Dynamic/Behavioural Analysis

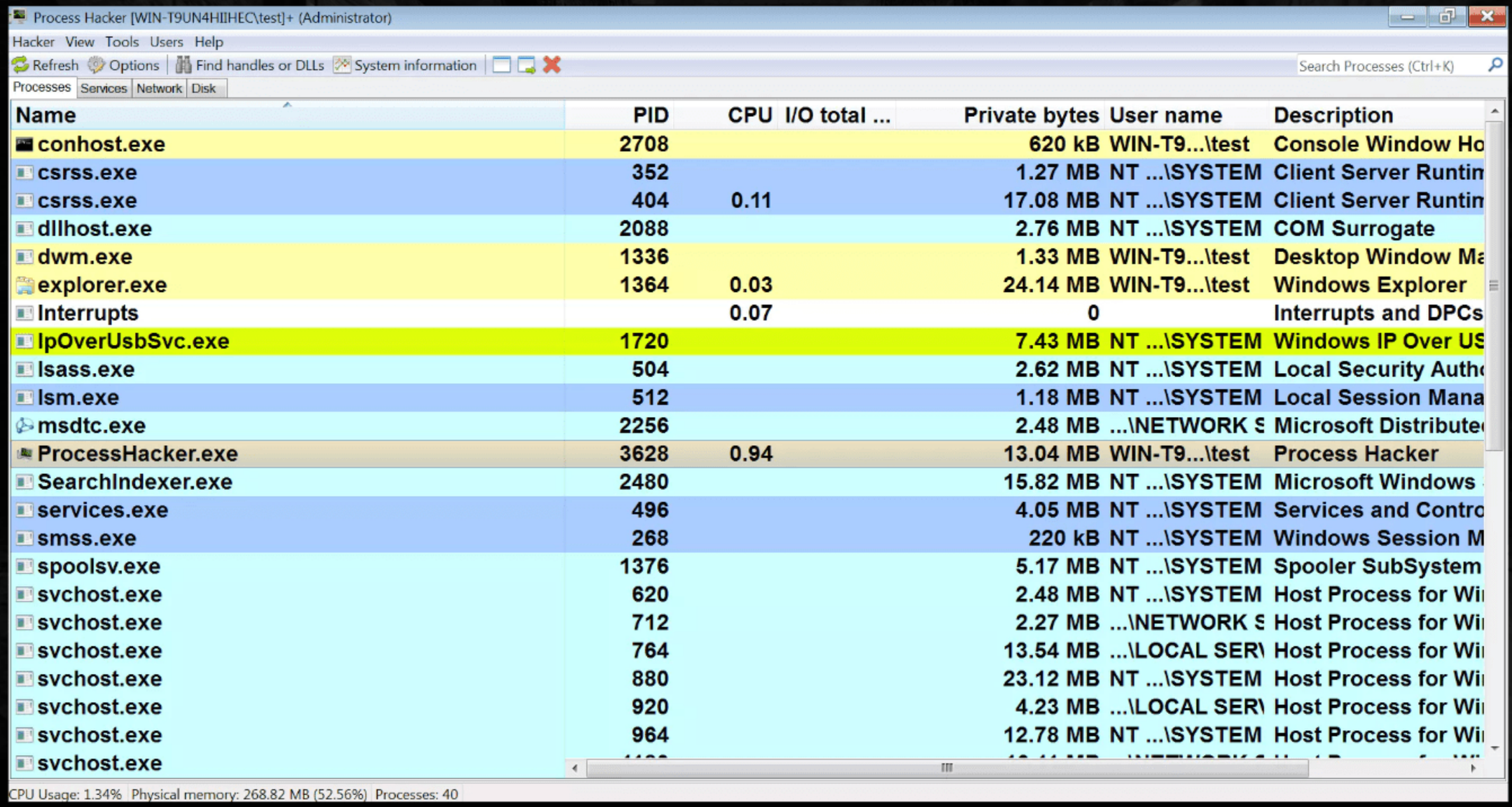
Dynamic/Behavioral Analysis Steps

- Run Monitoring tools
- Run the malware
- Stop the malware process
- Stop the monitoring tools
- Analyze the results

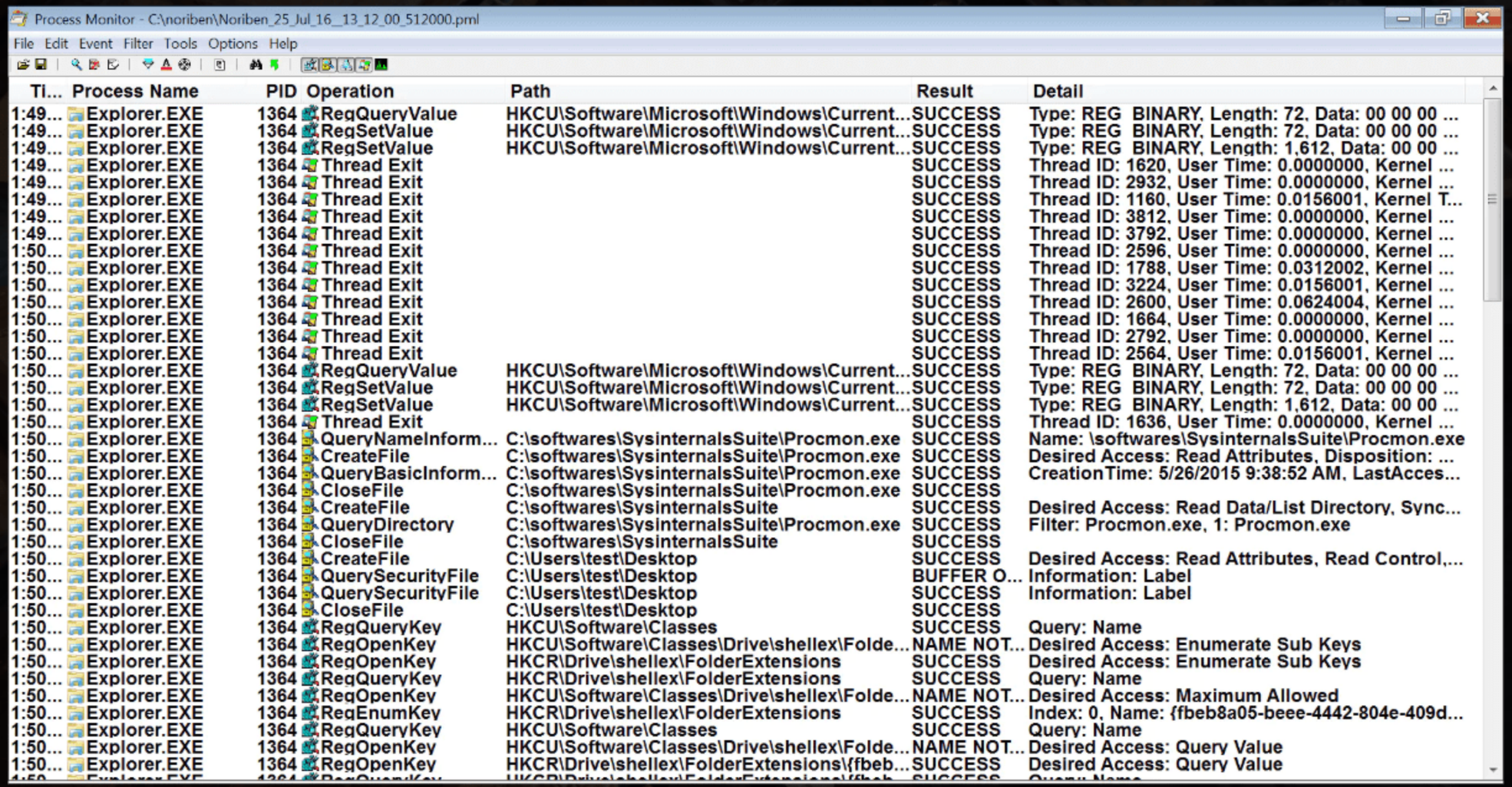
Dynamic Analysis Tools

- ***Process Hacker:*** list the process and examine process attributes
- ***Process Monitor:*** Monitor process interaction with the system
- ***Wireshark:*** Network sniffer which allows you to capture packets
- ***Noriben:*** Python script that works in conjunction with Process Monitor to collect, analyze, and report on run-time indicators of malware
- ***Fakedns:*** Gives fake DNS response, useful in redirecting network traffic to your lab system
- ***InetSim:*** Simulates Network Services

Process Hacker is an open source multi-purpose tool that helps in examining the system resources and process attributes.

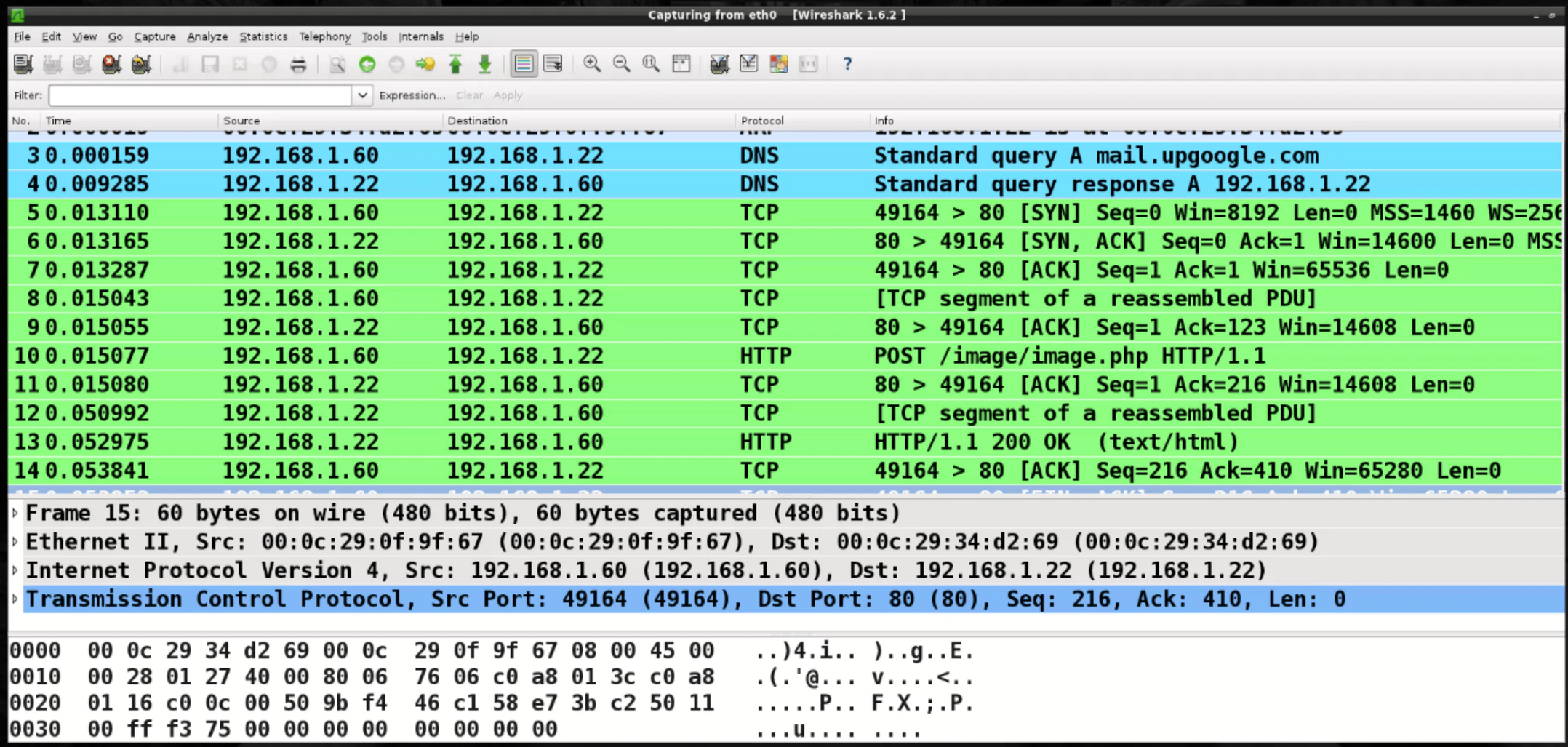


Process Monitor shows the real-time interaction of the processes with the system (like the File system, registry, process activity). It can be noisy but the filters can help you reduce the noise.



Ti...	Process Name	PID	Operation	Path	Result	Detail
1:49...	Explorer.EXE	1364	RegQueryValue	HKCU\Software\Microsoft\Windows\Current...	SUCCESS	Type: REG_BINARY, Length: 72, Data: 00 00 00 ...
1:49...	Explorer.EXE	1364	RegSetValue	HKCU\Software\Microsoft\Windows\Current...	SUCCESS	Type: REG_BINARY, Length: 72, Data: 00 00 00 ...
1:49...	Explorer.EXE	1364	RegSetValue	HKCU\Software\Microsoft\Windows\Current...	SUCCESS	Type: REG_BINARY, Length: 1,612, Data: 00 00 ...
1:49...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 1620, User Time: 0.0000000, Kernel ...
1:49...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 2932, User Time: 0.0000000, Kernel ...
1:49...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 1160, User Time: 0.0156001, Kernel T...
1:49...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 3812, User Time: 0.0000000, Kernel ...
1:49...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 3792, User Time: 0.0000000, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 2596, User Time: 0.0000000, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 1788, User Time: 0.0312002, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 3224, User Time: 0.0156001, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 2600, User Time: 0.0624004, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 1664, User Time: 0.0000000, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 2792, User Time: 0.0000000, Kernel ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 2564, User Time: 0.0156001, Kernel ...
1:50...	Explorer.EXE	1364	RegQueryValue	HKCU\Software\Microsoft\Windows\Current...	SUCCESS	Type: REG_BINARY, Length: 72, Data: 00 00 00 ...
1:50...	Explorer.EXE	1364	RegSetValue	HKCU\Software\Microsoft\Windows\Current...	SUCCESS	Type: REG_BINARY, Length: 72, Data: 00 00 00 ...
1:50...	Explorer.EXE	1364	RegSetValue	HKCU\Software\Microsoft\Windows\Current...	SUCCESS	Type: REG_BINARY, Length: 1,612, Data: 00 00 ...
1:50...	Explorer.EXE	1364	Thread Exit		SUCCESS	Thread ID: 1636, User Time: 0.0000000, Kernel ...
1:50...	Explorer.EXE	1364	QueryNameInform...	C:\softwares\SysinternalsSuite\Procmon.exe	SUCCESS	Name: \softwares\SysinternalsSuite\Procmon.exe
1:50...	Explorer.EXE	1364	CreateFile	C:\softwares\SysinternalsSuite\Procmon.exe	SUCCESS	Desired Access: Read Attributes, Disposition: ...
1:50...	Explorer.EXE	1364	QueryBasicInform...	C:\softwares\SysinternalsSuite\Procmon.exe	SUCCESS	CreationTime: 5/26/2015 9:38:52 AM, LastAcces...
1:50...	Explorer.EXE	1364	CloseFile	C:\softwares\SysinternalsSuite\Procmon.exe	SUCCESS	
1:50...	Explorer.EXE	1364	CreateFile	C:\softwares\SysinternalsSuite	SUCCESS	Desired Access: Read Data/List Directory, Sync...
1:50...	Explorer.EXE	1364	QueryDirectory	C:\softwares\SysinternalsSuite\Procmon.exe	SUCCESS	Filter: Procmon.exe, 1: Procmon.exe
1:50...	Explorer.EXE	1364	CloseFile	C:\softwares\SysinternalsSuite	SUCCESS	
1:50...	Explorer.EXE	1364	CreateFile	C:\Users\test\Desktop	SUCCESS	Desired Access: Read Attributes, Read Control,...
1:50...	Explorer.EXE	1364	QuerySecurityFile	C:\Users\test\Desktop	BUFFER O...	Information: Label
1:50...	Explorer.EXE	1364	QuerySecurityFile	C:\Users\test\Desktop	SUCCESS	Information: Label
1:50...	Explorer.EXE	1364	CloseFile	C:\Users\test\Desktop	SUCCESS	
1:50...	Explorer.EXE	1364	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
1:50...	Explorer.EXE	1364	RegOpenKey	HKCU\Software\Classes\Drive\shellex\Folde...	NAME NOT...	Desired Access: Enumerate Sub Keys
1:50...	Explorer.EXE	1364	RegOpenKey	HKCR\Drive\shellex\FolderExtensions	SUCCESS	Desired Access: Enumerate Sub Keys
1:50...	Explorer.EXE	1364	RegQueryKey	HKCR\Drive\shellex\FolderExtensions	SUCCESS	Query: Name
1:50...	Explorer.EXE	1364	RegOpenKey	HKCU\Software\Classes\Drive\shellex\Folde...	NAME NOT...	Desired Access: Maximum Allowed
1:50...	Explorer.EXE	1364	RegEnumKey	HKCR\Drive\shellex\FolderExtensions	SUCCESS	Index: 0, Name: {fbeb8a05-beee-4442-804e-409d...
1:50...	Explorer.EXE	1364	RegQueryKey	HKCU\Software\Classes	SUCCESS	Query: Name
1:50...	Explorer.EXE	1364	RegOpenKey	HKCU\Software\Classes\Drive\shellex\Folde...	NAME NOT...	Desired Access: Query Value
1:50...	Explorer.EXE	1364	RegOpenKey	HKCR\Drive\shellex\FolderExtensions\{fbeb...	SUCCESS	Desired Access: Query Value

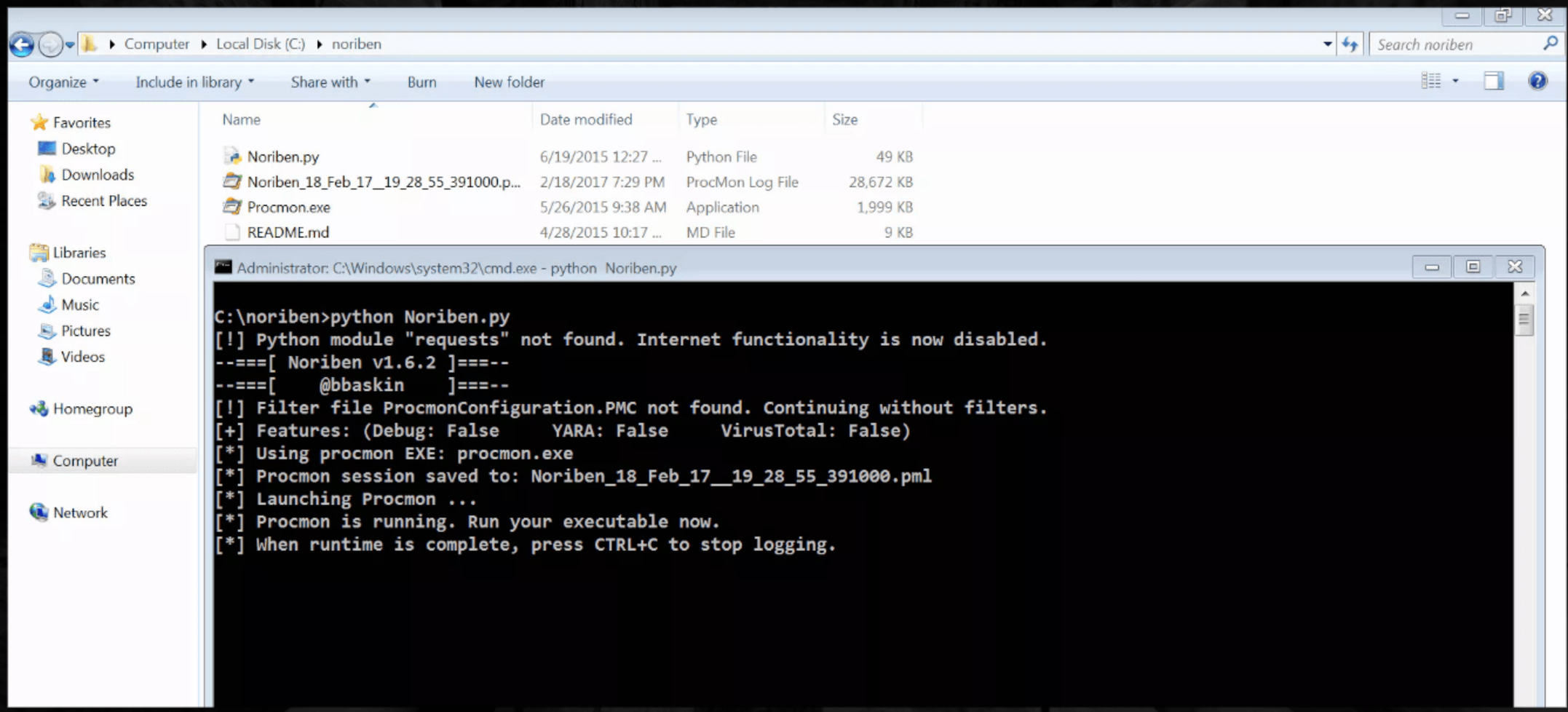
Wireshark is a packet sniffer. It allows you to capture the network traffic while you are executing the malware



Noriben

- Python script Works in conjunction with Process Monitor
- Helps in collecting, analyzing and reporting runtime indicators of malware
- Comes with pre-defined filters which help in reducing noise

Running Noriben - Copy ***procmon.exe*** and run the python script



Fakedns

- It runs DNS Service
- Resolves every domain to the IP of the machine on which it runs
- Useful in redirecting network traffic to your lab system

Below screenshot shows ***fakedns*** resolving the domains to the IP address of the system on which it is running

```
File  Edit  Tabs  Help
remnux@remnux:~$ myip
192.168.1.22
remnux@remnux:~$ fakedns
pyminifakeDNS:: dom.query. 60 IN A 192.168.1.22
Respuesta: 2.2.2.4.in-addr.arpa. -> 192.168.1.22
Respuesta: yahoo.com. -> 192.168.1.22
Respuesta: yahoo.com. -> 192.168.1.22
Respuesta: 2.2.2.4.in-addr.arpa. -> 192.168.1.22
Respuesta: google.com. -> 192.168.1.22
Respuesta: google.com. -> 192.168.1.22
Respuesta: 2.2.2.4.in-addr.arpa. -> 192.168.1.22
Respuesta: cysinfo.com. -> 192.168.1.22
Respuesta: cysinfo.com. -> 192.168.1.22
```

INetSim

- Simulates common internet services in a lab environment
- Useful in redirecting all the communication to your lab system
- Easy to configure and the configuration file is ***inetsim.conf***

Below screenshot shows *InetSim* simulating all the internet services

```
File  Edit  Jabs  Help
remnux@remnux:~$ inetsim
INetSim 1.2.4 (2013-08-15) by Matthias Eckert & Thomas Hungenberg
Using log directory:      /var/log/inetsim/
Using data directory:     /var/lib/inetsim/
Using report directory:   /var/log/inetsim/report/
Using configuration file: /etc/inetsim/inetsim.conf
Parsing configuration file.
Configuration file parsed successfully.
=== INetSim main process started (PID 3000) ===
Session ID:     3000
Listening on:   192.168.1.22
Real Date/Time: 2016-07-27 00:36:25
Fake Date/Time: 2016-07-27 00:36:25 (Delta: 0 seconds)
 Forking services...
  * ident_113_tcp - started (PID 3015)
  * irc_6667_tcp - started (PID 3012)
  * finger_79_tcp - started (PID 3014)
  * quotd_17_udp - started (PID 3026)
  * daytime_13_tcp - started (PID 3019)
  * echo_7_udp - started (PID 3022)
  * discard_9_udp - started (PID 3024)
  * syslog_514_udp - started (PID 3016)
  * chargen_19_tcp - started (PID 3027)
  * chargen_19_udp - started (PID 3028)
  * echo_7_tcp - started (PID 3021)
  * time_37_udp - started (PID 3018)
  * time_37_tcp - started (PID 3017)
  * daytime_13_udp - started (PID 3020)
```

Below screenshot shows the https traffic intercepted using *INetSim*

```
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: POST /webmail.php HTTP/1.1
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: Connection: Keep-Alive
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: Content-Type: application/x-www-form-
urlencoded; Charset=UTF-8
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: Accept: */*
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: User-Agent: (RF) : <exe> :(PC-Name: WIN-
T9UN4HIIHEC : Username: Administrator ; AV: NoAV)
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: Content-Length: 80
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: Host: webmail.duia.in
[2016-11-14 12:04:27] [192.168.1.60:49166] recv: <(POSTDATA)>
[2016-11-14 12:04:27] [192.168.1.60:49166] info: POST data stored to: /usr/share/inetsim/
data/http/postdata/6ba1e7ed0791c196aac167563c26b9f6f92a1e7f

POSTDATA: action=getfiles&username=000C290F9F67-WIN-T9UN4HIIHEC-Administrator&filename=exe
```

https://cysinfo.com/malware-actors-using-nic-cyber-security-themed-spear-phishing-target-indian-government-organizations/

Demo 1 - Analysis of SpyBot

Lab 2- The case of Remcos RAT (contd.)

This is the continuation of Lab 1, analyze the sample host.exe and answer the below questions

- What is the name of the file dropped by the malware?
- Where does it drop this file?
- Does the malware copies itself or does it drop a different component?
- How does malware execute this dropped file?
- What is the name of the C2 domain?
- What is the port on which malware is communicating?
- Do you see anything suspicious in the network traffic?
- How is Malware persisting on the system?