FOR508 | ADVANCED DIGITAL FORENSICS, INCIDENT RESPONSE, & THREAT HUNTING

SRL Intrusion-Exercise Workbook

SANS

THE MOST TRUSTED SOURCE FOR INFORMATION SECURITY TRAINING, CERTIFICATION, AND RESEARCH | sans.org

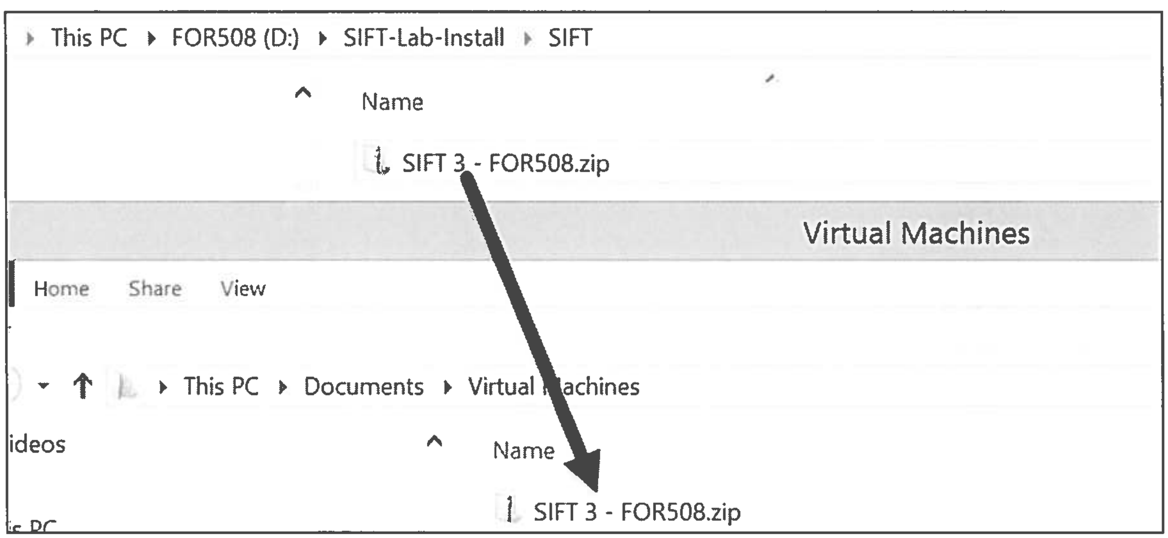
Exercise 0 – SIFT Lab Installation

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.

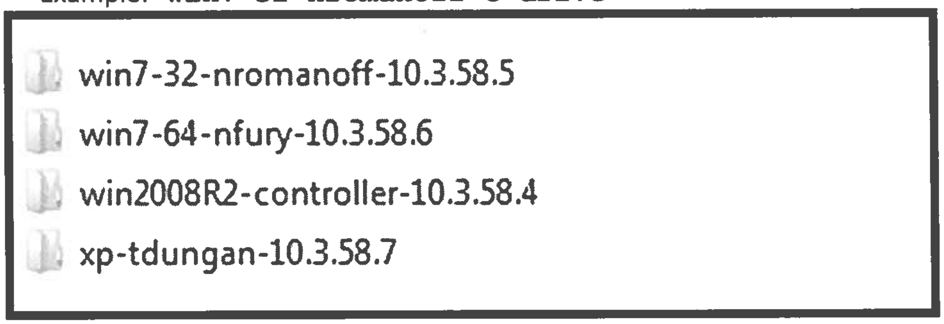
Exercise Preparation

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 install for the FULL version of .NET is found in `\SIFT-Lab-Install\redline.`
   - It is CRITICAL that you install and ensure 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 you are using a Windows host, install 7zip.*
   - Located on your USB under `\SIFT-Lab-Install\7zip.`
4. *Copy and install the SIFT workstation.*
   - Located on your USB under `\SIFT-Lab-Install\SIFT\SIFT 3 - FOR508.zip.`

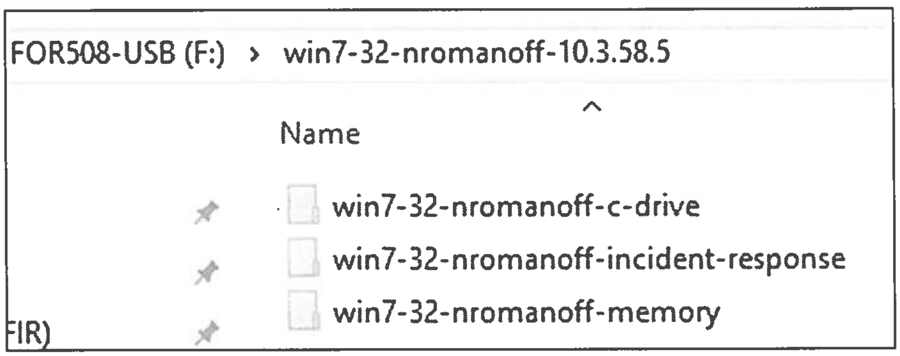


What Is on the USB?

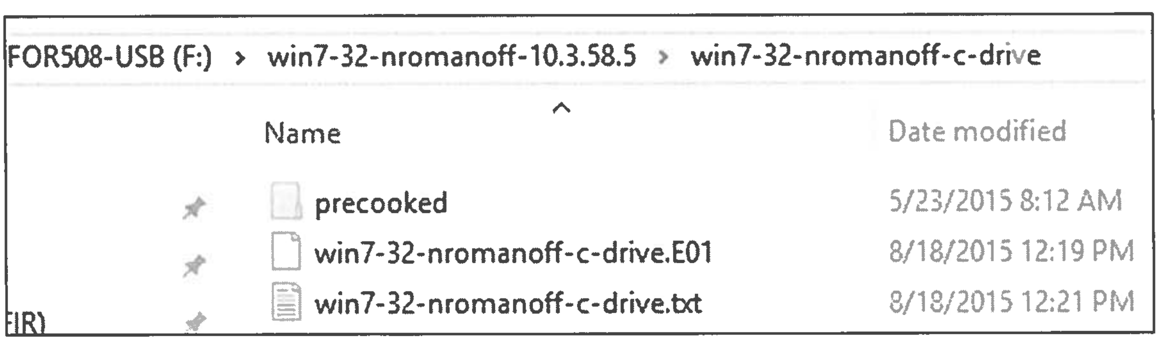
1. The USB contains the data from the four systems potentially involved in the intrusion breach.
2. The name of each of the directories is listed by:
    - Operating System
    - System owner
    - IP address
 - **OS-User-IP**
    - Example: **win7-32-nromanoff-c-drive**



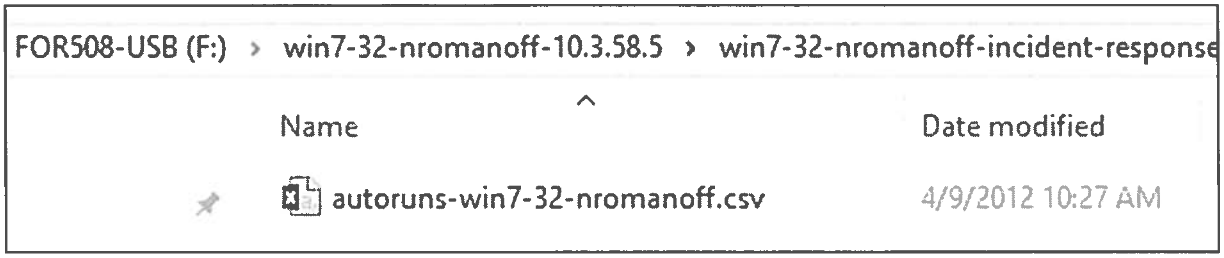
3. In each of the directories, you will find three additional directories.



   a. c-drive directory
    - This is the location of the collected hard drive images.
    - On the **win7-32-nromanoff-c-drive** directory, you will also see a **precooked** directory.
    - The precooked directory 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.

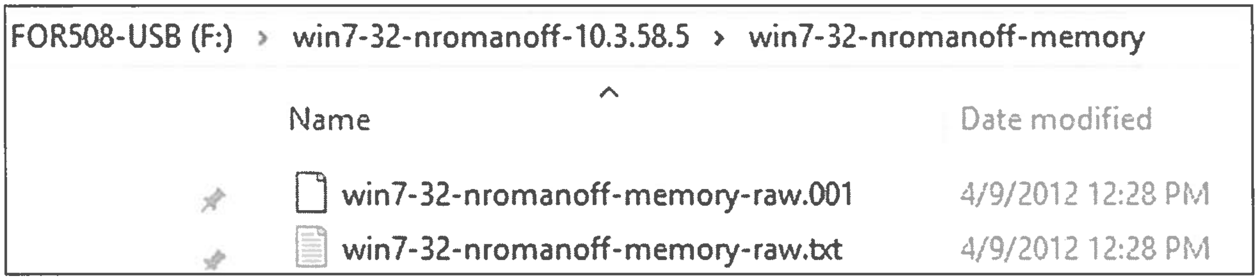


   b. Incident Response directory
    - This is the location of data collected during Incident Response on each system.



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\SIFT3 - FOR508\SIFT3 - FOR508.vmx`

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

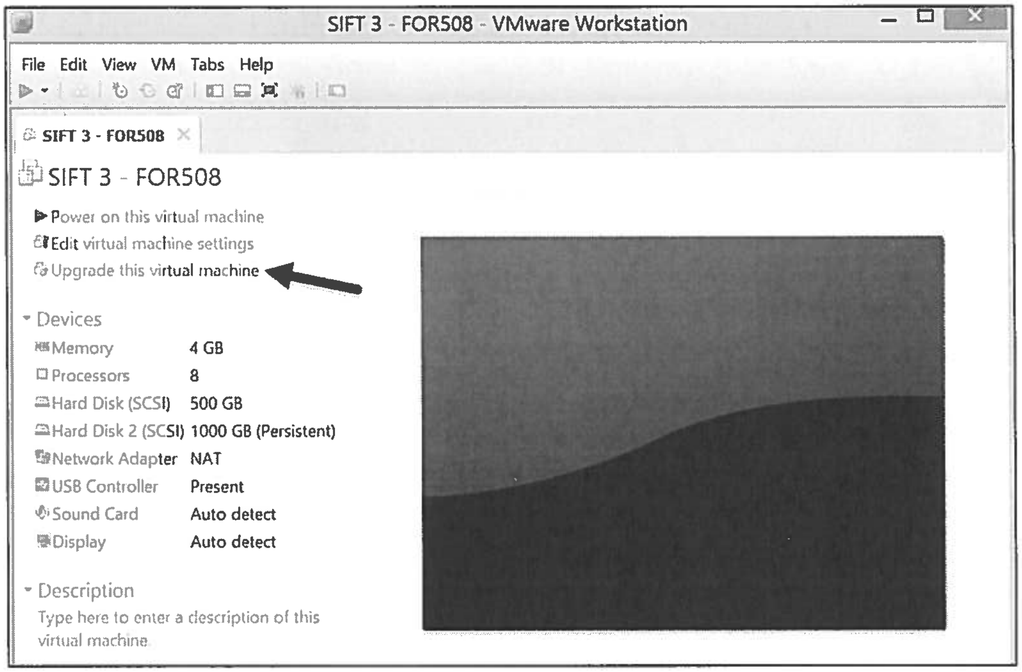


Figure 1 - Update Version to Latest

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

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

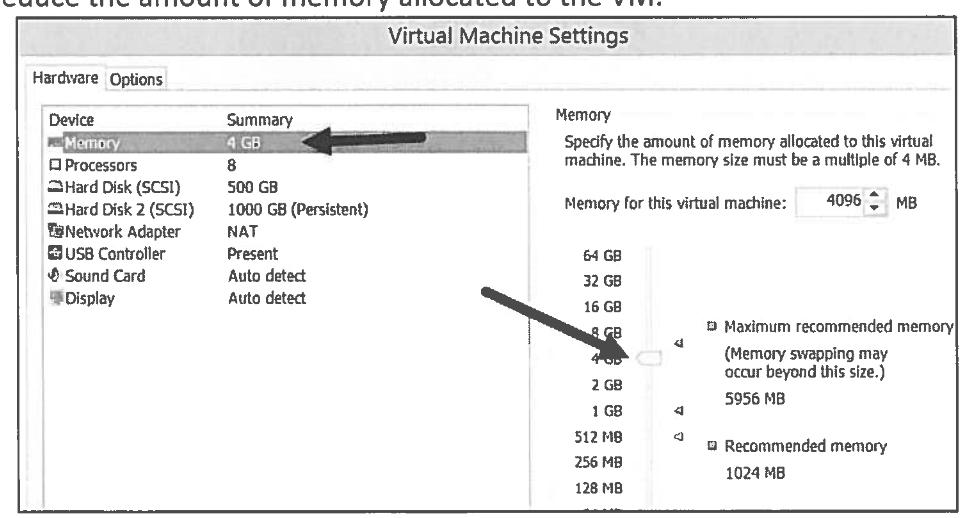
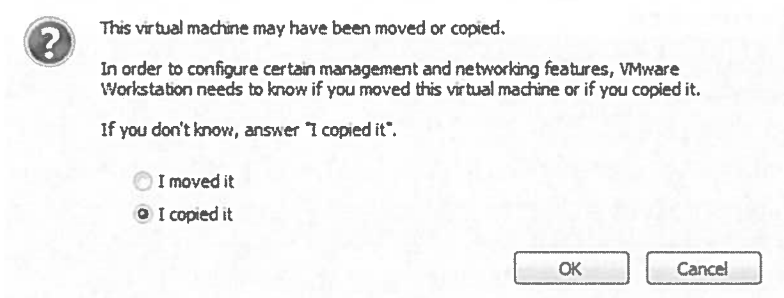
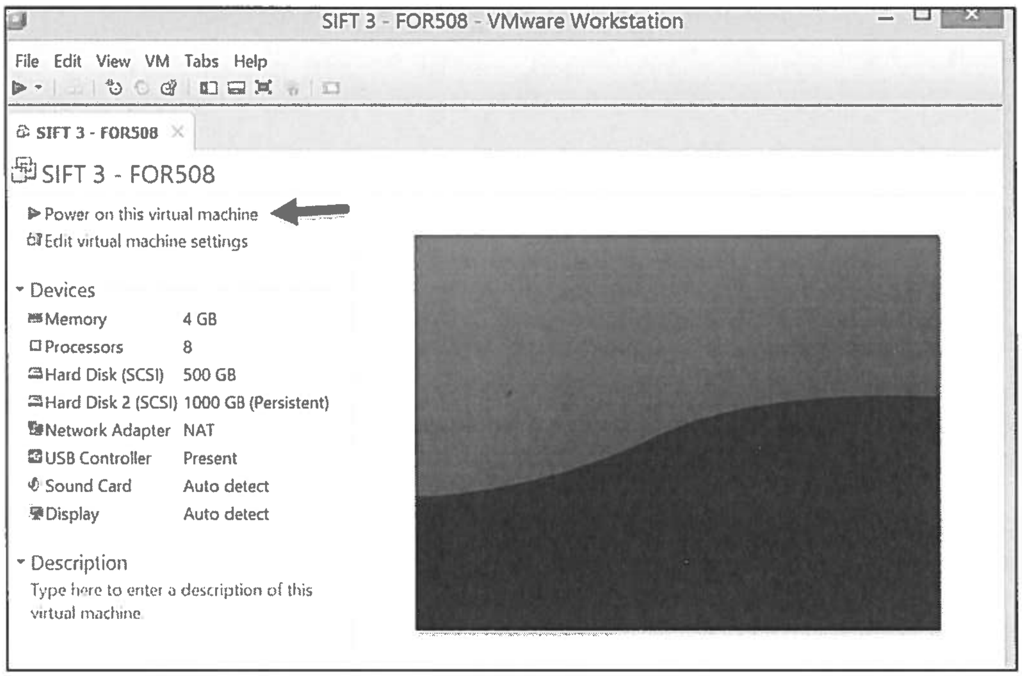


Figure 2 - Edit Memory Configuration

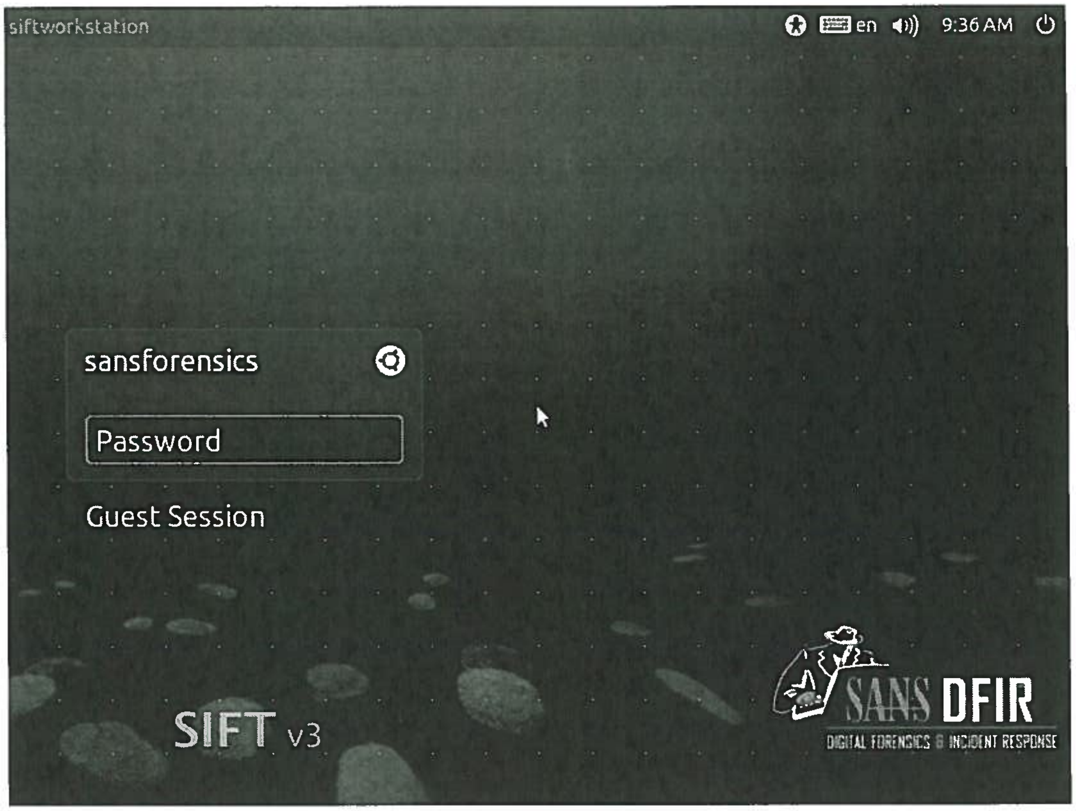
5. Power on your virtual machine.
    1. Press "Power on virtual machine" and select "I copied it."



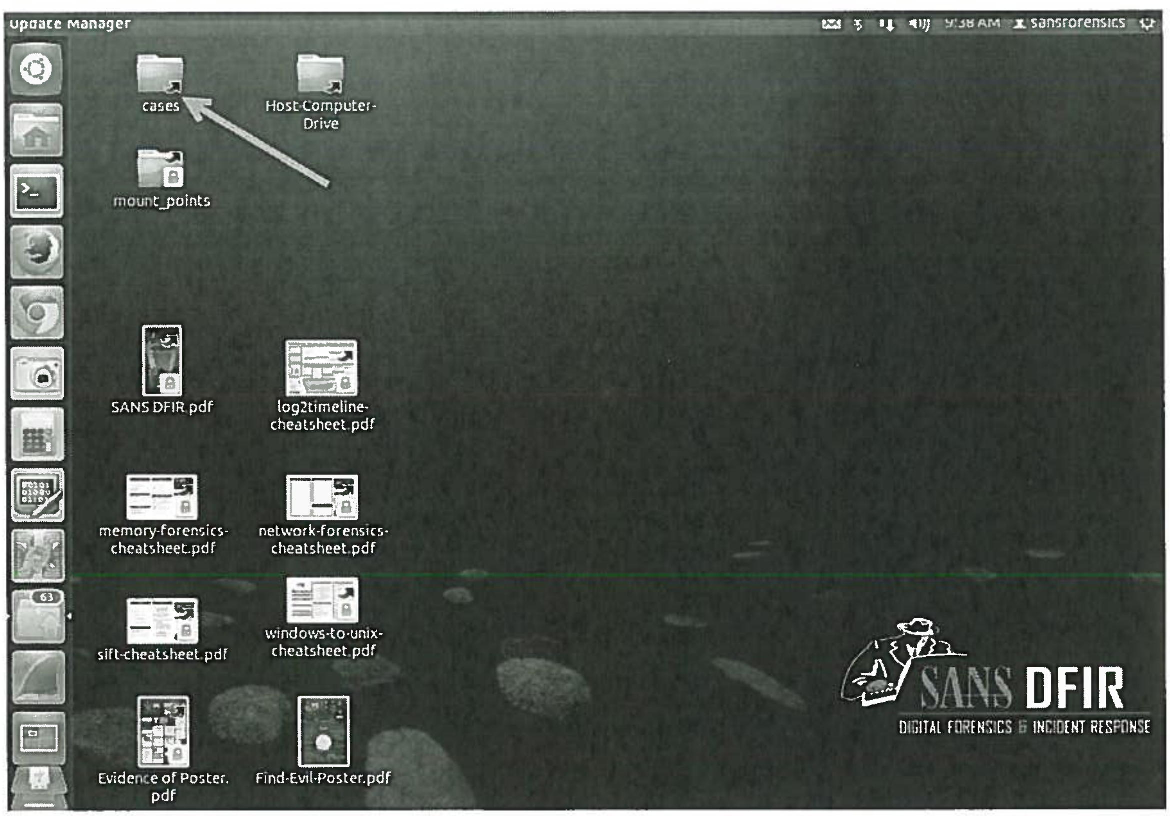
6. Start your SIFT VMware Workstation in VMware Workstation.



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



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



9. Check your initial Incident Response Data -- Change directories to the **/cases** directory in your terminal and run the file listing command "**ls.**" You should see all the directories here.

```
sansforensics@siftworkstation: /cases
sansforensics@siftworkstation:~$ cd /cases/
sansforensics@siftworkstation:/cases$ ls
example-memory-images        win7-32-nromanoff-incident-response
win7-32-nromanoff-c-drive    win7-32-nromanoff-memory
sansforensics@siftworkstation:/cases$
```

10. You have successfully prepped your SIFT Workstation for FOR508 – Advanced Incident Response for the first part of the week!

This page intentionally left blank.

STARK RESEARCH LABS

Exercise 1.1 - APT Intrusion Scenario

APT Scenario Written, Designed, and Created by Rob Lee

APT Team - John Strand and Timothy Tomes

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. Timothy Dungan has been working tirelessly on trying to find the missing formula for the past two years.

DIRECTOR OF LAB DESCRIBES AN APT THREAT – MARCH 27, 2012

M. Hill is the current SRL-LABS Director.

```
Date: 27 Mar 2012 07:08:48 -0700 (PDT)
From: Maria Hill <mhill.shield@yahoo.com>
Reply-To: Maria Hill <mhill.shield@yahoo.com>
Subject: APT Threat
To: Department Heads SRL

All Research Department Leads:

Our threat Intelligence Sharing and Analysis Center (ISAC) partnership has shared
that an APT group called HYDRA has taken specific interest in metallurgy research as
of late. They have compromised similar companies and research entities.

As far as we can tell, HYDRA APT group is sponsored by a nation state advanced
threat who targets specifc companies and organizations who specialize in metal and
bio-engineering research similar to ours. The exhibit similar quality in capability
compared to the "COMMENT CREW" or "APT1" This nation has started to infiltrate many
sectors in almost every industry stealing economic information, intellectual
property, and targeting key executives with spear phishing attacks.

The ISAC has released a specific IOC for our Hunting/Incident Response teams to use
to help identify potential presence in our organization. Also, their intent is
targeting breakthrough research, so we will likely begin taking a closer look at
systems and management involved in the vibranium research project.

Metals Research Subnet: 10.3.58.X

If you have any questions, please reply to this email.

-MH
```

Note: APT1 REPORT is found on your USB under **`\documents\Threat Intelligence Reports\MANDIANT_APT1_REPORT.pdf`**.

THREAT INTELLIGENCE REPORT - New Malware (HTTPPUMP) in use by APT Group HYDRA

Description: Members of this family are full-featured backdoors that communicate with a Web-based Command & Control (C2) server over HTTP port 80 using XMLRPC over HTTP for encoding. Features include interactive shell, gathering system information, 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 runkey launchers to establish a 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.

Basic Threat Indicator Provided for HTTPPUMP

- MD5 = **`c4b0458c04abdaa773348c2668212b45`**
- OR
    - Filename = a.exe or b.exe or 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`**

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

Timothy Dungan @Tim Reply Retweet Favorite · Open

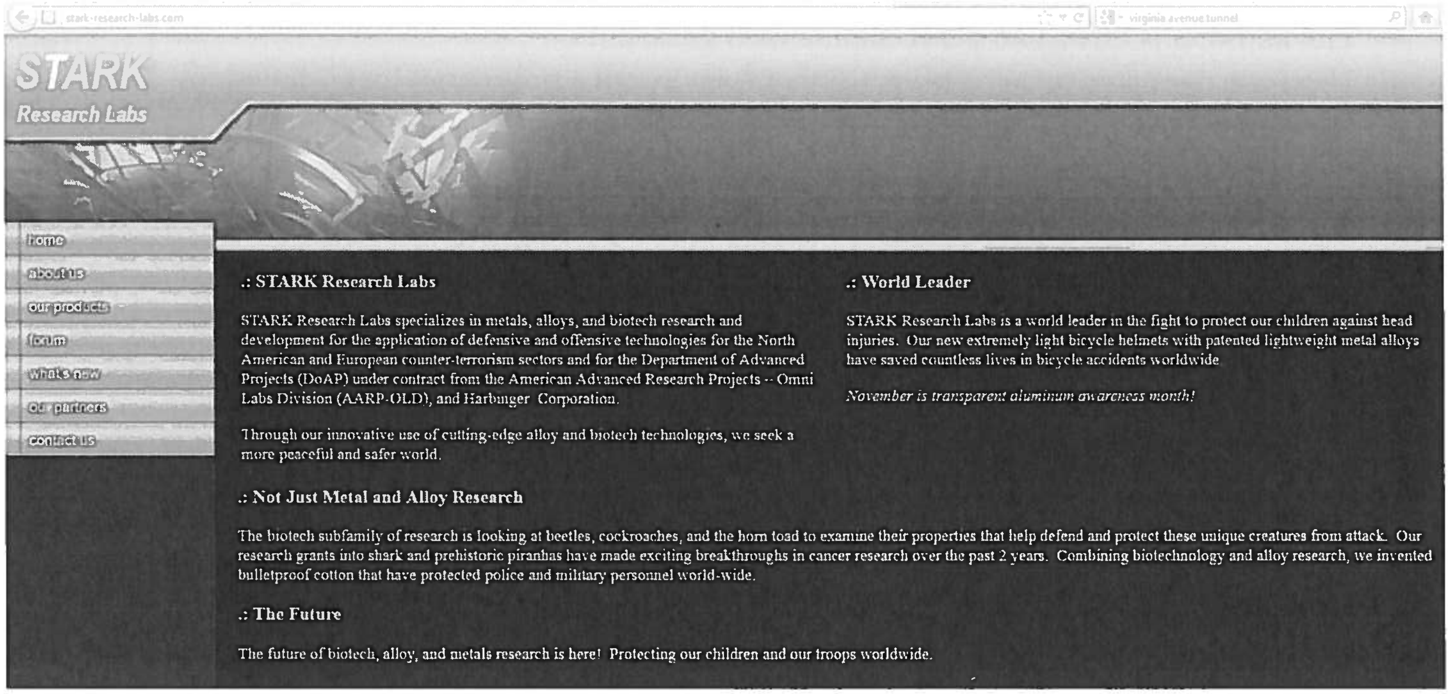
FINALLY We found it!!! Thanks to PETER!!!

FAST FORWARD – APRIL 6, 2012 @5PM

The SRL Threat Hunting Team begins hunting for the APT group hydra. Your mission?

- Answer the key question: Have we been breached?
- Begin/perform Incident Response if a breach is detected.

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



COMPANY NETWORK

Enterprise network with more than 1,000 users and systems.

The target of the analysis will begin looking at the subnet involved in the vibranium research, which also contains a domain controller for that segment of the network.

A few of the key systems on that subnet include:

- 10.3.58.4 Win2008R2 Domain controller
- 10.3.58.5 Win7-SP1-32bit N. Romanoff Workstation – Vibranium Program Manager
- 10.3.58.6 Win7-SP1-64bit N. Fury Workstation – SRL Management
- 10.3.58.7 WinXP-SP3-32bit Timothy Dungan Workstation – Vibranium Lead Researcher

Additional Key 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 is 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, and Chrome).
- Fully patched as of 6 April 2012 (today). Patches are automatically installed.
- Enterprise Incident Response agents (F-Response Enterprise).
- Enterprise A/V and -capability (McAfee Endpoint Protection—Antivirus, Antispyware, Safe surfing, Antispam, Device Control, Onsite Management, and Host Intrusion Prevention [HIPS]).
- Network Using HBSS (Host Based Security System) per the DOD recommendations.
- Firewall allowed inbound 25, 80, 443 and outbound 25, 80, 443 only.
- Users have been "using" this network for more than a year prior to the attack. That way it has the look and feel of something real. These users have set up social media (yes, they are on Twitter, and you might be friends with them), e-mail, Skype, and so on. Each character user has a backstory and a reason to be there working.
- Rsydow is the sole domain administrator.
- SRL-HELPDESK account is local admin account with a shared password.

ADDITIONAL INFORMATION REGARDING SRL DOMAIN

- Local Admin User (SRL-Helpdesk) was found on each system with the 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.

Exercise 1.2 – Mounting Evidence Using SIFT

Objectives

1. Mount an acquired disk image into the SIFT workstation so that you can see the files and folders and begin analysis of the system.
2. Learn the process for examining and manipulating compressed images, such as split E01s, using the SIFT workstation.
3. Learn alternative ways of mounting disk images using **`imageMounter.py`** and **`mountwin`**.

Exercise Preparation

1. *Start your SIFT VMware Workstation in VMware Workstation.*
2. *Log in to 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/win7-32-nromaoff-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/win7-32-nromaoff-c-drive/

# ls -lh
```

```
root@siftworkstation:/cases/win7-32-nromanoff-c-drive# ls -lh
total 9.1G
drwxrwxrwx 3 sansforensics sansforensics 4.0K May 23  2015 precooked
-rwxrw-rw- 1 sansforensics sansforensics 9.1G Aug 18  2015 win7-32-nromanoff-c-drive.E01
-rwxrw-rw- 1 sansforensics sansforensics 1.9K Aug 18  2015 win7-32-nromanoff-c-drive.txt
```

3. Mount the **`win7-32-nromaoff-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, and such if present) 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 win7-32-nromanoff-c-drive.E01 /mnt/ewf_mount/

# cd /mnt/ewf_mount

# ls -lh
```

```
root@siftworkstation:/# cd /cases/win7-32-nromanoff-c-drive/
root@siftworkstation:/cases/win7-32-nromanoff-c-drive# ewfmount win7-32-nromanoff-c-drive.E01 /mnt/ewf_mount
ewfmount 20140608

root@siftworkstation:/cases/win7-32-nromanoff-c-drive# cd /mnt/ewf_mount
root@siftworkstation:/mnt/ewf_mount# ls -lh
total 0
-r--r--r-- 1 root root 25G Jun  7 18:35 ewf1
```

4. Mount the win7-32-nromaoff-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 -- <u>no</u> 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
```

```
root@siftworkstation:/mnt/windows_mount# ls
$AttrDef      Boot          config.sys              $LogFile      PerfLogs       $Recycle.Bin               Users
autoexec.bat  $Boot         Documents and Settings  $MFTMirr      ProgramData    $Secure                    $Volume
$BadClus      bootmgr       $Extend                 MSOCache      Program Files  System Volume Information  Windows
$Bitmap       BOOTSECT.BAK  hiberfil.sys            pagefile.sys  Recovery       $UpCase
```

Alternative Exercise – Using Alias "`mountwin`" to Reduce Typing and to Make Mounting Easier

```
root@siftworkstation:/# alias
alias mountwin='mount -o ro,loop,show_sys_files,streams_interface=windows'
```

1. Unmount the previously mounted files and remount using the alias of the mount command shown above using "**`mountwin`**" instead.

```
# umount /mnt/windows_mount

# cd /mnt/ewf_mount

# mountwin ewf1 /mnt/windows_mount

# cd /mnt/windows_mount

# ls
```

Alternative Exercise – Using `imageMounter.py` to Make it Easier to Mount Physical and Logical Disks

Mounting disk images, although easy to do, is still a multistep process using the SIFT Workstation. To shorten this step, Kevin Breen updated his imageMounter.py script to mount logical and physical disks automatically to the location of your choice. The exciting thing about the code is that it handles both GPT- and MSDOS-based physical disk structures and portioning schemes. As a result, an individual no longer needs to spend precious time pulling out calculators performing simple but time-consuming operations to simply mount a partition inside the SIFT workstation.

`imageMounter.py` points at the physical/logical disk structure as the first argument, and the second is the location in the /mnt directory where the analyst would like to mount the disk image.

If the analyst/responder is mounting a logical disk structure, then he would add the (**`-s`**) option to the file. If the file is contained as an E01 file, it will detect the format and perform the necessary mounts needed to adjust.

```
Usage: imageMounter.py [options] image_name mnt_point

[USEFULL OPTIONS}
  -s                    Single partition in image
  -i                    Just display the information
  -e                    Use ewfmount to mount E0 Evidence Files
  -b                    Mount Bitlocker Drive with Recovery Key
  -k RECOVER_KEY        Bitlokcer Recovery Key
```

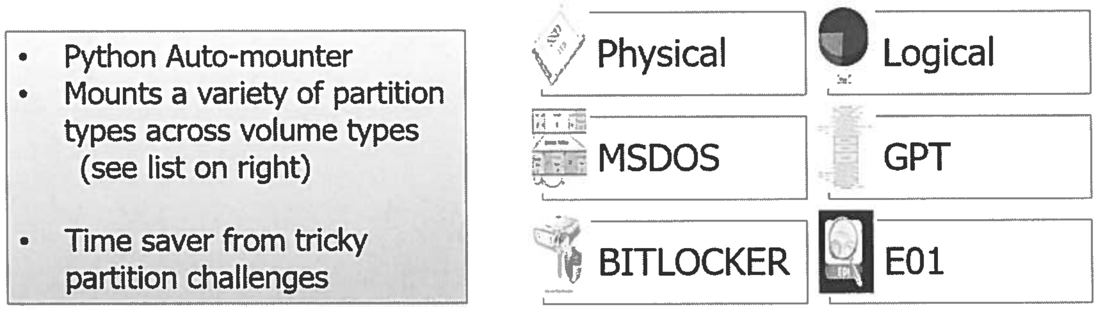


Figure 2 imageMounter.py Command Options and Explanations

In the end, I can see a tool such as imageMounter.py helping analysts avoid being annoyed slightly by having to do this manually. **`imageMounter.py`** will go a long way in speeding up the mounting process.

1. Unmount the previously mounted files and remount using "**`imageMounter.py.`**"

```
# umount /mnt/windows_mount

# umount /mnt/ewf_mount

# cd /cases/win7-32-nromaoff-c-drive/

# imageMounter.py -s win7-32-nromaoff-c-drive.E01 /mnt/windows_mount

# cd /mnt/windows_mount

# ls
```

Exercise 1.3 – CTI - Indicator Creation and Examination

Objectives

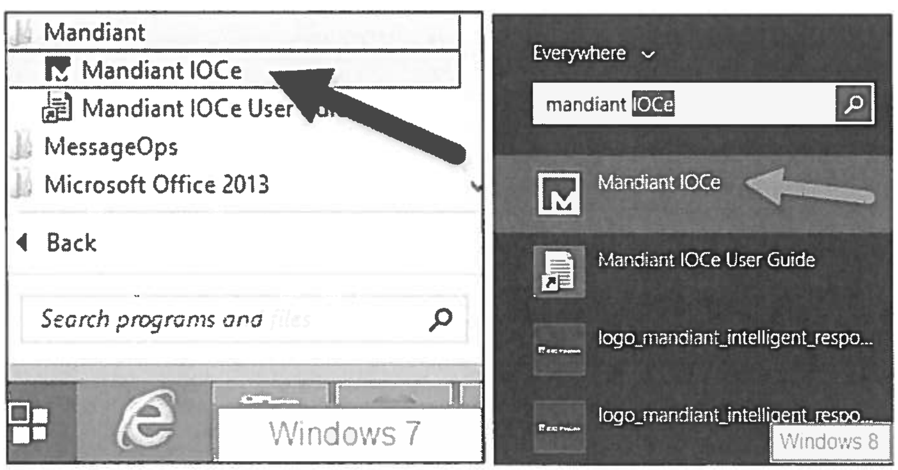
1. Install and launch IOC Editor on your Windows workstation.
2. Examine IOC Editor's interface and examine default APT1 "Comment Crew" indicators.
3. Build your own IOC based on the information provided by the government agency that contacted you.
4. Use Boolean logic and different operators to ensure your IOC is properly configured.

Exercise Preparation

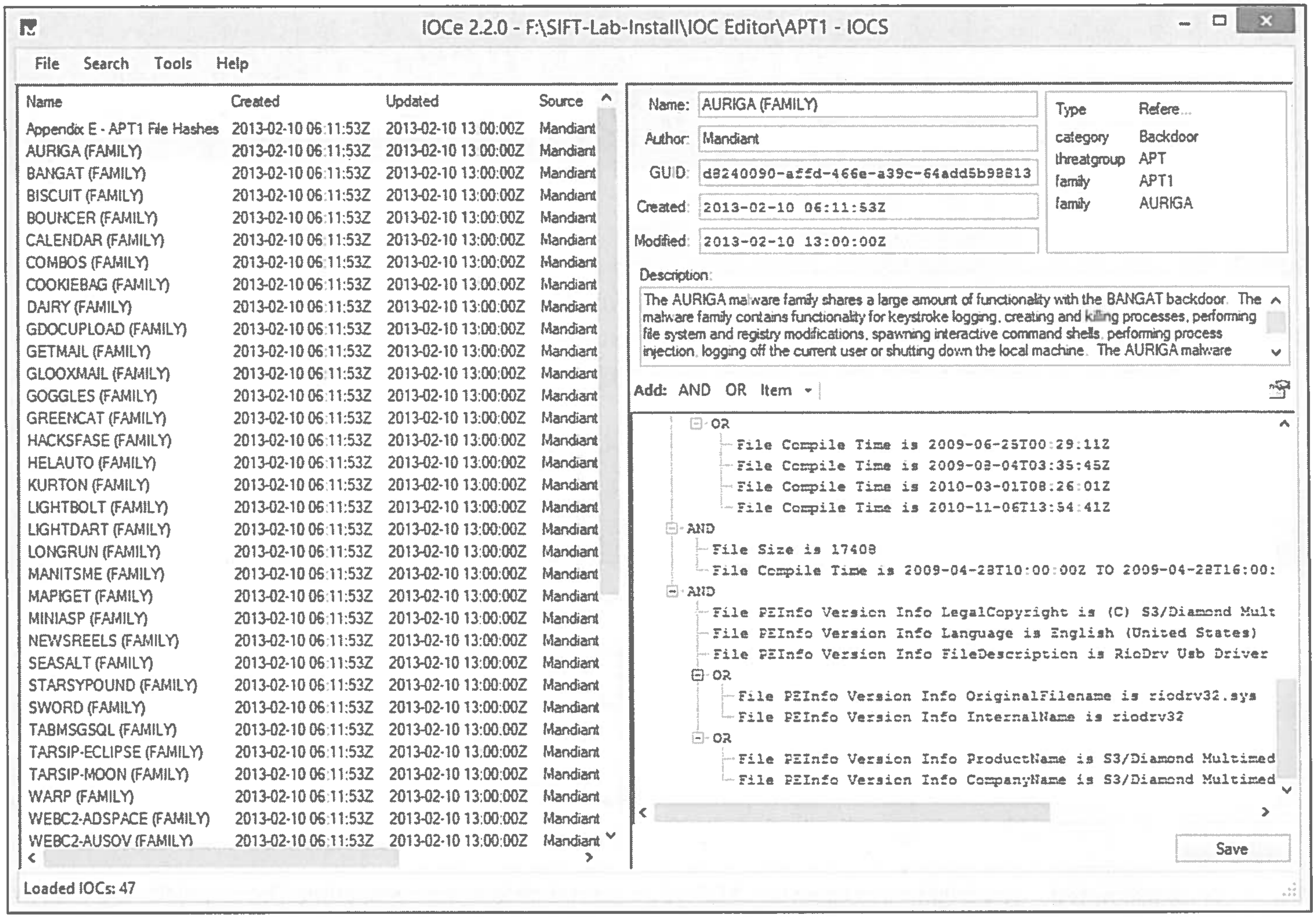
- *Install Mandiant IOC Editor from your USB on your Windows Host (or VM) in the* **`\SIFT-Lab-Install\IOC Editor`** *folder.*

- If prompted -- you might need to install .NET 3.5 or greater before the installation. The install for the FULL version of .NET is found in `\SIFT-Lab-Install\IOC Editor\DOTNET`
  a. NOTE: If you have Win10 and are encountering installation issues, there is a guide at end of exercise

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

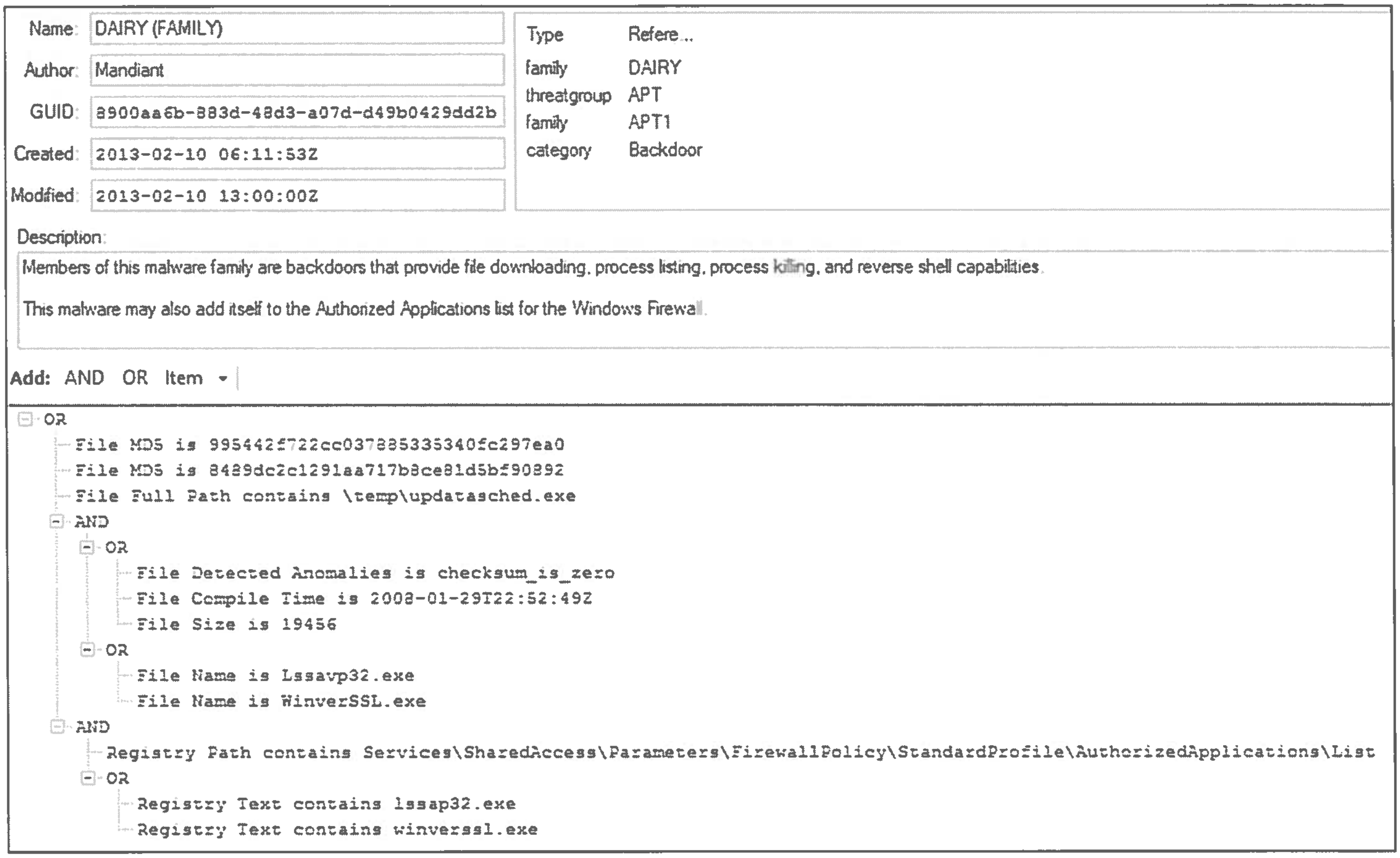


- *Open "IOC Directory"* **`\SIFT-Lab-Install\IOC Editor\APT1 - IOCS\`**.



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 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 a 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 has 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 text file (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 communicate 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.

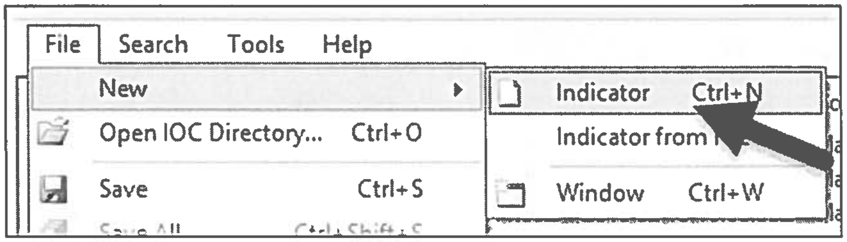
Some members of this family rely on launchers to establish persistence mechanism for them. Several variants use %USER%\Local Settings\Temp or %USER%\AppData\Local\Temp as working directories; additional malware artifacts may be found there.

IOC Provided

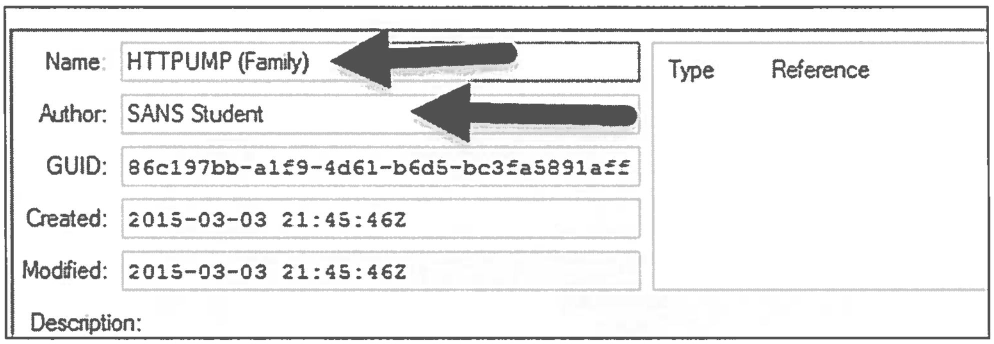
- (MD5 = **`c4b0458c04abdaa773348c2668212b45`** or Filename = **`a.exe`**, **`b.exe`**, **`c.exe`**)
    - AND
        - (Compile Time = **`2011-10-13 04:19:53`** or **`2011-10-19 02:39:12`**)
        - File Size = **`9216`** or **`9245`**
        - AND
            - Directory location = **`\Temp`**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

Exercise – Build the IOC Template Family

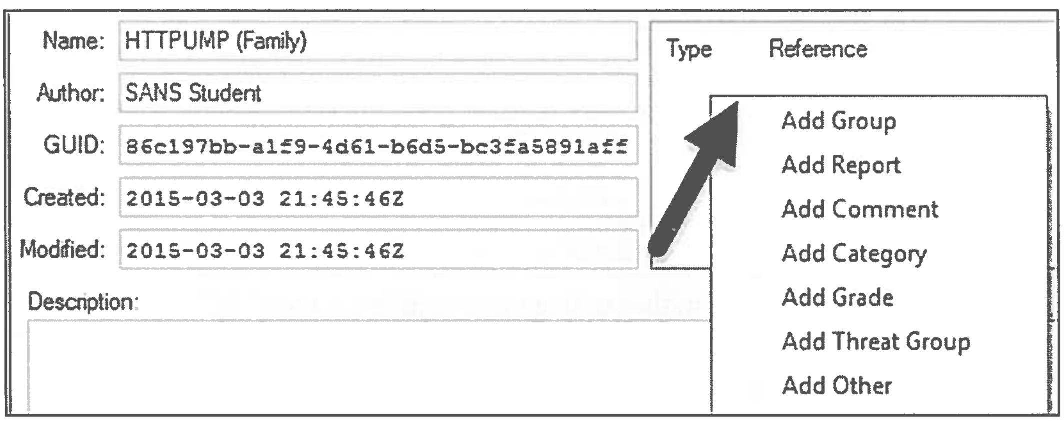
1. Select File -> New -> Indicator.



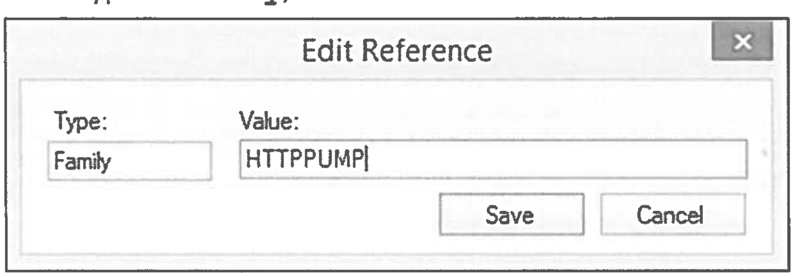
2. Set Name = **`HTTPPUMP (Family)`.**
3. Set Author = **`SANS Student`.**



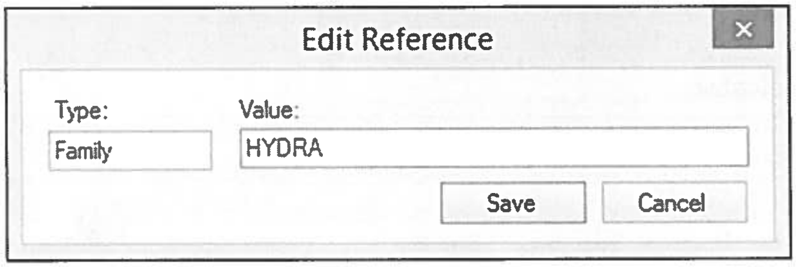
4. Right-click the area to the right of the Name and Author location, which should say "**`Type/Reference`.**"



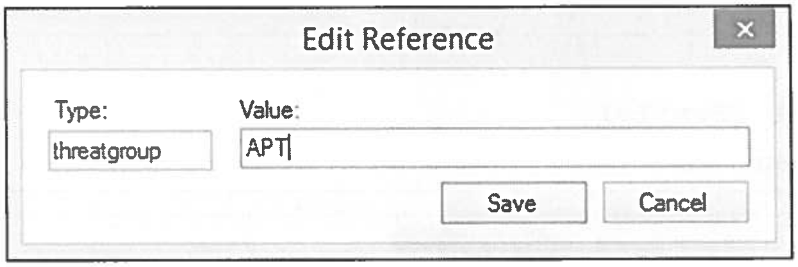
5. Right-click -> Add Other -> Type = **`Family`**, Value = **`HTTPPUMP`.**



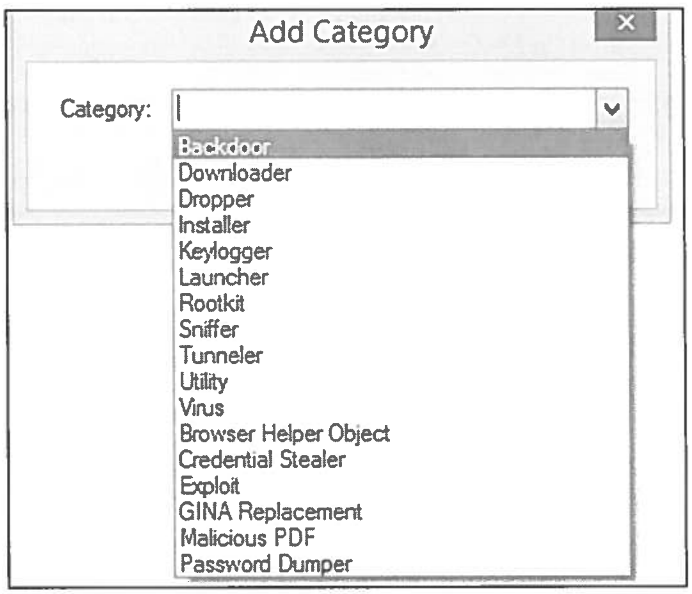
6. Right-click -> Add Other -> Type = **`Family`**, Value = **`HYDRA`**.



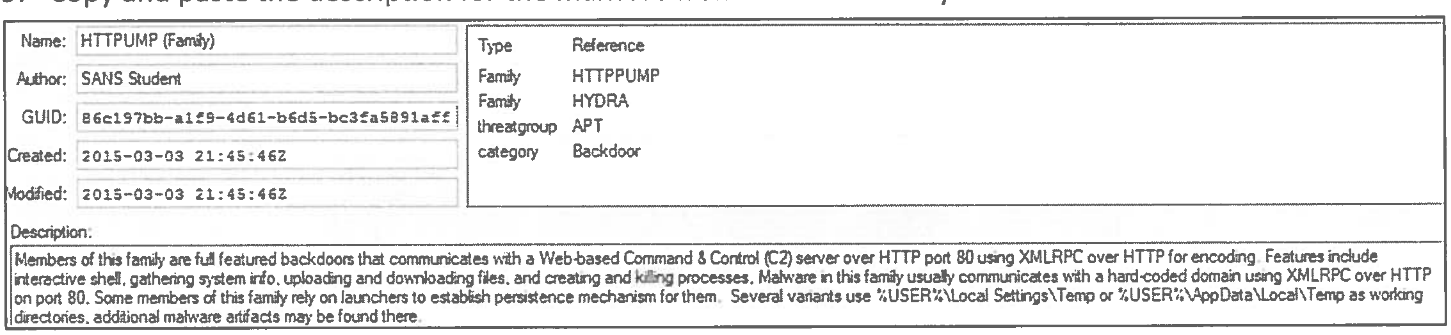
7. Right-click -> Add Threat Group -> Type "**`APT`**."



8. Right-click -> Add Category -> **`Backdoor`**.

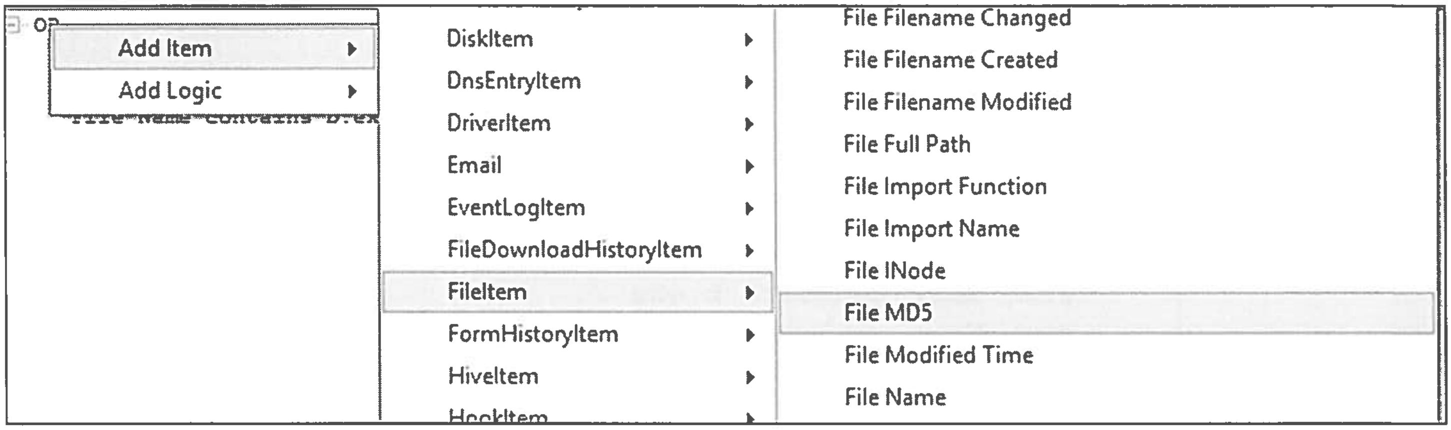


9. Copy and paste the description for the malware from the textfile on your USB.

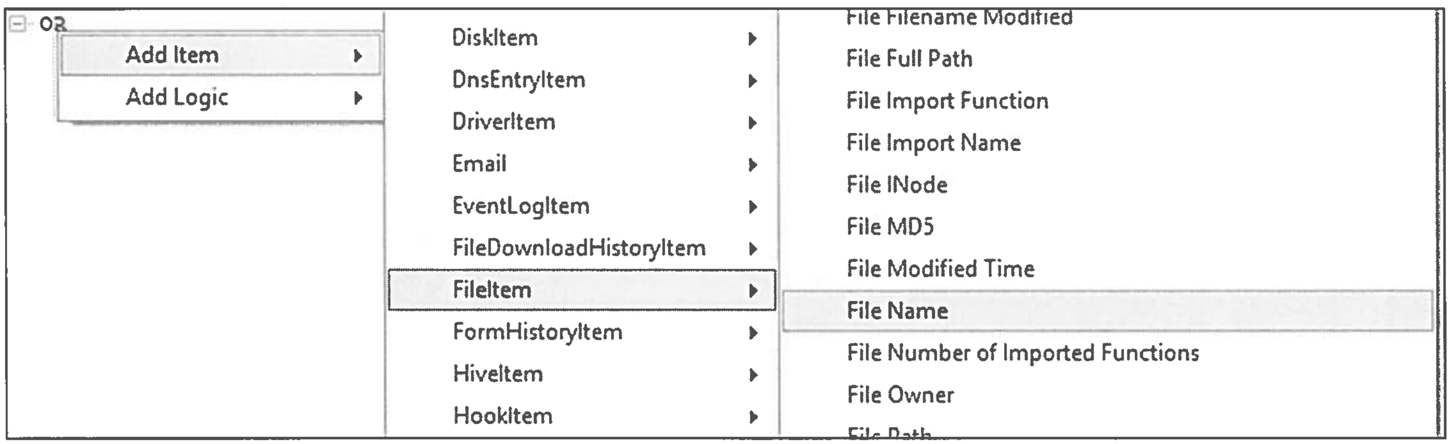


Exercise – Build the IOC Using the Provided Information

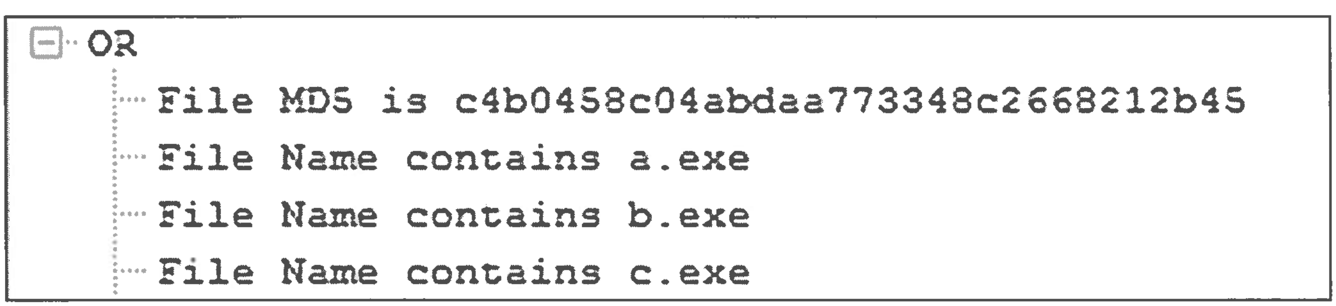
1. Right-click *1st* "OR" -> Add Item -> File Item -> File MD5 -> Type "**`c4b0458c04abdaa773348c2668212b45`**"



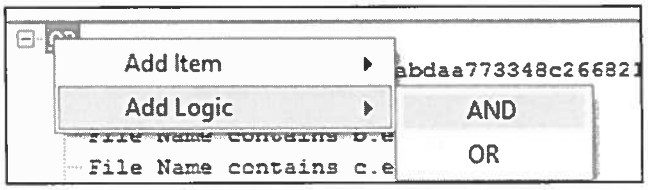
2. Right-click *1st* "OR" -> Add Item -> File Item -> File Name -> Type "**`a.exe`**"



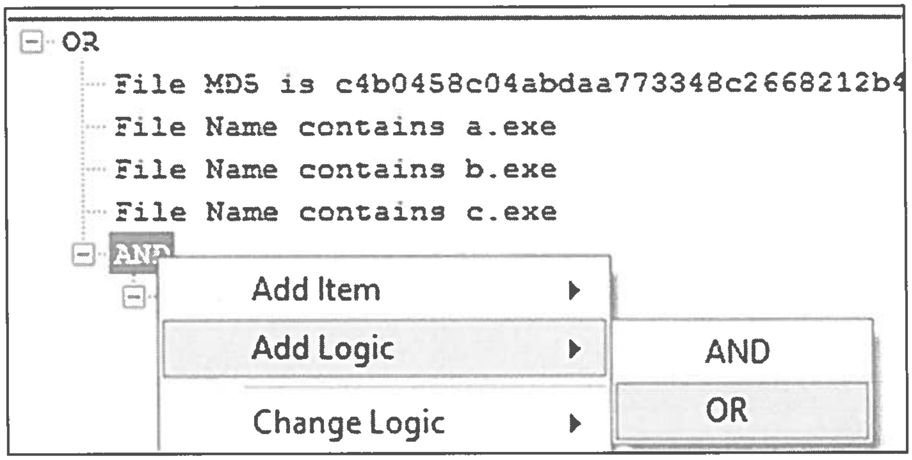
3. Right-click *1st* "**OR**" -> Add Item -> File Item -> File Name -> Type "**`b.exe`**"
4. Right-click *1st* "**OR**" -> Add Item -> File Item -> File Name -> Type "**`c.exe`**"
5. Check your work. Your IOC should look like this:



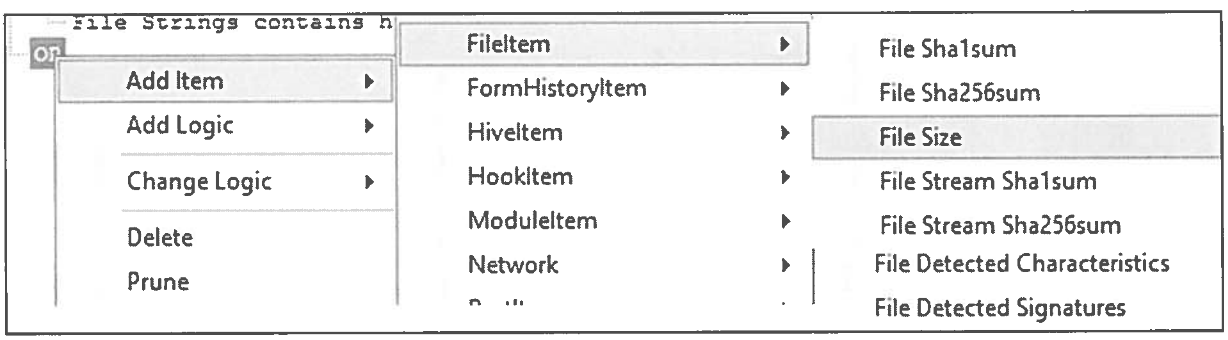
6. Right-click "**OR**" -> Add Logic -> Select "**AND**"



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

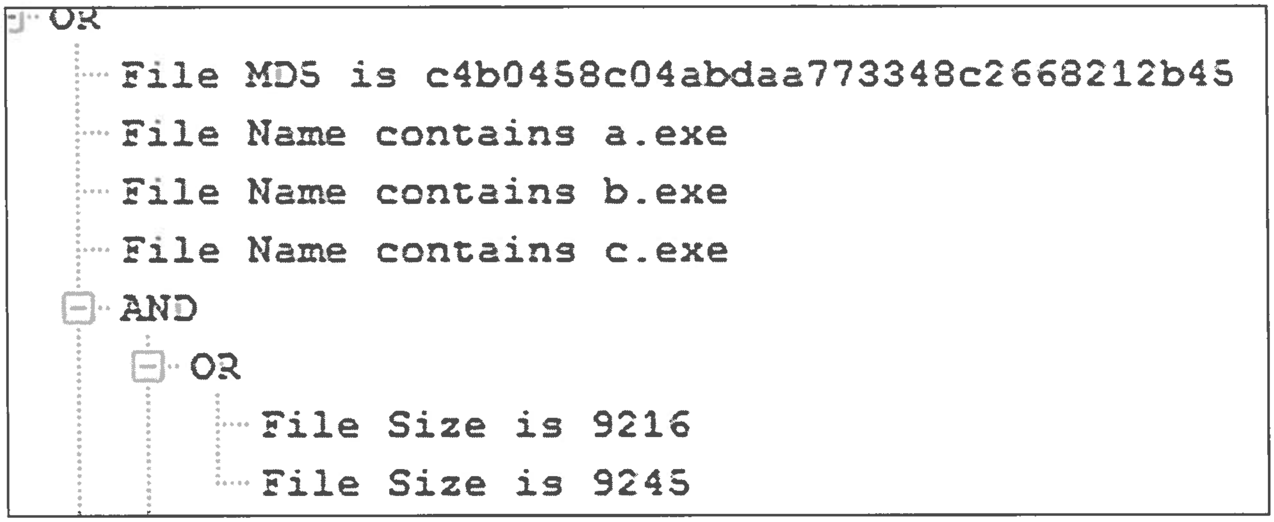


8. Right-click *2nd* "**OR**" -> Add Item -> File Item -> File Size -> Type "**9216**"

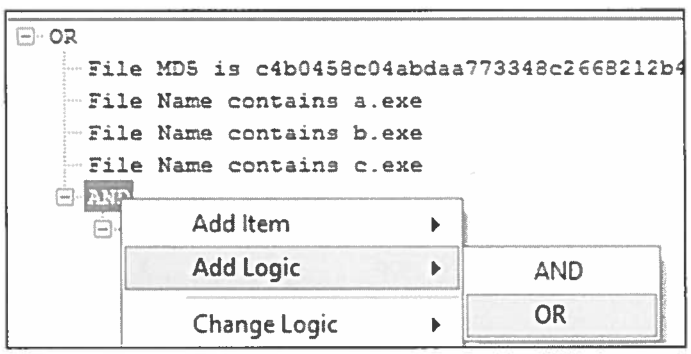


9. Right-click *2nd* "**OR**" -> Add Item -> File Item -> File Size -> Type "**9245**"

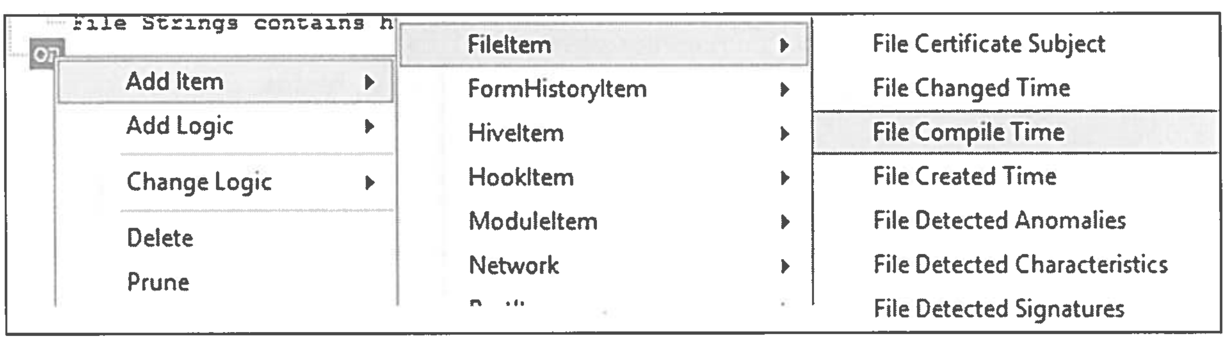
10. Check your work. Your IOC should look like this:



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



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



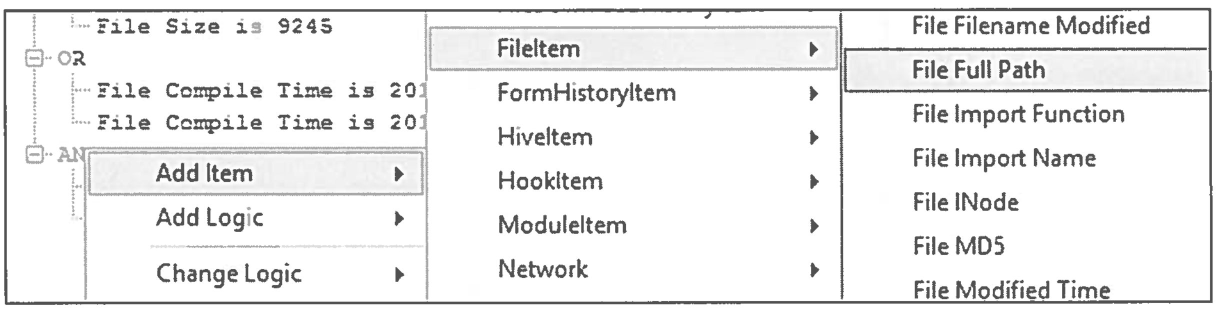
13. Right-click 3rd "OR" -> Add Item -> File Item -> File Compile Time -> Type "**`2011-10-19T02:39:12Z`**"

14. Check your work. Your IOC should now look like this:

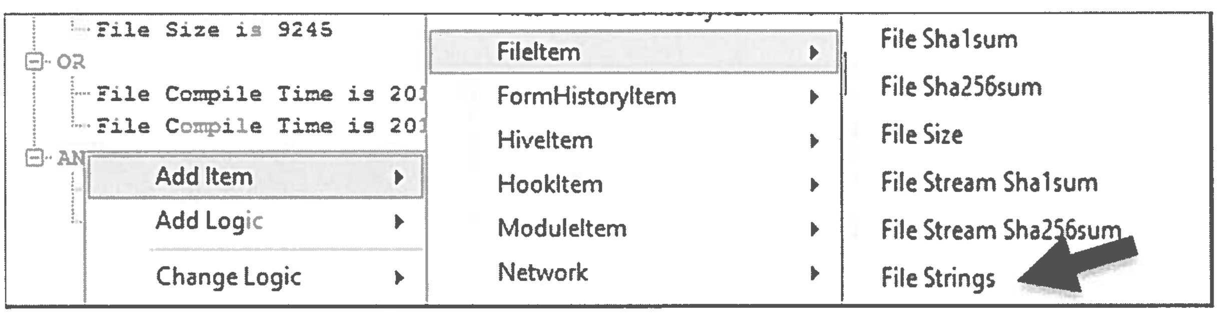
```
OR
    File MD5 is c4b0458c04abdaa773348c2668212b45
    File Name contains a.exe
    File Name contains b.exe
    File Name contains c.exe
    AND
        OR
            File Size is 9216
            File Size is 9245
        OR
            File Compile Time is 2011-10-13T04:19:53Z
            File Compile Time is 2011-10-19T02:39:12Z
```

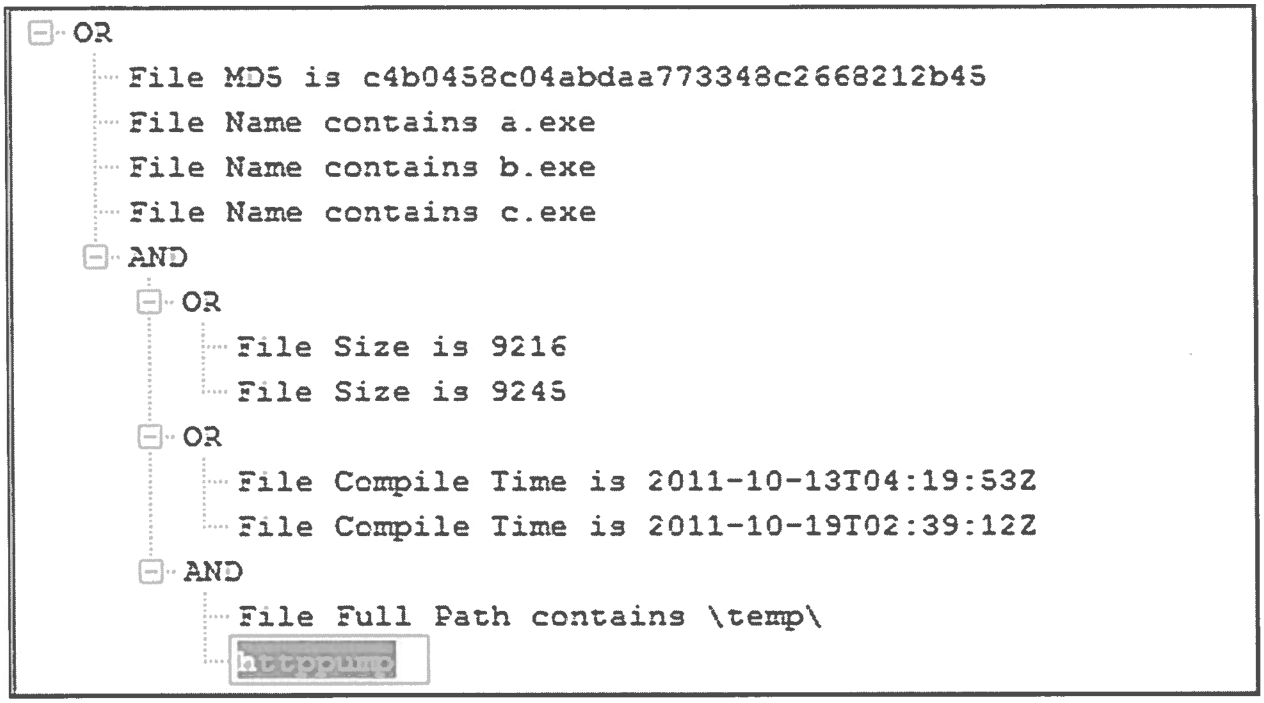
15. Right-click *1st* “**AND**” -> Add Logic -> Select “**AND**”

16. Right-click 2nd “AND” -> Add Item -> File Item -> File Full Path -> Type “\temp\”

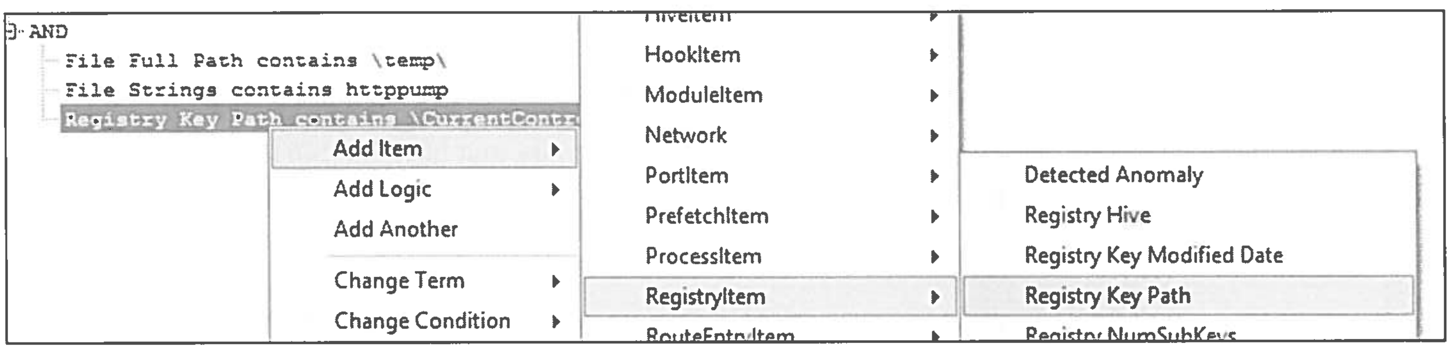


17. Right-click 2nd “AND” -> Add Item -> File Item -> File Strings -> Type “httppump”





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



19. Congrats!! You have completed and added your first IOC. 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 2011-10-19T02:39:12Z
      ⊟ AND
          File Full Path contains \temp\
          File Strings contains httppump
          Registry Key Path contains \CurrentControlSet\Services\Netman\domain
```

Exercise Takeaways

Building a good IOC is sometimes 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 important during incident response.

In upcoming sections, we will talk about tools like Redline and Volatility that have built-in OpenIOC support. Using capabilities such as Live Analysis with Redline, it is feasible to scan a single target with both host and process based indicators.

Understanding how an indicator is built and how it can be used to look for malware variants is important for incident response and hunting. The best information your malware analysis team can provide is a strong threat indicator for the host and the network scanning teams. Learning the basics of how to build one is useful and is as much of an art as a science.

As you work through an investigation, discovering artifacts and creating indicators is a key skill since threat indicator scanning is one of the best ways to scope large enterprises.

WIN10 DOT NET INSTALLATION ISSUES? – TRY THIS

1. Press Windows key and R at the same time
2. Type appwiz.cpl in the Run command box and press Enter
3. In the Programs and Features window, click on the link Turn Windows features on or off
4. Check if the .NET Framework 3.5 (includes .NET 2.0 and 3.0) option is available in it
5. If yes, then enable it and then click on OK
6. Follow the on-screen instructions, complete the installation and restart the computer, if the process requests that from you

Besides installing .NET Framework 3.5 through the Control Panel, you can also install it on-demand. If a certain app requires the .NET Framework 3.5, but this feature isn't enabled on your computer, the installation wizard will display the prompt for installing .NET Framework 3.5 on demand. To install .NET Framework 3.5, in this prompt choose the option "Install this feature," and .NET Framework 3.5 will be automatically installed on your computer.

But some users have reported that they get a certain error message after they try to install .NET Framework 3.5 through the Control Panel or on-demand. To avoid this error, try to install .NET Framework 3.5 with the Command Prompt. To do that, follow these steps:

1. Press Windows key and R at the same time
2. Type cmd in the Run dialog box and press Enter
3. And enter the following line in the Command Prompt: DISM /Online /Enable-Feature /FeatureName:NetFx3 /All /LimitAccess /Source:installationMediaDrive:\sources\sxs

This page intentionally left blank.

Exercise 1.4 – Autostart Persistence Analysis

Objectives

1. Filter and examine output from **autorunsc.exe**.
2. Identify suspicious processes using the "What Is Normal Poster?" and an understanding of core Windows operating system processes and services

Exercise – Are There Suspicious Persistence Mechanisms on 10.3.58.5?

The **autorunsc.exe** tool was executed during live response on system 10.3.58.5 (example only – you do not need to run this as the output file has already been provided to you).

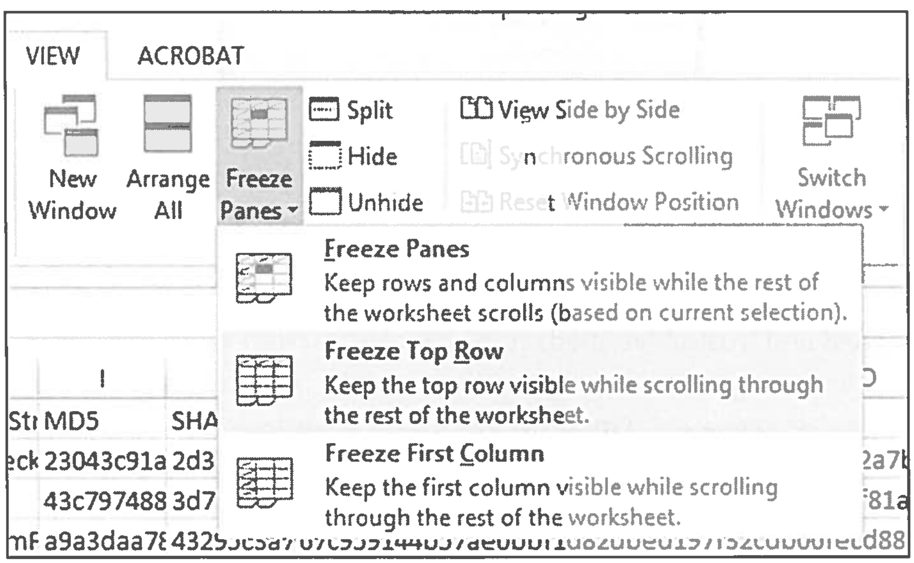
```
C:\> autorunsc.exe -accepteula -a * -s -h -c > autoruns-win7-32-nromanoff.csv
```

Your mission in this exercise is to review the output file and find evil.

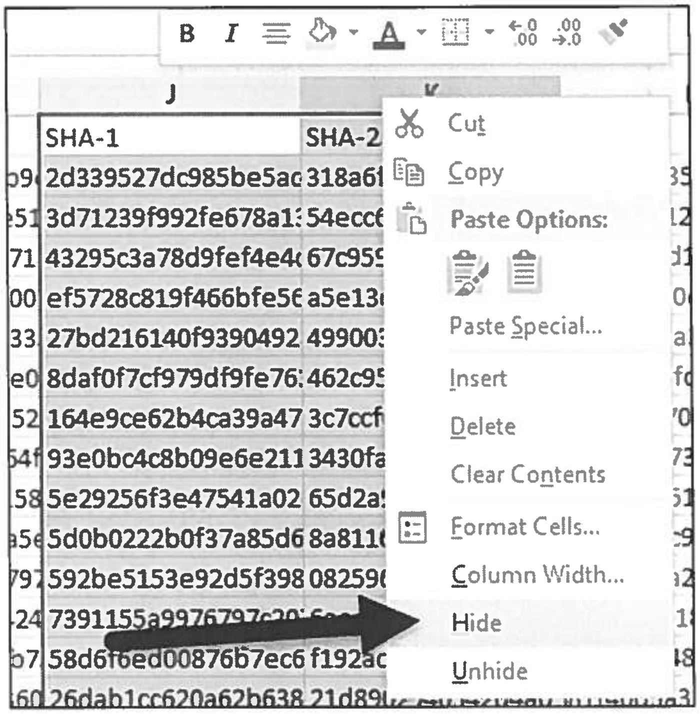
STEP-BY-STEP – Autoruns Filtering

1. Using Microsoft Excel, open the **autoruns-win7-32-nromanoff.csv** file found on your USB under **win7-32-nromanoff-10.3.58.5\win7-32-nromanoff-incident-response**. Simply double-clicking **autoruns-win7-32-nromanoff.csv** usually works.

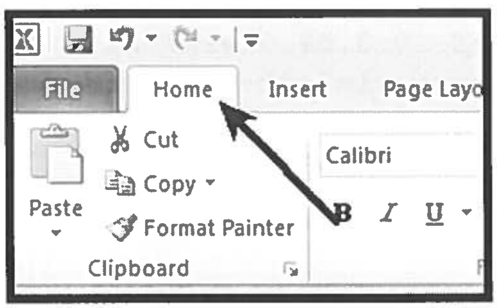
2. Select View -> Freeze Panes -> Freeze Top Row.



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



4. Select Home Ribbon.



5. Select All Cells or "CTRL-A."

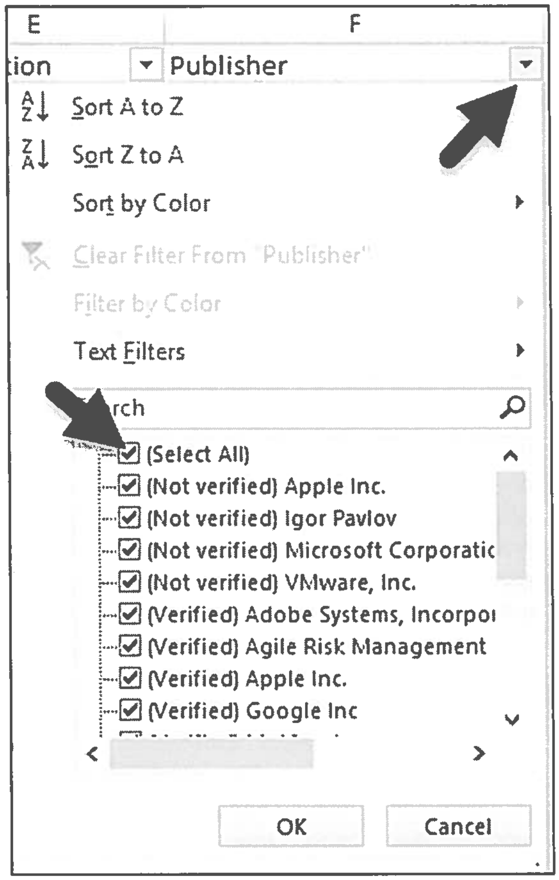
6. In Home Ribbon -> Sort and Filter -> Filter

7. Filter out all signed and trusted (verified) startup locations from view.

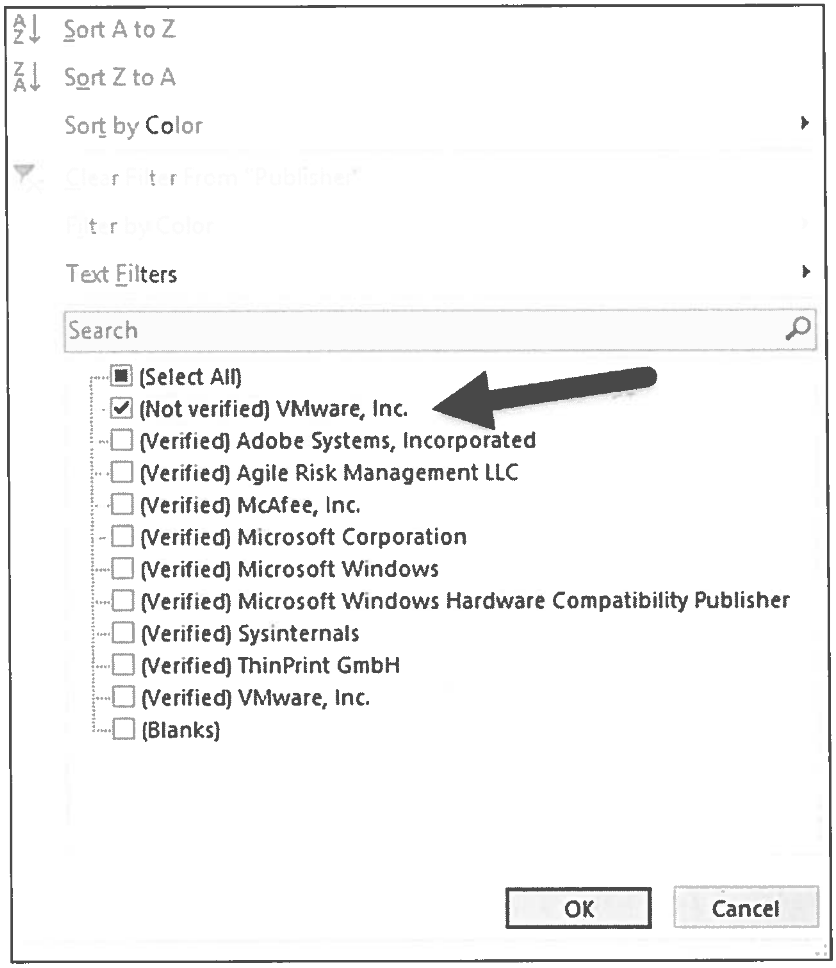
   Select the "Publisher" column and filter for only those publishers that you are not sure about. Although this can lead to false negatives, it is a reasonable first step towards data reduction, making abnormalities easier to identify.

   Note: There are some examples of well-known trusted publishers 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. Although not impossible, it is extremely rare that malware will be using a trusted certificate from Microsoft to sign its code.

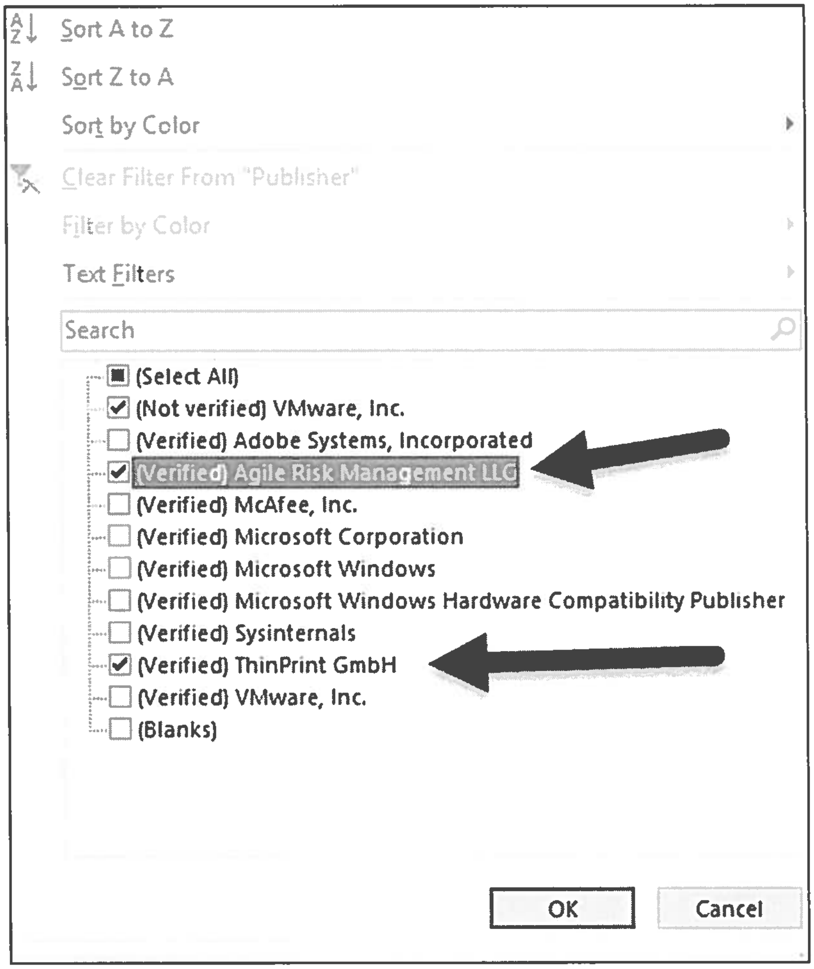
a. Select the drop-down filter next to "**Publisher**" and unselect "(Select All)".



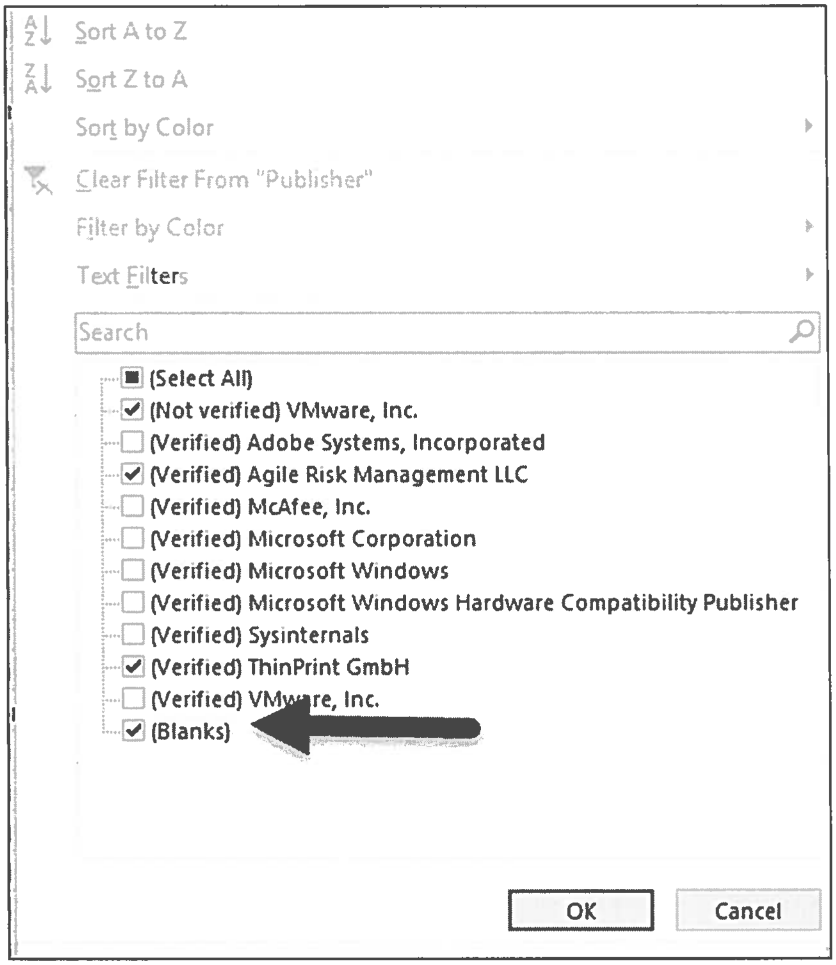
b. Select all publishers that are "**Not Verified.**"



c. Also select any publishers not immediately recognized as a core software publisher (for example, something you might need to "Google search" to verify they are a real company).



d. Finally, before you click "OK," also select "`blanks.`" Blanks include those entries with no publisher information and no digital signature.



8. You are now ready to start your analysis!

Exercise – Autoruns Output Analysis

For this exercise:

1. Briefly review your filtered autorunsc.exe output (or use the excerpt on the next page).

2. Record the "Image Path" for any suspicious entries worth investigating further. You do not need to use every line provided. (Hint: "File not found" means there is an entry pointing to a file that is no longer present in the file system. This could be caused by an uninstalled application that did not clean up its autostart entry. They are worth reviewing, but not inherently suspicious.)

   ____

   ____

   ____

   ____

   ____

3. Compare image paths identified in the last question against any system processes that are present on the "What Is Normal" Poster (available in digital form on your SIFT desktop and on your class USB). What is the most likely candidate for a suspicious process?

   ____

4. List what is wrong or odd about the selected suspicious process. (**Advanced:** There are at least five things wrong with the Autoruns entry).

   ____

   ____

   ____

   ____

   ____

5. What resources could you use to check your other suspicious processes for validity?

   ____

Entry Location	Entry	Category	Image Path	Launch String
HKLM\System\CurrentControlSet\Services	FResponse Service	Services	c:\windows\system32\f-response-ent.exe	f-response-ent.exe
HKLM\System\CurrentControlSet\Services	mfeavfk01	Drivers	File not found: C:\Windows\System32\Drivers\mfeavfk01.sys	mfeavfk01
HKLM\System\CurrentControlSet\Services	Synth3dVsc	Drivers	File not found: System32\drivers\synth3dvsc.sys	System32\drivers\synth3dvsc.sys
HKLM\System\CurrentControlSet\Services	tsusbhub	Drivers	File not found: system32\drivers\tsusbhub.sys	system32\drivers\tsusbhub.sys
HKLM\System\CurrentControlSet\Services	VGPU	Drivers	File not found: System32\drivers\rdvgkmd.sys	System32\drivers\rdvgkmd.sys
HKLM\SYSTEM\CurrentControlSet\Control\Print\Monit	ThinPrint Print Port N	Print Monitors	c:\windows\system32\tpvmmon.dll	TPVMMon.dll
HKLM\System\CurrentControlSet\Services\WinSock2\Pa	VMCI sockets DGRAN	Network Providers	c:\program files\vmware\vmware tools\vsock sdk\bin\win32\vsocklib.dll	C:\Program Files\VMware\VMware Tools\VSock SDK\bin\win32\vsocklib.dl
HKLM\System\CurrentControlSet\Services\WinSock2\Pa	VMCI sockets STREAN	Network Providers	c:\program files\vmware\vmware tools\vsock sdk\bin\win32\vsocklib.dll	C:\Program Files\VMware\VMware Tools\VSock SDK\bin\win32\vsocklib.dl
HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\	svchost	Logon	c:\windows\system32\dllhost\svchost.exe	c:\windows\system32\dllhost\svchost.exe
Task Scheduler	\Microsoft\Windows\	Tasks	File not found: %ProgramW6432%\Windows Media Player\wmpnscfg.exe	"%ProgramFiles%\Windows Media Player\wmpnscfg.exe"

STEP-BY-STEP – Autoruns Output Analysis

1. Briefly review your filtered autorunsc.exe output (or use the excerpt on the previous page).

2. Record the "Image Path" for any suspicious entries worth investigating further. You do not need to use every line provided. (Hint: "File not found" means there is an entry pointing to a file that is no longer present in the file system. This could be caused by an uninstalled application that did not clean up its autostart entry. They are worth reviewing, but not inherently suspicious.)

   c:\windows\system32\f-response-ent.exe (has a verified digital signature)

   c:\windows\system32\tpvmmon.dll (has a verified digital signature)

   c:\program files\vmware\vmware tools\vsock sdk\bin\win32\vsocklib.dll

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

   While some of the "File not found" entries could look suspicious, they are not currently present and thus cannot be active persistence mechanisms. Also note that since they were not present, Autoruns could not provide their description, publisher, and digital signature verification. In a real investigation, it might be worth checking against a baseline for those file names or doing a cursory Google search, but we probably would not invest a lot of time.

3. Compare image paths identified in the last question against any system processes that are present on the "What Is Normal" Poster (available in digital form on your SIFT desktop and on your class USB). What is the most likely candidate for a suspicious process?

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

check Evil-poster on SIFT desktop.

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; although, 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 (for example, 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.

4. List what is wrong or odd about the selected suspicious process. (**Advanced:** There are at least five things wrong with the Autoruns entry).

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. (Given this, we would expect the Entry Location to be HKLM\System\CurrentControlSet\Services.)**
c. **Because svchost.exe is present in a "Run" registry key, its start time will be when a user logs on to the system, not at boot as identified on the poster.**
d. **The Launch String (command line) should include a "-k" parameter 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 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 actual file 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!

5. What resources could you use to check your other suspicious processes for validity?

   After checking for digital signatures, the next logical step in checking validity of suspicious entries would be to compare the file hash provided by Autoruns against a known good hash database. We'll discuss hash databases later in the course. Researching the vendor and product listed in the "Publisher" and "Description" fields would be another way to gather more information. Finally, comparing this output with baselined output from a similar system (hopefully known to not be compromised) could help to identify abnormalities.

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 to verify potential intrusions.

Unless you already know and understand Windows OS processes, it is fairly difficult to identify suspicious processes solely via tool output. Although having a big collection of output from various incident response tools is nice, it is not useful if you cannot quickly and easily identify anomalies. 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 **`10.3.58.5`**, we have identified a suspicious process called **`svchost`** in an incorrect directory **`c:\windows\system32\dllhost`** that starts automatically when a user logs into the system.

Looking up processes on the "What is Normal" poster was intended to help facilitate memorization of a key Windows process, **`svchost.exe`**, and the properties that it should exhibit on a Windows operating system. You should also take it as homework to familiarize yourself with the other processes listed on the "What is Normal" poster.

Shortly we will begin memory analysis and attempt to identify if **`svchost`** is running and if it has any additional suspicious indicators.

This page intentionally left blank.

Optional Exercise - Exercise 1.5A – 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 is typically in classes where we are live and have limited access to additional systems for this exercise.

When you return 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, but 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, but it also 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. *Log in 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 the 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. Log in to see the available nodes. (Note in this example that 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 provides 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. Log in to Remote Hard Drive Target. It is usually Disk-0. Note in the preceding example, my system name was **`jotumheim`**. The exact name of the system you are logging in to will change for each system you are logging in to. 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 following command.

```
# 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 the output locating a 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 the 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
```

Exercise – Attach remote memory and perform some basic memory examination commands

1. Log in to the 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.jotumheim: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.com.f-response.jotumheim:pmem, portal: 192.168.112.1,3260]
Login to [iface: default, target: iqn.2008-02.com.f-response.jotumheim:pmem, portal: 192.168.112.1,3260]: successful
IQN:iqn.2008-02.com.f-response.jotumheim: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, you will learn how to use Volatility and Redline during memory analysis.

Some examples of commands we could run if connected over to memory on a remote system

View running processes **`# vol.py -f /dev/sdd pslist`**
View open TCP network connections **`# vol.py -f /dev/sdd connections`**
Recover Internet Explorer history from RAM **`# vol.py -f /dev/sdd iehistory > /cases/iehistory.txt`**
`# less /cases/iehistory.txt`
View registry hives present in memory **`# vol.py -f /dev/sdd hivelist`**
Parse Userassist key from RAM **`# vol.py -f /dev/sdd userassist > /cases/userassist.txt`**
`# less /cases/userassist.txt`

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 systems 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 you the ability to scale from a single system to thousands.

Optional Exercise - Exercise 1.5B – 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 have 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 also 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 a live conference or onsite course, the instructor will set up a demonstration system (or two) to have you connect to; please follow Exercise 4A. Obtain that IP address from the instructor.

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

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

1. Install and configure an enterprise incident response and forensics capability.
2. Create a deployable agent.
3. Deploy agents on to target systems in your environment.
4. 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. *Log in 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 the 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.

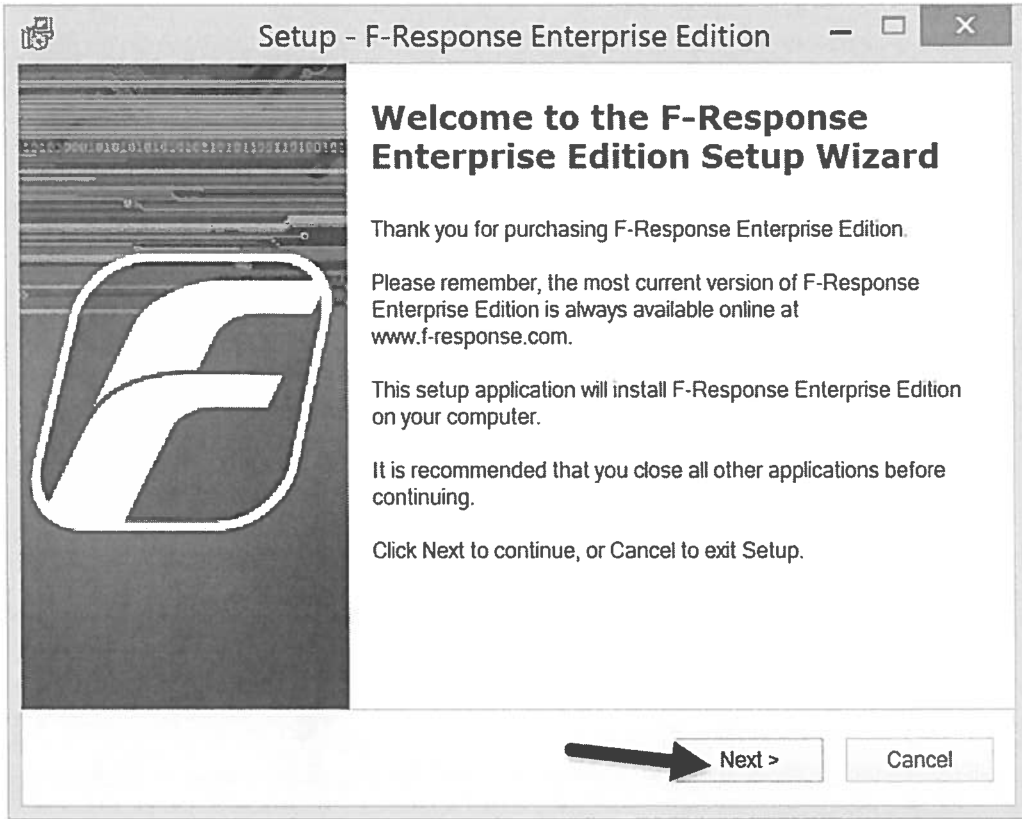
6. <u>**Windows system: Turn off**</u> your windows system firewall or open up ports on your enterprise network to allow for the following ports to communicate over TCP (Control Panel -> System and Security -> Windows Firewall -> Turn Firewall On or Off) .

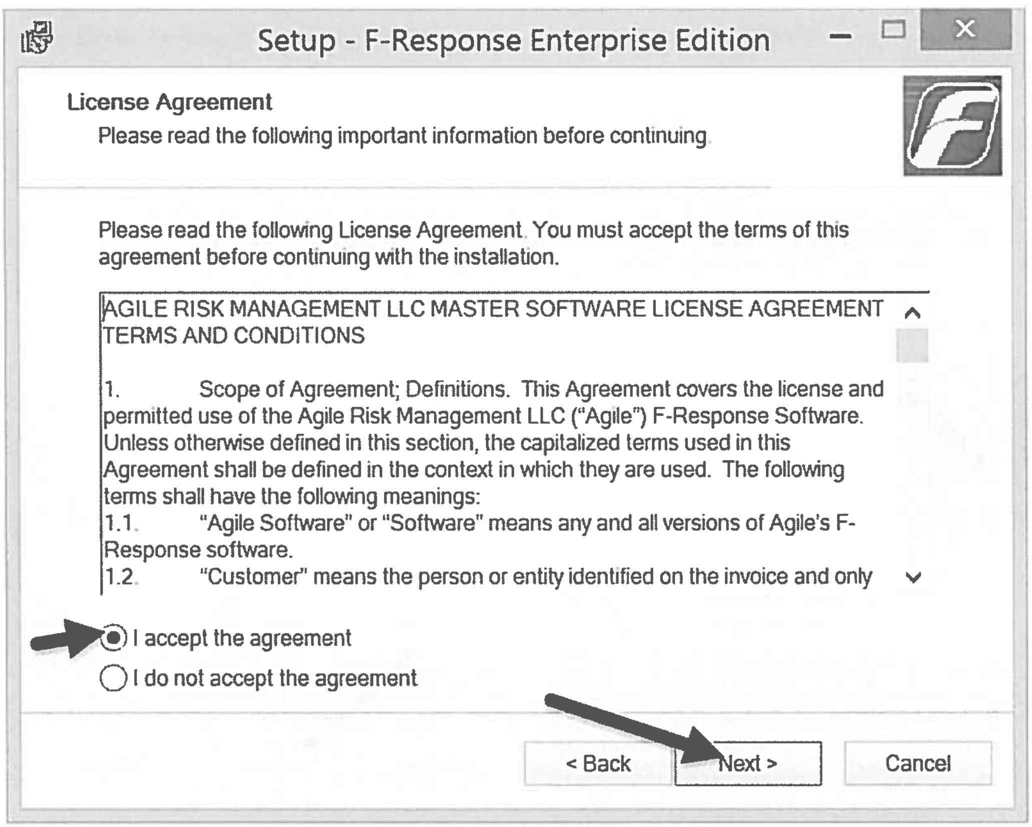
   1. **3260**: This is the main port F-Response uses to communicate with the target machine's resources (iSCSI TCP).

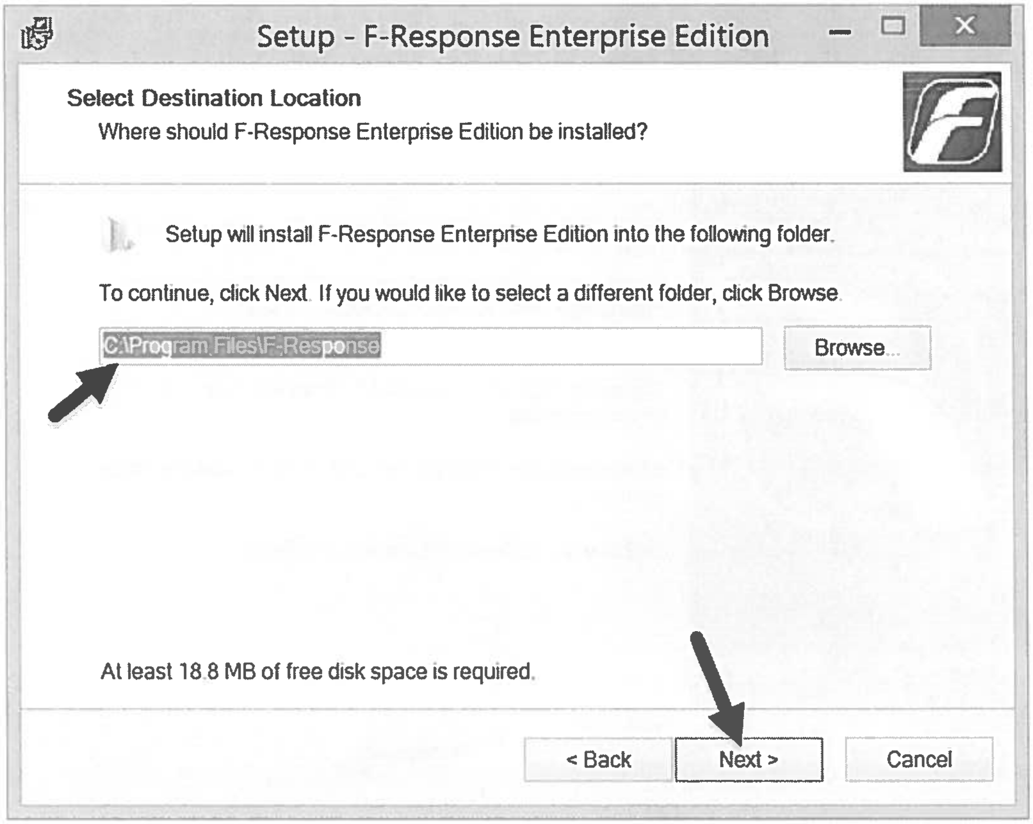
   2. **5681**: This port is used to communicate with the F-Response Licensing Manager. (TCP is used for most editions of F-Response; UDP is used for F-Response Tactical Edition.)

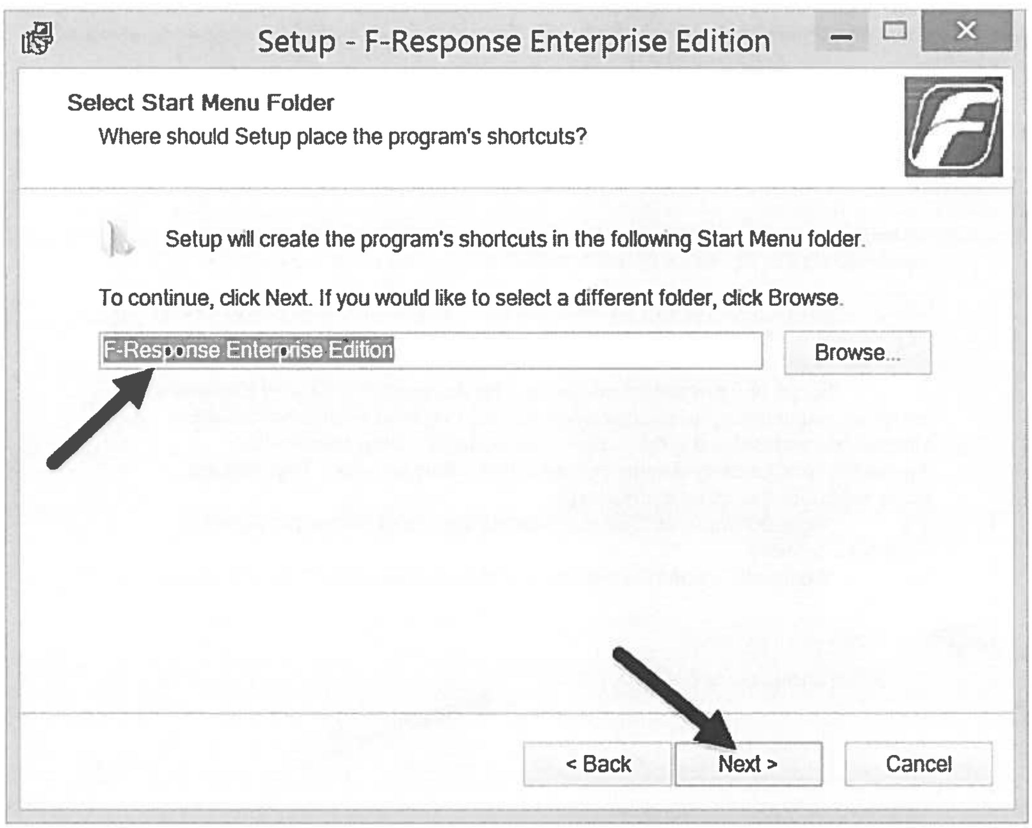
7. Insert your **COURSE USB with DFIR** on the side into your Windows system.
   - Navigate to \SIFT-Lab-Install\F-Response-Enterprise.

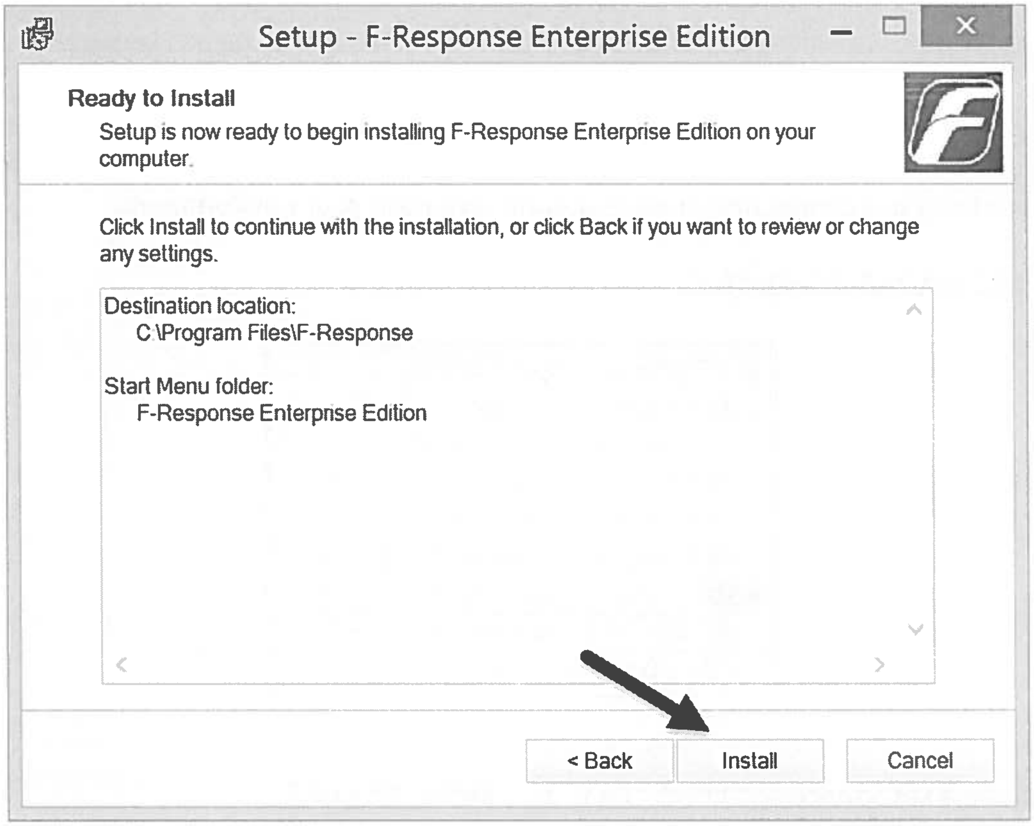
8. Install "**F-Response Enterprise**" software (**F-ResponseEnterprise.exe**) on your Windows system. All steps require you to simply select "**Next**" and finally "**Install**" and "**Finish.**"

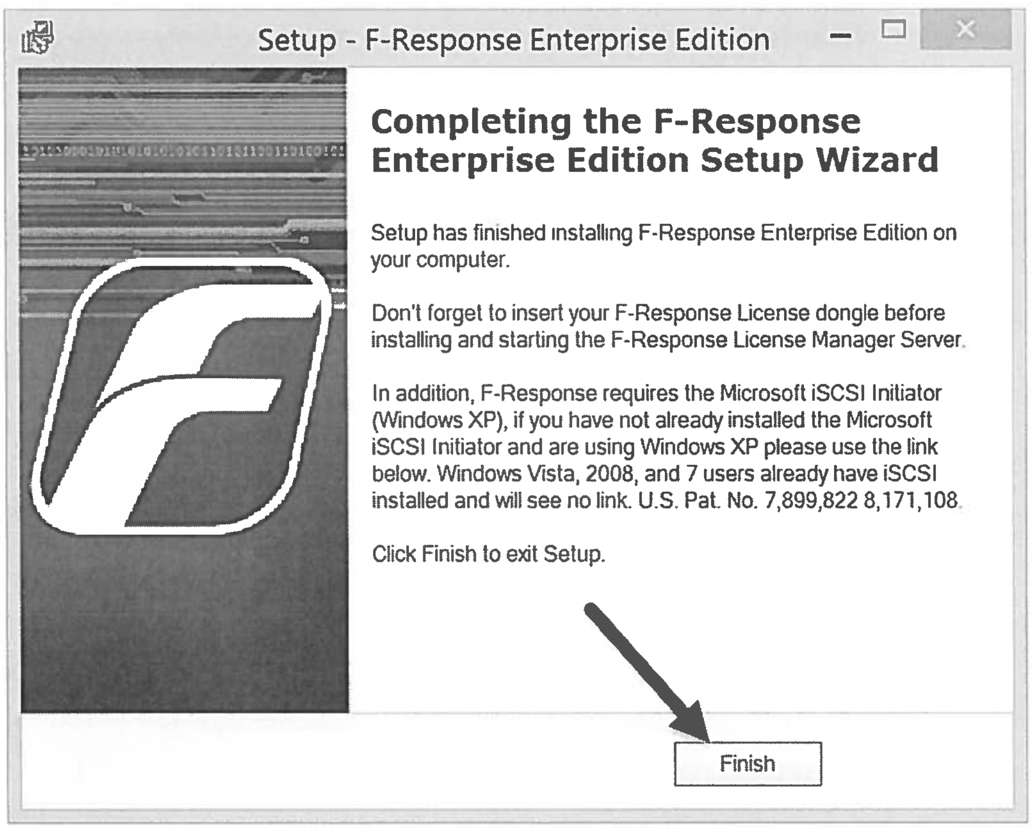

Setup - F-Response Enterprise Edition
Welcome to the F-Response Enterprise Edition Setup Wizard
Thank you for purchasing F-Response Enterprise Edition.
Please remember, the most current version of F-Response Enterprise Edition is always available online at www.f-response.com.
This setup application will install F-Response Enterprise Edition on your computer.
It is recommended that you close all other applications before continuing.
Click Next to continue, or Cancel to exit Setup.
Next >
Cancel


Setup - F-Response Enterprise Edition
License Agreement
Please read the following important information before continuing.
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.
AGILE RISK MANAGEMENT LLC MASTER SOFTWARE LICENSE AGREEMENT TERMS AND CONDITIONS
1. Scope of Agreement; Definitions. This Agreement covers the license and permitted use of the Agile Risk Management LLC ("Agile") F-Response Software. Unless otherwise defined in this section, the capitalized terms used in this Agreement shall be defined in the context in which they are used. The following terms shall have the following meanings:
1.1. "Agile Software" or "Software" means any and all versions of Agile's F-Response software.
1.2. "Customer" means the person or entity identified on the invoice and only
I accept the agreement
I do not accept the agreement
< Back
Next >
Cancel


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

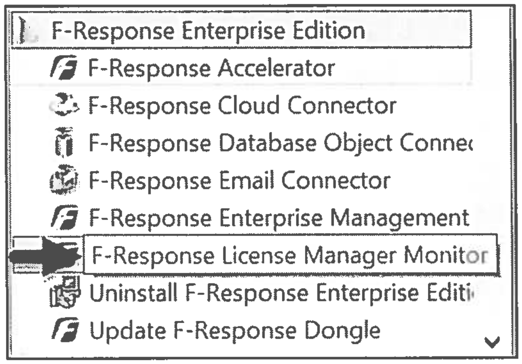

Setup - F-Response Enterprise Edition
Ready to Install
Setup is now ready to begin installing F-Response Enterprise Edition on your computer.
Click Install to continue with the installation, or click Back if you want to review or change any settings.
Destination location:
C:\Program Files\F-Response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

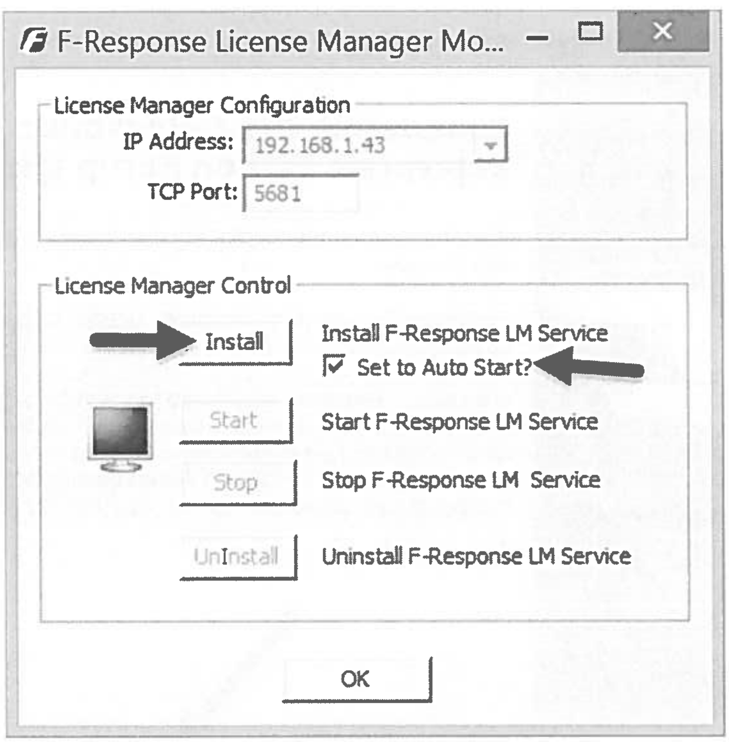
Exercise – Set Up the F-Response Enterprise License Manager

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.

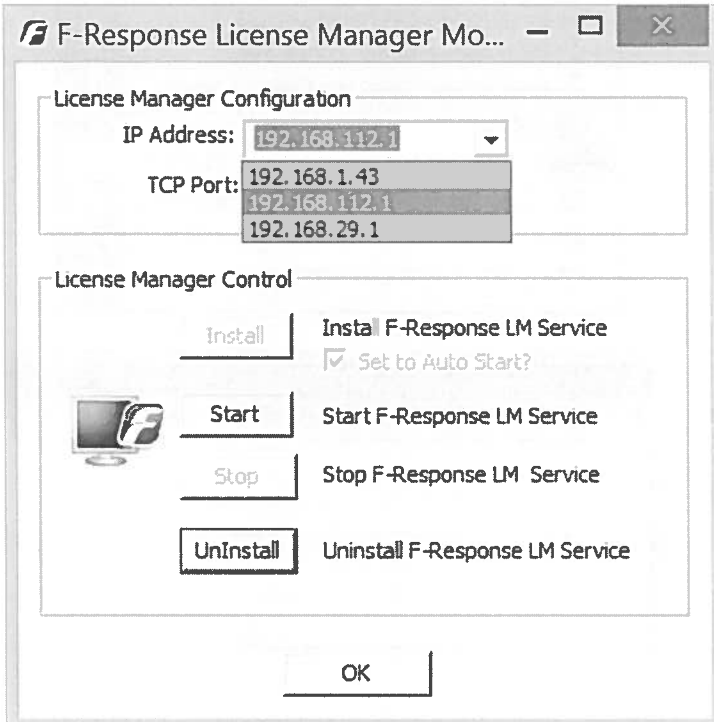
2. Launch **LICENSE MANAGER MONITOR**.



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



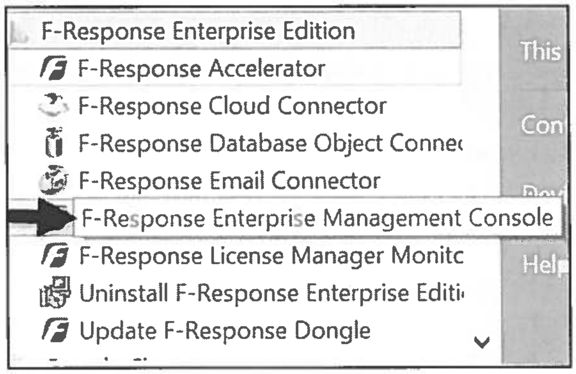
4. From the "IP Address:" drop-down select the IP address that matches the same subnet of your SIFT Workstation IP address documented in step 4 during your preceding Exercise Preparation. NOTE: In the example listed here, our SIFT IP address was "**`192.168.112.138`**"; thus the IP address of the License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



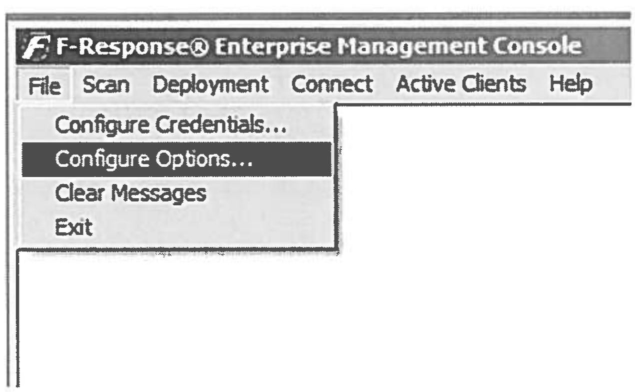
5. Start the "**`F-Response LM Service`**" and select "**`OK.`**"

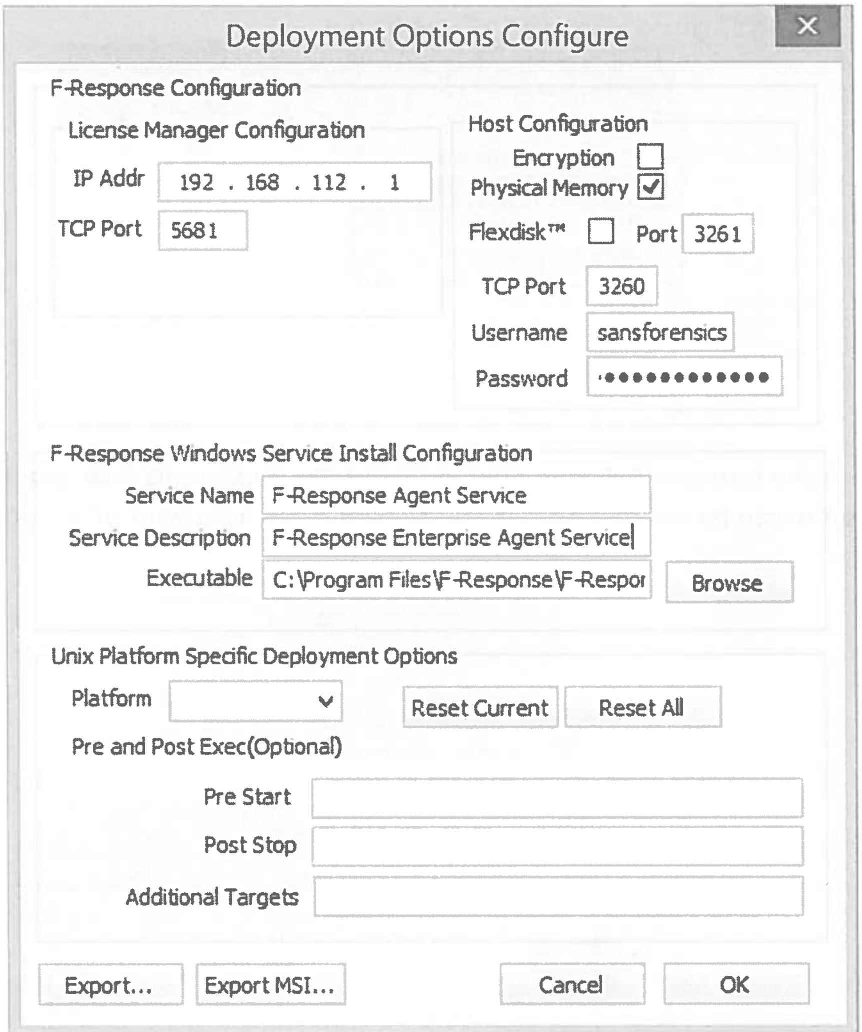
Exercise – Create a Deployable Remote Forensic Agent

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



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



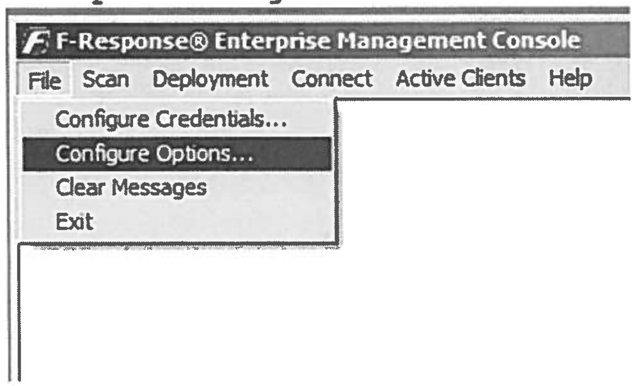


You will need to configure the following:

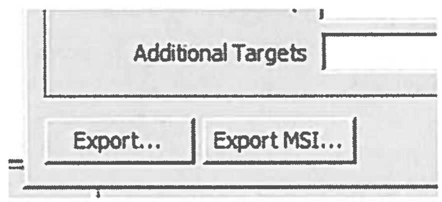
- License Manager IP -> **DO NOT CHANGE**
- License Manager Port -> **5681 – DO NOT CHANGE**
- Host Configuration-> **Physical Memory-> CHECKED**
- Username -> **`sansforensics`**
- Password -> **`forensics1234`**
- Service Name -> **`F-Response Agent Service`**
- Service Desc-> **`F-Response Enterprise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

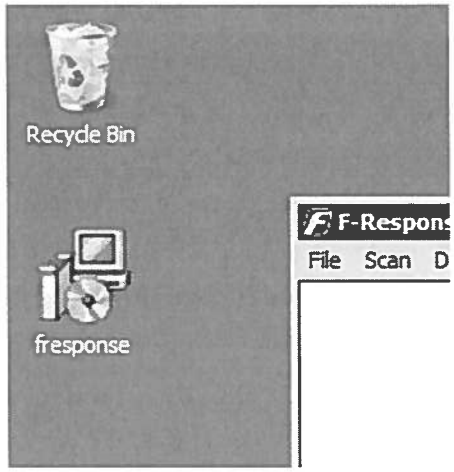
Press **Ok** to save changes.

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



4. Click the button at the bottom of the window to Export the deployable **F-ResponseAgent.MSI.** When 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 it.**

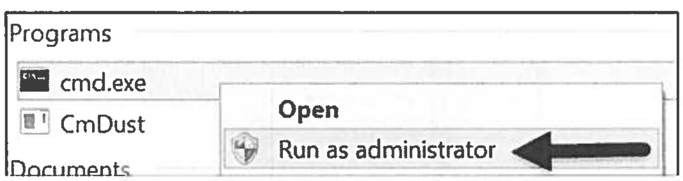
Typically, we would deploy the agent to other systems in our environment via Group Policy or the F-Response Enterprise Management Console.

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              :
```

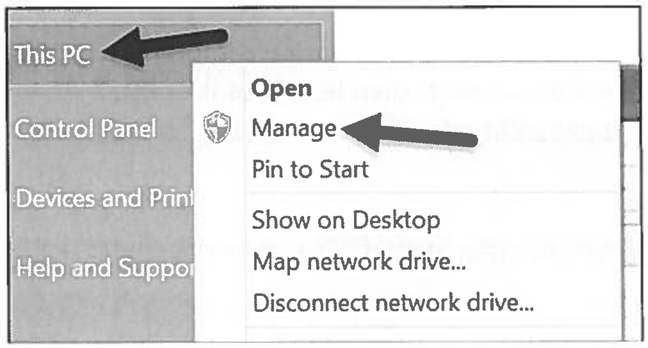
7. Write down the IP address of the system you deployed the AGENT to by typing “ipconfig”

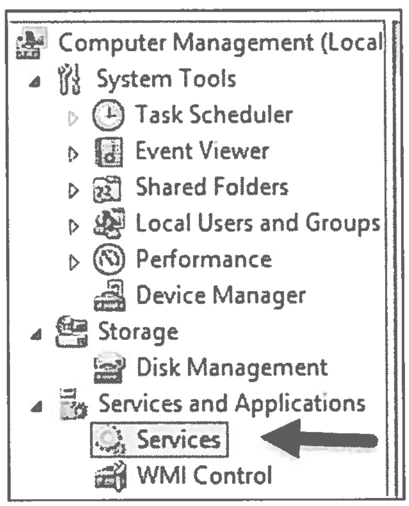
AGENT IP ADDRESS = ____

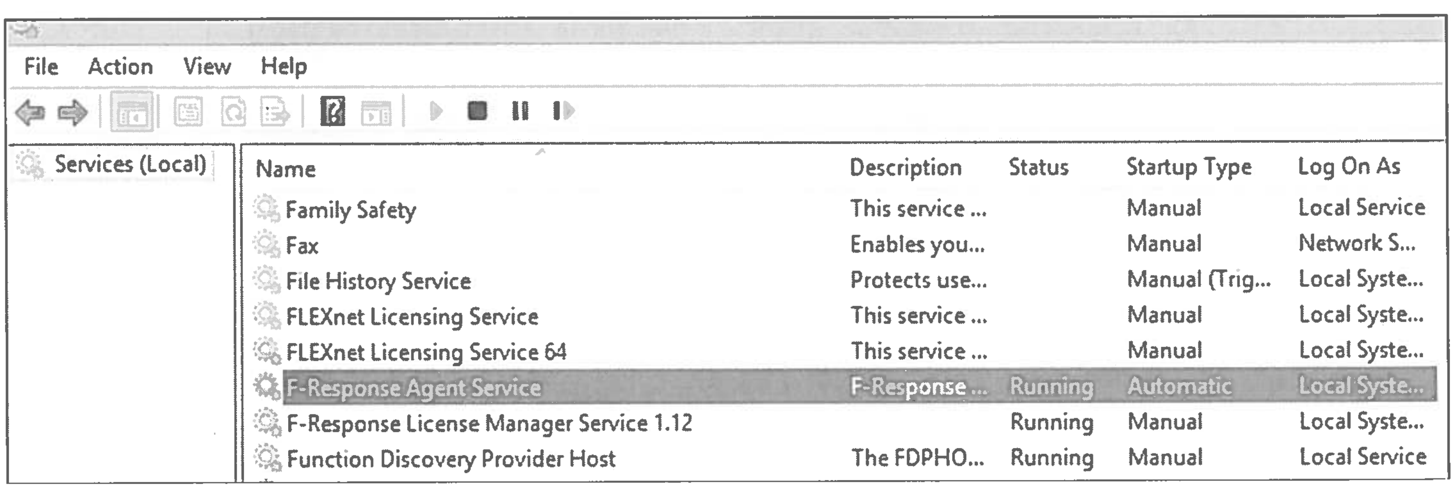
In this example, we are likely only using VMNET8.

8. OPTIONAL -> Using the Services GUI, start the “F-Response Agent Service.”

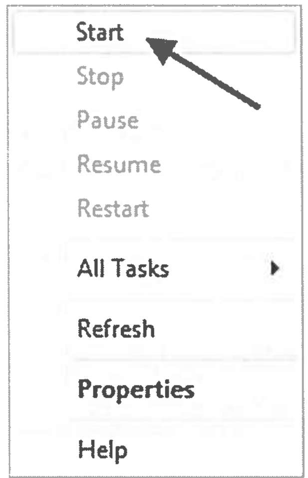
Right-click “Your System” and select “Manage.”







Right-click and 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. Log in to see the available nodes. (Note in this example that 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. Log in to remote hard drive target; usually Disk-0. Note in the preceding 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 a 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 a 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

1. Log in 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.jotumheim: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.com.f-response.jotumheim:pmem, portal: 192.168.112.1,3260]
Login to [iface: default, target: iqn.2008-02.com.f-response.jotumheim:pmem, portal: 192.168.112.1,3260]: successful
IQN:iqn.2008-02.com.f-response.jotumheim: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, you will learn how to use Volatility and Redline during memory analysis.

Some examples of commands we could run if connected over to memory on a remote system

View running processes	**# vol.py -f /dev/sdd pslist**
View open TCP network connections	**# vol.py -f /dev/sdd connections**
Recover Internet Explorer history from RAM	**# vol.py -f /dev/sdd iehistory > /cases/iehistory.txt** **# less /cases/iehistory.txt**
View registry hives present in memory	**# vol.py -f /dev/sdd hivelist**
Parse Userassist key from RAM	**# vol.py -f /dev/sdd userassist > /cases/userassist.txt** **# less /cases/userassist.txt**

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. Is the F-Response Agent Service not starting correctly?

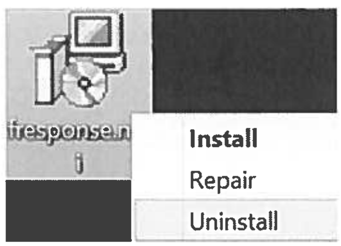
    a. Ensure your F-Response Enterprise Dongle is inserted and you have started the F-Response Enterprise License Manager.

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

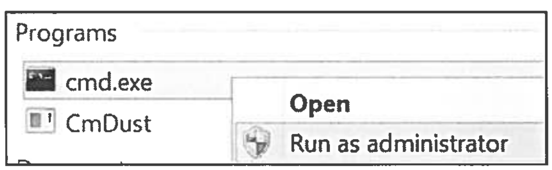
    c. Check the configuration options set in the preceding step 2 of “Creating a Remote Deployable Agent” portion of this exercise. Re-create the **`.msi`** file if necessary.

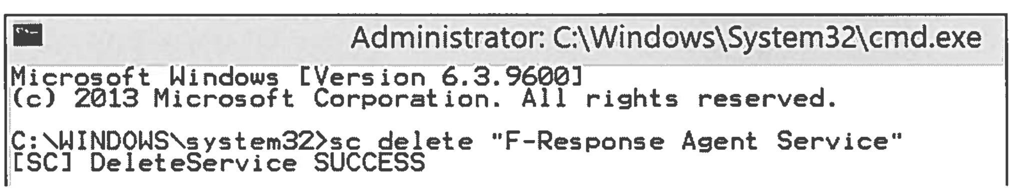
2. Did you misconfigure the F-ResponseAgent.MSI and need to reconfigure/reinstall it?

    a. Right-click 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 you the ability to scale from a single system to thousands.

Optional Exercise 2.1 – Redline Pre-Process

Objectives

1. Set up Mandiant Redline to conduct a memory audit in preparation for the memory analysis exercise.
2. 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 install 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 **`win7-32-nromanoff-memory-raw.001`**image (it should be located in **`win7-32-nromanoff-10.3.58.5\win7-32-nromanoff-memory\`**).

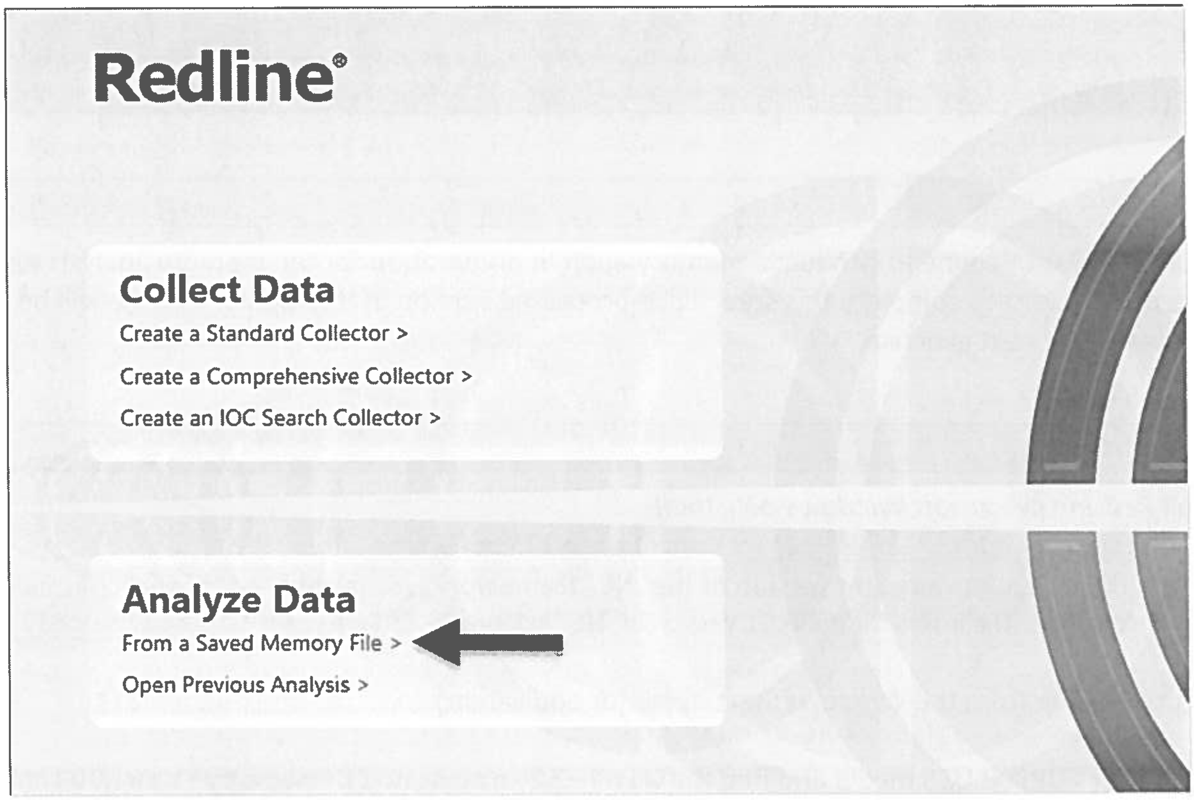
4. Copy the **`win7-32-nromanoff-memory-raw.001`** file to a local drive (i.e. **`C:\Users\<username>\Desktop`**).

Exercise – Step-by-Step Guide

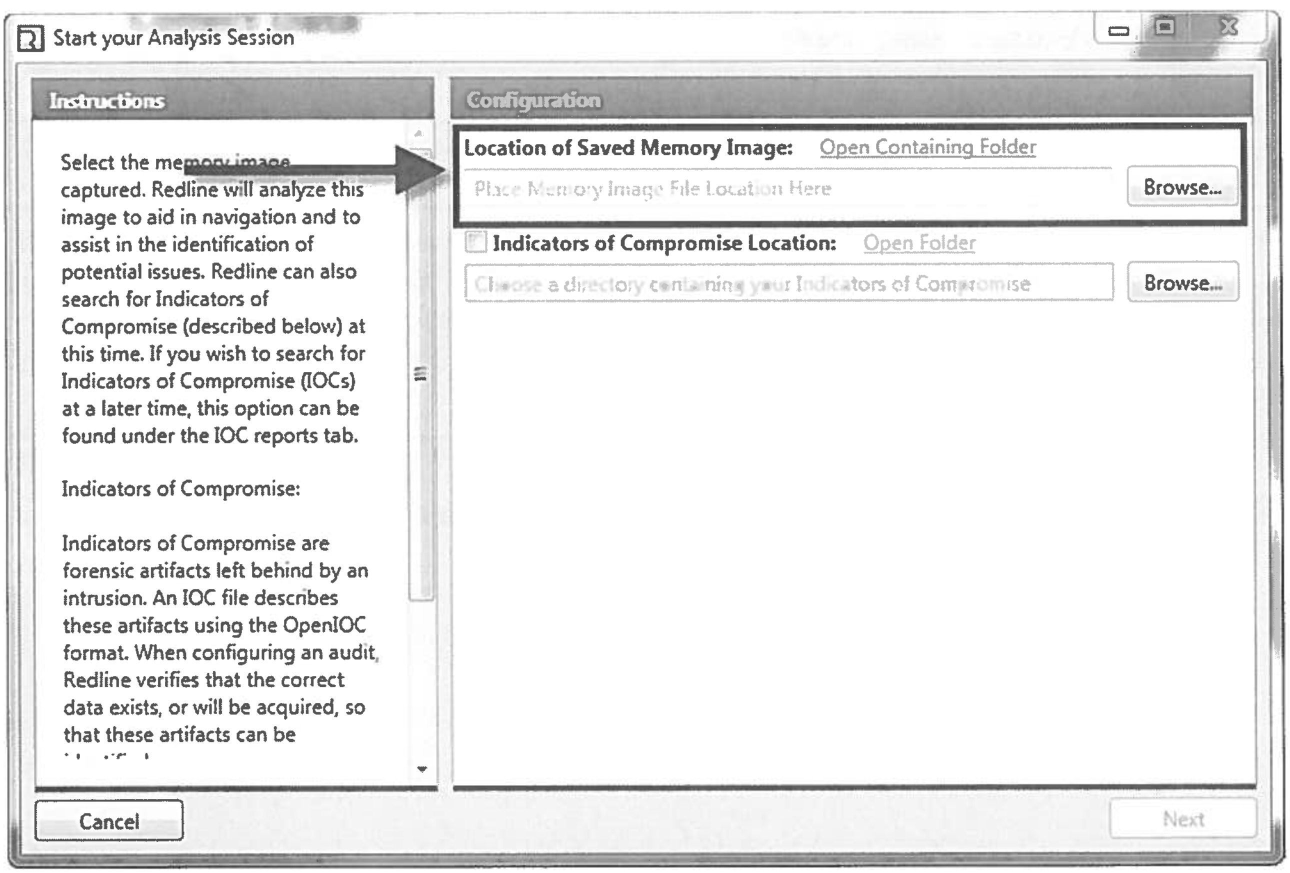
> **NOTE**: Some older laptops and VM setups may take a *very* long time to process this memory image (VMware Fusion is notoriously slow). If your system does not finish processing, **a pre-processed analysis file has been saved on your USB in the "`win7-32-nromanoff-10.3.58.5\win7-32-nromanoff-c-drive\precooked\redline`" folder.** Instructions for loading the pre-processed copy are present in the next exercise.

Perform a "Saved Memory File" analysis of the **`win7-32-nromanoff-memory-raw.001`** image.

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

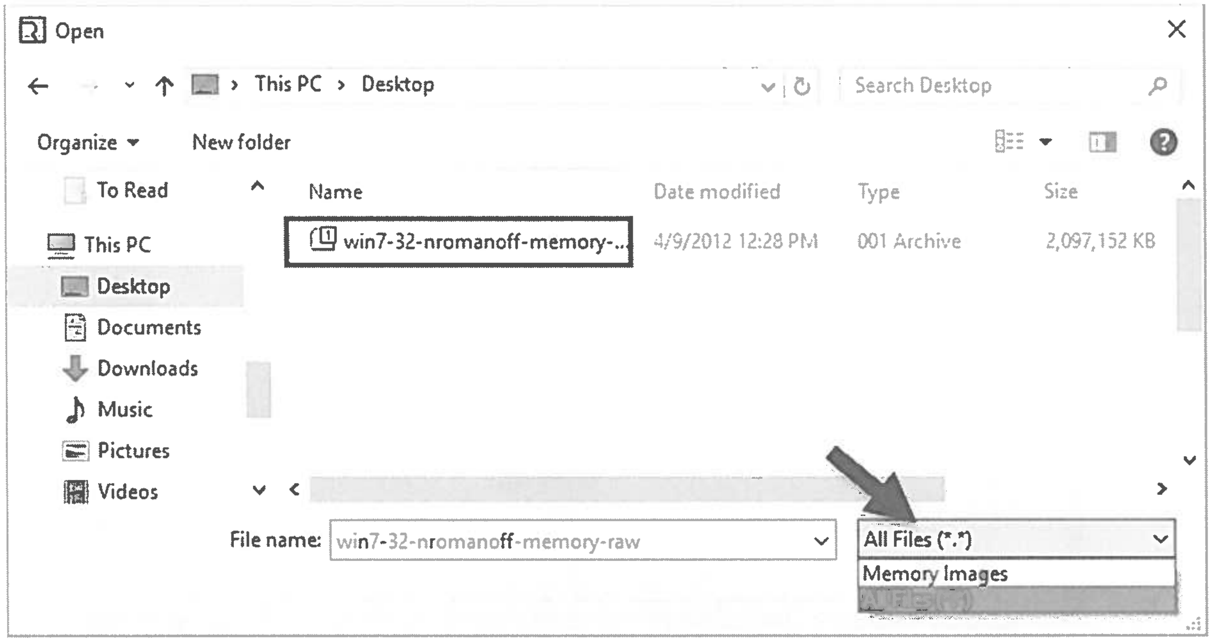


2. Find the "Location of Saved Memory Image" dialog.



3. Browse to the **`win7-32-nromanoff-memory-raw.001`** file.

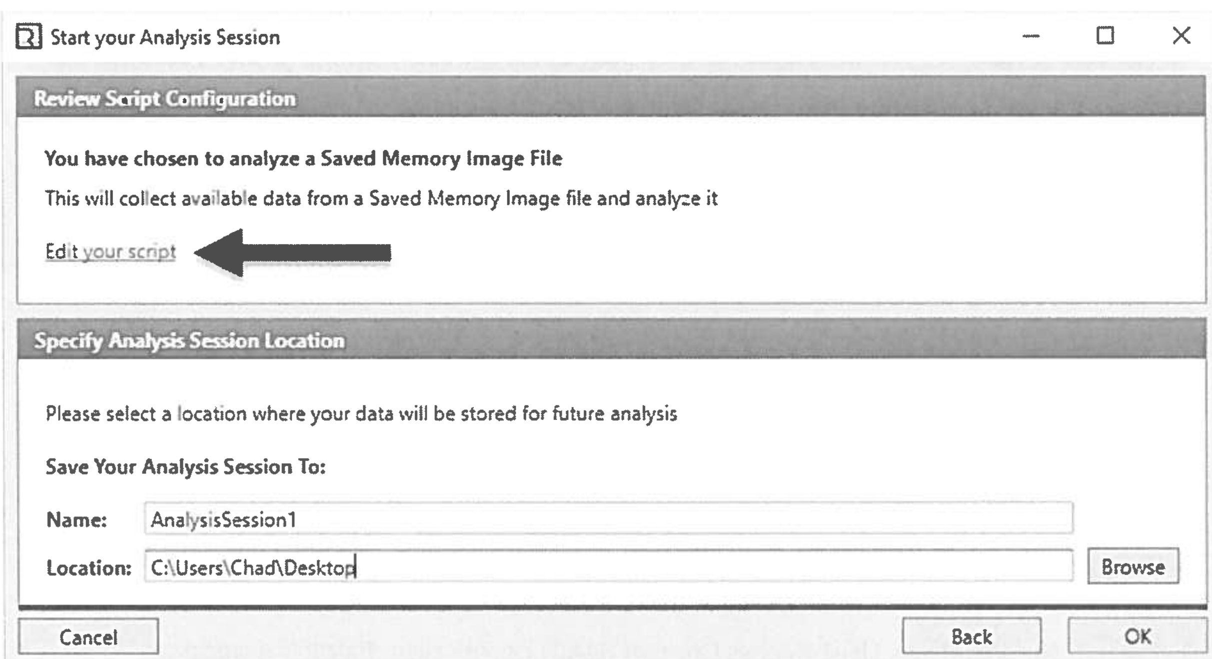
   a. You may need to select "All Files" on the drop-down to see the ".001" file extension.



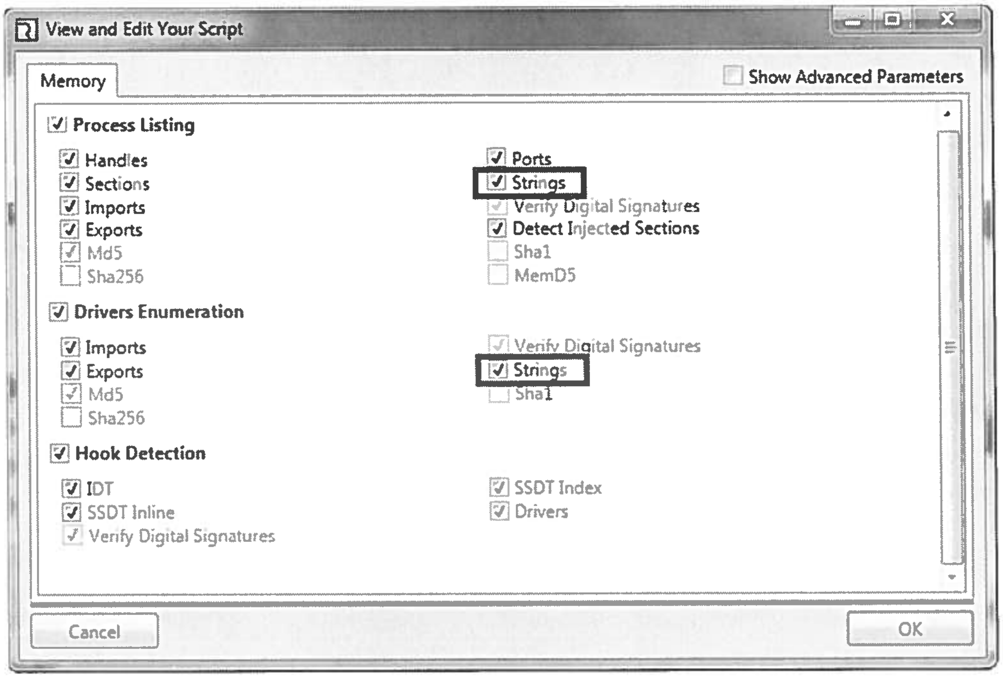
   b. Leave "Indicators of Compromise Location" unchecked.

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

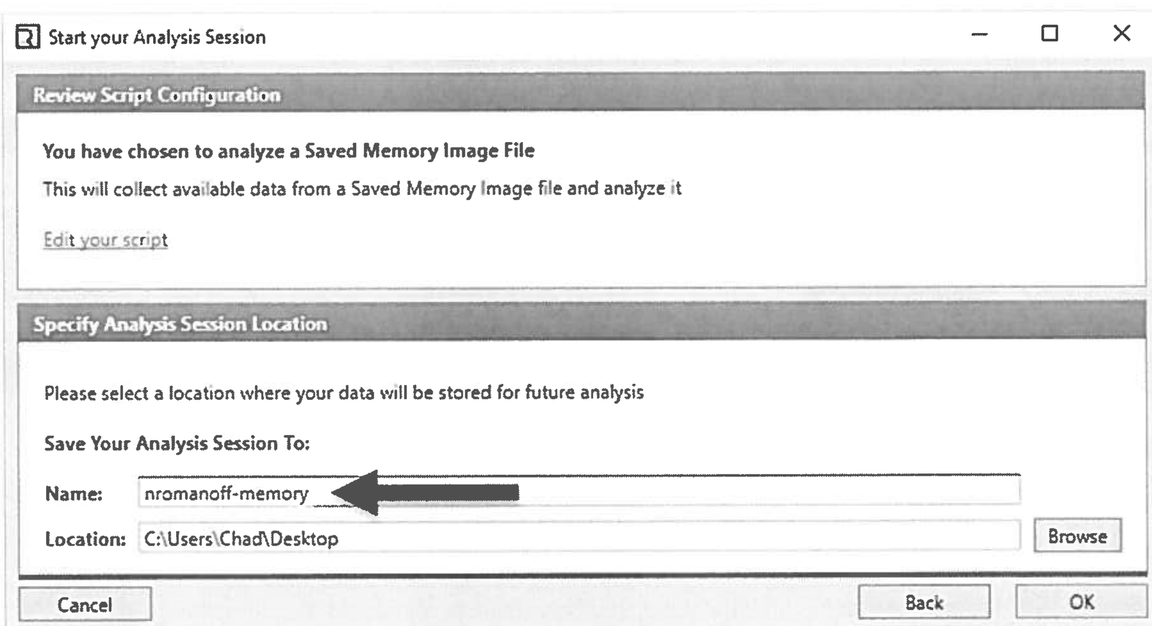
5. Select "Edit your script" to choose analysis options.



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 choose a name you will remember. The name **nromanoff-memory** works fine, and you can save it 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 data!

Exercise 2.2 – Redline Memory Analysis

Objectives

1. Learn memory forensics using the prescribed memory analysis steps.
2. Gain experience using Redline to analyze data present in memory structures.
3. Witness the value of string searching in Windows processes.
4. Understand the pros and cons of live analysis collections.

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 install for the FULL version of .NET is found in `\SIFT-Lab-Install\redline.`

2. Perform a full memory audit of the **`win7-32-nromanoff-memory-raw.001`** image, including process and driver strings output (see the "Optional Redline Pre-Process" exercise).

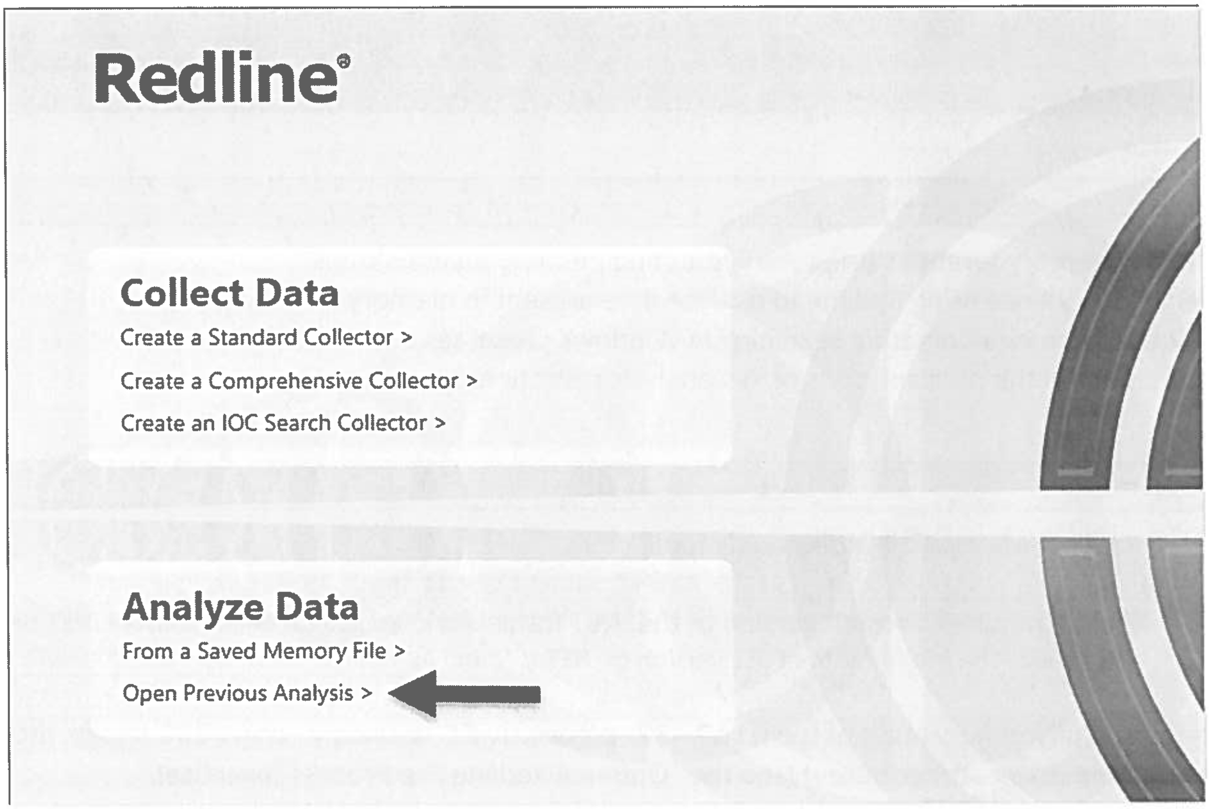
> **NOTE**: Some older laptops and VM setups may take a *very* long time to process this memory image. (VMware Fusion is notoriously slow.) If your system is still pre-processing at the start of this exercise, cancel processing. **A pre-processed analysis file has been saved on your USB in the "`win7-32-nromanoff-10.3.58.5\win7-32-nromanoff-c-drive\precooked\redline`" folder.**

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

   - Copy the "**`win7-32-nromanoff-10.3.58.5\win7-32-nromanoff-c-drive\precooked\redline`**" folder from the course USB to a local drive. (Your desktop is one good option.)

   - Double-click the **`nromanoff.mans`** file located in the folder you just copied.

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



5. Mount your disk image so that you can interact with files and folders on the win7-32-nromanoff-c-drive.

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 to the **/cases/win7-32-nromanoff-c-drive** directory. Then, uncompress and mount your evidence files so that you can see the mounted filesystem in /mnt/windows_mount.

```
$ sudo su -

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

# 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
```

Exercise – Questions

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

     ____

   - Why did the MRI score this process so high?

     ____

     ____

   - What is the parent process?

     ____

   - What user account (SID) was used to spawn the process?

     ____

   - List the process binary path.

     ____

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

       What additional file is located in the same suspicious directory?

       ____

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

       ____

   - View the hierarchical process list. Do you see any additional suspicious processes? (Ignore the two junk processes with unusually large PID values -- they are false positives returned during Redline's executive process block carving.)

     ____

     ____

2. **Analyze Process DLLs and Handles.**
    - Review the "Handles" tab of each suspicious process identified in Question 1. Sort by the "Occurrence" column to narrow your search using least frequency of occurrence (LFO) and ignore any unnamed handles (no information present in Handle Name column).
        - Review handles in PID 6404 and document the pointer to the interesting registry key found in the last question.

          ____

        - Focus on interesting file handles in PID 2956.

          ____

          ____

          ____

    - List the loaded DLLs for each suspicious process by using the "Section" tab. Sort by the "Count" column to narrow your search using least frequency of occurrence (LFO) and ignore any entries with an empty "Section Name".
        - What DLL loaded by PID 1328 might make a good indicator?

          ____

3. **Review network artifacts (select "Ports" in the Analysis Data pane).**
    - Do any network artifacts appear suspicious? What process spawned them?

      ____

      ____

      ____

4. **Use the Redline "Strings" output to look for additional evidence.**
    - Search svchost.exe (PID 6404). Be creative. Terms such as "`http://`," "`https://`," "`ftp://`," "`.exe`," "`C:\`," and names of suspected malware or support files can sometimes yield fruit.

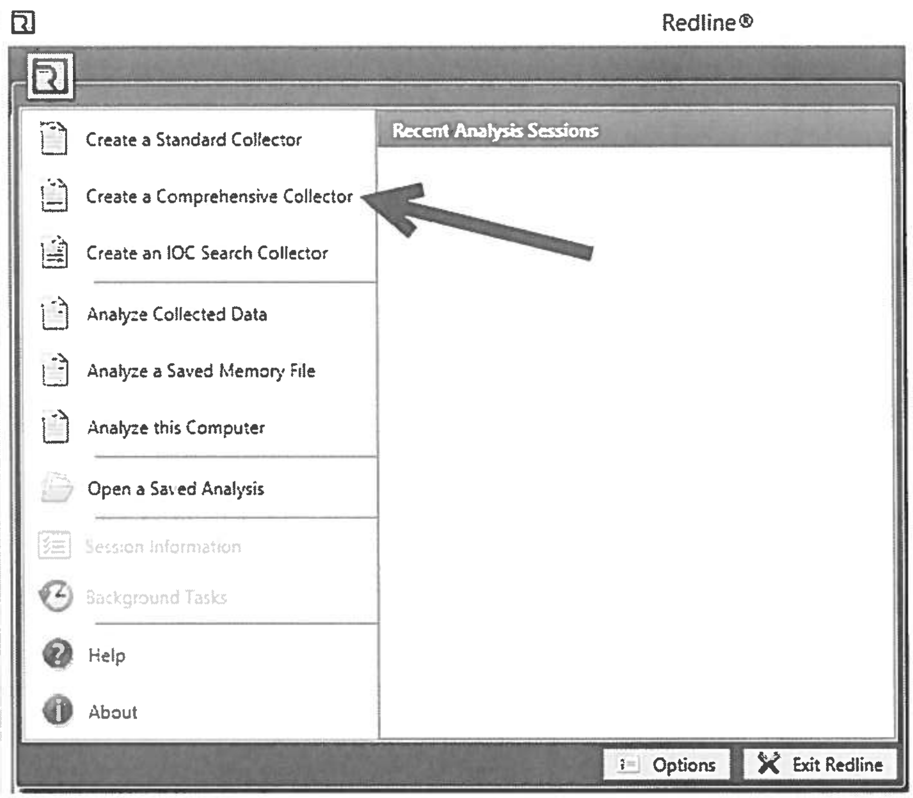
      ____

      ____

      ____

5. **Optional homework: Create and run your own live analysis via a Redline Collector (<u>NOTE: Redline does not currently support Windows 10 collection and analysis</u>).**

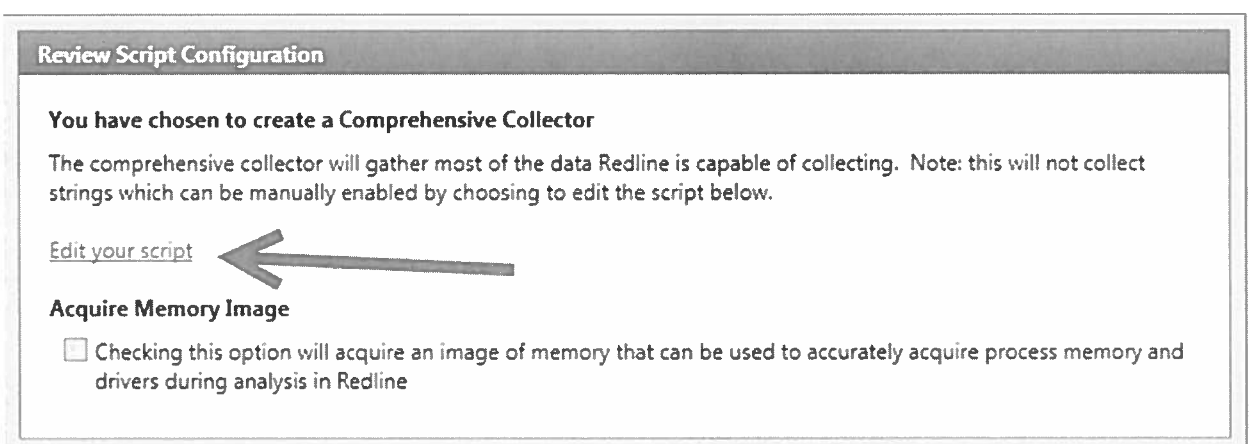
a. Click the Top Left Icon *"Create a Comprehensive Collector"*.



b. Edit your script.

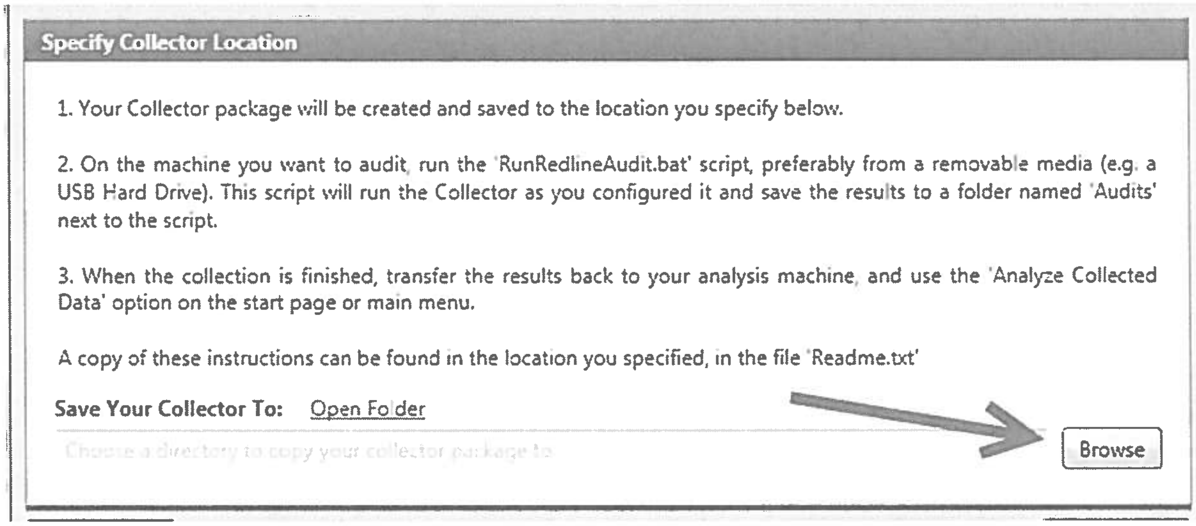
i. Go through the various menus and select what you would like the script to collect. Because we selected "Comprehensive Collector," most options are already enabled. Any additional selections will likely require a long time to complete.

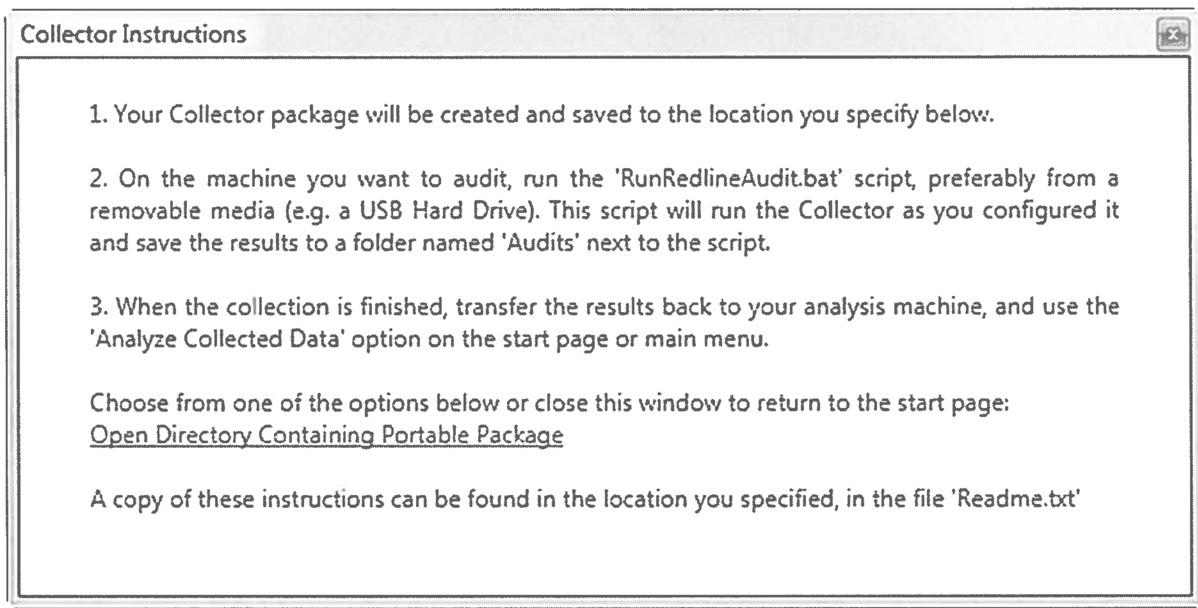
ii. If you would also like the Collector to take a full memory image, select "Acquire Memory Image."



c. Specify Collector Location.

i. Place this on a USB thumb drive if you have one, or, for testing purposes you can save and run the Collector from a local system drive.





d. 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.

```
Administrator: Command Prompt

D:\Temp\RedlineCollector>dir
 Volume in drive D is Data
 Volume Serial Number is 9250-D7B6

 Directory of D:\Temp\RedlineCollector

09/18/2014  08:29 AM    <DIR>          .
09/18/2014  08:29 AM    <DIR>          ..
06/09/2014  03:25 PM             1,108 elevate.cmd
06/09/2014  03:25 PM             4,023 elevate.vbs
06/09/2014  03:24 PM             1,770 finishAnalysis.js
06/09/2014  03:24 PM               839 getNextSessionFolder.js
06/09/2014  02:00 PM             2,179 Helper.bat
09/18/2014  08:29 AM             5,208 MemoryzeAuditScript.xml
06/09/2014  03:25 PM               550 Readme.txt
06/09/2014  03:26 PM               832 RunRedlineAudit.bat
09/18/2014  08:29 AM    <DIR>          x64
09/18/2014  08:29 AM    <DIR>          x86
               8 File(s)         16,509 bytes
               4 Dir(s)  155,205,468,160 bytes free

D:\Temp\RedlineCollector>RunRedlineAudit.bat
```

```
C:\Windows\System32\cmd.exe

Executing command w32drivers-modulelist, 1.4.46.0
Pre-execution diagnostics for command w32drivers-modulelist
PageFaultCount: 27042 PeakWorkingSetSize: 29302784 WorkingSetSize: 15917056 Quot
aPeakPagedPoolUsage: 243848 QuotaPagedPoolUsage: 221344 QuotaPeakNonPagedPoolUsa
ge: 20240 QuotaNonPagedPoolUsage: 16832 PagefileUsage: 7680000 PeakPagefileUsage
: 9678848
CommitTotal: 2267418 CommitLimit: 4814297 CommitPeak: 2892849 PhysicalTotal: 419
1705 SystemCache: 265833 KernelTotal: 96609 KernelPaged: 63363 KernelNonpaged: 3
3246 PageSize: 4096 HandleCount: 46957 ProcessCount: 115 ThreadCount: 1420
<Issue number="0" level="Info" summary="System range 0x0000000000000000 - 0x0000
000000001000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x0000000100000000 - 0x0000
000430000000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x0000000000103000 - 0x0000
0000cf65f000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x0000000000100000 - 0x0000
000000102000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x000000000001f000 - 0x0000
00000009a000" context="EnumerateDevices"/>

<Issue number="17003" level="Info" summary="Internal InformationAlgorithm found
```

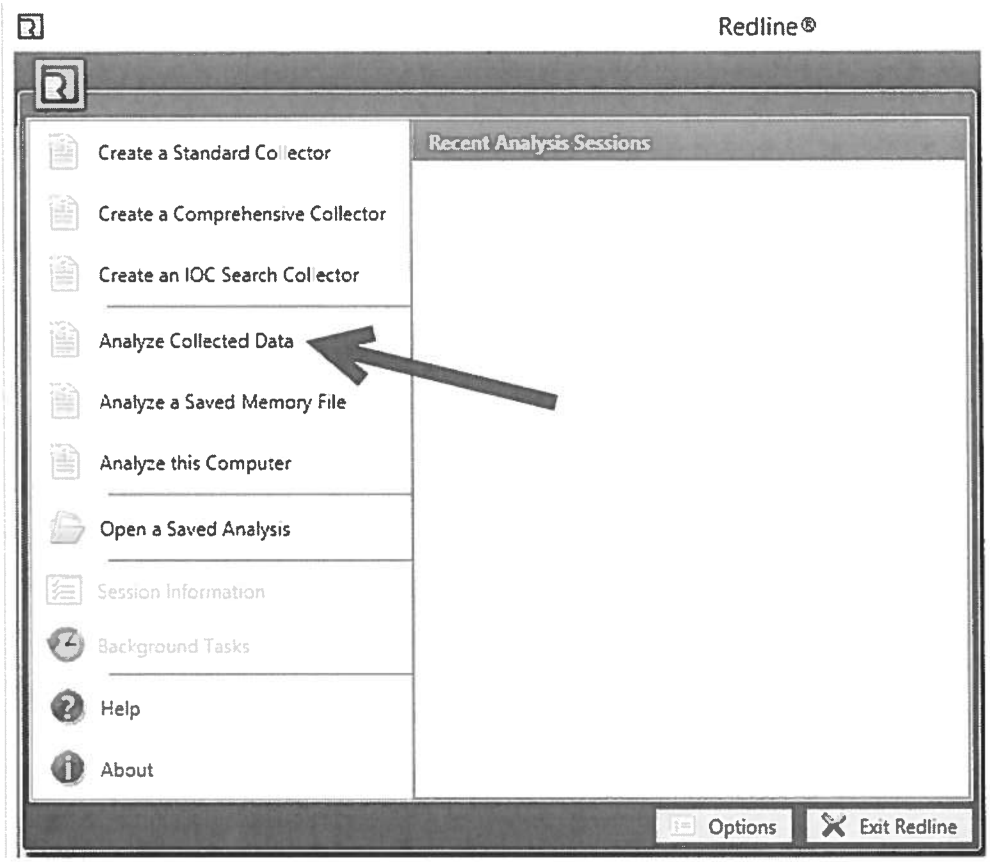
i. A full collection script can take hours to run. (In the future, you can remove unnecessary options in the audit script to speed up processing.) Your results will be located in a folder named according to the System Hostname in the same location as your Collector.

(D:) ▸ Temp ▸ RedlineCollector ▸ Sessions ▸ AnalysisSession1 ▸ Audits ▸ M4500 ▸ 20140918143212

Name	Date modified	Type	Size
w32drivers-modulelist.gmrE87wpBGdbV...	9/18/2014 8:37 AM	GMRE87WPBGDB...	80 KB
w32drivers-modulelist.issues.fGef3Dzm7...	9/18/2014 8:37 AM	FGEF3DZM7KBDX...	2 KB
w32drivers-signature.hjlt2KUmjtYdZHbw...	9/18/2014 8:35 AM	HJIT2KUMJTYDZH...	1,033 KB
w32drivers-signature.issues.lP7KxpEMoo...	9/18/2014 8:35 AM	LP7KXPEMOOSCY...	42 KB
w32kernel-hookdetection.ea4oTX9SCR9d...	9/18/2014 8:39 AM	EA4OTX9SCR9DRI...	1 KB
w32kernel-hookdetection.issues.kYwkVu...	9/18/2014 8:39 AM	KYWKVUXMJQED...	2 KB
w32processes-memory.issues.kPjBmfiGk...	9/18/2014 8:39 AM	KPJBMFIGKZSFLG...	0 KB

e. Open your results in Redline and start your analysis.

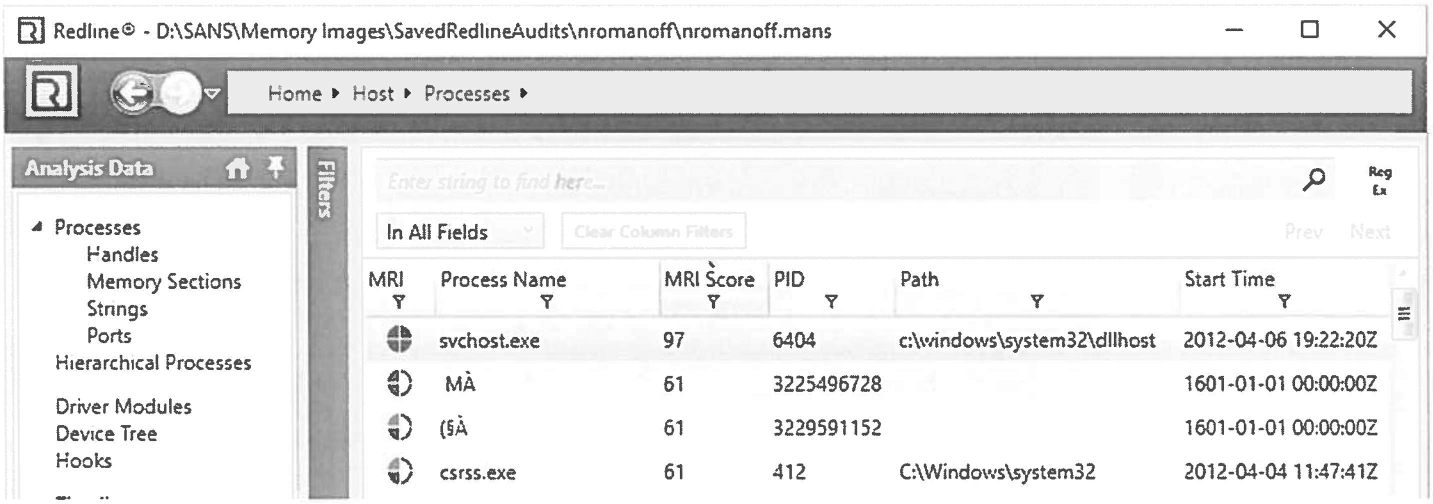
i. <Top Left Icon> → Analyze Collected Data → Audit Location → <subfolder of folder named according to System Hostname>. (Or just double-click the `.mans` file in the same folder.)



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 6404)



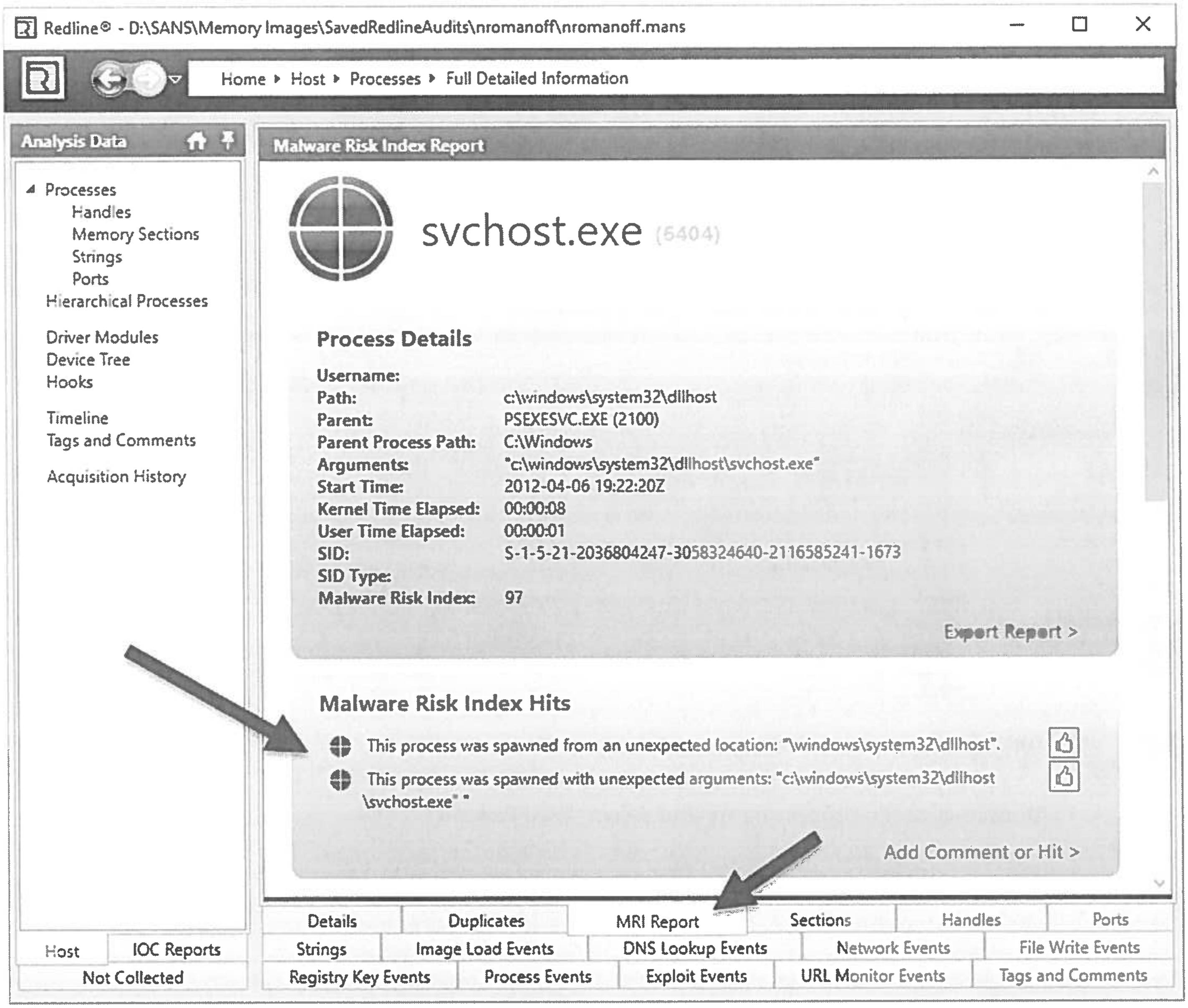
- Why did the MRI score this process so high?

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

  Answer: Unexpected location and unexpected arguments

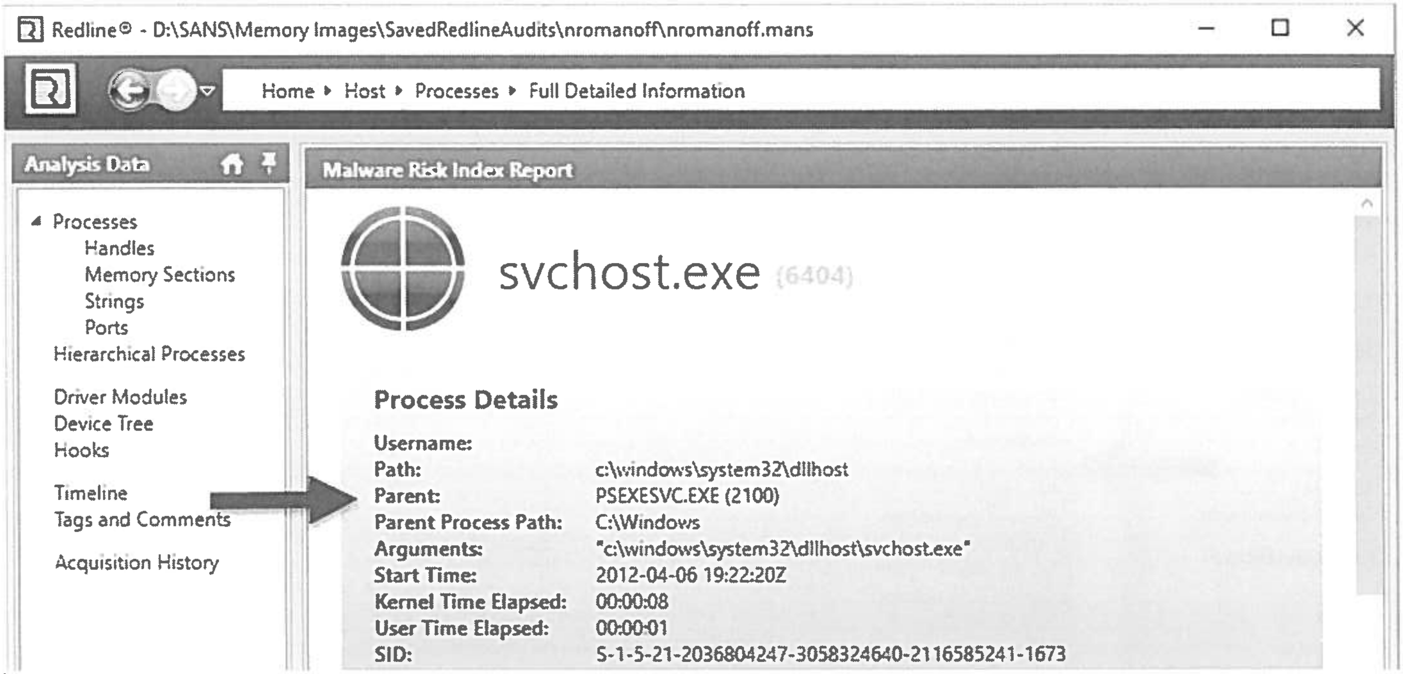
missing -k

search it.



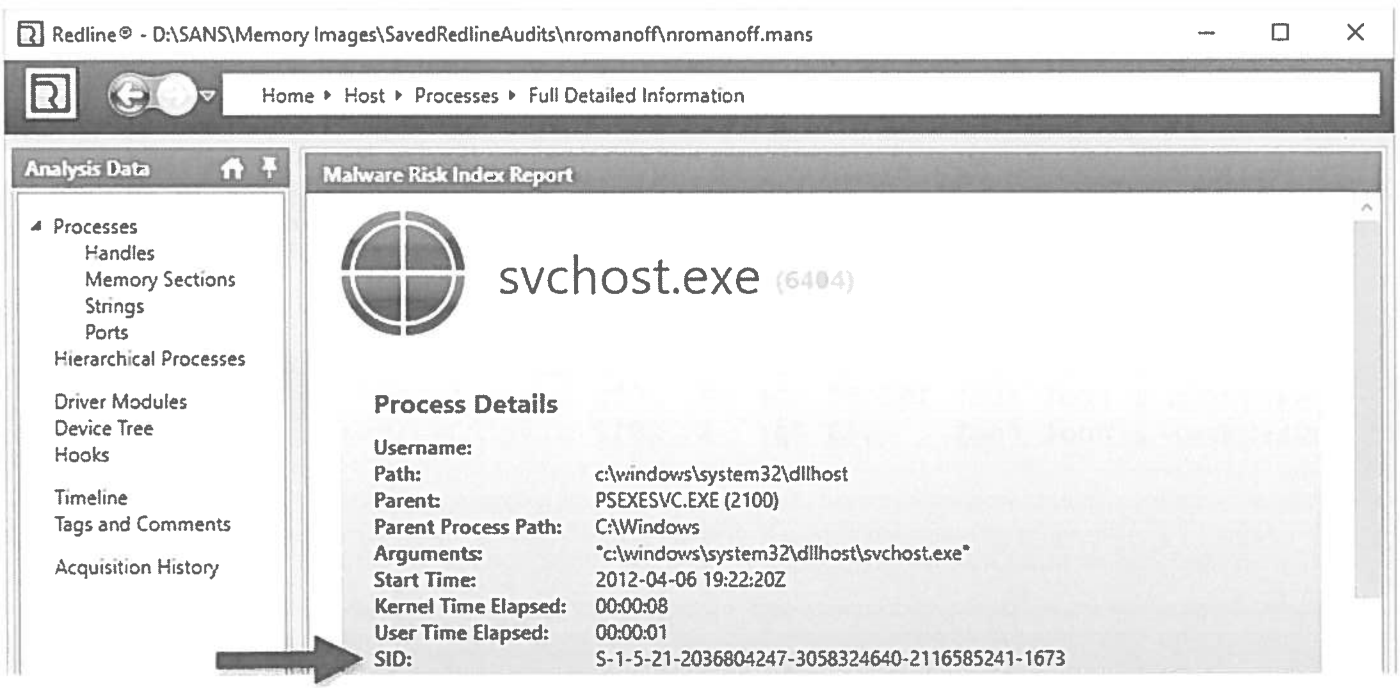
- What is the parent process?

PSEXESVC.EXE (PID 2100) -- aka "SysInternals PsExec"



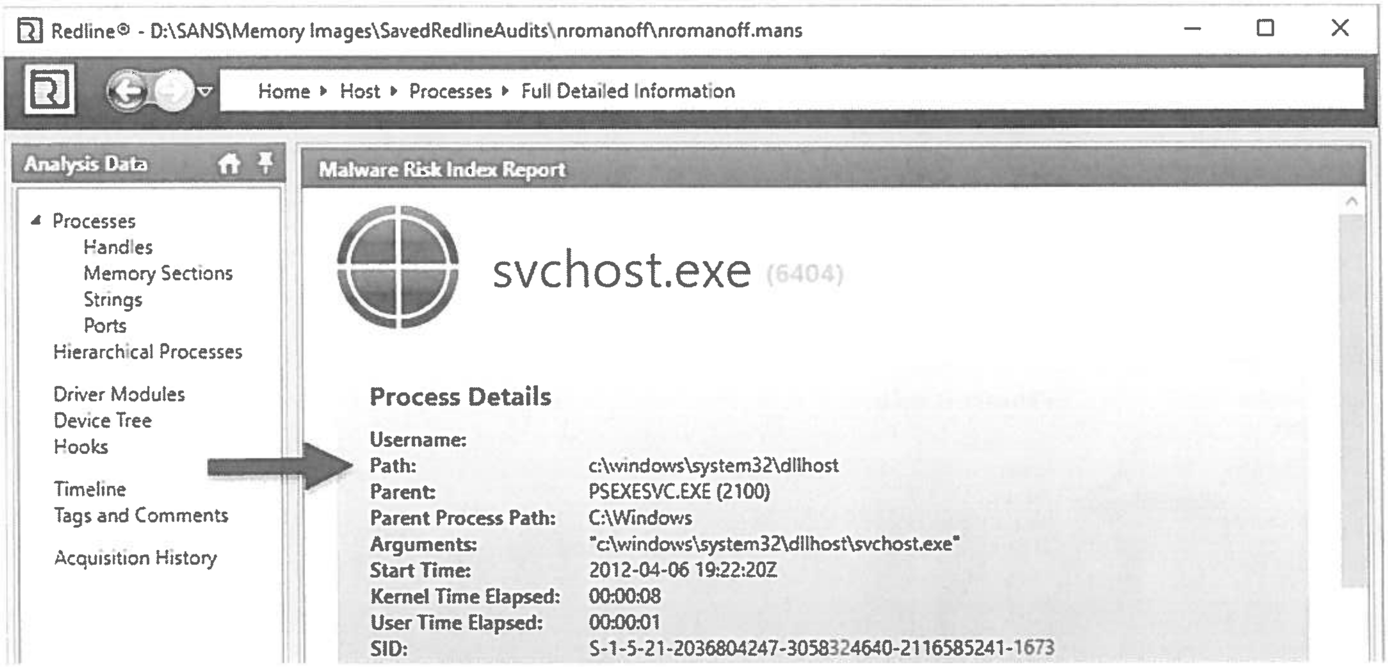
- What user account (SID) was used to spawn the process?

  <u>S-1-5-21-2036804247-3058324640-211658241-1673</u>



- List the process binary path.

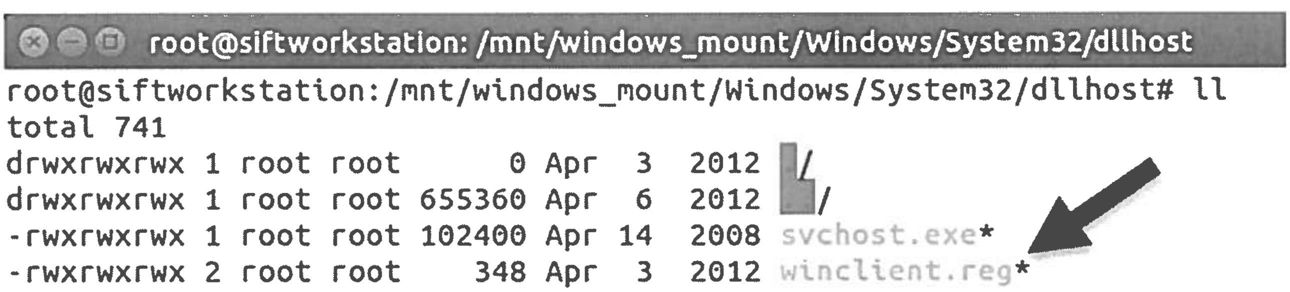
  <u>c:\windows\system32\dllhost</u>

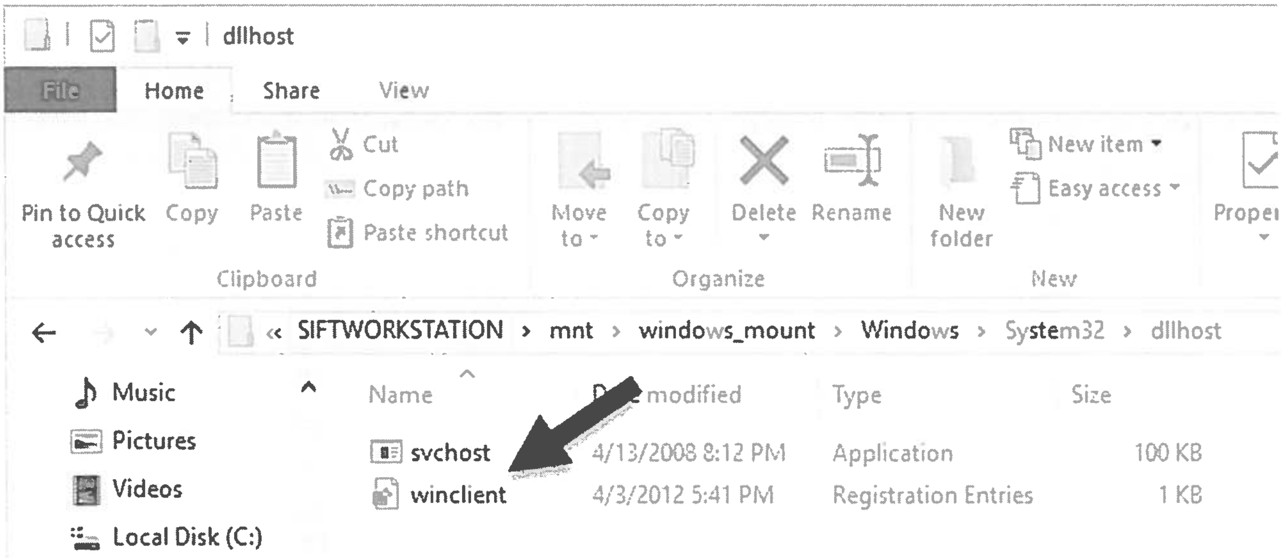


- Review the contents of the folder where this binary is located within the nromanoff disk image (during the Exercise Preparation section you should have mounted nromanoff's disk image to /mnt/windows_mount).

  What additional file is located in the same suspicious directory?

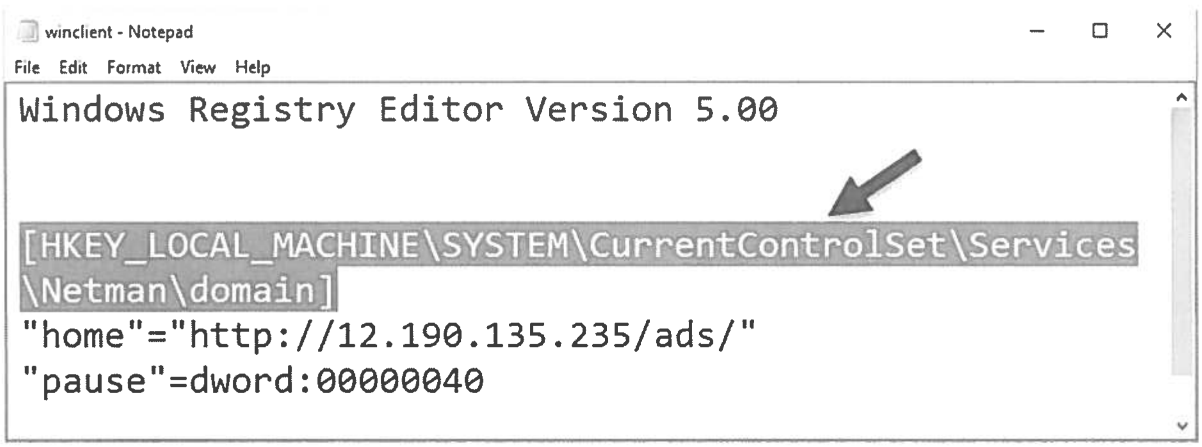
  winclient.reg





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

HKLM\SYSTEM\CurrentControlSet\Services\Netman\domain

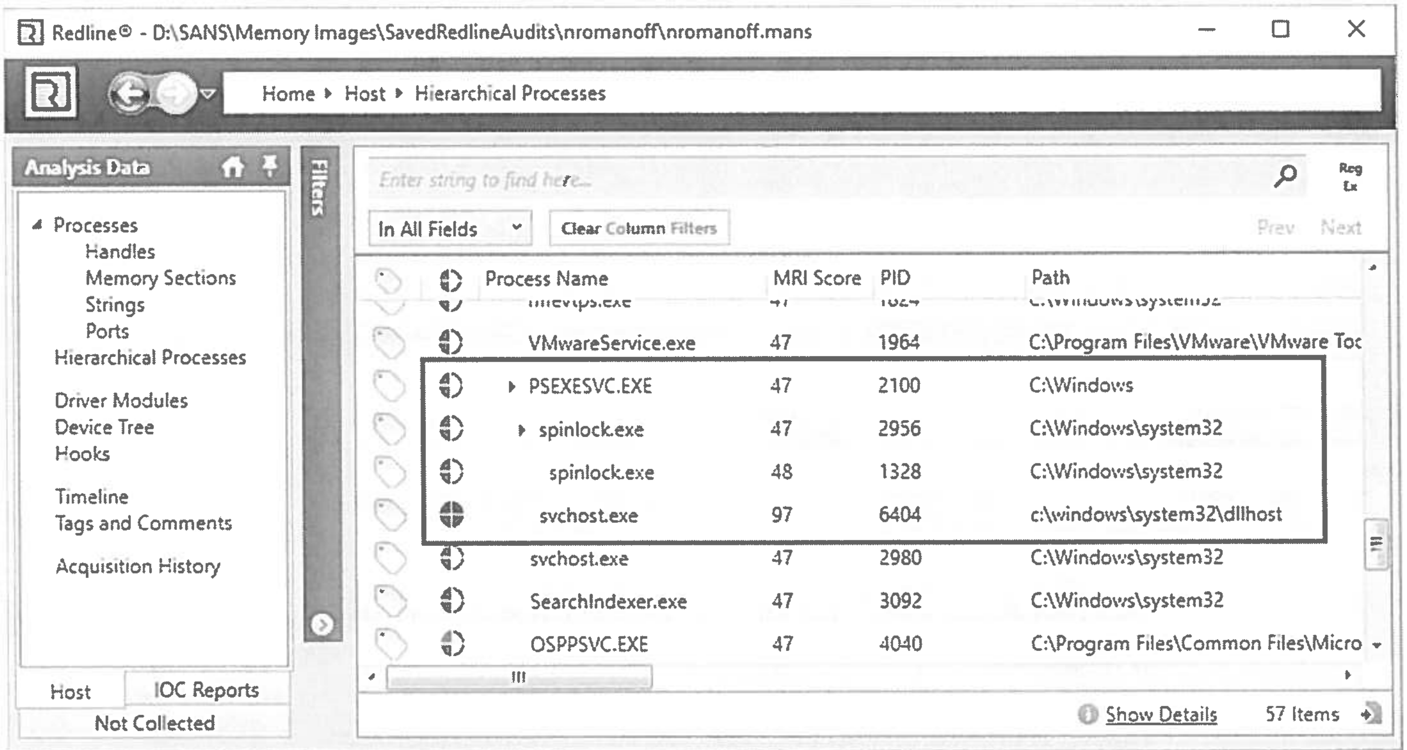


If we believe svchost.exe is suspicious, the close proximity of winclient.reg also makes it suspicious. Although we do not yet know the significance of the registry key and IP address information stored within winclient.reg, they may later become useful indicators of compromise.

- View the hierarchical process list. Do you see any additional suspicious processes? (Ignore the two junk processes with unusually large PID values -- they are false positives returned during Redline's executive process block carving.)

  spinlock.exe (PID 2956)
  spinlock.exe (PID 1328)

  Since both spinlock processes have the same PsExec parent as the suspicious svchost.exe process, it would be prudent to look deeper into those processes. Open source research also shows that spinlock.exe is not a common process name on Windows systems.

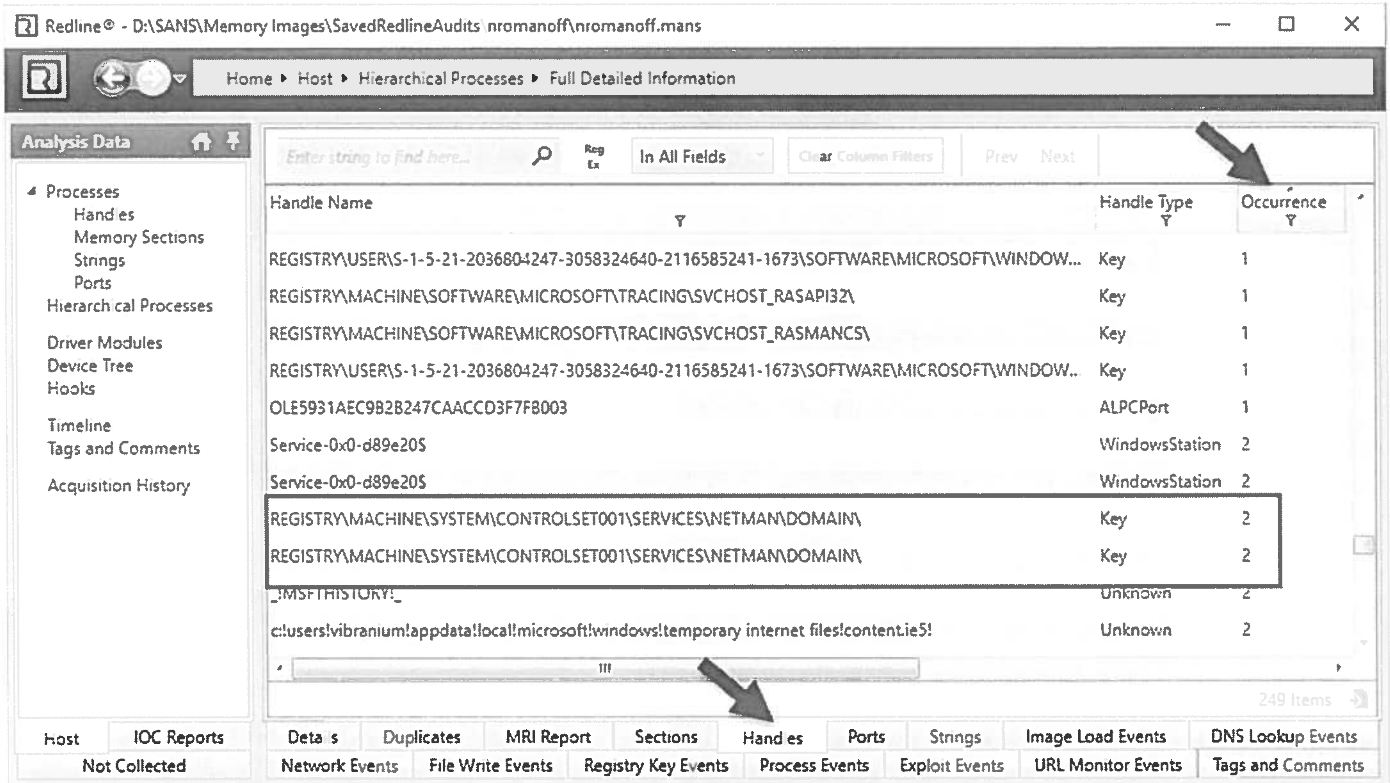


2. **Analyze Process DLLs and Handles.**
   - Review the "Handles" tab of each suspicious process identified in Question 1. Sort by the "Occurrence" column to narrow your search using least frequency of occurrence (LFO) and ignore any unnamed handles (no information present in Handle Name column).

     - Review handles in PID 6404 and document the pointer to the interesting registry key found in the last question.

       <u>svchost.exe (PID 6404):</u>

       <u>The key REGISTRY\MACHINE\SYSTEM\CONTROLSET001\SERVICES\NETMAN\DOMAIN has a low occurrence count. Recall that this is the same key we identified in step one, referenced within the winclient.reg file.</u>

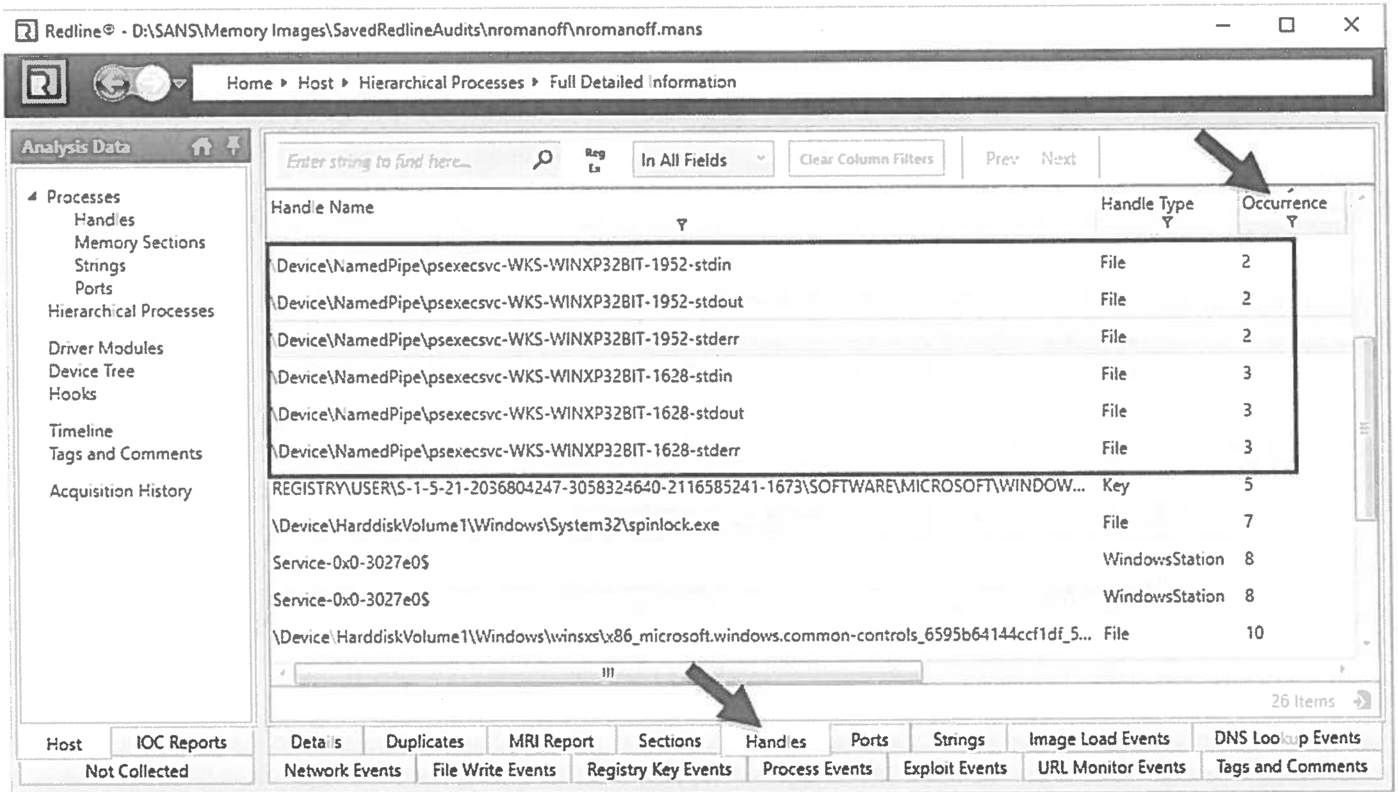


- Focus on interesting file handles in PID 2956.

spinlock.exe (PID 2956):

\Device\NamedPipe\psexecsvc-WKS-WINXP32BIT-1952-stdin
\Device\NamedPipe\psexecsvc-WKS-WINXP32BIT-1952-stdout
\Device\NamedPipe\psexecsvc-WKS-WINXP32BIT-1952-stderr
\Device\NamedPipe\psexecsvc-WKS-WINXP32BIT-1628-stdin
\Device\NamedPipe\psexecsvc-WKS-WINXP32BIT-1628-stdout
\Device\NamedPipe\psexecsvc-WKS-WINXP32BIT-1628-stderr

Named pipes are a built-in way for processes to communicate with one another. They can be used to communicate between processes on the same computer, or even between two computers on a network. In this case, it looks like this pipe is named for the PsExec service and also contains the hostname of a remote computer that initiated the pipe (WKS-WINXP32BIT).



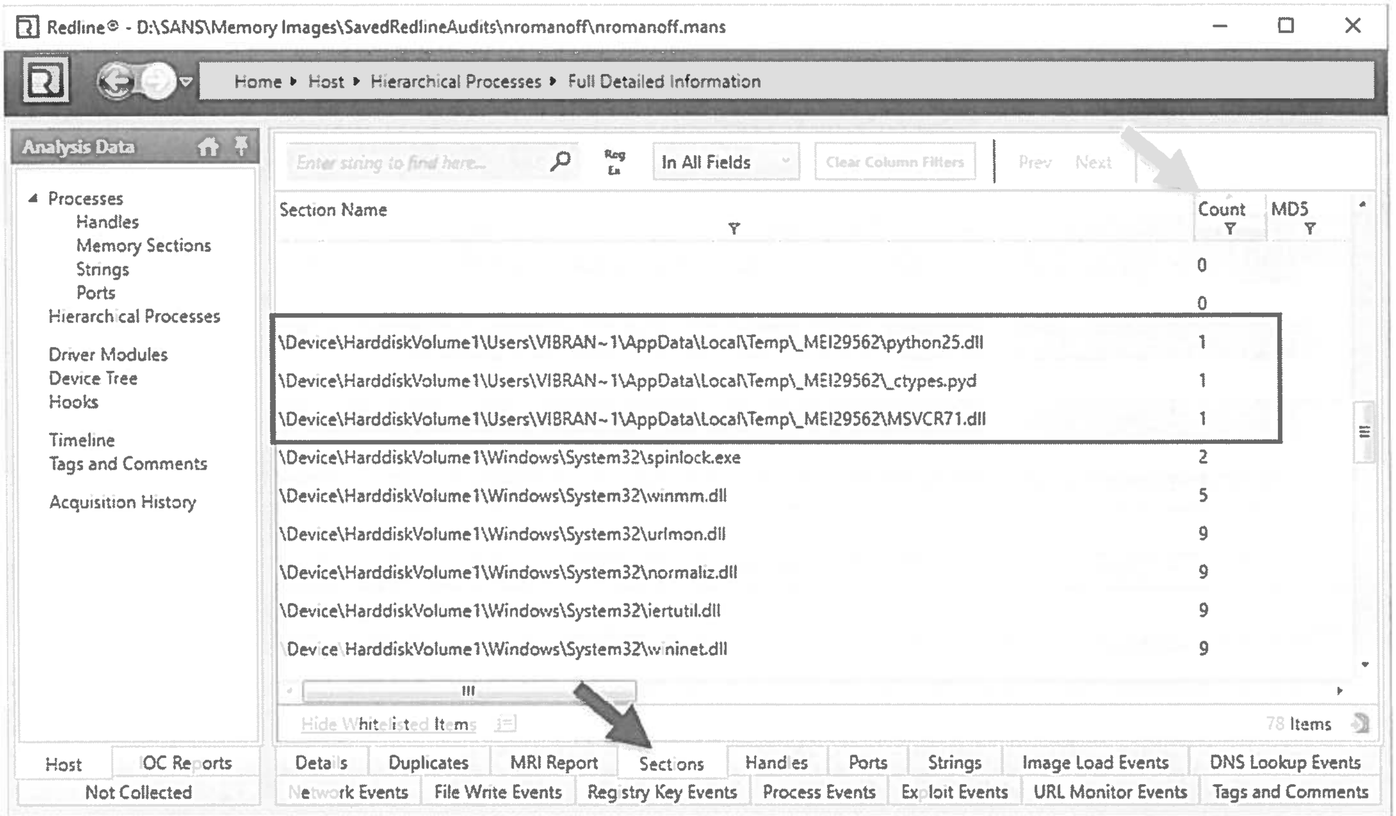
- List the loaded DLLs for each suspicious process by using the “Section” tab. Sort by the “Count” column to narrow your search using least frequency of occurrence (LFO) and ignore any entries with an empty “Section Name”.

  - What DLL loaded by PID 1328 might make a good indicator?

spinlock.exe (PID 1328):

\Device\HarddiskVolume1\Users\VIBRAN~1\AppData\Local\Temp_MEI29562\python25.dll

All three files loaded from the “AppData\Local\Temp” folder are interesting, and likely indicate some sort of Python code being executed by this process. Since Python DLLs are somewhat uncommon on standard user workstations, the presence of this DLL might help identify variants of the spinlock malware. The MSVCR71.dll file is related, but more commonly found on workstations (though not necessarily in the “%USERPROFILE%\AppData\Local\Temp” folder).



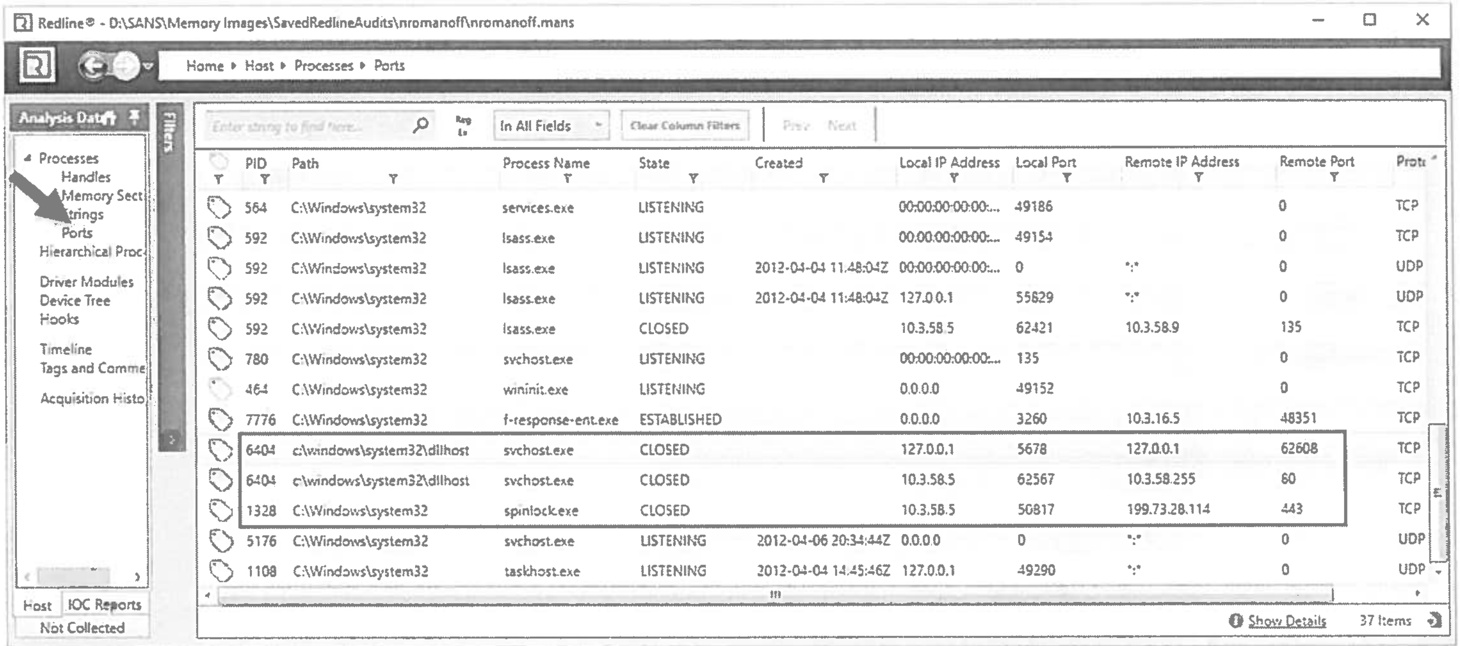
3. **Review network artifacts (select "Ports" in the Analysis Data pane).**
   - Do any network artifacts appear suspicious? What process spawned them?

     svchost.exe (PID 6404): 127.0.0.1:5678 to 127.0.0.1:62608 (localhost)

     svchost.exe (PID 6404): 10.3.58.5:62567 to 10.3.58.255:80

     spinlock.exe (PID 1328): 10.3.58.5:50817 to 199.73.28.114:443

     The svchost.exe network activity is certainly strange, but difficult to understand without reverse engineering the executable. The spinlock.exe connection looks like a classic beacon: outbound connection to either port 80 or 443 from a process that should not be communicating via HTTP or HTTPS.



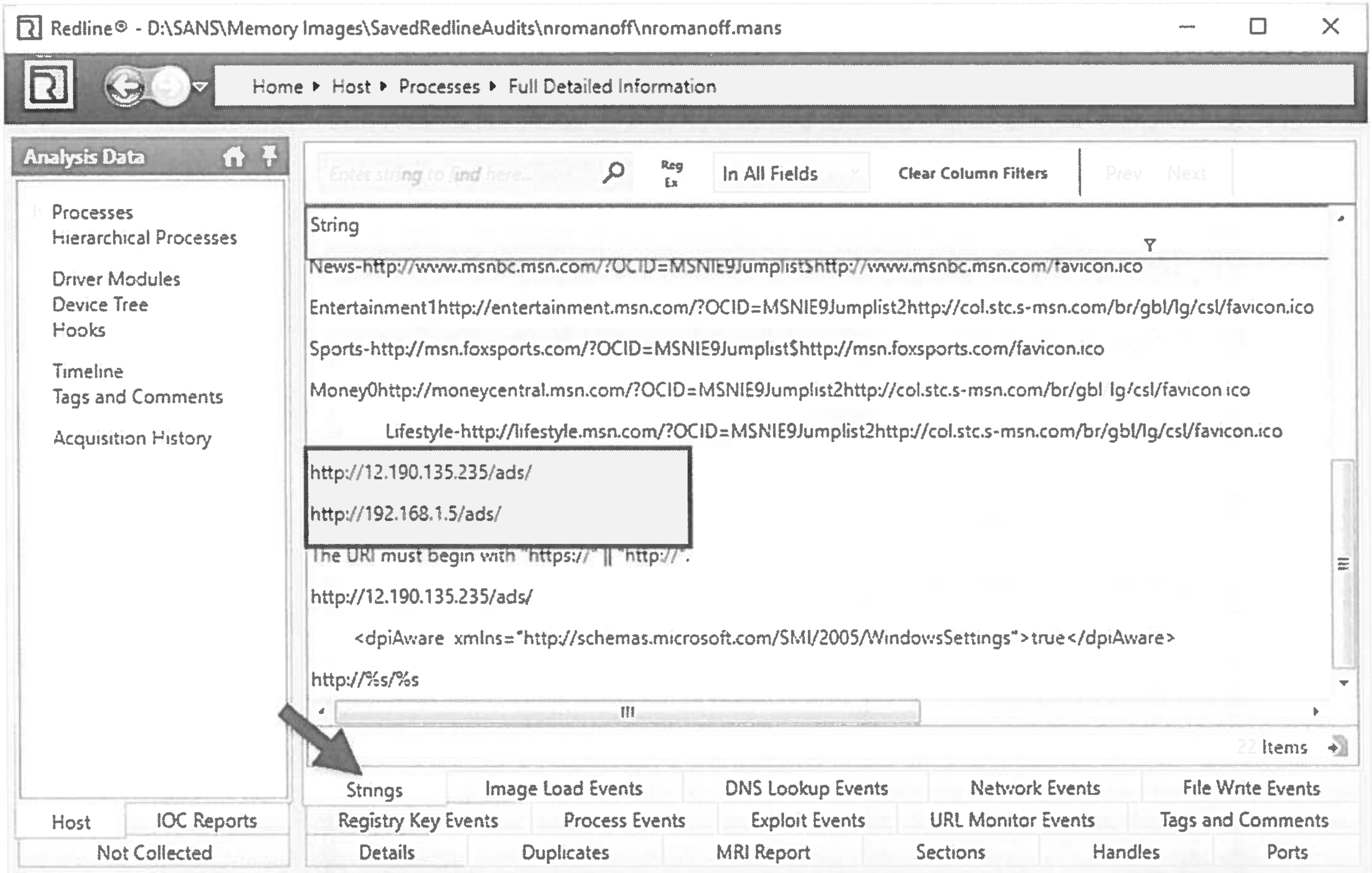
PID	Path	Process Name	State	Created	Local IP Address	Local Port	Remote IP Address	Remote Port	Prot
564	C:\Windows\system32	services.exe	LISTENING		00:00:00:00:00:...	49186		0	TCP
592	C:\Windows\system32	lsass.exe	LISTENING		00:00:00:00:00:...	49154		0	TCP
592	C:\Windows\system32	lsass.exe	LISTENING	2012-04-04 11:48:04Z	00:00:00:00:00:...	0	*:*	0	UDP
592	C:\Windows\system32	lsass.exe	LISTENING	2012-04-04 11:48:04Z	127.0.0.1	55629	*:*	0	UDP
592	C:\Windows\system32	lsass.exe	CLOSED		10.3.58.5	62421	10.3.58.9	135	TCP
780	C:\Windows\system32	svchost.exe	LISTENING		00:00:00:00:00:...	135		0	TCP
464	C:\Windows\system32	wininit.exe	LISTENING		0.0.0.0	49152		0	TCP
7776	C:\Windows\system32	f-response-ent.exe	ESTABLISHED		0.0.0.0	3260	10.3.16.5	48351	TCP
6404	c:\windows\system32\dllhost	svchost.exe	CLOSED		127.0.0.1	5678	127.0.0.1	62608	TCP
6404	c:\windows\system32\dllhost	svchost.exe	CLOSED		10.3.58.5	62567	10.3.58.255	80	TCP
1328	C:\Windows\system32	spinlock.exe	CLOSED		10.3.58.5	50817	199.73.28.114	443	TCP
5176	C:\Windows\system32	svchost.exe	LISTENING	2012-04-06 20:34:44Z	0.0.0.0	0	*:*	0	UDP
1108	C:\Windows\system32	taskhost.exe	LISTENING	2012-04-04 14:45:46Z	127.0.0.1	49290	*:*	0	UDP

4. **Use the Redline "Strings" output to look for additional evidence.**
   - Search svchost.exe (PID 6404). Be creative. Terms such as "`http://`," "`https://`," "`ftp://`," "`.exe`," "`C:\`," and names of suspected malware or support files can sometimes yield fruit.

     <u>Some interesting URIs can be found in the strings for the svchost.exe process:</u>

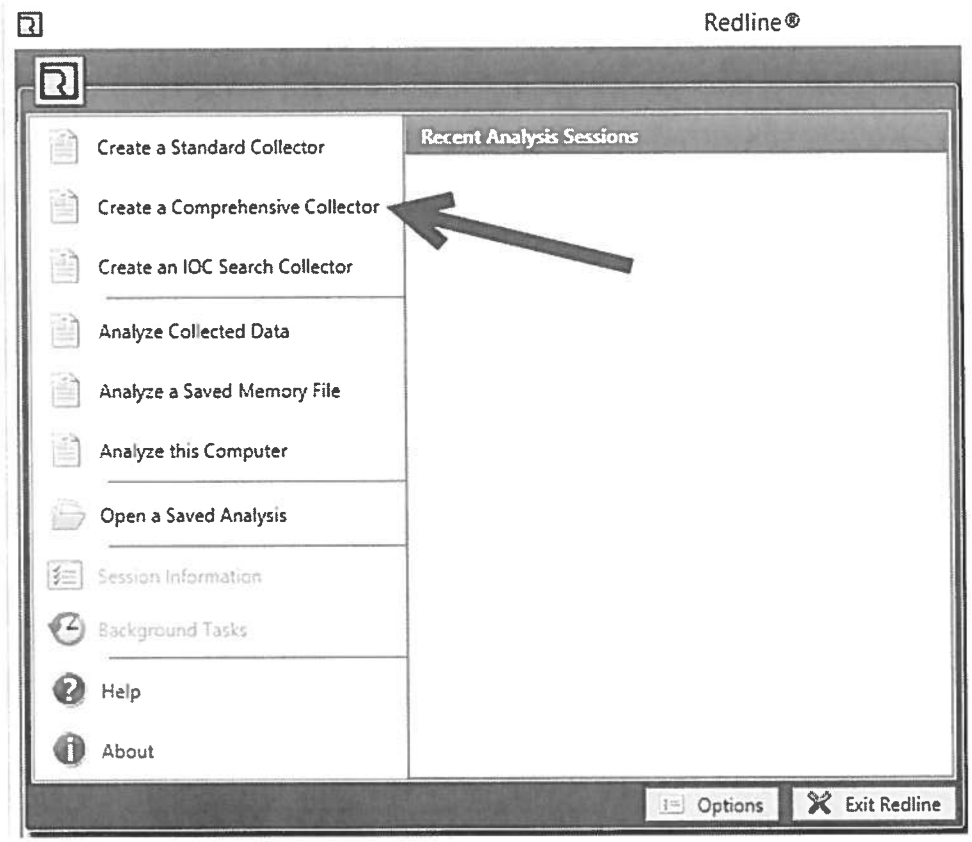
     <u>http://12.190.135.235/ads</u>

     <u>http://192.168.1.5/ads</u>

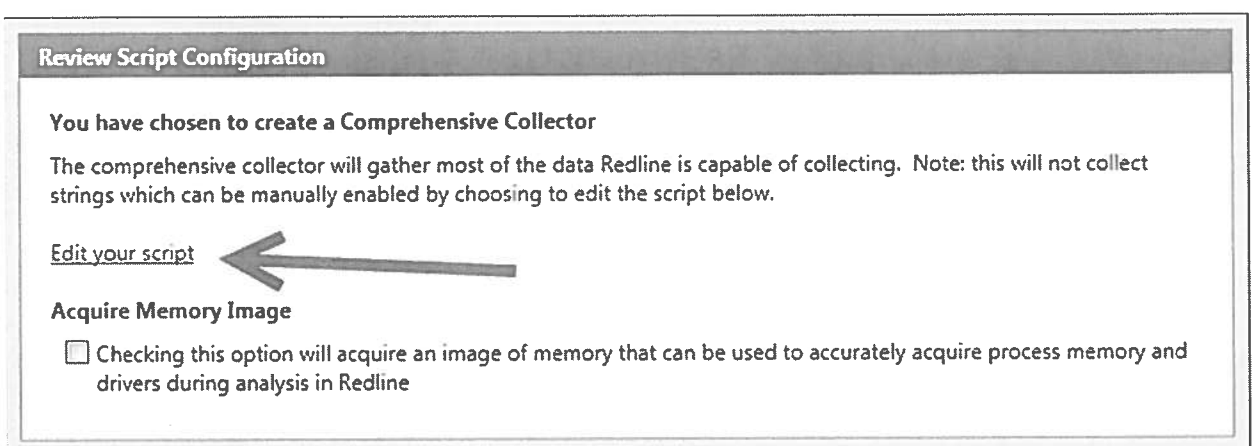

Redline® - D:\SANS\Memory Images\SavedRedlineAudits\nromanoff\nromanoff.mans
Home ▸ Host ▸ Processes ▸ Full Detailed Information
Analysis Data
Processes
Hierarchical Processes
Driver Modules
Device Tree
Hooks
Timeline
Tags and Comments
Acquisition History
Host
IOC Reports
Not Collected
Enter string to find here...
Reg Ex
In All Fields
Clear Column Filters
Prev
Next
String
News-http://www.msnbc.msn.com/?OCID=MSNIE9Jumplist$http://www.msnbc.msn.com/favicon.ico
Entertainment1http://entertainment.msn.com/?OCID=MSNIE9Jumplist2http://col.stc.s-msn.com/br/gbl/lg/csl/favicon.ico
Sports-http://msn.foxsports.com/?OCID=MSNIE9Jumplist$http://msn.foxsports.com/favicon.ico
Money0http://moneycentral.msn.com/?OCID=MSNIE9Jumplist2http://col.stc.s-msn.com/br/gbl lg/csl/favicon ico
Lifestyle-http://lifestyle.msn.com/?OCID=MSNIE9Jumplist2http://col.stc.s-msn.com/br/gbl/lg/csl/favicon.ico
http://12.190.135.235/ads/
http://192.168.1.5/ads/
The URI must begin with "https://" || "http://".
http://12.190.135.235/ads/
<dpiAware xmlns="http://schemas.microsoft.com/SMI/2005/WindowsSettings">true</dpiAware>
http://%s/%s
22 Items
Strings
Image Load Events
DNS Lookup Events
Network Events
File Write Events
Registry Key Events
Process Events
Exploit Events
URL Monitor Events
Tags and Comments
Details
Duplicates
MRI Report
Sections
Handles
Ports

5. **Optional Homework: Create and run your own live analysis via a Redline Collector (NOTE: Redline does not currently support Windows 10 collection and analysis).**

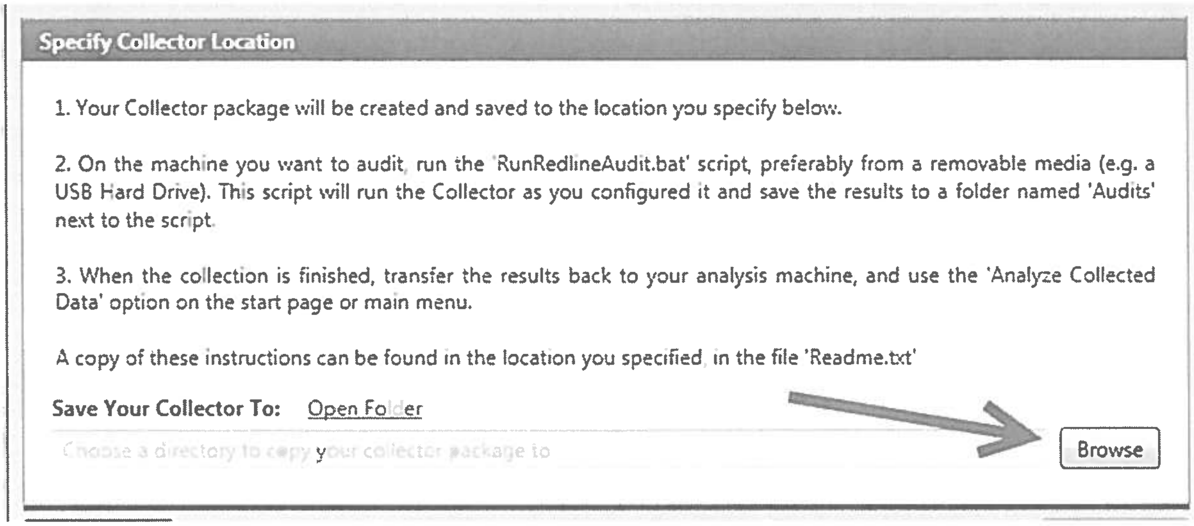
1) <Top Left Icon> → Create a Comprehensive Collector

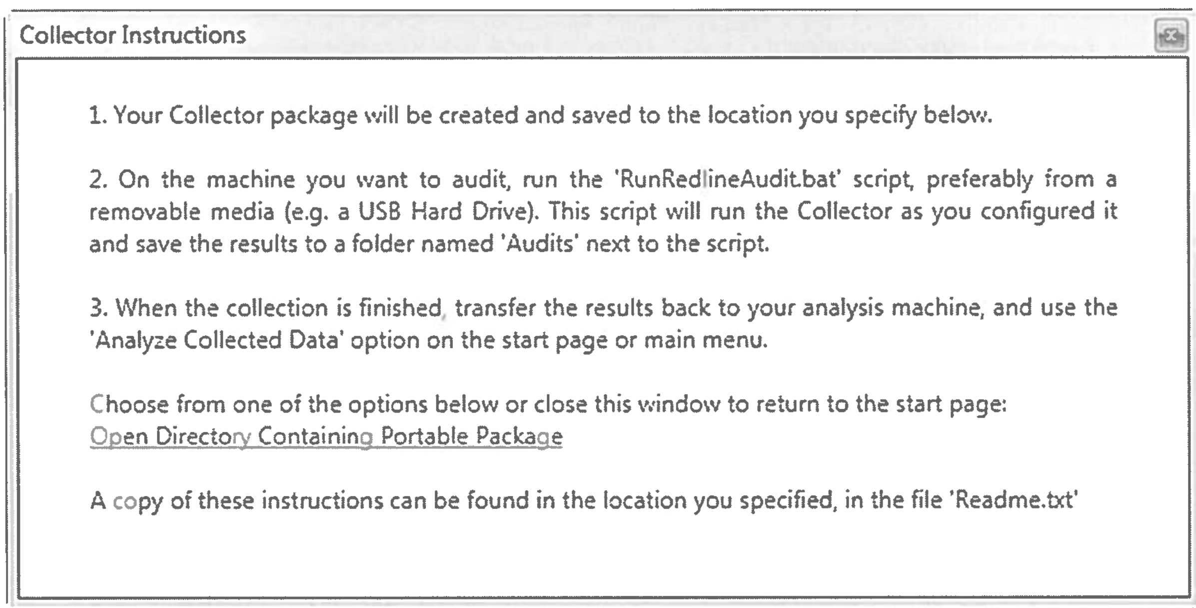


2) Edit your script:
   a. Go through the various menus and select what you would like the script to collect. Because we selected "Comprehensive Collector," most options are already enabled. Any additional selections will likely require a long time to complete.
   b. If you would also like the collector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



3) Specify Collector Location
   a. Place this on a USB thumb drive 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.

```
Administrator: Command Prompt

D:\Temp\RedlineCollector>dir
 Volume in drive D is Data
 Volume Serial Number is 9250-D7B6

 Directory of D:\Temp\RedlineCollector

09/18/2014  08:29 AM    <DIR>          .
09/18/2014  08:29 AM    <DIR>          ..
06/09/2014  03:25 PM             1,108 elevate.cmd
06/09/2014  03:25 PM             4,023 elevate.vbs
06/09/2014  03:24 PM             1,770 finishAnalysis.js
06/09/2014  03:24 PM               839 getNextSessionFolder.js
06/09/2014  02:00 PM             2,179 Helper.bat
09/18/2014  08:29 AM             5,208 MemoryzeAuditScript.xml
06/09/2014  03:25 PM               550 Readme.txt
06/09/2014  03:26 PM               832 RunRedlineAudit.bat
09/18/2014  08:29 AM    <DIR>          x64
09/18/2014  08:29 AM    <DIR>          x86
               8 File(s)         16,509 bytes
               4 Dir(s)  155,205,468,160 bytes free

D:\Temp\RedlineCollector>RunRedlineAudit.bat
```

```
C:\Windows\System32\cmd.exe

Executing command w32drivers-modulelist, 1.4.46.0
Pre-execution diagnostics for command w32drivers-modulelist
PageFaultCount: 27042 PeakWorkingSetSize: 29302784 WorkingSetSize: 15917056 Quot
aPeakPagedPoolUsage: 243848 QuotaPagedPoolUsage: 221344 QuotaPeakNonPagedPoolUsa
ge: 20240 QuotaNonPagedPoolUsage: 16832 PagefileUsage: 7680000 PeakPagefileUsage
: 9678848
CommitTotal: 2267418 CommitLimit: 4814297 CommitPeak: 2892849 PhysicalTotal: 419
1705 SystemCache: 265833 KernelTotal: 96609 KernelPaged: 63363 KernelNonpaged: 3
3246 PageSize: 4096 HandleCount: 46957 ProcessCount: 115 ThreadCount: 1420
<Issue number="0" level="Info" summary="System range 0x0000000000000000 - 0x0000
000000001000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x0000000100000000 - 0x0000
000430000000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x0000000000103000 - 0x0000
0000cf65f000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x0000000000100000 - 0x0000
000000102000" context="EnumerateDevices"/>

<Issue number="0" level="Info" summary="System range 0x000000000001f000 - 0x0000
00000009a000" context="EnumerateDevices"/>

<Issue number="17003" level="Info" summary="Internal InformationAlgorithm found
```

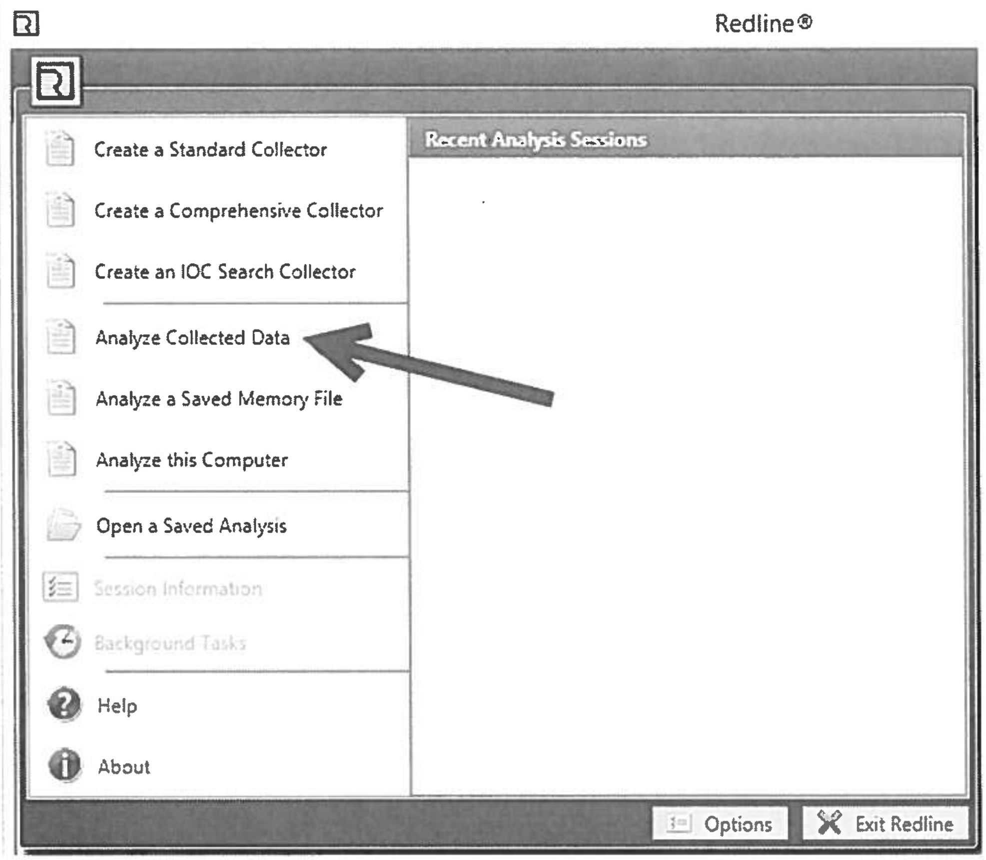
a. A full collection script can take hours to run. (In the future you can remove unnecessary options in the audit script to speed up processing.) Your results will be located in a folder named according to the System Hostname in the same location as your collector.

(D:) ▸ Temp ▸ RedlineCollector ▸ Sessions ▸ AnalysisSession1 ▸ Audits ▸ M4500 ▸ 20140918143212

Name	Date modified	Type	Size
w32drivers-modulelist.gmrE87wpBGdbV...	9/18/2014 8:37 AM	GMRE87WPBGDB...	80 KB
w32drivers-modulelist.issues.fGef3Dzm7...	9/18/2014 8:37 AM	FGEF3DZM7KBDX...	2 KB
w32drivers-signature.hjlt2KUmjtYdZHbw...	9/18/2014 8:35 AM	HJIT2KUMJTYDZH...	1,033 KB
w32drivers-signature.issues.lP7KxpEMoo...	9/18/2014 8:35 AM	LP7KXPEMOOSCY...	42 KB
w32kernel-hookdetection.ea4oTX9SCR9d...	9/18/2014 8:39 AM	EA4OTX9SCR9DRI...	1 KB
w32kernel-hookdetection.issues.kYwkVu...	9/18/2014 8:39 AM	KYWKVUXMJQED...	2 KB
w32processes-memory.issues.kPjBmfiGk...	9/18/2014 8:39 AM	KPJBMFIGKZSFLG...	0 KB

5) Open your results in Redline and start your analysis:

   a. <Top Left Icon> → Analyze Collected Data → Audit Location → <subfolder of folder named according to System Hostname>. (Or just double click the `.mans` file in the same folder.)



Exercise – Key Takeaways

- Three suspicious binaries were identified:
    - C:\windows\system32\dllhost\svchost.exe
    - C:\windows\system32\spinlock.exe (two instances)

- Lateral movement tradecraft found in the form of PsExec usage
    - Process PSEXESVC.exe
    - PsExec named pipes were identified in spinlock.exe handles referencing a remote system name WKS-WINXP32BIT

- An interesting indicator for spinlock.exe was identified
    - %USERPROFILE%\AppData\Local\Temp\python25.dll

- A suspicious registry key was discovered within the winclient.reg file.
    - MACHINE\SYSTEM\CONTROLSET001\SERVICES\NETMAN\DOMAIN

- Potentially malicious network connections were found:
    - Several embedded strings for http://12.190.135.235/ads were found in the malicious "svchost.exe" process.
    - A suspicious connection to 199.73.28.114 from spinlock.exe was identified

Exercise 2.3 – Volatility Memory Analysis

Objectives

1. Gain experience using Volatility to implement a memory forensics process.
2. Understand the various ways to display EPROCESS information.
3. Practice analyzing network connections found in memory.
4. Begin to employ "automation" and "baseline" plugins to find anomalies.

Exercise Preparation

1. Please ensure your evidence is mounted correctly at **/mnt/windows_mount**; if not, please follow the previous exercise "**Mounting Evidence Using SIFT,**" again to mount your evidence.

```
# cd /mnt/windows_mount

# ls
```

```
root@siftworkstation:/mnt/windows_mount# ls
$AttrDef      bootmgr                 $LogFile      Program Files              Users
autoexec.bat  BOOTSECT.BAK            $MFTMirr      Recovery                   $Volume
$BadClus      config.sys              MSOCache      $Recycle.Bin               Windows
$Bitmap       Documents and Settings  pagefile.sys  $Secure
Boot          $Extend                 PerfLogs      System Volume Information
$Boot         hiberfil.sys            ProgramData   $UpCase
```

If you do not see the above, then please follow directions in the Mounting Evidence Using SIFT exercise.

2. Change directory to the /cases/ win7-32-nromanoff-memory folder and make sure the win7-32-nromanoff-memory-raw.001 memory image is present. (You should have previously copied it into your SIFT Workstation VM)

```
# cd /cases/win7-32-nromanoff-memory/
# ls -al
```

```
root@siftworkstation:/cases/win7-32-nromanoff-memory# ls -al
total 2817304
drwxr-xr-x 3 root root       4096 Apr 17 21:48 .
drwxrwxrwx 9 root root       4096 Apr 14 21:39 ..
drwxr-xr-x 2 root root       4096 Apr 17 21:47 baseline-memory
-rwxr-xr-x 1 root root 2147483648 Apr 14 22:01 win7-32-nromanoff-memory-raw.001
-rw------- 1 root root       1540 Apr 17 21:48 win7-32-nromanoff-memory-raw.txt
```

Exercise – Questions

1. **Image identification.**
    - Run the **imageinfo** plugin against the win7-32-nromanoff-memory-raw.001 image.

```
# vol.py -f win7-32-nromanoff-memory-raw.001 imageinfo
```

    i. When was the memory image created?

    ____

    ii. Which profile should be specified when analyzing this image?

    ____

2. **Identify rogue processes.**
    - Use the **psscan** plugin to scan the memory image for EPROCESS blocks. Do you see any interesting process names that did not show up in Redline?

    ____

    - What does the "Time exited" column mean from the **psscan** output?

    ____

    - Use the **pstree** plugin to identify the parent of the suspicious svchost.exe (PID 6404) process. What is its PID?

    ____

        i. Do you see any other interesting processes with that parent we have not seen before?

        ____

        ii. Keep reviewing the **pstree** output. Can you find another obviously suspicious block of processes by visualizing their parent/child relationships?

        ____

        ____

        ____

- Use the **pstotal** plugin to create a visualization graph of the processes.

```
# vol.py -f win7-32-nromanoff-memory-raw.001
--profile=Win7SP1x86 pstotal --cmd --output=dot --output-
file=/cases/win7-32-nromanoff-memory/memory-graph.dot

# dot -Tpng /cases/win7-32-nromanoff-memory/memory-graph.dot
-o /cases/win7-32-nromanoff-memory/memory-graph.png

# xdot /cases/win7-32-nromanoff-memory/memory-graph.dot
```

Use the built in **xdot** tool to open the native **memory-graph.dot** file or browse to the **/cases/win7-32-nromanoff-memory** directory, and double-click the file **memory-graph.png** to view the graphic.

3. **Analyze process DLLs and handles.**
   - Use the **dlllist** plugin to identify the full path and command line of the svchost.exe (PID= 6404) binary.

     ____

   - Run **getsids** on the svchost.exe (PID 6404) process to identify the primary token and user account name used to start the process. (Note: This is a compromised account and should be listed in our indicator list!)

     ____

     i. What level of permissions does this account have?

        ____

     ii. Extra credit: What other forensic artifacts could be used to validate the SID to account name mapping performed by **getsids**?

        ____

- Review process handles
    i. Run the **handles** plugin on the rogue svchost.exe process.

    ii. Looking at the output, it should be clear why least frequency of occurrence is useful in narrowing down items to analyze. Now narrow your view to just registry "Key" handles. (Use the "-t Key" option.) Do you see a handle for the suspicious registry key previously found in the Redline exercise?

    ____

4. **Automate the search for anomalous system processes.**
    - Run **malsysproc** to identify system processes with the wrong path, parent, command line, number running (count), and so on. What processes are identified as anomalous? (Hint: Look for columns that have "False" values.)

    ____

5. **Review network artifacts.**
    - Run **netscan** and review the output for suspicious connections.

    ____

    ____

    ____

    ____

    - What connection is a good candidate for a malware beacon?

    ____

    - Port 445 is used for SMB over IP traffic, and is commonly an indication of mounted file shares. What IP addresses are connected to using this port?

    ____

    ____

6. **Optional homework: Use a baseline image to filter output.**
You will find a baseline memory image for a SRL Win7SP1x86 system in the folder **`/cases/win7-32-nromanoff-memory/baseline-memory`**. In this optional exercise we will use that baseline image to help filter and whitelist large amounts of noise data.

Services

- Determine how many services are found in the **`win7-32-nromanoff-memory-raw.001`** image using the **`svcscan`** plugin, **`grep`**, and **`wc`**.

```
# vol.py -f win7-32-nromanoff-memory-raw.001
--profile=Win7SP1x86 svcscan | grep "Service Name" | wc -l
```

____

- Use the baseline plugin **`servicebl`** with the –K (known) option to show services that exist in both the current image and in the baseline. Review the output to get a feel for the format.

```
# vol.py -f win7-32-nromanoff-memory-raw.001
--profile=Win7SP1x86 -B ./baseline-
  memory/Win7SP1x86-baseline.img servicebl -K
```

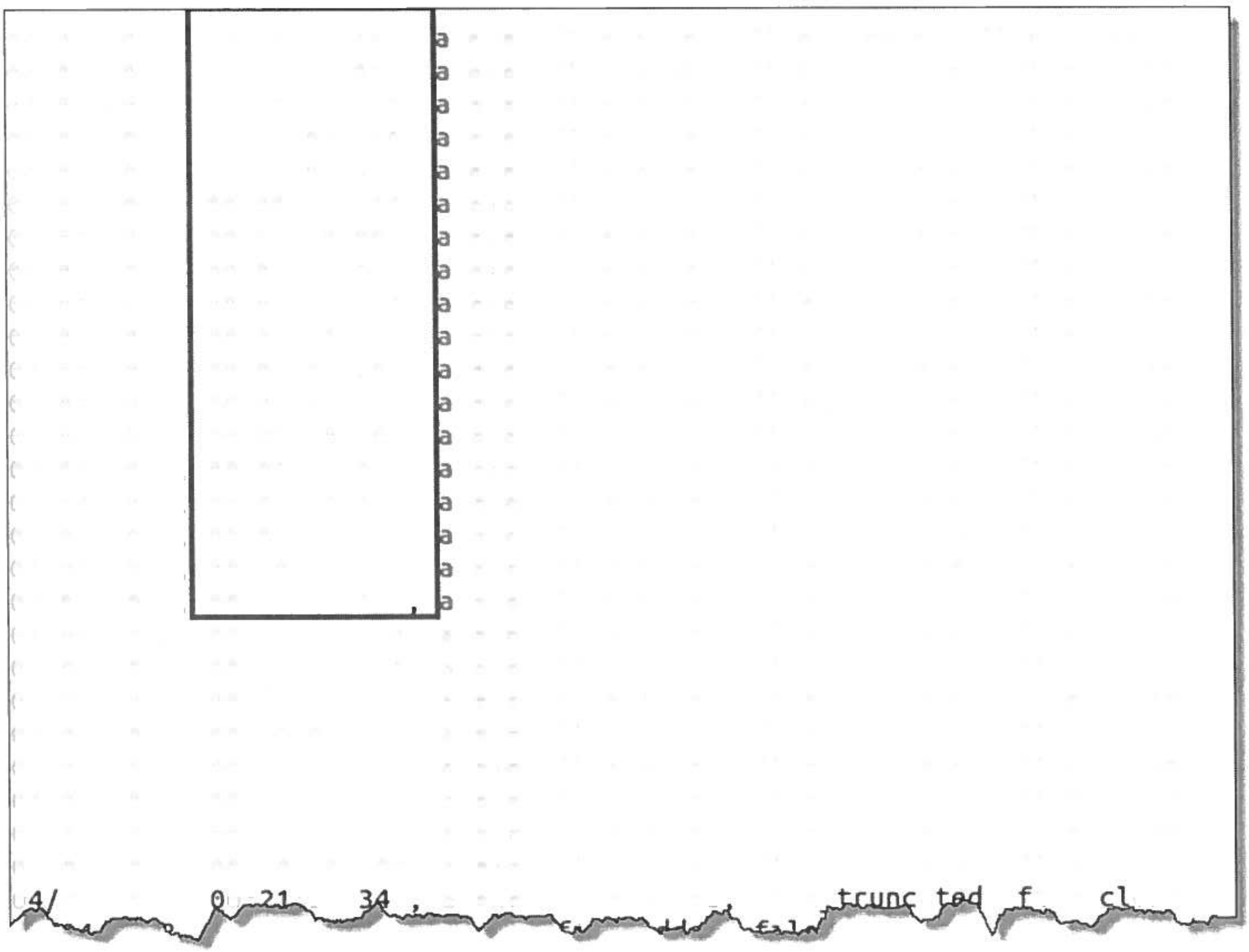
- Run the same **`servicebl`** command using the –K option but now use **`grep`** to filter for known services that are different from the baseline (contain "False" values in one or more columns). How many services were found? Later, you can use **`svcscan`** to review details for each service in the nromanoff image and the baseline image.

```
# vol.py -f win7-32-nromanoff-memory-raw.001
--profile=Win7SP1x86 -B ./baseline-
memory/Win7SP1x86-baseline.img servicebl -K | grep False
```

____

____

```
# [illegible]

# jp -image /mnt/ewf_mount/ewf1 -[illegible] | grep x\.exe > /cases/win7-
32-nromanoff-c-drive/jp-bodyfile.txt

# less /cases/win7-32-nromanoff-c-drive/jp-bodyfile.txt
```



Exercise 6.1 – APT Challenge Prep

```
# sudo su -
# cd /cases/[illegible]-c-drive/
```

```
# [illegible] /mnt/ewf_mount/
# mountwin /mnt/ewf_mount/ewf1 /mnt/windows_mount
```

[redacted]

[redacted]

Exercise 6.2 - 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. *Log in to the VMware machine.*
   - *LOGIN* = `sansforensics`
   - *PASSWORD* = `forensics`
3. *Start a terminal, elevate your privs to root, and change to the /cases/win7-64-nfury-c-drive directory.*

```
$ sudo su -

# cd /cases/win7-64-nfury-c-drive
```

4. *Mount your evidence files so that 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/

# mountwin /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 mountwin $i /mnt/shadow_mount/$i; done
```

Create Supertimeline across the core image and the VSS image

Create a VSS Supertimeline – Step-by-Step Guide

1. Run Log2timeline.py.

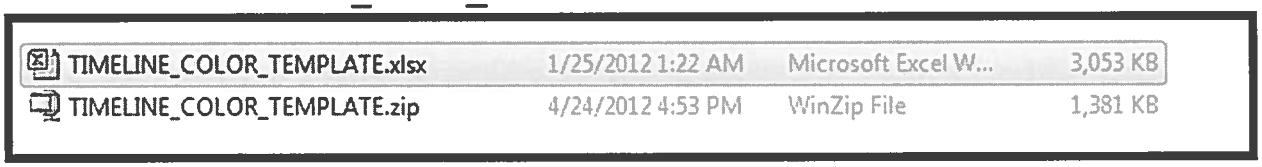
```
# cd /cases/win7-64-nfury-c-drive

# 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`.

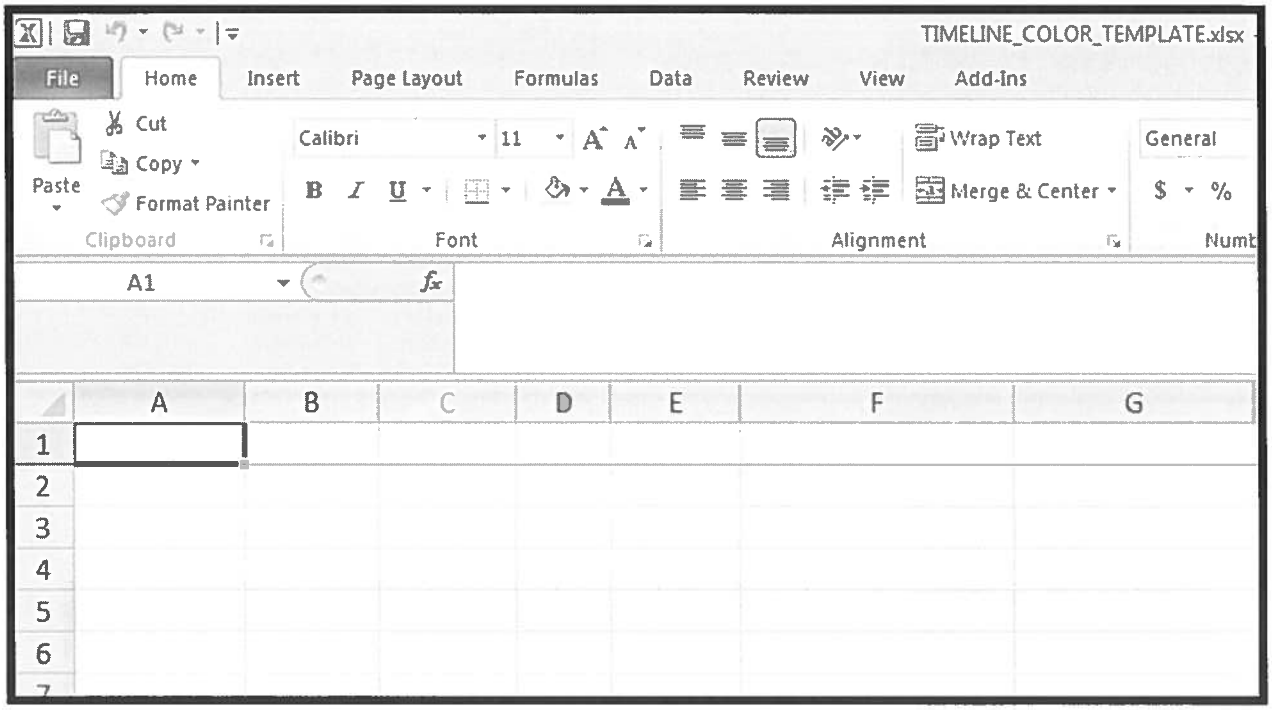
```
# psort.py -z "EST5EDT" -o L2tcsv win7-64-nfury-plaso.dump "date >
'2012-04-02 20:00:00' AND date < '2012-04-07 00:00:00'" >
nfury.timeline.csv
```

3. Switch over to your Windows Workstation (host system preferred) and open the \\SIFTWORKSTATION from an open folder on your Desktop.

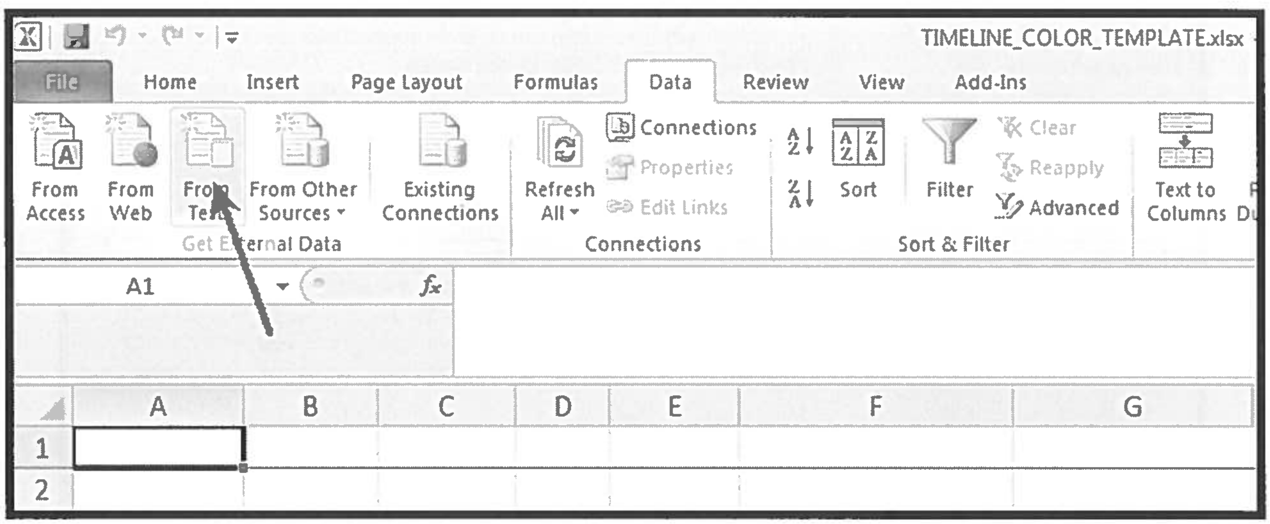
4. Take the filtered timeline and open it in Excel using the color-timeline template from your Windows system.
    a. Open Timeline Color Template:
        i. `TIMELINE_COLOR_TEMPLATE.xlsx`



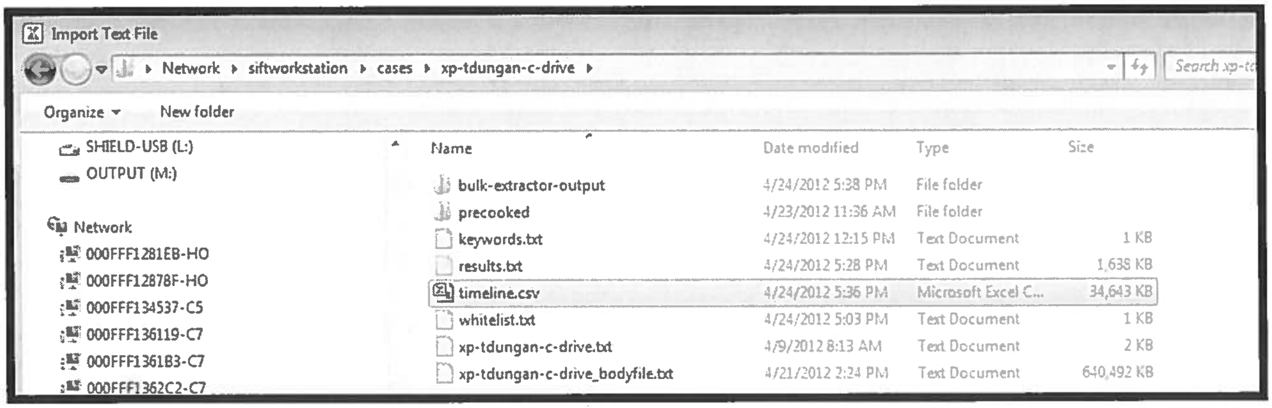
    b. Switch to the Color Timeline worksheet/tab.
    c. Click Cell A-1.



d. Select the "DATA" Ribbon.
e. Import Data "FROM TEXT."

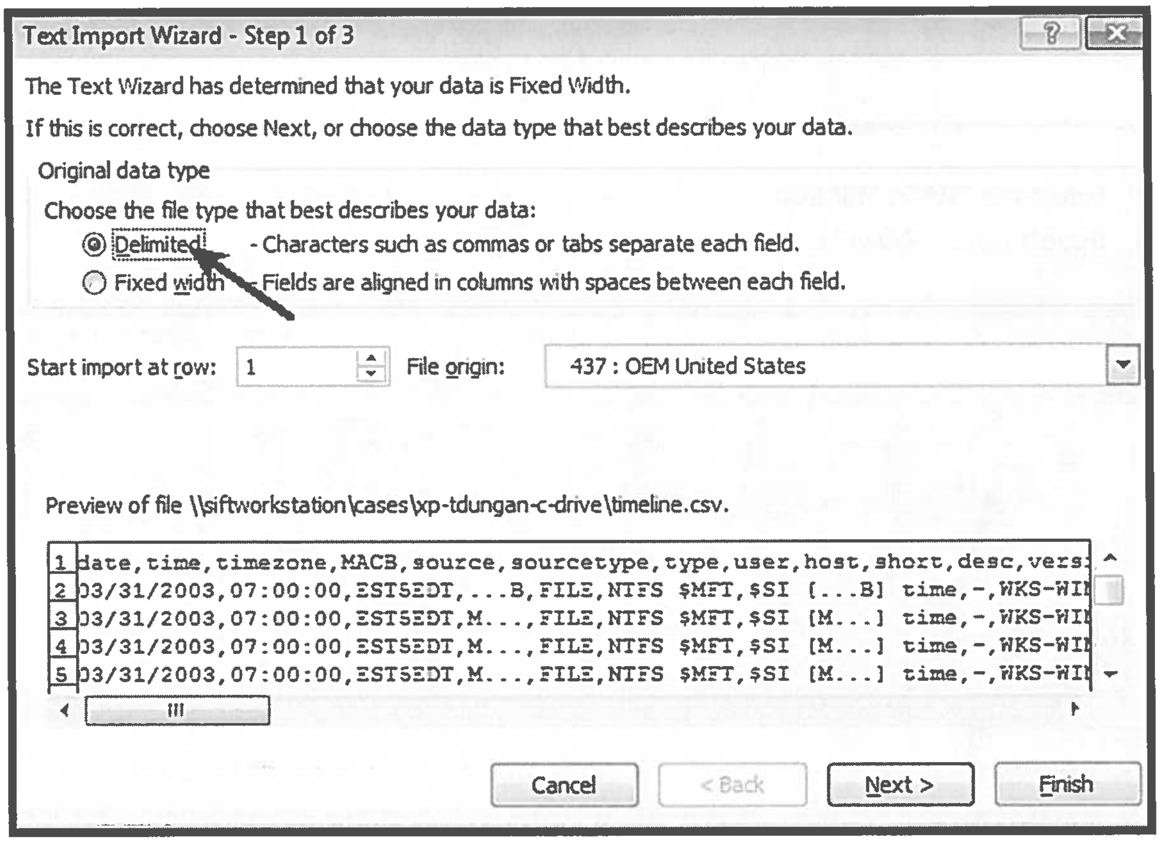


f. Select **nfury.timeline.csv** file ->**\\siftworkstation\cases\win7-64-nfury-c-drive.**

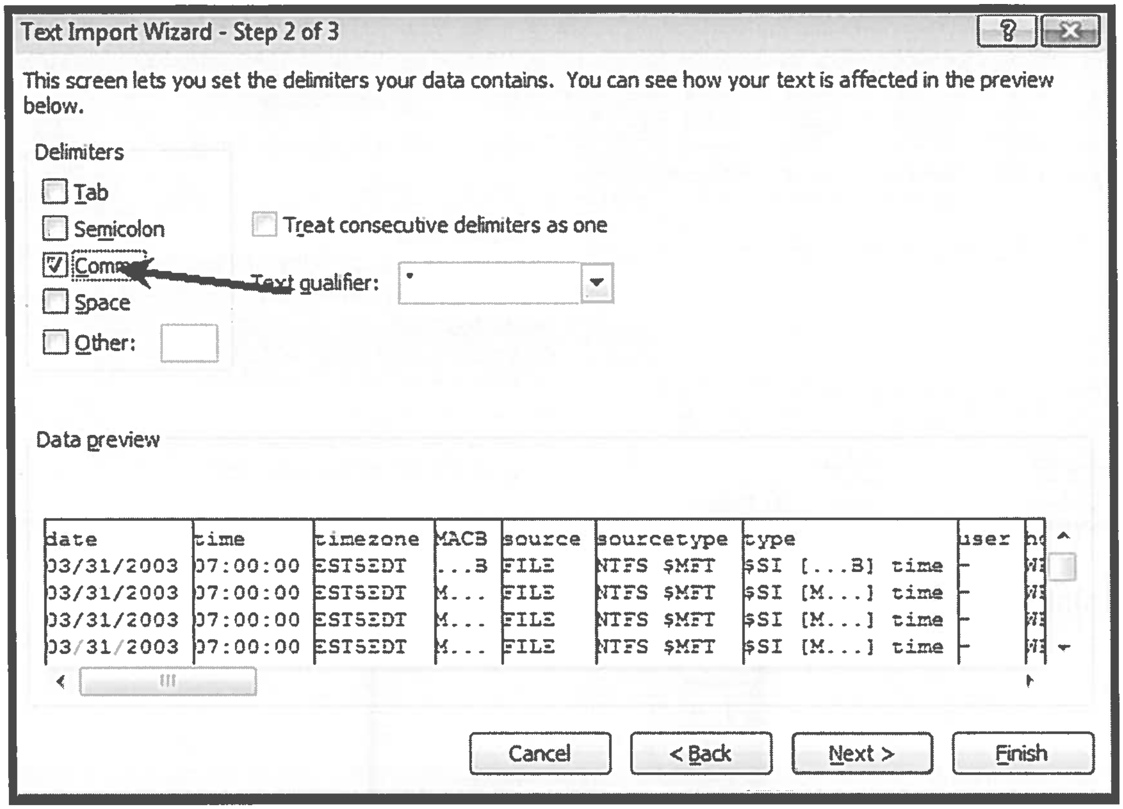


g. The Text Import Wizard will start.

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

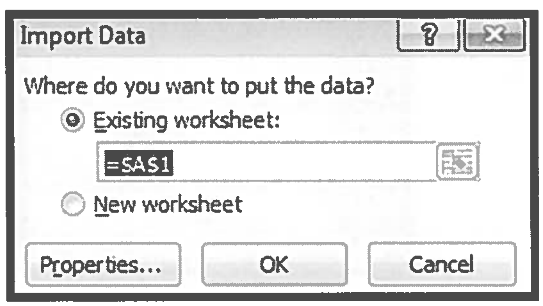


i. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT.

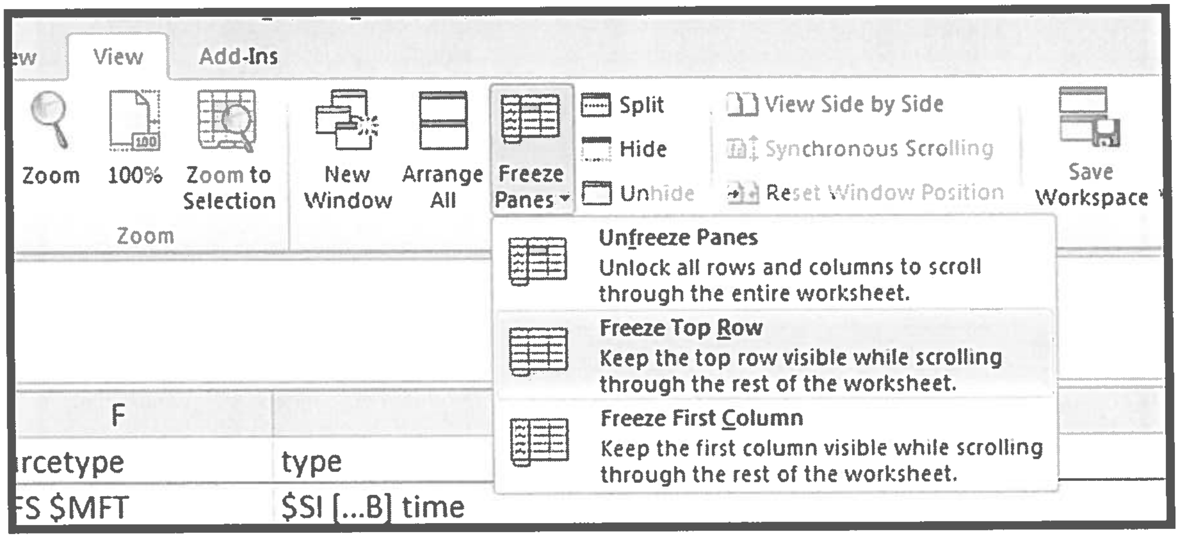


j. Step 3 ->Select Finish.

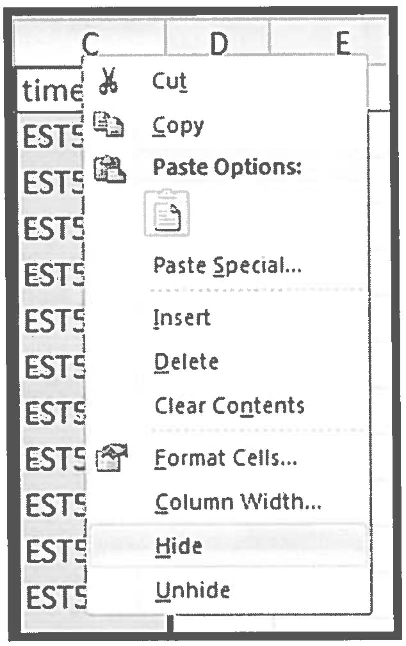
k. Where do you want to put the data? Simply Select OK.



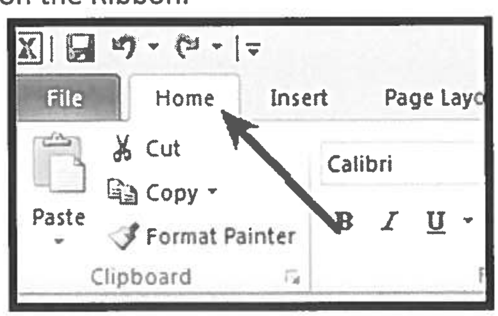
l. When imported View -> Freeze Panes -> Freeze Top Row.



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

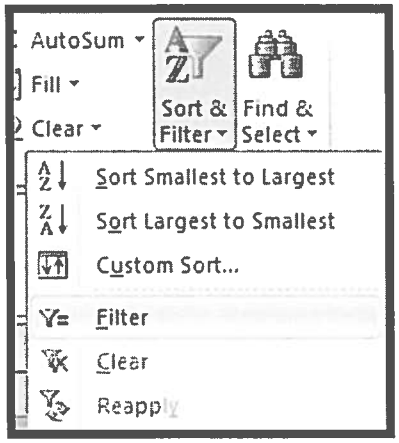


n. Select the HOME tab on the Ribbon.



o. Select All Cells or "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.

Exercise 6.3 XP TDUNGAN 10.3.58.7 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. *Log in to the VMware machine.*
   - *LOGIN* = `sansforensics`
   - *PASSWORD* = `forensics`
3. *Start a terminal, elevate your privs to root, and change in the /cases/win7-c-drive directory.*

```
$ sudo su -

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

4. *Mount your evidence files so that you can see the xp-tdungan-c-drive raw image and files/folders from the system.*

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

# mountwin /mnt/ewf_mount/ewf1 /mnt/windows_mount
```

Create Supertimeline across the core image and the VSS image

Create a VSS Supertimeline – Step-by-Step Guide

1. Run Log2timeline.

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

# log2timeline.py xp-tdungan-plaso.dump xp-tdungan-c-drive.E01
```

2. Filter using `psort.py`.

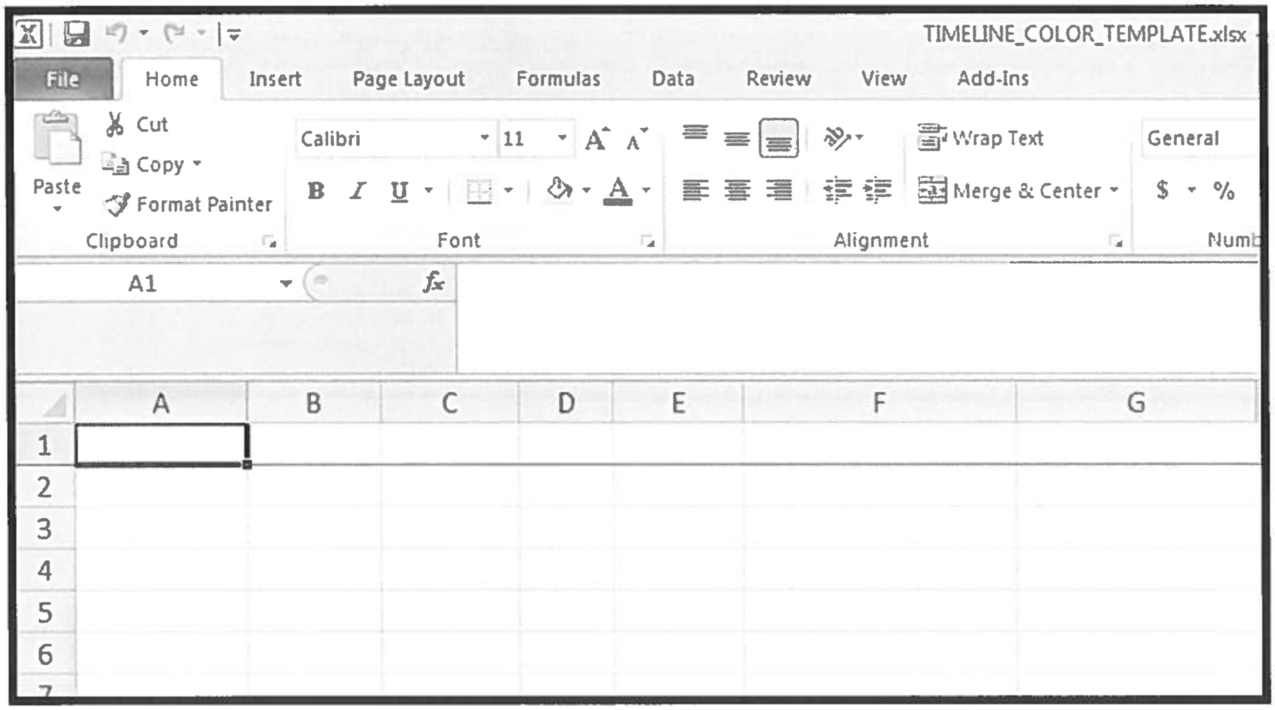
```
# psort.py -z "EST5EDT" -o L2tcsv xp-tdungan-plaso.dump
"date > '2012-04-02 20:00:00' AND date < '2012-04-07 00:00:00'" >
tdungan.timeline.csv
```

Switch to your Windows workstation (host system preferred) and open the \\SIFTWORKSTATION from an open folder on your Desktop.

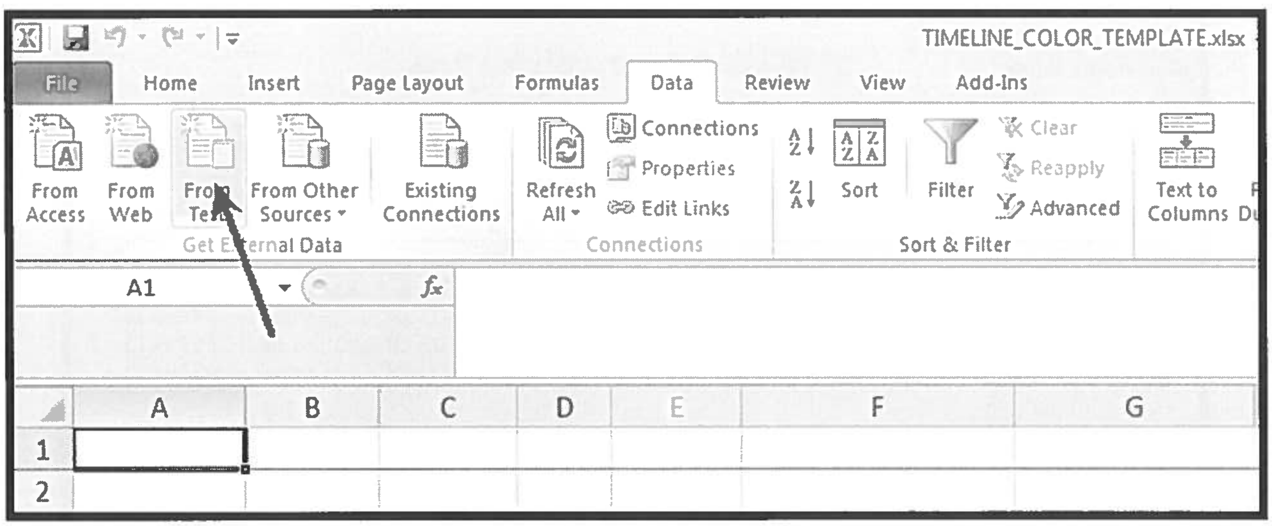
3. Take the filtered timeline and open it in Excel using the color-timeline template from your Windows system.
   a. Open Timeline Color Template: `TIMELINE_COLOR_TEMPLATE.xlsx.`

TIMELINE_COLOR_TEMPLATE.xlsx	1/25/2012 1:22 AM	Microsoft Excel W...	3,053 KB
TIMELINE_COLOR_TEMPLATE.zip	4/24/2012 4:53 PM	WinZip File	1,381 KB

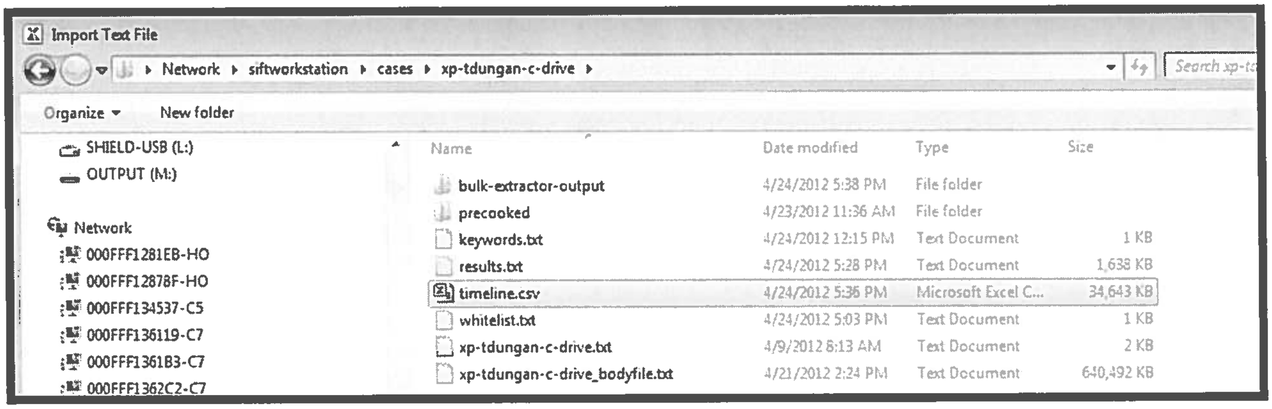
   b. Switch to Color Timeline worksheet/tab.
   c. Click Cell A-1.



d. Select "DATA" on the Ribbon.
e. Import Data "FROM TEXT."

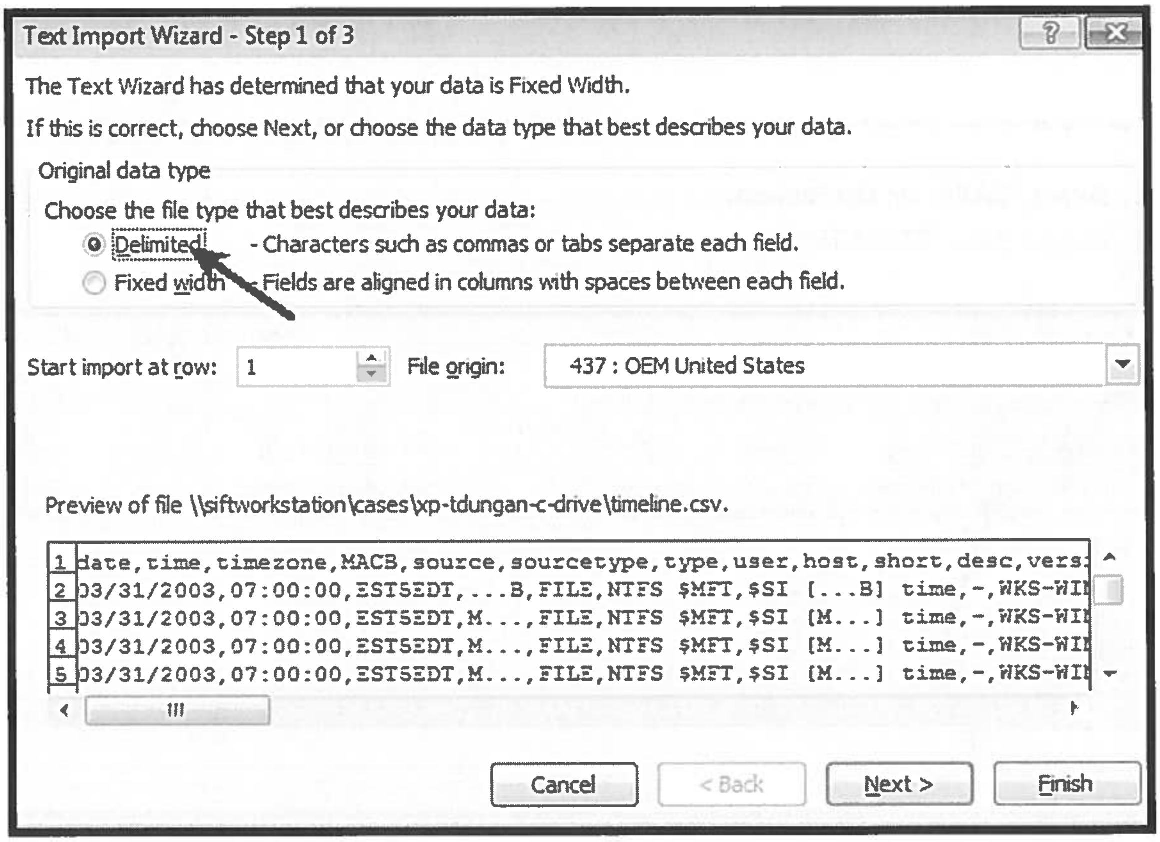


f. Select `nromaoff.timeline.csv` file ->`\\siftworkstation\cases\xp-tdungan-c-drive`.

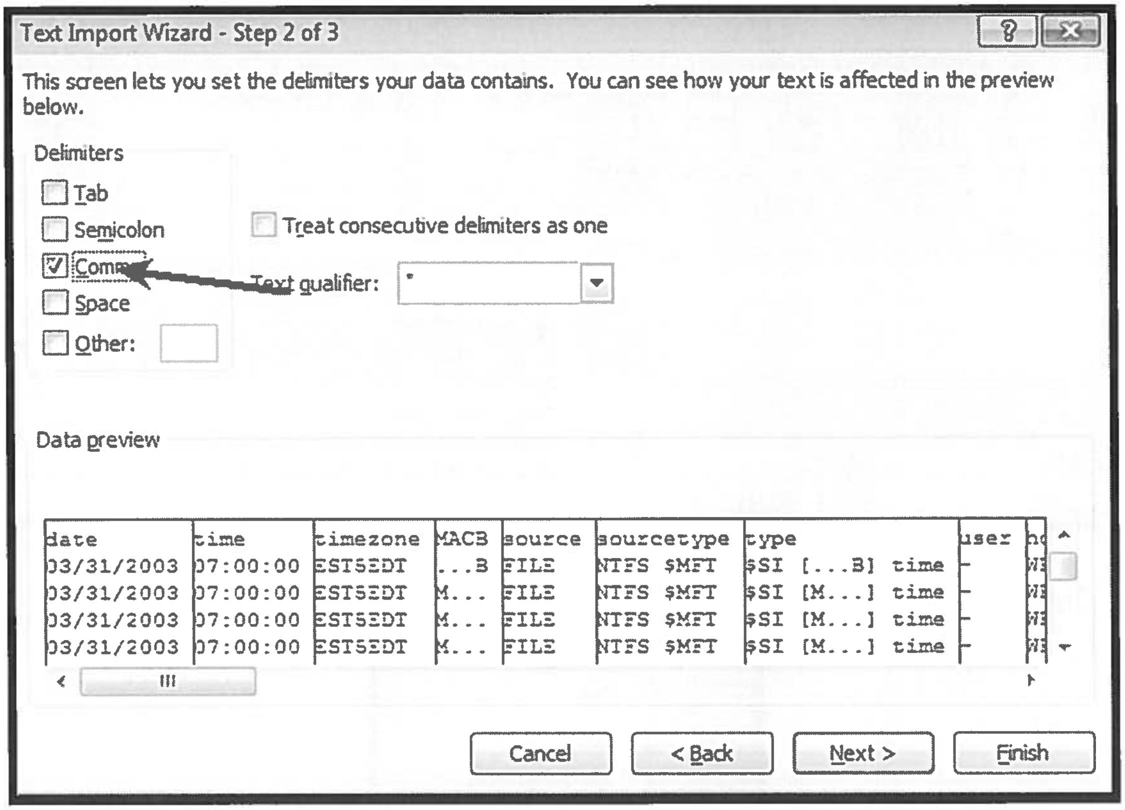


g. The Text Import Wizard will start.

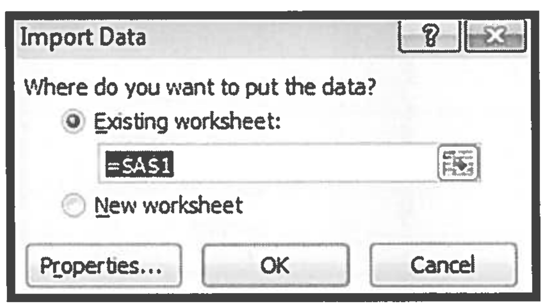
h. Step 1 -> Select Delimited -> Select NEXT.



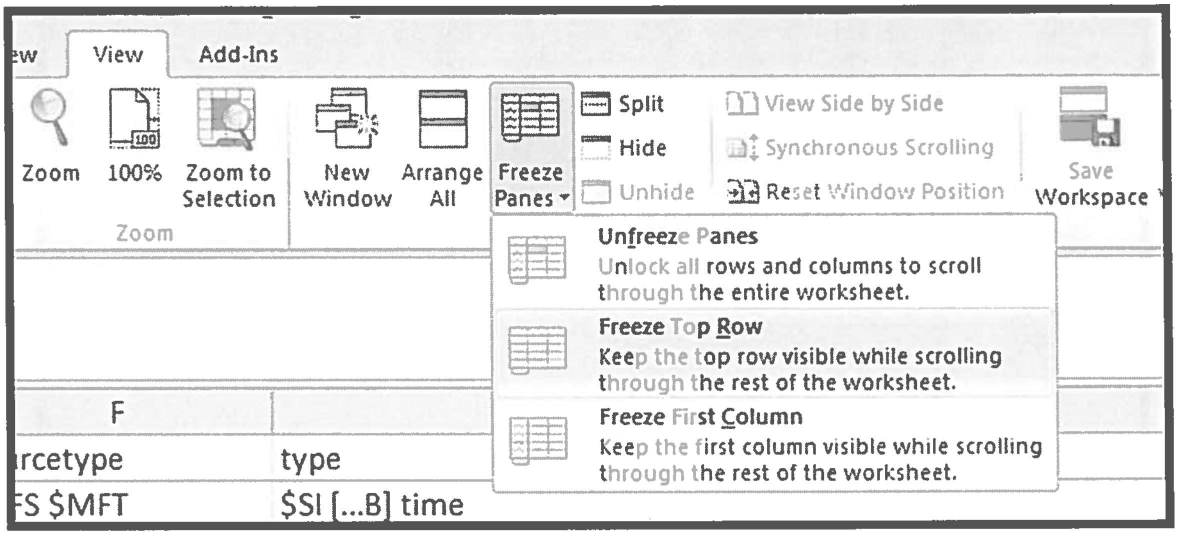
i. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT >.



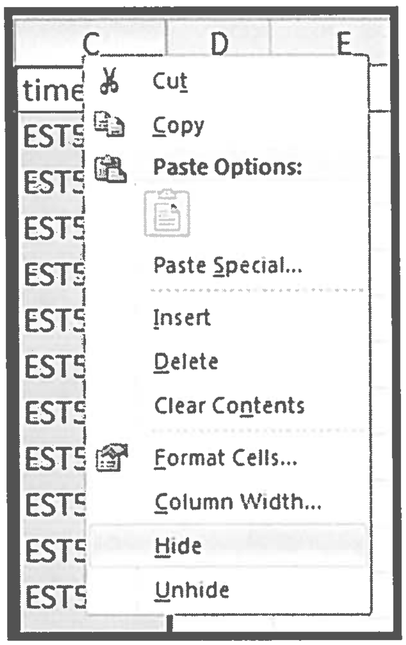
j. Step 3 ->Select Finish.
k. Where do you want to put the data? Simply Select OK.



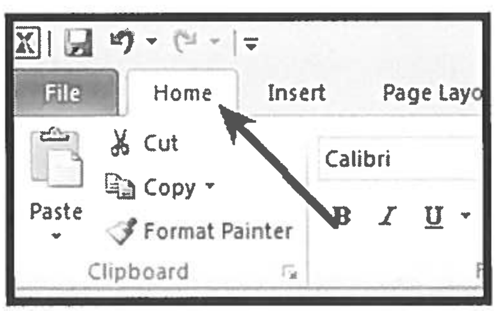
l. When imported View -> Freeze Panes -> Freeze Top Row.



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

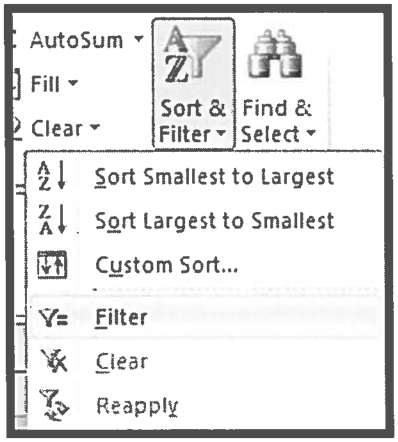


n. Select HOME on the Ribbon.



o. Select all Cells or "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/vss-supertimeline.xlsx`

This page intentionally left blank.

Exercise 6.4 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/

# mountwin /mnt/ewf/ewf1 /mnt/windows_mount
```

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`

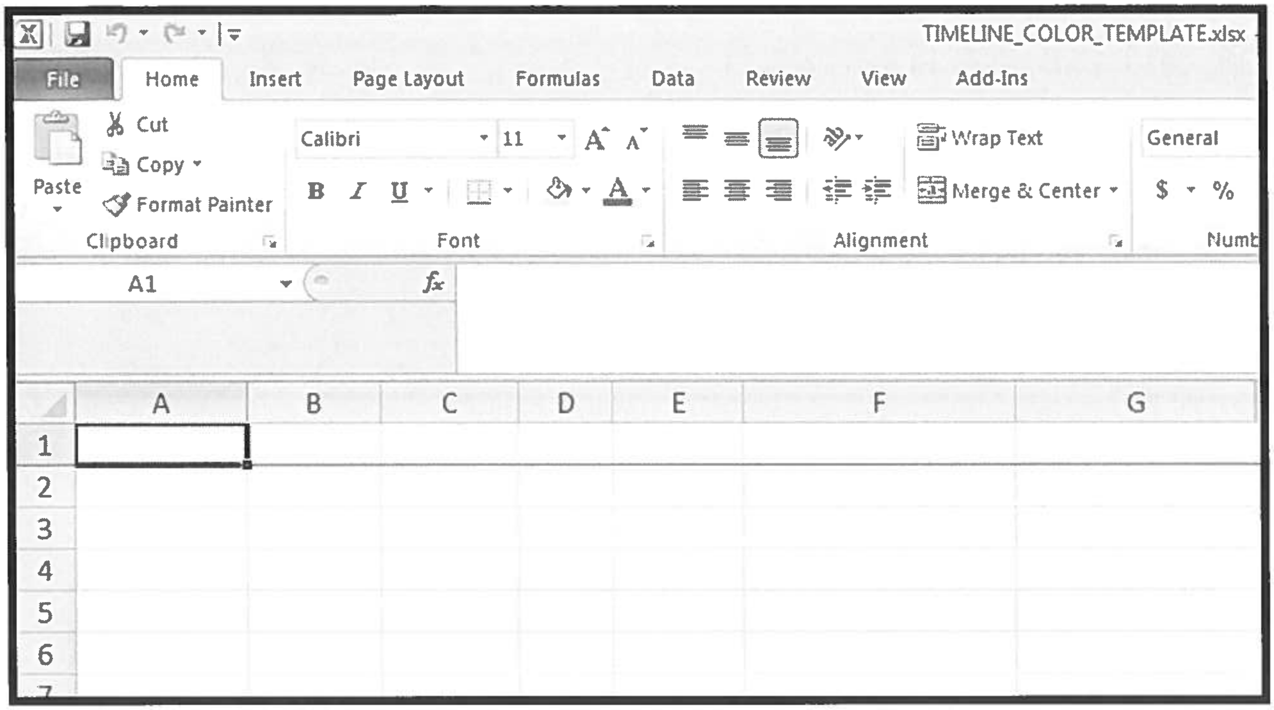
```
# psort.py -z "EST5EDT" -o L2tcsv win2008-dc-plaso.dump
"date > '2012-04-02 20:00:00' AND date < '2012-04-07 00:00:00'" >
2008dc.timeline.csv
```

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`**

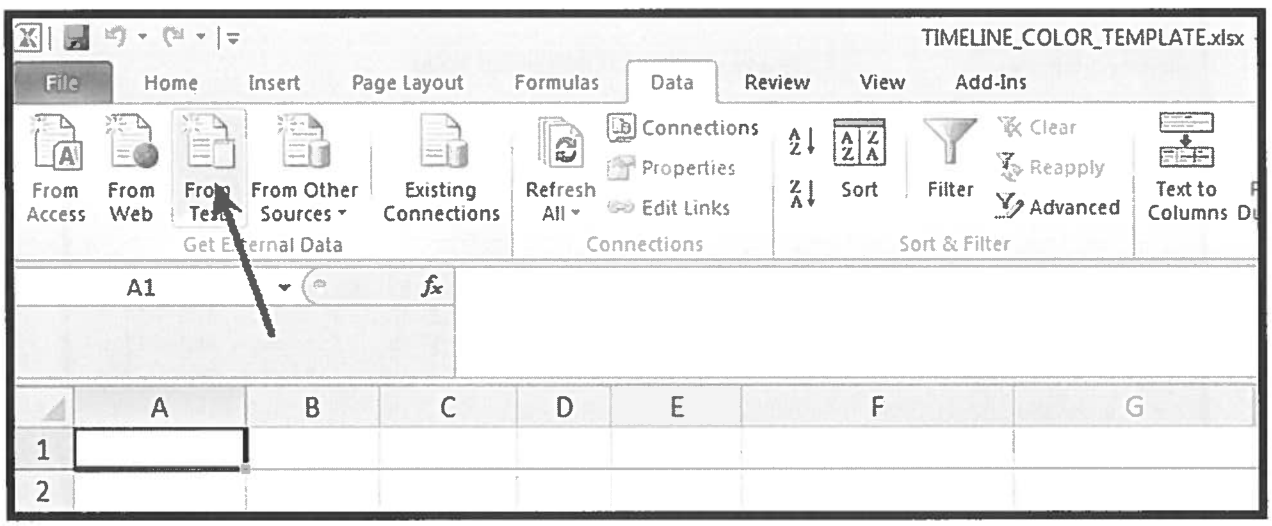
TIMELINE_COLOR_TEMPLATE.xlsx	1/25/2012 1:22 AM	Microsoft Excel W...	3,053 KB
TIMELINE_COLOR_TEMPLATE.zip	4/24/2012 4:53 PM	WinZip File	1,381 KB

   b. Switch to Color Timeline worksheet/tab.
   c. Click on Cell A-1.

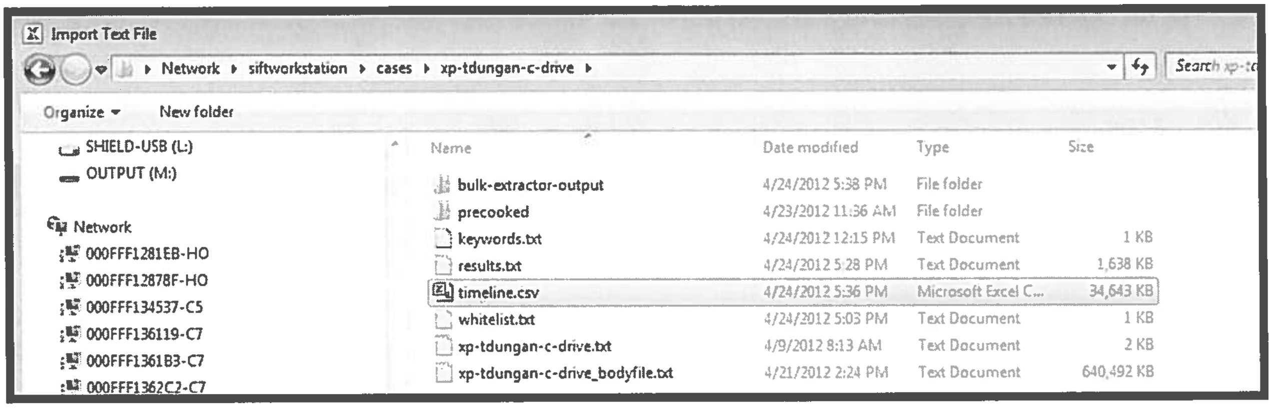


d. Select 'DATA' Ribbon.

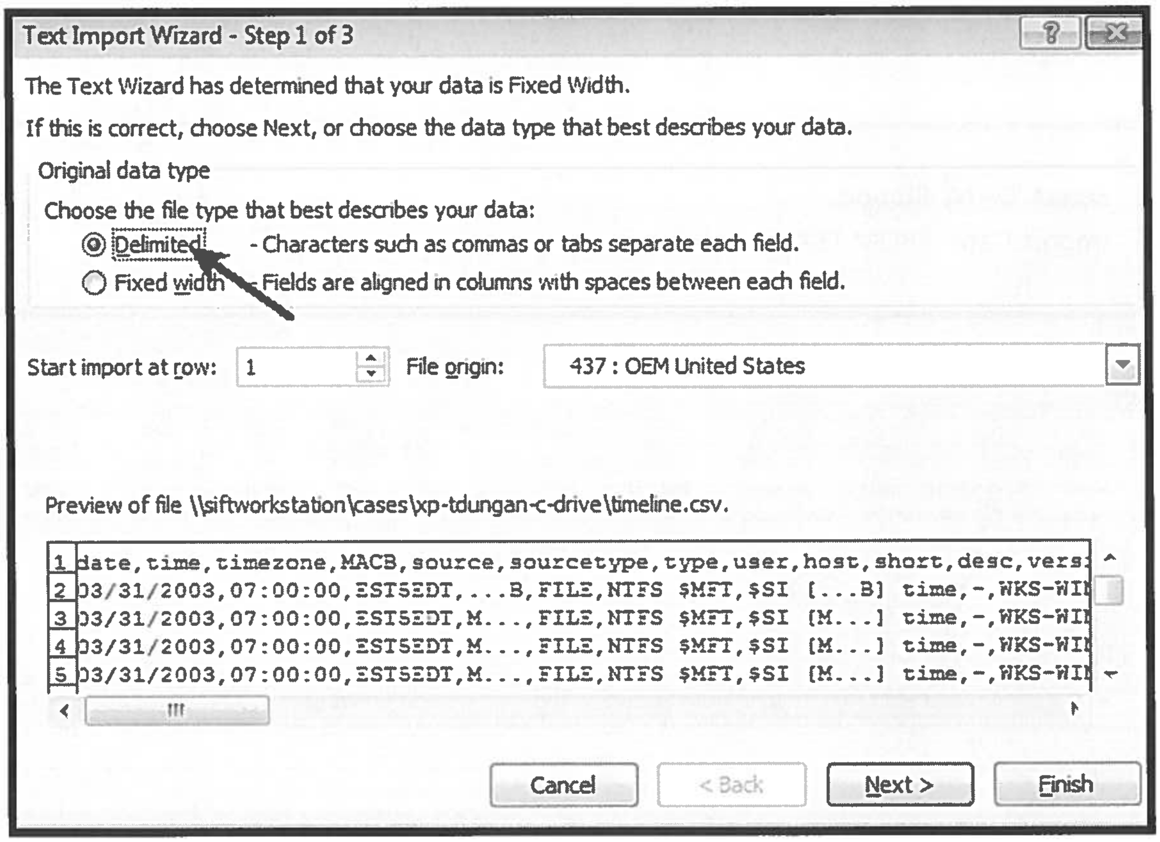
e. Import Data "FROM TEXT".



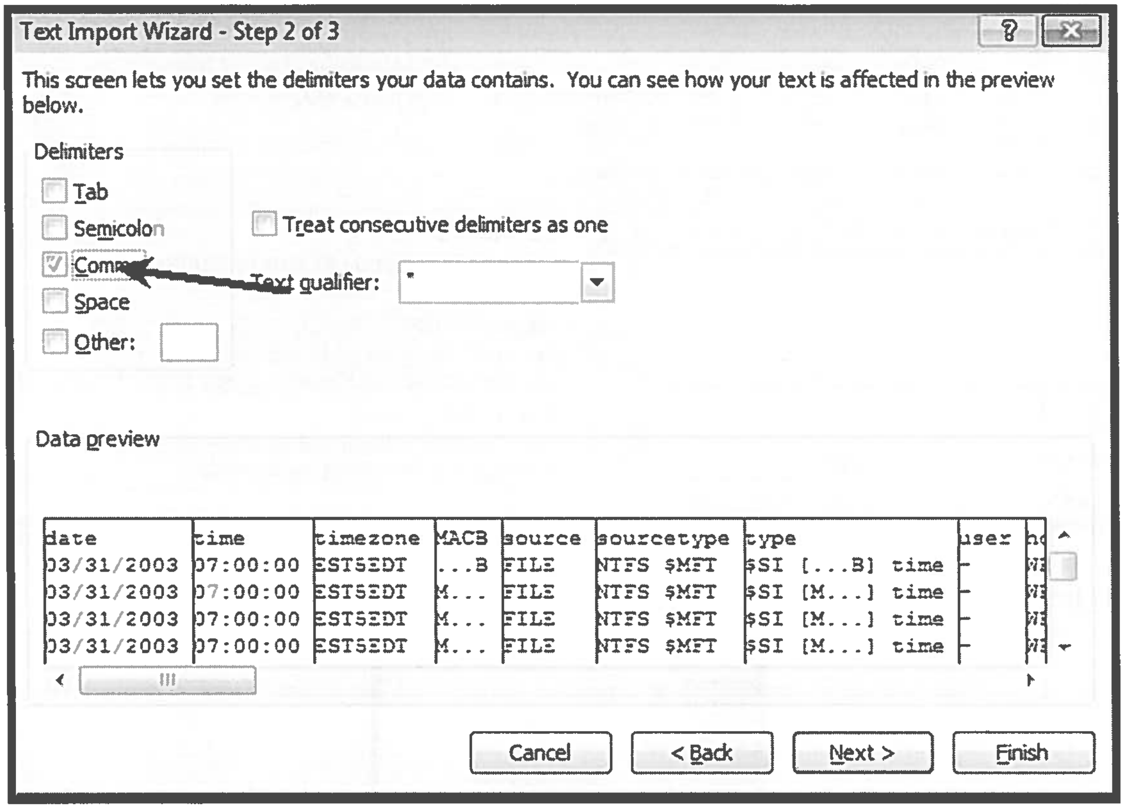
f. Select **2008dc.timeline.csv** file -> **\\siftworkstation\cases\win2008R2-controller-c-drive**

g. 

h. TEXT IMPORT WIZARD Will Start.
i. Step 1 -> Select Delimited -> Select NEXT.

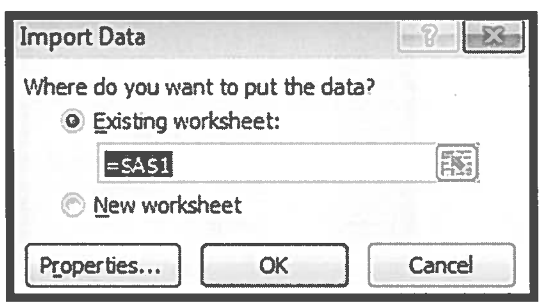


j. Step 2 -> Unselect Tab under Delimiters -> Select Comma under Delimiters -> Select NEXT >

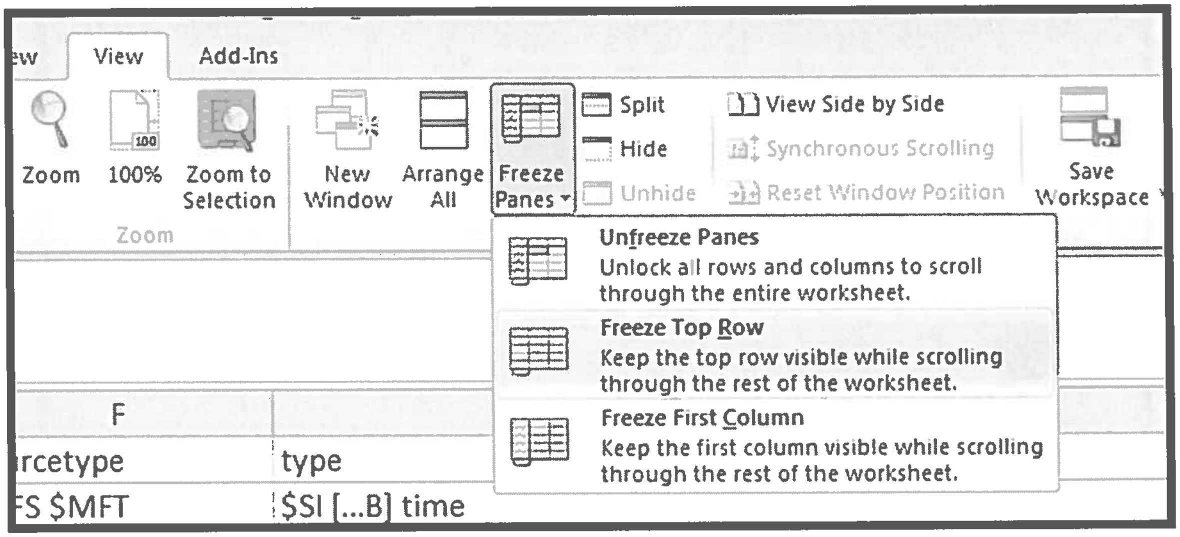


k. Step 3 ->Select Finish.

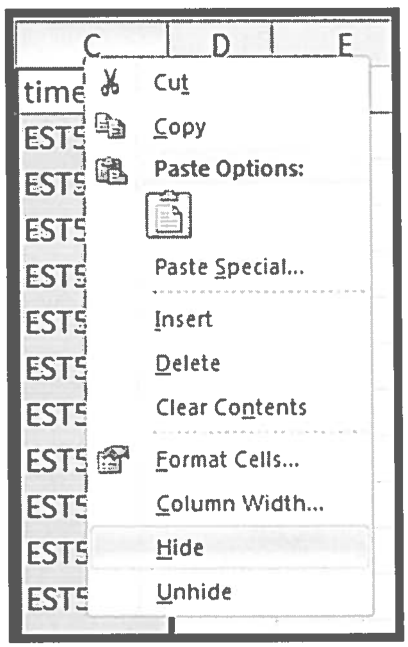
l. Where do you want to put the data? Simply Select OK.



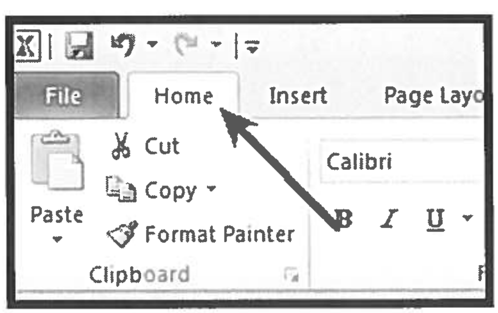
m. Once imported View -> Freeze Panes -> Freeze Top Row.



n. Optional Hide Columns Time Zone, Host, Version.

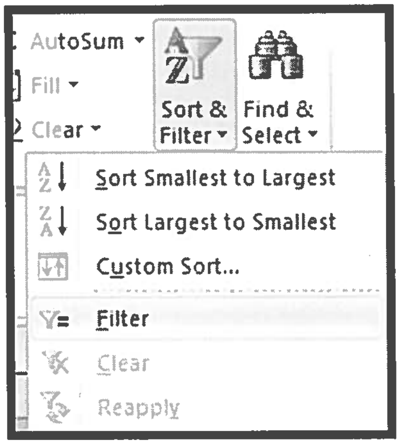


o. Select HOME Ribbon.



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`**