

Malware Traffic Analysis Using Wireshark

Sandbox

A sandbox is an isolated testing environment that enables users to run programs or execute files without affecting the application, system or platform on which they run. Software developers use sandboxes to test new programming code. Cybersecurity professionals use sandboxes to test potentially malicious software.

In general, a sandbox is used to test suspicious programs that may contain viruses or other malware, without allowing the software to harm the host devices.

Without sandboxing, an application or other system process could have unlimited access to all the user data and system resources on a network

A sandbox can also enable a mirrored production environment that an external developer can use to develop an app that uses a web service from the sandbox.

Windows Sandbox

Windows Sandbox provides a lightweight desktop environment to safely run applications in isolation. Software installed inside the Windows Sandbox environment remains "sandboxed" and runs separately from the host machine.

A sandbox is temporary. When it's closed, all the software and files and the state are deleted. You get a brand-new instance of the sandbox every time you open the application.

Prerequisites

- Windows 10 Pro or Enterprise build 18305 or later (Windows Sandbox is currently not supported on Home SKUs)
- AMD64 architecture
- Virtualization capabilities enabled in BIOS
- At least 4 GB of RAM (8 GB recommended)
- At least 1 GB of free disk space (SSD recommended)
- At least two CPU cores (four cores with hyperthreading recommended)

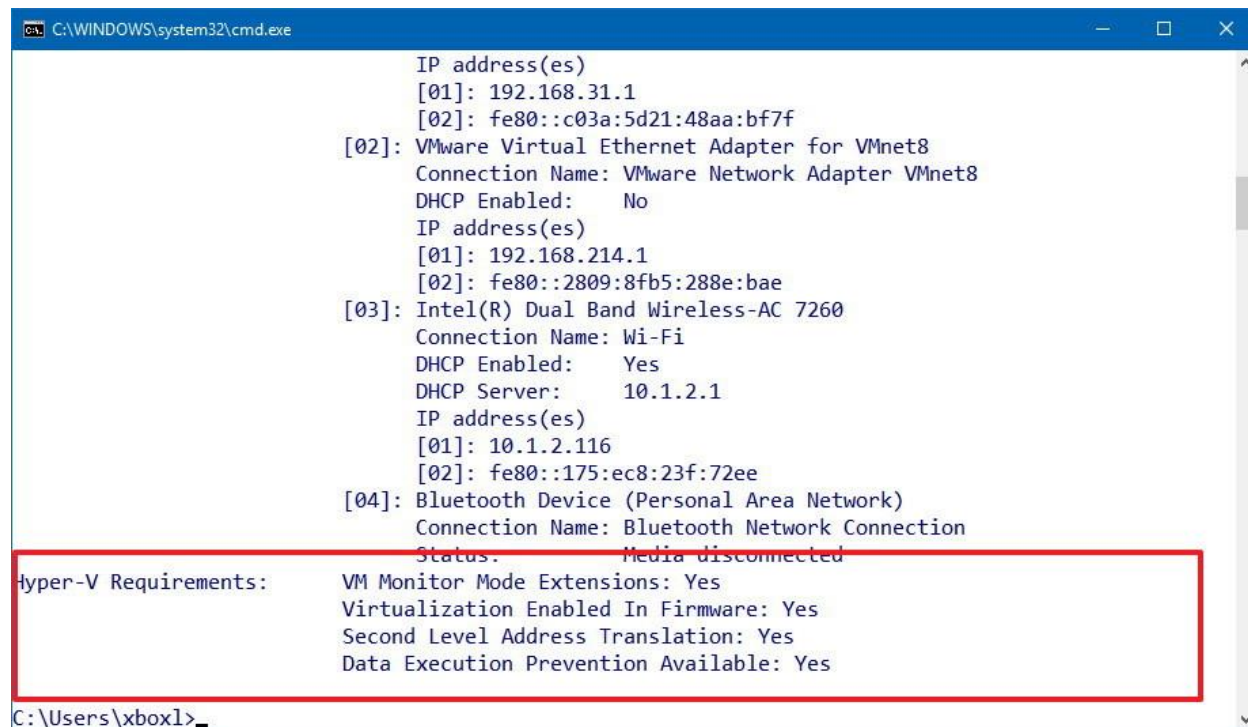
Enabling virtualization

Before you can enable Windows Sandbox, you must make sure that your device supports hardware virtualization and the feature is enabled in the Basic Input/Output System (BIOS) or Unified Extensible Firmware Interface (UEFI) firmware.

To check if your device supports virtualization, use these steps:

1. Open **Start**.
2. Search for *Command Prompt* and click the top result.
3. Type the following command and press *Enter*:
systeminfo.exe

Under the "Hyper-V Requirements" section, if "Virtualization Enabled in Firmware" reads **Yes**, then you can use Windows Sandbox.



```
C:\WINDOWS\system32\cmd.exe

IP address(es)
[01]: 192.168.31.1
[02]: fe80::c03a:5d21:48aa:bf7f
[02]: VMware Virtual Ethernet Adapter for VMnet8
Connection Name: VMware Network Adapter VMnet8
DHCP Enabled: No
IP address(es)
[01]: 192.168.214.1
[02]: fe80::2809:8fb5:288e:bae
[03]: Intel(R) Dual Band Wireless-AC 7260
Connection Name: Wi-Fi
DHCP Enabled: Yes
DHCP Server: 10.1.2.1
IP address(es)
[01]: 10.1.2.116
[02]: fe80::175:ec8:23f:72ee
[04]: Bluetooth Device (Personal Area Network)
Connection Name: Bluetooth Network Connection
Status: Media disconnected

Hyper-V Requirements:      VM Monitor Mode Extensions: Yes
                          Virtualization Enabled In Firmware: Yes
                          Second Level Address Translation: Yes
                          Data Execution Prevention Available: Yes

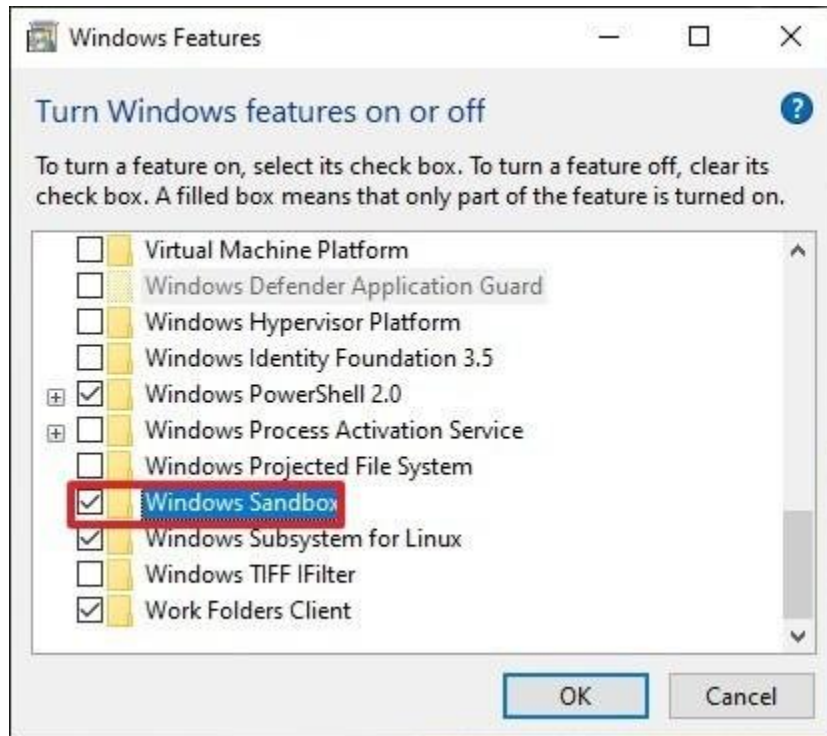
C:\Users\xbox1>
```

If virtualization isn't enabled, you'll need to start your device in its BIOS or UEFI firmware and enable the feature. This process typically requires hitting one of the function keys (F1, F2, F3, F10, or F12), the ESC, or Delete key as soon as you start your computer. However, these settings will vary by manufacturer, and even by computer model. So make sure to check your device manufacturer's support website for more specific instructions.

Enabling Windows Sandbox

To enable Windows Sandbox on Windows 10 version 1903 or later, use these steps:

1. Open **Start**.
2. Search for **Turn Windows features on or off**, and click the top result to open the Windows Features experience.
3. Check the **Windows Sandbox** option.



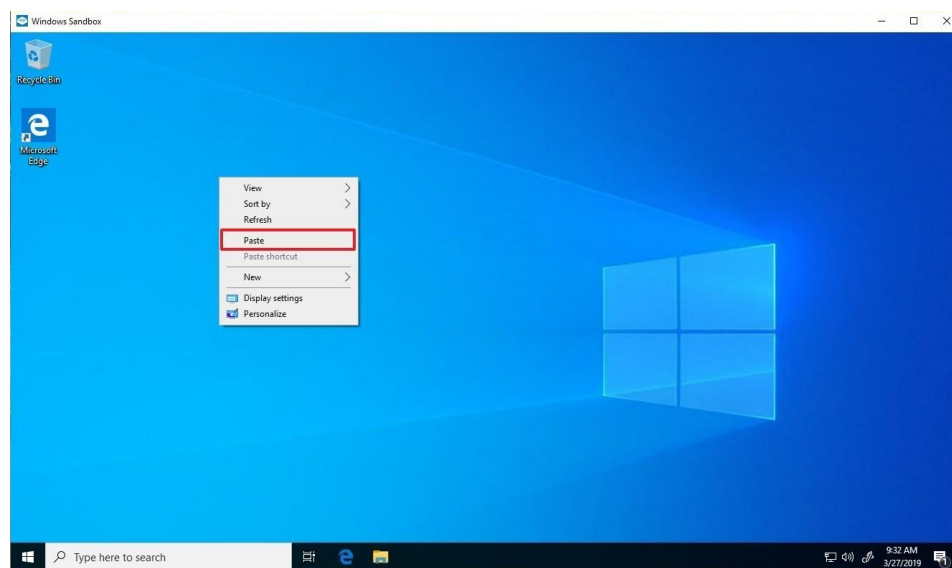
4. Click the **OK** button.
5. Click the **Restart now** button.

After completing the steps, you can start using the new virtualization layer to temporarily install and test untrusted applications.

How to use Windows Sandbox on Windows 10

The process to get started using Windows Sandbox is easy; just follow these steps:

1. Open **Start**.
2. Search for **Windows Sandbox**, right-click the top result, and select the **Run as administrator** option.
3. On your physical device, right-click the app installer that you want to test, and select the **Copy** option.
4. Inside the Windows Sandbox experience, right-click on the desktop and select the **Paste** option to transfer the executable.



Quick tip: Although you can't drag and drop from your main installation into Windows Sandbox to transfer files, you can also download the application files using Microsoft Edge.

5. Double-click the installer (.exe, .msi, etc) to begin the installation.
6. Continue with the on-screen directions to complete the installation.

Once you complete the steps, you can start using the untrusted app normally just like any other application. In addition, you can use the **Ctrl + Alt + Break** (or Pause) keyboard shortcut to enter and exit Windows Sandbox in full-screen mode. If you use high contrast mode, inside the virtualization experience, you can use the **Shift + Alt + PrintScreen** keyboard shortcut to enable high contrast.

After you're done testing the application, click the **X** button, on the top-right corner, and click the **OK** button to close Sandbox. As you terminate the experience, the virtual machine and its content will be erased from your device permanently without affecting your device installation of Windows 10.

Malware Analysis

Link for Sample PCAP File: (Password for zip file - *infected*)

WARNING: The password protected zip files contain real malware

<https://github.com/bandarawmd/Malware-traffic-Analysis-using-Wireshark/blob/master/Sample-PCAP-File.pcap.zip>

TARGETED QUESTIONS AND ANSWERS:

- On the infected Windows host, What is the;
 - a. IP address? *10.5.28.229*
 - b. Host name? *Cat-Bomb-W7-PC*
 - c. User account name? *phillip.ghent*
 - d. MAC address? *00-08-02-1C-47-AE*
- What is the other Windows client;
 - e. Host name? *CAT-BOMB-W10-PC*
 - f. User account name? *timothy.sizemore*
- What is the infected user's email password? *gh3ntf@st*
- What are the SHA256 hashes for these files?

cursor.png

4e76d73f3b303e481036ada80c2eeba8db2f306cbc9323748560843c80b2fed1

imgpaper.png

934c84524389ecfb3b1dfcb28f9697a2b52ea0ebcaa510469f0d2d9086bcc79a

ANSWERS EXPLAINED:

1) When Trickbot successfully infects a Windows host, it sends an HTTP

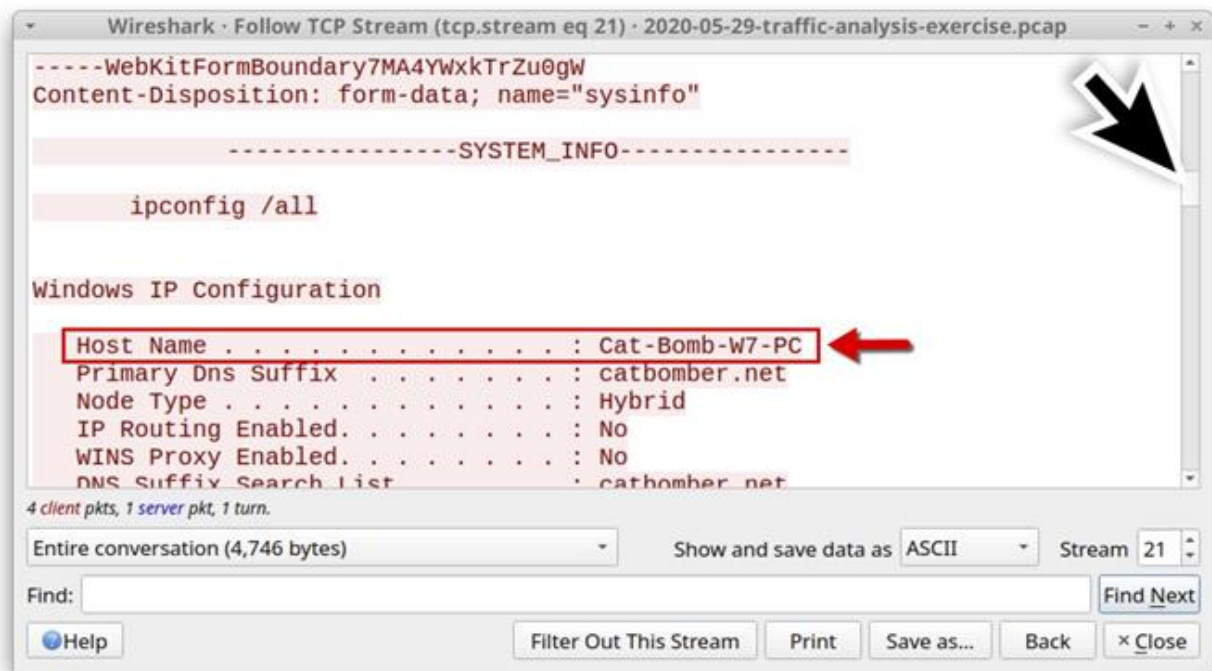
POST request with the system data, usually over TCP port 8082. The URL ends with /90, so use the following Wireshark filter to find that URL and follow the TCP stream:

`http.request.uri contains "/90"`

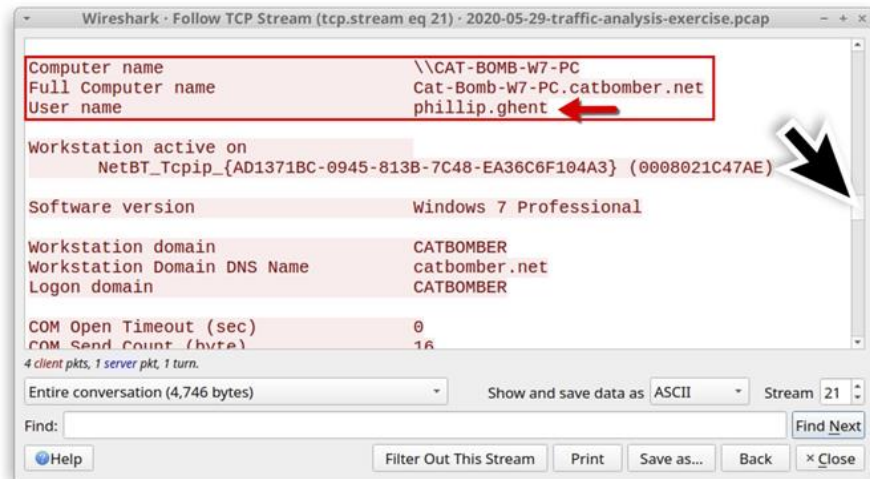
This should return two URLs in your Wireshark column display, one for the infected Windows client (CAT-BOMB-W7-PC), and one for the domain controller (CATBOMBER-DC).

http.request.uri contains "/90"						
Time	Src	Dst	port	Host	Info	
2020-05-28 18:09...	10.5.28.229	203.176.135.102	8082	203.176.135.102:8082	POST	/yas33/CAT-BOMB-W7-PC_W
2020-05-28 18:17...	10.5.28.8	203.176.135.102	8082	203.176.135.102:8082	POST	/jim734/CATBOMBER-DC_W6

Shown above: Filter results looking for the "/90" URLs in the pcap



Shown above: Scroll down a bit in the TCP stream window to find the host name



Shown above: Scroll down further to find the infected host's user account name

2) In the replies to these "/90" URLs, you'll also find a section named "LOCAL_MACHINE_DATA" in both the URL for the client and the DC. This should include all hosts found on the network, including other clients and the DC. I've only found this in cases where the infected client attempts to infect the DC.

Just scroll down near the end of the TCP stream we were looking at to find this info.



Shown above: Local_Machine_Data section with information on another Windows client in the catbomber.net internal network.

3) HTTP POST requests that end in "/81" is where we find password data exfiltrated from an infected Windows host. Use the following Wireshark filters to find email passwords:

http.request.uri contains "/81" and ip contains mail

http.request.uri contains "/81" and ip contains smtp

http.request.uri contains "/81" and ip contains mail						
Time	Src	Dst	port	Host	Info	
2020-05-28 18:04...	10.5.28.229	36.89.106.69	80	36.89.106.69	POST /yas33/CAT-BOMB-W7-PC_w617601.107	

Shown above: Finding a URL ending in "/81" for password exfiltration that contains the string "mail" in the response text.

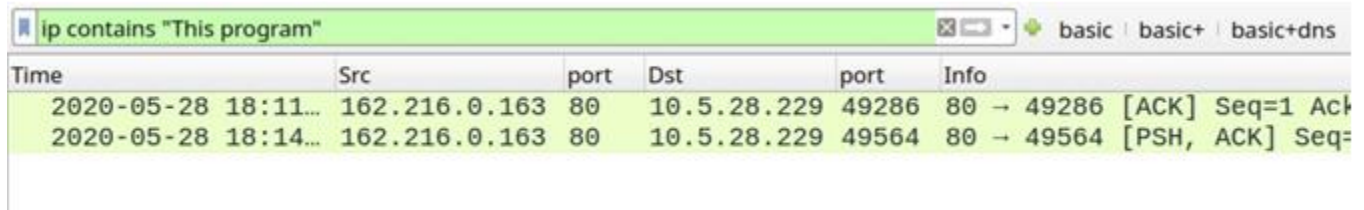


Shown above: Following the TCP stream and finding the password used for phillip.ghent's email at catbomber.net.

4) We can quickly filter on traffic to see if there's any Windows executable (EXE) files pass in the clear (not as encoded or encrypted data) using the following filter:

IP contains "This program"

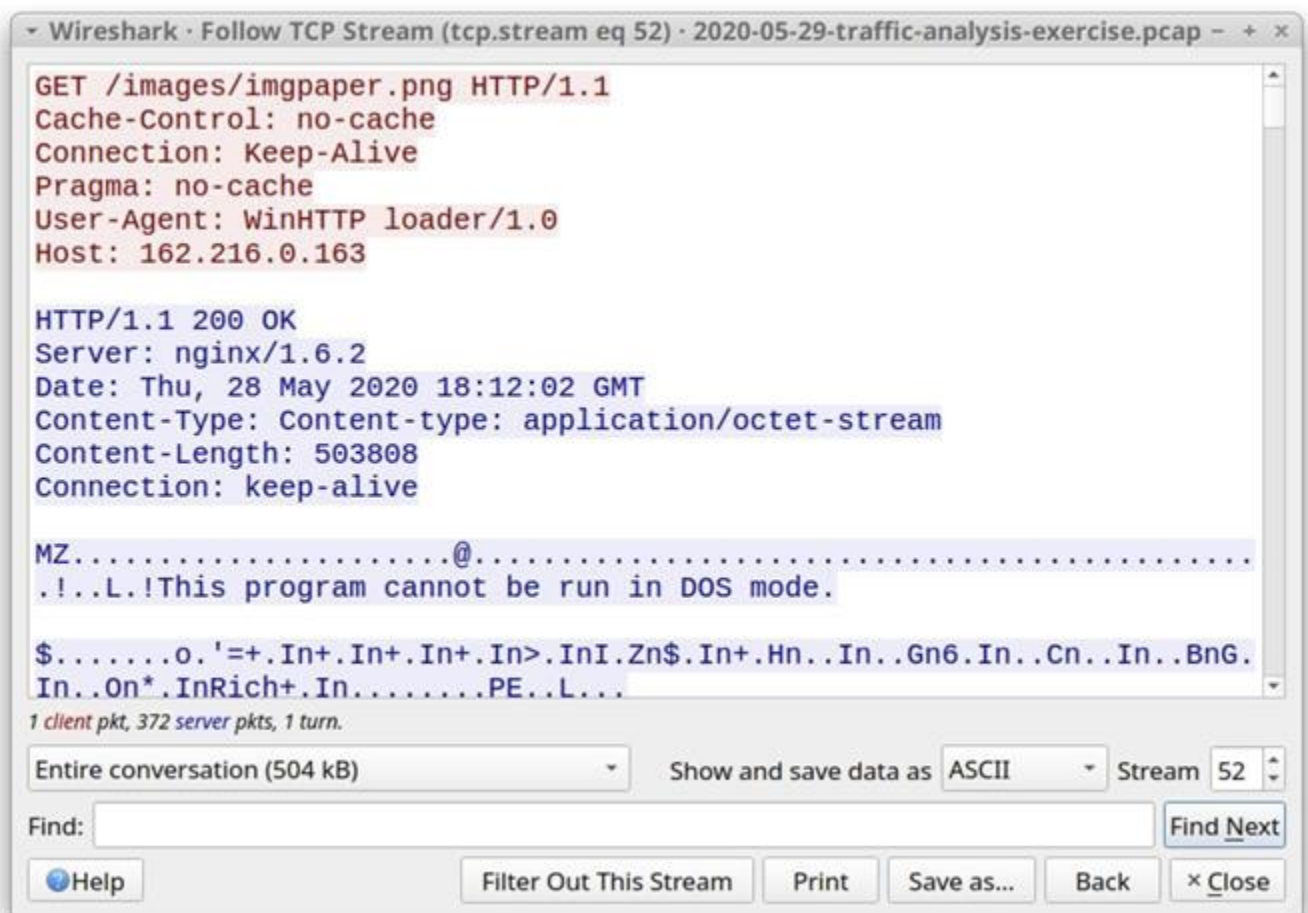
This doesn't work every single time, but it works for most EXE files. It should return two frames in your column display. Follow each of these TCP streams



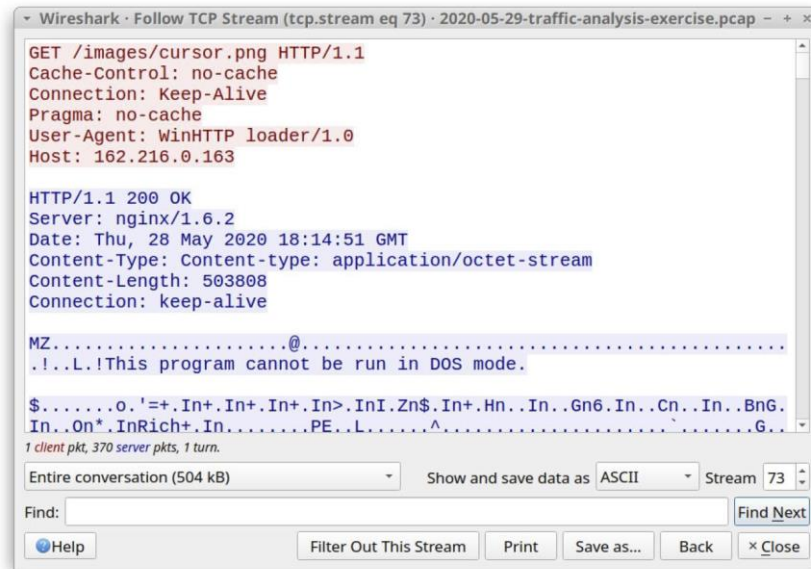
The image shows a Wireshark packet list with a filter bar at the top containing the text "ip contains 'This program'". The filter bar has icons for clearing the filter and a list of filter types: basic, basic+, and basic+dns. Below the filter bar is a table with the following columns: Time, Src, port, Dst, port, and Info. The table contains two rows of data:

Time	Src	port	Dst	port	Info
2020-05-28 18:11...	162.216.0.163	80	10.5.28.229	49286	80 → 49286 [ACK] Seq=1 Ack=
2020-05-28 18:14...	162.216.0.163	80	10.5.28.229	49564	80 → 49564 [PSH, ACK] Seq=

Shown above: Filtering to find EXE files in the pcap

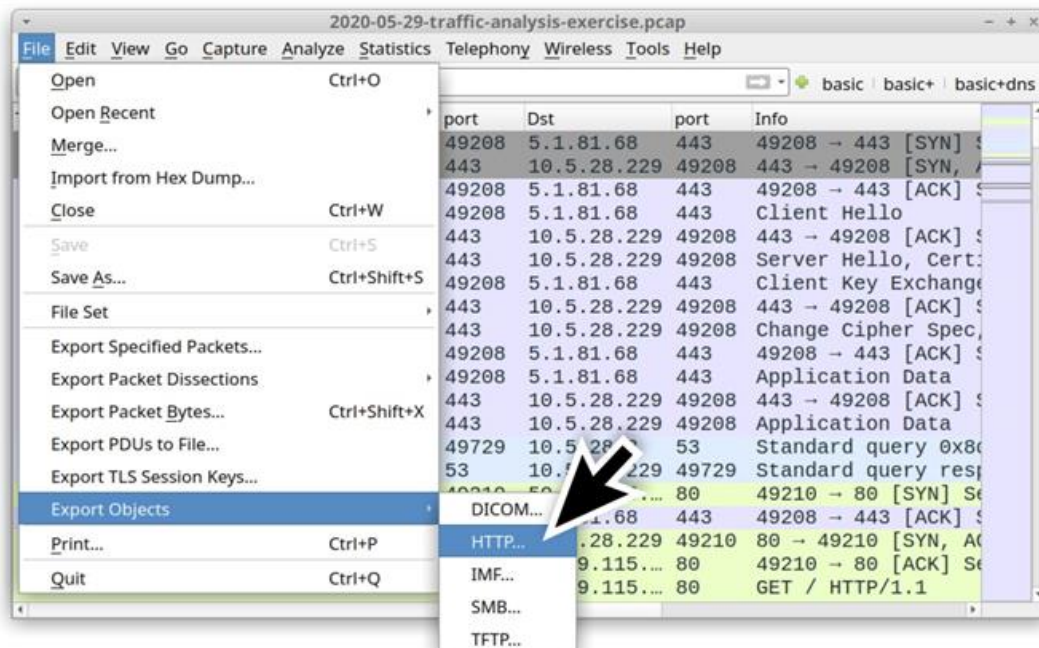


Shown above: The first TCP stream shows an EXE file returned from a URL that ends in imgpaper.png



Shown above: The second TCP stream shows an EXE file returned from a URL that ends in cursor.png.

Now we've confirmed there are two EXE files in this pcap: one from a URL ending in imgpaper.png and one with a URL ending in cursor.png. Make your way to the Export HTTP objects window to export these two files.



Shown above: Exporting HTTP objects from the pcap.

Wireshark · Export · HTTP object list

Packet	Hostname	Content Type	Size	Filename
22	api.ipify.org	text/plain	15 bytes	/
1561	36.89.106.69	multipart/form-data	282 bytes	83
1568	36.89.106.69	text/plain	3 bytes	83
1600	36.89.106.69	multipart/form-data	260 bytes	81
1602	36.89.106.69	text/plain	3 bytes	81
1665	36.89.106.69	multipart/form-data	219 bytes	81
1669	36.89.106.69	text/plain	3 bytes	81
1686	36.89.106.69	multipart/form-data	210 bytes	81
1688	36.89.106.69	text/plain	3 bytes	81
2256	203.176.135.102:8082	multipart/form-data	4,362 bytes	90
2258	203.176.135.102:8082	text/plain	3 bytes	90
2835	162.216.0.163	content-type:	106 kB	VidT6cErs
2975	162.216.0.163	content-type:	105 kB	VidT6cErs
3036	wtfismyip.com	text/plain	16 bytes	text
3047	icanhazip.com	text/plain	16 bytes	/
4585	162.216.0.163	content-type:	503 kB	imgpaper.png
9771	162.216.0.163	content-type:	503 kB	cursor.png
12886	203.176.135.102:8082	multipart/form-data	4,343 bytes	90
12892	203.176.135.102:8082	text/plain	3 bytes	90

Text Filter:

Help Save All Close Save

Shown above: The two objects you need to export for the EXE files.

Now need to take Hash value (Using HashMyFiles) from those saved tow file to analysis the content using VirusTotal

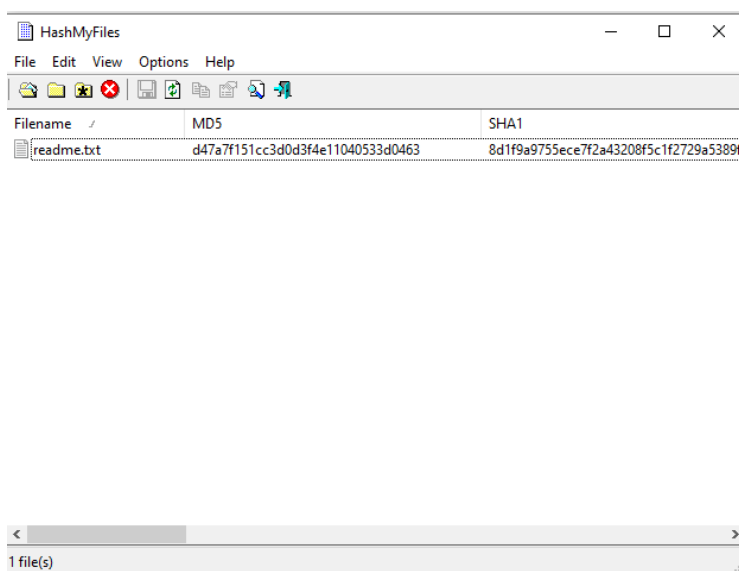
Hash My Files

HashMyFiles is small utility that allows you to calculate the MD5, SHA1, CRC32, SHA256, SHA384 and SHA512 hashes of one or more files in your system. You can easily copy the hashes list into the clipboard, or save them into text/html/xml file

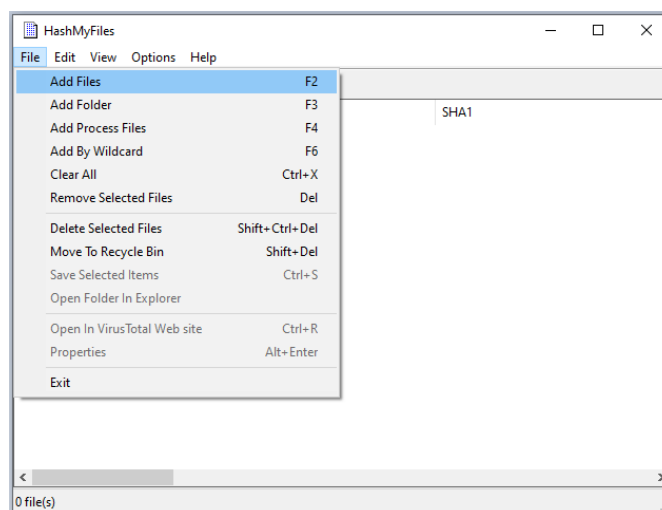
Download Link:

<https://github.com/bandarawmd/HashMyFiles-x64>

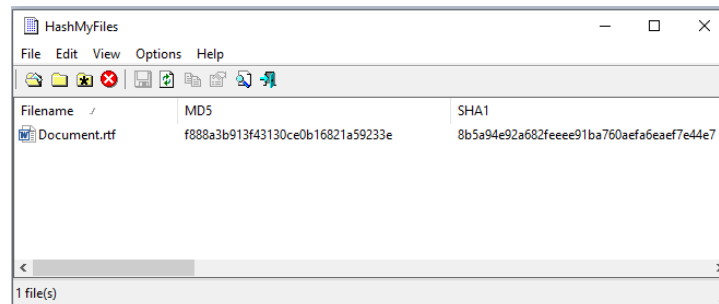
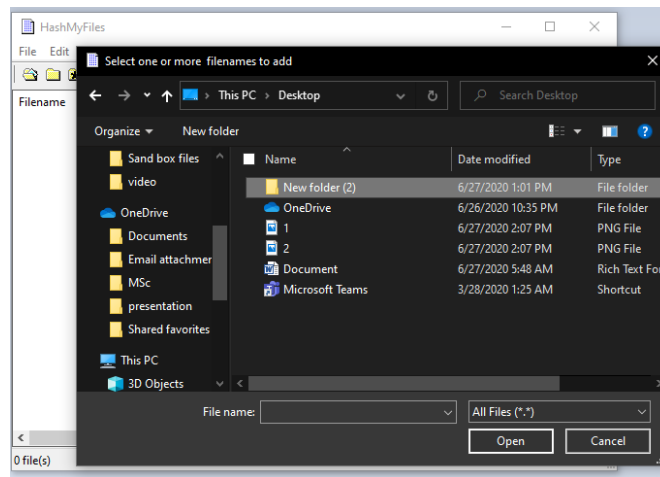
Run *HashMyFiles.exe*



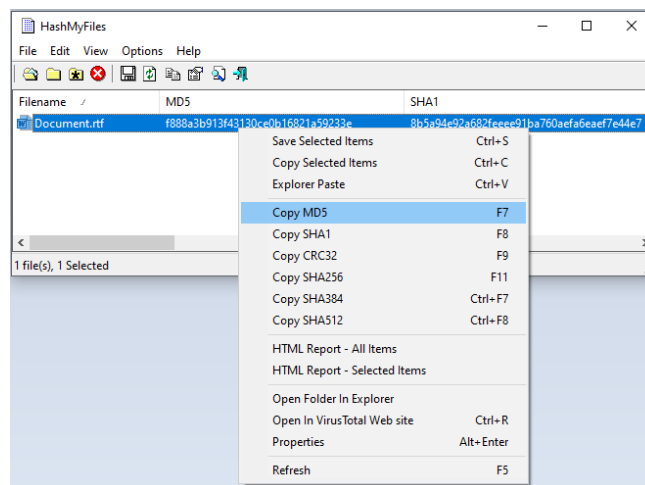
File -> Add File



Select File and Open



Right Click and copy the hash value




VIRUSTOTAL

VirusTotal is an online service that analyzes files and URLs enabling the detection of viruses, worms, trojans and other kinds of malicious content using antivirus engines and website scanners. It also can be used to detect false positives.

Can uploaded the suspicious files into this website and able to see what are initial results.


Is the infected file is a personal file or it is a company business related, can generate hash value of the file and search the result.

[Intelligence](#) [Hunting](#) [Graph](#) [API](#) Sign in




Analyze suspicious files and URLs to detect types of malware, automatically share them with the security community

FILE URL SEARCH



By submitting data below, you are agreeing to our [Terms of Service](#) and [Privacy Policy](#), and to the sharing of your Sample submission with the security community. Please do not submit any personal information; VirusTotal is not responsible for the contents of your submission. [Learn more](#)

Choose file



4e76d73f3b303e481036ada80c2eeba8db2f306cbc9323748560843c80b2fed1

43
/ 70

Community Score

43 engines detected this file

4e76d73f3b303e481036ada80c2eeba8db2f306cbc9323748560843c80b2fed1

cursor.png

492.00 KB
Size

2020-06-05 07:55:41 UTC
21 days ago

EXE

checks-user-input direct-cpu-clock-access peexe runtime-modules

DETECTION

DETAILS

RELATIONS

BEHAVIOR

COMMUNITY 2

Ad-Aware	Trojan.GenericKDZ.67476	AegisLab	Trojan.Win32.Mansabo.41c
AhnLab-V3	Malware/Win32.RL.Generic.R338752	Alibaba	Trojan.Win32/Mansabo.80692ca9
ALYac	Trojan.GenericKDZ.67476	SecureAge APEX	Malicious
Arcabit	Trojan.Generic.D10794	Avast	Win32-Trojan-gen
Avira (no cloud)	TR/Kryptik.cxxvm	BitDefender	Trojan.GenericKDZ.67476
CAT-QuickHeal	Trojan.Mansabo	Comodo	TrojWare.Win32.Emotet.SL@8sg2qb
CrowdStrike Falcon	Win/malicious_confidence_100% (W)	Cyren	W32/Kryptik.BNU.geniEldorado