

LAB 8: Configuring a Malware Lab

Purpose

- Basic Static Techniques
- Basic Dynamic Techniques

What you need:

A Windows 2008 Server virtual machine with a Kali virtual machine running INetSim, which you preped in the previous project.

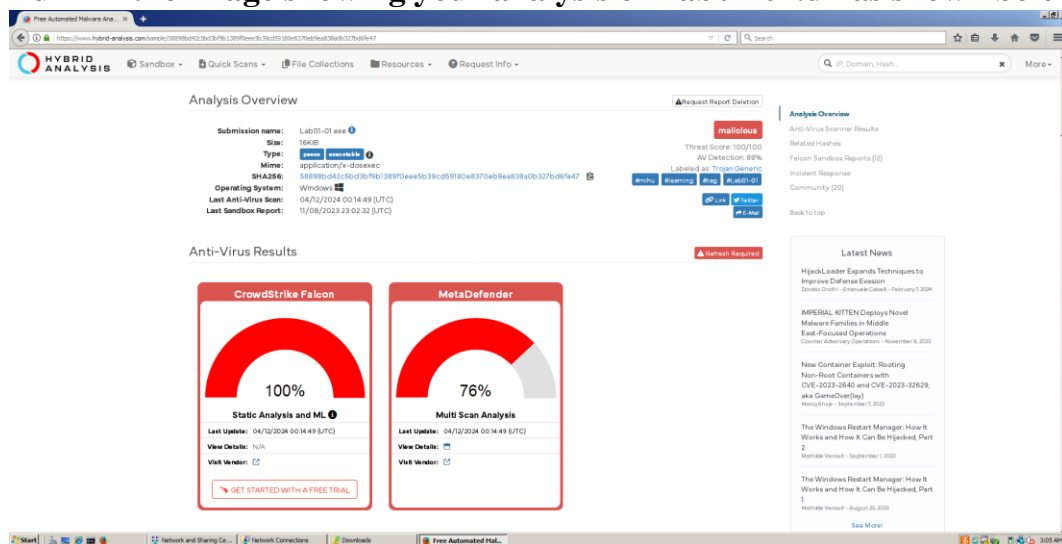
1. Basic Static Techniques

a. Lab01-01.exe

<https://www.hybrid-analysis.com/>:

Upload the Lab01-01.exe and Lab01-01.dll files to <https://www.hybrid-analysis.com>

Turn in the image showing your analysis of Lab01-01.dll as shown below.



Analysis Overview

Request Report Deletion

Submission name: Lab01-01.exe
Size: 16KiB
Type: [peexe](#) [executable](#)
Mime: application/x-dosexec
SHA256: 58898bd42c5bd3bf9b1389f0eee5b39cd59180e8370eb9ea838a0b327bd6fe47
Operating System: Windows
Last Anti-Virus Scan: 04/12/2024 00:14:49 (UTC)
Last Sandbox Report: 11/08/2023 23:02:32 (UTC)

malicious

Threat Score: 100/100

AV Detection: 88%

Labeled as: Trojan.Generic

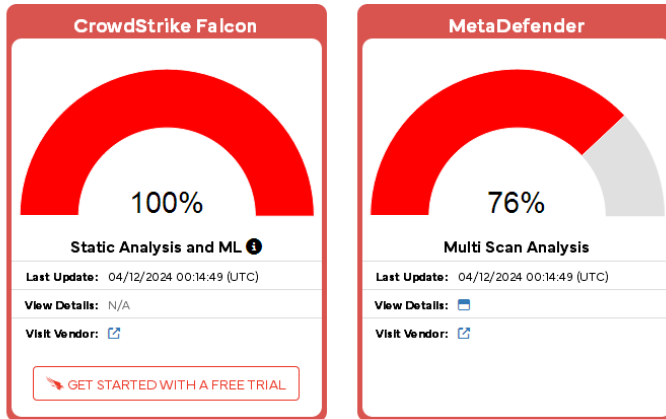
[#mihu](#) [#learning](#) [#tag](#) [#Lab01-01](#)

[Link](#) [Twitter](#)

[E-Mail](#)

Anti-Virus Results

Refresh Required



The screenshot shows the Hybrid Analysis website interface. The top navigation bar includes 'Free Automated Malware Analysis', 'Sandbox', 'Quick Scans', 'File Collections', 'Resources', and 'Request Info'. The main content area displays the 'Analysis Overview' for a submission named 'Lab01-01.dll'. The submission details include a size of 160KiB, a type of 'peexe', a mime type of 'application/x-dosexec', and a SHA256 hash of 'f50e42c8d4fab649bde0398867e930b86c2a599e8db83b8260393082268f2dba'. The operating system is 'Windows', and the last anti-virus scan was on 04/11/2024 22:06:12 (UTC). The last sandbox report was on 10/28/2022 07:27:05 (UTC). The analysis results show a 'Threat Score' of 100/100, 'AV Detection' of 80%, and a label of 'Malware'. The results are categorized as 'Trojan', 'Installer', 'Downloader', 'Backdoor', 'Injector', 'ransomware', 'Zloader', and 'worm'. The 'Anti-Virus Results' section shows 'CrowdStrike Falcon' with a 'Static Analysis and ML' score of 100% and 'MetaDefender' with a 'Multi Scan Analysis' score of 60%. Both results include a 'Last Update' of 04/11/2024 22:06:12 (UTC) and a 'View Details' link. The 'GET STARTED WITH A FREE TRIAL' button is also present.

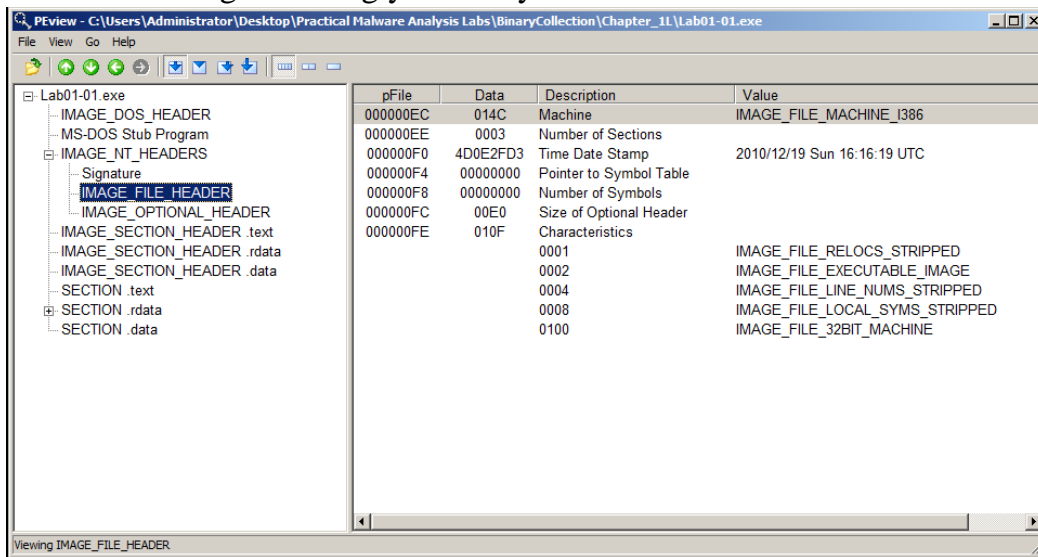
PEview

You can download PEview from here: <http://wjradburn.com/software/>

Open the files in PEview. For each file, find the "Time Date Stamp" as shown below.

The files were both compiled on the same date within a minute of each other, indicating that they are part of the same package.

Turn in the image showing your analysis of Lab01-01.exe as shown below.



PEiD

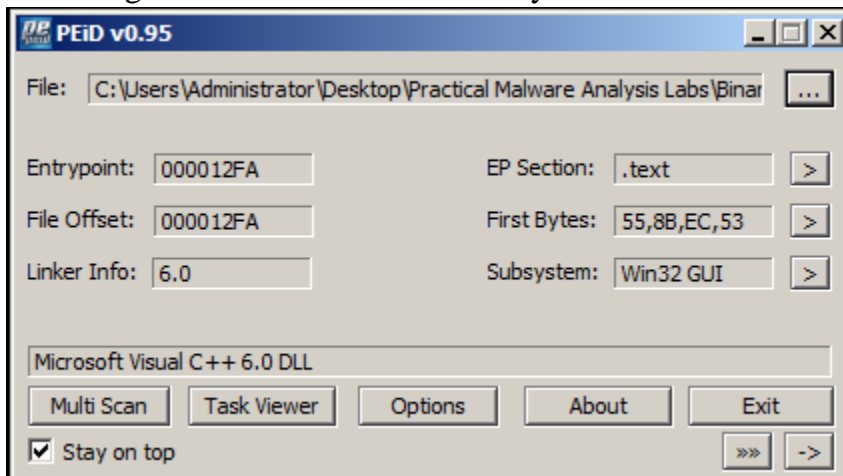
You can download PEiD here:

<http://www.softpedia.com/progDownload/PEiD-updated-Download-4102.html>

Open the files in PEiD. They are identified as "Microsoft Visual C++" files, which shows that they are unpacked.

Turn in the image showing your analysis of Lab01-01.dll as shown below.

We will grade it based on the "First Bytes".



Strings

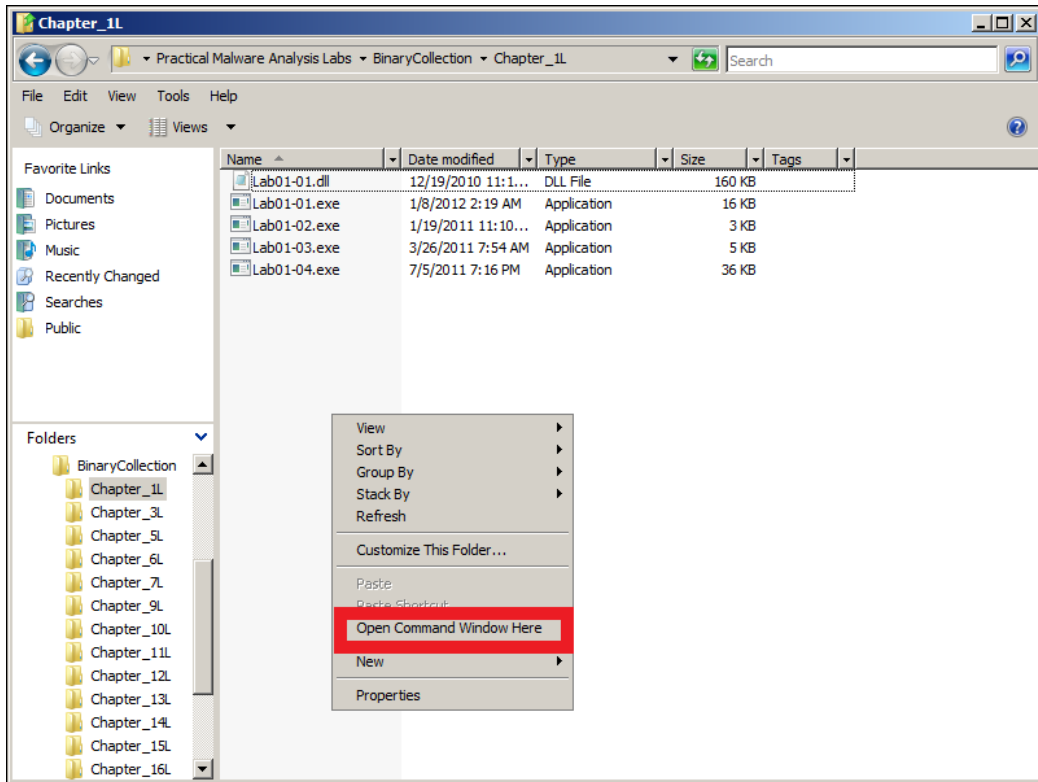
You can download Strings for Windows go here:

<http://technet.microsoft.com/en-us/sysinternals/bb897439>

Click the "Download Strings" link.

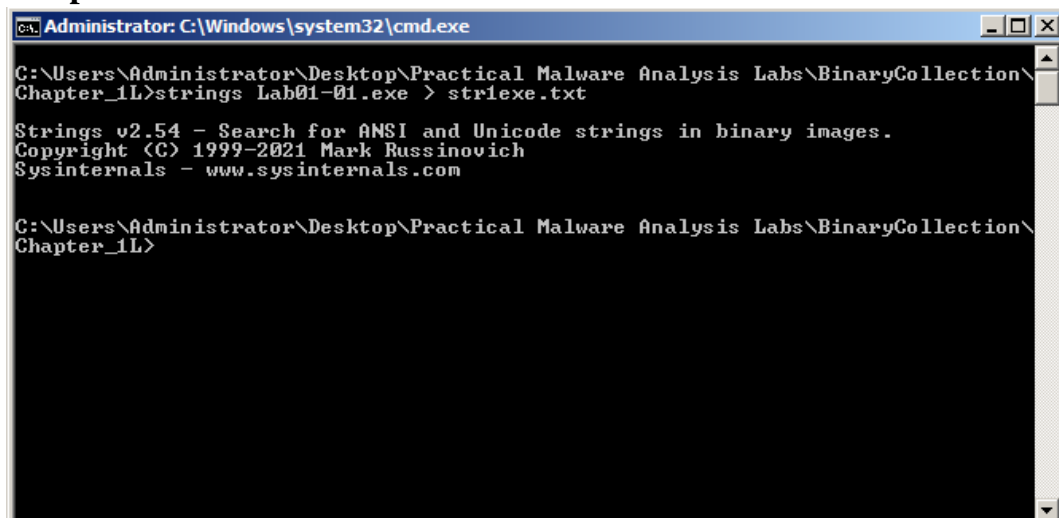
Save the Strings.zip file on your desktop. Unzip it, and copy **strings.exe** to the **C:\Windows\System32** folder.

Open a Command Prompt and use the CD command to move to the directory containing your lab files. Then collect the strings from the Lab01-01.exe file. On my machine, Right-click in the **Chapter_1L** folder, and select **Open Command Window Here**.



strings Lab01-01.exe > str1exe.txt

notepad str1exe.txt



Notice these items, as shown below:

- "FindNextFileA" and "FindFirstFileA" -- Windows functions to find files

- ".exe" -- suggesting that it will search for EXE files
- "C:\windows\system32\kerne132.dll" -- fake DLL with "kerne132" instead of "kernel32"
- "C:\Windows\System32\Kernel32.dll" -- the real Windows kernel

```

str1exe.txt - Notepad
File Edit Format View Help
FindClose
FindNextFileA
FindFirstFileA
CopyFileA
KERNEL32.dll
malloc
exit
MSVCRT.dll
_exit
_xcptFilter
__p__initenv
__getmainargs
_initterm
__setusermatherr
_adjust_fdiv
__p__commode
__p__fmode
__set_app_type
_except_handler3
_controlfp
_stricmp
kerne132.dll
kerne132.dll
.exe
C:\*
C:\windows\system32\kerne132.dll
Kernel32.
Lab01-01.dll
C:\windows\System32\Kernel32.dll
WARNING_THIS_WILL_DESTROY_YOUR_MACHINE

```

Look at the strings for **Lab01-01.dll**.
Notice these items, as shown below:

- "exec" and "sleep" -- commands that can be sent over the network to control this backdoor malware
- ".CreateProcessA" -- used to launch a program in response to the "exec" command
- "Sleep" -- used to put the backdoor to sleep in response to the "sleep" command

Turn in the image showing your analysis of Lab01-01.dll as shown below.

Below "sleep" and "hello" there is an IP address, starting with 127.

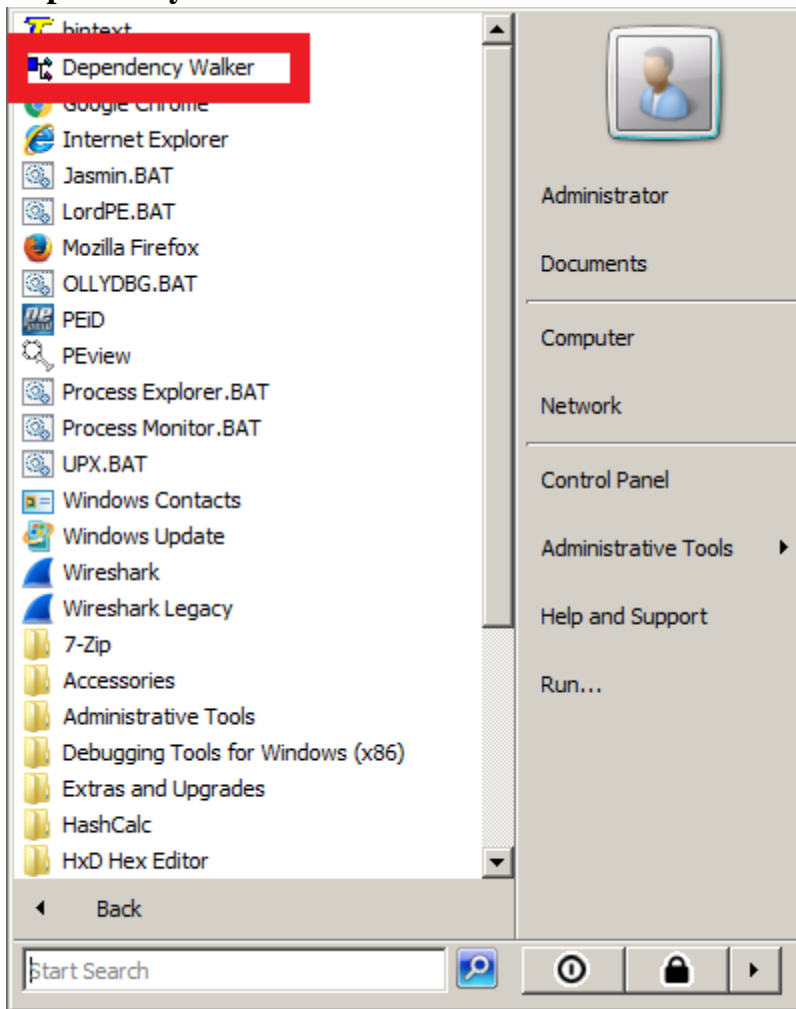
We will grade it by checking the last digits of the IP address.

```

Administrator: cmd - Shortcut (2)
NWVS
u7WPS
u&WVS
WVS
^[
%
CloseHandle
Sleep
CreateProcessA
CreateMutexA
OpenMutexA
KERNEL32.dll
WS2_32.dll
strncmp
MSVCRT.dll
free
_initterm
malloc
_adjust_fdiv
exec
sleep
hello
127.
SADFHUHF
/0I0[0h0p0
141G1[111

```

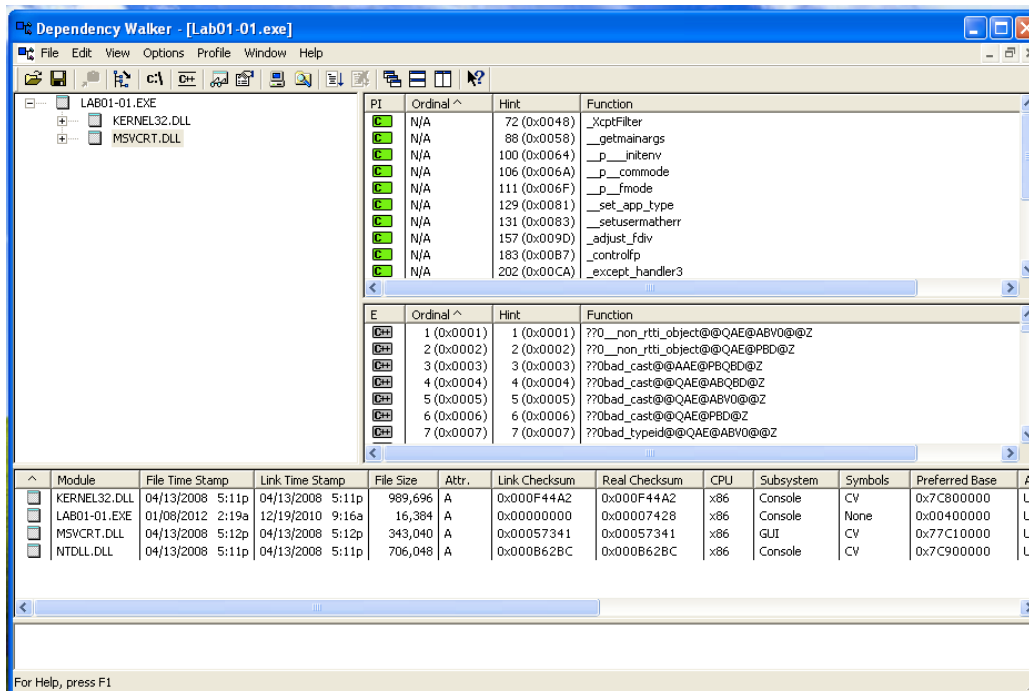
Dependency Walker



Open Lab01-01.exe in Dependency Walker.

In the left pane, click MSVCRT.DLL as shown below.

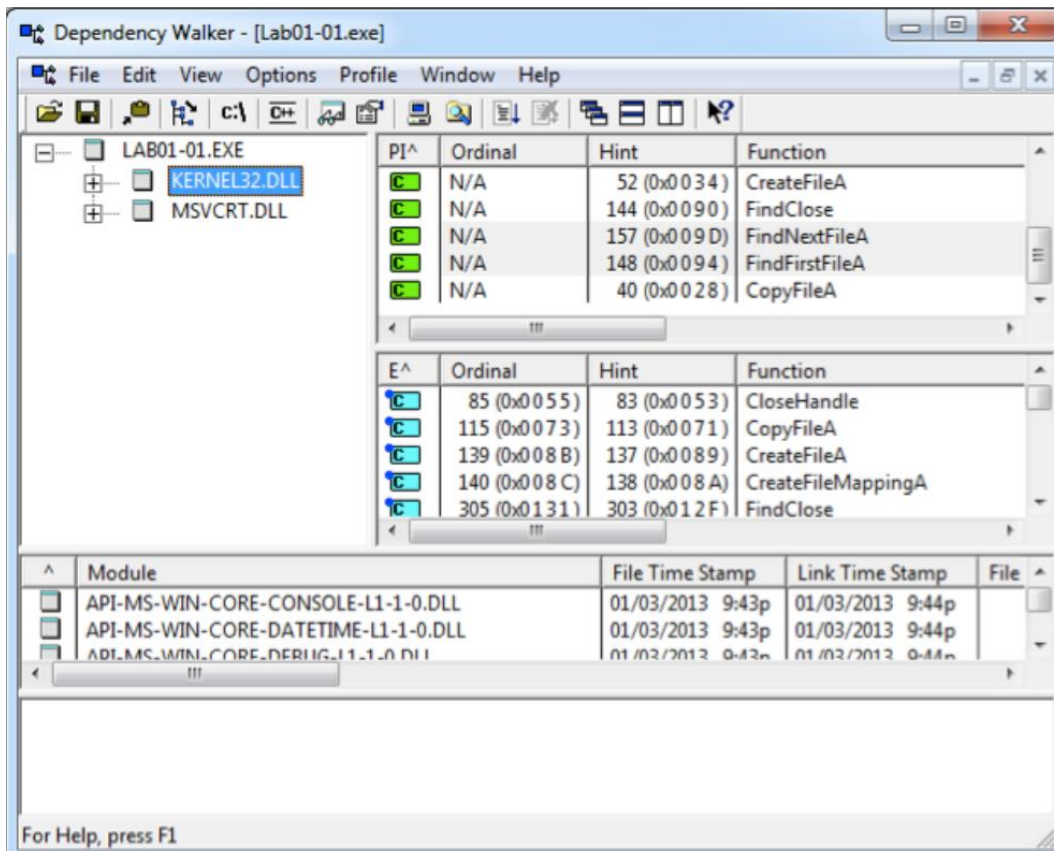
There are several imports in the upper right pane, and exports in the middle right pane. Scan through them--these are normal for any EXE



In the left pane, click KERNEL32.DLL.

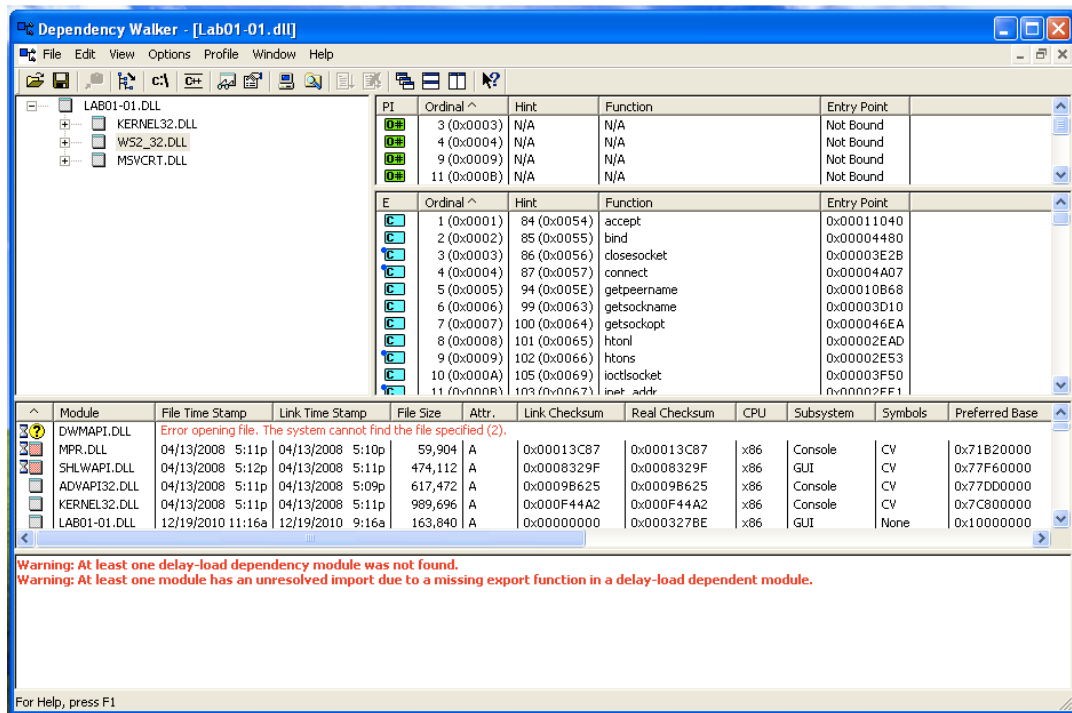
Turn in the image showing your analysis of Lab01-01.exe as shown below.

In the "PI" section (Parent Import), you should see FindNextFileA and FindFirstFileA as shown below.



Open **Lab01-01.dll** in Dependency Walker. Notice that it imports functions from "WS2_32.DLL".

WS2_32.DLL has networking functions. The right center pane shows function names that perform networking tasks, such as "bind", "closesocket", and "connect", as shown below.



b. Lab01-02.exe

HYBRID ANALYSIS

Sandbox - Quick Scans - File Collections - Resources - Request Info -

Analysis Overview

Submission name: Lab01-02.exe

Size: 3KiB

Type: process executable

Mime: application/x-dosexec

SHA256: c876a332d7dd8da331cbBee7ab7bf32752834d4b2b54eaa362674a2a48f64a6

Operating System: Windows

Last Anti-Virus Scan: 04/11/2024 22:05:32 (UTC)

Last Sandbox Report: 03/10/2024 15:43:21 (UTC)

malicious

Threat Score: 100/100

AV Detection: 90%

Labeled as: Malware

#backdoor #crypt #downloader #exploit #injector #keylogger #ransomware #riskware #rootkit #toolbar #worm

Link Twitter E-Mail

Anti-Virus Results

Refresh Required

CrowdStrike Falcon

100%

Static Analysis and ML

Last Update: 11/20/2023 06:16:04 (UTC)

View Details

Visit Vendor

GET STARTED WITH A FREE TRIAL

MetaDefender

66%

Multi Scan Analysis

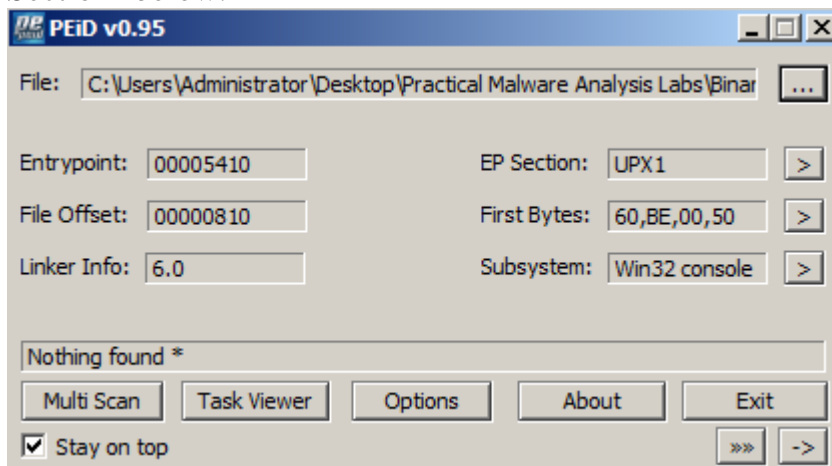
Last Update: 11/20/2023 06:16:04 (UTC)

View Details

Visit Vendor

Unpacking the File

Run PEiD on the file. It shows that the file is packed with UPX, as shown in the "EP Section" below.



Download the UPX Zip file from here:

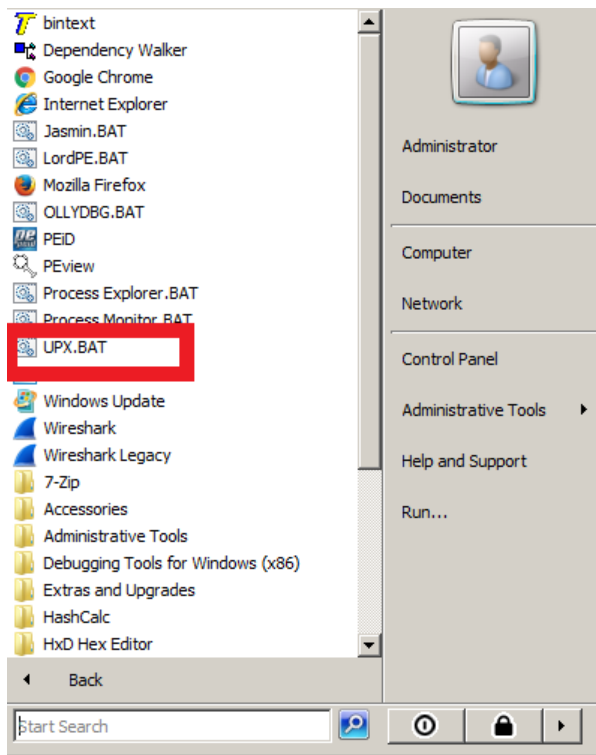
<http://upx.sourceforge.net/>

Download the **upx391w.zip** file, as shown below.



Unzip it and put upx.exe in your C:\Windows\System32 folder.

On server 2008 I have prepared it, we can open it as shown



Open a Command Prompt window and execute this command:

UPX

You see a UPX help message, as shown below:

```
C:\Windows\System32>upx

C:\Windows\System32>"C:\Program Files\upx394w\upx.exe"
      Ultimate Packer for eXecutables
      Copyright (C) 1996 - 2017
UPX 3.94w      Markus Oberhumer, Laszlo Molnar & John Reiser   May 12th 2017

Usage: upx [-123456789dlthUL] [-qvfkl] [-o file] file..

Commands:
  -1      compress faster                -9      compress better
  -d      decompress                    -l      list compressed file
  -t      test compressed file          -U      display version number
  -h      give more help                -L      display software license

Options:
  -q      be quiet                      -v      be verbose
  -oFILE  write output to 'FILE'
  -f      force compression of suspicious files
  -k      keep backup files
file..   executables to <de>compress

Type 'upx --help' for more detailed help.

UPX comes with ABSOLUTELY NO WARRANTY; for details visit https://upx.github.io

C:\Windows\System32>
```

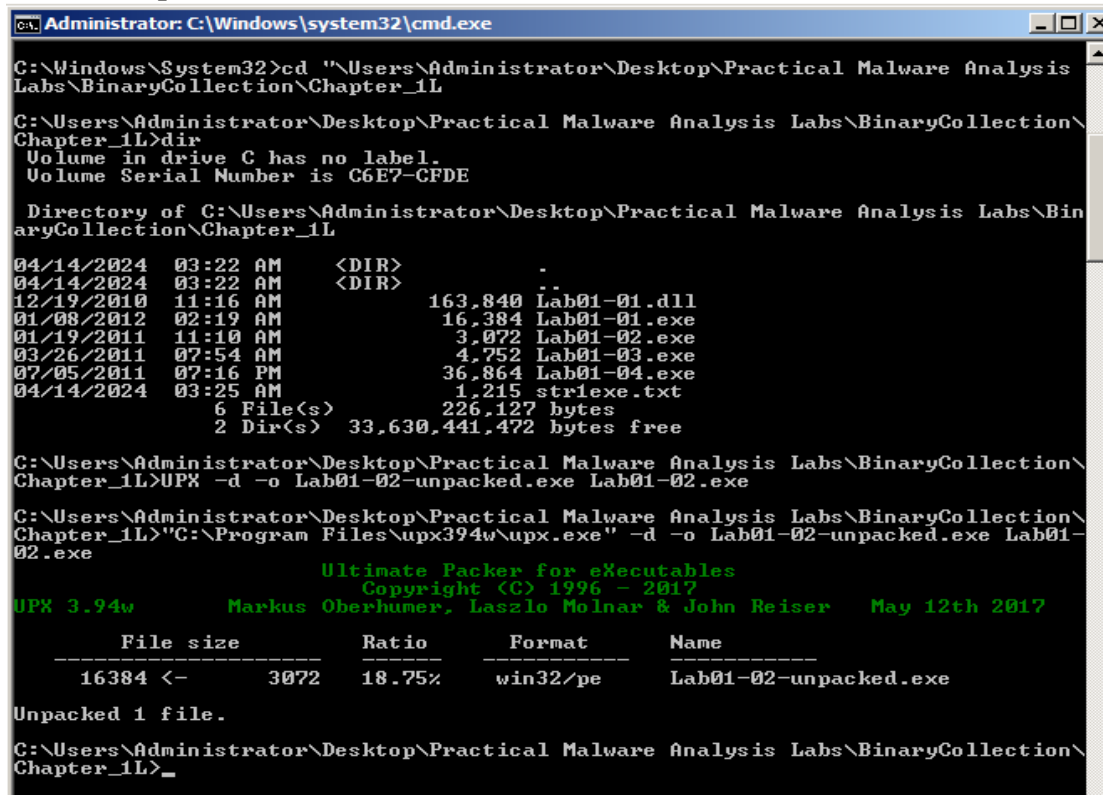
Use the CD command to move to the directory containing your malware samples. On my machine, I used this command:

cd "\\Users\\Administrator\\Desktop\\Practical Malware Analysis Labs\\BinaryCollection\\Chapter_1L"

Execute this command to unpack the file:

UPX -d -o Lab01-02-unpacked.exe Lab01-02.exe

The file unpacks, as shown below



```
C:\Windows\System32>cd "\\Users\\Administrator\\Desktop\\Practical Malware Analysis Labs\\BinaryCollection\\Chapter_1L"

C:\Users\Administrator\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L>dir
Volume in drive C has no label.
Volume Serial Number is C6E7-CFDE

Directory of C:\Users\Administrator\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L

04/14/2024  03:22 AM    <DIR>          .
04/14/2024  03:22 AM    <DIR>          ..
12/19/2010  11:16 AM             163,840 Lab01-01.dll
01/08/2012  02:19 AM             16,384 Lab01-01.exe
01/19/2011  11:10 AM              3,072 Lab01-02.exe
03/26/2011  07:54 AM              4,752 Lab01-03.exe
07/05/2011  07:16 PM             36,864 Lab01-04.exe
04/14/2024  03:25 AM              1,215 strl.exe.txt
               6 File(s)              226,127 bytes
               2 Dir(s)  33,630,441,472 bytes free

C:\Users\Administrator\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L>UPX -d -o Lab01-02-unpacked.exe Lab01-02.exe

C:\Users\Administrator\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L>"C:\Program Files\upx394w\upx.exe" -d -o Lab01-02-unpacked.exe Lab01-02.exe

Ultimate Packer for eXecutables
Copyright (C) 1996 - 2017
UPX 3.94w      Markus Oberhumer, Laszlo Molnar & John Reiser   May 12th 2017

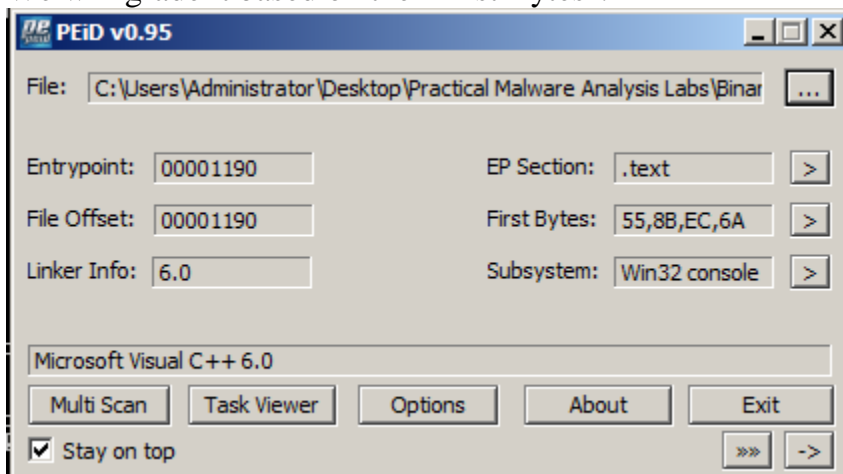
-----
File size      Ratio      Format      Name
-----
16384 <-      3072      18.75%     win32/pe     Lab01-02-unpacked.exe

Unpacked 1 file.

C:\Users\Administrator\Desktop\Practical Malware Analysis Labs\BinaryCollection\Chapter_1L>
```

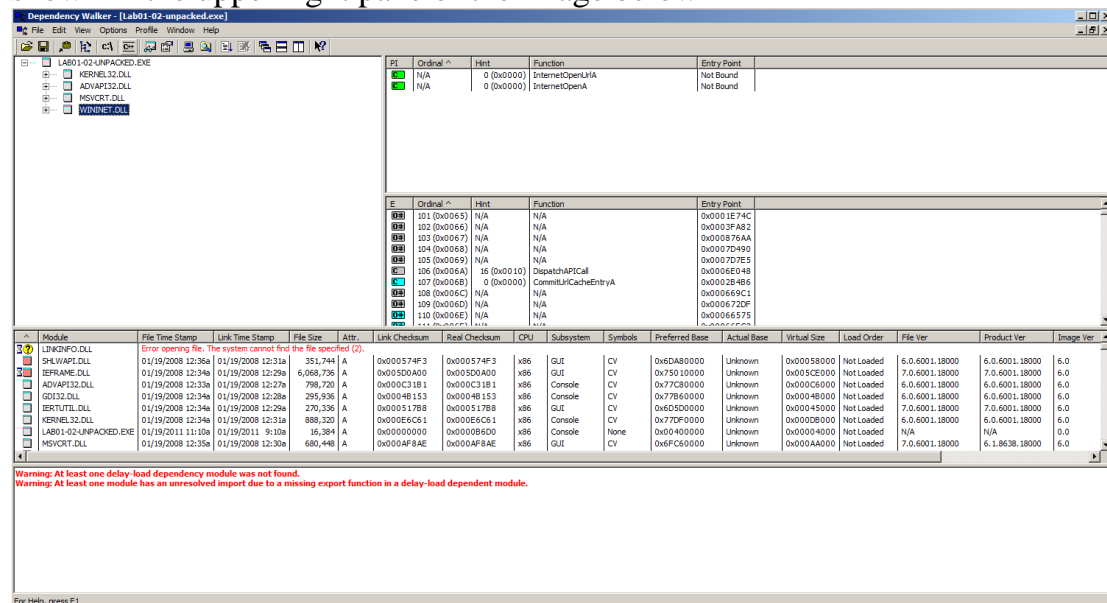
Analyze the unpacked file with PEiD. It now is recognized as a "Microsoft Visual C++ 6.0" file, as shown below.

Turn in the image showing your analysis of **Lab01-02-unpacked.exe** as shown below. We will grade it based on the "First Bytes".



Find the unpacked file's imports with Dependency Walker.

Turn in the image showing the two functions **InternetOpenUrlA** and **InternetOpenA** as shown in the upper right pane of the image below



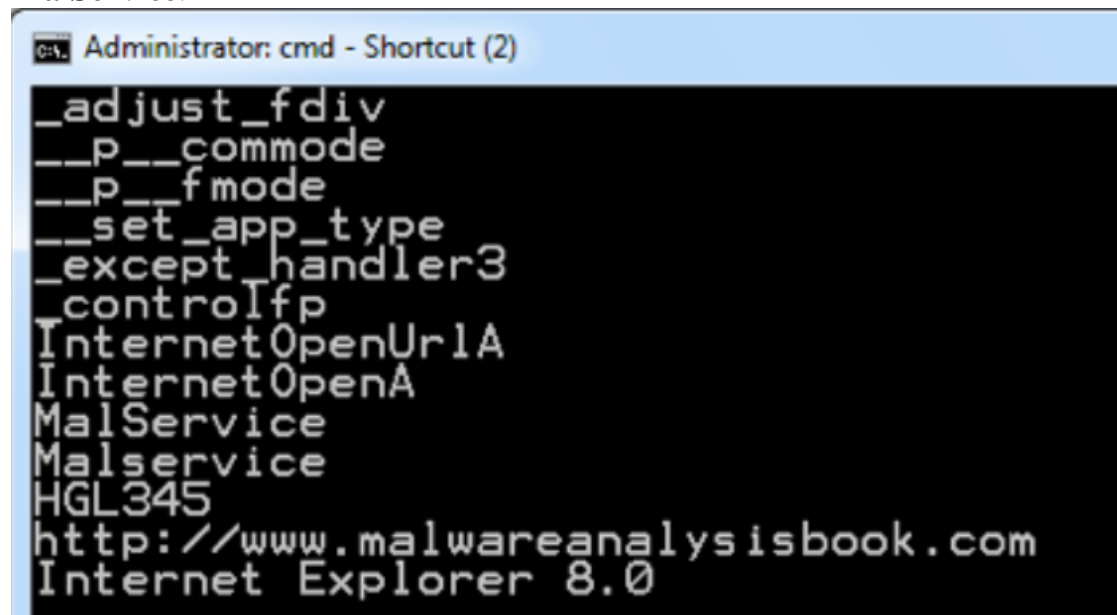
Strings

Find the strings in the unpacked file.

You should see **MalService** and **http://www.malwareanalysisbook.com** as shown below.

These suggest that infected machines will connect to

http://www.malwareanalysisbook.com and will show a running service named **MalService**.



2. Basic Dynamic Techniques

What you need: A Windows 2008 Server virtual machine with a Kali virtual machine running INetSim, which you preped in the previous project.

Purpose

You will practice the techniques in chapter 3.

This project follows **Lab 3-1** in the textbook. There are more detailed solutions in the back of the book.

Downloading Software

At the end of the previous project, you ended up with your Windows 2008 Server machine's DNS address set to your Kali machine's IP address, which means it cannot reach the Internet.

In order to download software, you need to configure a real DNS server, such as 8.8.8.8.

Setting the DNS Server to 8.8.8.8

On your Windows VM, in Control Panel, open "Network Connections". Right-click "Local Area Connection" and click **Properties**.

Double-click "Internet Protocol (TCP/IP)".

Set your DNS server to 8.8.8.8

Required Downloads

Make sure you have these items:

Lab Files from <http://practicalmalwareanalysis.com/labs/> -- download and unzip them.

PEview from <http://wjradburn.com/software/> -- download and install

Strings from <http://technet.microsoft.com/en-us/sysinternals/bb897439> -- Click "Download Strings" to get **Strings.zip**; unzip it, and copy **strings.exe** to the C:\Windows\System32 folder.

Process Monitor from <http://technet.microsoft.com/en-us/sysinternals/bb896645> -- download and unzip

Process Explorer from <http://technet.microsoft.com/en-us/sysinternals/bb896653.aspx> -- download and unzip

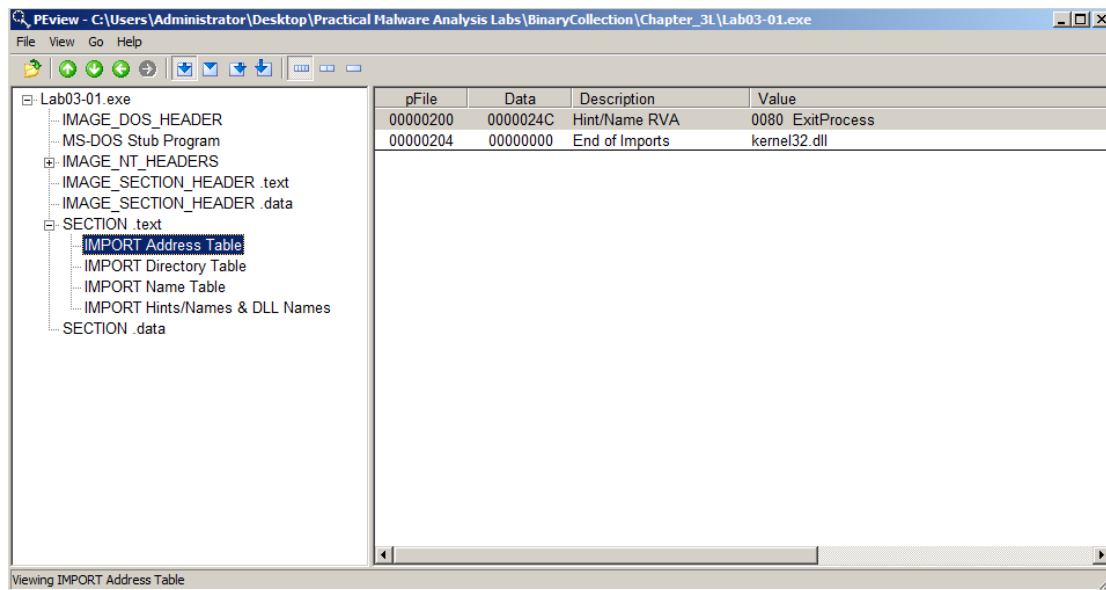
Wireshark from <http://www.wireshark.org/> -- download and install

Using PEview

Open **Lab03-01.exe** in PEview. As shown below, the only DLL imported is kernel32.dll, and the only function imported is ExitProcess. That doesn't tell us much--perhaps this malware is packed and the real imports will come at runtime.

Turn in the image showing the imports of **Lab03-01.exe** as shown below.

We will grade it by checking the Data value.



Using Strings

Examine the strings in **Lab03-01.exe** and find these items, as shown below.

SOFTWARE\Classes\http\shell\open\commandV -- A registry location

www.practicalmalwareanalysis.com -- a URL

VideoDriver

These readable strings are surprising--if the malware were packed, the strings would not be readable.

```
Administrator: C:\Windows\system32\cmd.exe
QU1M
4~v
X:a
3sg
6l*h<8
^~m~m<!<!<IM
o/o/
000U
advapi32
ntdll
user32
Jbh
ww!
1+KY
x<w
#zli
>>*K
40j
QQUP
ucj
JJJJJJ
advpack
hk?
^Pj
<2f
StubPath
SOFTWARE\Classes\http\shell\open\commandU
Software\Microsoft\Active Setup\Installed Components\
test
www.practicalmalwareanalysis.com
admin
VideoDriver
WinUMX32-
vmx32to64.exe
SOFTWARE\Microsoft\Windows\CurrentVersion\Run
Ph?
U5h
U>U
SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders
PWj
AppData
j0h
UQj
UiW
UzX_
C:\Users\Administrator\Desktop\Practical Malware Analysis Labs\BinaryCollection\
Chapter_3L>
```

Preparing for Dynamic Analysis

Dynamic analysis will help us to understand this malware better.

Here is the process detailed below:

1. Set up INetSim to simulate the Internet
2. Setting the DNS Server
3. Run Process Explorer
4. Run Wireshark
5. Run Process Monitor

1. Start INetSim

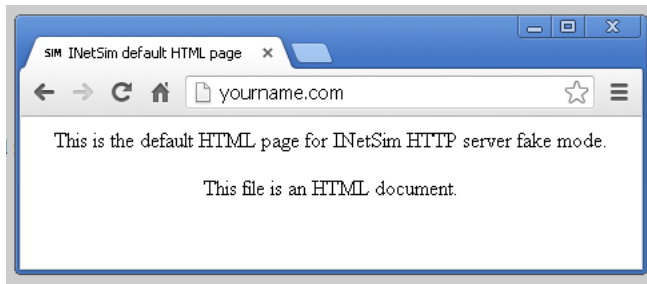
Start both the Windows and Linux VMs.

In Linux, start inetsim, as you did in the previous project.

Set the Windows DNS server to the Linux machine's IP address, as you did in the previous project.

Test it by opening a Web browser to this URL: **YOURNAME.com**

You should see the "INetSIM HTTP server" page, as shown below:

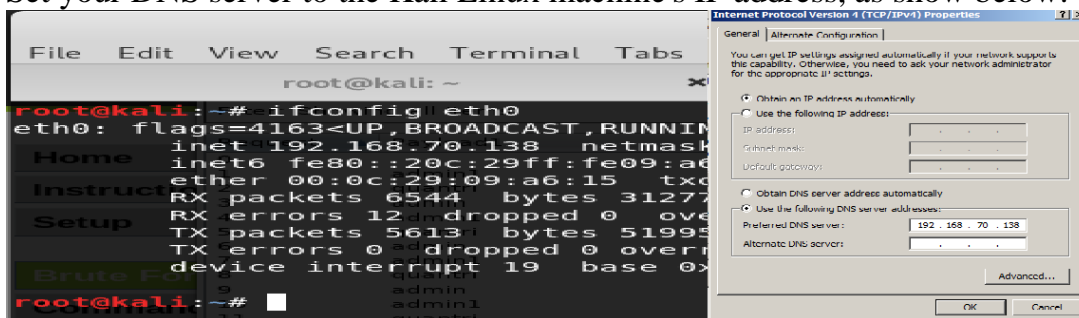


2. Setting the DNS Server

On your Windows VM, in Control Panel, open "Network Connections". Right-click "Local Area Connection" and click **Properties**.

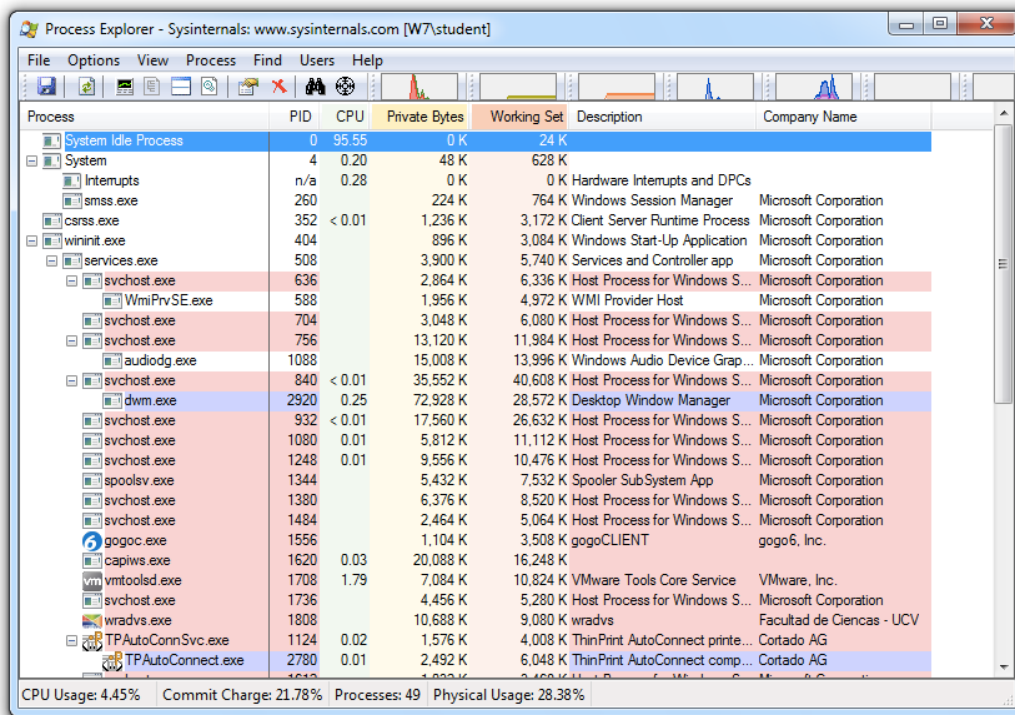
Double-click **"Internet Protocol (TCP/IP)"**.

Set your DNS server to the Kali Linux machine's IP address, as show below:



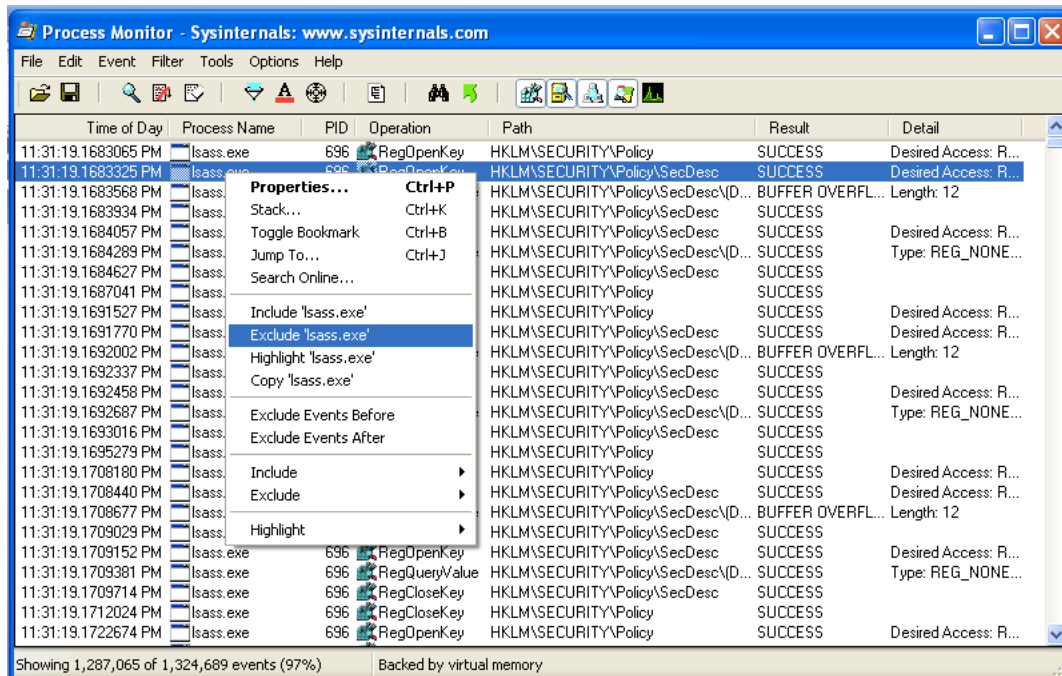
3. Run Process Explorer

Open Process Explorer, as shown below:



4. Run Wireshark

In Process Monitor, right-click the name of one of the visible processes, such as **lsass**, and click "**exclude 'lsass.exe'**", as shown below:



Wait while the event filter is applied.

Right-click a remaining process, such as "svchost.exe" and exclude it too.

Repeat the process until all current processes are hidden, as shown below. When I did it,

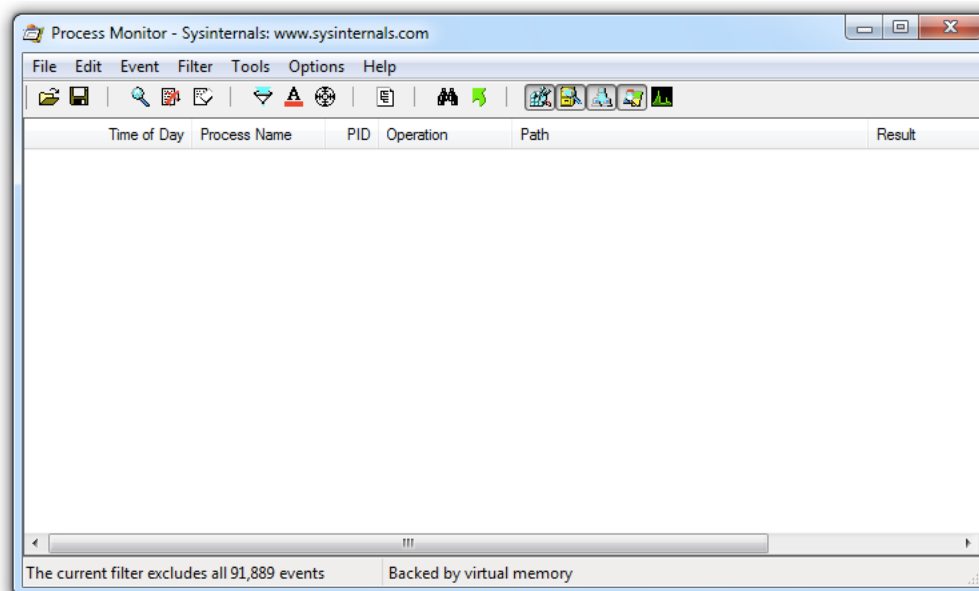
the remaining processes to exclude were csrss.exe, explorer.exe,

services.exe, vmtoolsd.exe, iexplore.exe, VMwareTray.exe, verclsid.exe, winlogon.exe,

wmiprvse.exe, wuauclt.exe, regshot.exe, spoolsv.exe, alg.exe, rundll.exe,

WMIADAP.EXE, GoogleUpdate.exe, GoogleCrashHandler.exe, chromeinstaller.exe, and

setup.exe.



Run the Lab03-01.exe File

Now double-click the Lab03-01.exe File.

Viewing the Running Malware in Process Explorer

In Process Explorer, in the top pane, find **Lab03-01.exe** and click it.

Troubleshooting

If the Lab03-01.exe process does not appear in Process Explorer, that probably means that the malware has already been run on this VM.

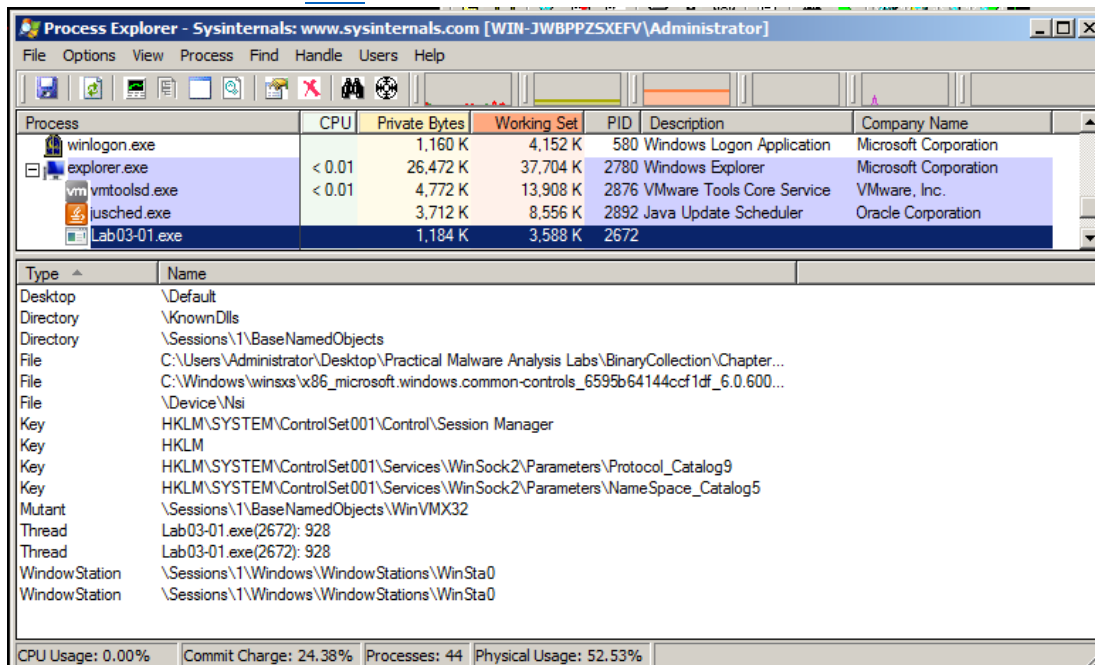
To make the malware run properly again, restart the VM, press F8, enter Safe Mode, and delete this file:

C:\Windows\System32\vmx32to64.exe

Then restart the VM in normal mode.

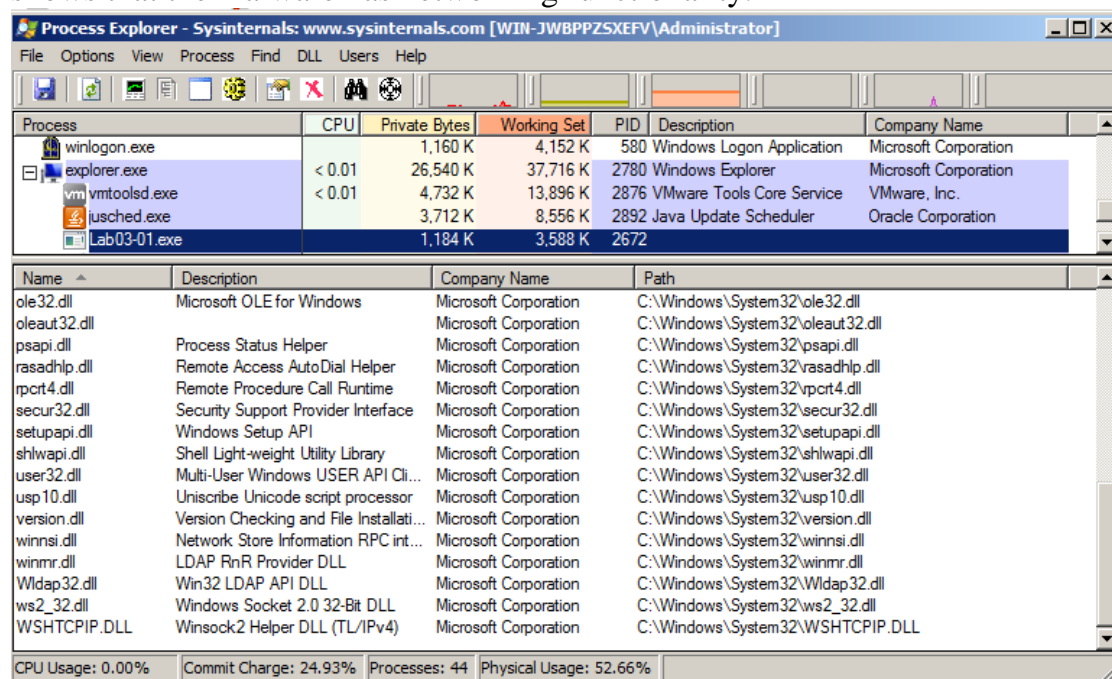
In Process Explorer, click **View, "Lower Pane View", Handles**.

You see the **WinVMX32** mutant, as highlighted below. A mutant, also called a mutex, is used for interprocess communication. A wonderful explanation of mutexes in terms of rubber chickens is [here](#).



In Process Explorer, click **View, "Lower Pane View", DLLs**.

Scroll to the bottom to find **ws2_32.dll** and **WSHTCPIP.DLL**, as shown below. This shows that the malware has networking functionality.

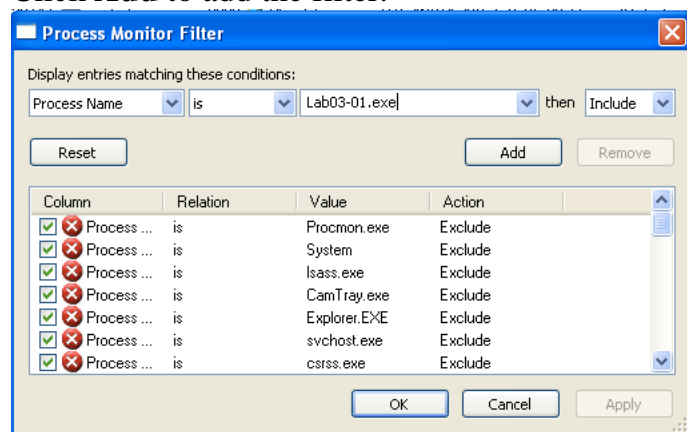


Viewing the Malicious Process's Events in Process Monitor

In Process Monitor, click the magnifying glass icon on the toolbar to stop capturing events.

In Process Monitor, click **Filter, Filter**. Enter a Filter for "**Process Name**" is **Lab03-01.exe**, **Include**, as shown below.

Click **Add** to add the filter.



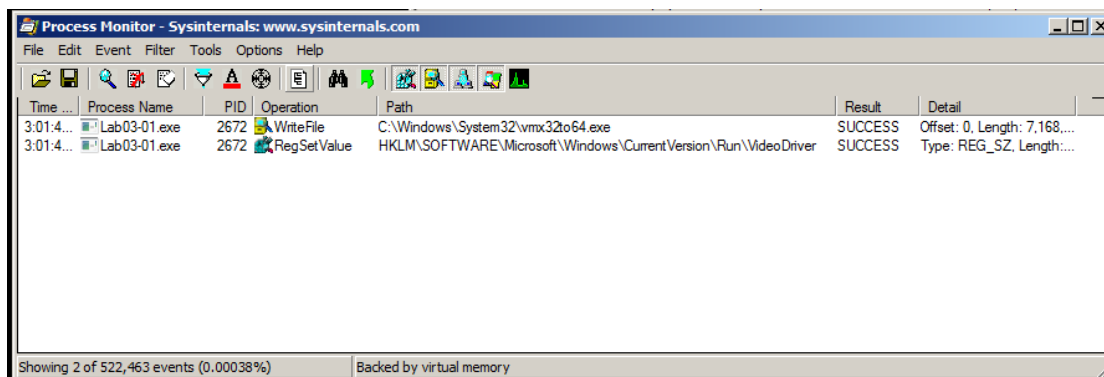
Add two more filters:

Operation of RegSetValue

Operation of WriteFile

Click **OK**.

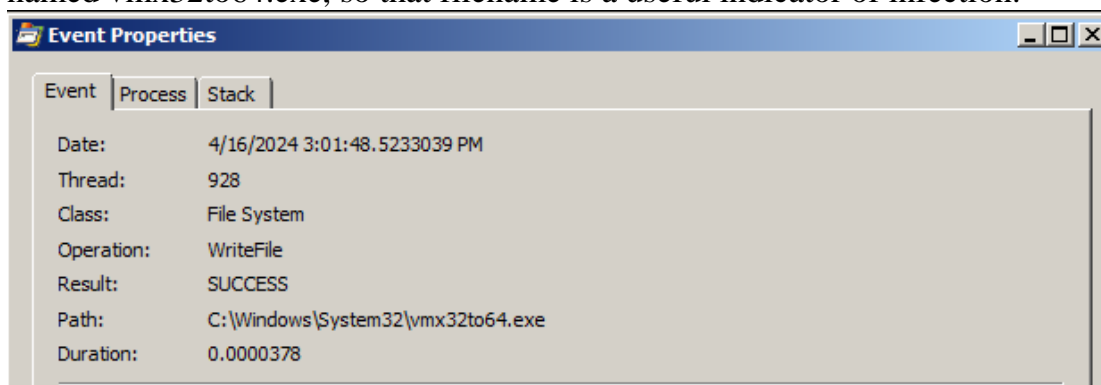
You end up the two events shown below. (Windows XP has an additional 8 events with Paths ending in "Cryptography\RNG\Seed" -- if you see those events, just ignore them.)



Only the second and third events are interesting.

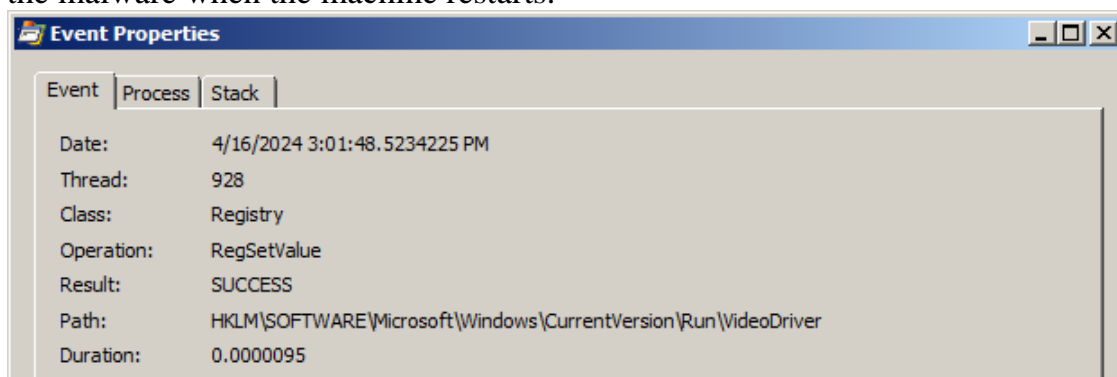
Double-click the event with a Path ending in **vmx32to64.exe**. The Properties sheet shows that this event creates a file named vmx32to64.exe, as shown below.

As explained in more detail in the book, this event has copied the malware itself to a file named vmx32to64.exe, so that filename is a useful indicator of infection.



Double-click the with a Path ending in **VideoDriver**.

This creates a new a Run key in the registry named "VideoDriver" with a value of "C:\WINDOWS\system32\vmx32to64.exe" -- this is a persistence mechanism, to relaunch the malware when the machine restarts.



Viewing INetSim Logs

On the Kali Linux machine, click in the window running inetsim.

Press **Ctrl+C**. A message appears telling you where the Report file is, as shown below:


```
root@kali: ~
File Edit View Search Terminal Tabs Help

root@kali: ~ x root@kali: ~ x [icon] [dropdown]

* daytime_13_udp - stopped (PID 1536)
* finger_79_tcp - stopped (PID 1530)
* ident_113_tcp - stopped (PID 1531)
* pop3_110_tcp - stopped (PID 1523)
* daytime_13_tcp - stopped (PID 1535)
* smtp_25_tcp - stopped (PID 1521)
* time_37_udp - stopped (PID 1534)
* smtps_465_tcp - stopped (PID 1522)
* time_37_tcp - stopped (PID 1533)
* ftps_990_tcp - stopped (PID 1526)
* https_443_tcp - stopped (PID 1520)
* http_80_tcp - stopped (PID 1519)
* syslog_514_udp - stopped (PID 1532)
* pop3s_995_tcp - stopped (PID 1524)
* dns_53_tcp_udp - stopped (PID 1518)
* ftp_21_tcp - stopped (PID 1525)
* ntp_123_udp - stopped (PID 1529)
* tftp_69_udp - stopped (PID 1527)
* irc_6667_tcp - stopped (PID 1528)
Simulation stopped.
Report written to '/var/log/inetsim/report/report.1514.txt' (310 lines)
=== INetSim main process stopped (PID 1514) ===
.
root@kali:~#
```

In the Linux machine, execute this command, replacing "report.3384.txt" with the correct name of your report file.

nano /var/log/inetsim/report/report.3384.txt

Scroll to the bottom and you should see DNS connections to

www.practicalmalwareanalysis.com, as shown below:

```
GNU nano 4.3 /var/log/inetsim/report/report.1514.txt
2024-04-16 14:47:46 HTTP connection, method: GET, URL: http://dinhmh.com/favicon.ico, file name: /var/lib/inetsim/http/fakefiles/favicon.ico
2024-04-16 14:47:46 HTTP connection, method: GET, URL: http://dinhmh.com/favicon.ico, file name: /var/lib/inetsim/http/fakefiles/favicon.ico
2024-04-16 14:47:47 DNS connection, type: A, class: IN, requested name: detectportal.firefox.com
2024-04-16 14:47:47 DNS connection, type: A, class: IN, requested name: detectportal.firefox.com
2024-04-16 14:47:47 HTTP connection, method: GET, URL: http://detectportal.firefox.com/success.txt, file name: /var/lib/inetsim/http/fakefil
2024-04-16 14:47:50 DNS connection, type: A, class: IN, requested name: detectportal.firefox.com
2024-04-16 14:47:50 DNS connection, type: A, class: IN, requested name: detectportal.firefox.com
2024-04-16 14:47:50 HTTP connection, method: GET, URL: http://detectportal.firefox.com/success.txt, file name: /var/lib/inetsim/http/fakefil
2024-04-16 14:47:53 DNS connection, type: A, class: IN, requested name: detectportal.firefox.com
2024-04-16 14:47:53 DNS connection, type: A, class: IN, requested name: detectportal.firefox.com
2024-04-16 14:47:53 HTTP connection, method: GET, URL: http://detectportal.firefox.com/success.txt, file name: /var/lib/inetsim/http/fakefil
2024-04-16 14:50:36 DNS connection, type: A, class: IN, requested name: www.wireshark.org
2024-04-16 14:52:05 DNS connection, type: A, class: IN, requested name: time.windows.com
2024-04-16 14:52:05 NTP connection, time received: 1713304159, time sent: 1713253931, difference: 50228
2024-04-16 14:52:07 DNS connection, type: PTR, class: IN, requested name: 138.70.168.192.in-addr.arpa
2024-04-16 14:53:15 DNS connection, type: PTR, class: IN, requested name: 2.70.168.192.in-addr.arpa
2024-04-16 14:56:45 DNS connection, type: PTR, class: IN, requested name: 255.70.168.192.in-addr.arpa
2024-04-16 14:57:40 DNS connection, type: ANY, class: IN, requested name: wpad
2024-04-16 14:59:20 DNS connection, type: ANY, class: IN, requested name: wpad
2024-04-16 15:01:20 DNS connection, type: ANY, class: IN, requested name: wpad
2024-04-16 15:01:48 DNS connection, type: A, class: IN, requested name: www.practicalmalwareanalysis.com
2024-04-16 15:05:42 DNS connection, type: PTR, class: IN, requested name: e.4.0.a.d.7.3.2.d.c.d.9.3.6.5.7.0.0.0.0.0.0.0.0.0.0.0.8.e.f.ip
2024-04-16 15:05:42 DNS connection, type: PTR, class: IN, requested name: 3.0.0.0.1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.2.0.f.f.ip
2024-04-16 15:05:42 DNS connection, type: PTR, class: IN, requested name: 1.70.168.192.in-addr.arpa
2024-04-16 15:05:42 DNS connection, type: PTR, class: IN, requested name: 252.0.0.224.in-addr.arpa
2024-04-16 15:05:42 Last simulated date in log file
```