to run individual Kansa modules
- Optional: Enable PowerShell remoting on your system and execute Kansa

Exercise Preparation

1. Open an **Administrator** command prompt by right-clicking on the **cmd.exe** icon (Windows 7) or right-clicking on the Windows icon and selecting "Command Prompt (Admin)".



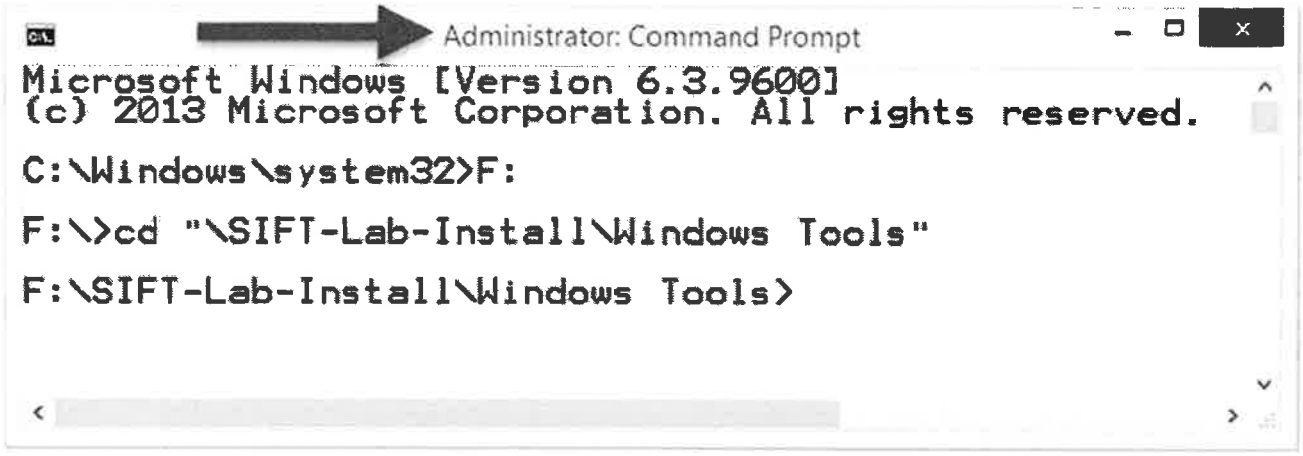
Windows 8



Windows 7

2. Ensure your FOR508 USB drive is plugged in and change volumes to the one represented by the USB drive ("F:" is assumed here, but your system may assign a different drive letter). Change directory to "SIFT-Lab-Install\Windows Tools"

```
C:\Windows\system 32> F:
F:\> cd "SIFT-Lab-Install\Windows Tools"
```


Administrator: Command Prompt
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.
C:\Windows\system32>F:
F:\>cd "\SIFT-Lab-Install\Windows Tools"
F:\SIFT-Lab-Install\Windows Tools>

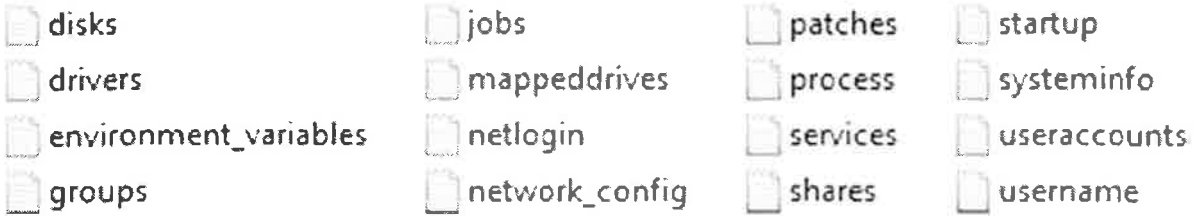
Exercise – Questions with Step-by-Step

1. **Live Collection with WMI**
   - Use notepad or another text editor to open the **`wmic_lr_local.cmd`** file from the "SIFT-Lab_Install\Windows Tools" folder on your FOR508 USB. Take a few minutes to read and get a feel for the commands that are included in the script.

   - Execute **`wmic_lr_local.cmd`**. You should see the script run and output files will be created in the same folder. Answer "no" to any questions for faster collection.

```
F:\SIFT-Lab-Install\Windows Tools> wmic_lr_local.cmd
```

   - You should have 15 or more new text files in the "Windows Tools" folder. Take five minutes to review the output for some of the files with your favorite text editor.



2. **Introducing PowerShell Scripts**

   - Type **`powershell`** in your command terminal to open Windows PowerShell.

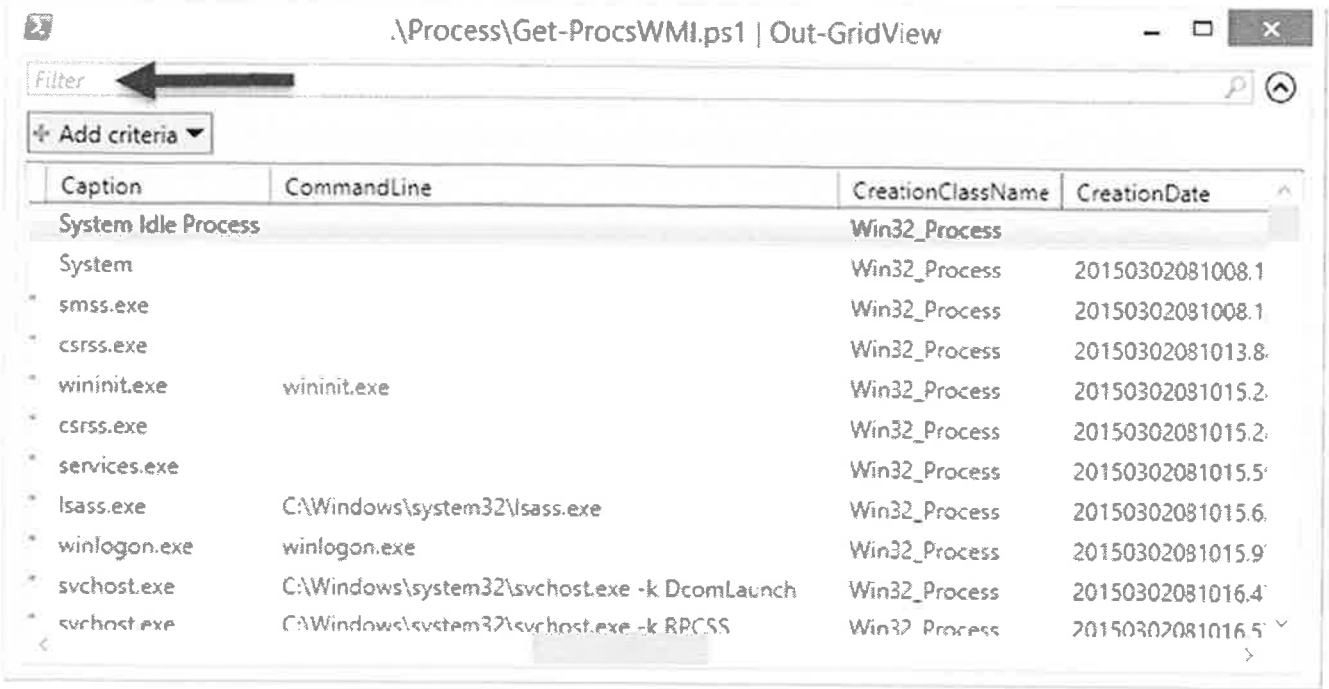
```
F:\SIFT-Lab-Install\Windows Tools> powershell
```

- Change directory to "\SIFT-Lab_Install\Windows Tools\Kansa-master\Modules", set your policy to allow all PowerShell scripts to run, and remove any Zone.Identifier alternate data streams (using Unblock-File).

```
PS F:\SIFT-Lab-Install\Windows Tools> cd Kansa-master\Modules

PS F:\SIFT-Lab-Install\Windows Tools> Set-ExecutionPolicy
unrestricted

PS F:\SIFT-Lab-Install\Windows Tools> ls -r *.ps1 | Unblock-File

   (Unblock-File is only available in PowerShell version 3+)
```

- Execute **.\Process\Get-ProcsWMI.ps1 | Out-GridView** and review the output.

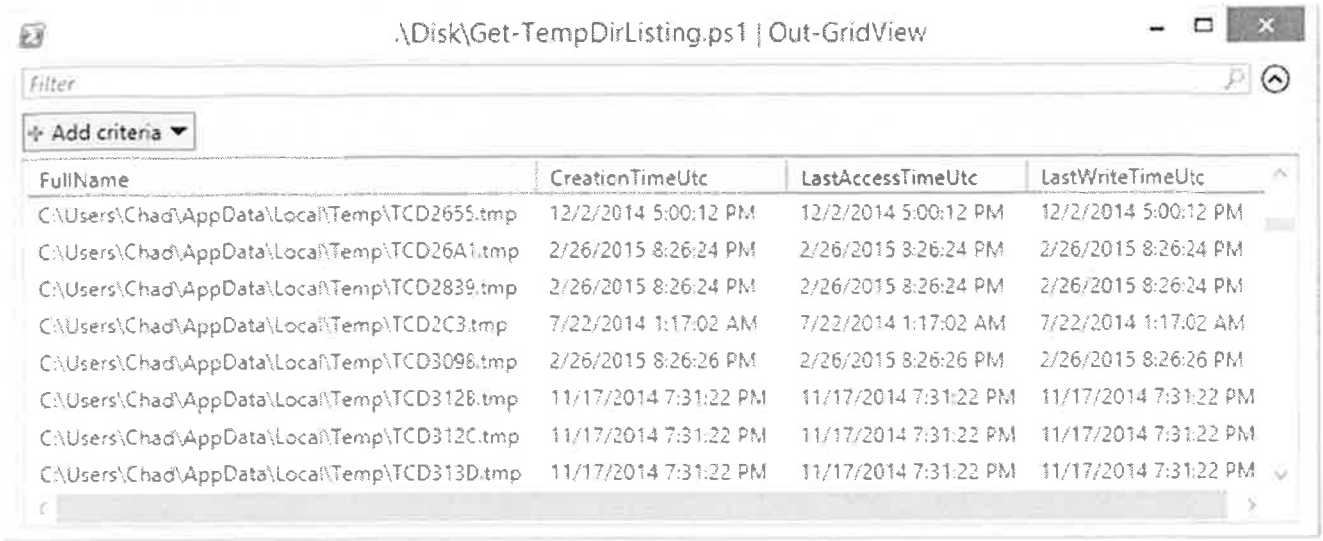
```
PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master\Modules>
.\Process\Get-ProcsWMI.ps1 | Out-GridView
```



WMI provides a wealth of data about each process, and the PowerShell script nicely formats it into a PowerShell object making it easy to view in the default outputter "Out-GridView". Notice that you can move columns and filter via the search form at the top of the dialog. In a real-world investigation we would review data for suspicious processes. Close the viewer when you are finished.

- Execute **.\Disk\Get-TempDirListing.ps1 | Out-GridView** and review the output.

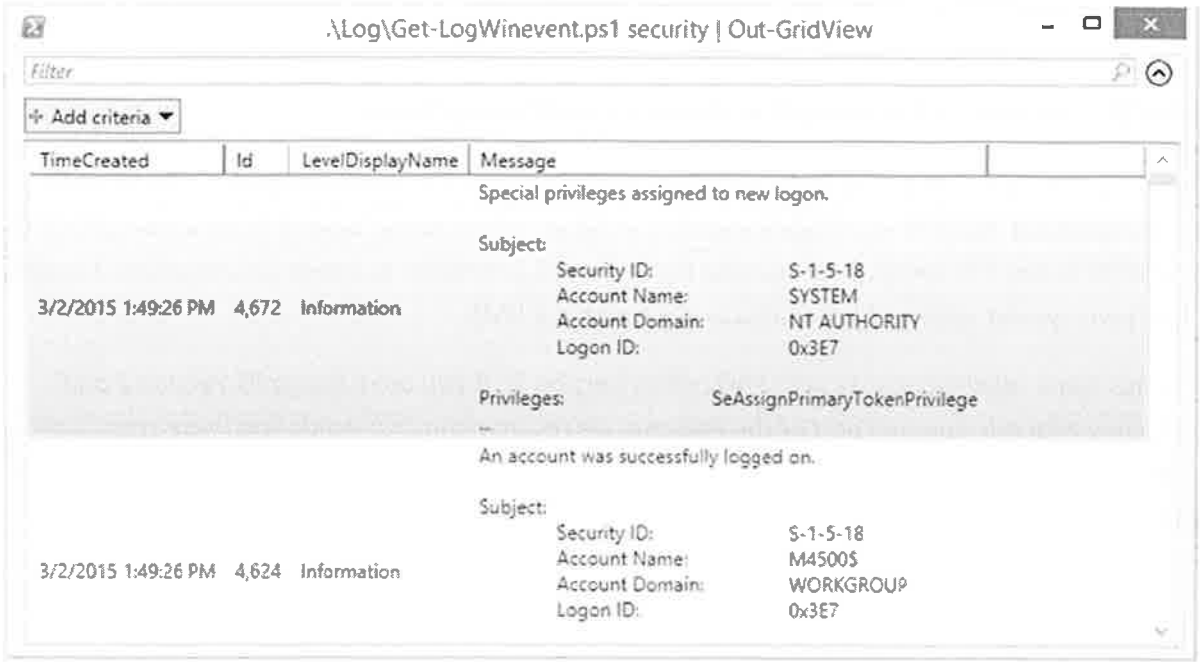
```
PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master\Modules>
.\Disk\Get-TempDirListing.ps1 | Out-GridView
```



This script is designed to gather information on files located in common "Temp" folders in Windows. Since these locations are often abused by droppers and other malware, we would review the output looking for unusual filenames. If you don't often run Disk Cleanup, expect a lot of files, many with strange names (which is one reason these folders are such a magnet for malware). Creation timestamps can sometimes help identify anomalies, and later in the course we will be putting data like this in context via Supertimelines, which make it easier to spot evil.

- Execute **`.\Log\Get-LogWinEvent.ps1 security | Out-GridView`** and review the output.

```
PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master\Modules>
.\Log\Get-LogWinEvent.ps1 security | Out-GridView
```



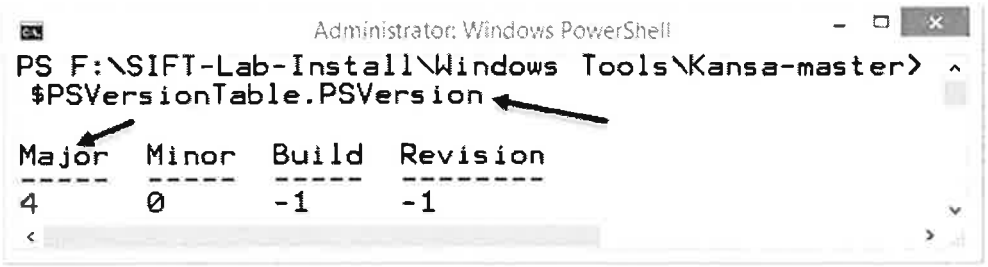
i. Choose an Event ID type from the "ID" column and use the filter bar to view only events of that type.

In this example we used a PS script to extract the Security Event Log and display it in an outputter (Out-GridView). In a real-world case, the script would default to writing the log to a file on disk. These logs could then be collected in a database with similar logs from other systems and analyzed for anomalies or used to track compromised account usage in the environment.

3. Optional (Advanced): Enable PowerShell Remoting and Execute Kansa

*Many Kansa scripts were not designed to be run alone and need to be run via **kansa.ps1** to be successful. Since running **kansa.ps1** requires PowerShell remoting enabled on the target system (your localhost in this case), we opted to leave it as an optional exercise for when you get access to a test system where it is okay to tinker with your system settings (it should also work within a VM).*

Note: Kansa has some reliability issues with PowerShell Version 2. If you are running PS Version 2 and having difficulties with this optional part of the exercise, we recommend first validating these steps with a similar system that has PowerShell version 3 (or perhaps upgrading your current PowerShell instance). You can find out your version by typing the following into PowerShell: `$PSVersionTable.PSVersion`



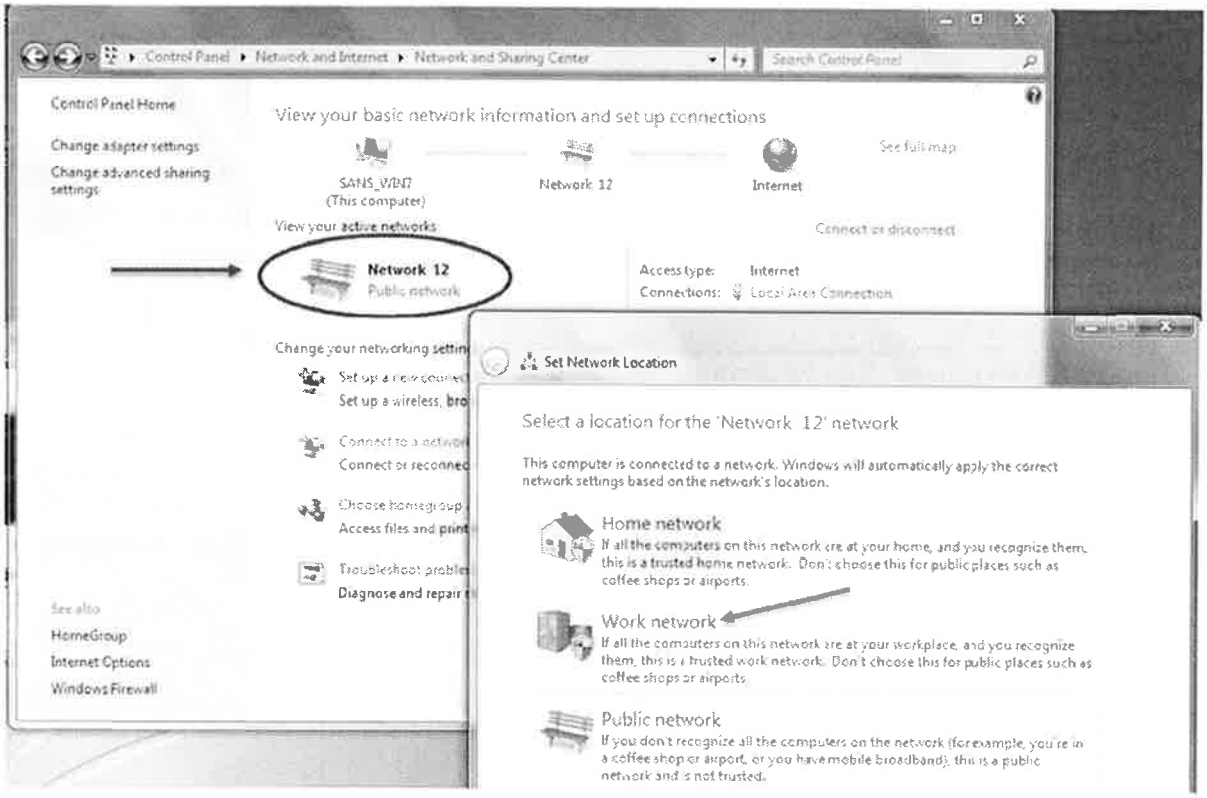
Reference the TechNet article "Tip: Enable and Use Remote Commands in Windows PowerShell" to get started with enabling WinRM, and/or follow the instructions below. [1] Keep in mind that PS remoting can be difficult to enable on some systems due to existing configurations (conversely, it may also already be enabled on some systems). Also, there are multiple different versions of PowerShell, and not all commands below will work with all versions.

Step 1: PowerShell remoting requires that all network connections be set to something other than "Public". This will be your first hurdle. If you are running PowerShell version 4+, you can run the following commands:

```
Get-NetConnectionProfile

Set-NetConnectionProfile -InterfaceIndex XX -NetworkCategory
Private

(XX is the number of the interface set to Public in Get-NetConnectionProfile. Run this
command for each profile set to public)
```

```
Administrator: Windows PowerShell

PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master> Get-NetConnectionProfile

Name             : Unidentified network
InterfaceAlias   : VMware Network Adapter VMnet8
InterfaceIndex   : 13
NetworkCategory  : Public
IPv4Connectivity : LocalNetwork
IPv6Connectivity : LocalNetwork

Name             : Unidentified network
InterfaceAlias   : Ethernet
InterfaceIndex   : 3
NetworkCategory  : Public
IPv4Connectivity : LocalNetwork
IPv6Connectivity : LocalNetwork


PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master> Set-NetConnectionProfile -Int
erfaceIndex 13 -NetworkCategory Private
```

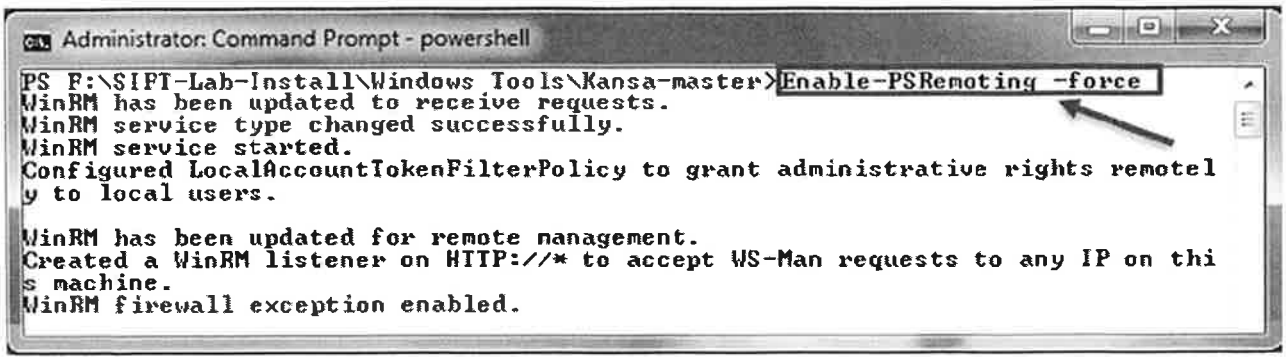
For older versions of PowerShell, open the Network and Sharing Center in the system Control Panel. Change any active networks to "Work Network" or "Private" depending on the version of Windows you are using.



Step 2: Enable PowerShell Remoting. Use the following command to enable PSRemoting:

```
Enable-PSRemoting -force
```

If this completes successfully, then configuration should be complete. If it does not complete successfully, it may be worth trying a slightly different method using the command "**Winrm quickconfig**".



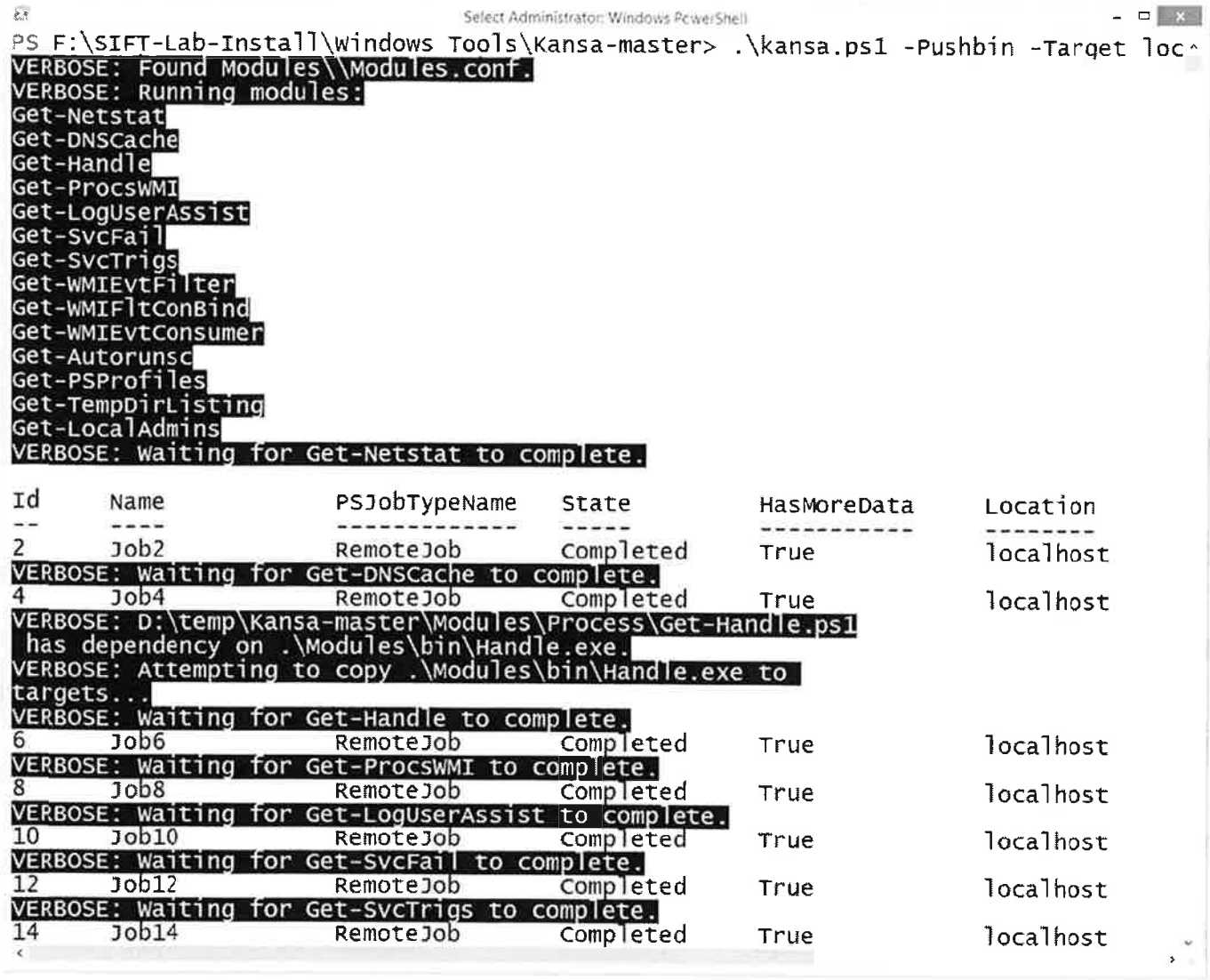
Step 3: Run Kansa on the Local Host.

First, make sure you are in the "Kansa-master" folder and that your PS script execution policy is set to unrestricted. Just in case you missed the step in a previous part of the exercise, use "Unblock-File" to remove any Zone.Identifier alternate data streams from the Kansa script directories.

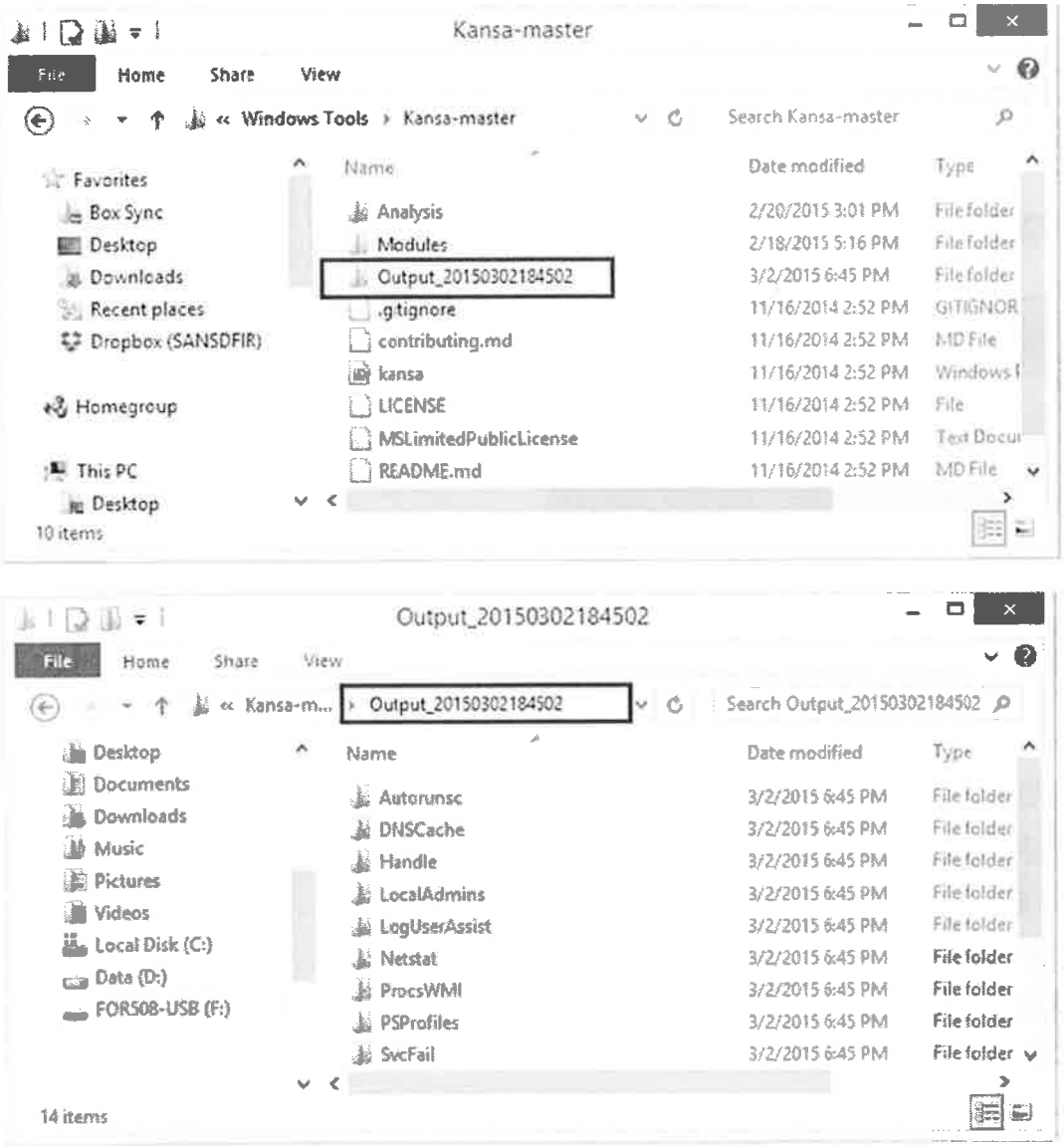
```
PS F:\SIFT-Lab-Install\Windows Tools> cd "\SIFT-Lab-
Install\Windows Tools\Kansa-master"

PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master>
Set-ExecutionPolicy unrestricted

PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master> ls -r *.ps1 |
Unblock-File

(Unblock-File is only available in PowerShell version 3+)
```

Now run Kansa specifying the localhost as the target.

```
PS F:\SIFT-Lab-Install\Windows Tools\Kansa-master> .\kansa.ps1
-Pushbin -Target localhost
```



An output folder will be created in the "Kansa-master" folder and inside will be sub-folders for each script that was successfully completed. Review the content in each folder to gain an understanding of the type of data that Kansa collects. In a real-world environment, we would have used the "-Targetlist" option to run simultaneously against multiple hosts, and the output of each host would be in the corresponding Output folders for review.



[1] Starting the WinRM service: https://technet.microsoft.com/en-us/magazine/ff700227.aspx

Exercise – Key Takeaways

- **WMI and PowerShell are a powerful (and free) means to collect system data across a Windows enterprise.**
- **It is important to test your scripts before you need them! While most commands and scripts are straightforward, individual configuration choices within the enterprise can greatly affect your success rate.**

This page intentionally left blank.

Optional Exercise 7 –Redline Pre-Process

Objectives

- Set up Mandiant Redline to conduct a memory audit in preparation for the memory analysis exercise.
- If you do not want to complete this step, a pre-processed version of the memory image will be provided in the next exercise.

Exercise Preparation

1. *Install RedLine (Microsoft Windows only tool)*
   - Located on your USB under `\SIFT-Lab-Install\redline`
   - Redline requires a recent version of the .NET framework, you might need to install .NET before the installation. The installed for the FULL version of .NET is found in `\SIFT-Lab-Install\redline`

2. Execute Redline from the Windows Start menu (or equivalent)

3. Attach your FOR508 USB media and find the **`xp-tdungan-memory-raw.001`** image (it should be located in **`\xp-tdungan-10.3.58.7\xp-tdungan-memory\`**)

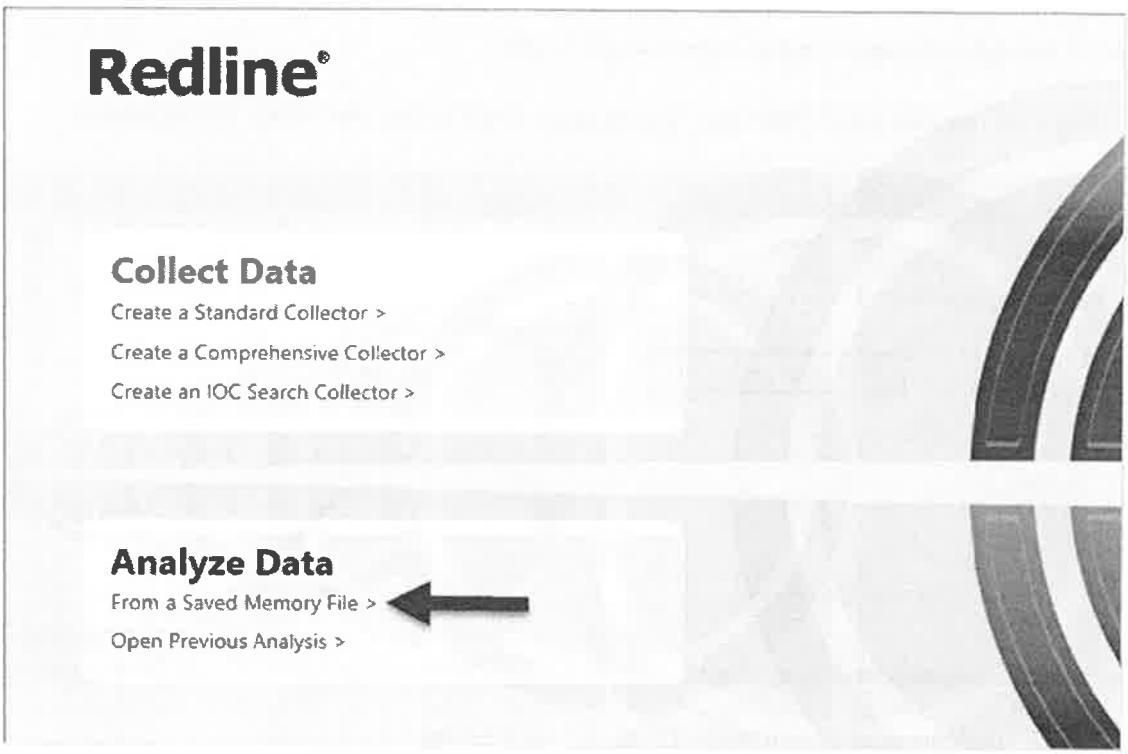
4. Copy the **`xp-tdungan-memory-raw.001`** file to a local drive (i.e. **`C:\Users\<username>\Desktop`**)

Exercise – Step-by-Step Guide

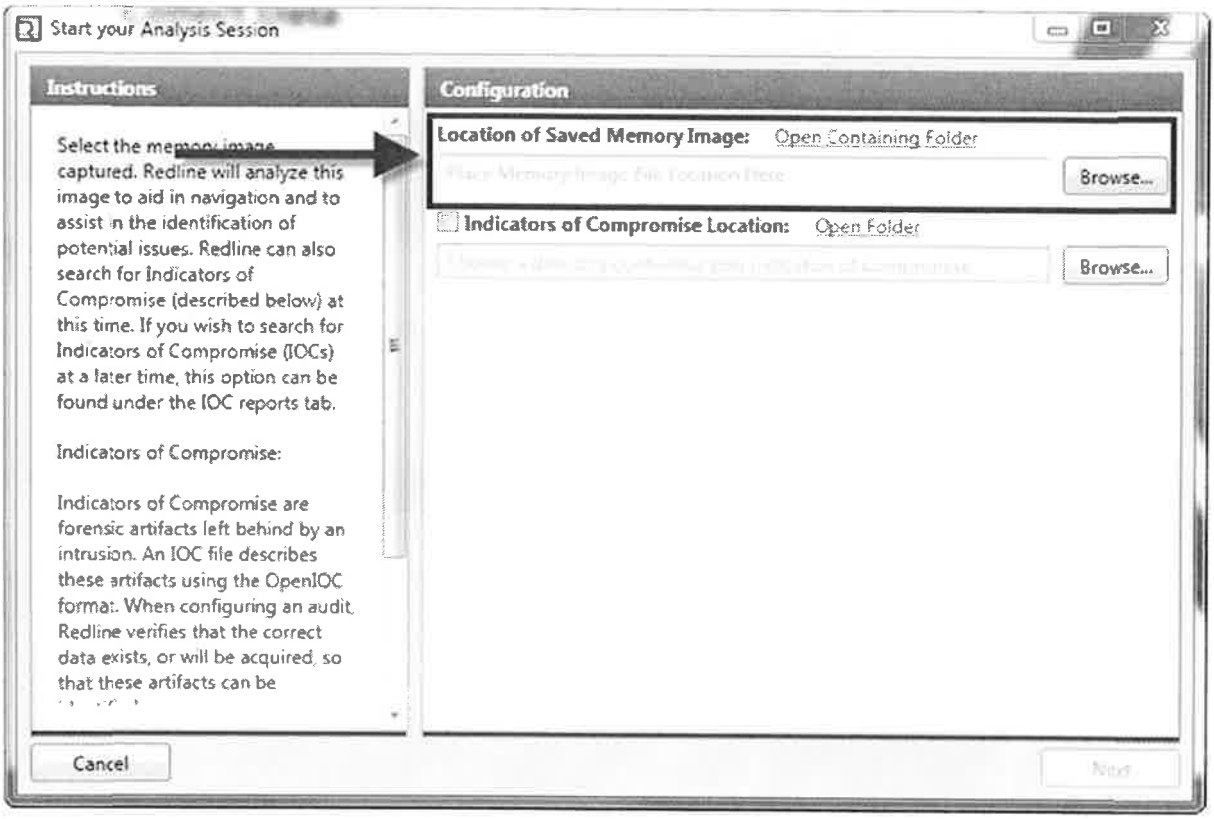
NOTE: Some older laptops and VM setups may take a *very* long time to process this memory image (VMware Fusion is notoriously bad). If your system does not finish processing, **a pre-processed analysis file has been saved on your USB in the "`xp-tdungan-10.3.58.7\xp-tdungan-c-drive\precooked\redline`" folder.** Instructions for loading the pre-processed copy are present in the next exercise.

Perform a "Saved Memory File" analysis of the **`xp-tdungan-memory-raw.001`** image.

1. Analyze Data
   a. Select "From a Saved Memory File"

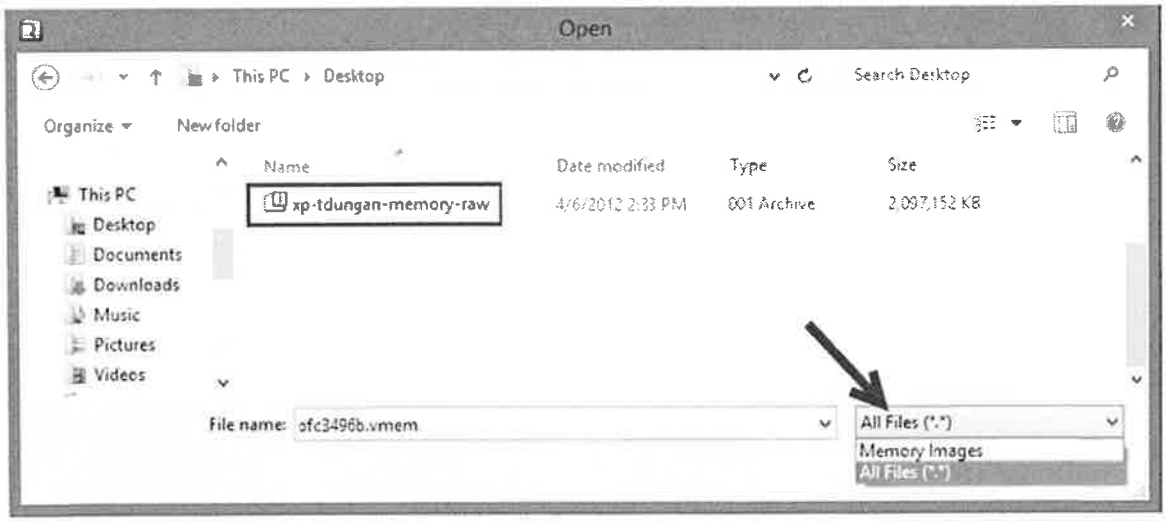


2. Find the "Location of Saved Memory Image" dialog



3. Browse to the **`xp-tdungan-memory-raw.001`** file

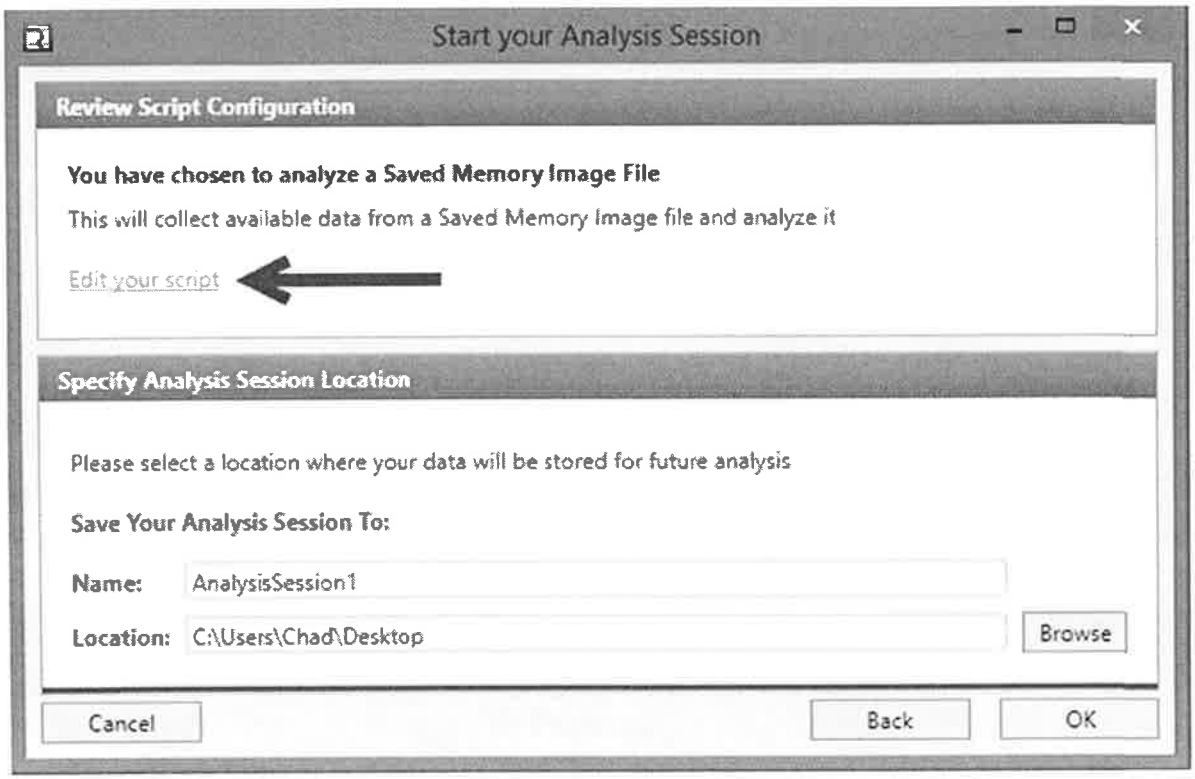
   a. You may need to select "All Files" on the drop down to see the ".001" file extension.



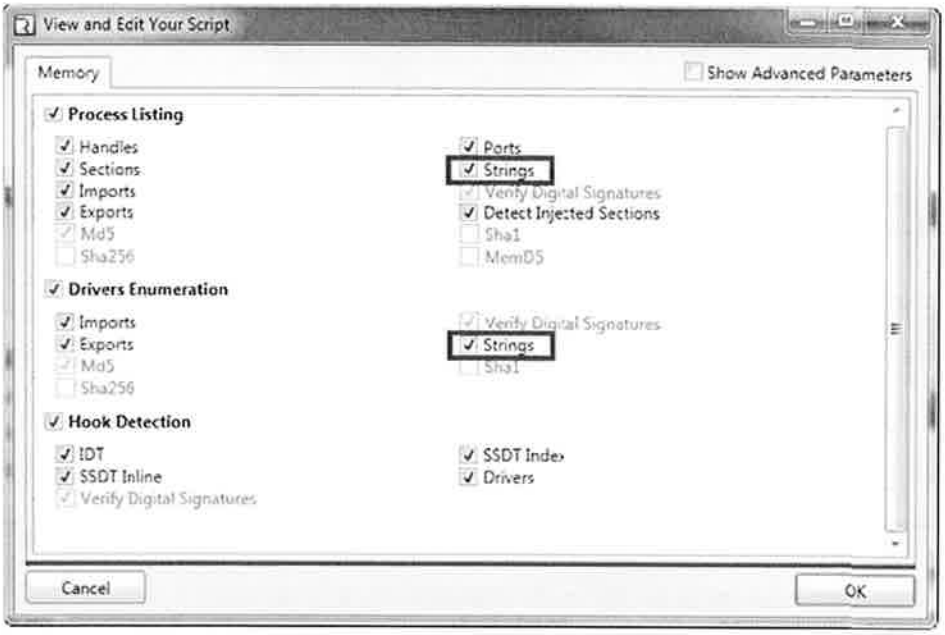
   b. Leave "Indicators of Compromise Location" unchecked.

4. Click "Next" in the bottom right corner.

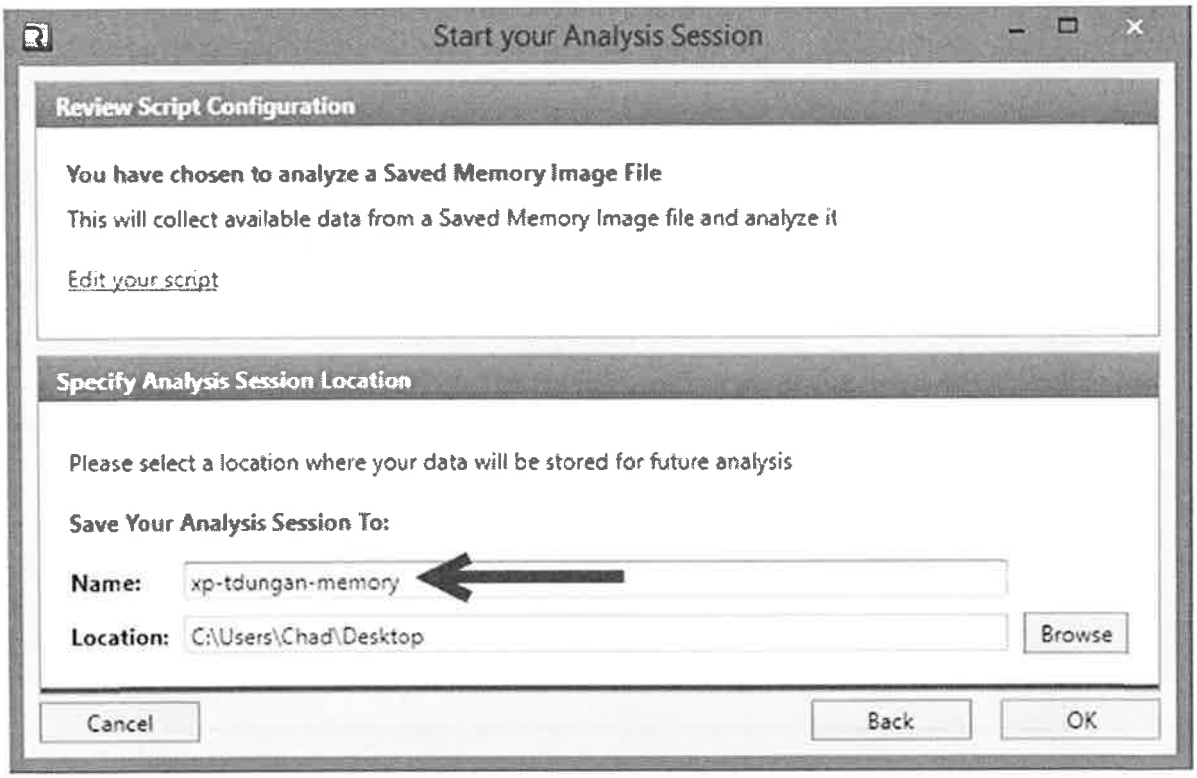
5. Select "Edit your script" to choose analysis options



6. Check "Strings" in both the **Process Listing** and **Drivers Enumeration** sections. Strings collection is one of the most time intensive actions. Note that some options will be unavailable (greyed out), due to this not being a live analysis. Click "Ok".



7. Select **`Save Your Analysis Session To`**: Please **rename** the save location to a directory and name you will remember. **`xp-tdungan-memory.mans`** will work fine saved to your **`C:\Users\<username>\Desktop`**. Click "Ok".



8. Click through the Windows User Access Control (UAC) permission dialog if it appears.

9. When Redline completes processing, you will be ready to analyze!

This page intentionally left blank.

Exercise 8 – Memory Analysis with Redline

Objectives

- Learn memory forensics using the prescribed memory analysis steps.
- Gain experience using Redline to analyze data present in memory structures.
- Witness the value of string searching in Windows processes.

Exercise Preparation

1. *Install RedLine (Microsoft Windows only tool)*
   - Located on your USB under `\SIFT-Lab-Install\redline`
   - Redline requires a recent version of the .NET framework, you might need to install .NET before the installation. The installed for the FULL version of .NET is found in `\SIFT-Lab-Install\redline`

2. Perform a full memory audit of the **`xp-tdungan-memory-raw.001`** image, including process and driver strings output (see Optional Redline Pre-Process exercise).

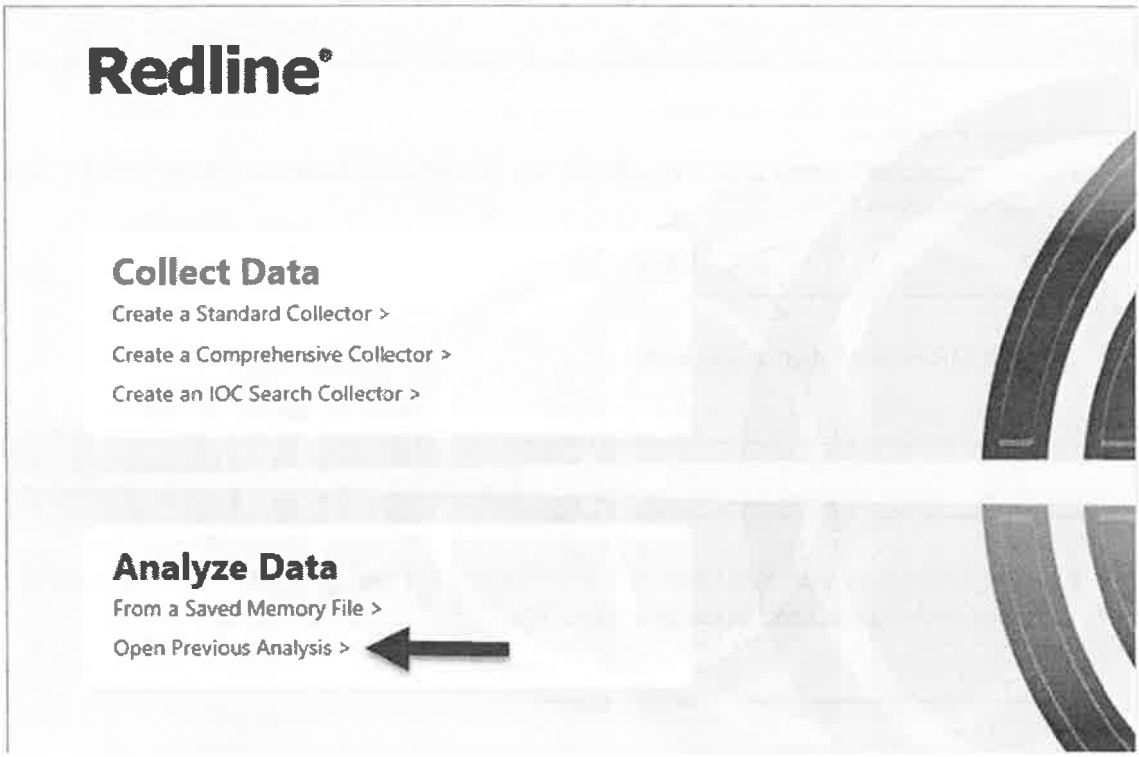
NOTE: Some older laptops and VM setups may take a *very* long time to process this memory image (VMware Fusion is notoriously bad). If your system is still pre-processing at the start of this exercise, cancel processing. **A pre-processed analysis file has been saved on your USB in the "`xp-tdungan-10.3.58.7\xp-tdungan-c-drive\precooked\redline`" folder**

3. If you are using the pre-cooked image (i.e. you did not complete the "Optional Redline Pre-Process" exercise), do the following:

   - Copy the "**`xp-tdungan-10.3.58.7\xp-tdungan-c-drive\precooked\redline`**" folder from the course USB to a local drive (your desktop is one good option).

   - Double-click the **`xp-tdungan.mans`** file located in the folder just copied

4. If for some reason double-clicking the **`xp_tdungan.mans`** file does not work, it can also be opened directly via the "Open Previous Analysis" option in the main Redline menu.



Mount your disk image so you can interact with files/folders on the xp-tdungan-c-drive

5. If your evidence is not already mounted in the SIFT workstation (from a previous exercise), you will need to open a terminal in your SIFT workstation, elevate your privileges to root, and change into the **`/cases/xp-tdungan-c-drive`** directory. Then uncompress and mount your evidence files so you can see the **`xp-tdungan-c-drive`** file system in /mnt/windows_mount.

```
$ sudo su -

# cd /cases/xp-tdungan-c-drive/

# ewfmount xp-tdungan-c-drive.E01 /mnt/ewf_mount

# mount -o ro,loop,show_sys_files,streams_interface=windows
/mnt/ewf_mount/ewf1 /mnt/windows_mount
```

Exercise – Questions

1. **Identify Rogue Processes**
   - Review processes by MRI scores to identify any suspicious processes. What process has the highest MRI score?

     svchost.exe pid: 3296

   - Why did MRI score this process so high?

     wrong path

     parent not services.exe / run from user AK

   - Was the process executed at boot? (Hint: compare to the start times of smss.exe, csrss.exe, lsass.exe, svchost.exe, and other processes that typically start at boot)

     No

   - What user account (SID) was logged on when the process was spawned?

     1004

   - List the full path of the process binary:

     c:\windows\system32\lllhost\svchost.exe

     - Review the contents of the folder where this binary is located within the tdungan disk image (during the Exercise Preparation section you should have previously mounted tdungan's disk image to /mnt/windows_mount).

       What additional file is located in the same suspicious directory?

       winclient.reg

       Open the newly discovered file in gedit (SIFT) or Notepad (Windows). What registry key is referenced?

       HKEY_Local_Machine\system\currentcontrolset\services\NullmanU
       domain

- View the hierarchical process list. Do you see any additional suspicious processes?

spinlock.exe pid:12244

svchst.exe pid:1488

- Why is information missing on some of the interesting processes?

2. Analyze Process DLLs and Handles

- Review Handles of each suspicious process identified in Question 1.
  - Use least frequency of occurrence (LFO) to narrow your search. Are there any handles worth looking into further?

The reg, File, Mutant's Handle of

pID: 3296.

- List the loaded DLLs for each suspicious process and sort by occurrence using the MRI Report tab in "Full Detailed Information" (found via a double-click on any process name)
  - What DLLs (if any) might call for further analysis?

netmsg.dll.

3. Review Network Artifacts

- Do any network artifacts appear suspicious? What process spawned them?

no

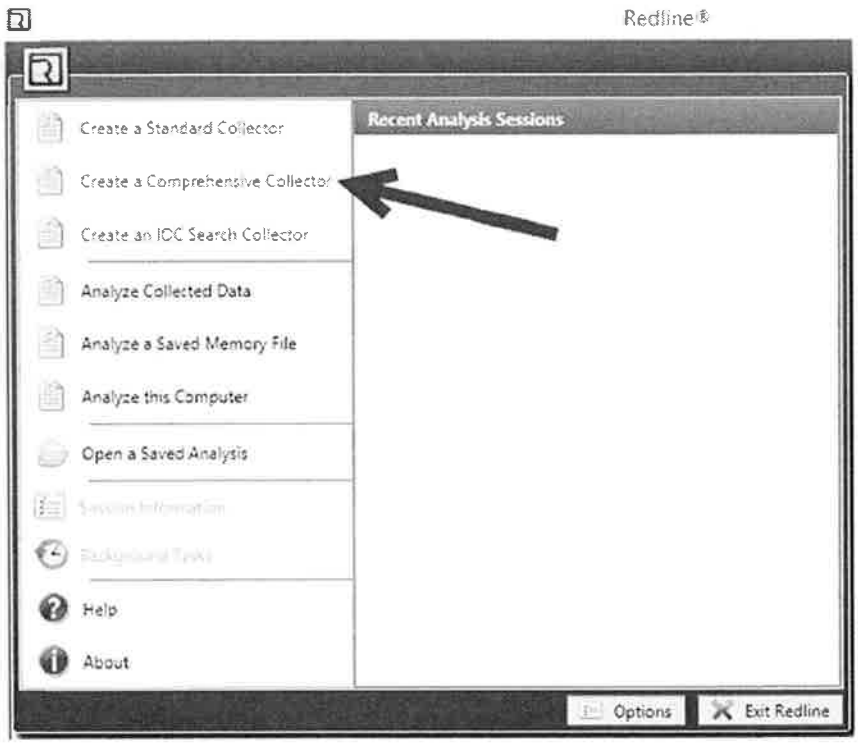
4. **Use the Redline "Strings" output to look for additional evidence**
   - Search svchost.exe (PID 3296).
     - Be creative. Terms like "`http://`", "`https://`", "`ftp://`", "`.exe`", "`C:\`", and names of suspected malware or support files can sometimes bear fruit.

   - Search csrss.exe (PID 976)
     - The CSRSS process is the Client/Server Run-Time Subsystem and in Windows XP it is responsible for the console and processes such as `cmd.exe`. Look for any command line entries by searching for "`cmd.exe`" and other command-line executable names.

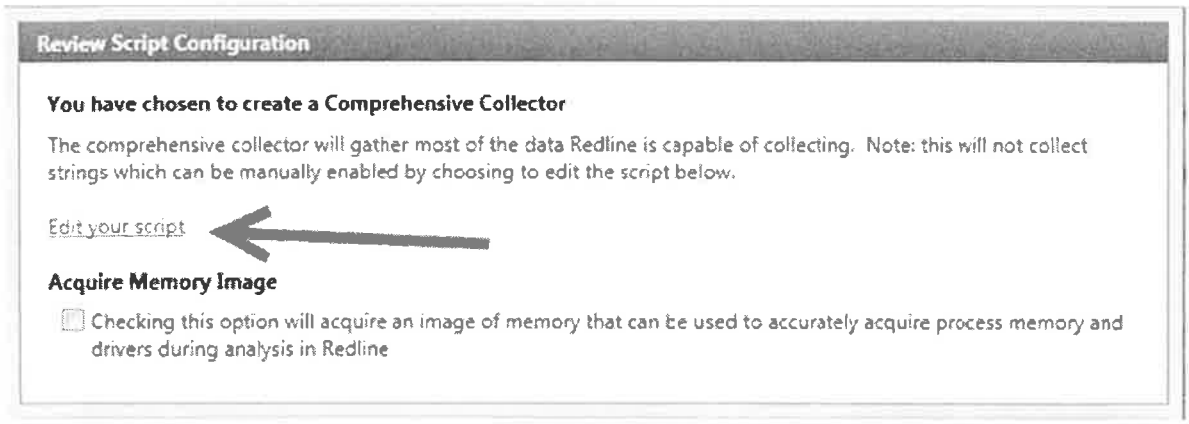
search in string

5. **Optional Homework: Create and run your own Redline Collector**

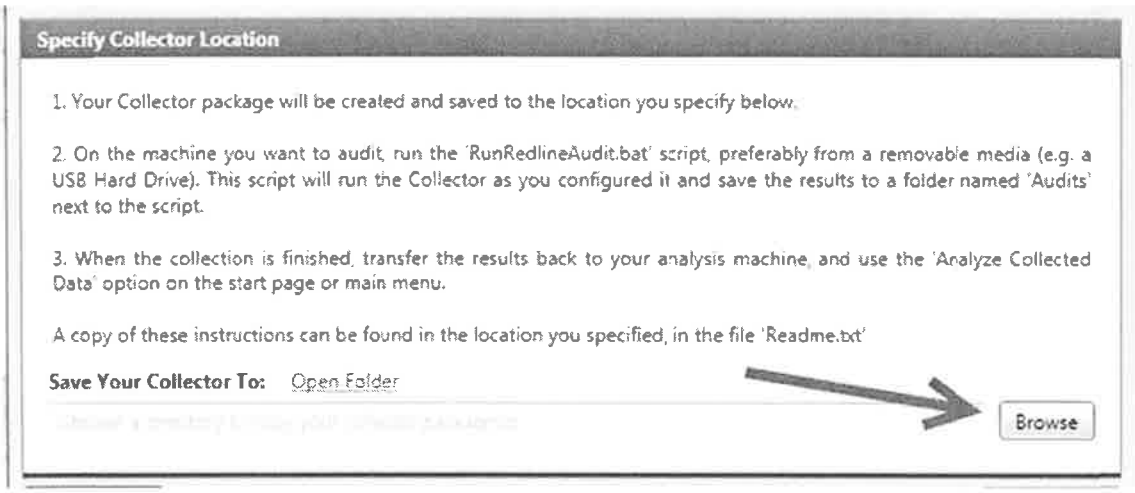
   1) <Top Left Icon> → Create a Comprehensive Collector

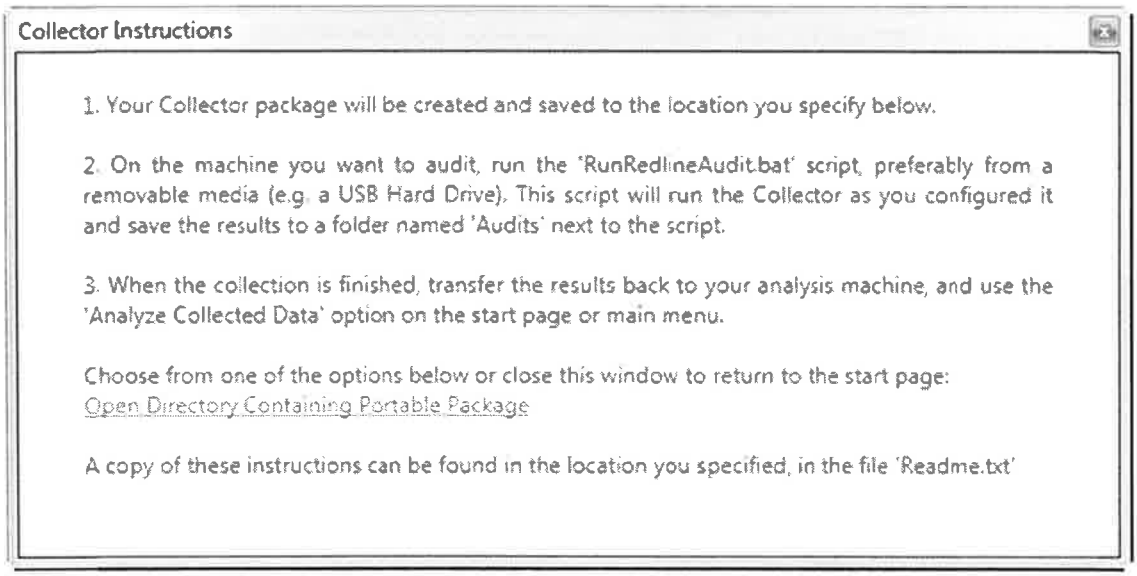


   2) Edit your script
      a. Go through the various menus and select what you would like the script to collect. Since we selected "Comprehensive Collector", most options are already enabled. Any additional selections will likely require a long time to complete.
      b. If you would also like the collector to take a full memory image, select Acquire Memory Image

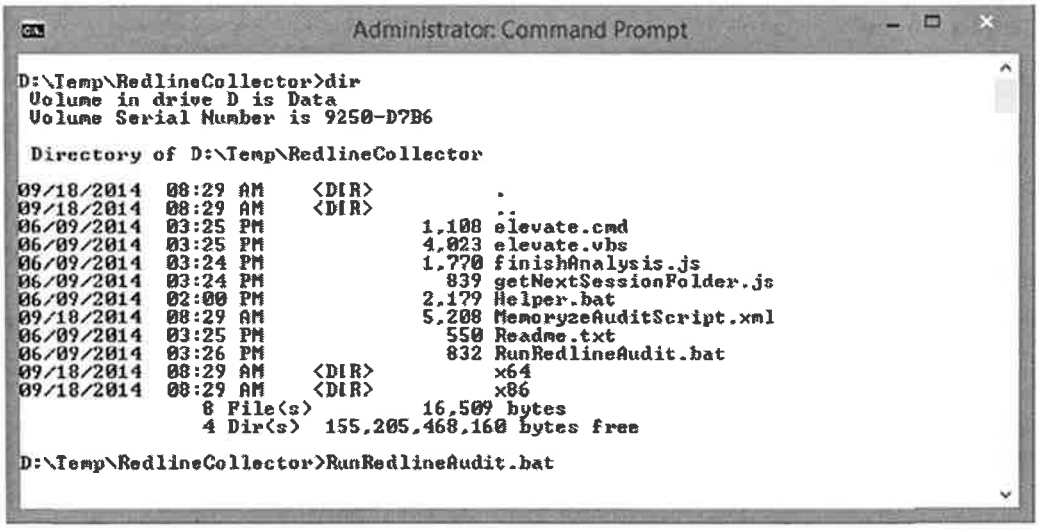


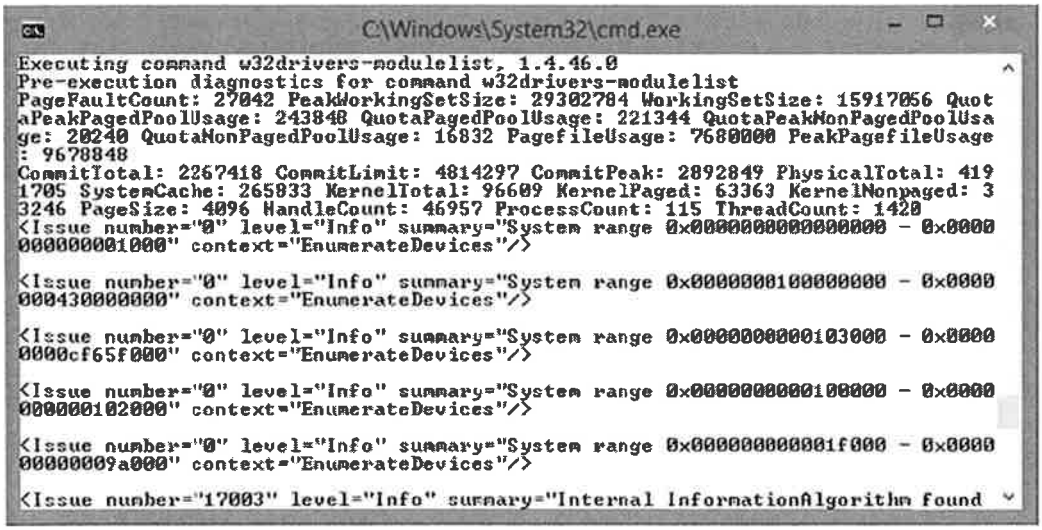
3) Specify Collector Location
    a. Place this on a USB thumbdrive if you have one, or for testing purposes you can save and run the collector from a local system drive.



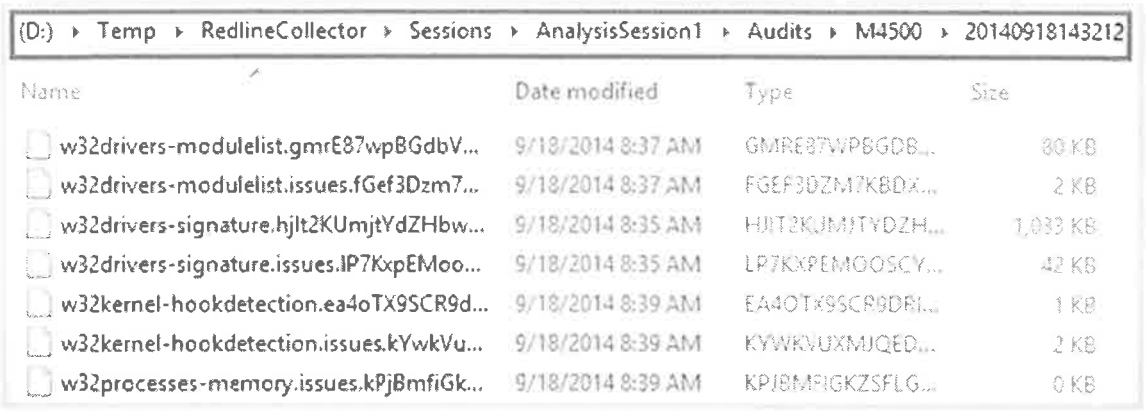


4) Open a command prompt with Administrator permissions on your target system and traverse to the location of your Redline Collector. Execute the RunRedlineAudit.bat script. A new command shell will be open and you should see output from the collectors being executed.

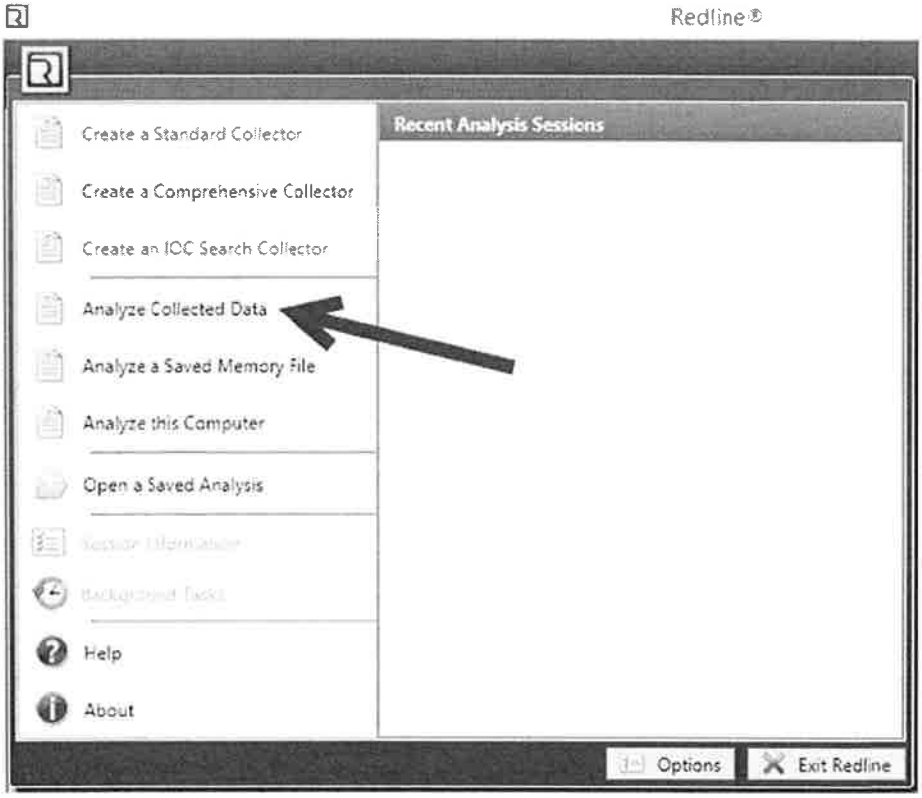




a. A full collection script can take hours to run (in the future you can remove unnecessary options in the audit script to speed up processing). Your results will be located in a folder named according the System Hostname in the same location as your collector.



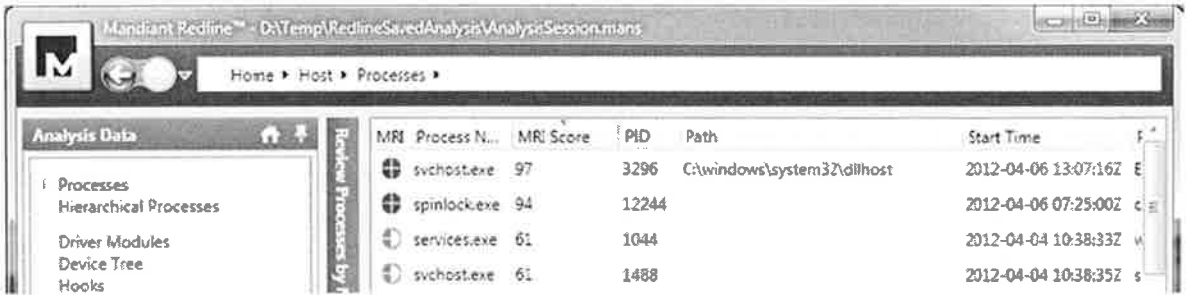
5) Open your results in Redline and start your analysis
   a. <Top Left Icon> → Analyze Collected Data → Audit Location → <sub-folder of folder named according to System Hostname> (or just double click the **`.mans`** file in the same folder)



Exercise – Questions With Step-by-Step

1. **Identify Rogue Processes**
   - Review processes by MRI scores to identify any suspicious processes. What process has the highest MRI score?

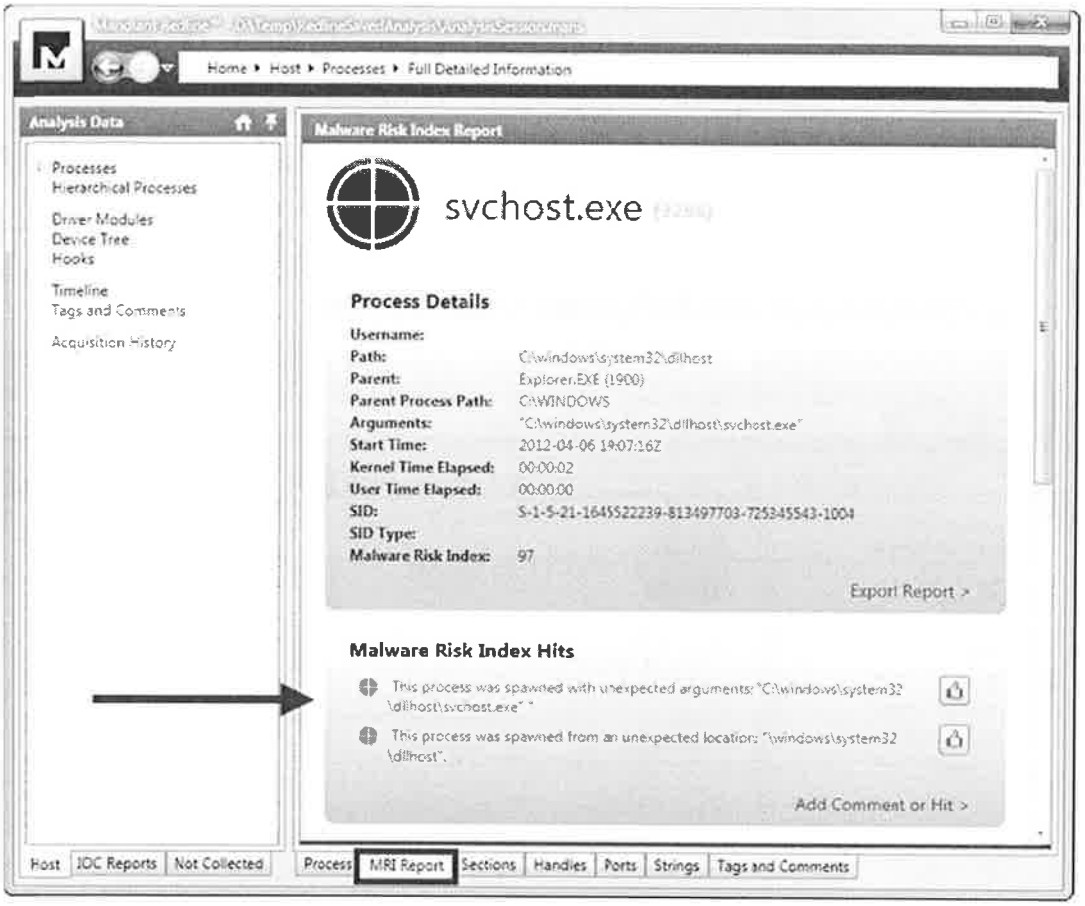
     svchost.exe (PID 3296)



   - Why did MRI score this process so high?

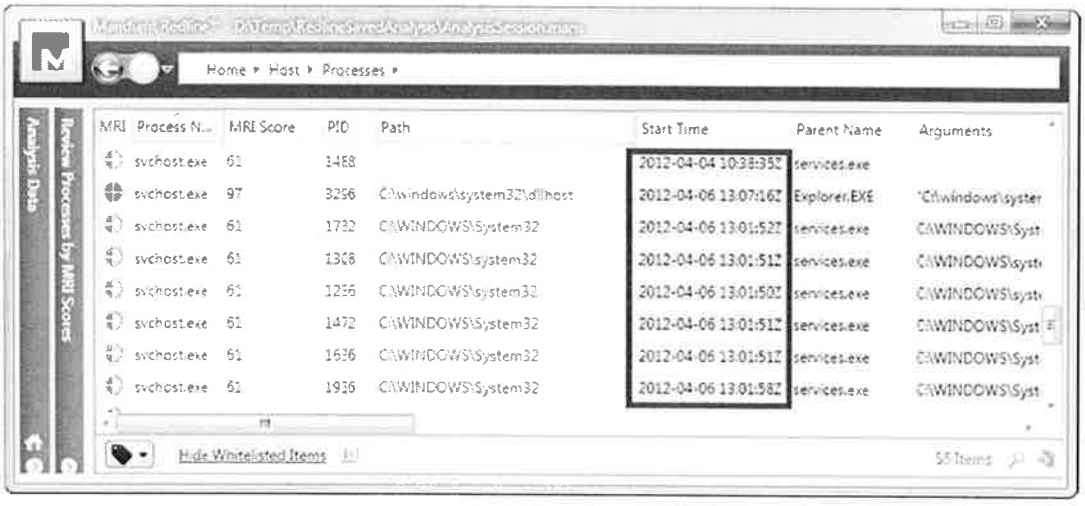
     Double-click on the process name and select "MRI Report"

     Answer: Unexpected location and arguments



- Was the process executed at boot? (Hint: compare to the start times of smss.exe, csrss.exe, lsass.exe, svchost.exe, and other processes that typically start at boot)

Compare the Start Time of svchost.exe PID 3296 with the other svchost processes. It appears to have started >5 minutes after (notice there is one svchost process from 4/4/2012 which was likely a previous boot cycle). Also note that PID 3296 was started not long after the "winlogon.exe" process. This could be an indication the malicious process was started when a user logon occurred.



- What user account (SID) was logged on when the process was spawned?

  The account assigned SID `S-1-5-21-1645522239-813497703-725245543-1004`

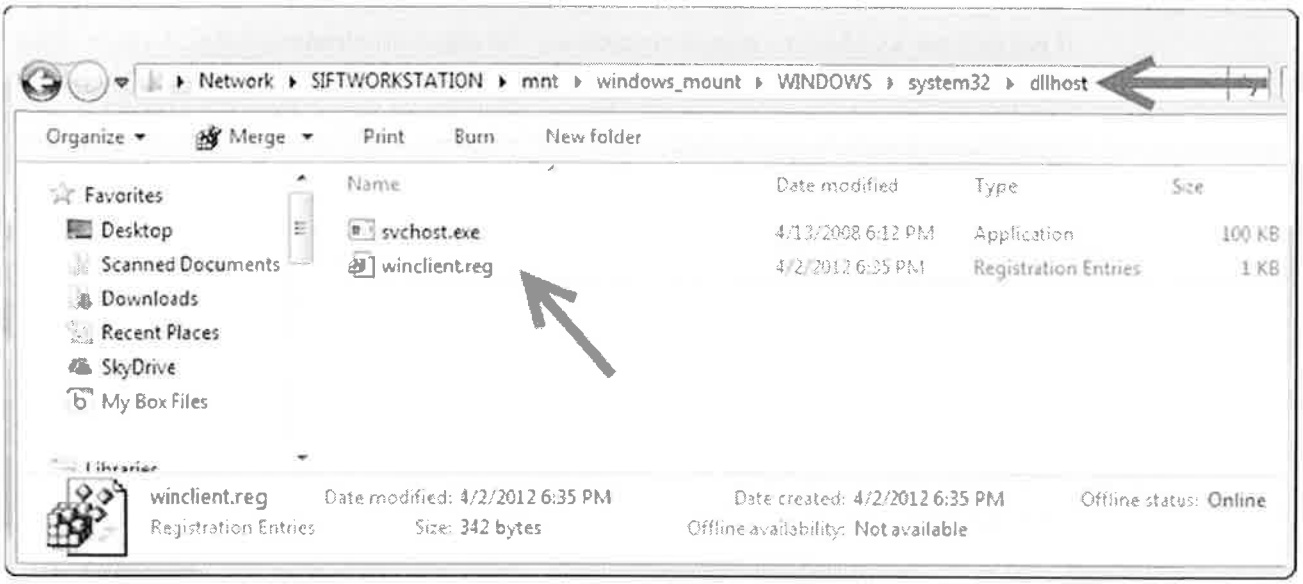
- List the full path of the process binary:

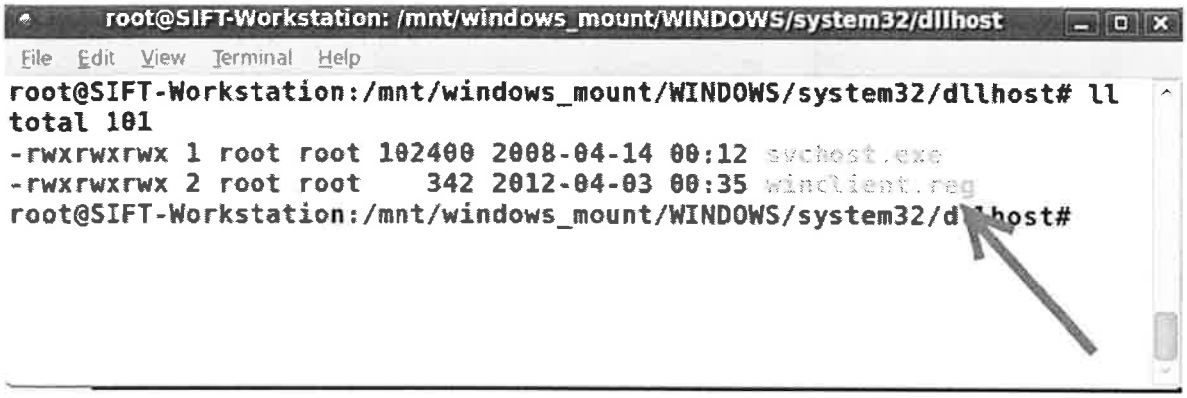
  `C:\windows\system32\dllhost\svchost.exe`

- Review the contents of the folder where this binary is located within the tdungan disk image (during the Exercise Preparation section you should have previously mounted tdungan's disk image to /mnt/windows_mount).

  What additional file is located in the same suspicious directory?

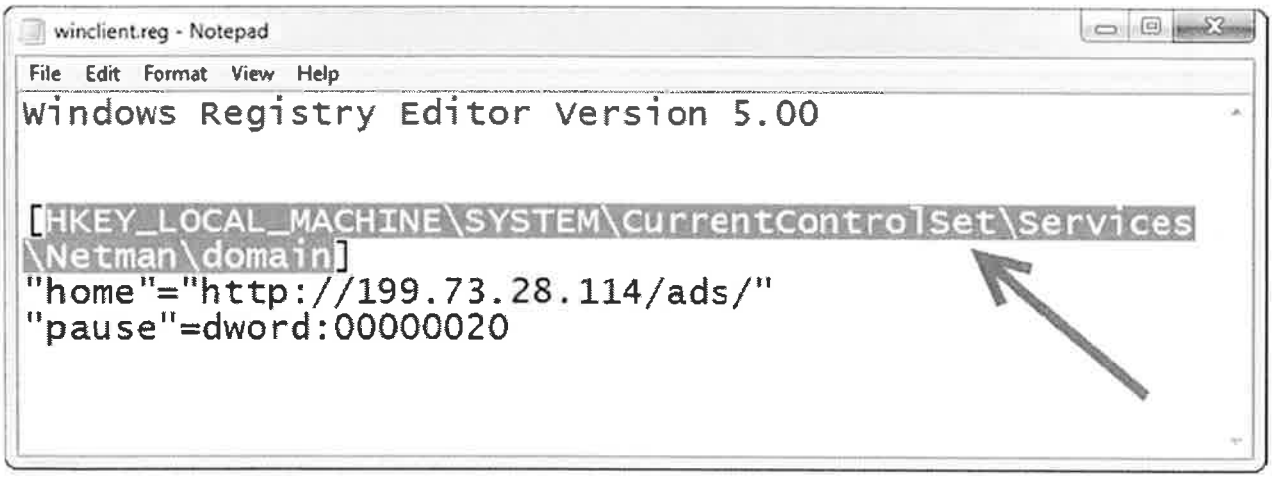
  winclient.reg





Open the newly discovered file in gedit (SIFT) or Notepad (Windows). What registry key is referenced?

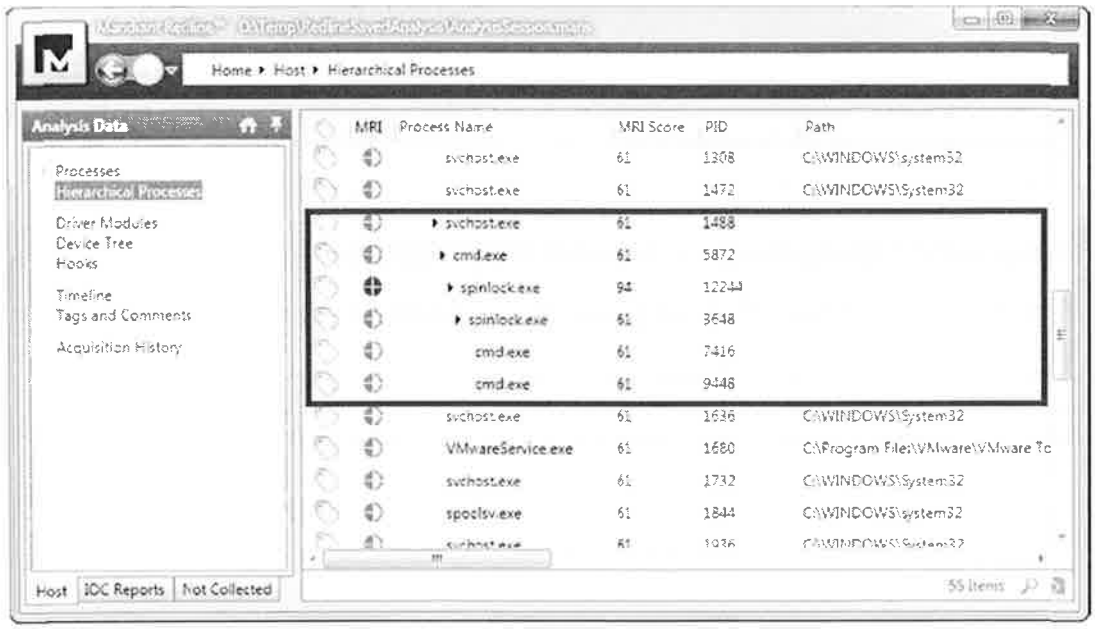
HKLM\SYSTEM\CurrentControlSet\Services\Netman\domain



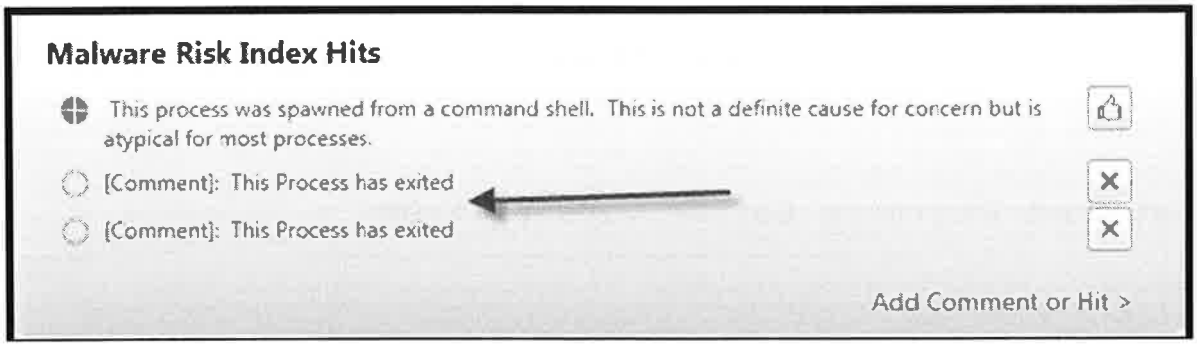
If we believe `svchost.exe` is suspicious, the close proximity of `winclient.reg` also makes it suspicious. While we do not yet know the significance of the registry key and IP address information stored within `winclient.reg`, they may later become useful indicators of compromise.

- View the hierarchical process list. Do you see any additional suspicious processes?

There are two instances of "`spinlock.exe`" and two "`cmd.exe`" command prompts spawned by the second "`spinlock.exe`" process.



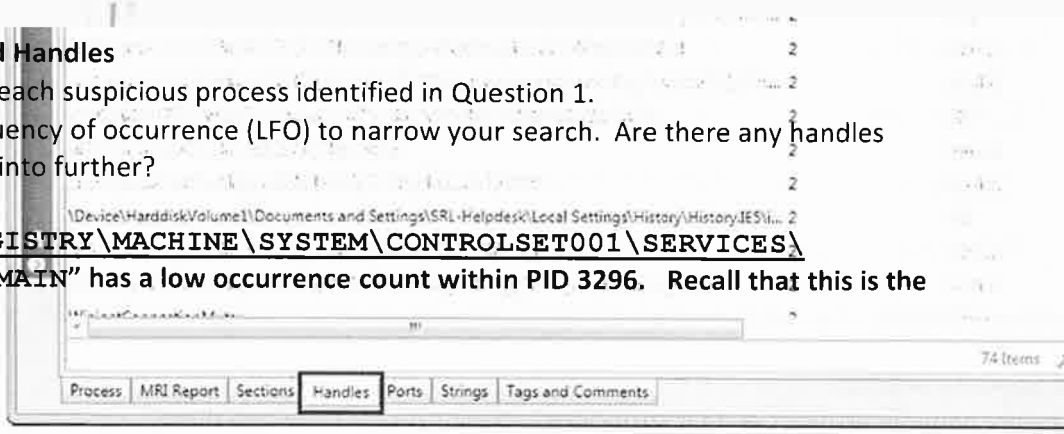
- Why is information missing on some of the interesting processes?

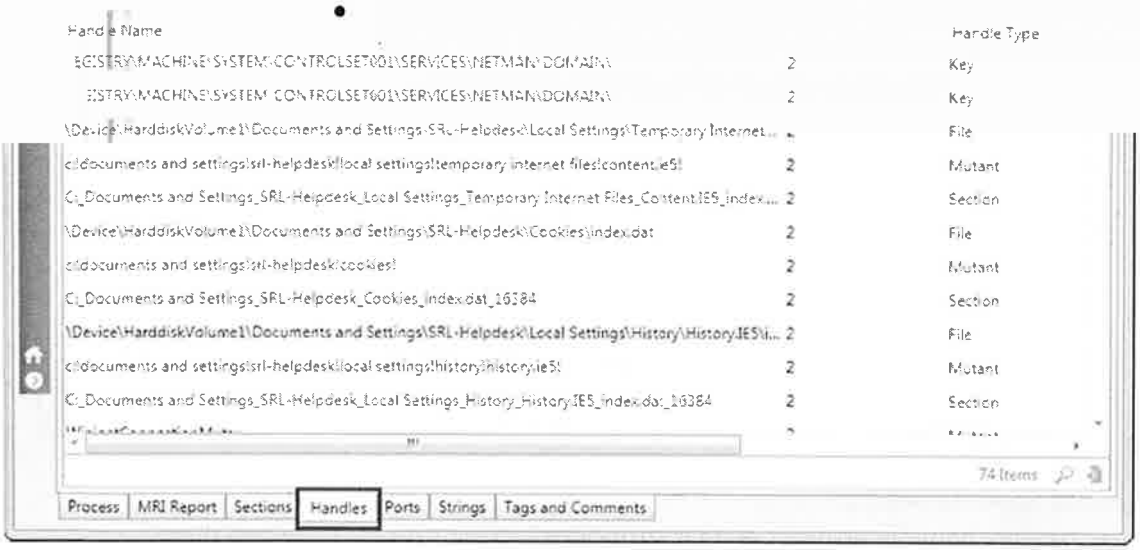


Several of the processes exited prior to the memory acquisition

2. **Analyze Process DLLs and Handles**
   - Review Handles of each suspicious process identified in Question 1.
     - Use least frequency of occurrence (LFO) to narrow your search. Are there any handles worth looking into further?

       The key "REGISTRY\MACHINE\SYSTEM\CONTROLSET001\SERVICES\NETMAN\DOMAIN" has a low occurrence count within PID 3296. Recall that this is the

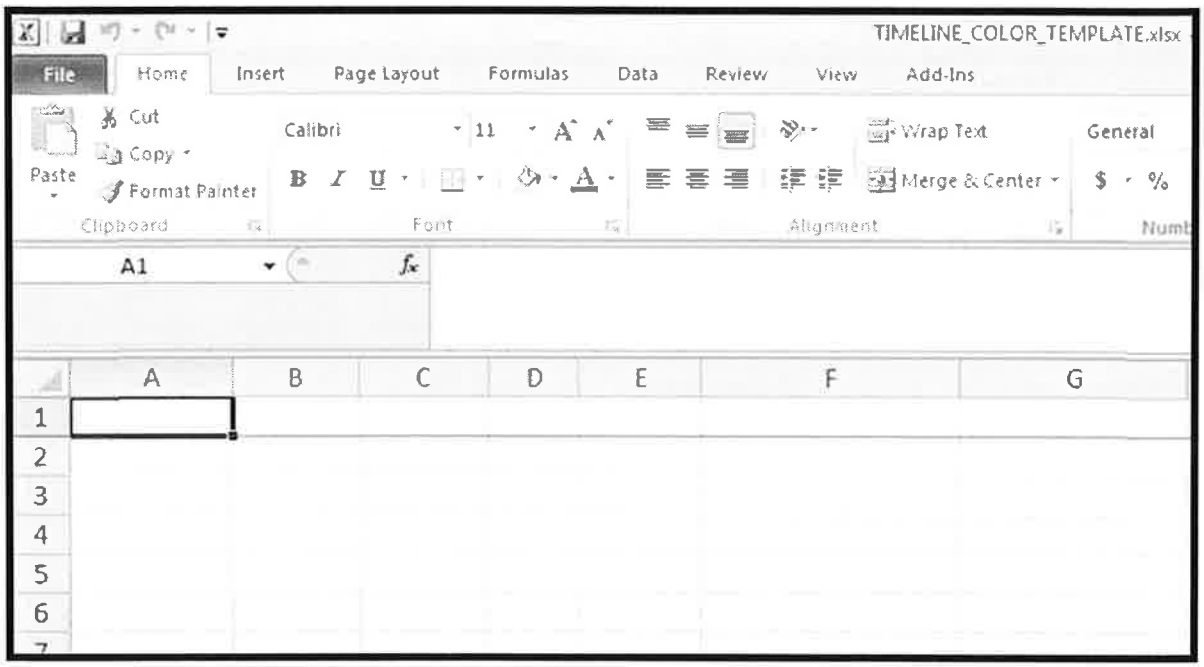




   - List the loaded DLLs for each suspicious process and sort by occurrence using the MRI Report tab in "Full Detailed Information" (found via a double-click on any process name)
     - What DLLs might call for further analysis?

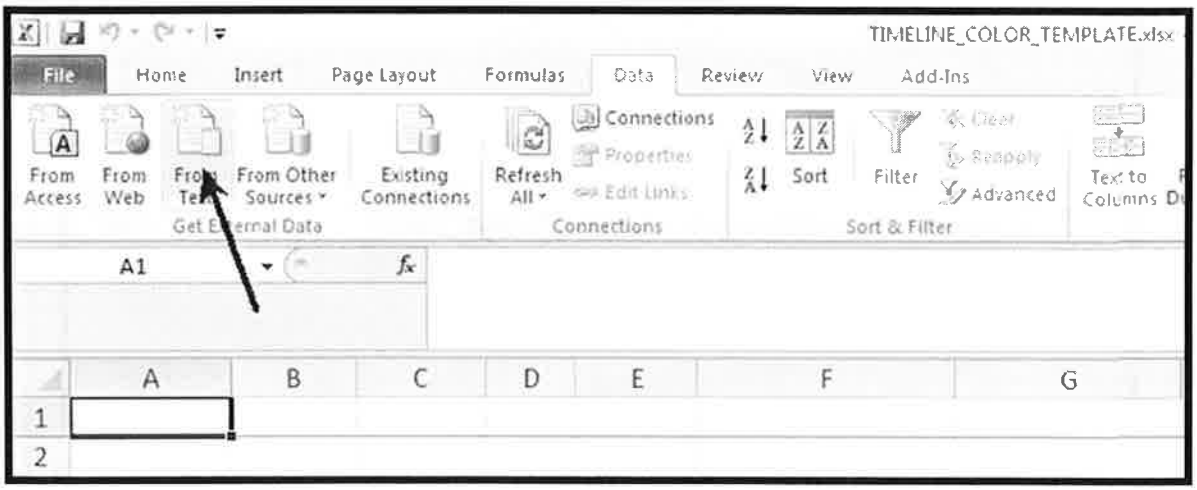
       \Windows\System32\netmsg.dll could be worth checking due to its very low occurrence count (it is only referenced by this process in all of memory). Also notice the many injected memory sections shown in this report view.

c. Click on Cell A-1.

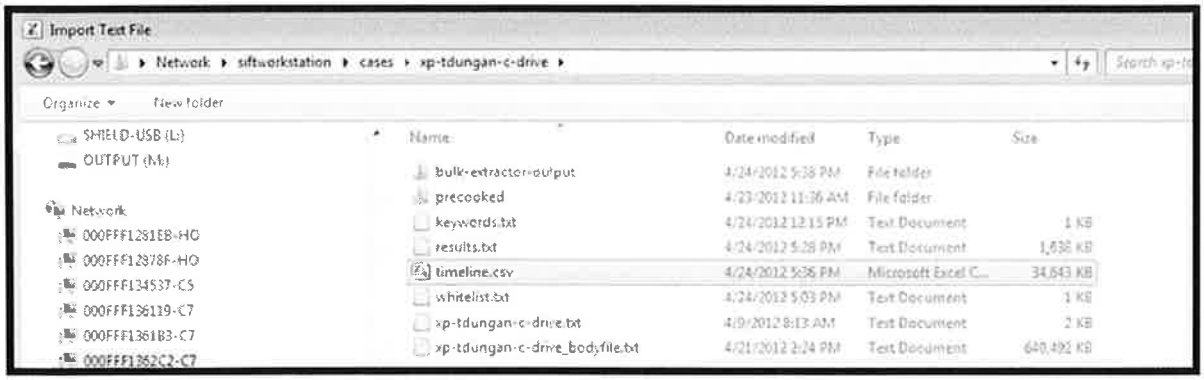


d. Select 'DATA' Ribbon.

e. Import Data "FROM TEXT".

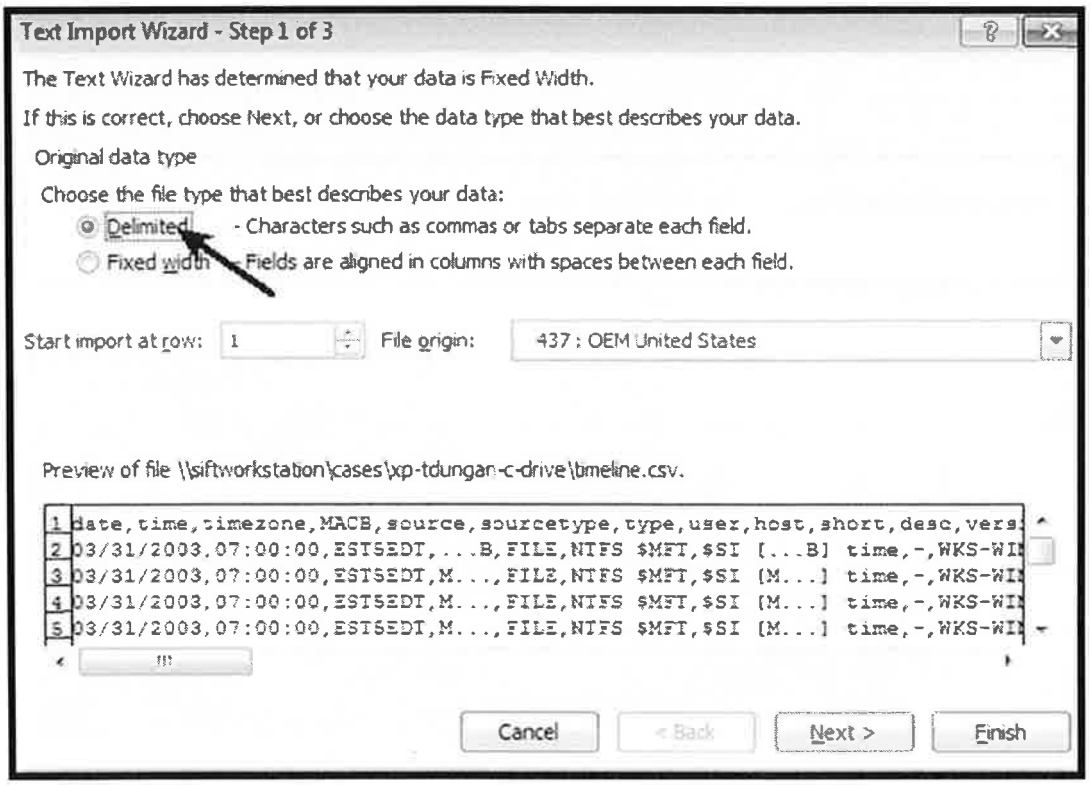


f. Select **timeline.csv** file ->**\\siftworkstation\cases\xp-tdungan-c-drive**

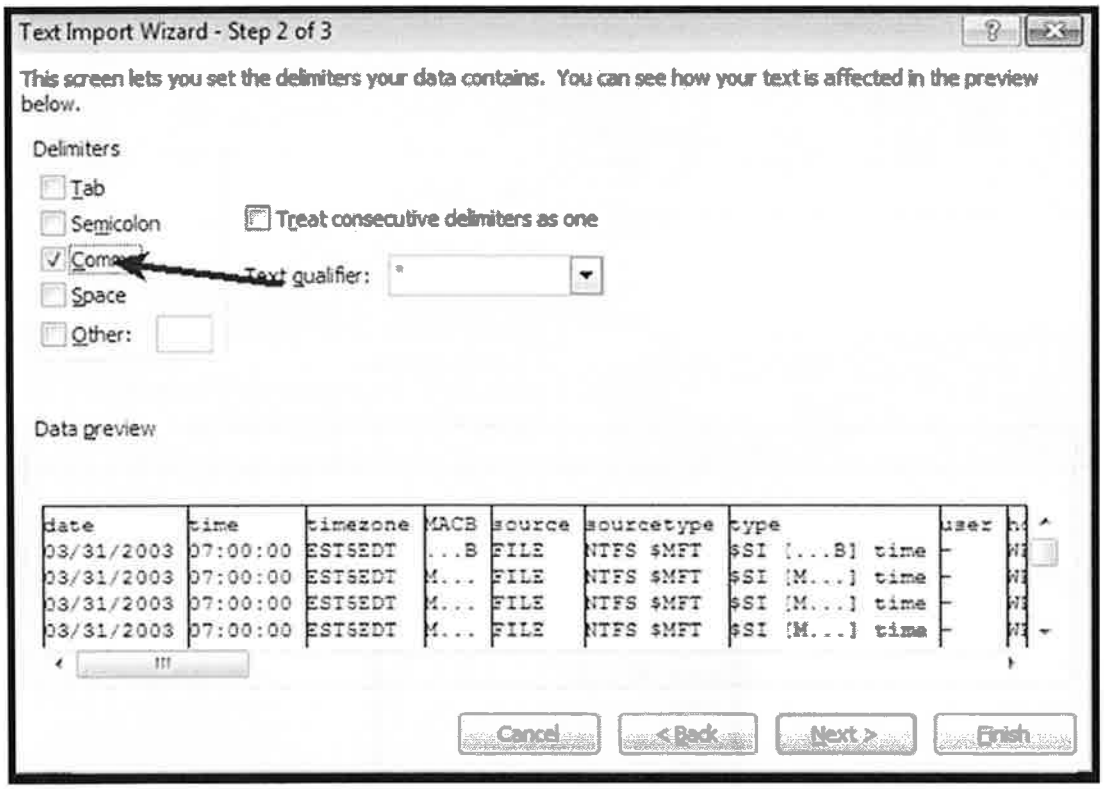


g. TEXT IMPORT WIZARD Will Start.

h. Step 1 -> Select Delimited -> Select NEXT.

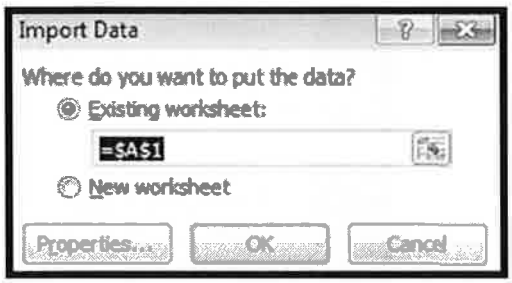


i. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT >

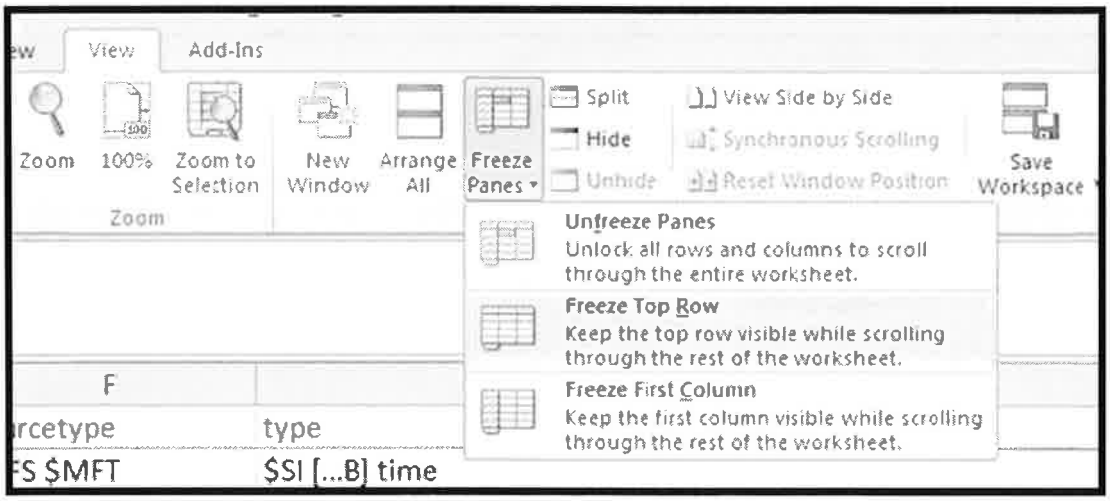


j. Step 3 ->Select Finish.

k. Where do you want to put the data? Simply Select OK.

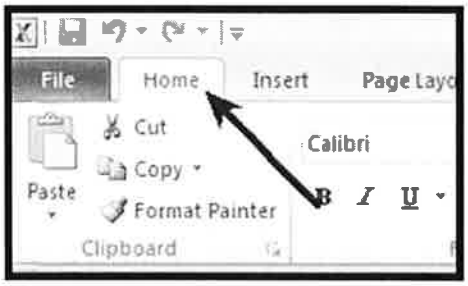


l. Once imported View -> Freeze Panes -> Freeze Top Row.



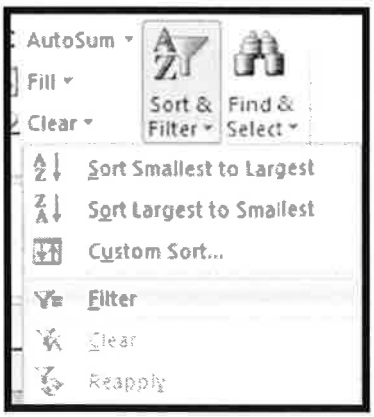
m. Optional: Hide Columns Time Zone, Host, Version.

n. Select HOME Ribbon.



o. Select all Cells "CTRL-A".

p. In Home Ribbon -> Sort and Filter – Filter and you will be ready to begin analysis.



q. Before you analyze too much – please save your new Color Timeline as an XLSX file:
`/cases/xp-tdungan-c-drive/supertimeline.xlsx`

Takeaways

- Log2timeline is not a quick process to run.
- Compare length of time to create a super timeline vs. a file system timeline.
- If you are in a rush, could you begin basic analysis with the much quicker file system timeline?

Exercise 14 – Super Timeline Analysis

Objectives

- Conduct a full system super timeline analysis of a compromised system
- Identify when the hacker/intruder was first active on the system and where they possibly went
- Gain experience analyzing the colorized timeline output
- Gain experience performing contextual based analysis of artifacts

Exercise Preparation

1. Please ensure your evidence is mounted correctly at **`/mnt/windows_mount`** if not, please follow **Exercise 2 – Mounting Evidence Using SIFT** again to mount your evidence.

```
# cd /mnt/windows_mount

# ls
```

```
root@SIFT-Workstation:/mnt/windows_mount# ls
$AttrDef        Documents and Settings  MSOCache        System Volume Information
AUTOEXEC.BAT    $Extend                 NTDETECT.COM    Temp
$BadClus        hiberfil.sys            ntldr           $UpCase
$Bitmap         IO.SYS                  pagefile.sys    $Volume
$Boot           $LogFile                Program Files   WINDOWS
boot.ini        $MFTMirr                RECYCLER
CONFIG.SYS      MSDOS.SYS               $Secure
```

If you do not see the above, then please follow directions in Exercise 2 – Mounting Evidence Using SIFT

Pre-Cooked Supertimeline

If your supertimeline has not finished processing yet, please use the pre-cooked version found in: **`/cases/xp-tdungan-c-drive/precooked/timeline/ XP-TDUNGAN-TIMELINE-FINAL.xlsx`**

Exercise – Questions

1. Initially scan through the timeline for getting a feel for the difference between the filesystem timeline and the super timeline. (5 Minutes Max)

2. Using the same questions from Exercise File System Timeline, see if they would have been easier to answer using the Color Super Timeline. (10-15 Minutes Max) Do you see any additional information that you can discern.

3. What additional artifacts are now clearly visible around the time that winclient.reg and the **`C:\windows\system32\dllhost`** directory were created? What time was the reg.exe executable run? Why was reg.exe executed at this time? What time was the **`\ControlSet001\Services\Netman\domain`** key last modified?

   - 4/2/2012 20:34:26 / 4/2/2012 20:36:03 / 4/2/2012 20:37:14

4. Near the time that winclient.reg was imported on the system using the reg.exe command, we can see an interesting event at McLogEvent. What time was this event detected? Why do you think this event is meaningful for the case?

```
McLogEvent/257;Info;The Scan was unable to scan password protected
file 2011-W2.zip/2011-W2.pdf. Scan engine version used is 5400.1158
DAT version 6498.0000.
```

   - 4/2/2012 20:28:17
   -
   -
   -

5. In many cases, an easy way to find adversary footprints on your network is to identify TTPs that they use. The use of the "AT" command is a common advanced adversary technique to help in lateral movement by first uploading a file to remote system and then using the "AT" command to execute it at a predetermined time.

   For the tdungan system, what time was the AT.EXE command first executed? Immediately surrounding that event we see an indication that a remote machine might have been involved. What directory was mounted of the remote machine? What is the IP address of that system? Why do you think the remote system was involved? What is the significance of the remote system?

   - 4/4/2012 13:24:43
   - _____
   - _____
   - _____

6. List all the any other systems remotely used through mounting the C$ share. When were the network shares last mounted? Why is it possible that we can see that 10.3.58.5 was mounted twice? Which file and directory did the first entry originate from?

   - _____
   - _____
   - _____
   - _____

7. On 4/4/2012, the user vibranium executed EXCEL.exe, what two files did he open for the first time?

   - _____
   - _____
   - _____
   - _____

   What types of files are those? Where are the LNK files originally located?

```
# cd /mnt/windows_mount/Documents\ and\ Settings/vibranium/Recent/

# exiftool Metal\ Alloy\ List\ Research.lnk
```

- _____
- _____

```
# exiftool Detailed\ Vibranium\ R\&D\ Documents.lnk
```

- _____
- _____

8. When was the FIND.EXE Command executed last? Can you tell what the attacker was possibly looking for? Which directory name were most of those files located in?

- _____
- _____
- _____

9. When was the last **cmd.exe** executed?

- _____

10. When was the last **net.exe** executed? What do you think the remote IP was that net was used against?

- _____

11. Have we seen the combination of cmd.exe, net.exe, 10.3.58.5, and Z: used in any previous analysis prior to the super timeline? What was the full command that was executed based on prior analysis and what can we correlate via timeline analysis?

- _____
- _____
- _____

- _____

12. *EXTRA:* Also in the memory exercise, we saw that pe.exe was one of the artifacts found in the output from csrss that executed `c:\windows\system32\dllhost\svchost.exe` on 10.3.58.5. When was **`pe.exe`** last executed? What do you think the original name of this service is? Are there any other systems that might have been the target of the pe.exe? What is the IP address of that system?

- _____

13. *EXTRA:* Recover the email from the Outlook.pst file that has the 2011-W2.zip file as an attachment. What is the md5 of the message that it was a part of? What is the md5 of the zipfile? What is the IP address of the sender?

```
# cd /mnt/windows_mount/Documents and Settings/tdungan/Local
Settings/Application Data/Microsoft/Outlook

# cp Outlook.pst /cases/xp-tdungan-c-drive/

# cd /cases/xp-tdungan-c-drive

# pffexport Outlook.pst

# cd Outlook.pst.export

# find . | grep zip
```

```
# cd Top\ of\ Personal\ Folders/Inbox/Message00274/

# md5sum Message.txt

# cat Message.txt

# md5sum Message.txt
```

- _____

```
# cd Attachments/

# md5sum 2011-W2.zip
```

- _____

```
# cd ..

# cat InternetHeaders.txt
```

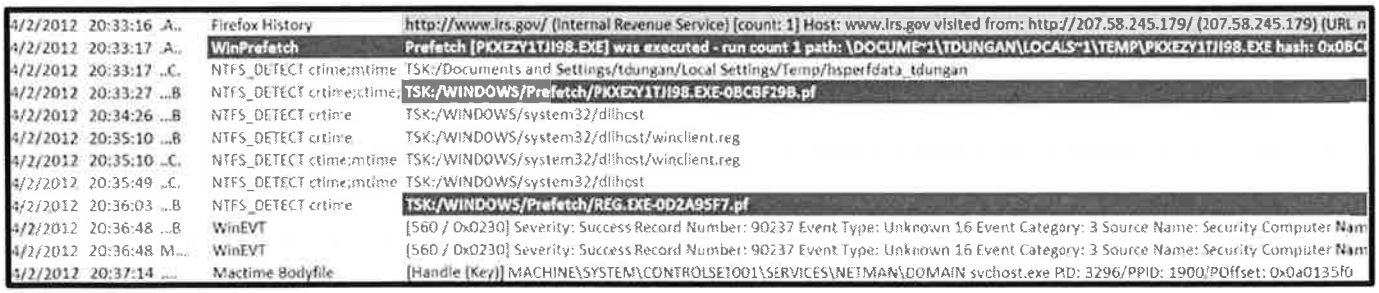
- _____

Exercise – Questions With Step-by-Step

1. Initially scan through the timeline for getting a feel for the difference between the filesystem timeline and the super timeline. (5 Minutes Max)

2. Using the same questions from Exercise File System Timeline, see if they would have been easier to answer using the Color Super Timeline. (10-15 Minutes Max) Do you see any additional information that you can discern?

3. What additional artifacts are now clearly visible around the time that winclient.reg and the `C:\windows\system32\dllhost` directory were created? What time was the reg.exe executable run? Why was reg.exe executed at this time? What time was the `\ControlSet001\Services\Netman\domain` key last modified?



4/2/2012 20:33:16 .A..	Firefox History	http://www.irs.gov/ (Internal Revenue Service) [count: 1] Host: www.irs.gov visited from: http://207.58.245.179/ (207.58.245.179) (URL n
4/2/2012 20:33:17 .A..	WinPrefetch	Prefetch [PKXEZY1TJI98.EXE] was executed - run count 1 path: \DOCUME~1\TDUNGAN\LOCALS~1\TEMP\PKXEZY1TJI98.EXE hash: 0x0BCI
4/2/2012 20:33:17 ..C.	NTFS_DETECT ctime;mtime	TSK:/Documents and Settings/tdungan/Local Settings/Temp/hsperfdata_tdungan
4/2/2012 20:33:27 ...B	NTFS_DETECT crtime;ctime;	TSK:/WINDOWS/Prefetch/PKXEZY1TJI98.EXE-0BCBF29B.pf
4/2/2012 20:34:26 ...B	NTFS_DETECT crtime	TSK:/WINDOWS/system32/dllhost
4/2/2012 20:35:10 ...B	NTFS_DETECT crtime	TSK:/WINDOWS/system32/dllhost/winclient.reg
4/2/2012 20:35:10 ..C.	NTFS_DETECT ctime;mtime	TSK:/WINDOWS/system32/dllhost/winclient.reg
4/2/2012 20:35:49 ..C.	NTFS_DETECT ctime;mtime	TSK:/WINDOWS/system32/dllhost
4/2/2012 20:36:03 ...B	NTFS_DETECT crtime	TSK:/WINDOWS/Prefetch/REG.EXE-0D2A95F7.pf
4/2/2012 20:36:48 ...B	WinEVT	[560 / 0x0230] Severity: Success Record Number: 90237 Event Type: Unknown 16 Event Category: 3 Source Name: Security Computer Nam
4/2/2012 20:36:48 M...	WinEVT	[560 / 0x0230] Severity: Success Record Number: 90237 Event Type: Unknown 16 Event Category: 3 Source Name: Security Computer Nam
4/2/2012 20:37:14	Mactime Bodyfile	[Handle (Key)] MACHINE\SYSTEM\CONTROLSET001\SERVICES\NETMAN\DOMAIN svchost.exe PID: 3296/PPID: 1900/POffset: 0x0a0135f0

- **`C:\WINDOWS\system32\dllhost` was created at 4/2/2012 at 20:34:26 EDT**
- **reg.exe was executed at 20:36:03 -10 seconds – 20:35:53**
- **reg.exe was executed to add the beacon's (svchost.exe) configuration information**
- **`/ControlSet001/Services/Netman/domain` was last modified at 4/2/2012 at 20:37:14**

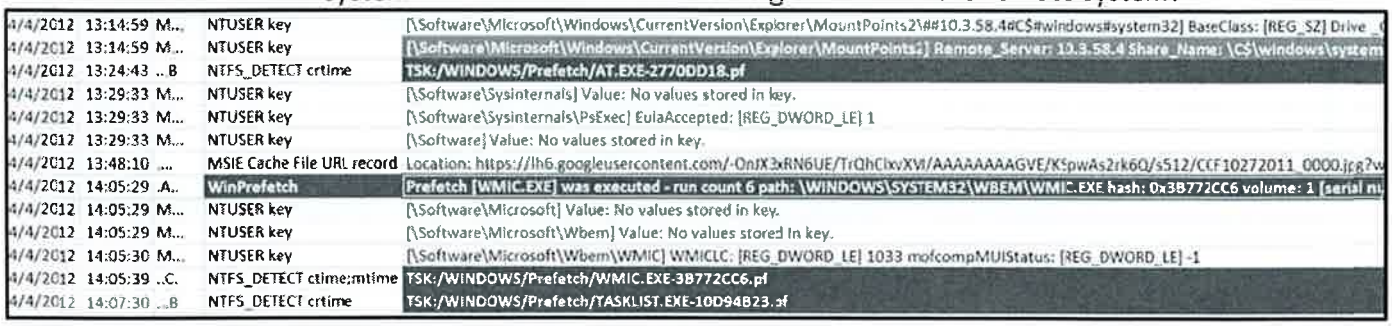
4. Immediately before the winclient.reg was executed on the system using the reg.exe command, we can see an interesting event at McLogEvent. What time was this event detected? What do you think this could mean?

```
McLogEvent/257;Info;The Scan was unable to scan password protected
file 2011-W2.zip/2011-W2.pdf. Scan engine version used is 5400.1158
DAT version 6498.0000.
```

- **PDF's are often used for spearphishing attacks. This could be an indicator that it was in a protected zipfile to protect it from being scanned by McAfee A/V protection.**

5. In many cases, an easy way to find adversary footprints on your network is to identify TTPs that they use. The use of the "AT" command is a common advanced adversary technique to help in lateral movement by first uploading a file to remote system and then using the "AT" command to execute it at a predetermined time.

For the tdungan system, what time was the AT.EXE command first executed? Immediately surrounding that event we see an indication that a remote machine might have been involved. What directory was mounted of the remote machine? What is the IP address of that system? Why do you think the remote system was involved? What is the significance of the remote system?

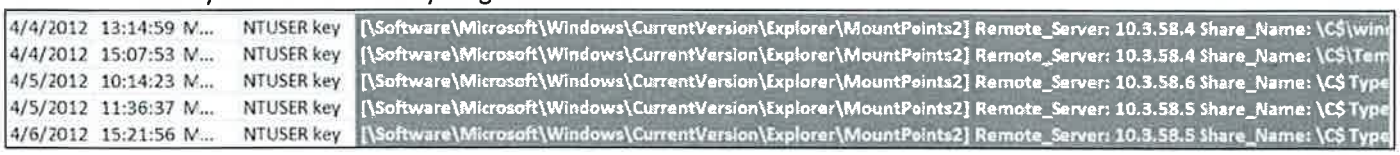


4/4/2012	13:14:59	M...	NTUSER key
4/4/2012	13:14:59	M...	NTUSER key
4/4/2012	13:24:43	..B	NTFS_DETECT crtime
4/4/2012	13:29:33	M...	NTUSER key
4/4/2012	13:29:33	M...	NTUSER key
4/4/2012	13:29:33	M...	NTUSER key
4/4/2012	13:48:10	...	MSIE Cache File URL record
4/4/2012	14:05:29	.A.	WinPrefetch
4/4/2012	14:05:29	M...	NTUSER key
4/4/2012	14:05:29	M...	NTUSER key
4/4/2012	14:05:30	M...	NTUSER key
4/4/2012	14:05:39	..C.	NTFS_DETECT ctime;mtime
4/4/2012	14:07:30	...B	NTFS_DETECT crtime

[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\##10.3.58.4#C$#windows#system32] BaseClass: [REG_SZ] Drive
[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.4 Share_Name: \C$\windows\system
TSK:/WINDOWS/Prefetch/AT.EXE-2770DD18.pf
[\Software\Sysinternals] Value: No values stored in key.
[\Software\Sysinternals\PsExec] EulaAccepted: [REG_DWORD_LE] 1
[\Software] Value: No values stored in key.
Location: https://lh6.googleusercontent.com/-OnJX3xRN6UE/TrQhCIxvXVI/AAAAAAAAGVE/K5pwAs2rk6Q/s512/CCF10272011_0000.jpg?w
Prefetch [WMIC.EXE] was executed - run count 6 path: \WINDOWS\SYSTEM32\WBEM\WMIC.EXE hash: 0x3B772CC6 volume: 1 [serial nu
[\Software\Microsoft] Value: No values stored in key.
[\Software\Microsoft\Wbem] Value: No values stored in key.
[\Software\Microsoft\Wbem\WMIC] WMICLC: [REG_DWORD_LE] 1033 mofcompMUIStatus: [REG_DWORD_LE] -1
TSK:/WINDOWS/Prefetch/WMIC.EXE-3B772CC6.pf
TSK:/WINDOWS/Prefetch/TASKLIST.EXE-10D94B23.pf

- **AT.EXE was first executed on 4/4/2012 at 13:24:43 EDT minus 10 seconds @ 13:24:33 EDT**
- **The IP Address of the remote system is 10.3.58.4**
- **The directory mounted was c:\WINDOWS\system32**
- **10.3.58.4 is SHIELDBASE Domain Controller**

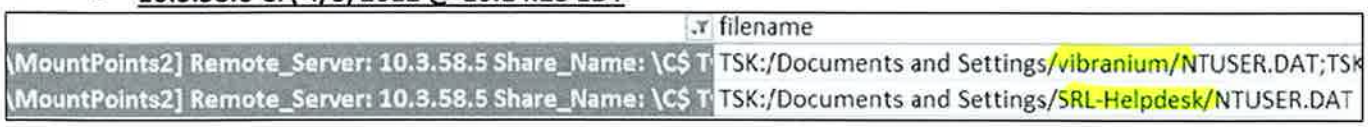
6. List all the any other systems remotely use by mounting the C$ share. When were they last mounted? Why is it possible that we can see that 10.3.58.5 was mounted twice? Which file and directory did the first entry originate from?



4/4/2012	13:14:59	M...	NTUSER key
4/4/2012	15:07:53	M...	NTUSER key
4/5/2012	10:14:23	M...	NTUSER key
4/5/2012	11:36:37	M...	NTUSER key
4/6/2012	15:21:56	M...	NTUSER key

[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.4 Share_Name: \C$\win
[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.4 Share_Name: \C$\Tem
[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.6 Share_Name: \C$ Type
[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ Type
[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ Type

- **10.3.58.4 c:\Temp was last mounted at 4/4/2012 @ 15:07:53 EDT**
- **10.3.58.5 C:\ was last mounted at 4/6/2012 @ 15:21:56 EDT**
- **10.3.58.6 C:\ 4/5/2012 @ 10:14:23 EDT**



filename
\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ T	TSK:/Documents and Settings/vibranium/NTUSER.DAT;TSK
\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ T	TSK:/Documents and Settings/SRL-Helpdesk/NTUSER.DAT

- **10.3.58.5 was mounted by two separate users based on the directories. One of the entries came from the NTUSER.dat file associated with the vibranium user. The 2nd entry came from the SRL-Helpdesk/NTUSER.DAT user account**

7. On 4/4/2012, the user vibranium executed EXCEL.exe, what two files did he open for the first time? What types of files are those? Where are they originally located?

4/4/2012	12:51:21 MACB	FileExts key	File extension .xlsx opened by EXCEL.EXE
4/4/2012	12:51:22 MACB	UserAssist key	UEME_RUNPATH:C:/Program Files/Microsoft Office/Office12/EXCEL.EXE
4/4/2012	12:51:22 ..C.	NTFS $MFT	C:/Program Files/Microsoft Office/Office12/EXCEL.EXE
4/4/2012	12:51:22 ...B	NTFS $MFT	C:/Documents and Settings/vibranium/Recent/Metal Alloy List Research.lnk
4/4/2012	12:51:22 ...B	NTFS $MFT	C:/Documents and Settings/vibranium/Recent/Detailed Vibranium R&D Documents.lnk
4/4/2012	12:51:22 MACB	NTUSER key	Software/Microsoft/Windows/CurrentVersion/Explorer
4/4/2012	12:51:22 ...B	NTFS $MFT	C:/Documents and Settings/vibranium/Local Settings/History/History.IE5/MSHist0120
4/4/2012	12:51:22 M..B	NTFS $MFT	C:/Documents and Settings/vibranium/Local Settings/History/History.IE5/MSHist0120
4/4/2012	12:51:22 MACB	NTUSER key	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/ExtensibleCa
4/4/2012	12:51:22 MACB	NTUSER key	Software/Microsoft/Windows/CurrentVersion/InternetSettings/5.0/Cache/ExtensibleCa
4/4/2012	12:51:29 MACB	XP Prefetch	EXCEL.EXE-34CB65E9.pf: EXCEL.EXE was executed

- `C:\Documents and Settings\vibranium\Recent\Metal Alloy List Research.lnk`
- `C:\Documents and Settings\vibranium\Recent\Detailed Vibranium R&D Documents.lnk`

```
# cd /mnt/windows_mount/Documents\ and\ Settings/vibranium/Recent/

# exiftool Metal\ Alloy\ List\ Research.lnk
```

```
ExifTool Version Number         : 8.10
File Name                       : Metal Alloy List Research.lnk
Directory                       : .
File Size                       : 1349 bytes
File Modification Date/Time     : 2012:04:04 16:51:39+00:00
File Permissions                : rwxrwxrwx
File Type                       : Windows Shortcut
MIME Type                       : application/octet-stream
Flags                           : IDList, LinkInfo, RelativePath, WorkingDir, Unicode
File Attributes                 : Archive
Create Date                     : 2012:03:08 22:11:26+00:00
Access Date                     : 2012:04:04 16:51:38+00:00
Modify Date                     : 2012:03:08 22:11:26+00:00
Target File Size                : 68346
Icon Index                      : (none)
Run Window                      : Normal
Hot Key                         : (none)
Target File DOS Name            : METALA~1.XLS
Drive Type                      : Fixed Disk
Volume Label                    :
Local Base Path                 : C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Docume
nts\Metal Alloy List Research.xlsx
Relative Path                   : ..\..\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Documents\Metal Alloy List
 Research.xlsx
Working Directory               : C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Docume
nts
Machine ID                      : wks-winxp32bit
```

- `C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Documents\Metal Alloy List Research.xlsx`
- The file is an EXCEL workbook

```
# exiftool Detailed\ Vibranium\ R\&D\ Documents.lnk
```

```
ExifTool Version Number         : 8.10
File Name                       : Detailed Vibranium R&D Documents.lnk
Directory                       : .
File Size                       : 954 bytes
File Modification Date/Time     : 2012:04:04 16:59:30+00:00
File Permissions                : rwxrwxrwx
File Type                       : Windows Shortcut
MIME Type                       : application/octet-stream
Flags                           : IDList, LinkInfo, RelativePath, Unicode
File Attributes                 : Directory
Create Date                     : 2012:03:08 22:10:51+00:00
Access Date                     : 2012:04:04 16:59:30+00:00
Modify Date                     : 2012:04:04 16:59:30+00:00
Target File Size                : 0
Icon Index                      : (none)
Run Window                      : Normal
Hot Key                         : (none)
Drive Type                      : Fixed Disk
Volume Label                    :
Local Base Path                 : C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Docume
nts
Relative Path                   : ..\..\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Documents
Machine ID                      : wks-winxp32bit
```

- **C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Vibranium R&D Documents (FOLDER)**

- **The file points to a folder, not a file (note with LNK files, you sometimes will have a hard time differentiating between a file and a folder without examining the actual target file information stored inside the LNK file itself.)**

8. When was the FIND.EXE Command executed last? Can you tell what the attacker was possibly looking for? Which directory name were most of those files located in?

4/5/2012	10:14:23	MACB	MountPoints2 key ##10.3.58.68C$ (remote) Drive mounted
4/5/2012	10:17:30	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Backstopped Accounts - R&D Costs Alloy Research/Credit-Card-Numbers-For-Research.xls
4/5/2012	10:17:30	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Backstopped Accounts - R&D Costs Alloy Research/CC-Backstopped-Accounts.xlsx
4/5/2012	10:18:24	MACB	XP Prefetch FIND.EXE-0EC32F1E.pf: FIND.EXE was executed
4/5/2012	10:18:24	.A..	NTFS $MFT C:/WINDOWS/system32/find.exe
4/5/2012	10:18:34	M.CB	NTFS $MFT C:/WINDOWS/Prefetch/FIND.EXE-0EC32F1E.pf
4/5/2012	10:20:25	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/SUCCESS-TEST-PLAN-VIBRANIUM-ALLOY-I
4/5/2012	10:20:38	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/ADAMANTIUM-Background.docx
4/5/2012	10:20:52	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/Metal Alloy List Research.xlsx
4/5/2012	10:21:05	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/Researched Sub-Atomic Particles.xlsx
4/5/2012	10:21:16	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/The Shield Background and Ongoing Rese
4/5/2012	10:21:31	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/VIBRANIUM.docx
4/5/2012	10:21:31	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/Vibrainium.doc
4/5/2012	10:21:31	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/Vibrainium[1].doc
4/5/2012	10:21:47	.A..	NTFS $MFT C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Vibranium R&D Documents/Dossier - Dr Myron MacLain.docx

- Find.exe was executed on 4/5/2012 at 10:18:24 EDT

- Based on the last access time of the files, it looks as though many document files in the directory "**Detailed Vibranium R&D**" were last accessed.

9. When was the last **cmd.exe** executed?

4/6/2012	15:20:22	.A.	WinPrefetch	Prefetch [CMD.EXE] was executed - run count 60 path: \WINDOWS\SYSTEM32\CMD.EXE hash: 0x087B4001 volume: 1 [s
4/6/2012	15:20:31	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/CMD.EXE-087B4001.pf
4/6/2012	15:21:24	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/system32/cmd.exe
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/net.exe
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netui1.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET269.tmp;TSK:/WINDOWS/system32/ntlanman.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netui0.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netmsg.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET3B4.tmp;TSK:/WINDOWS/system32/davclnt.dll
4/6/2012	15:21:27	.A.	WinPrefetch	Prefetch [NET.EXE] was executed - run count 66 path: \WINDOWS\SYSTEM32\NET.EXE hash: 0x01A53C2F volume: 1 [ser
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET374.tmp;TSK:/WINDOWS/system32/drprov.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/NET.EXE-01A53C2F.pf
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Value: No values stored in key.
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ T
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\##10.3.58.5#C$] BaseClass: [REG_SZ] Drive _Cc
4/6/2012	15:22:20	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/pe.exe
4/6/2012	15:22:20	.A.	WinPrefetch	Prefetch [PE.EXE] was executed - run count 30 path: \WINDOWS\SYSTEM32\PE.EXE hash: 0x0DC593C2 volume: 1 [serial
4/6/2012	15:22:20	M...	NTUSER key	[\Software\Sysinternals] Value: No values stored in key.
4/6/2012	15:22:20	M...	NTUSER key	[\Software] Value: No values stored in key.
4/6/2012	15:22:20	M...	NTUSER key	[\Software\Sysinternals\PsExec] EulaAccepted: [REG_DWORD_LE] 1
4/6/2012	15:22:21	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/PE.EXE-0DC593C2.pf

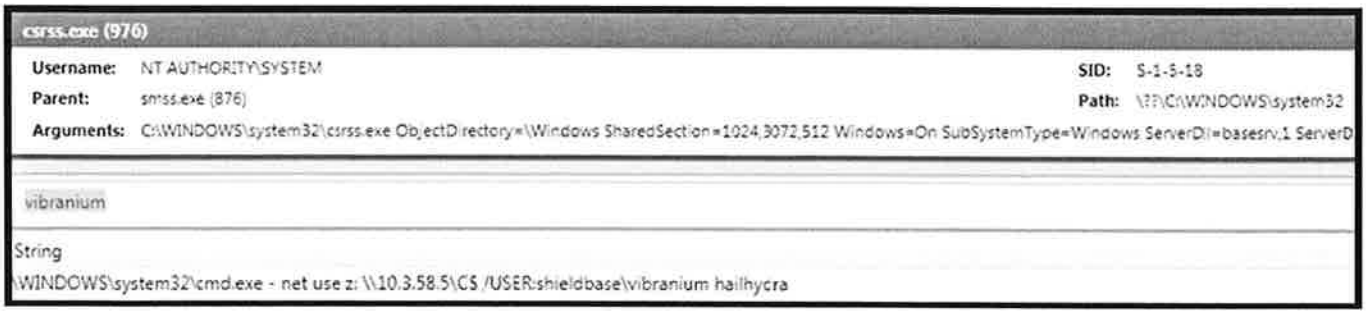
- **cmd.exe was last executed on 4/6/2012 @ 15:20:22 EDT**

10. When was the last **`net.exe`** executed? What do you think the remote IP was that net was used against?

4/6/2012	15:20:22	.A.	WinPrefetch	Prefetch [CMD.EXE] was executed - run count 60 path: \WINDOWS\SYSTEM32\CMD.EXE hash: 0x087B4001 volume: 1 [s
4/6/2012	15:20:31	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/CMD.EXE-087B4001.pf
4/6/2012	15:21:24	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/system32/cmd.exe
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/net.exe
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netui1.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET269.tmp;TSK:/WINDOWS/system32/ntlanman.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netui0.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netmsg.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET3B4.tmp;TSK:/WINDOWS/system32/davclnt.dll
4/6/2012	15:21:27	.A.	WinPrefetch	Prefetch [NET.EXE] was executed - run count 66 path: \WINDOWS\SYSTEM32\NET.EXE hash: 0x01A53C2F volume: 1 [ser
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET374.tmp;TSK:/WINDOWS/system32/drprov.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/NET.EXE-01A53C2F.pf
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Value: No values stored in key.
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ T
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\##10.3.58.5#C$] BaseClass: [REG_SZ] Drive _Cc
4/6/2012	15:22:20	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/pe.exe
4/6/2012	15:22:20	.A.	WinPrefetch	Prefetch [PE.EXE] was executed - run count 30 path: \WINDOWS\SYSTEM32\PE.EXE hash: 0x0DC593C2 volume: 1 [serial
4/6/2012	15:22:20	M...	NTUSER key	[\Software\Sysinternals] Value: No values stored in key.
4/6/2012	15:22:20	M...	NTUSER key	[\Software] Value: No values stored in key.
4/6/2012	15:22:20	M...	NTUSER key	[\Software\Sysinternals\PsExec] EulaAccepted: [REG_DWORD_LE] 1
4/6/2012	15:22:21	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/PE.EXE-0DC593C2.pf

- **net.exe was last executed on 4/6/2012 @ 15:21:27**
- **net.exe was probably used against 10.3.58.5**

11. Have we seen the combination of cmd.exe, net.exe, pe.exe and 10.3.58.5 used in any previous analysis prior to the super timeline? What was the full command that was executed based on prior analysis and what can we correlate via timeline analysis?



- **Yes. We saw the command in our memory analysis of strings. We can see the use of the net.exe command remotely mounting the C$ of 10.3.58.5**

- **The full command is: `net use z: \\10.3.58.5\C$ /USER:sheildbase\vibranium hailhydra`**

12. When was **`pe.exe`** last executed?
    - **pe.exe was last executed on 4/6/2012 at 15:22:20 EDT**

4/6/2012	15:20:22	.A.	WinPrefetch	Prefetch [CMD.EXE] was executed - run count 60 path: \WINDOWS\SYSTEM32\CMD.EXE hash: 0x087B4001 volume: 1 [s
4/6/2012	15:20:31	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/CMD.EXE-087B4001.pf
4/6/2012	15:21:24	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/system32/cmd.exe
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/net.exe
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netui1.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET269.tmp;TSK:/WINDOWS/system32/ntlanman.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netui0.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/netmsg.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET3B4.tmp;TSK:/WINDOWS/system32/davclnt.dll
4/6/2012	15:21:27	.A.	WinPrefetch	Prefetch [NET.EXE] was executed - run count 66 path: \WINDOWS\SYSTEM32\NET.EXE hash: 0x01A53C2F volume: 1 [ser
4/6/2012	15:21:27	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/SET374.tmp;TSK:/WINDOWS/system32/drprov.dll
4/6/2012	15:21:27	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/NET.EXE-01A53C2F.pf
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Value: No values stored in key.
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2] Remote_Server: 10.3.58.5 Share_Name: \C$ T
4/6/2012	15:21:56	M...	NTUSER key	[\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoints2\##10.3.58.5#C$] BaseClass: [REG_SZ] Drive _Co
4/6/2012	15:22:20	.A.	NTFS_DETECT atime	TSK:/WINDOWS/system32/pe.exe
4/6/2012	15:22:20	.A.	WinPrefetch	Prefetch [PE.EXE] was executed - run count 30 path: \WINDOWS\SYSTEM32\PE.EXE hash: 0x0DC593C2 volume: 1 [serial
4/6/2012	15:22:20	M...	NTUSER key	[\Software\Sysinternals] Value: No values stored in key.
4/6/2012	15:22:20	M...	NTUSER key	[\Software] Value: No values stored in key.
4/6/2012	15:22:20	M...	NTUSER key	[\Software\Sysinternals\PsExec] EulaAccepted: [REG_DWORD_LE] 1
4/6/2012	15:22:21	.A.	NTFS_DETECT atime;ctim	TSK:/WINDOWS/Prefetch/PE.EXE-0DC593C2.pf

- What do you think the original name of this service is?
  - **pe.exe is "PSEXEC" – you can see registry keys associated with sysinternals during the 1st time of execution of pe.exe**
- Are there any other systems that might have been the target of the pe.exe?
  - **Yes**
- What is the IP address of that system?
  - **The remote IP address of the system is 10.3.58.5**

13. *EXTRA:* Recover the email from the Outlook.pst file that has the 2011-W2.zip file as an attachment. What is the md5 of the message.txt that it was a part of? Were there any hyperlinks in that email message? What is the md5 of the zipfile? What is the IP address of the sender?

```
# cd /mnt/windows_mount/Documents\ and\ Settings/tdungan/Local
Settings/Application\ Data/Microsoft/Outlook

# cp Outlook.pst /cases/xp-tdungan-c-drive/

# cd /cases/xp-tdungan-c-drive

# pffexport Outlook.pst

# cd Outlook.pst.export

# find . | grep zip
```

```
./Top of Personal Folders/Sent Items/Message00005/Attachments/New-Site-HQ-And-Landing-Pad.zip
./Top of Personal Folders/Inbox/Message00274/Attachments/2011-W2.zip
```

```
# cd Top\ of\ Personal\ Folders/Inbox/Message00274/

# md5sum Message.txt

# cat Message.txt
```

- 3e7cead18a5e237a106ef13850c8daee Message.txt

```
Dear Timothy Dungan,


There were last minute changes to the IRS W-2 reporting system for filing
your tax returns for fiscal year 2011. A new IRS form W-2 needed to be
generated to account for the 2011 payroll tax extension recently passed
by congress that was not accounted for in the previous W-2 mailed to
your home. For this year, we recommend you file an amendment to your
tax return. Not only will you see a bigger income tax refund, but your overall
taxable income amount will also drop, possibly resulting in qualification for a lower
tax bracket.


Below is a link to your new 2011 W-2 from Stark Research Labs that includes the payroll tax
credit.  We do apologize for getting this out so late and immediately
prior to your filing deadline on April 15, 2012.  The password is your username of your system login.

http://bit.ly/GEUMQQ


Please forward any questions back to me at mhill.shield@yahoo.com


Thank you,
Maria Hill
Director, SHIELD
```

- **Hyperlink http://bit.ly/GEUMQQ**

```
# cd Attachments/

# md5sum 2011-W2.zip
```

- **7317b667e2fa1cf4732f946e61f0bd72 2011-W2.zip**

```
# cd ..

# cat InternetHeaders.txt
```

```
Return-Path: mhill.shield@yahoo.com
Received: from nm5-vm2.bullet.mail.ne1.yahoo.com ([98.138.90.153])
        by mail.stark-research-labs.com
        ; Tue, 3 Apr 2012 00:27:41 +0000
Received: from [98.138.90.56] by nm5.bullet.mail.ne1.yahoo.com with NNFMP; 03 Apr 2012 00:27:46 -0000
Received: from [98.138.89.252] by tm9.bullet.mail.ne1.yahoo.com with NNFMP; 03 Apr 2012 00:27:46 -0000
Received: from [127.0.0.1] by omp1044.mail.ne1.yahoo.com with NNFMP; 03 Apr 2012 00:27:46 -0000
X-Yahoo-Newman-Property: ymail-3
X-Yahoo-Newman-Id: 938012.46875.bm@omp1044.mail.ne1.yahoo.com
Received: (qmail 36874 invoked by uid 60001); 3 Apr 2012 00:27:46 -0000
DKIM-Signature: v=1; a=rsa-sha256; c=relaxed/relaxed; d=yahoo.com; s=s1024; t=1333412866; bh=rR1A8d0rfVR2WGbp2uy1vorX1FncxvMU
xDONWm54rss=; h=X-YMail-OSG:Received:X-Mailer:References:Message-ID:Date:From:Reply-To:Subject:To:In-Reply-To:MIME-Version:Co
ntent-Type; b=ws4tZJCah+d+VeziIn/rM5RL9G8WtAZ3JOVhhduHyw4N/WBTD1lnw7mudJ2aYBcdlyb/pzc/RdWTYmf6bfqn8fcr9yX1icmY3FcJ4Xd3Ts8YBC2
xCoShFKtds7A6p+fuCYNOO5juKglCAMmEJVNtUL4mFpllb4uPHAxOX1NojvE=
DomainKey-Signature:a=rsa-sha1; q=dns; c=nofws;
  s=s1024; d=yahoo.com;
  h=X-YMail-OSG:Received:X-Mailer:References:Message-ID:Date:From:Reply-To:Subject:To:In-Reply-To:MIME-Version:Content-Type;
  b=RUg6zTgoqVxG6PG7iXzq+FrYyaPn6jOtdoD9WioWZVeTBr1G32acxCyobUjdJDfooSYaD4y14jMgps0nNiGrj7ax6g0xRWGzf+Id0vXsNb2ZLiQ2lDc9fcU9y
RNoVVEQ0yjVdVFOW6qMSCpWm4XEjlVYIX5otKww1Xvhpg5aP6U=;
X-YMail-OSG: q2HCFz0VM1kV0Bot8cSYkYp2mW78E2QIuIRflQQiNXk9SRg
 83PWVpZxIKDTgq.s5Y0AkzMPa11cJmWy.PXissJoqDC86szOO.1Qaolq_B_N
 F9m5RLb9A1tFrmj1A0S5uUM2L8k0huuHm_VzUyx9pt2S2CeLhE5UoIGm3DDH
 i4bRcUQw6pW6lI17z6ee.53aOE7yDMsqWfEs0155BITZvMEa1K.MP43PRC.S
 DfzqoxwwN_LNxcInamWjonCXsTU6aSFY0503GqiKuM_plXe7VrgDaySD4dhy
 hPi_AyLc.dQsOskd4KkJiLciFfSOZ8o0jlkw3hgRmjvrLSTr8N2pBUHBkh0n
 49d4C2vjybd9G3f989NPmgTaBrhqT5cksWySxMFl0L041Fxk.fQvL44CrhYs
 9iLvmf.mtsc30bmUsJ14f_L.QqP.h7wyDjb3mfZdOqIDOQ1MlEcfjvPISZDf
 c3a50d.MnkRRfa85TgKag5A--
Received: from [72.241.5.37] by web122206.mail.ne1.yahoo.com via HTTP; Mon, 02 Apr 2012 17:27:46 PDT
X-Mailer: YahooMailWebService/0.8.117.340979
References: <1333384103.62371.YahooMailNeo@web122204.mail.ne1.yahoo.com> <1333387407.39356.YahooMailNeo@web122206.mail.ne1.ya
hoo.com> <1333390217.32396.YahooMailNeo@web122205.mail.ne1.yahoo.com>
```

- **IP Address = 72.241.5.37**

Exercise – Key Takeaways combined with Exercise File System Timeline

- **Suspicious binaries were identified in the following directory**
  - **`C:\windows\system32\dllhost\`**
  - **The dllhost directory was created on Mon Apr 02 2012 20:34:26 EST5EDT**

- **`C:/WINDOWS/system32/dllhost/winclient.reg` contained the registry settings for the svchost.exe beacon**

- **`reg.exe executed winclient.reg to add the registry settings for the svchost.exe file.`**

- **Additional possible malware was identified: `C:\Windows\System32\hydrakatz.exe`**

- **The executable a.exe was executed multiple times and is usually last executed around the time the hacker was last active on the system.**

- **The rogue account, vibranium, was created around Wed Apr 04 2012 12:40:18**

- **The following files were opened by the vibranium account:**
  - Metal Alloy List Research
  - Detailed Vibranium R&D Documents
  - SUCCESS-TEST-PLAN-VIBRANIUM-ALLOY-RESULTS
  - Dossier - Dr Myron MacLain

- **The vibranium user added several outlook accounts and created a new pst file on Thu Apr 05 2012 12:07:58 found in `C:/Documents and Settings/vibranium/Local Settings/Application Data/Microsoft/Outlook/EXFIL.pst`**

- **The original spear phishing email was found in tdungan's Outlook.pst file. It contained two methods to try and execute code: a link and a zipfile containing a PDF**

- **It is clear that 10.3.58.7 was a hop point to 3 other systems in the SHIELDBASE environment**
  - 10.3.58.4 (Domain Controller)
  - 10.3.58.5 (Natasha Romanoff's Win7 32 Bit system)
  - 10.3.58.6 (Nick Fury's Win7 64 Bit system)

This page intentionally left blank.

Exercise 15 – Volume Shadow Examinations

Objectives

- To examine methods to mount and examine volume shadows using the SIFT workstation
- Explore techniques to extract data from multiple volume shadows in a Windows 7, Win8, or Windows Server
- Explore the RecentFileCache.bcf registry hive
- Use Timeline analysis techniques to quickly ascertain where key data might be found inside multiple volume shadow snapshots in a disk image

Mount Windows 7 Image and Mount Multiple Volume Shadows

(Note: Some of this might already be accomplished via earlier exercises, but this is the state that we hope your system is in prior to the start of this exercise. Just in case your system rebooted, we are including a guide to help you get back to the proper analysis starting point prior to the beginning of this exercise.)

1. *Start your SIFT VMware Workstation in VMware Workstation.*
2. *Login the VMware machine.*
   - *LOGIN =* **`sansforensics`**
   - *PASSWORD =* **`forensics`**
3. *Start a terminal, elevate your privs to root, and change into the /cases/win7-c-drive directory.*

```
$ sudo su -

# cd /cases/win7-32-nromanoff-c-drive
```

4. *Mount your evidence files so you can see the win7-32-nromanoff-c-drive raw image and files/folders from the system.*

```
# ewfmount win7-32-nromanoff-c-drive.E01 /mnt/ewf_mount2/

# mount -o ro,loop,show_sys_files,streams_interface=windows
/mnt/ewf_mount2/ewf1 /mnt/windows_mount2
```

5. *Examine how many VSS snapshots exist inside the win7-32-nromanoff-c-drive raw image*

```
# vshadowinfo /mnt/ewf_mount2/ewf1
```

How many Snapshots stores exist? 4

You should see 4 stores.

```
Store 1: Creation Time = Mar 15, 2012 23:12:03
Store 2: Creation Time = Mar 23, 2012 04:00:15
Store 3: Creation Time = Mar 31, 2012 04:00:12
Store 4: Creation Time = Apr 04, 2012 20:05:02
```

6. *Mount raw image VSS using vshadowmount and mount all logical filesystems found in snapshot*

```
# vshadowmount /mnt/ewf_mount2/ewf1 /mnt/vss

# cd /mnt/vss

# for i in vss*; do mount -o
ro,loop,show_sys_files,streams_interface=windows $i
/mnt/shadow_mount/$i; done
```

When you press <RETURN> after inputting the above command – you might see an error pop up. One of the snapshots in this image is corrupted. Change directories to the **/mnt/shadow_mount** directory and see if you can determine the store # that failed to mount properly.

```
# cd /mnt/shadow_mount/

# ls

vss1  vss2  vss3  vss4
```

You will notice that the vss1 failed to mount properly. That is the corrupted VSS. This happens from time to time and is nothing to worry about. But notice the error, this will be a good way to identify if a VSS has been corrupted.

VSS RecentFileCache.bcf Examinations – Questions

1. Examine the RecentFileCache.bcf file found in the **C:\Windows\AppCompat\Programs** folder in each of the 3 mounted volume shadow stores. Which of the volume shadow snapshot RecentFileCache.bcf files will likely contain data and why?

```
# cd /mnt/windows_mount2/Windows/AppCompat/Programs
# ls -la (Note: If RecentFileCache.bcf is 20 bytes or smaller it is likely empty)
# rfc.pl RecentFileCache.bcf
# cd /mnt/shadow_mount/vss2/Windows/AppCompat/Programs
# ls -la
# rfc.pl RecentFileCache.bcf
# cd /mnt/shadow_mount/vss3/Windows/AppCompat/Programs
# ls -la
# rfc.pl RecentFileCache.bcf
# cd /mnt/shadow_mount/vss4/Windows/AppCompat/Programs
# ls -la.
# rfc.pl RecentFileCache.bcf
```

2. Parse the RecentFileCache.bcf file that you discovered in step 1. List the programs that were executed?

```
# cd /mnt/shadow_mount/vss4/Windows/AppCompat/Programs
# rfc.pl RecentFileCache.bcf
```

psexesvc.exe
taskhlistime

3. On what day were these programs executed? What was the time the last program (unknown) was executed?

```
# ls --full-time /mnt/shadow_mount/vss4/Windows/AppCompat/Programs
```

VSS RecentFileCache.bcf Examinations – Questions with STEP BY STEP

1. Examine the RecentFileCache.bcf file found in the `C:\Windows\AppCompat\Programs` folder in each of the 3 mounted volume shadow stores and the current image. Which of the volume shadow snapshot RecentFileCache.bcf files will likely contain data and why?

VSS4 – it is the only RecentFileCache.bcf file that has a filesize larger than 20.

```
root@siftworkstation:/mnt/windows_mount/Windows/AppCompat/Programs# ls -la
total 32
drwxrwxrwx 1 root root  4096 Apr  6  2012 .
drwxrwxrwx 1 root root     0 Jul 13  2009 ..
-rwxrwxrwx 2 root root 20676 Apr  6  2012 AEINV_PREVIOUS.xml
-rwxrwxrwx 2 root root    20 Apr  6  2012 RecentFileCache.bcf
root@siftworkstation:/mnt/windows_mount/Windows/AppCompat/Programs# cd /mnt/shad
ow_mount/vss4/Windows/AppCompat/Programs/; ls -la
total 32
drwxrwxrwx 1 root root  4096 Apr  4  2012 .
drwxrwxrwx 1 root root     0 Jul 13  2009 ..
-rwxrwxrwx 2 root root 20676 Apr  4  2012 AEINV_PREVIOUS.xml
-rwxrwxrwx 2 root root   142 Apr  4  2012 RecentFileCache.bcf
root@siftworkstation:/mnt/shadow_mount/vss4/Windows/AppCompat/Programs# cd /mnt/
shadow_mount/vss3/Windows/AppCompat/Programs/; ls -la
total 28
drwxrwxrwx 1 root root  4096 Mar 20  2012 .
drwxrwxrwx 1 root root     0 Jul 13  2009 ..
-rwxrwxrwx 2 root root 20479 Mar 20  2012 AEINV_PREVIOUS.xml
-rwxrwxrwx 2 root root    20 Mar 20  2012 RecentFileCache.bcf
root@siftworkstation:/mnt/shadow_mount/vss3/Windows/AppCompat/Programs# cd /mnt/
shadow_mount/vss2/Windows/AppCompat/Programs/; ls -la
total 28
drwxrwxrwx 1 root root  4096 Mar 20  2012 .
drwxrwxrwx 1 root root     0 Jul 13  2009 ..
-rwxrwxrwx 2 root root 20479 Mar 20  2012 AEINV_PREVIOUS.xml
-rwxrwxrwx 2 root root    20 Mar 20  2012 RecentFileCache.bcf
root@siftworkstation:/mnt/shadow_mount/vss2/Windows/AppCompat/Programs#
```

2. Parse the RecentFileCache.bcf file that you discovered in step 1. List the programs that were executed?

```
# cd /mnt/shadow_mount/vss4/Windows/AppCompat/Programs

# rfc.pl RecentFileCache.bcf
```

C:\windows\psexesvc.exe
C:\windows\system32\tasklist.exe

```
root@siftworkstation:/mnt/shadow_mount/vss4/Windows/AppCompat/Programs# rfc.pl RecentFileCache.bcf

c:\windows\psexesvc.exe
c:\windows\system32\tasklist.exe
```

3. On what day were these programs executed? What was the time the last program (unknown) was executed?

Use the last modification time of the file. 2012-04-04 15:48:47 UTC

```
root@siftworkstation:/mnt/shadow_mount# ls --full-time vss4/Windows/AppCompat/Programs/
total 28
-rwxrwxrwx 2 root root 20676 2012-04-04 13:01:31.162634700 +0000 AEINV_PREVIOUS.xml
-rwxrwxrwx 2 root root   142 2012-04-04 15:48:47.660714700 +0000 RecentFileCache.bcf
root@siftworkstation:/mnt/shadow_mount#
```

VSS SuperTimeline Examinations – Questions

A VSS Supertimeline will take a long time to complete. We cover how to accomplish it in the example following this exercise, but in the meantime we have already prepped the timeline for you. You can find a copy precooked for you in the **`/cases/win7-32-nromanoff-c-drive/precooked/volume-shadow/vss-supertimeline.xlsx`**

Open the copy of precooked for you in the **`/cases/win7-32-nromanoff-c-drive/precooked/volume-shadow/vss-supertimeline.xlsx`**

1. Examine the PreFetch File IEXPLORE.EXE-1B894AFB.pf. Find the instances of that file in the timeline, based on what we can examine what is the probable last 3 execution times of Internet Explorer based on the Timeline only? *HINT: The creation time of a Prefetch (.pf) file is the first time that application has run. The modification time of a Prefetch (.pf) file is the last time the application was run. We are seeing multiple modification times in the timeline due to additional entries provided via Volume Shadow Copies*

   Execution Time 1: _____
   Execution Time 2: _____
   Execution Time 3: _____

2. How many times can we tell that DEFRAG.EXE was executed based on the historical data provided by the original image and the VSS images?

   _____
   _____
   _____
   _____
   _____

3. Examine the NTUSER.DAT information on the system for both vibranium and for nromanoff. Did the vibranium account exist prior to 4/2/2012? Why? When was the vibranium account created on the nromanoff system indicating the first time the user interactively logged in?

   _____
   _____
   _____
   _____

4. Does the C:\Windows\System32\dllhost\svchost.exe file exist on this system? First, what is the creation time of the file? Second, when was the file likely REALLY created (per Access timestamp)?

   _____
   _____
   _____
   _____

VSS SuperTimeline Examinations – Step-by-Step Guide

A VSS Supertimeline will take a long time to complete. We cover how to accomplish it in the example following this exercise, but in the meantime we have already prepped the timeline for you. You can find a copy precooked for you in the **`/cases/win7-32-nromanoff-c-drive/precooked/volume-shadow/vss-supertimeline.xlsx`**

1. Examine the PreFetch File IEXPLORE.EXE-1B894AFB.pf. Find the instances of that file in the timeline, based on what we can examine what is the probably last 3 execution times of Internet Explorer based on the Timeline only? *HINT: The creation time of a Prefetch (.pf) file is the first time that application has run. The modification time of a Prefetch (.pf) file is the last time the application was run. We are seeing multiple modification times in the timeline due to additional entries provided via Volume Shadow Copies*

A	B	C	D	J
date	time	timezon	MAC	short
3/22/2012	11:03:42	EST5EDT	.A.B	/Windows/Prefetch/IEXPLORE.EXE-1B894AFB.pf
3/22/2012	11:11:38	EST5EDT	M.C.	/Windows/Prefetch/IEXPLORE.EXE-1B894AFB.pf
4/3/2012	18:33:51	EST5EDT	M...	/Windows/Prefetch/IEXPLORE.EXE-1B894AFB.pf
4/4/2012	7:48:23	EST5EDT	..C.	/Windows/Prefetch/IEXPLORE.EXE-1B894AFB.pf
4/4/2012	16:11:21	EST5EDT	M.C.	/Windows/Prefetch/IEXPLORE.EXE-1B894AFB.pf

Execution Time 1: **03/22/2012 – 11:11:28 (remember the -10 rule)**
Execution Time 2: **04/03/2012 – 18:33:41**
Execution Time 3: **04/04/2012 – 16:11:11**

2. How many times can we tell that DEFRAG.EXE was executed based on the historical data provided by the original image and the VSS images?

date	time	timezon	MAC	short
9/21/2011	1:16:03	EST5EDT	.A.B	/Windows/Prefetch/DEFRAG.EXE-738093E8.pf
3/20/2012	18:47:11	EST5EDT	M.C.	/Windows/Prefetch/DEFRAG.EXE-738093E8.pf
3/30/2012	5:20:13	EST5EDT	M.C.	/Windows/Prefetch/DEFRAG.EXE-738093E8.pf
4/2/2012	6:21:46	EST5EDT	M...	/Windows/Prefetch/DEFRAG.EXE-738093E8.pf
4/5/2012	7:54:36	EST5EDT	M.C.	/Windows/Prefetch/DEFRAG.EXE-738093E8.pf

Execution Time 1: **09/21/2011 – 01:15:53 (remember the -10 rule)**
Execution Time 2: **03/20/2012 – 18:47:01**
Execution Time 3: **03/30/2012 – 05:20:03**
Execution Time 4: **04/02/2012 – 06:21:36**
Execution Time 5: **04/05/2012 – 07:54:26**

3. Examine the NTUSER.DAT information on the system for both **vibranium** and for **nromanoff** accounts. Did the **vibranium** account exist prior to 4/2/2012? Why?

date	time	timezon	MAC	short
11/10/2010	3:22:13	EST5EDT	...B	/Users/nromanoff/NTUSER.DAT
3/22/2012	23:57:59	EST5EDT	M.C.	/Users/nromanoff/NTUSER.DAT
3/30/2012	23:48:07	EST5EDT	M.C.	/Users/nromanoff/NTUSER.DAT
4/3/2012	17:19:53	EST5EDT	...B	/Users/vibranium/NTUSER.DAT
4/4/2012	15:10:55	EST5EDT	M.C.	/Users/vibranium/NTUSER.DAT
4/4/2012	16:04:56	EST5EDT	M.C.	/Users/nromanoff/NTUSER.DAT
4/6/2012	15:44:17	EST5EDT	MA..	/Users/nromanoff/NTUSER.DAT
4/7/2012	17:04:56	EST5EDT	M.C.	/Users/vibranium/NTUSER.DAT

The user account vibranium's birthdate was created on 4/3/2012. We also know that there were multiple VSS snapshots that were taken prior to 4/4/2012. This explains why we see multiple modification times to the NTUSER.DAT files that existed for nromanoff and not for the vibranium account.

4. Does the C:\Windows\System32\dllhost\svchost.exe file exist on this system? First, what is the creation time of the file? Second, when was the file likely REALLY created (per Access timestamp)?

3/31/2003	9:00:00	EST5EDT	...B	/Windows/System32/dllhost/svchost.exe
4/3/2012	18:40:19	EST5EDT	.A..	/Windows/System32/dllhost/svchost.exe
4/3/2012	18:40:36	EST5EDT	.A..	/Windows/System32/dllhost/winclient.reg
4/3/2012	18:40:39	EST5EDT	...B	/Windows/System32/dllhost/winclient.reg

Yes, the svchost.exe file exists in the C:\Windows\System32\dllhost directory. It looks as though it might have been created and "last accessed" around 04/03/2012 at 18:40:19. The birthdate of the file was 3/31/2003 at 0900. This would indicate that the file is probably timestomped.

Remember since access time is turned off on Win7/8 systems by default, the only time the "accessed time" is updated is when the file is created on a volume. The attackers might not have known this and forgot to backdate the file. Note that this analysis technique will only work on environments that have the access time turned off (Win7/Win8). Otherwise A/V is known to ruin access times on system and the MFT $Filename timestamps would be the best way to tell. We will cover $Filename timestamps in the next couple of sections of the course.

Exercise – Key Takeaways

- Volume Shadow Copies are critical to examinations on Windows 7, Windows 8, and Windows Server systems. Make sure your administrators do not turn them off via Group Policy.
- **`Vibranium`** User first interactively logged into this system around 4/3/2012 at 17:19:53 EDT
- The **`svchost.exe`** malware is found on the system and created around 4/3/2012 at 18:40:19 EDT

Exercise 16 – Extracting Stream Based Data

Objectives

- Extract stream based data from both the xp-tdungan memory image and disk image.
- Create a list of keywords based on case data we know about from earlier sources (timeline and memory forensics).
- Examine the stream data for both the memory image and disk image to determine what relevant data could be analyzed in the case.

Exercise Preparation

1. Please ensure your evidence is mounted correctly at **`/mnt/windows_mount`** if not, please follow **Exercise 2 – Mounting Evidence Using SIFT** again to mount your evidence.

```
# cd /mnt/windows_mount

# ls
```

```
root@SIFT-Workstation:/mnt/windows_mount# ls
$AttrDef        Documents and Settings  MSOCache           System Volume Information
AUTOEXEC.BAT    $Extend                 NTDETECT.COM       Temp
$BadClus        hiberfil.sys            ntldr              $UpCase
$Bitmap         IO.SYS                  pagefile.sys       $Volume
$Boot           $LogFile                Program Files      WINDOWS
boot.ini        $MFTMirr                RECYCLER
CONFIG.SYS      MSDOS.SYS               $Secure
```

If you do not see the above, then please follow directions in Exercise 2 – Mounting Evidence Using SIFT

There are two exercises question sets below

EXERCISE 1 - Extract Stream Data from Memory

1. Step-by-Step Guide Setup
2. Questions
3. Questions with Step-By-Step Guide

EXERCISE 2 - Extract Stream Data from Disk Image

1. Step-by-Step Guide Setup
2. Questions
3. Questions with Step-by-Step Guide

Extract Stream Data from Memory – Step-by-Step Guide

1. Create a Keyword List.

```
# kedit /cases/xp-tdungan-c-drive/keywords.txt
```

2. Add the following keywords that we have seen through timeline analysis or memory analysis. In ***(parentheses – do not add this text)*** we mention where we initially discovered the string during analysis.

`hailhydra`	***(found in memory image in string search used in "net use")***
`hydrakatz`	***(found in timeline and executed during attack)***
`spinlock`	***(exited process found in memory image)***
`hyvy`	***(found in timeline and executed during attack)***
`\\dllhost\\svchost\.exe`	***(found in memory image and timeline- beacon)***
`199\.73\.28\.114`	***(IP Address found in memory and in .reg file)***
`\\netman\\domain`	***(registry path where beacon config stored)***
`EXFIL\.pst`	***(name of PST file generated – found in timeline)***
`\\winclient\.reg`	***(found in timeline, location of beacon config)***
`\\Temp\\a\.exe`	***(found in timeline and memory, active beacon)***
`\\pe\.exe`	***(found in timeline and memory, psexec)***

```
# cp /cases/xp-tdungan-c-drive/keywords.txt /cases/xp-tdungan-
memory
```

3. Change directories to the `/cases/xp-tdungan-memory`.

```
# cd /cases/xp-tdungan-memory/
```

(PLEASE READ: The processing done in the next step will take time. As a result, it is recommended for the next part of the exercise, we recommend that you use the pre-cooked version listed below while the command above is extracting the data for you. We won't usually "cheat" like this, but some data extraction is very lengthy and your time in class is precious. Close your eyes and imagine those 2 hours have passed and you are going to begin your data analysis.)

4. Run `bulk_extractor` using your keywords against the collected memory file.

```
bulk_extractor -F keywords.txt -e net -e aes -e wordlist -o /cases/xp-
tdungan-memory/bulk-extractor-memory-output /cases/xp-tdungan-
memory/xp-tdungan-memory-raw.001
```

Pre-Cooked Extracted Stream Data for Memory Image

1. If your bulk_extractor process has not finished processing yet, please use the pre-cooked version found in: **/cases/xp-tdungan-c-drive/precooked/bulk-extractor/bulk-extractor-memory-output.zip**

```
# cd /cases/xp-tdungan-c-drive/precooked/bulk-extractor

# unzip bulk-extractor-memory-output.zip
```

Examine Memory Extracted Data Using BEViewer

1. Run bulk_extractor Viewer and Open the Bulk Extractor Output Directory.

```
# BEViewer
```

Open the report.xml found in:
/cases/xp-tdungan-c-drive/precooked/bulk-extractor/bulk-extractor-memory-output/report.xml

Select the image folder as **"Use Path From Report"** pointing it to the memory image file:
/cases/xp-tdungan-memory/xp-tdungan-memory-raw.001

2. Spend some time going through the results of **bulk_extractor** from the memory image and answer the questions in the following two exercises.

Extract Stream Data from Memory – Questions

1. **How many times was the text** nfury@stark-research-labs.com **found in memory?**
   - 17

2. **In the `rfc822.txt` output, what is the byte offset of the first hit for `199.73.28.114`?**
   - 16794879

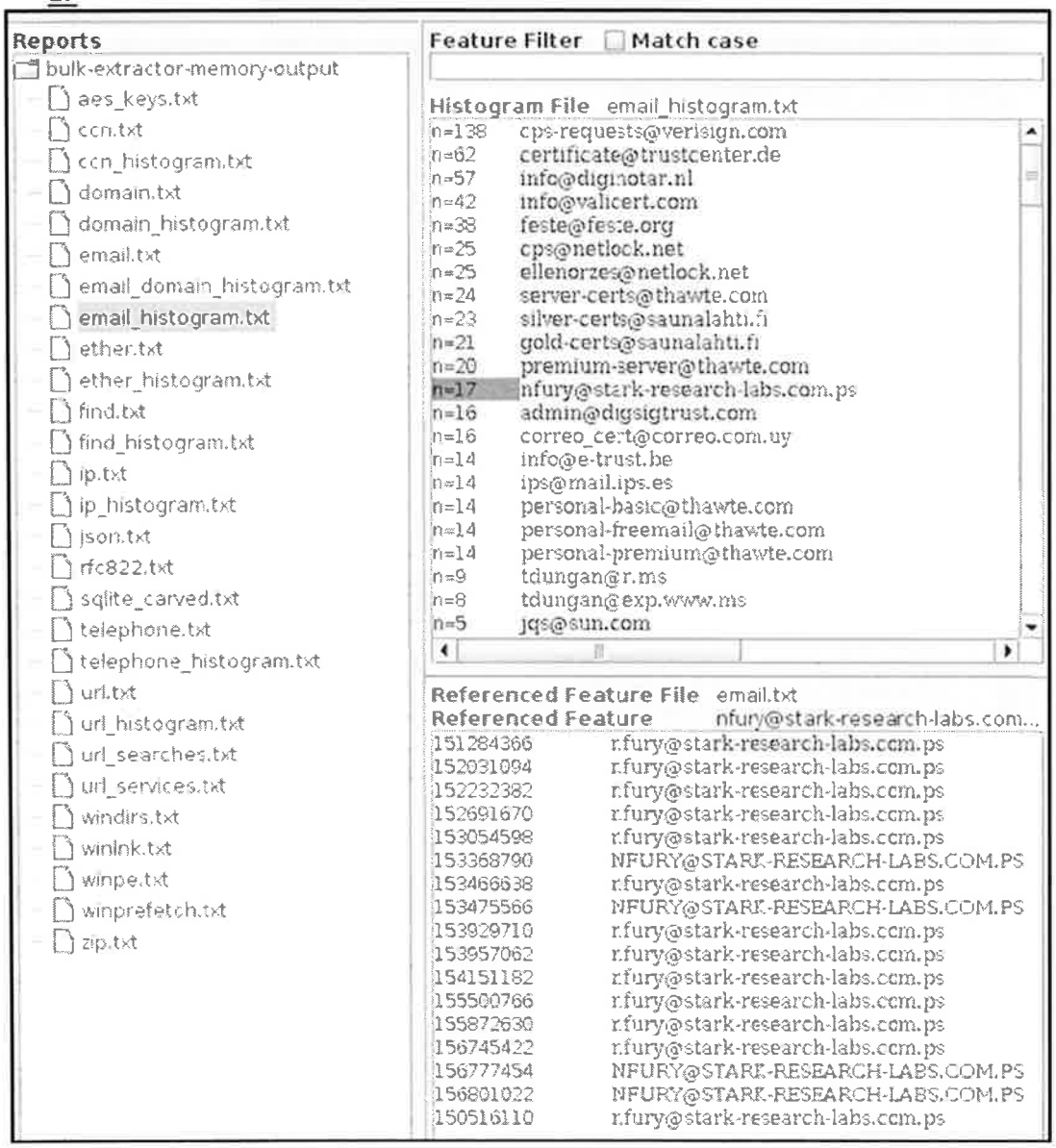
3. **In the `url_histogram.txt`, find the `http://192.168.1.5/ads` hits. We are looking for additional C2 channel servers and possibly searching for /ads will aid us in finding additional beacon locations if the adversaries used more than one.**

   - **What registry key do we see in the memory image around this hit?**
     - system\currentcontrolset\services\winsock2
   - **What filename is around this hit?**
     - a.exe.
   - **What do you think this hit is related to? Why?**
     - program that runs at registry key

Questions Step-by-Step

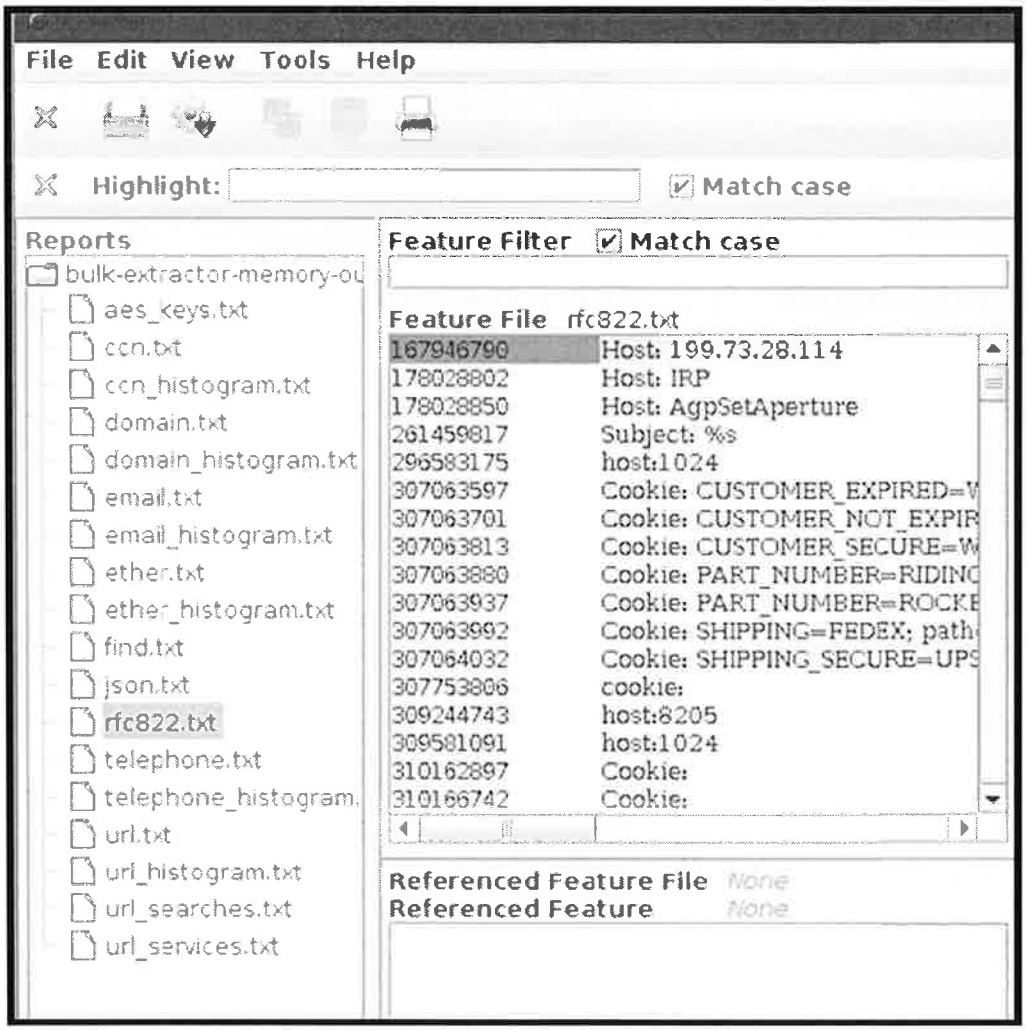
1. How many times was the text nfury@stark-research-labs.com found in memory?

- **17**



2. **In the `rcf822.txt` output, what is the byte offset of the first hit for `199.73.28.114`**

- **<u>167946790</u>**



3. **In the `url_histogram.txt`, find the `http://192.168.1.5/ads` hits. We are looking for additional C2 channel servers and possibly searching for `/ads` will aid us in finding additional beacon locations if the adversaries used more than one.**

   - What registry key do we see in the memory image around this hit?

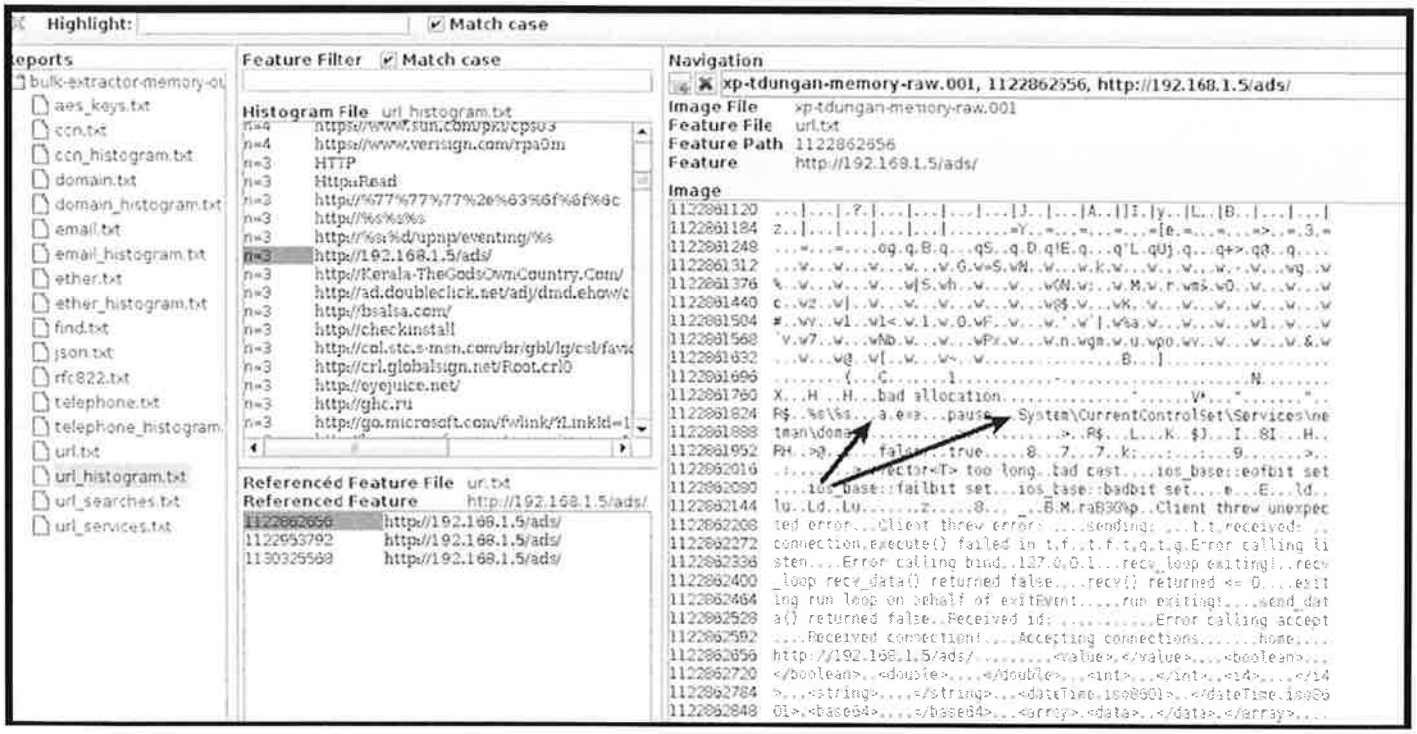
     - **`System\CurrentControlSet\Services\netman\domain`**

   - What filename is also around this hit?

     - **`a.exe`**

   - What do you think this hit is related to? Why could this be important?

     - **This file is related to the dllhost/svchost.exe beacon backdoor due to the existence of the registry key and file name. This could be important as the svchost.exe process loads in memory with the default IP address set to 192.168.1.5. That address could now possibly be used as an indicator of compromise for other systems**



Extract Stream Data from Disk Image – Step-by-Step Guide

1. Change directories to the **/cases/xp-tdungan-c-drive/**

```
# cd /cases/xp-tdungan-c-drive/
```

(PLEASE READ: This next step will take a LOT of time. We recommend for the next part of the exercise that you use the pre-cooked version listed below while the command above is extracting the data for you. We won't usually "cheat" like this, but some data extraction is very lengthy and your time in class is precious. Close your eyes and imagine those 7 hours have passed and you are going to begin your data analysis.)

C

2. Run **bulk_extractor** against using your keywords against the **xp-tdungan-c-drive.E01** file.

```
# bulk_extractor -F /cases/xp-tdungan-c-drive/keywords.txt -e net -e
aes -e facebook -e outlook -o /cases/xp-tdungan-c-drive/bulk-
extractor-c-drive-output /cases/xp-tdungan-c-drive/xp-tdungan-c-
drive.E01
```

3. Run **BEViewer** to examine the extracted streams from the disk image from the output in your precooked exercise location.

Pre-Cooked Extracted Stream Data from Disk Image

```
# cd /cases/xp-tdungan-c-drive/precooked/bulk-extractor

# unzip bulk-extractor-c-drive-output.zip

# BEViewer
```

1. Open the report.xml found in:
   /cases/xp-tdungan-c-drive/precooked/bulk-extractor/bulk-extractor-c-drive-output/report.xml

Select the image folder as **"Use Path From Report"** pointing it to the c-drive image: **/cases/xp-tdungan-c-drive/xp-tdungan-c-drive.E01**

2. Remember to point the original image back at **using BEViewer**

Extract Stream Data from Disk Image – Questions

1. **How many times was the text `tdungan@stark-research-labs.com` (`lower case only`) found in the disk image?**

   - 5875

2. **What is the first hit and byte offset in the find.txt report that is found from the hibernation file?** (Note: One of bulk_extractors exciting features is the ability to examine the content of the compressed ram image embedded in the hiberfil.sys files. Once the hiberfil.sys data stream is detected, it will extract the streams and make it easier to process. This question shows some of the content extracted from a hibernation file data location. HINT: *students should look for byte offsets that include the keyword "HIBER")*

   - 101873.9832 - HIBERNATE - 5535

3. **How many times was the URL search term "`file wiper`" (`file+wiper` or `file%20wiper`) found in the disk image?**

   - 128

4. **Are any valid Credit Card Numbers found in the image?**

   - **What type of file are many of them found in? (Review: In FOR408 we learned that the new Office 2007/2010 format for documents that end in "x" (e.g. `docx`, `pptx`, `xlsx`) are actually zipfiles with embedded xml data. If the data is found in a zipfile, it is possible that it could be an office document.**

     - outlook / word / wpg.

   - **What is the byte offset of the beginning of the zipfile?**

     - 161706

5. **Using the url_services.txt file, perform a search for the IP address used for the beacon 199.73.28.114.**

   - **How many total hits were there for this IP address and for what port address?**

     - 21

   - **Based on the "Referenced Feature File" window pane below in your BEViewer output, what application was specifically using this port and IP address. (Best Guess)**

     - MYVY.exe

6. **Why do you think the following output exists inside of the url_searches.txt file?**

- _____
- _____
- _____

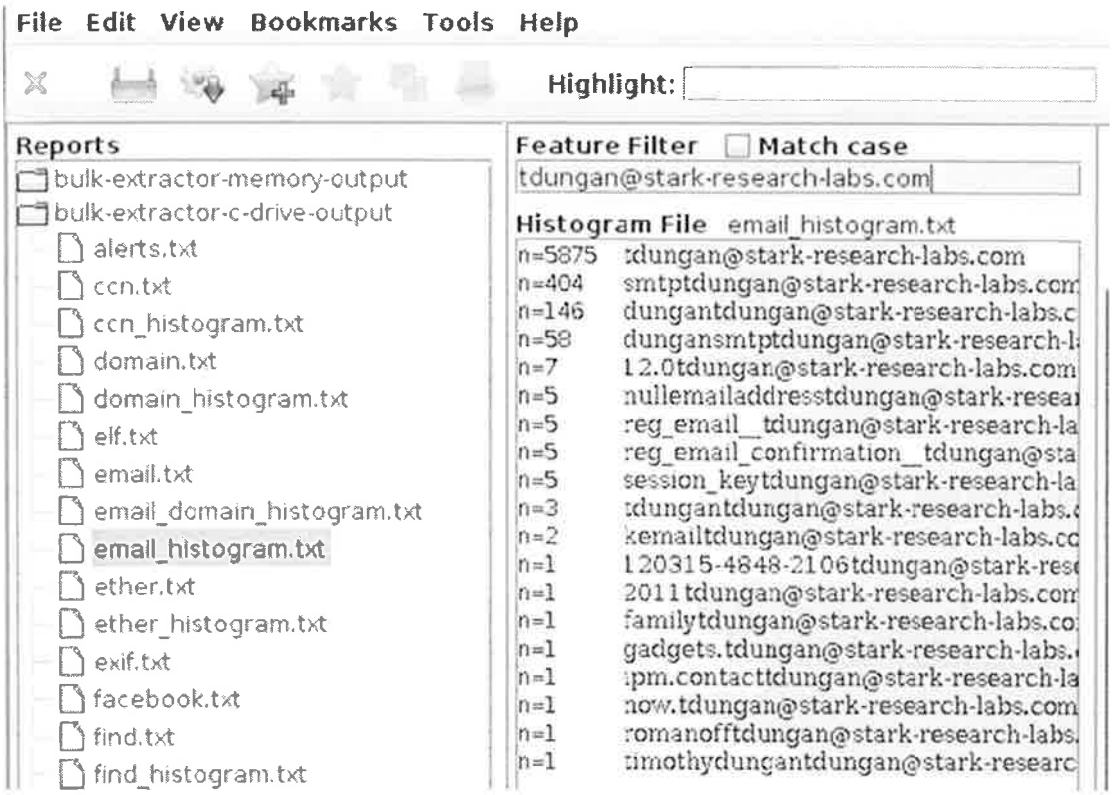
7. **Based on the `email_histogram`, who does the user of this machine receive e-mail from or send e-mail to the most times in the same domain? The user of the machine is `tdungan@stark-research-labs.com`**

- _____

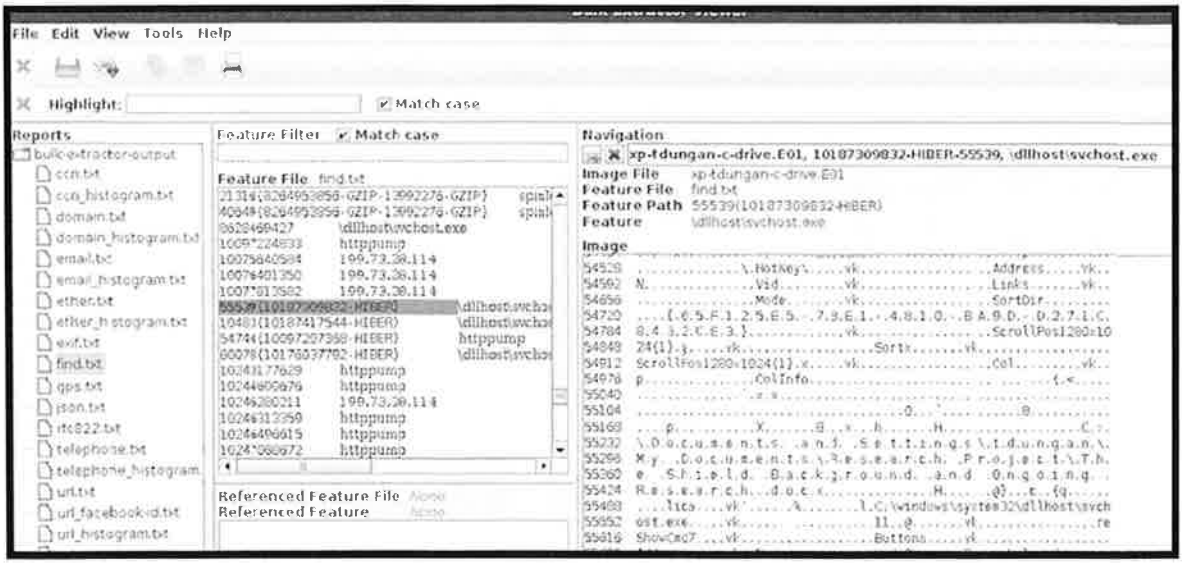
Extract Stream Data from Disk Image – Questions Step-by-Step

1. How many times was the text tdungan@stark-research-labs.com found in the disk image?

- **5875**

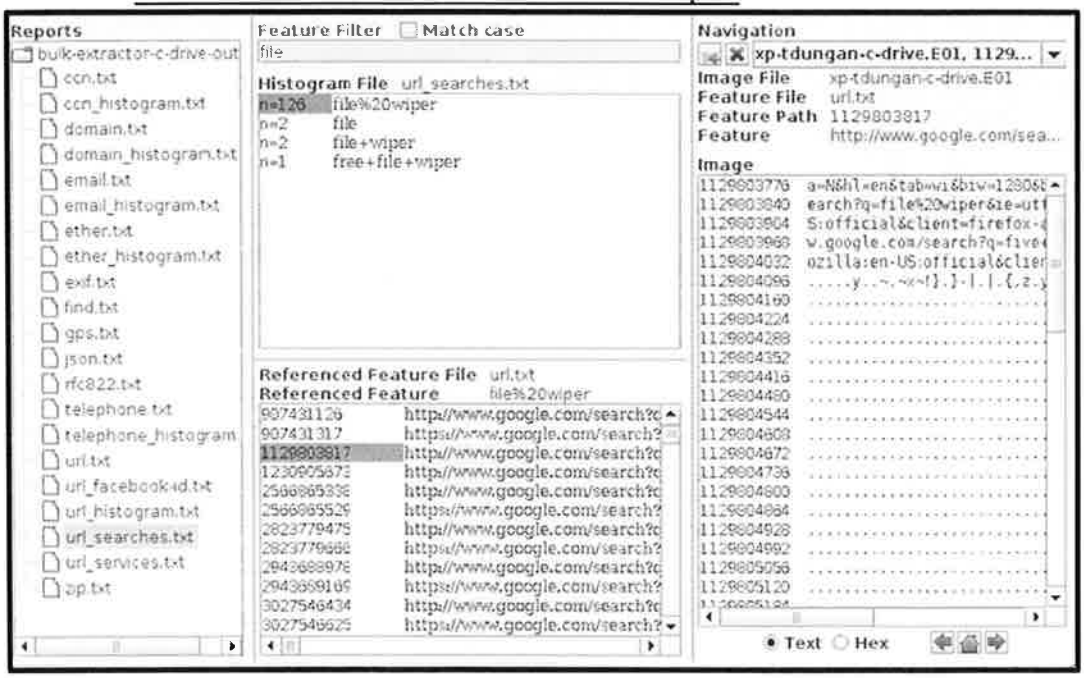


2. **What is the first hit and byte offset in the find.txt report that is found from the hibernation file?** (Note: One of bulk_extractors exciting features is the ability to examine the content of the compressed ram image embedded in the hiberfil.sys files. Once the hiberfil.sys data stream is detected, it will extract the streams and make it easier to process. This question shows some of the content extracted from a hibernation file data location)
    - **10187309832-HIBER-55539 `\dllhost\svchost.exe`**



3. How many times was the URL search term `file wiper` (`file+wiper` or `file%20wiper`) found in the disk image?

- **2 times based on the histogram output of url searches for file+wiper. You will also notice another 126 hits for the hit file%20wiper.**



4. Are any valid Credit Card Numbers found in the image?

   a. **What type of file are many of them found in? (Review: In FOR408 we learned that the new Office 2007/2010 format for documents that end in "x" (e.g. docx, pptx, xlsx) are actually**

zipfiles with embedded xml data. If the data is found in a zipfile, it is possible that it could be an office document.

i. **Many possible credit card numbers found in the image. Many of the found in a zip file embedded in XML which could mean that the file was part of an office 2007+ document.**

b. What is the byte offset of the beginning of the zipfile?

i. **1619706**

5. Using the url_services.txt file, perform a search for the IP address used for the beacon 199.73.28.114.

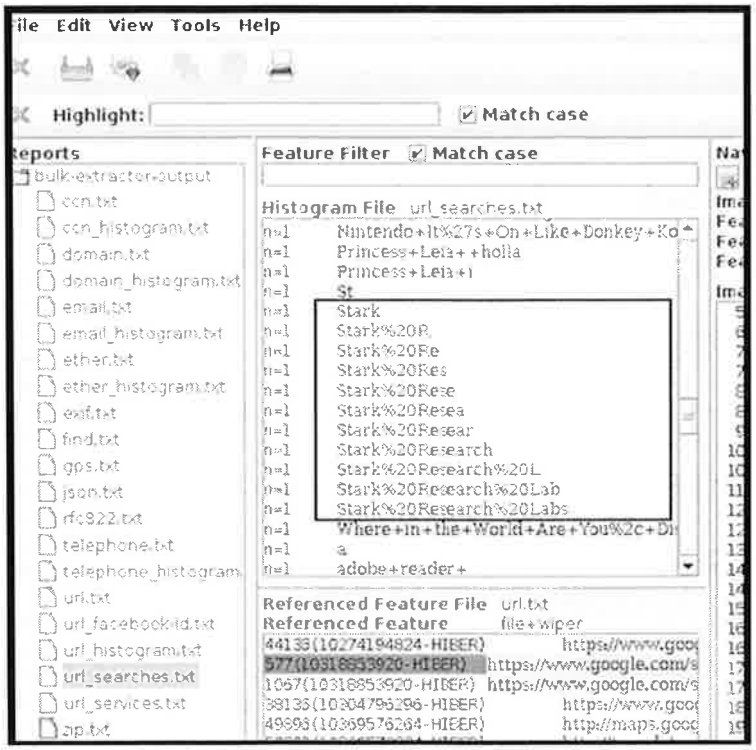
a. How many total hits were there for this IP address and for what port address?

i. **252 hits for port 443**
ii. **9 hits for port 8000**

b. Based on the "Referenced Feature File" window pane below in your BEViewer output, what application was specifically using this port and IP address? (Best Guess)

i. **Hyvy.exe (multiple references to port 443 and this IP)**

6. Why do you think the following output exists inside of the url_searches.txt file?



i. **<u>This is common in the autocomplete in the Google search bar. It recorded the autocomplete format as the user was typing it in. This will show that this is probably user input into the Google or similar search engine.</u>**

7. Based on the **`email_histogram`**, who does the user of this machine receive e-mail from or send e-mail to the most times in the same domain? The user of the machine is `tdungan@stark-research-labs.com`

i. **It looks as though `nfury@stark-research-labs.com` appears 13057 times in the output from bulk extractor. `nromanoff@stark-research-labs.com` appears 15038 times.**

Exercise – Key Takeaways

- Many possible credit card numbers found in the image. Many of them were found in a zip file embedded in XML which could mean that the file was part of a word document.
- Need to determine what file points to cluster 1619706
- Hyvy.exe (multiple references to port 443 and this IP)
- bulk_extractor has the ability to scan through hibernation files and compressed data
- bulk_extractor has the ability to help you profile the user on the system by examining searches, histograms of e-mail addresses and URLs, and more

Exercise 17 – File Carving

Objectives

- Extract Unallocated Space from the **xp-tdungan-c-drive** image
- Select signatures from the carving configuration file
- Add specific signatures (.exe and .pf) to the carving configuration file
- Carve files out of unallocated space
- Analyze the files recovered from unallocated space

Exercise Preparation

1. Please ensure your evidence is mounted correctly at **/mnt/windows_mount** if not, please follow **Exercise 2 – Mounting Evidence Using SIFT** again to mount your evidence.

```
# cd /mnt/windows_mount

# ls
```

```
root@SIFT-Workstation:/mnt/windows_mount# ls
$AttrDef        Documents and Settings  MSOCache              System Volume Information
AUTOEXEC.BAT    $Extend                 NTDETECT.COM          Temp
$BadClus        hiberfil.sys            ntldr                 $UpCase
$Bitmap         IO.SYS                  pagefile.sys          $Volume
$Boot           $LogFile                Program Files         WINDOWS
boot.ini        $MFTMirr                RECYCLER
CONFIG.SYS      MSDOS.SYS               $Secure
```

If you do not see the above, then please follow directions in Exercise 2 – Mounting Evidence Using SIFT

Carve Files from Unallocated Space – Step-by-Step Preparation

1. Extract Unallocated Space From your Disk Image and protect the new image by setting the immutable bit using **`chattr +i`**

```
# cd /cases/xp-tdungan-c-drive

# blkls xp-tdungan-c-drive.E01 > xp-tdungan-c-drive.blkls
```

2. Edit your **`foremost.conf`** file

```
# gedit /usr/local/etc/foremost.conf
```

Uncomment the following signatures to carve out:

```
gif
jpg
bmp
tif
doc
pst
pdf
htm
```

```
dat (index.dat)
lnk
chm
rdp
info2
zip
ico
rar
```

3. Below are the magic number signatures for (Prefetch and EXE) files; add them to your **`foremost.conf`** config file.

```
exe     y     80000     \x4d\x5a\x90\x00
pf      y     80000     ?\x00\x00\x00\x53\x43\x43\x41
```

4. Determine the cluster size of the image using **fsstat.**

```
# fsstat xp-tdungan-c-drive.E01
```

- **`Cluster Size =`** 4096

5. Execute Foremost with the edited configuration file against unallocated space.

```
# cd /cases/xp-tdungan-c-drive

# foremost -q -b 4096 -o foremost -c /usr/local/etc/foremost.conf
xp-tdungan-c-drive.blkls
```

6. Execute pe_carve.py to compare against the foremost data carve looking for .exe files

```
# mkdir pe_carve

# cd /cases/xp-tdungan-c-drive/pe_carve/

# ln -s ../xp-tdungan-c-drive.blkls xp-tdungan-c-drive.blkls

# pe_carve.py xp-tdungan-c-drive.blkls
```

Analyze Carved Files from Unallocated Space –Questions

1. **Analyze the EXE files**

   - **How many recovered executables are broken or false positives?**

To determine carved .exe that give errors

```
# cd /cases/xp-tdungan-c-drive/foremost/exe/

# exiftool 00000041.exe (examine the output of this tool)

# exiftool -csv * | grep -i error

# exiftool -csv * | grep -i error | wc -l
```

(Note the `-l` is a lowercase "L" not a "1")

   - 941

   - **How many executables you recovered are most likely true executables?**

```
To determine total executables in the foremost directory

# ls

# ls | wc -l

(Note the -l is a lowercase "L" not a "1")
```

- 1786

```
# cd /cases/xp-tdungan-c-drive/pe_carve

# ls -la

# ls | wc -l
```

- **How many executables were recovered by pe_carve.py?**
  - _____

- **Run a virus scan against the directory from your Windows system (Microsoft Windows Defender or ClamAV is fine). Do any `.exe` files get flagged as malware?**

```
# clamscan -i /cases/xp-tdungan-c-drive/foremost/exe

# clamscan -i /cases/xp-tdungan-c-drive/pe_carve/
```

- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____
- _____

- **Why do we get different results between pe_carve.py and the foremost.exe carve?**
  - _____
  - _____
  - _____

- Run **`exiftool`** against some of the executables that were flagged in either the **`pe_carve`** directory or the **`foremost`** directory? What is some of the most important data to pull from the output?
  - _____
  - _____
  - _____
  - _____
  - _____

- **Does this information match any existing executables on the system?**

```
# exiftool /mnt/windows_mount/Documents\ and\
Settings/tdungan/Local\ Settings/Temp/a.exe
```

  - _____

2. Analyze the recovered `PF` files

- Run the prefetch analyzer **`pf`** against some of the files.

```
# cd /cases/xp-tdungan-c-drive/foremost/pf/

# for i in *.pf; do pf -v $i; done | less
```

- Extract the data so you can parse it easily in EXCEL.

```
# cd /cases/xp-tdungan-c-drive/foremost/pf/

# for i in *.pf; do pf -csv $i; done | grep .pf, | cut -d,
-f2,3,4,5 > /cases/xp-tdungan-c-drive/recovered-pf-analysis.csv
```

Open .csv file for analysis in Excel from your Windows System accessing it via
`\\siftworkstation\cases\xp-tdungan-c-drive`

- **Based on the recovered `.pf` files, how many times did a prefetch file for `a.exe` show up in the csv file?**
  - 13
- **Based on the recovered `.pf` files, when was the last time that PE.EXE was executed?**
  - 4/6/2012 13:43:20
- **Based on the recovered `.pf` files, how many total times was PE.EXE executed?**
  - 28
- **Based on the `.pf` file found in C:\WINDOWS\Prefetch, when was the last time that PE.EXE was executed?**

```
# cd /mnt/windows_mount/WINDOWS/Prefetch/

# pf -v PE.EXE-0DC593C2.pf | grep run
```

  - 4/06/2012 19:22:20
- **Based on the `.pf` file found in C:\WINDOWS\Prefetch, how many total times was PE.EXE executed?**
  - 30

- Based on the comparison between the .pf file recovered for pe.exe and the .pf found in **C:\Windows\Prefetch**, how many times was **pe.exe** executed between **13:43** and 19:22 UTC on **04/06/2012**?

  - 3
  - _____
  - _____
  - _____

3. *OUT OF CLASS EXTRA QUESTIONS* **Analyze the LNK files**
   - Look at the LNK Files in "Thumbnail Mode" from your windows OS **\\siftworkstation\cases\xp-tdungan-c-drive\foremost\lnk**
   - **What can you observe by looking at the thumbnails?**
     - _____
   - Check validity of LNK files using **EXIFTOOL**

```
# cd /cases/xp-tdungan-c-drive/foremost/lnk/

# exiftool * | less
```

   - **Can you determine the original path of each of the files and what they pointed to?**

```
# exiftool * | grep Local\ Base\ Path
```

   - **Which programs (.exe) were executed the most?**

```
# exiftool * | grep Local\ Base\ Path | cut -d":" -f2,3 | sort
| uniq -c | sort -n
```

     - _____
     - _____
   - **How many times does the "Dossier – Dr Myron MacLain.docx" appear as a LNK file that was recovered?**
     - _____

- **What is the file name of the recovered LNK file that points to the "Dossier – Dr Myron MacLain.docx?" (note: Should look something like 00000182.lnk)**

```
# exiftool * | grep -A 2 -B 21 Myron
```

  - _____

- **Based on extracting the information from the LNK file, what are the Creation, Access, and Modification dates of the "`C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Documents\Dossier - Dr Myron MacLain.docx`" file?**

  - Create Date = _____
  - Access Date = _____
  - Modify Date = _____

- **What time zone are these times stored in?**

  - _____

- EXTRA - Run log2timeline against directory of LNK files and analyze the results.

```
# log2timeline -f win_link -r -z EST5EDT -w /cases/xp-tdungan-c-
drive/recovered_lnk_files.csv /cases/xp-tdungan-c-
drive/foremost/lnk

# l2t_process -b /cases/xp-tdungan-c-drive/recovered_lnk_files.csv
> /cases/xp-tdungan-c-drive/timeline_lnk_files.csv
```

  - Open the csv file found `\\siftworkstation\cases\xp-tdungan-c-drive\timeline_lnk_files.csv` in Excel

  - **According to this timeline, what is the last time of modification for the "Dossier – Dr Myron MacLain.docx?"**

    - _____

- **Why would inclusion of this data make your timeline more complete?**
  - _____
  - _____
  - _____

Analyze ***Carved Files from Unallocated Space –Questions with Step-by-Step Guide***

1. Analyze the EXE files
   a. **How many recovered executables are broken or false positives?**

- **940**

```
# cd /cases/xp-tdungan-c-drive/foremost/exe/
# exiftool -csv * | grep -i error
# exiftool -csv * | grep -i error | wc -l
```

(Note the `-l` is a lowercase "`L`" not a "`1`"

```
942
```

- How many executables you recovered are most likely true executables?

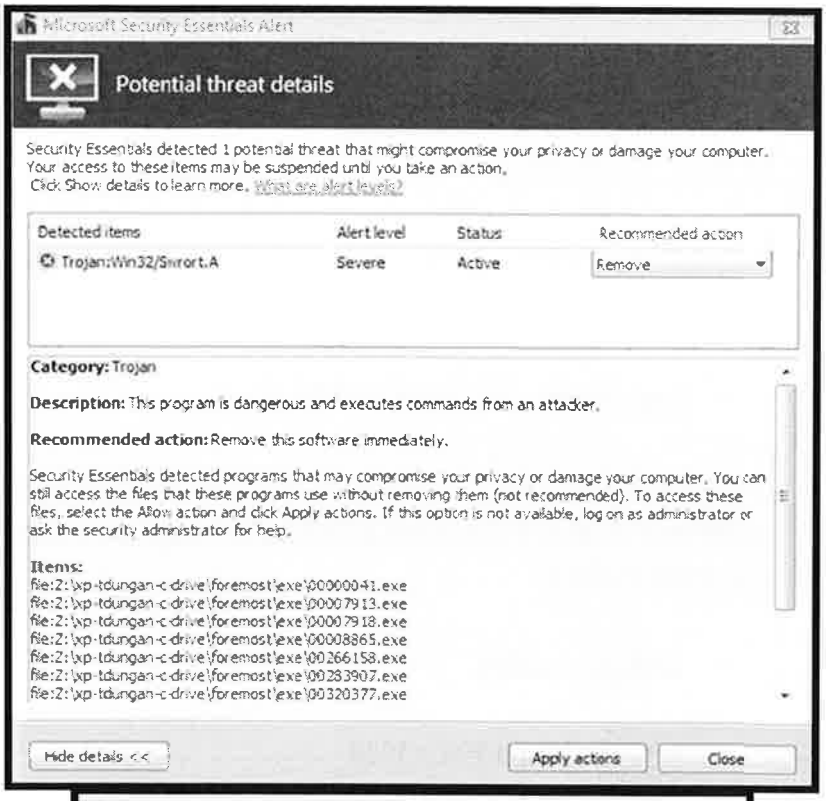
- **1956-942 = 1014**

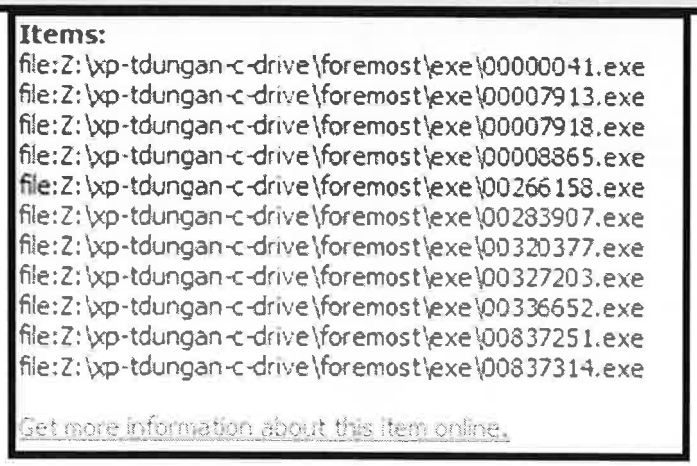
```
# ls
# ls | wc -l
```

(Note the `-l` is a lowercase "`L`" not a "`1`")

```
1956
```

- Run a virus scan against the directory? Do any **`.exes`** get flagged?





- **Why do we get different results between pe_carve.py and the foremost.exe carve?**

  - **Two different carving techniques resulted in different sets of files being retrieved. Foremost used the MZ header to carve out files at a static length (80,000 bytes). Pe carve.py scanned unallocated space looking for "This program" and then determined the file size of the executable. While both ended up producing files, it is likely both are good to examine. No tool is perfect, but it is always good to have options. Since .exe carving could be crucial to your case, we felt it best to expose you to several options of scanning**

for executables that could result in finding a key artifact in one of your upcoming cases.

- Run `exiftool` against some of the executables that were flagged. What is some of the most important data to pull from the output?
    - Time Stamp = This is the timestamp when the file was compiled
        - Time Stamp = 2011:10:13 04:19:53+00
    - Code Size and Initialized Data Size = These two values will help you determine the probable original file size when added together. While not exact, it is the best guess you might achieve.
        - Code Size = 5120
        - Initialized Data Size = 3584

```
ExifTool Version Number         : 8.10
File Name                       : 00007913.exe
Directory                       : .
File Size                       : 78 kB
File Modification Date/Time     : 2012:04:27 18:14:44+00:00
File Permissions                : rw-r--r--
File Type                       : Win32 EXE
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2011:10:13 04:19:53+00:00
PE Type                         : PE32
Linker Version                  : 9.0
Code Size                       : 5120
Initialized Data Size           : 3584
Uninitialized Data Size         : 0
Entry Point                     : 0x173f
OS Version                      : 6.1
Image Version                   : 6.1
Subsystem Version               : 5.1
Subsystem                       : Windows GUI
```

- Does this information match any existing executables on the system?
    - Yes. The file `a.exe` matches (Time Stamp, Code Size, Initialized Data Size, and more)

```
# exiftool /mnt/windows_mount/Documents\ and\ Settings/tdungan/Local\
Settings/Temp/a.exe
```

```
root@SIFT-Workstation:/# exiftool /mnt/windows_mount/Documents\ and\ Settings/t
dungan/Local\ Settings/Temp/a.exe
ExifTool Version Number         : 8.10
File Name                       : a.exe
Directory                       : /mnt/windows_mount/Documents and Settings/tdu
ngan/Local Settings/Temp
File Size                       : 9.0 kB
File Modification Date/Time     : 2012:04:04 14:01:32+00:00
File Permissions                : rwxrwxrwx
File Type                       : Win32 EXE
MIME Type                       : application/octet-stream
Machine Type                    : Intel 386 or later, and compatibles
Time Stamp                      : 2011:10:13 04:19:53+00:00
PE Type                         : PE32
Linker Version                  : 9.0
Code Size                       : 5120
Initialized Data Size           : 3584
Uninitialized Data Size         : 0
Entry Point                     : 0x173f
OS Version                      : 6.1
Image Version                   : 6.1
Subsystem Version               : 5.1
Subsystem                       : Windows GUI
```

2. Analyze the recovered **PF** files

- Run the prefetch analyzer **pf** against some of the files, can you extract the data

```
# cd /cases/xp-tdungan-c-drive/foremost/pf/

# for i in *.pf; do pf -v $i; done | less
```

- Extract the data so you can parse it easily in EXCEL

```
# cd /cases/xp-tdungan-c-drive/foremost/pf/

# for i in *.pf; do pf -csv $i; done | grep .pf, | cut -d,
-f2,3,4,5 > /cases/xp-tdungan-c-drive/recovered-pf-analysis.csv

Open .csv file for analysis in Excel from your Windows System accessing it via
\\siftworkstation\cases\xp-tdungan-c-drive
```

- Extract the data so you can parse it easily in EXCEL
- Based on the recovered **.pf** files, how many times did a prefetch file for **a.exe** show up in the csv file?

- **a.exe had 13 .pf files recovered for it**

FILENAME	Run Count	Last Executed At:
A.EXE	run 288 times	last run: 04/05/12 05:59:51.103
A.EXE	run 289 times	last run: 04/05/12 06:02:05.575
A.EXE	run 173 times	last run: 04/07/12 01:21:39.698
A.EXE	run 290 times	last run: 04/05/12 06:04:21.452
A.EXE	run 112 times	last run: 04/03/12 05:15:37.518
A.EXE	run 113 times	last run: 04/03/12 05:17:52.125
A.EXE	run 296 times	last run: 04/05/12 06:17:48.908
A.EXE	run 279 times	last run: 04/05/12 05:39:14.727
A.EXE	run 110 times	last run: 04/06/12 23:01:53.723
A.EXE	run 270 times	last run: 04/05/12 05:18:54.969
A.EXE	run 5 times	last run: 04/03/12 01:05:22.764
A.EXE	run 2 times	last run: 04/04/12 15:25:55.906
A.EXE	run 1 times	last run: 04/04/12 12:26:24.707
ACRORD32INFO.EXE	run 2 times	last run: 04/02/12 12:54:33.205
APSDAEMON.EXE	run 6 times	last run: 04/04/12 16:41:55.120
AT.EXE	run 13 times	last run: 04/05/12 17:44:55.171

- Based on the recovered `.pf` files, when was the last time that PE.EXE was executed?

  - **pe.exe was last executed at 04/06/2012 13:43:20 UTC**

- Based on the recovered `.pf` files, how many total times was PE.EXE executed?

  - **pe.exe runcount was 27**

1	FILENAME	Run Count	Last Executed At:
2	A.EXE	run 173 times	last run: 04/07/12 01:21:39.698
3	WMIPRVSE.EXE	run 543 times	last run: 04/07/12 01:03:14.267
4	GOOGLEUPDATE.EXE	run 6386 times	last run: 04/06/12 23:58:00.133
5	A.EXE	run 110 times	last run: 04/06/12 23:01:53.723
6	JAVAW.EXE	run 8 times	last run: 04/06/12 19:12:10.477
7	WMIPRVSE.EXE	run 542 times	last run: 04/06/12 19:06:52.005
8	DFRGNTFS.EXE	run 75 times	last run: 04/06/12 17:23:53.798
9	WUAUCLT.EXE	run 797 times	last run: 04/06/12 16:39:40.282
10	USERINIT.EXE	run 1517 times	last run: 04/06/12 16:38:35.411
11	USERINIT.EXE	run 1516 times	last run: 04/06/12 16:38:35.052
12	PE.EXE	run 27 times	last run: 04/06/12 13:43:20.168
13	MCSCRIPT_INUSE.EXE	run 402 times	last run: 04/06/12 07:00:11.009
14	WMIPRVSE.EXE	run 541 times	last run: 04/06/12 04:45:26.638
15	VERCLSID.EXE	run 105 times	last run: 04/06/12 04:01:18.932
16	WMIPRVSE.EXE	run 540 times	last run: 04/06/12 01:14:44.047
17	HELPSVC.EXE	run 208 times	last run: 04/06/12 01:14:28.190
18	MCSCRIPT_INUSE.EXE	run 401 times	last run: 04/05/12 21:10:04.532
19	CMD.EXE	run 54 times	last run: 04/05/12 17:48:55.939
20	AT.EXE	run 13 times	last run: 04/05/12 17:44:55.171
21	IEXPLORE.EXE	run 5 times	last run: 04/05/12 15:11:08.743
22	WINLOGON.EXE	run 65 times	last run: 04/05/12 14:41:58.318
23	CSRSS.EXE	run 79 times	last run: 04/05/12 14:41:58.068

- Based on the `.pf` files found in C:\WINDOWS\Prefetch, when was the last time that PE.EXE was executed?
  - **pe.exe was last executed at 04/06/2012 19:22:20 UTC**

- Based on the `.pf` files found in C:\WINDOWS\Prefetch, how many total times was PE.EXE executed?
  - **pe.exe runcount was 30**

```
# cd /mnt/windows_mount/WINDOWS/Prefetch/

# pf -v PE.EXE-0DC593C2.pf | grep run
```

```
PE.EXE, run 30 times, last run: 04/06/12 19:22:20.316
```

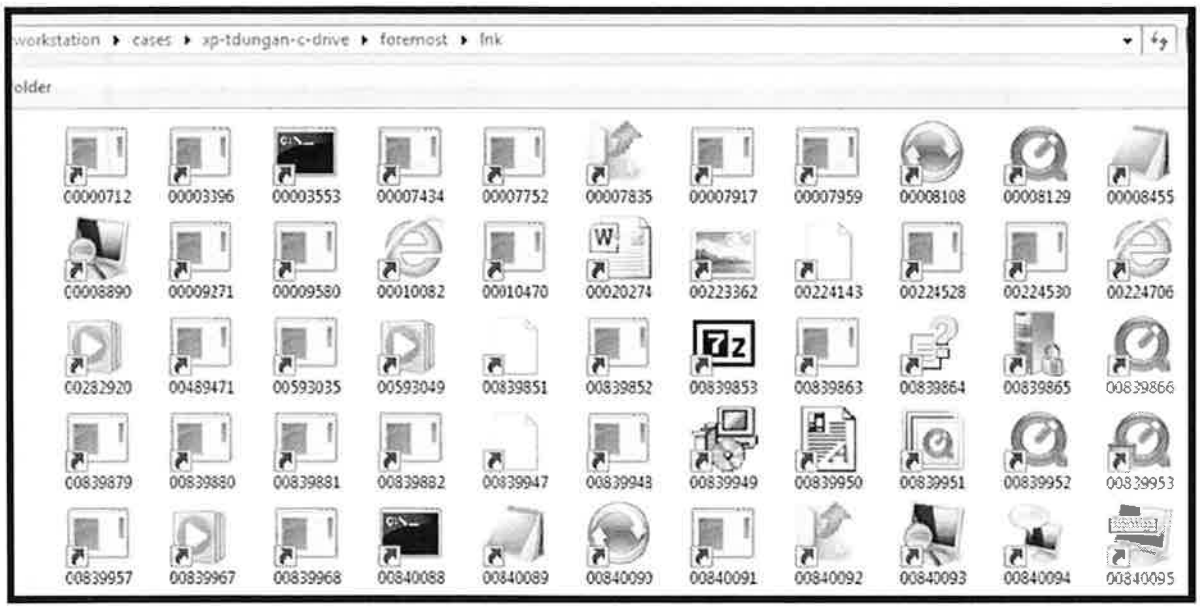
- Based on the comparison between the .pf file recovered for pe.exe and the .pf found in **C:\Windows\Prefetch**, how many times was **pe.exe** executed between **13:43** and 19:22 UTC on **04/06/2012**?

  - **pe.exe (recovered .pf) had a runcount of 27**
  - **pe.exe (C:\WINDOWS\Prefetch) had a runcount of 30**
  - **pe.exe was executed 2 more times (runcount 28 and 29)**

3. *OUT OF CLASS EXTRA QUESTIONS* **Analyze the LNK files**

   - Look at the LNK Files in "Thumbnail Mode." What do you see?

     - **Many of the LNK files have an ICON associated with them. In some cases you can even tell the exact application that is associated with it (e.g. 7zip, QuickTime, or Internet Explorer)**



   - Check validity of LNK files using EXIFTOOL

```
# cd /cases/xp-tdungan-c-drive/foremost/lnk/

# exiftool * | less
```

   - Can you determine the original path of each of the files and what they pointed to?

```
# exiftool * | grep Local\ Base\ Path
....
Local Base Path                 : C:\Program
Files\Messenger\msmsgs.exe
Local Base Path                 : C:\WINDOWS\system32\cmd.exe
Local Base Path                 : C:\WINDOWS\system32\rcimlby.exe
Local Base Path                 : C:\WINDOWS\explorer.exe
Local Base Path                 : C:\Program
Files\Google\Chrome\Application\18.0.1025.151\Installer\setup.exe
Local Base Path                 : C:\WINDOWS\system32\tourstart.exe
Local Base Path                 : C:\WINDOWS\system32\mobsync.exe
Local Base Path                 : C:\WINDOWS\system32\notepad.exe
Local Base Path                 : C:\WINDOWS\system32\narrator.exe
Local Base Path                 : C:\WINDOWS\system32\osk.exe
Local Base Path                 : C:\WINDOWS\system32\magnify.exe
...
```

- Which programs (`.exe`) were executed the most?
  - 4 \Program Files\Internet Explorer\iexplore.exe
  - 4 \Program Files\Outlook Express\msimn.exe

```
# exiftool * | grep Local\ Base\ Path | cut -d":" -f3 | sort | uniq -c
| sort -n
....
      2 \WINDOWS\system32\notepad.exe
      2 \WINDOWS\system32\osk.exe
      3 \Program Files\Google\Chrome\Application\chrome.exe
      3 \Program Files\Paragon Software\Partition Manager 11
Personal\program\launcher.exe
      3 \Program Files\Windows Media Player\wmplayer.exe
      3 \WINDOWS\explorer.exe
      3 \WINDOWS\system32\rcimlby.exe
      3 \WINDOWS\system32\tourstart.exe
      4 \Program Files\Internet Explorer\iexplore.exe
      4 \Program Files\Outlook Express\msimn.exe
```

- How many times does the "Dossier – Dr Myron MacLain.docx" appear as a LNK file that was recovered?
  - 1 Time

- What is the file name of the recovered LNK file that points to the "Dossier – Dr Myron MacLain.docx?"

```
# exiftool * | grep -A 2 -B 21 Myron
======== 00020274.lnk
ExifTool Version Number         : 8.10
File Name                       : 00020274.lnk
Directory                       : .
File Size                       : 4.9 kB
File Modification Date/Time     : 2012:04:27 18:14:44+00:00
File Permissions                : rw-r--r--
File Type                       : Windows Shortcut
MIME Type                       : application/octet-stream
Flags                           : IDList, LinkInfo, RelativePath,
WorkingDir, Unicode
File Attributes                 : Archive
Create Date                     : 2012:03:08 22:10:54+00:00
Access Date                     : 2012:03:12 21:27:54+00:00
Modify Date                     : 2012:03:12 21:27:52+00:00
Target File Size                : 140751
Icon Index                      : (none)
Run Window                      : Normal
Hot Key                         : (none)
Target File DOS Name            : DOSSIE~1.DOC
Drive Type                      : Fixed Disk
Volume Label                    :
Local Base Path                 : C:\Documents and Settings\tdungan\My
Documents\Alloy Research\Detailed Documents\Dossier - Dr Myron
MacLain.docx
Relative Path                   : ..\My Documents\Alloy
Research\Detailed Documents\Dossier - Dr Myron MacLain.docx
Working Directory               : C:\Documents and Settings\tdungan\My
Documents\Alloy Research\Detailed Documents
Machine ID                      : wks-winxp32bit
```

  - **`00020274.lnk` is the file that contains the information related to "`Dossier - Dr Myron MacLain.docx`"**

- Based on extracting the information from the LNK file, what is the Creation, Access, and Modification dates of the "`C:\Documents and Settings\tdungan\My Documents\Alloy Research\Detailed Documents\Dossier - Dr Myron MacLain.docx`" file

```
Create Date                     : 2012:03:08 22:10:54+00:00
Access Date                     : 2012:03:12 21:27:54+00:00
Modify Date                     : 2012:03:12 21:27:52+00:00
```

- What time zone are these times stored in?
  - **UTC**
- Run log2timeline against directory of LNK files and analyze the results.

```
# log2timeline -f win_link -r -z EST5EDT -w /cases/xp-tdungan-c-
drive/recovered_lnk_files.csv /cases/xp-tdungan-c-drive/foremost/lnk

# l2t_process -b /cases/xp-tdungan-c-drive/recovered_lnk_files.csv >
/cases/xp-tdungan-c-drive/timeline_lnk_files.csv
```

- Open the csv file found `\\siftworkstation\cases\xp-tdungan-c-drive\timeline_lnk_files.csv` in Excel
- According to this timeline, what is the last time of modification for the "Dossier – Dr Myron MacLain.docx?"
  - **Modification Time = 03/12/2012 17:27:52 EST5EDT**

date	time	MACB	short
3/12/2012	17:27:52	M...	C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Documents/Dossier - Dr Myron MacLain.docx
3/12/2012	17:27:54	.A..	C:/Documents and Settings/tdungan/My Documents/Alloy Research/Detailed Documents/Dossier - Dr Myron MacLain.docx

- Why would inclusion of this data make your timeline more complete?
  - **Since most of this data originated from deleted/recovered LNK files, it might give us additional timestamps in our overall timeline. It might not make a difference in every case, but it surely will make the case data more complete and accurate.**

Exercise – Key Takeaways

- **Foremost does a decent job of recovering key file types**
- **We can easily find malware by scanning recovered .exe files**
- **Certain file types you can run the targeted log2timeline against (such as LNK and EXE files)**
- **Use basic analysis capabilities of the SIFT Workstation to examine recovered files and determine if they are useful to your case**
- **No new data that we didn't already know, but if this system once had active malware on it and it was subsequently deleted, we would have a decent chance of recovering that malware from unallocated space as well as finding evidence of execution via recovered .pf and .lnk files**

This page intentionally left blank.

Exercise 18 – NTFS Filesystem Challenge

Objectives

- Determine the filename of a file if you are given a fragment of data from another tool such as bulk_extractor
- Gain experience using the Sleuthkit
- Understand how to map between the data layer, the metadata layer, and the filename layer

Exercise Preparation

1. Please ensure your evidence is mounted correctly at **`/mnt/windows_mount`** if not, please follow **Exercise 2 – Mounting Evidence Using SIFT** again to mount your evidence.

```
# cd /mnt/windows_mount

# ls
```

```
root@SIFT-Workstation:/mnt/windows_mount# ls
$AttrDef        Documents and Settings  MSOCache           System Volume Information
AUTOEXEC.BAT    $Extend                 NTDETECT.COM       Temp
$BadClus        hiberfil.sys            ntldr              $UpCase
$Bitmap         IO.SYS                  pagefile.sys       $Volume
$Boot           $LogFile                Program Files      WINDOWS
boot.ini        $MFTMirr                RECYCLER
CONFIG.SYS      MSDOS.SYS               $Secure
```

If you do not see the above, then please follow directions in Exercise 2 – Mounting Evidence Using SIFT

Windows Filesystem Challenge – Questions

1. **Perform a timestomp detection check on the possible malware using `istat`**

Using istat, determine which of the following files have been timestomped.

Filename	MFT Record #	$STD_INFO Creation	$FN Creation	Possible Timestomping? (Y/N)
C:\WINDOWS\system32\dllhost\svchost.exe				
C:\WINDOWS\system32\hyvy.exe	5237	3/31/2003 12:00	4/3/2012 6:30	Y
C:\WINDOWS\system32\hydrakatz.exe	4733	3/31/2003 12:00	4/3/2012 5:49	Y
C:\WINDOWS\system32\sekurlsa.dll	3260	3/31/2003 12:00	4/3/2012 5:32	Y
C:\WINDOWS\system32\spinlock.exe		4/4/2012 5:07:04 pm	4/4/2012 8:04	N

For each file above:

Determine MFT Record #, $SI, and $FN for each file. Finally, assess whether or not you think Timestomping possibly occurred. In this example, we will look at svchost.exe

```
# ls -li /mnt/windows_mount/WINDOWS/system32/dllhost/svchost.exe

    3022 -rwxrwxrwx 1 root root 102400 2008-04-14 00:12 svchost.exe
```

Once you know the MFT Record Number, the first number in the line, use istat to examine the contents of that record. Compare the creation time of the $FILENAME Attribute with the creation time of the $STANDARD_INFORMATION attribute. If the $STANDARD_INFORMATION creation time occurs prior to the $FILENAME creation time, there is possible timestomping occurring. (Note: If you use the $FILENAME creation time as a timeline pivot point, it would help determine if the assessment of timestomping is correct. In this case, the $FILENAME creation time occurs near the same time that the **`winclient.reg`** file and the **dllhost** directory were created on 2 April 2012 ~2030.

```
# istat /cases/xp-tdungan-c-drive/xp-tdungan-c-drive.E01 3022

MFT Entry Header Values:
Entry: 3022         Sequence: 10
$LogFile Sequence Number: 2199495480
Allocated File
Links: 1

$STANDARD_INFORMATION Attribute Values:
Flags: Archive
Owner ID: 0
Security ID: 392  ()
Last User Journal Update Sequence Number: 1894779272
Created:  Mon Mar 31 12:00:00 2003
File Modified: Mon Apr 14 00:12:36 2008
MFT Modified:  Fri Apr  6 19:07:16 2012
Accessed: Fri Apr  6 19:07:16 2012

$FILE_NAME Attribute Values:
Flags: Archive
Name: svchost.exe
Parent MFT Entry: 3015   Sequence: 15
Allocated Size: 102400         Actual Size: 102400
Created:  Tue Apr  3 00:35:02 2012
File Modified: Tue Apr  3 00:35:03 2012
MFT Modified:  Tue Apr  3 00:35:03 2012
Accessed: Tue Apr  3 00:35:03 2012

Attributes:
Type: $STANDARD_INFORMATION (16-0)   Name: N/A   Resident   size: 72
Type: $FILE_NAME (48-5)   Name: N/A   Resident   size: 88
Type: $DATA (128-4)   Name: N/A   Non-Resident   size: 102400
init_size: 102400
```

Filename	MFT Record #	$STD_INFO Creation	$FN Creation	Possible Timestomping? (Y/N)
C:\WINDOWS\system32\dllhost\svchost.exe	3022	3/31/03 12:00:00	4/3/12 00:35:02	YES

2. Data FROM Stream Based Extraction Exercise

- **Are any valid Credit Card Numbers found in the image?**
  - **What type of file are many of them found in?**
    - **Many possible credit card numbers found in the image. Many of the found in a zip file embedded in XML which could mean that the file was part of a word document.**
  - What is the byte offset of the beginning of the zipfile?
    - **1619706**
- **What file name is found with CC data that is part of byte offset 1619706?**

CC-Buchsbaum-Accounts.xlsx

1. Determine cluster size
   - Cluster size =`# fsstat xp-tdungan-c-drive.E01`
   - **Cluster size =** 4096
2. Determine cluster address using byte offset and cluster size
   - Original_Cluster_Num = **`(byte offset) / (cluster size)`**
   - **Original_Cluster_Num =** 394
3. Find the INODE_NUM that points to the cluster using ifind
   - MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d {ORIGINAL_CLUSTER_NUM}`
   - **MFT_Num =** 313
4. Ensure MFT Record Number points to cluster using istat
   - `# istat xp-tdungan-c-drive.E01 {MFT_NUM}`
5. Use icat to extract the file for analysis
   - `# icat xp-tdungan-c-drive.E01 {MFT_NUM} > /cases/xp-tdungan-c-drive/sample_recovered`
6. Using MFT Record number determine FILENAME using ffind

- Filename = `# ffind xp-tdungan-c-drive.E01 {MFT_NUM}`

- **Filename =** _____

3. **What file name is found based on the following bulk_extractor hit?**

```
25382928 HYDRAKATZ.EXE
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num =** _____

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d {ORIGINAL_CLUSTER_NUM}`

   b. **MFT_Num =** _____

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 {MFT_NUM}`

   b. **Filename =** _____

4. **What file name is found based on the following bulk_extractor hit?**

```
235565072 HYVY.EXE
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num =** _____

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d {ORIGINAL_CLUSTER_NUM}`

   b. **MFT_Num =** _____

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 {MFT_NUM}`

   b. **Filename =** _____

5. **What file name is found based on the following bulk_extractor hit?**

```
367259794 EXFIL.pst
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset) / (cluster size)`

   b. **Original_Cluster_Num =** 89663

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d {ORIGINAL_CLUSTER_NUM}`

   b. **MFT_Num =** 8245

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 {MFT_NUM}`

   b. **Filename =** _____

6. **What file name is found based on the following bulk_extractor hits?**

```
15010878359 <li><a href="HYDRAkatz.exe">HYDRAkatz.exe</a>
15010878405 <li><a href="HYvy.exe">HYvy.exe</a>
15010878473 <li><a href="sekurlsa.dll">sekurlsa.dll</a>
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset) / (cluster size)`

   b. **Original_Cluster_Num =** 3614768

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d {ORIGINAL_CLUSTER_NUM}`

   b. MFT_Num = 3150

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 {MFT_NUM}`

   b. **Filename =** _cacHAR~1_

7. NINJA QUESTION: Run "jp" against the xp-tdungan-c-drive.img and answer the question below.

```
# cd /mnt/ewf_mount

# jp -image /mnt/ewf_mount/ewf1 -base10 -bodyfile > /cases/xp-tdungan-c-drive/jp-bodyfile

# mactime -d -b /cases/xp-tdungan-c-drive/jp-bodyfile -z EST5EDT > /cases/xp-tdungan-c-drive/jp-timeline.csv

OPEN TIMELINE IN CSV FORMAT VIA EXCEL AND ANSWER THE QUESTION BELOW
```

How often is a.exe created on the NTFS filesystem? Asking the question another way ->Determine the beacon interval of the svchost.exe through examining the creation date/time of a.exe on the filesystem. Remember a .exe is created each time the beacon is active and attempts a connect out. We have assumed that the "pause" interval, set to the integer 32, is how often it beacons. Can we use the `jp` output to prove it?

INTERVAL = _____ **SECONDS**

Windows Filesystem Challenge – Questions with Step-by-Step Guide

1. **Perform a timestomp detection check on the possible malware using `istat`**

Using istat, determine which of the following files have been timestomped.

Filename	MFT Record #	$STD_INFO Creation	$FN Creation	Possible Timestomping? (Y/N)
C:\WINDOWS\system32\dllhost\svchost.exe	**3022**	**3/31/03 12:00:00**	**4/3/12 00:35:02**	**YES**
C:\WINDOWS\system32\hyvy.exe	**5237**	**3/31/03 12:00:00**	**4/3/12 16:30:02**	**YES**
C:\WINDOWS\system32\hydrakatz.exe	**4736**	**3/31/03 12:00:00**	**4/3/12 15:19:49**	**YES**
C:\WINDOWS\system32\sekurlsa.dll	**3260**	**3/31/03 12:00:00**	**4/3/12 15:30:13**	**YES**
C:\WINDOWS\system32\spinlock.exe	**7793**	**4/4/12 17:04:44**	**4/4/12 17:04:44**	**NO**

For each file above:

Determine MFT Record # for each file. In this example, we will look at svchost.exe

```
# ls -li /mnt/windows_mount/WINDOWS/system32/dllhost/svchost.exe

3022 -rwxrwxrwx 1 root root 102400 2008-04-14 00:12 svchost.exe
```

Once you know the MFT Record Number, the first number in the line, use istat to examine the contents of that record. Compare the creation time of the $FILENAME Attribute with the creation time of the $STANDARD_INFORMATION attribute. If the $STANDARD_INFORMATION creation time occurs prior to the $FILENAME creation time, there is possible timestomping occurring. (Note: If you use the $FILENAME creation time as a timeline pivot point, it would help determine if the assessment of timestomping is correct. In this case, the $FILENAME creation time occurs near the same time that the **`winclient.reg`** file and the **dllhost** directory were created on 2 April 2012 ~2030.

```
# istat /cases/xp-tdungan-c-drive/xp-tdungan-c-drive.E01 3022

MFT Entry Header Values:
Entry: 3022        Sequence: 10
$LogFile Sequence Number: 2199495480
Allocated File
Links: 1

$STANDARD_INFORMATION Attribute Values:
Flags: Archive
Owner ID: 0
Security ID: 392  ()
Last User Journal Update Sequence Number: 1894779272
Created:  Mon Mar 31 12:00:00 2003
File Modified: Mon Apr 14 00:12:36 2008
MFT Modified:  Fri Apr  6 19:07:16 2012
Accessed: Fri Apr  6 19:07:16 2012

$FILE_NAME Attribute Values:
Flags: Archive
Name: svchost.exe
Parent MFT Entry: 3015   Sequence: 15
Allocated Size: 102400        Actual Size: 102400
Created:  Tue Apr  3 00:35:02 2012
File Modified: Tue Apr  3 00:35:03 2012
MFT Modified:  Tue Apr  3 00:35:03 2012
Accessed: Tue Apr  3 00:35:03 2012

Attributes:
Type: $STANDARD_INFORMATION (16-0)   Name: N/A   Resident   size: 72
Type: $FILE_NAME (48-5)   Name: N/A   Resident   size: 88
Type: $DATA (128-4)   Name: N/A   Non-Resident   size: 102400
init_size: 102400
3397543 3397544 3397545 3397546 3397547 3397548 3397549 3397550
3397551 3397552 3397553 3397554 3397555 3397556 3397557 3397558
3397559 3397560 3397561 3397562 3397563 3397564 3397565 3397566
3397567
```

Filename	MFT Record #	$STD_INFO Creation	$FN Creation	Possible Timestomping? (Y/N)
C:\WINDOWS\system32\dllhost\svchost.exe	**3022**	**3/31/03 12:00:00**	**4/3/12 00:35:02**	**YES**

Data FROM Stream Based Extraction Exercise

Are any valid Credit Card Numbers found in the image?

- **What type of file are many of them found in?**
  - **Many possible credit card numbers found in the image. Many of the found in a zip file embedded in XML which could mean that the file was part of a word document.**
- What is the byte offset of the beginning of the zipfile?
  - **1619706**

2. **What file name is found with data that is part of byte offset 1619706?**

1. Determine cluster size
   a. Cluster size =`# fsstat xp-tdungan-c-drive.E01`

   b. **Cluster size =**`4096 bytes`

2. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num= `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num =**`1619706 / 4096 = Cluster 395`

3. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d 395`

   b. **MFT_Num =** `313-128-3`

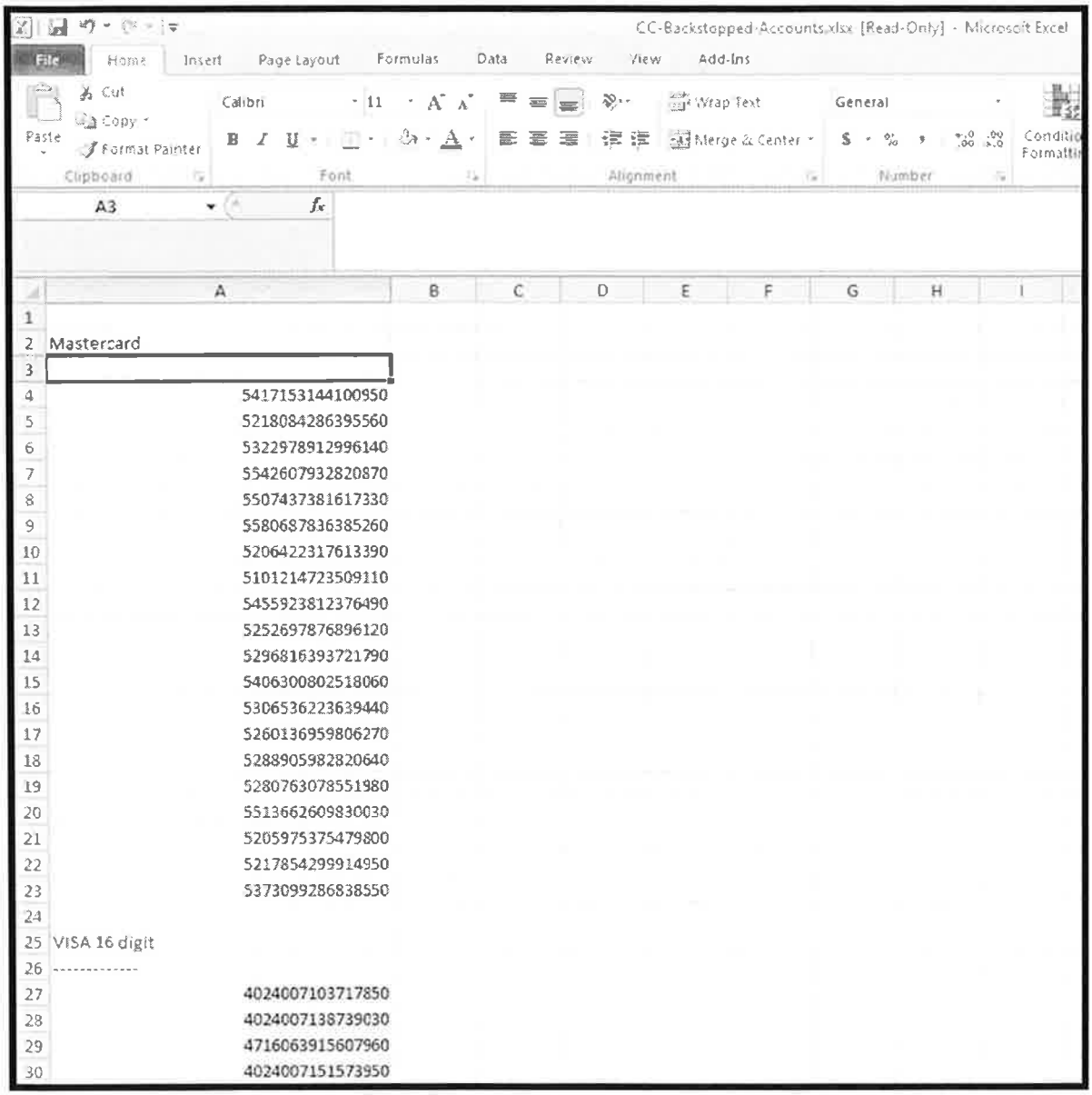
4. Ensure MFT Record Number points to cluster using istat
   a. `# istat xp-tdungan-c-drive.E01 313`

5. Use `icat` to extract the file for analysis
   a. `# icat xp-tdungan-c-drive.E01 313 > /cases/xp-tdungan-c-drive/sample_recovered_313`

6. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 313`

b. Filename =**/Documents and Settings/tdungan/My Documents/Backstopped Accounts - R&D Costs Alloy Research/CC-Backstopped-Accounts.xlsx**

Note: The above file was determined by bulk_extractor to contain credit card numbers.



3. **What file name is found based on the following bulk_extractor hit?**

```
25382928 HYDRAKATZ.EXE
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num** =`25382928 / 4096 = 6197`

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d 6197`

   b. **MFT_Num** =`4894`

3. Using MFT Record number determine FILENAME using ffind
   a. **Filename** = `# ffind xp-tdungan-c-drive.E01 4894`

   b. **Filename** =`/WINDOWS/Prefetch/HYDRAKATZ.EXE-27B49502.pf`

4. **What file name is found based on the following bulk_extractor hit?**

```
235565072 HYVY.EXE
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num** =`235565072 / 4096 = 57511`

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d 57511`

   b. **MFT_Num** =`5144`

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 5144`

   b. **Filename** =`/WINDOWS/Prefetch/HYVY.EXE-2A94EF14.pf`

5. **What file name is found based on the following bulk_extractor hit?**

```
367259794 EXFIL.pst
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num** =`367259794 / 4096 = 89663`

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d 89663`

   b. **MFT_Num** =`8245`

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 8245`

   b. **Filename** =`/Documents and Settings/vibranium/Local Settings/Application Data/Microsoft/Outlook`

6. **What file name is found based on the following bulk_extractor hits?**

```
15010878359 <li><a href="HYDRAkatz.exe">HYDRAkatz.exe</a>
15010878405 <li><a href="HYvy.exe">HYvy.exe</a>
15010878473 <li><a href="sekurlsa.dll">sekurlsa.dll</a>
```

1. Determine cluster address using byte offset and cluster size
   a. Original_Cluster_Num = `(byte offset)/(cluster size)`

   b. **Original_Cluster_Num** = `15010878405 / 4096 = 3664765`

2. Find the INODE_NUM that points to the cluster using ifind
   a. MFT_Num = `# ifind xp-tdungan-c-drive.E01 -d 3664765`

   b. **MFT_Num** = `3150`

3. Using MFT Record number determine FILENAME using ffind
   a. Filename = `# ffind xp-tdungan-c-drive.E01 3150`

b. Filename = /Documents and Settings/tdungan/Local Settings/Application Data/Mozilla/Firefox/Profiles/uimccx5n.default/Cache/_CACHE_001_

7. NINJA QUESTION: Run "jp" against the xp-tdungan-c-drive.img and answer the question below.

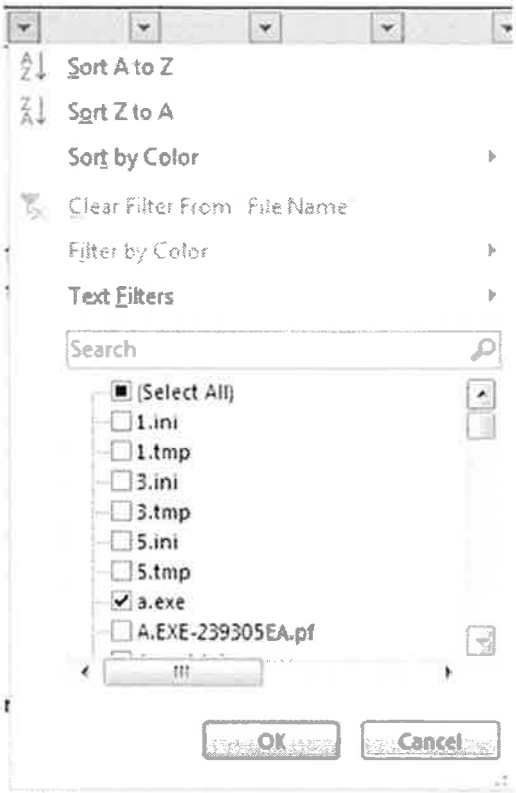
```
# cd /mnt/ewf_mount

# jp -image /mnt/ewf_mount/ewf1 -base10 -bodyfile > /cases/xp-tdungan-c-drive/jp-bodyfile

# mactime -d -b /cases/xp-tdungan-c-drive/jp-bodyfile -z EST5EDT > /cases/xp-tdungan-c-drive/jp-timeline.csv

OPEN TIMELINE IN CSV FORMAT VIA EXCEL AND ANSWER THE QUESTION BELOW
```

How often is a.exe created on the NTFS filesystem? Asking the question another way ->Determine the beacon interval of the svchost.exe through examining the creation date/time of a.exe on the filesystem. Remember a .exe is created each time the beacon is active and attempts a connect out. We have assumed that the "pause" interval, set to the integer 32, is how often it beacons. Can we use the `jp` output to prove it?

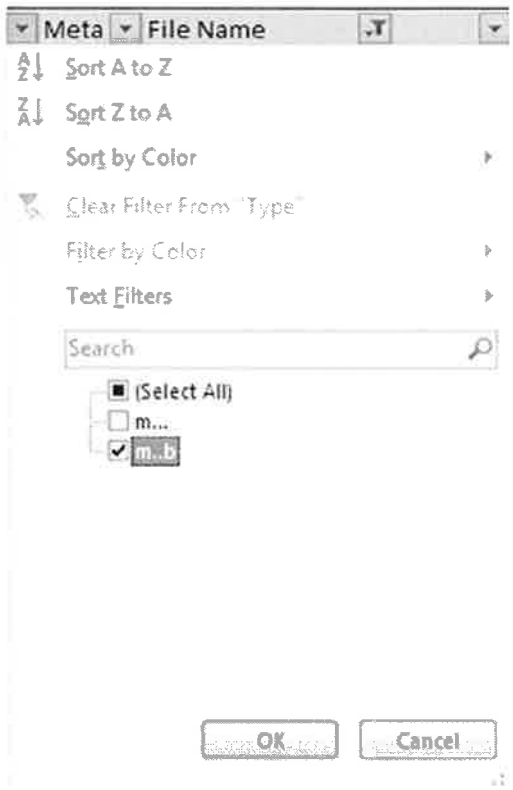
a. Open jp-timeline.csv in EXCEL
b. Hide all columns except Date, Type, Meta, File Name
c. Freeze Top Row = VIEW -> FREEZE PANES -> FREEZE TOP ROW
d. Filter Columns = HOME -> SORT & FILTER -> FILTER

	Date	Type	Meta	File Name
2	Fri Apr 06	m...	7372	a.exe
3	Fri Apr 06	m..b	7372	a.exe
4	Fri Apr 06	m...	7372	a.exe
5	Fri Apr 06	m..b	7372	a.exe
6	Fri Apr 06	m...	7372	a.exe
7	Fri Apr 06	m..b	7374	A.EXE-239305EA.pf
8	Fri Apr 06	m...	7374	A.EXE-239305EA.pf
9	Fri Apr 06	..c.	15838	1.ini
10	Fri Apr 06	..c.	15838	1.tmp

e. Filter only on "a.exe"



f. Filter only on type "m..b"



	A	B	C	D
1	Date	Type	Meta	File Name
3	Fri Apr 06 2012 19:48:00	m..b	7372	a.exe
5	Fri Apr 06 2012 19:48:33	m..b	7372	a.exe
32	Fri Apr 06 2012 19:49:06	m..b	7372	a.exe
34	Fri Apr 06 2012 19:49:39	m..b	7372	a.exe
36	Fri Apr 06 2012 19:50:13	m..b	7372	a.exe
38	Fri Apr 06 2012 19:50:46	m..b	7372	a.exe
46	Fri Apr 06 2012 19:51:19	m..b	7372	a.exe
48	Fri Apr 06 2012 19:51:52	m..b	7372	a.exe

g. Calculate the delta between each entry. Note you can use EXCEL to help you here or eyeball it. What you can easily see is that each one is almost 33-34 seconds apart. If you remember, in the \SYSTEM\CONTROLSET001\SERVICES\NETMAN\DOMAIN registry key, the "PAUSE" value was set to equal "32" seconds.

Name	Type	Data
home	REG_SZ	http://199.73.28.114/ads/
pause	REG_DWORD	0x00000020 (32)

If you add file creation and process execution on top of the interval you would probably see the file "a.exe" is created each time the beacon attempts to connect out. Below you can see the delta calculation in excel of the previous entry and the one following it. Please note it did take some hacking around EXCEL to get this to work. The reason we did it is to show that the interval is between **33-34 seconds** each time.

Date	Type	Meta	FileNan	
19:48:00	m..b	7372	a.exe	00:00:33
19:48:33	m..b	7372	a.exe	00:00:33
19:49:06	m..b	7372	a.exe	00:00:33
19:49:39	m..b	7372	a.exe	00:00:34
19:50:13	m..b	7372	a.exe	00:00:33
19:50:46	m..b	7372	a.exe	00:00:33
19:51:19	m..b	7372	a.exe	00:00:33
19:51:52	m..b	7372	a.exe	00:00:34
19:52:26	m..b	7372	a.exe	00:00:33
19:52:59	m..b	7372	a.exe	00:00:33
19:53:32	m..b	7372	a.exe	00:00:33
19:54:05	m..b	7372	a.exe	00:00:33
19:54:38	m..b	7372	a.exe	00:00:34
19:55:12	m..b	7372	a.exe	00:00:33
19:55:45	m..b	7372	a.exe	00:00:33
19:56:18	m..b	7372	a.exe	00:00:34

Exercise 19 – Challenge Prep

Objectives

1. Using all the skills we learned this week and using malware funneling methods we will begin prepping the evidence from other systems found on the Stark Research Labs Network.
2. Prepare Memory Analysis.
3. Prepare Timeline Analysis.
4. Anything else you might think of

*Exercise **Setup and Challenge Preparation Rules***

- For the final challenge each of you will be broken into teams of 3-4 individuals max. If we have an odd number then and only then will a team of 5 be allowed. Those leaving early from class should be all on one team.

- There are 3 systems that the attackers are suspected of moving laterally inside our enterprise environment at Stark Research Labs. The three systems that they are suspected to have moved to are:
  - **`Win7-64-nfury-10.3.58.6`**
  - **`Win7-32-nromanoff-10.3.58.5`**
  - **`Win2008R2-controller-10.3.58.4`**

- Each individual should select a system and will be the lead analyst for that system to prepare initially and analyze.
  - If you have more than more analysts than systems, divide the preparation tasks. One of you can start to prep memory analysis using redline, the other can create the super timeline.

- Once you have been assigned your system, begin to process the data for the challenge in the morning.

- It is assumed that you will need to keep processing through the night. Please accomplish the quicker tasks first and then the longer ones overnight.

- One final note: Since there is so much data to process, we would ask that each team ONLY process, but do not begin analysis of the data. Teams that start ahead the morning of Day 6 already having analyzed prepped case data will be penalized and in the event of a tie with another team will end up losing the challenge. Prep ONLY.

- At the beginning of the final section, Section/Day 6, the instructor will hand out the last exercise. (Note: It is NOT in your book. To win the challenge, you must answer the questions correctly and the team that has the most correct answers and is the furthest along in the case will win the challenge. The challenge is hard and it is expected that each team will end up working closely together to solve it.)

Exercise Preparation

1. **Reboot your SIFT workstation (I have found that working from a clean system works best).**
2. **Start your SIFT VMware Workstation in VMware Workstation.**
3. **Login the VMware machine.**
   - *LOGIN* = `sansforensics`
   - *PASSWORD* = `forensics`
4. **Re-Open First Exercise and follow the steps to copy the system (OS-USER-IP) that you have been assigned to analyze into your cases directory. You should copy three directories (c-drive, memory, and incident-response).**

Final Challenge Preparation

Mount your images for analysis in your system

- *Start a terminal, elevate your privs to root, and change into the /cases/OS-USER-c-drive directory.*

```
$ sudo su -

# cd /cases/OS-USER-c-drive/
```

- *Mount your evidence files so you can see the xp-tdungan-c-drive raw image and files/folders from the system.*

```
# ewfmount OS-USER-c-drive.E01 /mnt/ewf_mount/

# mount -o ro,loop,show_sys_files,streams_interface=windows
/mnt/ewf_mount/ewf1 /mnt/windows_mount
```

Your raw image is here:	`/mnt/ewf_mount/ewf1`
Your E01 image is here:	`/cases/OS-USER-c-drive/OS-USER-c-drive.E01`
Your mounted files/folders are here:	`/mnt/windows_mount`
You can view your files/folders here:	`\\siftworkstation\mnt\windows_mount`

Prepare Memory Analysis

- Each system lead should turn back to Redline Preprocess Exercise and perform the Redline Pre-Process against the memory raw memory image of their newly assigned system.
- Convert any Hibernation files (hiberfil.sys)
- Any options are allowed for prepping your memory images. It is your call on what you think is best for analyzing all of the memory from your machine.
- It is ok to ask your instructor about other options here for preparation that would be allowed

- Ask your instructor "if it is allowed" if you are in doubt. Usually something that will take over 10 minutes to process will generally be allowed.
- Do not begin to analyze. Prep only.

Prepare Timeline Analysis

- See addendums for each system
- Following this exercise, there are step-by-step guides for creating the overall timeline image for each additional system.

What else?

- We have just covered the core of your basic analysis preparation.
- If you or your team determine any additional steps you would like to accomplish, please consider them. In fact it is greatly encouraged. Remember if there is anything you would like to do that might take some time using any toolset out there that could help you, consider it.
- Please try and clear the additional processing with your instructor prior to beginning it.

This page intentionally left blank.

Win7 NFury10.3.58.6 PREP

Mount NFURY Windows 7 Image and Mount Multiple Volume Shadows

(Note: Some of this might already be accomplished via earlier exercises, but this is the state that we hope your system is in prior to the start of this exercise. Just in case your system rebooted, we are including a guide to help you get back to the proper analysis starting point prior to the beginning of this exercise.)

1. *REBOOT your SIFT VMware Workstation in VMware Workstation.*

2. *Login the VMware machine.*
   - *LOGIN =* **`sansforensics`**
   - *PASSWORD =* **`forensics`**
3. *Start a terminal, elevate your privs to root, and change into the /cases/win7-64-nfury-c-drive directory.*

```
$ sudo su -

# cd /cases/win7-64-nfury-c-drive
```

4. *Mount your evidence files so you can see the win7-64-nfury-c-drive raw image and files/folders from the system.*

```
# ewfmount win7-64-nfury-c-drive.E01 /mnt/ewf_mount/

# mount -o ro,loop,show_sys_files,streams_interface=windows
/mnt/ewf_mount/ewf1 /mnt/windows_mount
```

5. *Mount raw image VSS using vshadowmount and mount all logical filesystems found in snapshot*

```
# vshadowmount /mnt/ewf_mount/ewf1 /mnt/vss

# cd /mnt/vss

# for i in vss*; do mount -o
ro,loop,show_sys_files,streams_interface=windows $i
/mnt/shadow_mount/$i; done
```

Create SuperTimeline across the core image and the VSS image

Create a VSS SuperTimeline – Step-by-Step Guide

1. Step 1 – Run Log2timeline.py

```
# cd /cases/win7-64-nfury-c-drive

# log2timeline.py --parsers
"winevtx,filestat,winreg,webhist,lnk,prefetch" win7-64-nfury-
plaso.dump win7-64-nfury-c-drive.E01

Only include VSS snapshots 6,7
```

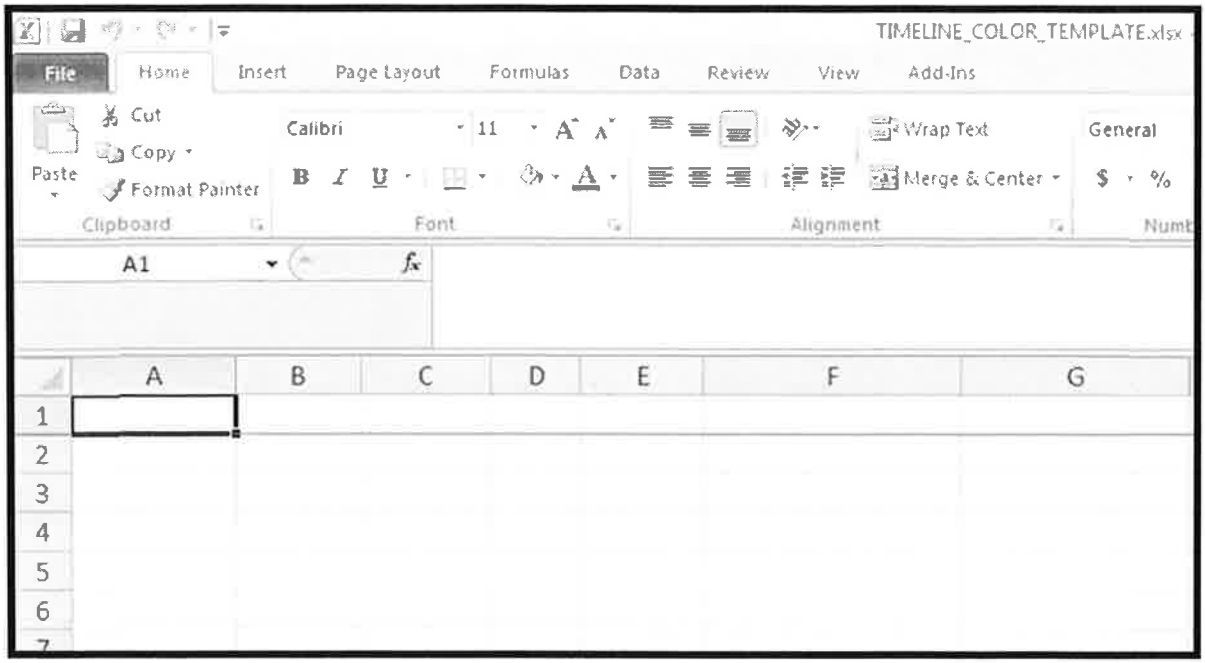
2. Filter using `psort.py`

```
# psort.py -z "EST5EDT" -o L2tcsv win7-64-nfury-plaso.dump "date >
'2012-04-02 20:00:00' AND date < '2012-04-07 00:00:00'" >
nfury.timeline.csv
```

3. Switch over to your Windows Workstation (Host System Preferred) and open the \\SIFTWORKSTATION from an open folder on your Desktop.

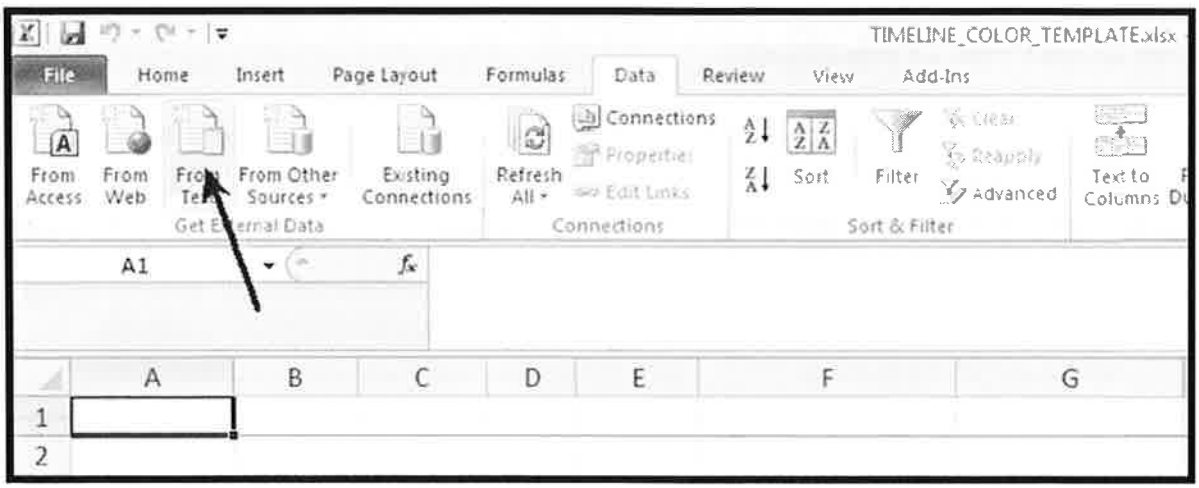
4. Take the filtered timeline and open it up in Excel using the color-timeline template from your windows system.
   a. Open Timeline Color Template
      i. **`TIMELINE_COLOR_TEMPLATE.xlsx`**

   b. Switch to Color Timeline worksheet/tab.
   c. Click on Cell A-1.

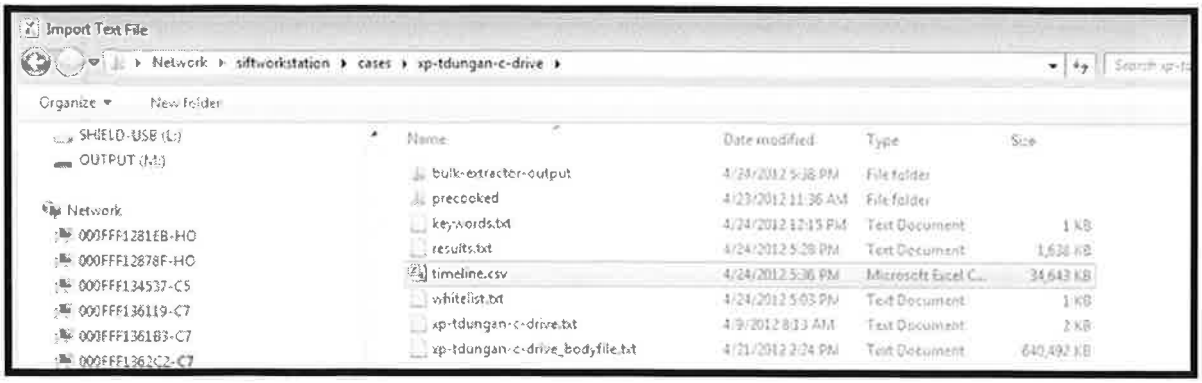


d. Select 'DATA' Ribbon.

e. Import Data "FROM TEXT".

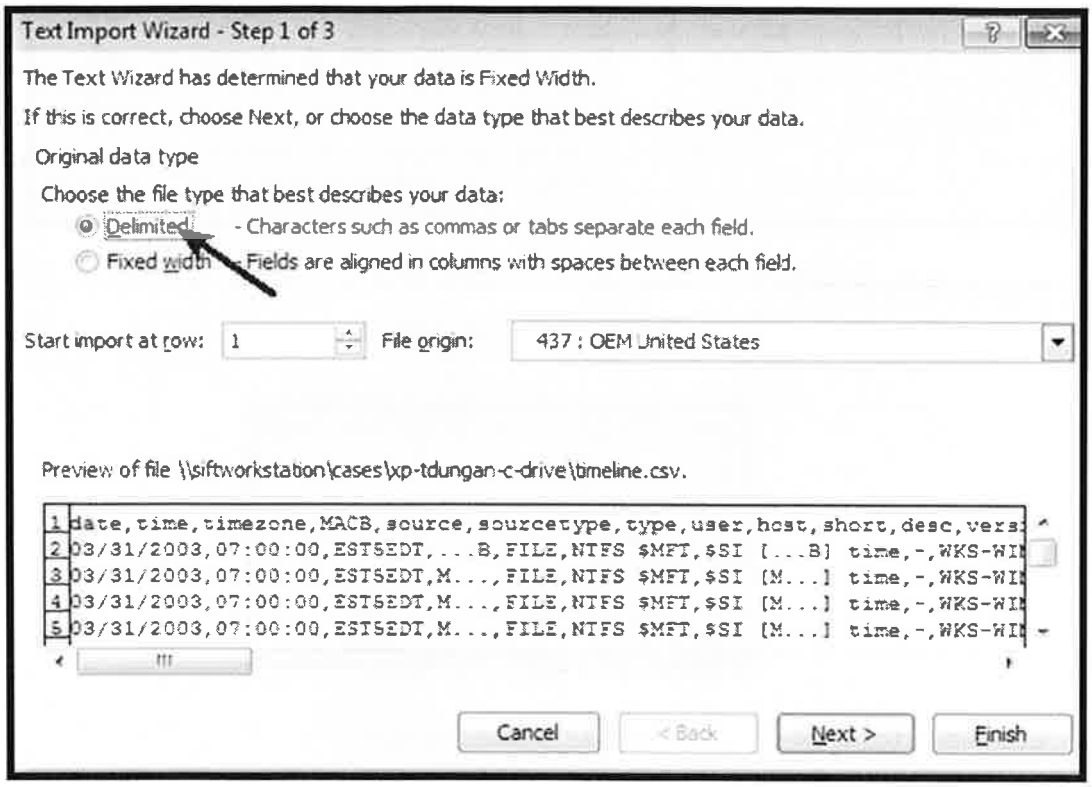


f. Select **`nfury.timeline.csv`** file ->**`\\siftworkstation\cases\win7-64-nfury-c-drive`**

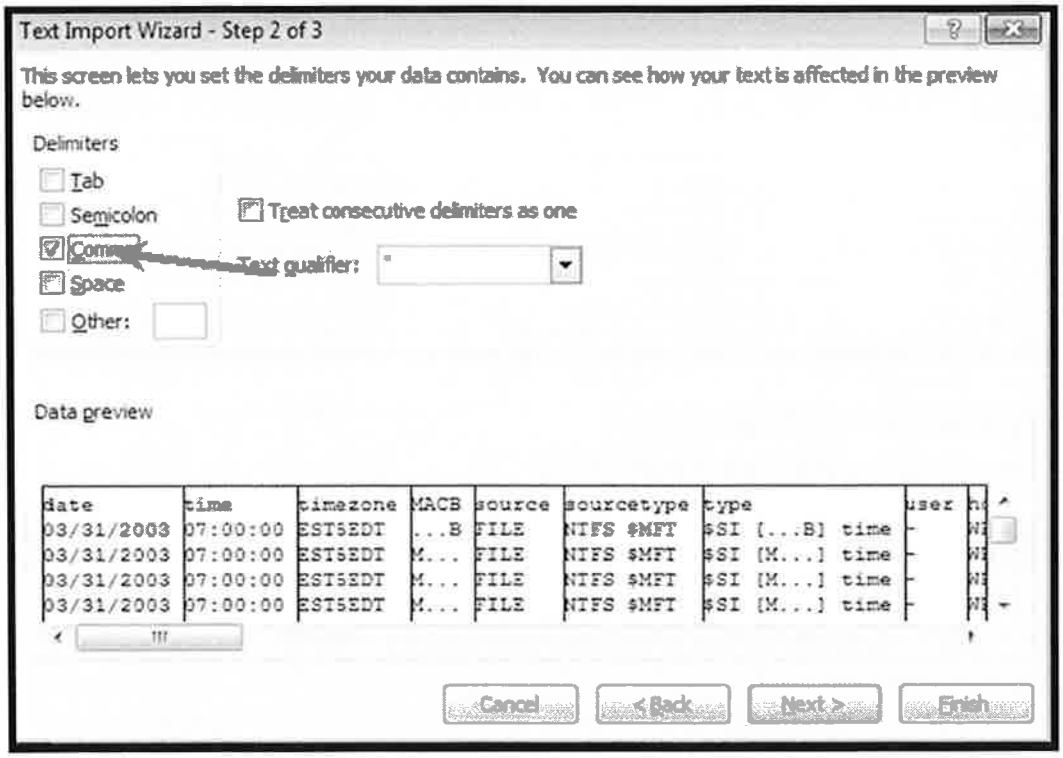


g. TEXT IMPORT WIZARD Will Start.

h. Step 1 -> Select Delimited -> Select NEXT.

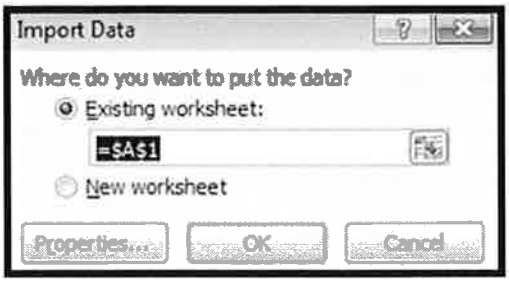


i. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT >

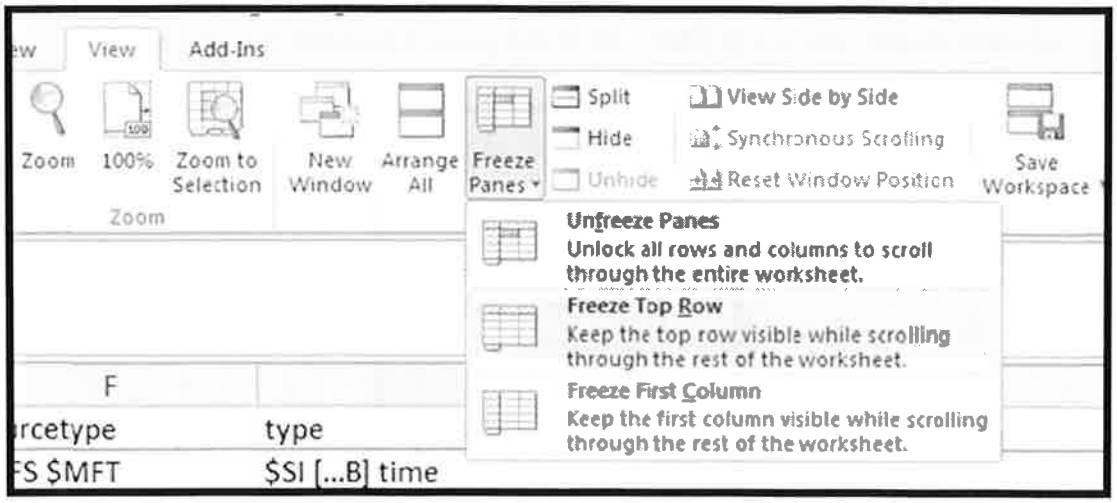


j. Step 3 ->Select Finish.

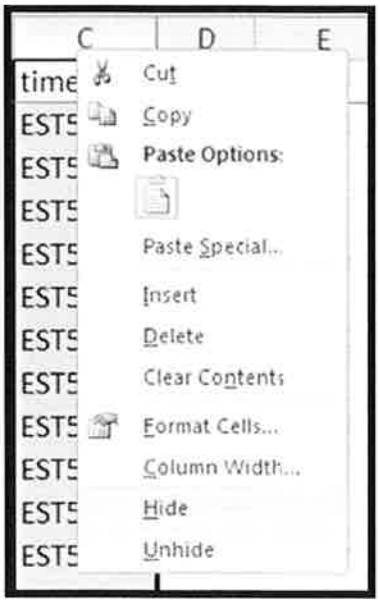
k. Where do you want to put the data? Simply Select OK.



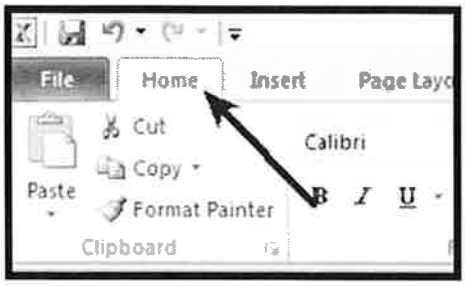
l. Once imported View -> Freeze Panes -> Freeze Top Row.



m. Optional Hide Columns Time Zone, Host, Version.



n. Select HOME Ribbon.



o. Select all Cells "CTRL-A".

p. In Home Ribbon -> Sort and Filter – Filter and you will be ready to begin analysis.

q. Before you analyze too much – please save your new Color Timeline as an XLSX file: **`/cases/win7-64-nfury-c-drive/vss-supertimeline.xlsx`**

This page intentionally left blank.

Win7 NRomanoff 10.3.58.5 PREP

Mount NROMANOFF Windows 7 Image and Mount Multiple Volume Shadows

(Note: Some of this might already be accomplished via earlier exercises, but this is the state that we hope your system is in prior to the start of this exercise. Just in case your system rebooted, we are including a guide to help you get back to the proper analysis starting point prior to the beginning of this exercise.)

1. *REBOOT your SIFT VMware Workstation in VMware Workstation.*

2. *Login the VMware machine.*
   - *LOGIN =* **`sansforensics`**
   - *PASSWORD =* **`forensics`**
3. *Start a terminal, elevate your privs to root, and change into the /cases/win7-c-drive directory.*

```
$ sudo su -

# cd /cases/win7-32-nromanoff-c-drive
```

4. *Mount your evidence files so you can see the win7-32-nromanoff-c-drive raw image and files/folders from the system.*

```
# ewfmount win7-32-nromanoff-c-drive.E01 /mnt/ewf_mount/

# mount -o ro,loop,show_sys_files,streams_interface=windows
/mnt/ewf_mount/ewf1 /mnt/windows_mount
```

5. *Mount raw image VSS using vshadowmount and mount all logical filesystems found in snapshot*

```
# vshadowmount /mnt/ewf_mount/ewf1 /mnt/vss

# cd /mnt/vss

# for i in vss[234]; do mount -o
ro,loop,show_sys_files,streams_interface=windows $i
/mnt/shadow_mount/$i; done
```

Create a VSS SuperTimeline – Step-by-Step Guide

1. Step 1 – Run Log2timeline

```
# cd /cases/win7-32-nromanoff-c-drive

# log2timeline.py --parsers
"winevtx,filestat,winreg,webhist,lnk,prefetch" win7-32-nromanoff-
plaso.dump win7-32-nromanoff-c-drive.E01

Only include VSS snapshots 3,4
```

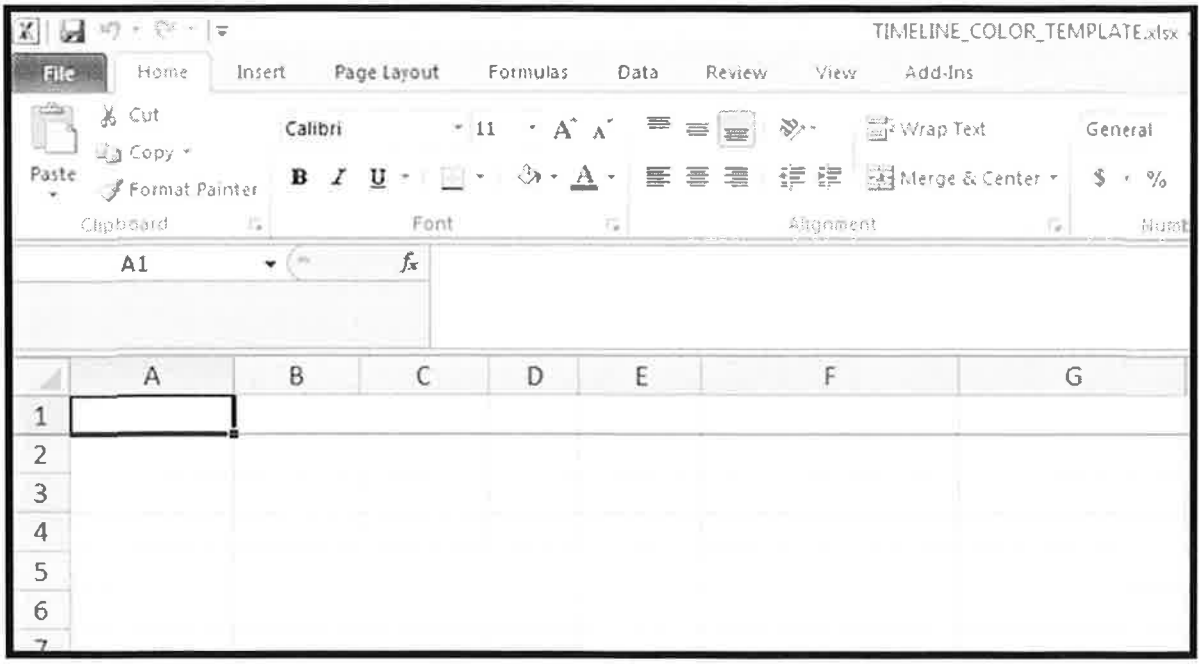
2. Filter using `psort.py`

```
# psort.py -z "EST5EDT" -o L2tcsv win7-32-nromanoff-plaso.dump
"date > '2012-04-02 20:00:00' AND date < '2012-04-07 00:00:00'" >
nromanoff.timeline.csv
```

3. Switch over to your Windows Workstation (Host System Preferred) and open the \\SIFTWORKSTATION from an open folder on your Desktop.

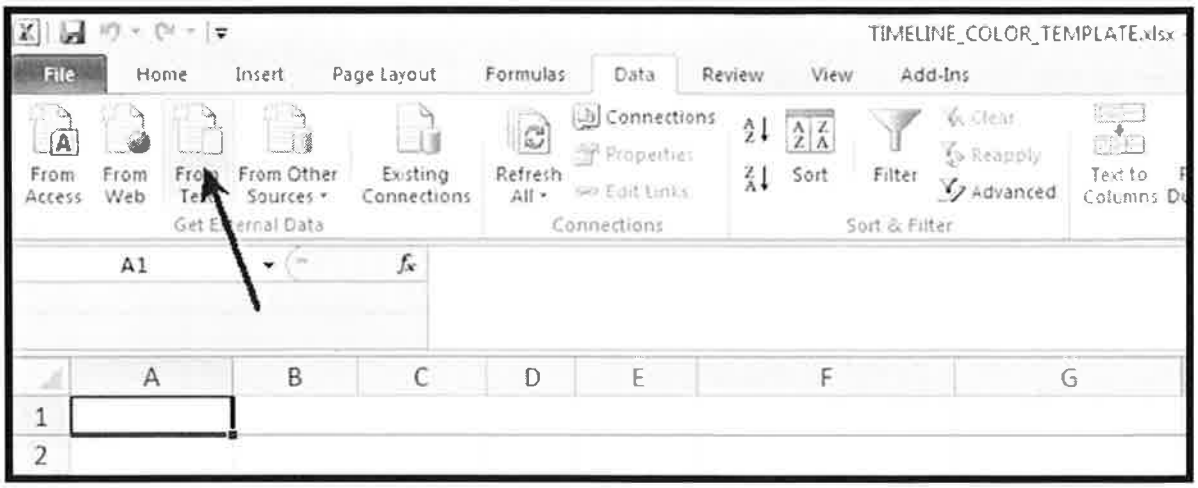
4. Take the filtered timeline and open it up in Excel using the color-timeline template from your windows system.
   a. Open Timeline Color Template
      i. **TIMELINE_COLOR_TEMPLATE.xlsx**

   b. Switch to Color Timeline worksheet/tab.
   c. Click on Cell A-1.

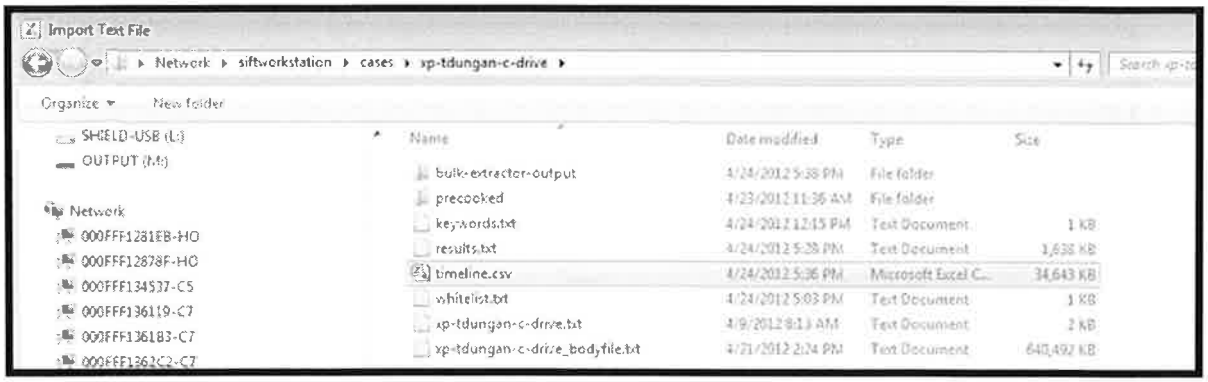


d. Select 'DATA' Ribbon.

e. Import Data "FROM TEXT".

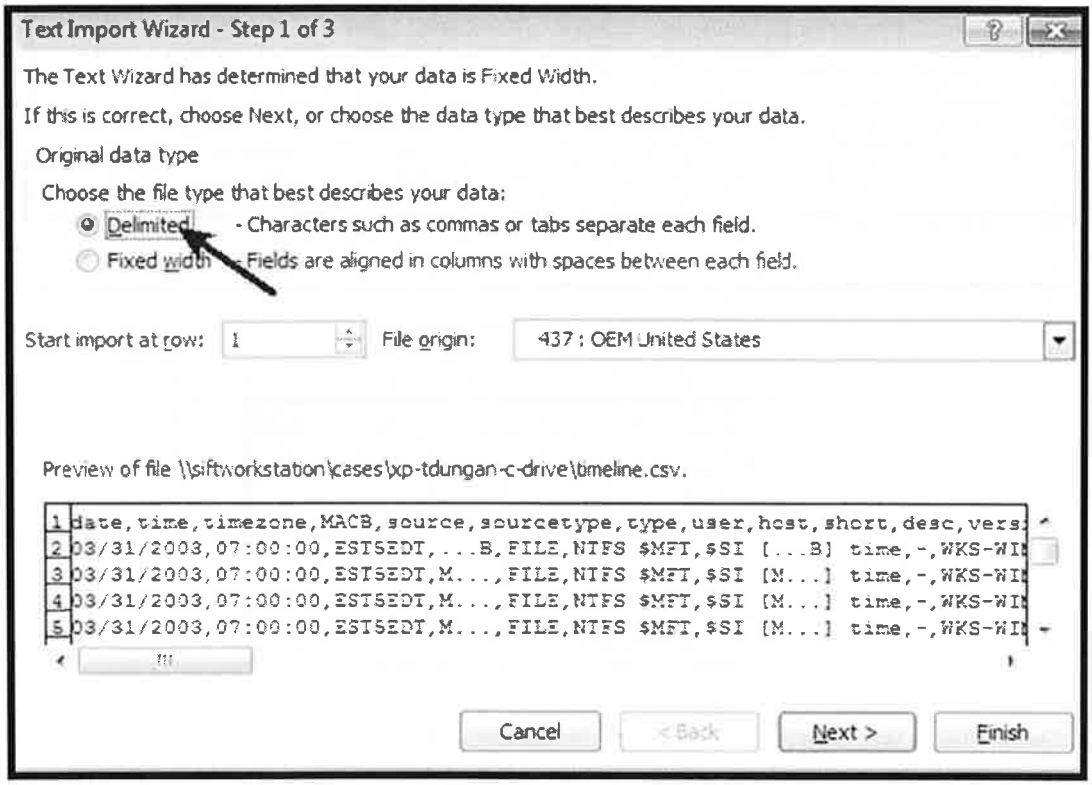


f. Select **`nromaoff.timeline.csv`** file ->**`\\siftworkstation\cases\win7-32-nromanoff-c-drive`**

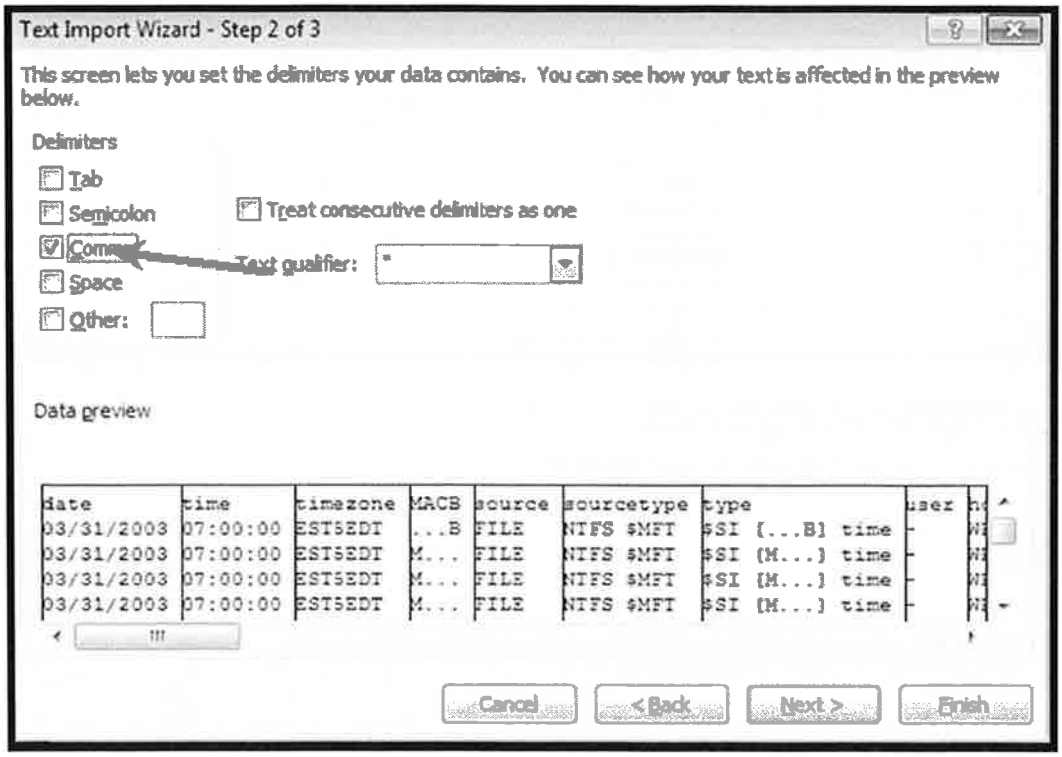


g. TEXT IMPORT WIZARD Will Start.

h. Step 1 -> Select Delimited -> Select NEXT.



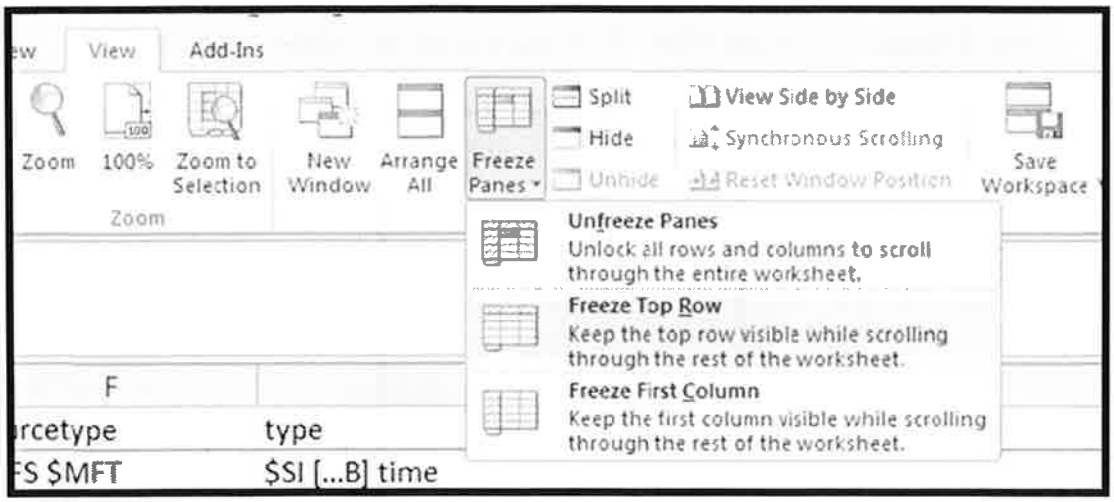
i. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT >



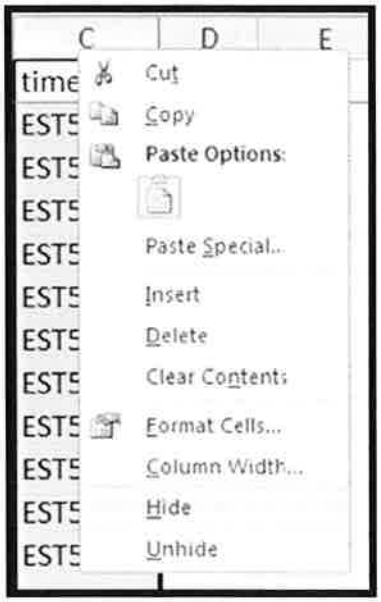
j. Step 3 ->Select Finish.

k. Where do you want to put the data? Simply Select OK.

l. Once imported View -> Freeze Panes -> Freeze Top Row.



m. Optional Hide Columns Time Zone, Host, Version.



n. Select HOME Ribbon.

o. Select all Cells "CTRL-A".

p. In Home Ribbon -> Sort and Filter – Filter and you will be ready to begin analysis.

q. Before you analyze too much – please save your new Color Timeline as an XLSX file: **`/cases/win7-32-nromanoff-c-drive/vss-supertimeline.xlsx`**

This page intentionally left blank.

win2008R2-controller 10.3.58.4 PREP

Mount win2008R2-Controller Image

(Note: Some of this might already be accomplished via earlier exercises, but this is the state that we hope your system is in prior to the start of this exercise. Just in case your system rebooted, we are including a guide to help you get back to the proper analysis starting point prior to the beginning of this exercise.)

1. ***REBOOT your SIFT VMware Workstation in VMware Workstation.***
2. *Login the VMware machine.*
   - *LOGIN =* **`sansforensics`**
   - *PASSWORD =* **`forensics`**
3. *Start a terminal, elevate your privs to root, and change into the /cases/win2008R2-controller-c-drivedirectory.*

```
$ sudo su -

# cd /cases/win2008R2-controller-c-drive
```

4. *Mount your evidence files so you can see the win2008R2-controller-c-driveraw image and files/folders from the system.*

```
# ewfmount win2008R2-controller-c-drive.E01 /mnt/ewf/

# mount -o ro,loop,show_sys_files,streams_interface=windows
/mnt/ewf/ewf1 /mnt/windows_mount
```

Create SuperTimeline of MFT hive data across the core image and the VSS image

Create a SuperTimeline – Step-by-Step Guide

1. Step 1 – Run Log2timeline.py

```
# cd /cases/ win2008R2-controller-c-drive

# log2timeline.py --parsers
"winevtx,filestat,winreg,webhist,lnk,prefetch" win2008-dc-plaso.dump
win2008R2-controller-c-drive.E01

Do Not Include Any Snapshots
```

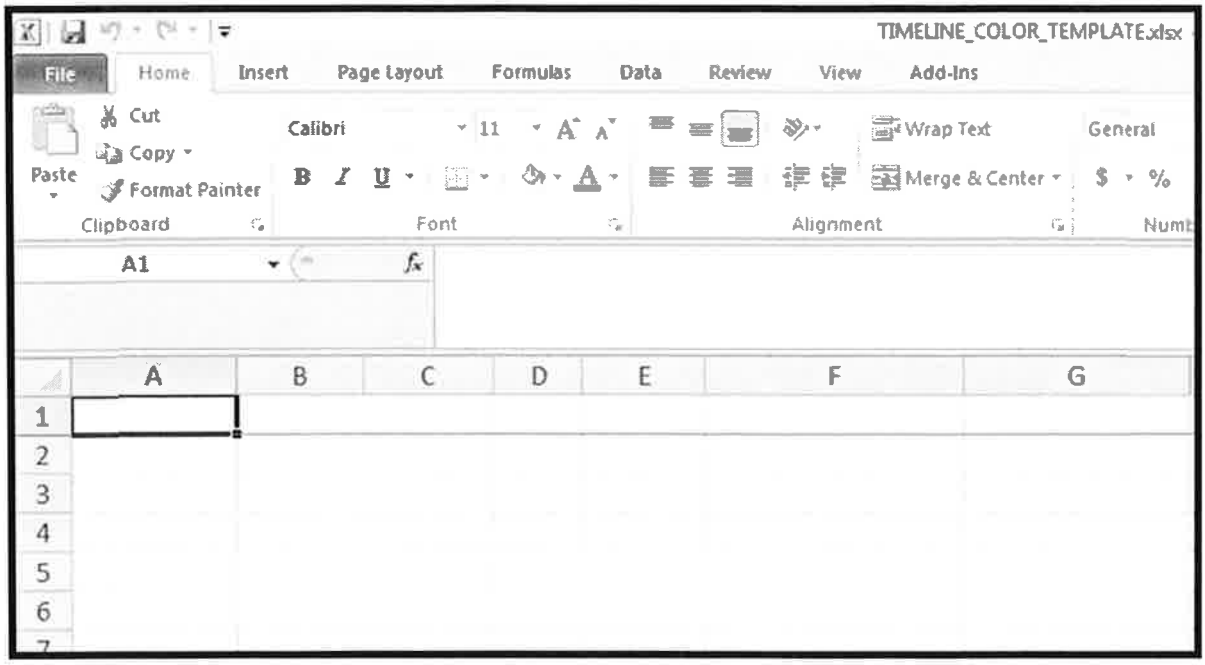
2. Filter using `psort.py`

```
# psort.py -z "EST5EDT" -o L2tcsv win2008-dc-plaso.dump
"date > '2012-04-02 20:00:00' AND date < '2012-04-07 00:00:00'" >
2008dc.timeline.csv
```

3. Switch over to your Windows Workstation (Host System Preferred) and open the \\SIFTWORKSTATION from an open folder on your Desktop.

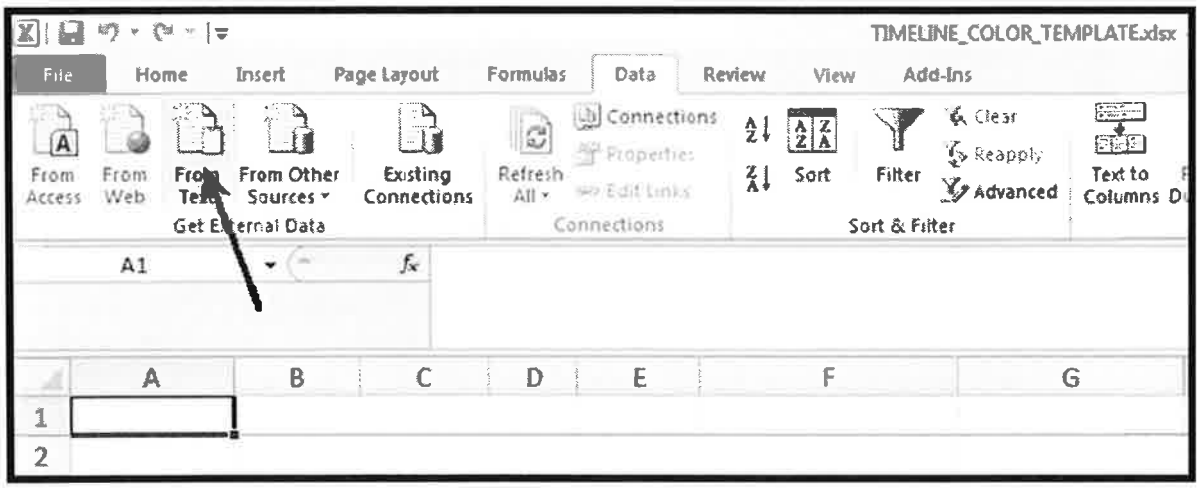
4. Take the filtered timeline and open it up in Excel using the color-timeline template from your windows system.
   a. Open Timeline Color Template
      i. **TIMELINE_COLOR_TEMPLATE.xlsx**

   b. Switch to Color Timeline worksheet/tab.
   c. Click on Cell A-1.



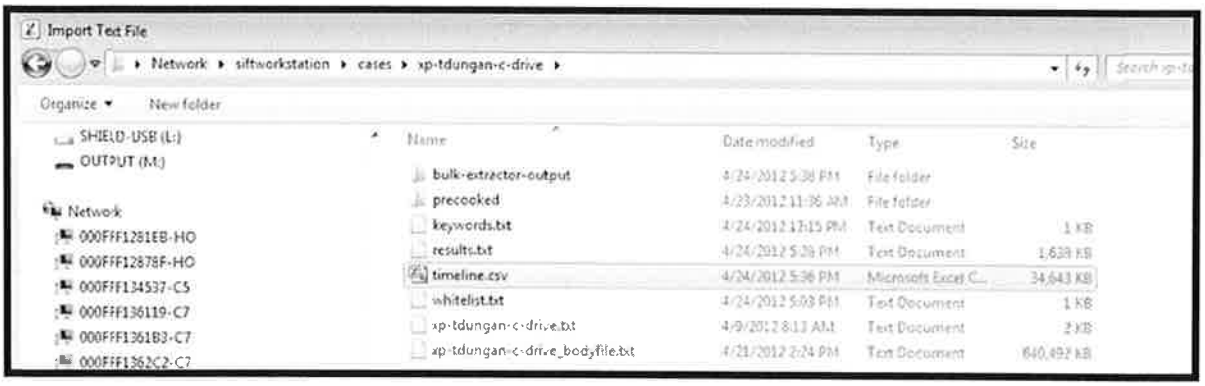
d. Select 'DATA' Ribbon.

e. Import Data "FROM TEXT".



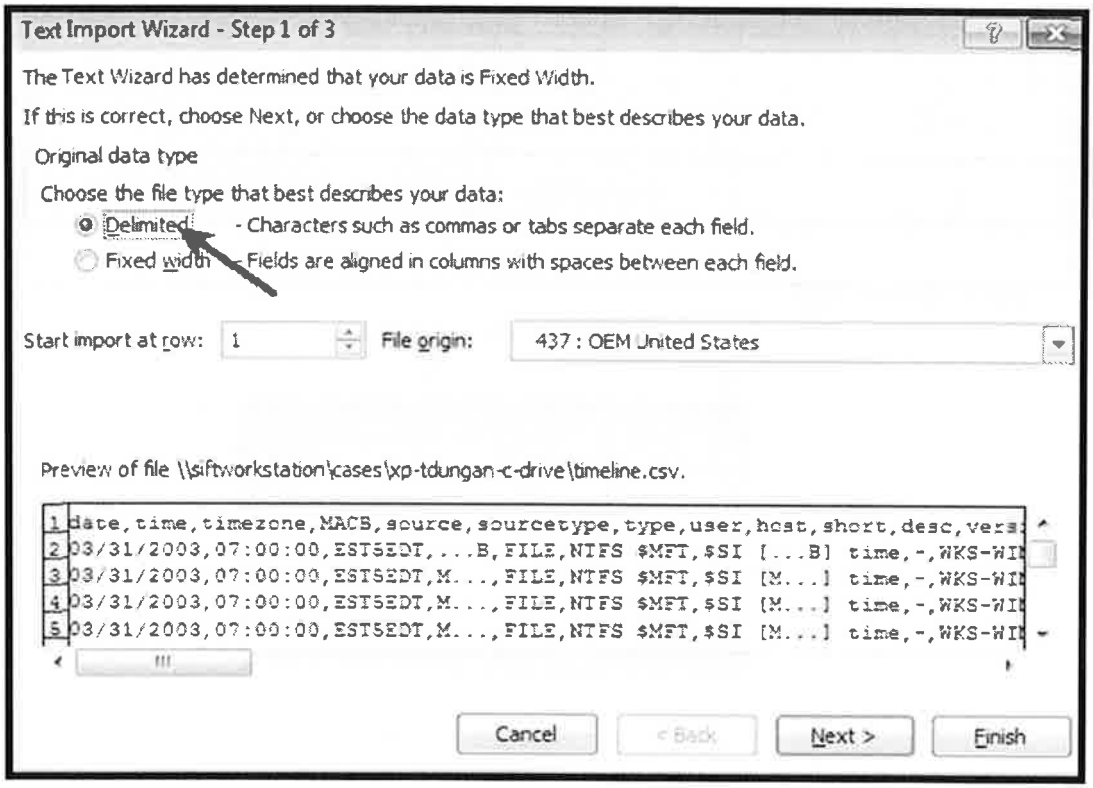
f. Select **`2008dc.timeline.csv`** file -> **`\\siftworkstation\cases\win2008R2-controller-c-drive`**

g.

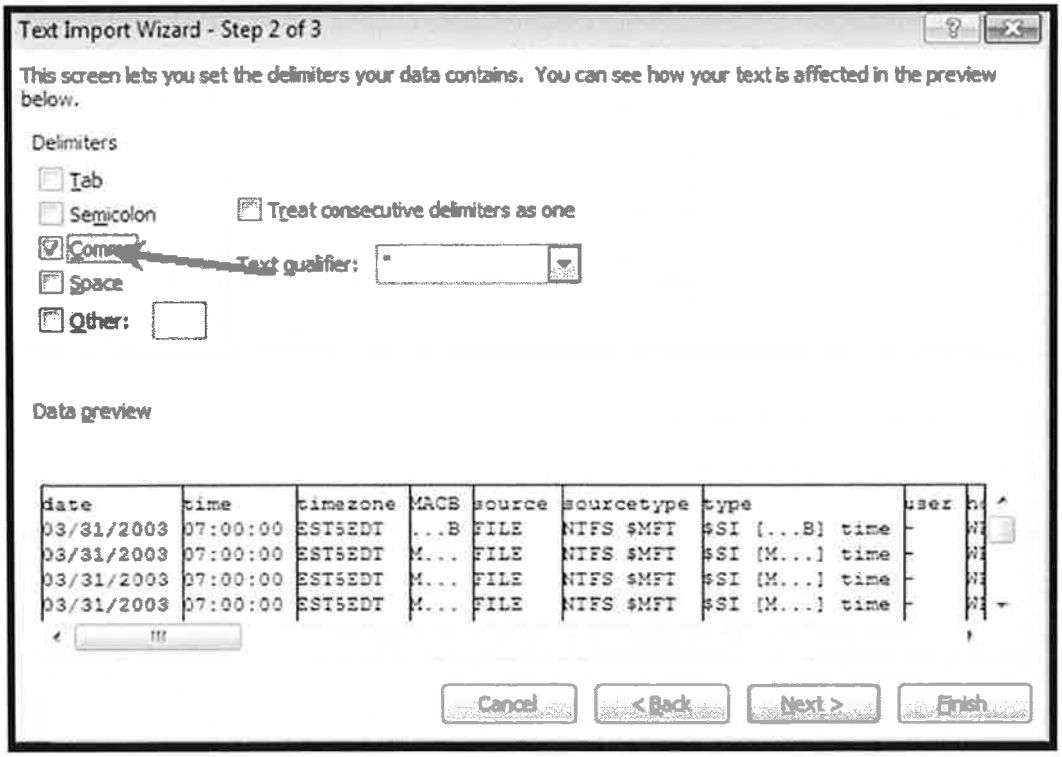


h. TEXT IMPORT WIZARD Will Start.

i. Step 1 -> Select Delimited -> Select NEXT.



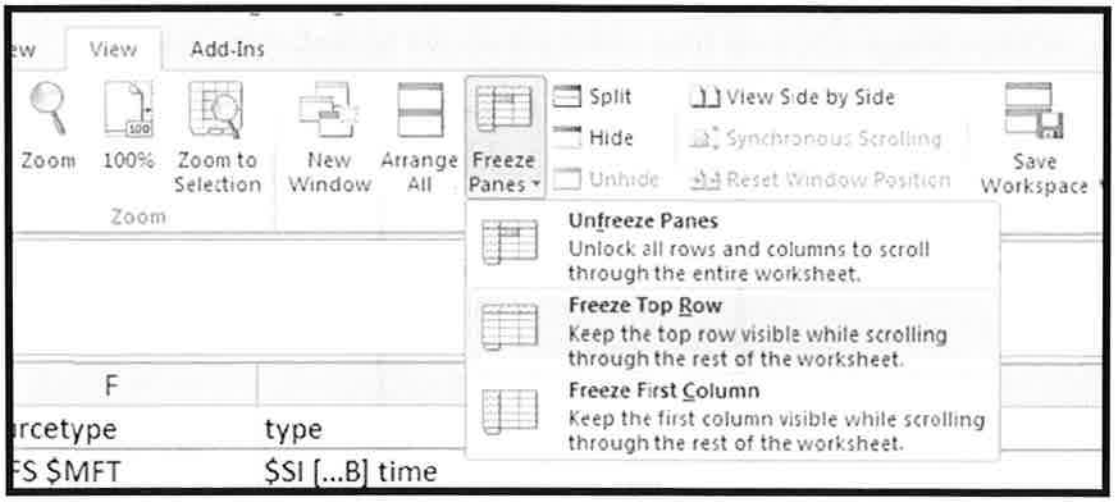
j. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT >



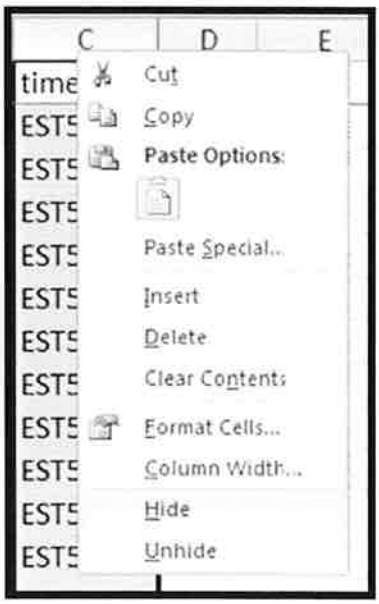
k. Step 3 ->Select Finish.

l. Where do you want to put the data? Simply Select OK.

m. Once imported View -> Freeze Panes -> Freeze Top Row.



n. Optional Hide Columns Time Zone, Host, Version.



o. Select HOME Ribbon.

p. Select all Cells "CTRL-A".

q. In Home Ribbon -> Sort and Filter – Filter and you will be ready to begin analysis.

r. Before you analyze too much – please save your new Color Timeline as an XLSX file: **`/cases/win2008R2-controller-c-drive/supertimeline.xlsx`**

ABOUT SANS

SANS is the most trusted and by far the largest source for information security training and certification in the world. It also develops, maintains, and makes available at no cost the largest collection of research documents about various aspects of information security, and it operates the Internet's early warning system – the Internet Storm Center.

The SANS (SysAdmin, Audit, Network, Security) Institute was established in 1989 as a cooperative research and education organization. Its programs now reach more than 165,000 security professionals around the world. A range of individuals from auditors and network administrators to chief information security officers are sharing the lessons they learn and are jointly finding solutions to the challenges they face. At the heart of SANS are the many security practitioners in varied global organizations from corporations to universities working together to help the entire information security community.

SANS provides intensive, immersion training designed to help you and your staff master the practical steps necessary for defending systems and networks against the most dangerous threats – the ones being actively exploited. This training is full of important and immediately useful techniques that you can put to work as soon as you return to your office. Courses were developed through a consensus process involving hundreds of administrators, security managers, and information security professionals, and they address both security fundamentals and awareness and the in-depth technical aspects of the most crucial areas of IT security. www.sans.org

IN-DEPTH EDUCATION AND CERTIFICATION

During the past year, more than 12,000 security, networking, and system administration professionals attended multi-day, in-depth training by the world's top security practitioners and teachers. Next year, SANS programs will educate thousands more security professionals in the US and internationally.

Global Information Assurance Certification

The Global Information Assurance Certification (GIAC) was founded in 1999 to validate the real-world skills of IT security professionals. GIAC's purpose is to provide assurance that a certified individual has practical awareness, knowledge, and skills in key areas of computer, network, and software security. GIAC currently offers certifications for over 20 job-specific responsibilities that reflect the current practice of information security. GIAC is unique in measuring specific knowledge areas instead of general purpose information security knowledge. 22,365 students have obtained GIAC certifications with hundreds more in the process of doing so. www.giac.org

SANS BREAKS THE NEWS

SANS NewsBites is a semi-weekly, high-level executive summary of the most important news articles that have been published on computer security during the last week. Each news item is very briefly summarized and includes a reference on the Web for detailed information, if possible. www.sans.org/newsletters/newsbites

@RISK: The Consensus Security Alert is a weekly report summarizing the vulnerabilities that matter most and steps for protection. www.sans.org/newsletters/risk

Ouch! is the first consensus monthly security awareness report for end users. It shows what to look for and how to avoid phishing and other scams plus viruses and other malware using the latest attacks as examples. www.sans.org/newsletters/ouch

The Internet Storm Center (ISC) was created in 2001 following the successful detection, analysis, and widespread warning of the Li0n worm. Today, the ISC provides a free analysis and warning service to thousands of Internet users and organizations and is actively working with Internet Service Providers to fight back against the most malicious attackers. http://isc.sans.org

TRAINING WITHOUT TRAVEL ALTERNATIVES

Nothing beats the experience of attending a live SANS training event with incomparable instructors and guest speakers, vendor solutions expos, and myriad networking opportunities. Sometimes, though, travel costs and a week away from the office are just not feasible. When limited time and/or budget keeps you or your co-workers grounded, you can still get great SANS training close to home.

SANS OnSite *Your Location – Your Schedule*

With SANS OnSite program you can bring a unique combination of high-quality and world-recognized instructors to train your professionals at your location and realize significant savings. For organizations that need to train a large number of people, the SANS OnSite program is simply hard to beat!

Six reasons to consider SANS OnSite:

1. Enjoy the same great certified SANS instructors and unparalleled courseware
2. Flexible scheduling – conduct the training when it is convenient for you
3. Focus on internal security issues during class and find solutions
4. Keep staff close to home
5. Realize significant savings on travel expenses
6. Enable dispersed workforce to interact with one another in one place

DoD or DoD Contractor working to meet the stringent requirements of DoD-Directive 8570? SANS OnSite is the best way to help you achieve your training and certification objectives. Contact us today for more information at onsite@sans.org or 678-714-5712.

SANS OnDemand *Online Security Training & Assessments*

When you want access to SANS' high-quality training 'anytime, anywhere', choose our advanced online delivery method! OnDemand is designed to provide a very convenient, comprehensive, and highly effective means for information security professionals to receive the same intensive, immersion training that SANS is famous for. Students will receive:

- Four months access to online training
- Integrated lectures by SANS top-rated instructors
- Assessments to reinforce your knowledge throughout the course
- Hard copy of course books
- Access to our SANS Virtual Mentor
- Labs & hands-on exercises
- Progress Reports

SANS @Home *Personal SANS Instruction at Home*

SANS @Home delivers live instruction via the Web using various Internet-based technologies. Streaming audio, instant messaging, online forums, and e-mail are all leveraged to make the student's online learning experience as fun and engaging as possible.

Visit our Web site for more ways to Train Without Travel
www.sans.org/training/without_travel

For group programs, please contact us at groupsales@sans.org