

# Network Analysis

**Network analysis is a method used to**

- study the relationships between entities in a network.
- analyze the connections, or links, between the entities.
- study a wide range of systems, including social networks, transportation networks.
- identify the most efficient routes between destinations.
- identify bottlenecks or other issues that may affect the performance of the network.

# Why Network Analysis?

- Performance problems
- Analyze application behaviors
- Troubleshooting
- Locate and detect security breaches (الإختراقات الأمنية)
- ❖ use the network of **social media** users as an example. Analyzing this network helps in
  - Identifying the most influent person/people in a group
  - Defining characteristics of groups of users
  - Prediction of suitable items for users

# Network Analysis using Wireshark

## Wireshark

- Is a powerful Network/packet analyzer tool or Network/packet sniffer tool
- Free and open source.
- allowing you to capture and analyze network traffic.
- break down packets of data being transferred across different networks. The user can search and filter for specific packets of data and analyze how they are transferred across their network.

## Wireshark Core Features

- Capture live packet data
- Import packets from text files
- View packet data and protocol information
- Save captured packet data
- Display packets
- Filter packets
- Search packets
- Colorize packets
- Generate Statistics

## **When should Wireshark be used?**

Wireshark helps:

- Network administrators troubleshoot problems across a network
- Security engineers examine security issues across a network
- QA engineers verify applications
- Developers debug protocol implementations
- Network users learn about a specific protocol

## **windows packet capture WinPcap / Npcap**

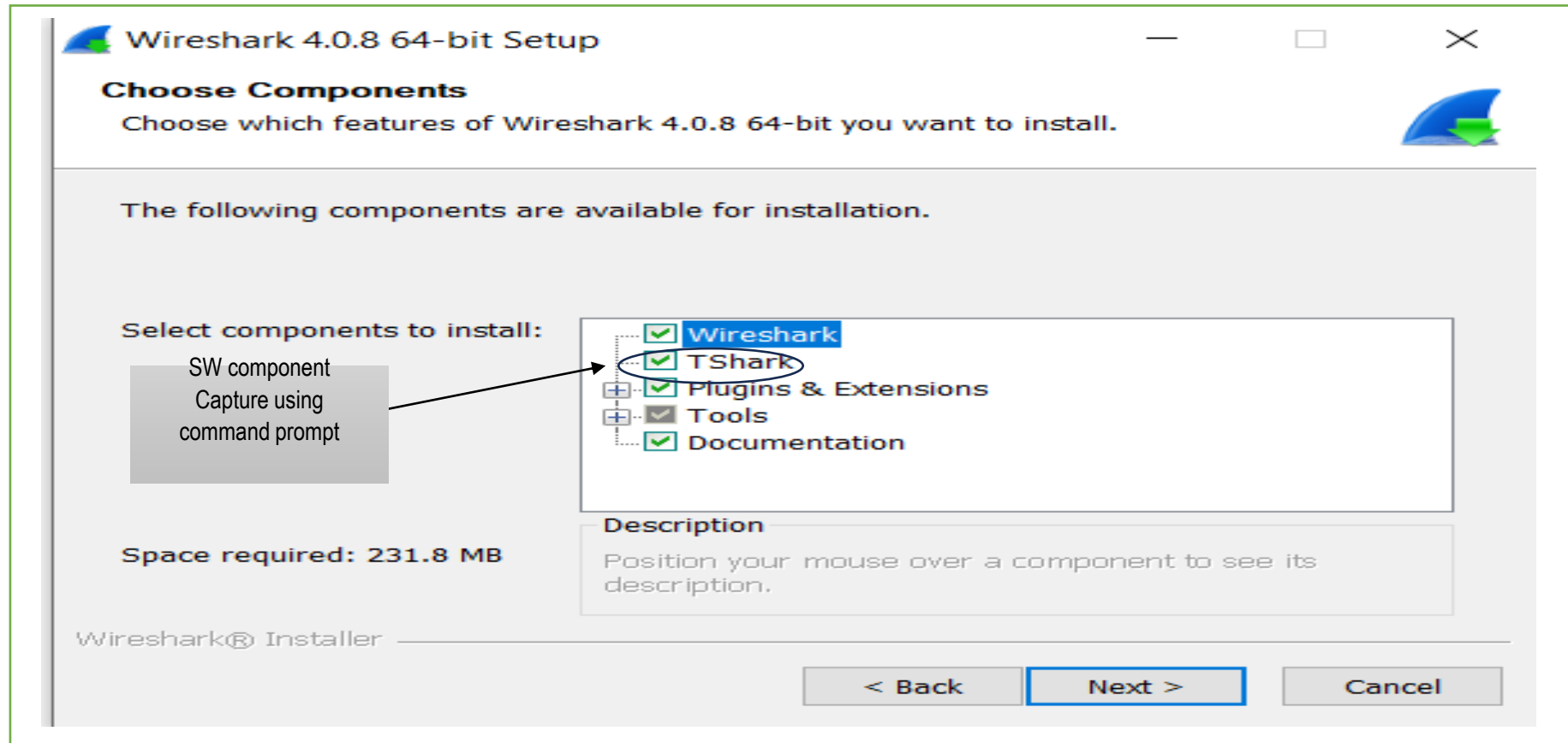
- A software component added to the OS
- open-source library used to capture live network traffic; Without this component you will only be able to open saved capture files.

## **Software requirements: – Wireshark is a Cross platform**

- support any version of Windows
- supports macOS
- runs on most Unix and Linux

## Install Wireshark on Windows

- download and install WinPcap on your computer.
- download and install Wireshark on your computer.



# SECTION 1

## Capturing data packets on Wireshark

### capture network traffic

1. start capture network traffic
2. how to capture dual/ two network interface
3. how to save network traffic
4. how to open saved traffic files

### save network traffic

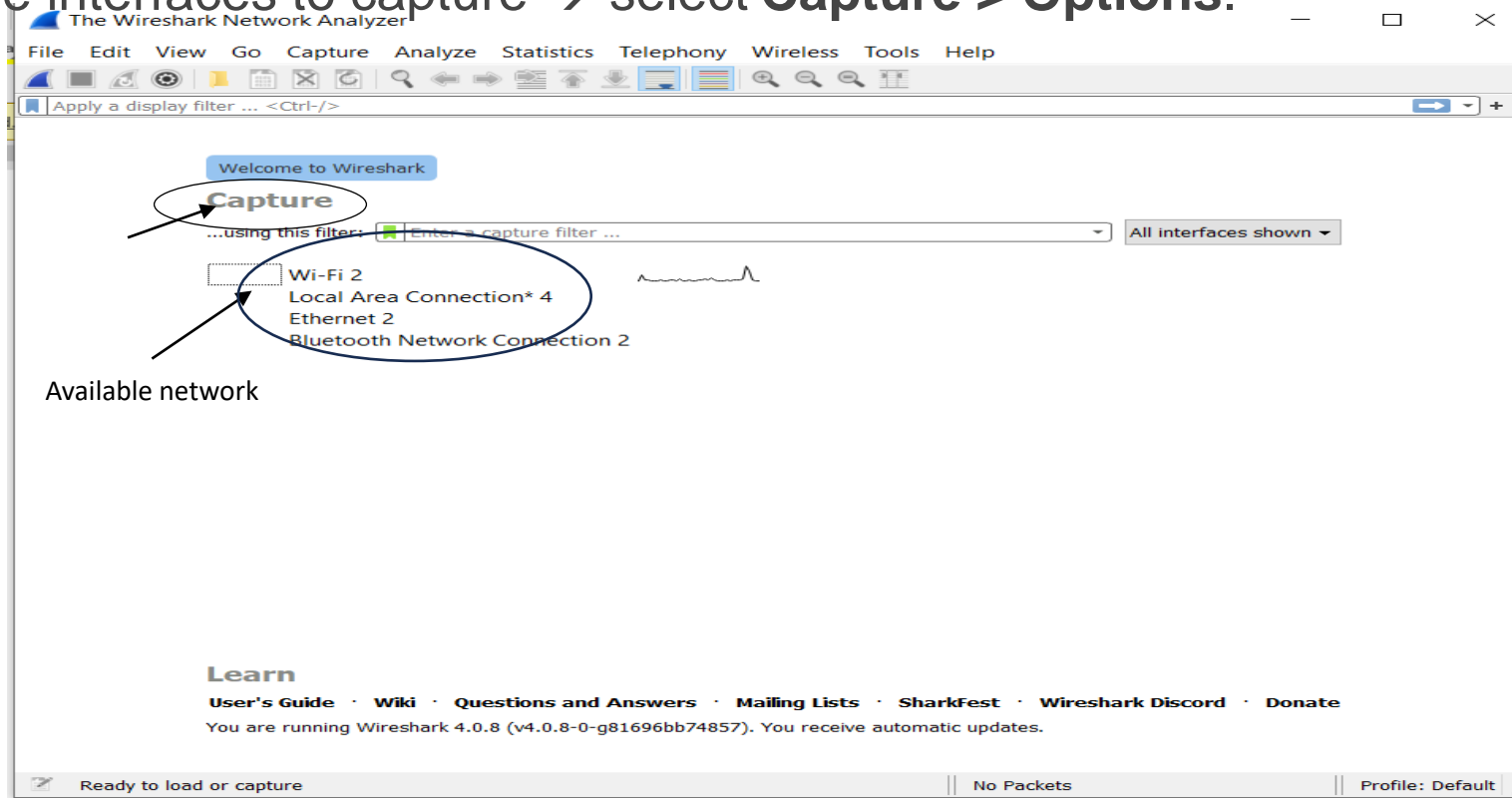
file → save as → name the file → extension of the file (.pcpng) → close the file.

**pcpng** → packet capture next generation.

# start capture network traffic

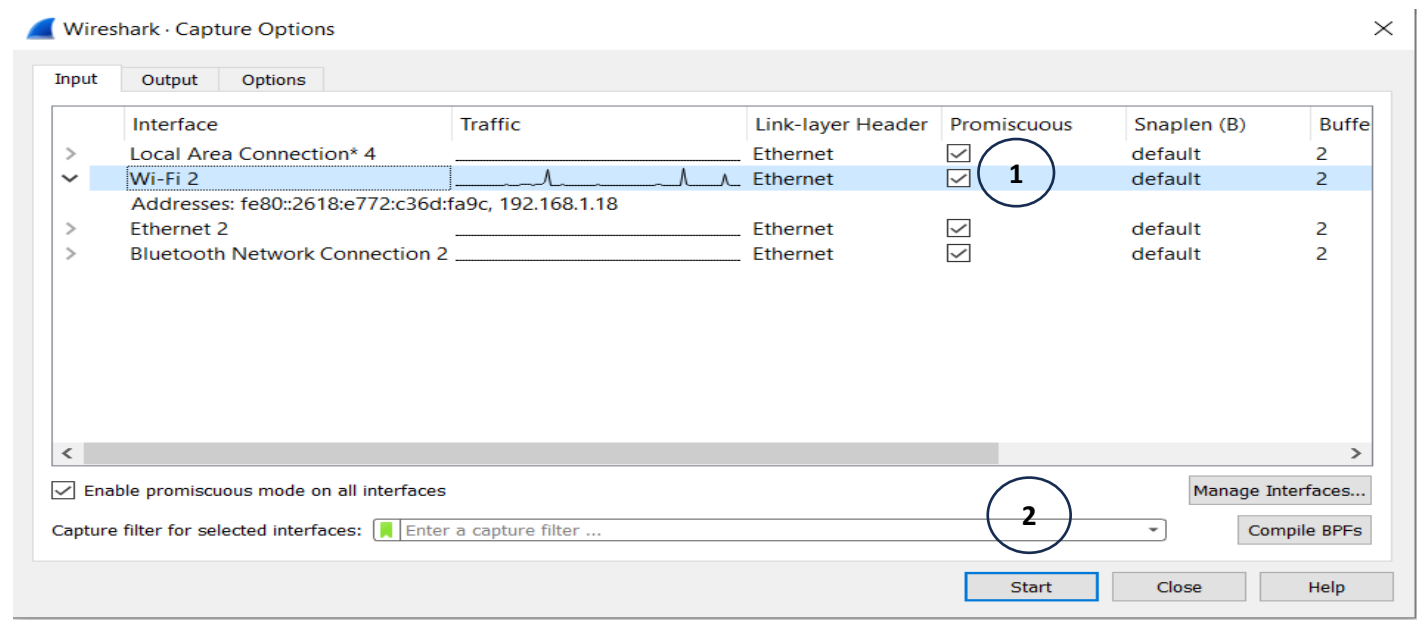
## A. capture data from **one** source / **multiple** sources.

look available interfaces to capture → select **Capture > Options**.

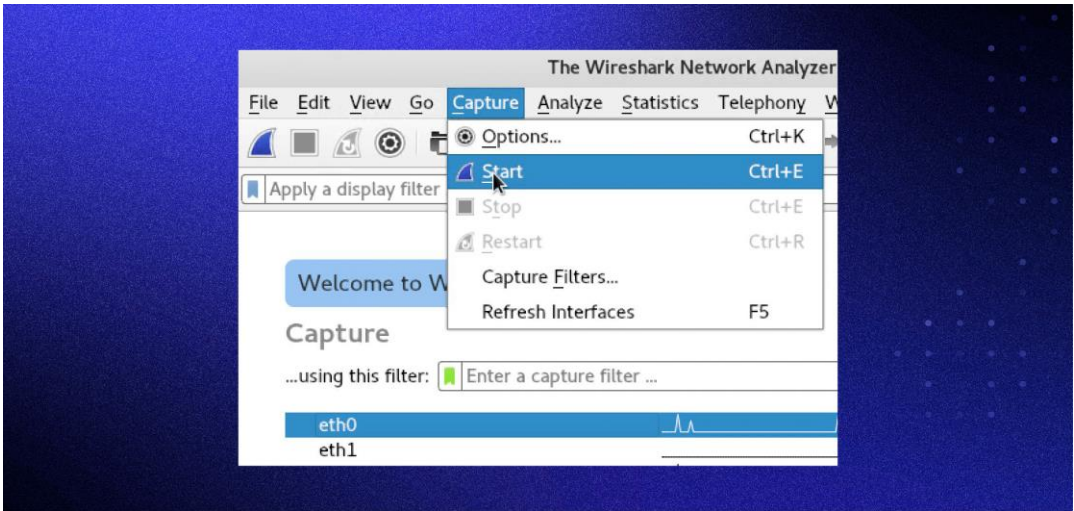
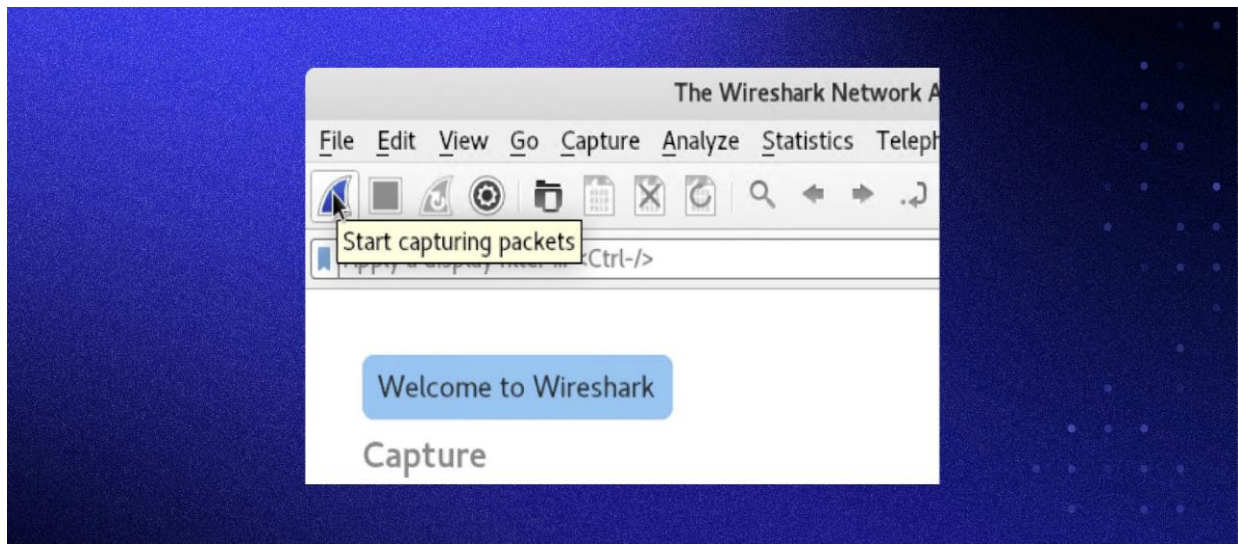




Check the box of the interface you want to capture and press the **Start** button to start



Click the first button on the toolbar, titled “Start capturing packets.” You can select the menu item Capture -> Start.



No.	Time	Source	Destination	Protocol	Length	Info
8	61.440392100	192.168.0.3	192.168.0.1	TCP	66	52060 → 445 [ACK]
9	66.559903000	Microsof_d0:8b:06	Microsof_d0:8b:01	ARP	42	Who has 192.168.0.1
10	66.561858700	Microsof_d0:8b:01	Microsof_d0:8b:06	ARP	42	192.168.0.1 is at
11	83.533524600	fe80::2c14:87e5:857...	ff02::1:2	DHCPv6	164	Solicit XID: 0xcd5
12	84.545422700	fe80::2c14:87e5:857...	ff02::1:2	DHCPv6	164	Solicit XID: 0xcd5
13	86.549466300	fe80::2c14:87e5:857...	ff02::1:2	DHCPv6	164	Solicit XID: 0xcd5
14	90.565378200	fe80::2c14:87e5:857...	ff02::1:2	DHCPv6	164	Solicit XID: 0xcd5

▶ Frame 1: 138 bytes on wire (1104 bits), 138 bytes captured (1104 bits) on interface 0  
 ▶ Ethernet II, Src: Microsof\_d0:8b:06 (00:15:5d:d0:8b:06), Dst: Microsof\_d0:8b:01 (00:15:5d:d0:8b:01)  
 ▶ Internet Protocol Version 4, Src: 192.168.0.3, Dst: 192.168.0.1  
 ▶ Transmission Control Protocol, Src Port: 52060, Dst Port: 445, Seq: 1, Ack: 1, Len: 72  
 ▶ NetBIOS Session Service  
 ▶ SMB2 (Server Message Block Protocol version 2)

0000	00 15 5d d0 8b 01 00 15	5d d0 8b 06 08 00 45 00	..].....].....E.
0010	00 7c 55 e5 40 00 40 06	63 42 c0 a8 00 03 c0 a8	.. U.@. cB.....
0020	00 01 cb 5c 01 bd a6 a7	5f 0b 10 a1 ac 33 80 18	...\\....._....3..
0030	01 06 01 c2 00 00 01 01	00 00 31 00 60 00 00 55	...1.....4.....

## Analyzing data packets on Wireshark

**No.:** This is the number order of the packet captured. The bracket indicates that this packet is part of a conversation.

**Time:** This column shows how long after you started the capture this particular packet was captured. You can change this value in the Settings menu to display a different option.

**Source:** This is the address of the system that sent the packet.

**Destination:** This is the address of the packet destination.

**Protocol:** This is the type of packet. For example: TCP, DNS, DHCPv6, or ARP.

**Length:** This column shows you the packet's length, measured in bytes.

**Info:** This column shows you more information about the packet contents, which will vary depending on the type of packet.

# SECTION 2

## Explain Wireshark interface

The image shows the Wireshark network protocol analyzer interface. It features a menu bar at the top, a toolbar with various icons, and a main display area divided into three panes. Eight numbered callouts point to specific parts of the interface:

- 1: Points to the File menu.
- 2: Points to the Help menu.
- 3: Points to the toolbar.
- 4: Points to the display filter bar.
- 5: Points to the packet list pane.
- 6: Points to the packet details pane.
- 7: Points to the packet bytes pane.
- 8: Points to the status bar.

The packet list pane displays a table of captured packets. The selected packet (No. 91) is highlighted in blue. The packet details pane shows the structure of the selected packet, including Ethernet II, Internet Protocol Version 4, and User Datagram Protocol. The packet bytes pane shows the raw data of the selected packet in hexadecimal and ASCII.


No.	Time	Source	Destination	Protocol	Length	Info
88	0.667449	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
89	0.667450	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
90	0.667451	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
91	0.668240	192.168.1.8	62.68.246.46	UDP	136	60778 → 443 Len=94
92	0.670261	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
93	0.670267	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
94	0.670268	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
95	0.670905	192.168.1.8	62.68.246.46	UDP	136	60778 → 443 Len=94
96	0.680054	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
97	0.680060	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
98	0.680062	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
99	0.680063	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
100	0.680064	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357

Frame 91: 136 bytes on wire (1088 bits), 136 bytes captured (1088 bits) on interface 0  
Ethernet II, Src: c6:cc:40:bb:f7:80 (c6:cc:40:bb:f7:80), Dst: 02:00:11:00:00:00  
Internet Protocol Version 4, Src: 192.168.1.8, Dst: 62.68.246.46  
User Datagram Protocol, Src Port: 60778, Dst Port: 443  
Data (94 bytes)  
Data: 71f545aefb38a3716b4e033b2f3a67269cf4ddb04277bb12 [Length: 94]

0000 b0 89 00 73 80 69 c6 cc 40 bb f7 80 08 00 45 b8 ...s-i.. @.....E-  
0010 00 7a ed 3e 40 00 80 11 00 00 c0 a8 01 08 3e 44 ..z->@... ..>D  
0020 f6 2e ed 6a 01 bb 00 66 c2 9c 71 f5 45 ae fb 38 ..j...f..q·E..8  
0030 a3 71 6b 4e 03 3b 2f 3a 67 26 9c f4 dd b0 42 77 ..qkN.;/: g&....Bw  
0040 bb 12 89 ef 25 8f ec 77 de 53 13 dc 82 61 13 40 ....%..w ·S...a·@  
0050 29 d1 48 94 33 8b f5 f3 f3 15 c4 50 9f 68 33 86 )·H·3... ..P·h3·  
0060 4b 15 63 0c 0f 29 c9 79 51 e2 13 4c 3d 77 22 f5 K·c··)·y Q··L=w"·  
0070 b9 2c 98 76 4a dd d7 8d 3a c6 1f 6f 60 13 cb 31 ..·vJ... :...o`·1  
0080 04 3b 35 4b 5c 32 5e 31 ..;5K\2^1

Data (data.data), 94 bytes | Packets: 100 · Displayed: 100 (100.0%) | Profile: Default

## 1. The Title

 first capture.pcapng

## 2. The Main Menu



## 3. The “Main” Toolbar



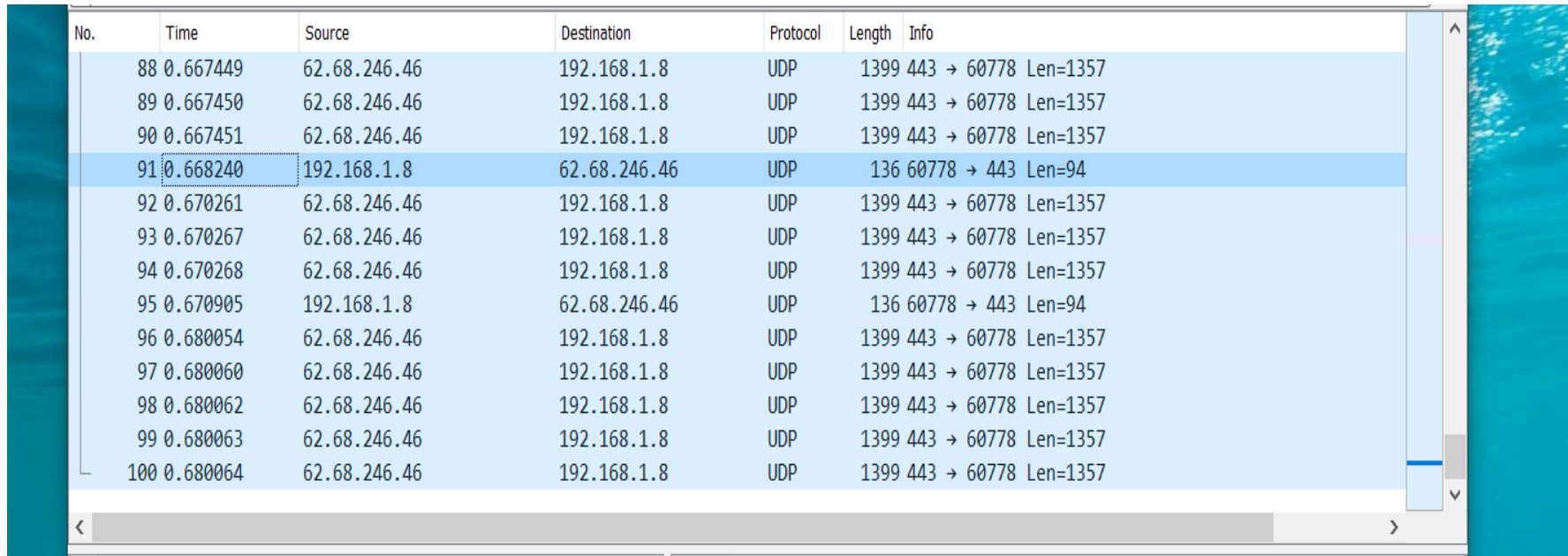
## 4. The “Filter” Toolbar



A syntax check of your filter string is done while you are typing. The background will turn **red** if you enter an **invalid** string, and will become **green** when you enter a **valid** string.



## 5. The “Packet List” Pane



No.	Time	Source	Destination	Protocol	Length	Info
88	0.667449	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
89	0.667450	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
90	0.667451	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
91	0.668240	192.168.1.8	62.68.246.46	UDP	136	60778 → 443 Len=94
92	0.670261	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
93	0.670267	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
94	0.670268	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
95	0.670905	192.168.1.8	62.68.246.46	UDP	136	60778 → 443 Len=94
96	0.680054	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
97	0.680060	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
98	0.680062	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
99	0.680063	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
100	0.680064	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357

Each line in the packet list corresponds to **one packet** in the capture file. If you select a line in this pane, more details will be displayed in the “Packet Details” and “Packet Bytes” panes.

## 6. The “Packet Details” Pane

shows the current packet (selected in the “Packet List” pane) in a more detailed form.

No.	Time	Source	Destination	Protocol	Length	Info
82	0.662615	192.168.1.8	62.68.246.46	UDP	136	60778 → 443 Len=94
83	0.667438	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
84	0.667444	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
85	0.667445	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
86	0.667446	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
87	0.667447	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357
88	0.667449	62.68.246.46	192.168.1.8	UDP	1399	443 → 60778 Len=1357

> Frame 84: 1399 bytes on wire (11192 bits), 1399 bytes captured (11192 bits) on interface \Device\NPF\_{4FEECC8F-E70D-469E-9721-30DB6D3D3}

> Ethernet II, Src: HuaweiTe\_73:80:69 (b0:89:00:73:80:69), Dst: c6:cc:40:bb:f7:80 (c6:cc:40:bb:f7:80)

> Internet Protocol Version 4, Src: 62.68.246.46, Dst: 192.168.1.8

> User Datagram Protocol, Src Port: 443, Dst Port: 60778

▼ Data (1357 bytes)

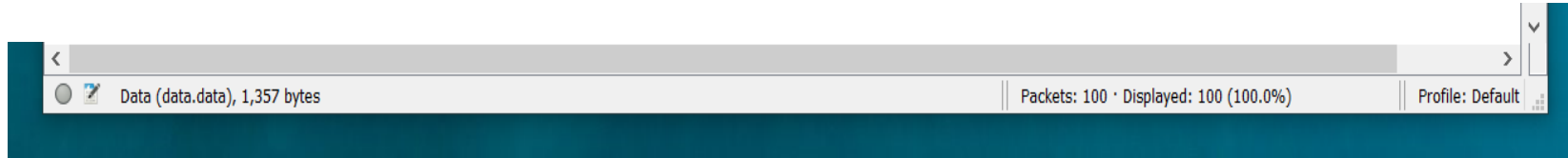
Data: 5f79cf6ccfc1666a9a38376b82aa8c18fa6249fcf8d9c85f7d029854e823566796d8c476...  
[Length: 1357]

## 7. The “Packet Bytes” Pane

0000	c6 cc 40 bb f7 80 b0 89 00 73 80 69 08 00 45 00	..@.....s.i..E.
0010	05 69 8d 88 00 00 7d 11 b4 d8 3e 44 f6 2e c0 a8	.i....}.>D...
0020	01 08 01 bb ed 6a 05 55 30 6c 5f 79 cf 6c cf c1	.....j·U 0l_y·l..
0030	66 6a 00 78 77 6b 82 8a 8c 10 fa 62 49 fc f8 d9	fj·87k...·bI...
0040	c8 5f 00 70 77 6b 82 8a 8c 10 fa 62 49 fc f8 d9	l T # V α v *
0050	4a 90 00 70 77 6b 82 8a 8c 10 fa 62 49 fc f8 d9	
0060	5f a5 e6 00 fe 17 fe 84 ed 05 5a f7 2c a2 5f 57	
0070	bc e1 fb 3c 3a fb 10 e8 b1 e4 ee 55 c4 50 b0 14	...<?...·U·P..
0080	c6 e5 8a fc 84 df 01 8f 0a 58 22 0d f3 b4 d1 f6	.....·X".....
0090	09 27 22 61 a6 3b df 2c d4 cb 16 cb 72 8f e3 4a	·'"a·;·, ····r··J
00a0	47 91 70 e2 96 1c d8 b8 ae cb c6 bb 86 47 e0 8f	G·p.....·G..
00b0	2c 06 af 08 b5 82 a2 3d d1 71 0a de ab ea 23 75	,.....= ·q....#u
00c0	41 5b 16 7e 42 23 65 00 9f 2c ad e0 2f 4a 77 94	A[·~B#e· ·,··/Jw·
00d0	35 7a 2d 09 d9 57 8f c1 65 7d 3f 28 3f c4 0d 4e	5z-··W··e}?(?·N

Each line contains the data offset, **sixteen hexadecimal** bytes, and **sixteen ASCII bytes**. Non-printable bytes are replaced with a period (“.”).

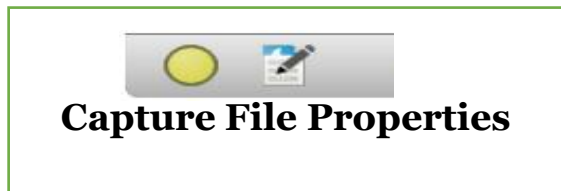
## 8. The Status bar



**Packets:** The number of captured packets.

**Displayed:** The number of packets currently displayed when using filter.

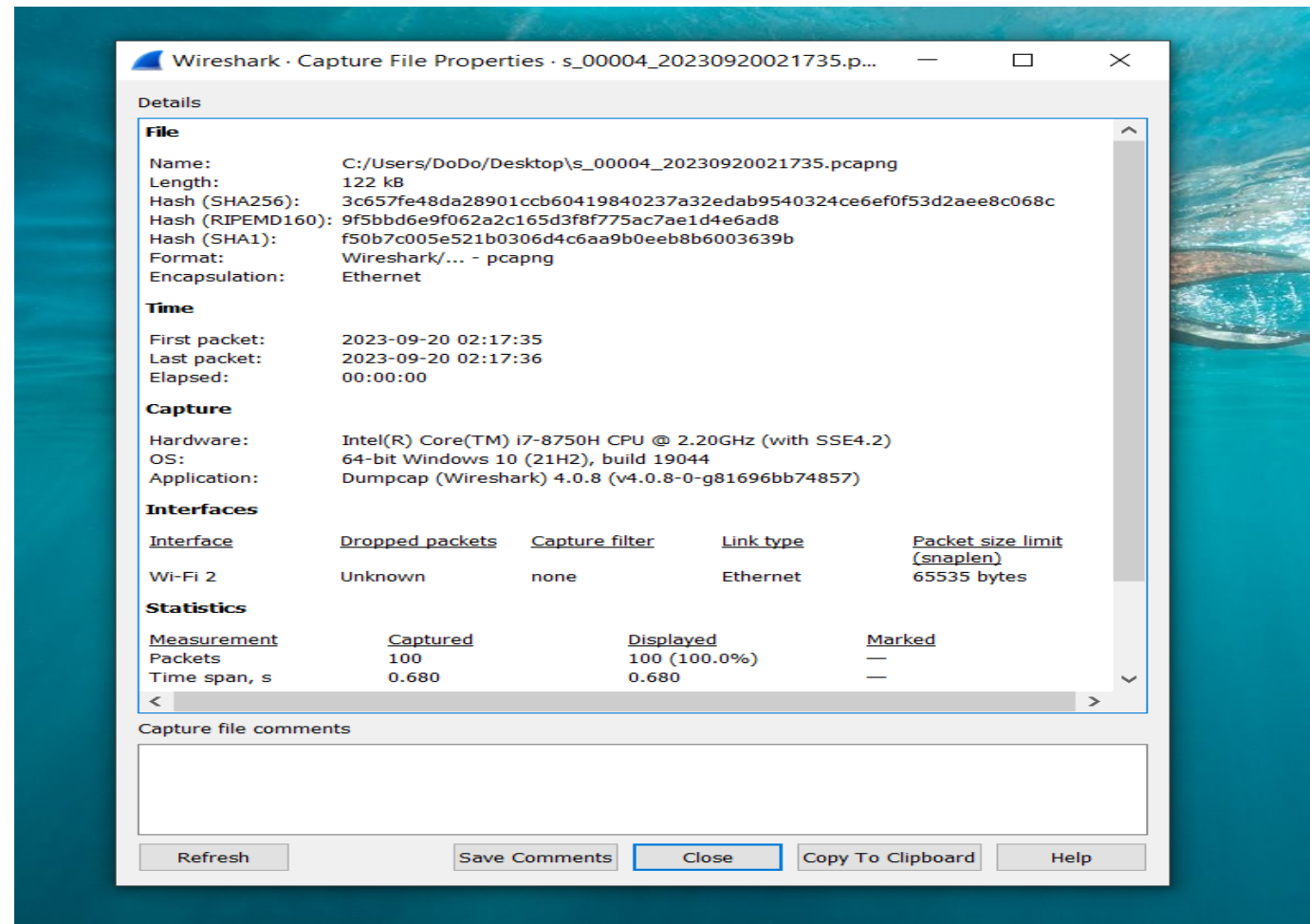
**Profile:** selecting from this list will change the configuration profile.



add a comment to the capture file , Show properties of capture file

**Comment:** Can be used to add a descriptive comment for the interface.

**.pcapng** → only support comment











# SECTION 3

- 1) Explain main toolbar
- 2) Save captured packets in more than one file

## Main toolbar items

Toolbar Icon	Toolbar Item	Menu Item	Description
	Start	Capture → Start	Starts capturing packets
	Stop	Capture → Stop	Stops the currently running capture
	Restart	Capture → Restart	Restarts the current capture session.
	Options...	Capture → Options...	Opens the “Capture Options” dialog box.
	Open...	File → Open...	Opens the file open dialog box, which allows you to load a capture file for viewing.
	Save As...	File → Save As...	Save the current capture file to whatever file you would like.

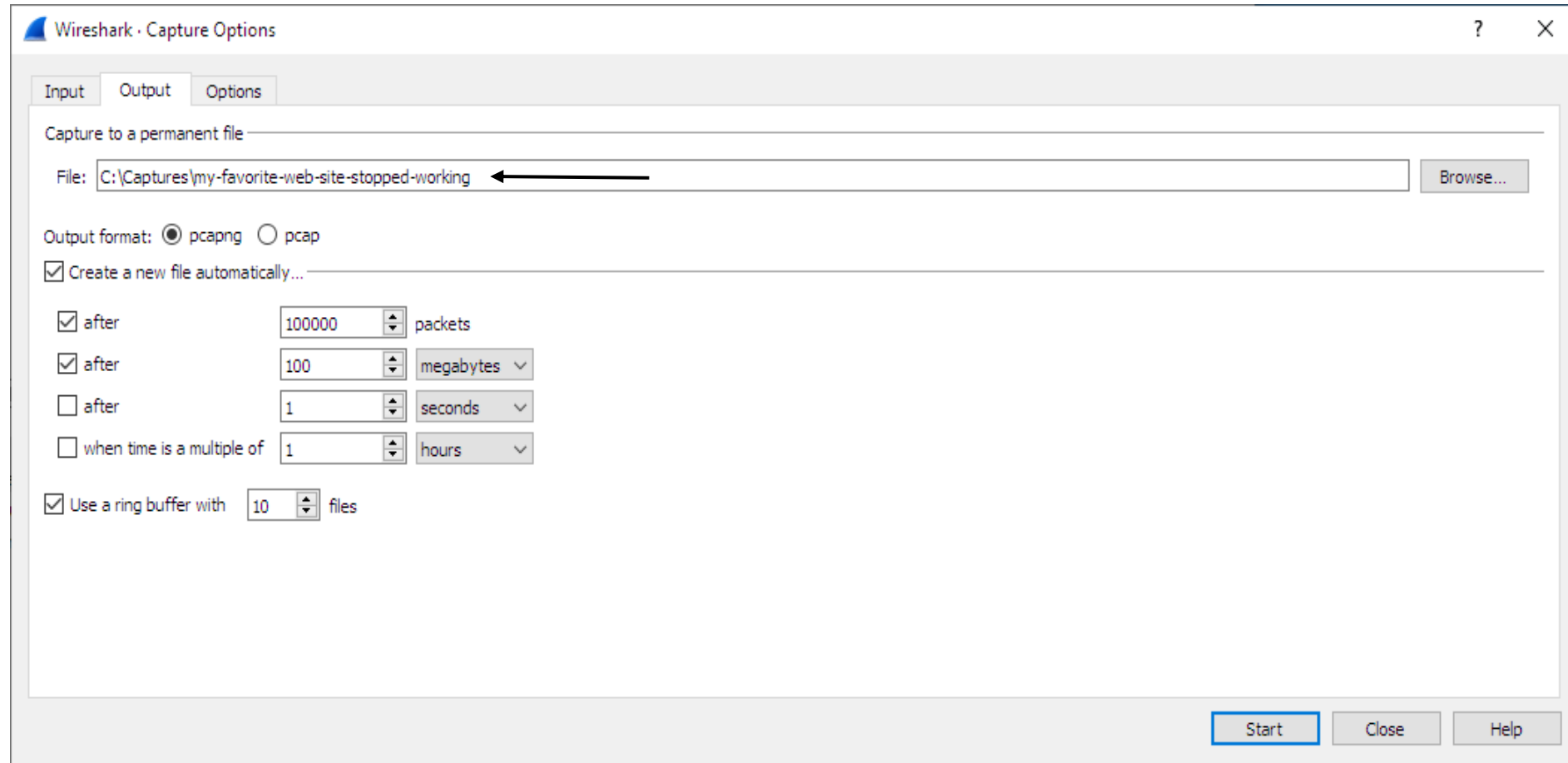


Toolbar Icon	Toolbar Item	Menu Item	Description
	Open...	File → Open...	Opens the file open dialog box, which allows you to load a capture file for viewing.
	Save As...	File → Save As...	Save the current capture file to whatever file you would like.
	Close	File → Close	Closes the current capture. If you have not saved the capture, you will be asked to save it first.
	Reload	View → Reload	Reloads the current capture file.
	Find Packet...	Edit → Find Packet...	Find a packet based on different criteria.
	Go Back	Go → Go Back	Jump back in the packet history (previous packet)
	Go Forward	Go → Go Forward	Jump forward in the packet history (next packet)
	Go to Packet...	Go → Go to Packet...	Go to a specific packet.
	Go To First Packet	Go → First Packet	Jump to the first packet of the capture file.
	Go To Last Packet	Go → Last Packet	Jump to the last packet of the capture file.
	Auto Scroll in Live Capture	View → Auto Scroll in Live Capture	Auto scroll packet list while doing a live capture (or not).
	Colorize	View → Colorize	Colorize the packet list (or not).
	Zoom In	View → Zoom In	Zoom into the packet data (increase the font size).
	Zoom Out	View → Zoom Out	Zoom out of the packet data (decrease the font size).
	Normal Size	View → Normal Size	Set zoom level back to 100%.
	Resize Columns	View → Resize Columns	size columns, so <b>the</b> content fits into them.

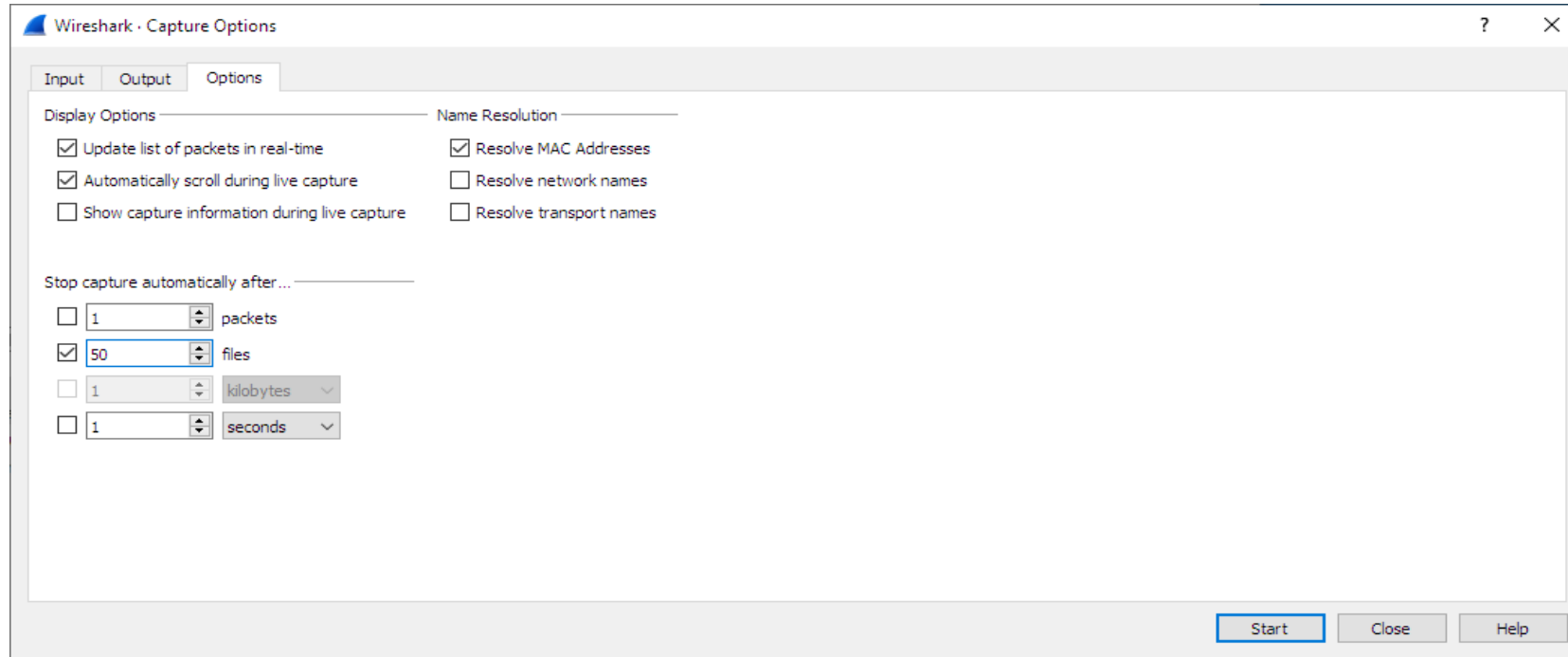
# Save capture packet in more than one file

## 1.The “Capture Options” output tab

Set the format of the capture file. **pcapng** is the default

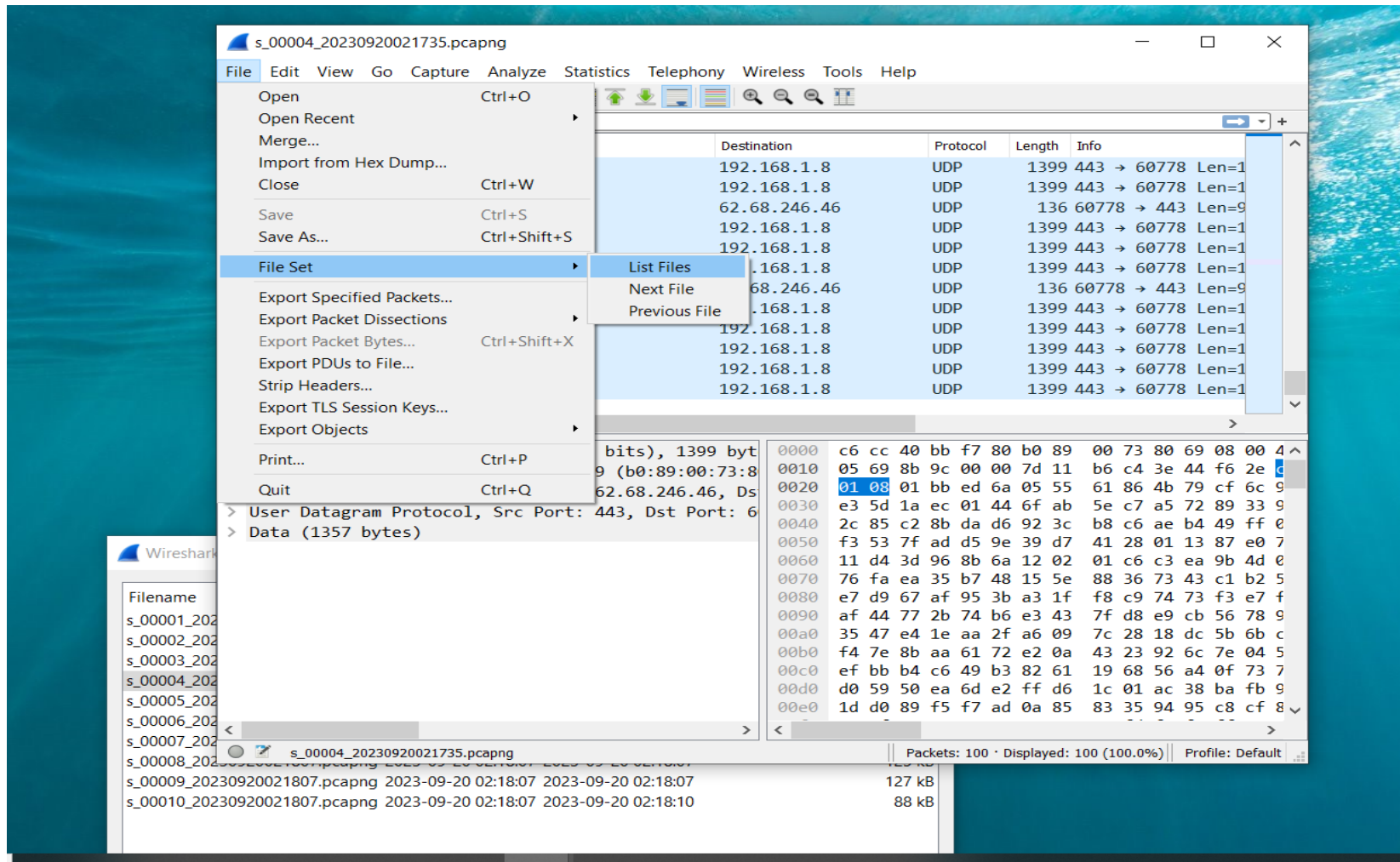


## •The “Capture Options” options tab

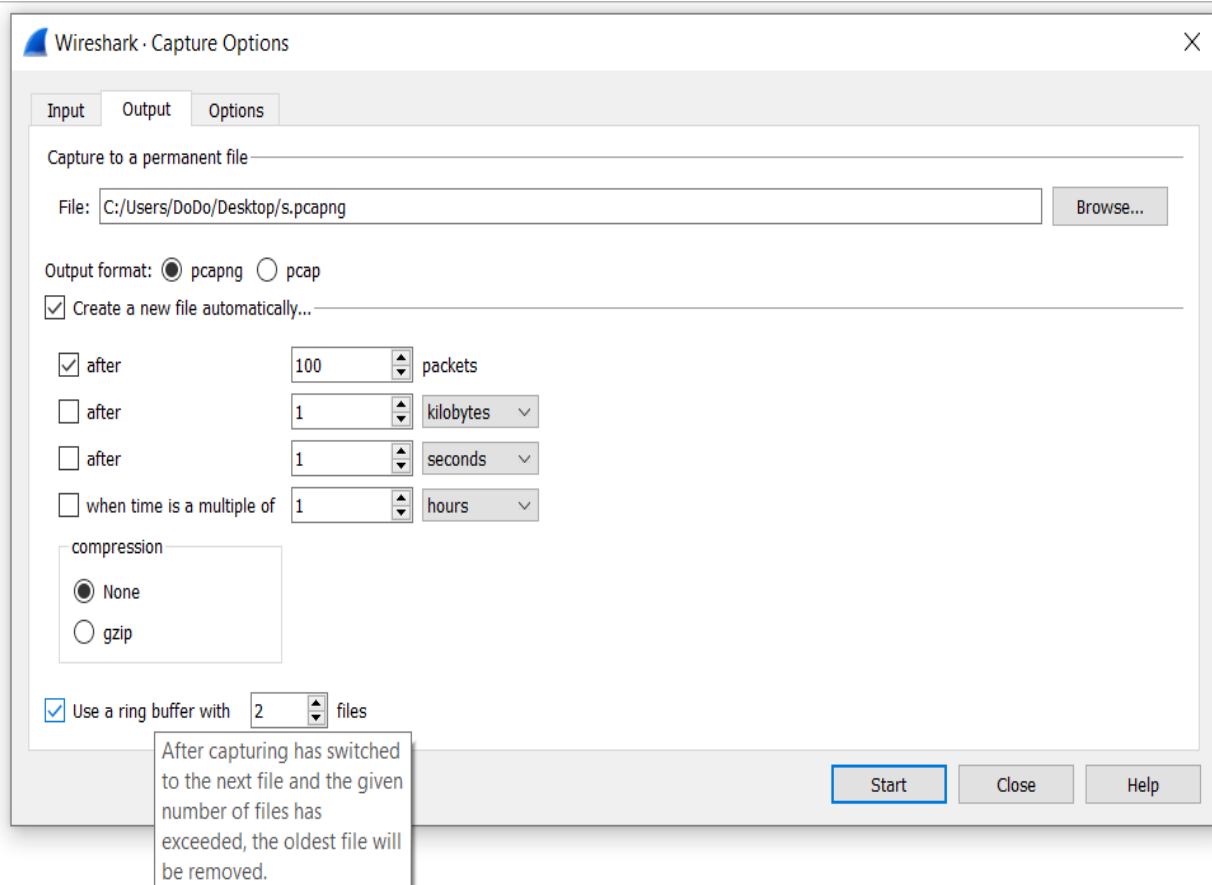


### **Capturing can be stopped automatically based on the following conditions:**

- The number of packets in the capture file.
- The number of capture files.
- The capture file size.
- The capture file duration.



- Show all created trace file after capturing.



- **Multiple files, ring buffer**

This will be a newly created file if value of “Ring buffer with n files” is not reached, otherwise it will replace the oldest of the formerly used files (thus forming a “ring”).

# Remote Packet Capture Daemon

**rpcapd** → stands for **Remote Packet Capture Daemon**  
**used for capture packet remotely**

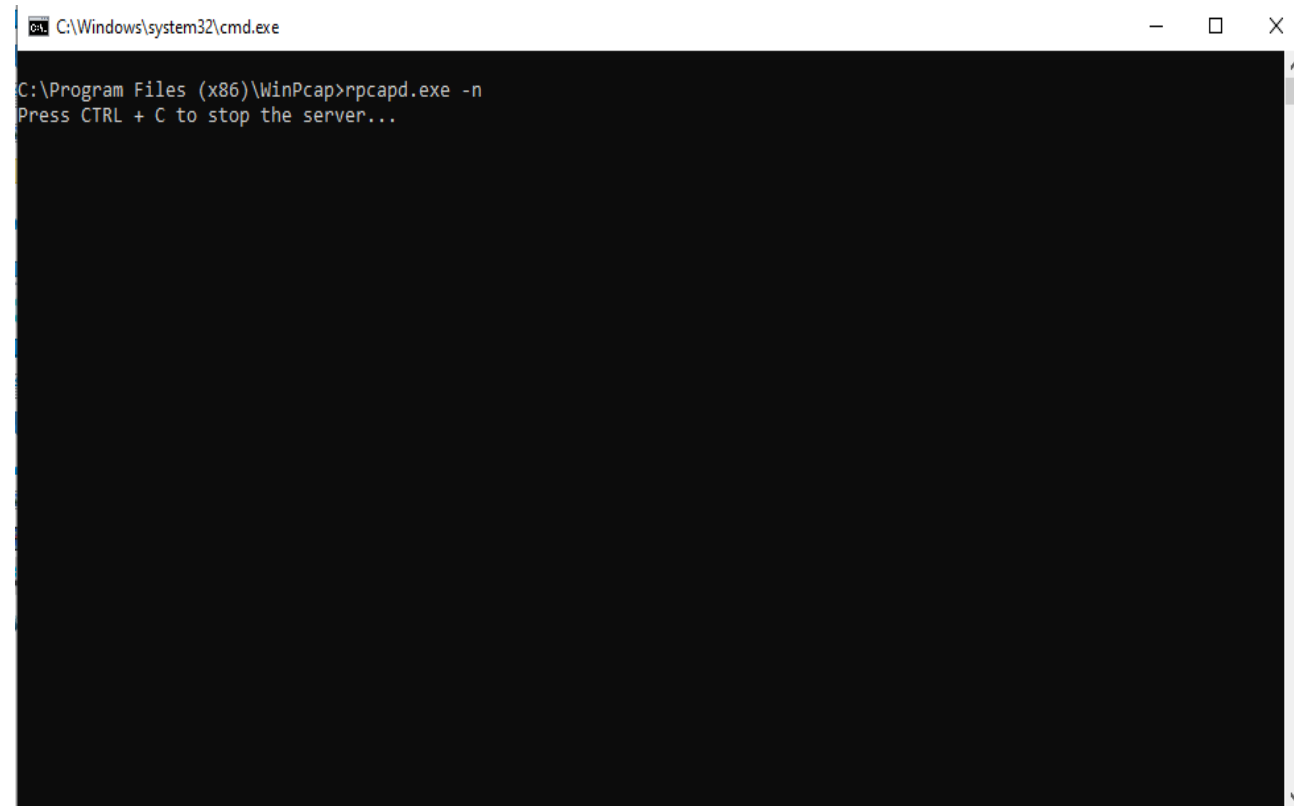
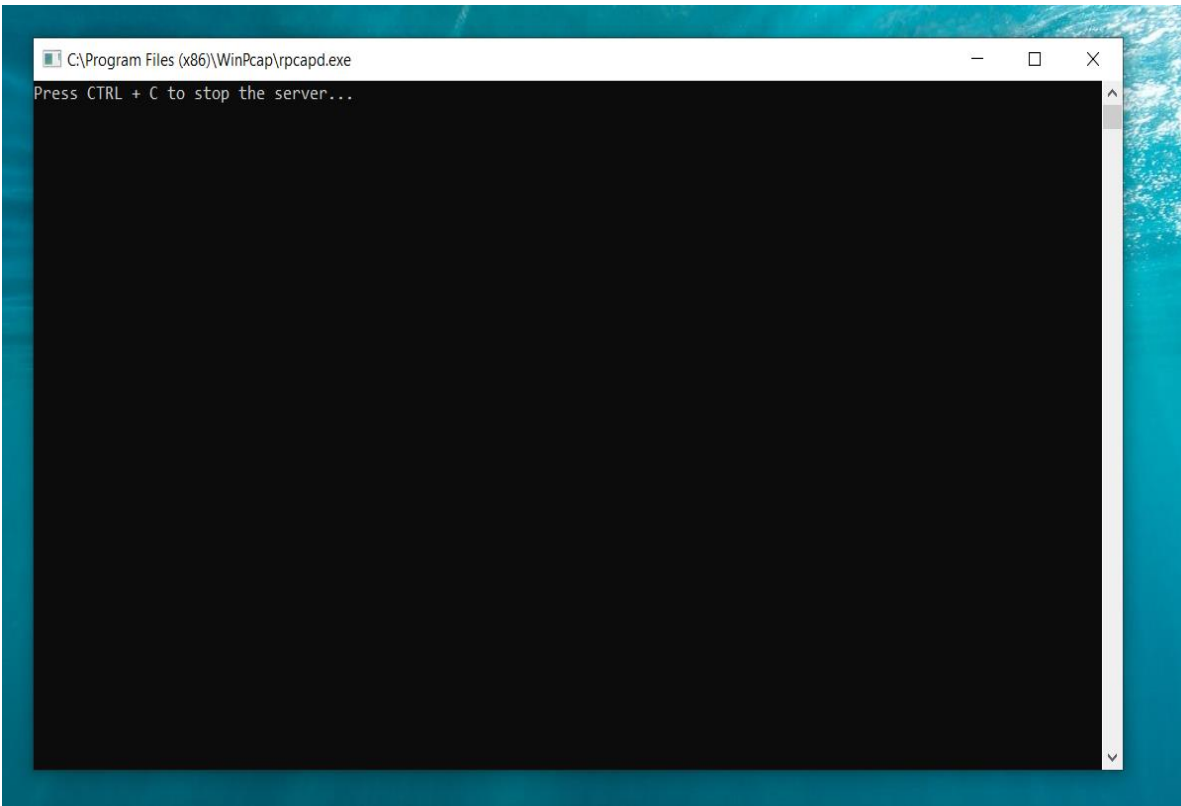
***rpcapd.exe* file** is a software component of *Remote Packet Capture Daemon*

**WinPcap** is a packet sniffing tool that provides access to link-layer networks for Windows machines.

Rpcapd.exe is part of the WinPcap packet library.

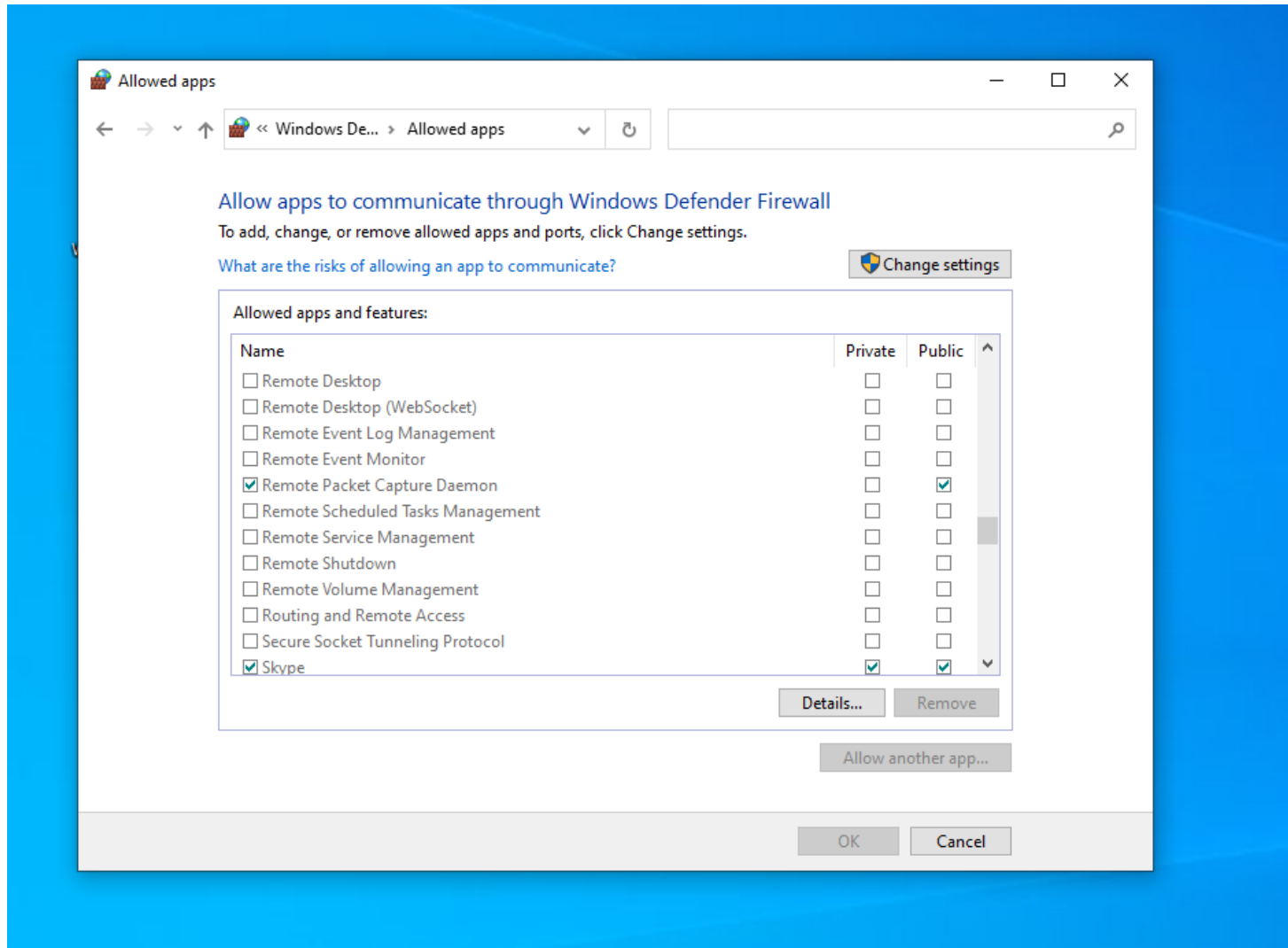
## In the remote desktop

1. Just install **WinPcap** to remote
2. **C:\Program Files (x86)\WinPcap**
3. Run **rpcapd.exe**.
4. Write this command in cmd to run null authentication and change the port→  
**rpcapd.exe -n -p 2004**



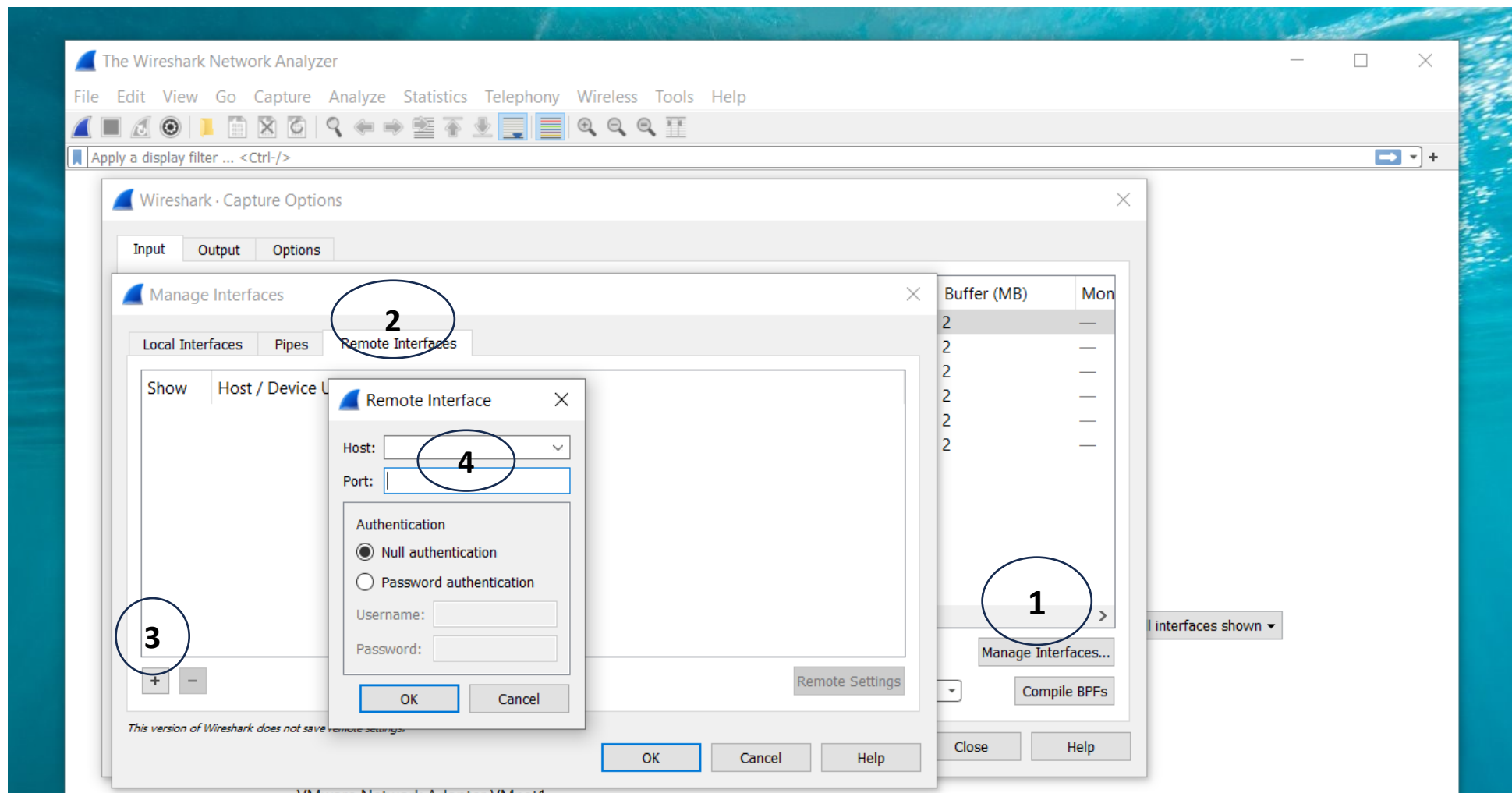


## 5. Goto Control Panel\All Control Panel Items\Windows Defender Firewall\Allowed apps → to allow daemon through firewall



## In the captured desktop

## 6. Open Wireshark to add the Ip address of remote desktop



7. In Wireshark Choose the remote desktop network and start capture

8. Open cmd in **remote desktop** and write **ping google.com -n 2**

9. the Wireshark send two command pings of request and reply.

The screenshot displays two windows. The top window is Wireshark, showing a packet capture on the 'icmp' filter. It lists four packets: two requests and two replies. The bottom window is a Windows 10 x64 VM running VMware Workstation 17 Player. Inside, a Command Prompt window shows the execution of 'ping google.com -n 2'. The output shows two successful pings to 172.217.18.46 with 32 bytes of data, 75ms-77ms round trip times, and 128 TTL. Below the Command Prompt, a network statistics window shows the following data:

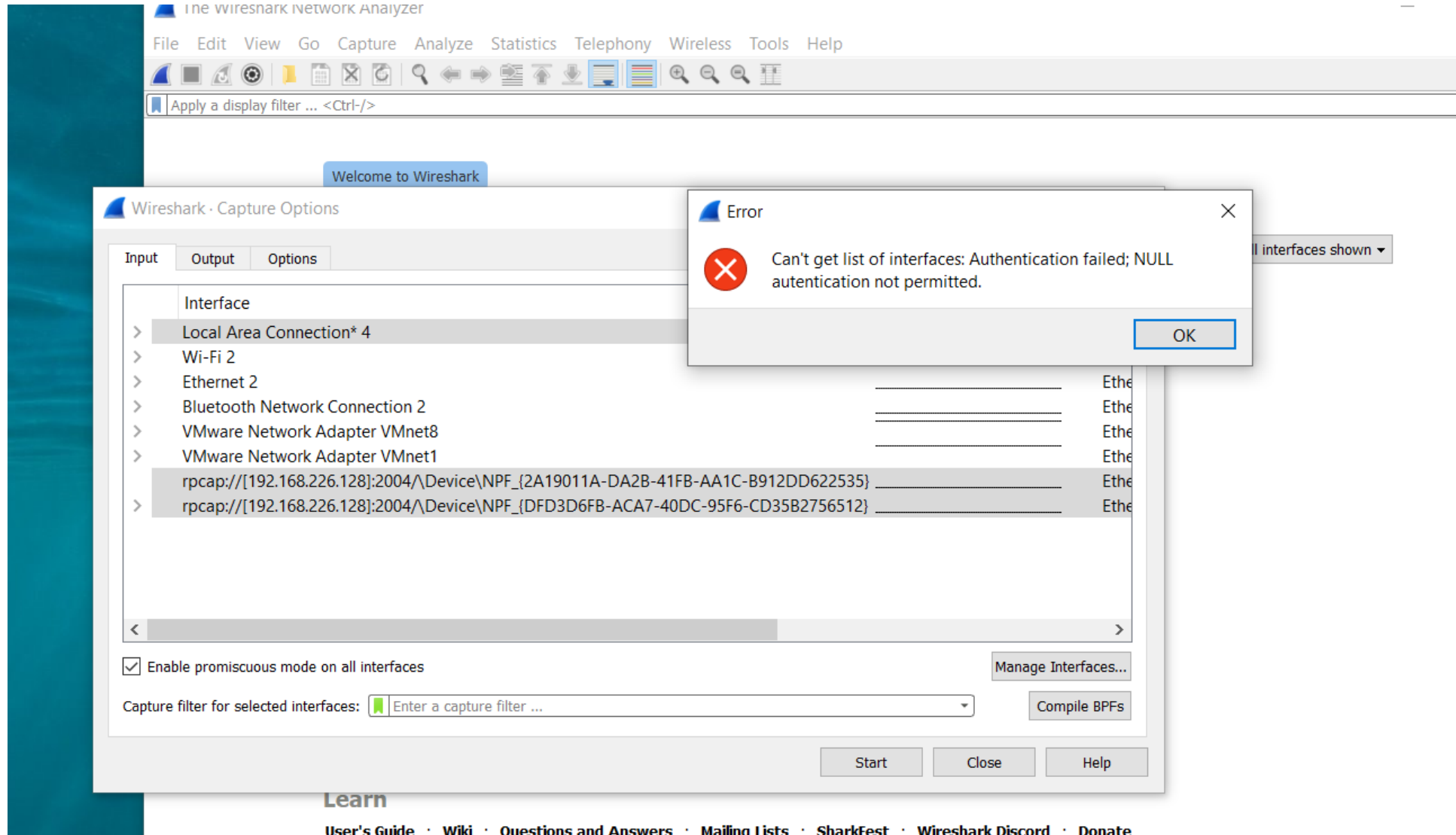
Time	Source	Destination	Protocol	Length	Info
325 65.443439	192.168.226.128	172.217.18.46	ICMP	74	Echo (ping) request id=0x0001, seq=17/4352, ttl=128 (re
326 65.520338	172.217.18.46	192.168.226.128	ICMP	74	Echo (ping) reply id=0x0001, seq=17/4352, ttl=128 (re
327 66.484384	192.168.226.128	172.217.18.46	ICMP	74	Echo (ping) request id=0x0001, seq=18/4608, ttl=128 (re
328 66.558788	172.217.18.46	192.168.226.128	ICMP	74	Echo (ping) reply id=0x0001, seq=18/4608, ttl=128 (re

ping statistics for 172.217.18.46:  
Packets: Sent = 2, Received = 2, Lost = 0 (0% loss),  
approximate round trip times in milli-seconds:  
Minimum = 75ms, Maximum = 77ms, Average = 76ms

**icmp** → **ping** protocol

**Netstat.exe -n -p 2004 1 192.168.1.10,192.168.1.50,... → determine Ip for remote connection**

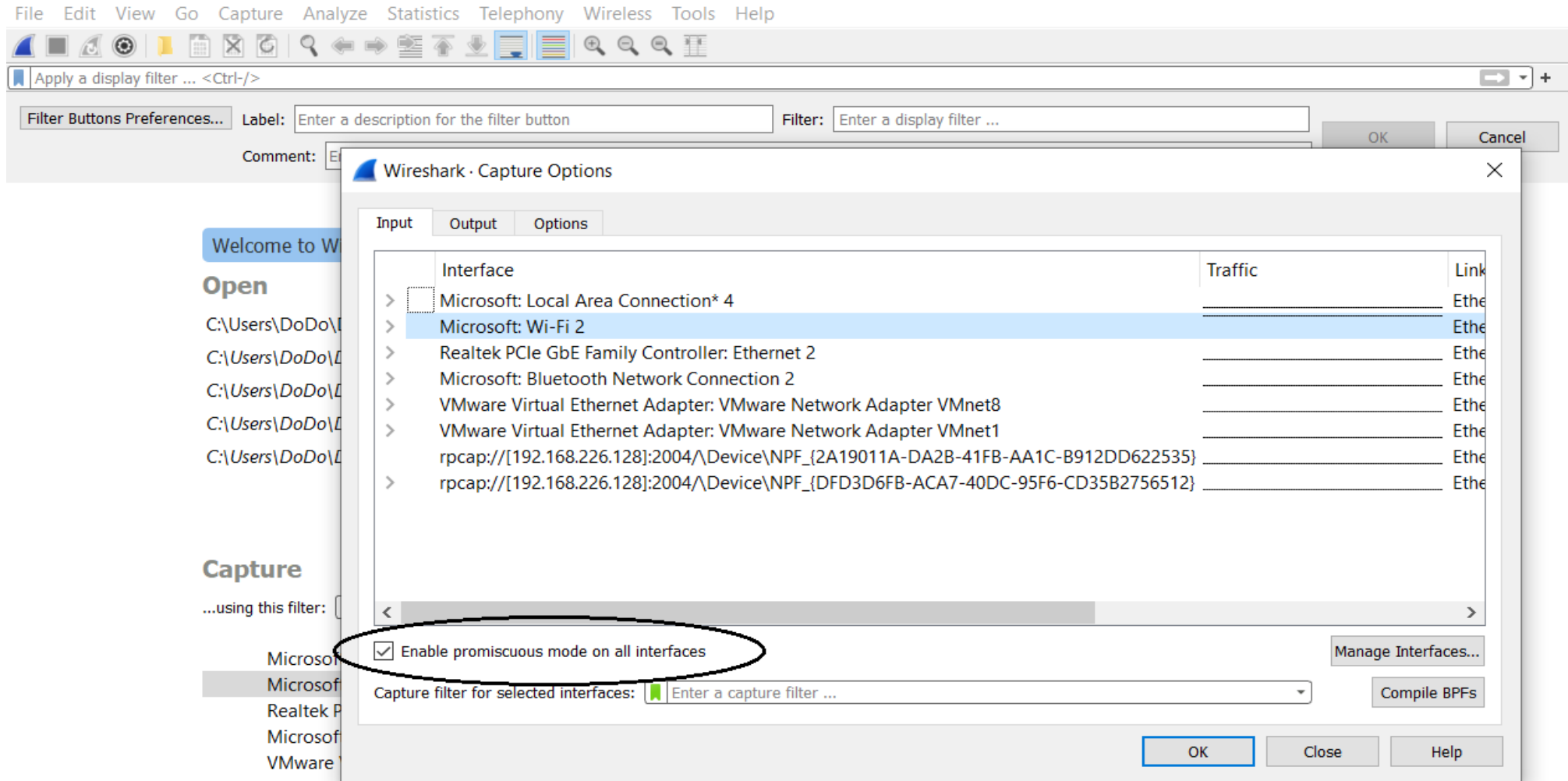
**If not allowed for connection → return this message in Wireshark**



# What is promiscuous mode

Allows a network device to intercept and read each network packet

is a mode of operation in which every **data packet** transmitted can be received and read by a network **adapter**.



# Filter in Wireshark

## capture filters and display filters

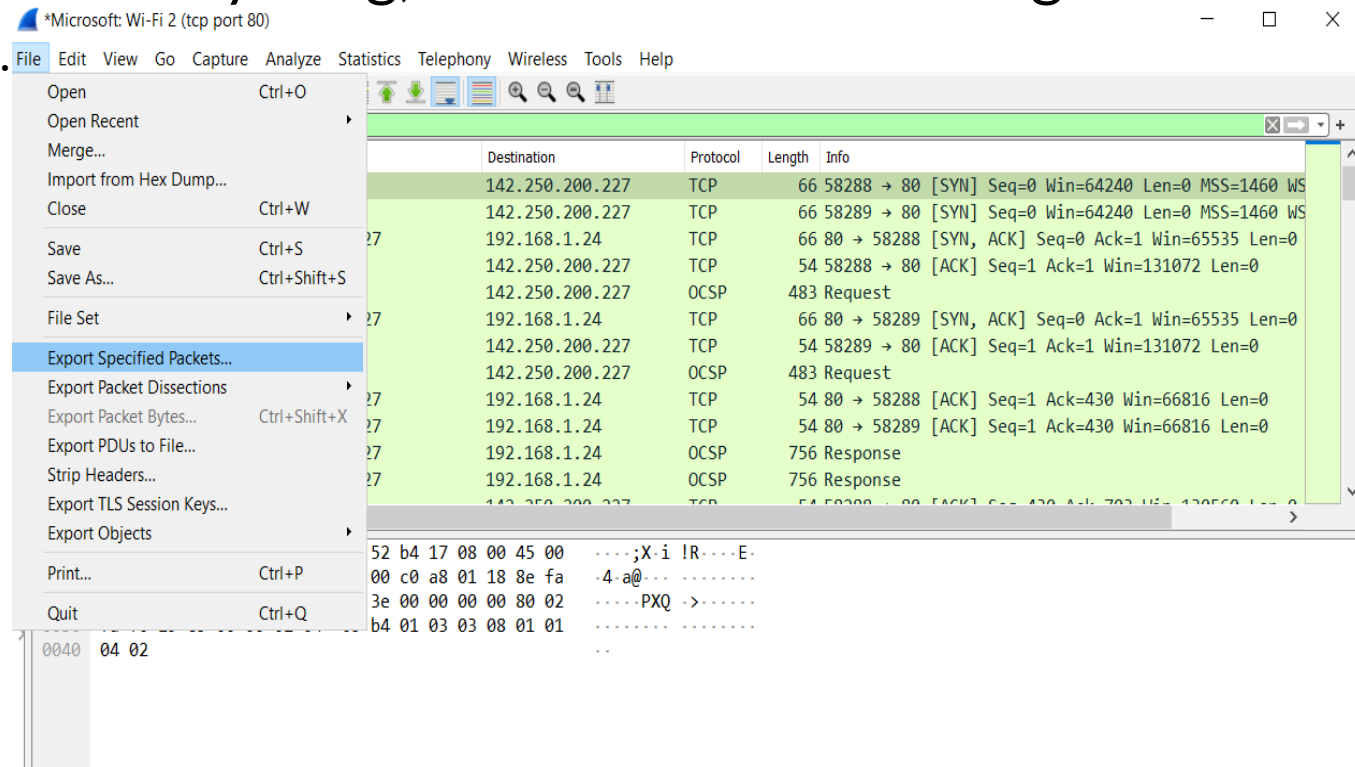
Capture filters (like `tcp port 80`) are not to be confused with display filters (like `tcp.port == 80`)

**Capture filters** :- only keep copies of packets that match the filter.

**Display filters** :- used when you've captured everything, but need to cut through the noise to analyze specific packets or flows.

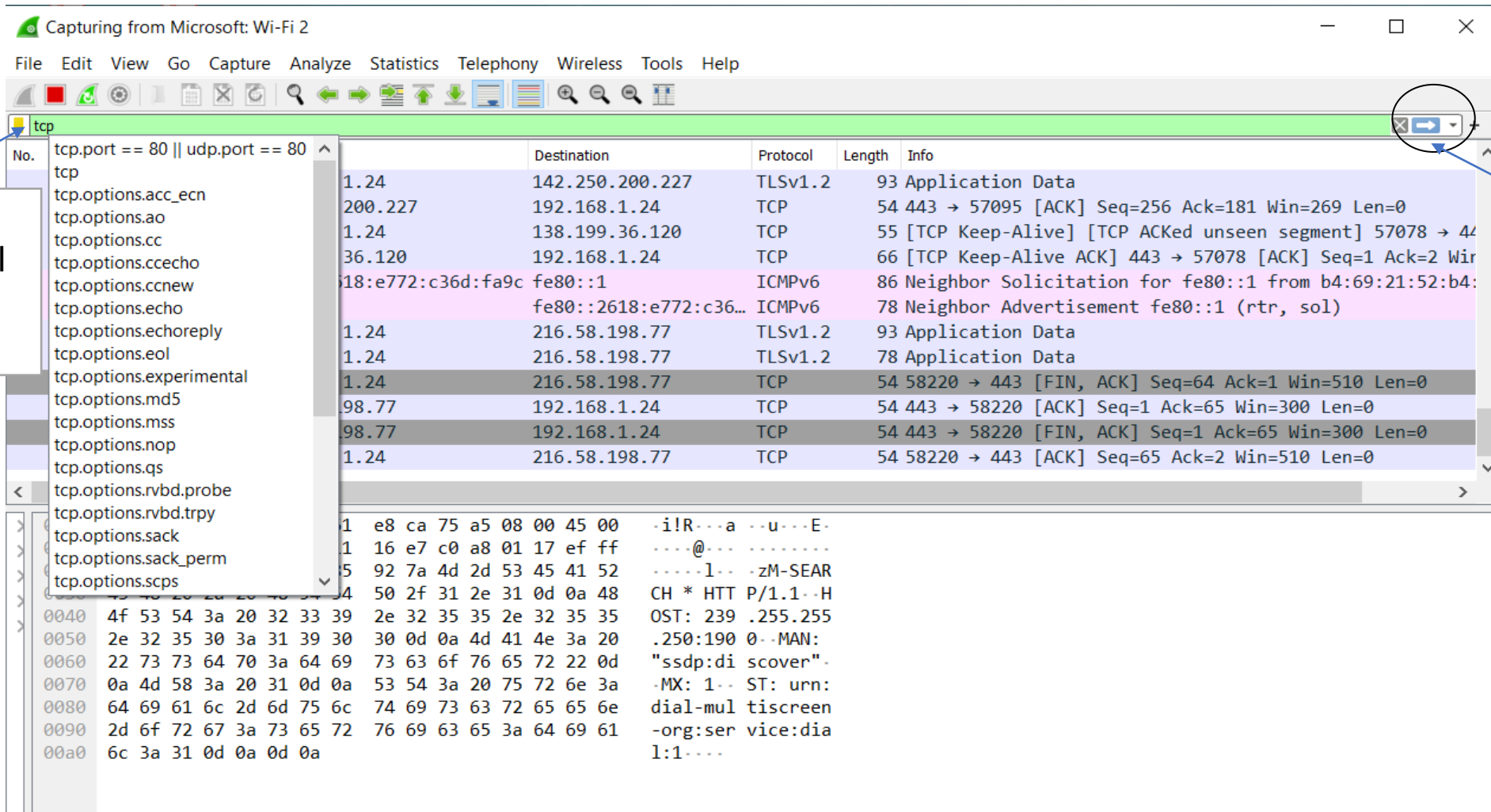
We can **save** the part of traffic when using filter

Stop filter → choose **file** menu → **export specified packet** → write the file name → save



## Capture then filter

For example, to only display TCP packets, type `tcp` into Wireshark's display filter toolbar.



Write protocol  
name

Press enter to  
filter the result

# Capture filters

filter then Capture

Capture 3

...using this filter: tcp port 80 1

Microsoft Local Area Connection\* 4

Microsoft Wi-Fi 2

Realtek PCIe GbE Family Controller: Ethernet 2

Microsoft Bluetooth Network Adapter: Bluetooth 2

VMware Virtual Ethernet Adapter: VMware Network Adapter VMnet8

VMware Virtual Ethernet Adapter: VMware Network Adapter VMnet1

Choose the network

Capturing from Microsoft Wi-Fi 2 (tcp port 80)

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help



Apply a display filter ... <Ctrl-/>

No.	Time	Source	Destination	Protocol	Length	Info
7	0.056154	192.168.1.24	142.250.200.227	TCP	54	58289 → 80 [ACK] Seq=1 Ack=1 Win=131072 Len=0
8	0.056487	192.168.1.24	142.250.200.227	OCSP	483	Request
9	0.095951	142.250.200.227	192.168.1.24	TCP	54	80 → 58288 [ACK] Seq=1 Ack=430 Win=66816 Len=0
10	0.103550	142.250.200.227	192.168.1.24	TCP	54	80 → 58289 [ACK] Seq=1 Ack=430 Win=66816 Len=0
11	0.109253	142.250.200.227	192.168.1.24	OCSP	756	Response
12	0.117863	142.250.200.227	192.168.1.24	OCSP	756	Response
13	0.151162	192.168.1.24	142.250.200.227	TCP	54	58288 → 80 [ACK] Seq=430 Ack=703 Win=130560 Len=0
14	0.166025	192.168.1.24	142.250.200.227	TCP	54	58289 → 80 [ACK] Seq=430 Ack=703 Win=130560 Len=0
15	10.123726	192.168.1.24	142.250.200.227	TCP	55	[TCP Keep-Alive] 58289 → 80 [ACK] Seq=429 Ack=703 Win=
16	10.123727	192.168.1.24	142.250.200.227	TCP	55	[TCP Keep-Alive] 58288 → 80 [ACK] Seq=429 Ack=703 Win=
17	10.168520	142.250.200.227	192.168.1.24	TCP	66	[TCP Keep-Alive ACK] 80 → 58288 [ACK] Seq=703 Ack=430
18	10.168719	142.250.200.227	192.168.1.24	TCP	66	[TCP Keep-Alive ACK] 80 → 58289 [ACK] Seq=703 Ack=430