Wireshark

2.8 Wireshark

- The best way to deeply understand the topics of this module is to see the actual protocols in action. You can do that by using a sniffer tool.

- This section will enhance your Networking and **Wireshark skills**.

2.8 Wireshark

-  As you know, Wireshark is a network sniffer and protocol analyzer.

- This means that you can use it to analyze every packet, traffic stream, or connection **that hits your computer network interface(s)**.

2.8 Wireshark

- Knowing this tool is extremely important to understand how networking works.
- Wireshark is widely used by network administrators, networking protocol researchers, and hackers.

2.8 Wireshark

- Wireshark can capture all the traffic **seen** by the network card of the computer running it.

- To understand what traffic a network card sees, you have to know that most network cards, also known as Network Interface Cards (NIC), can work in **promiscuous or monitor mode**.

2.8.1 NIC Promiscuous Mode

- During normal operations, a network card **discards** any packet addressed to another NIC. In **promiscuous mode**, a network card will accept and process **any** packet it receives.

- For example, in a hub-based network, a NIC will receive traffic addressed to other machines. The NIC usually drops these packets but accepts them while in promiscuous mode.

2.8.1 NIC Promiscuous Mode

- With the introduction of switched networks, sniffing other machines Ethernet traffic got harder. You have to perform an attack such as ARP poisoning or MAC flooding in order to do that.

- WiFi medium (the air), instead, is broadcast by nature, so it's possible to still detect traffic destined to a different host. In this chapter, we will concentrate on Ethernet traffic only.

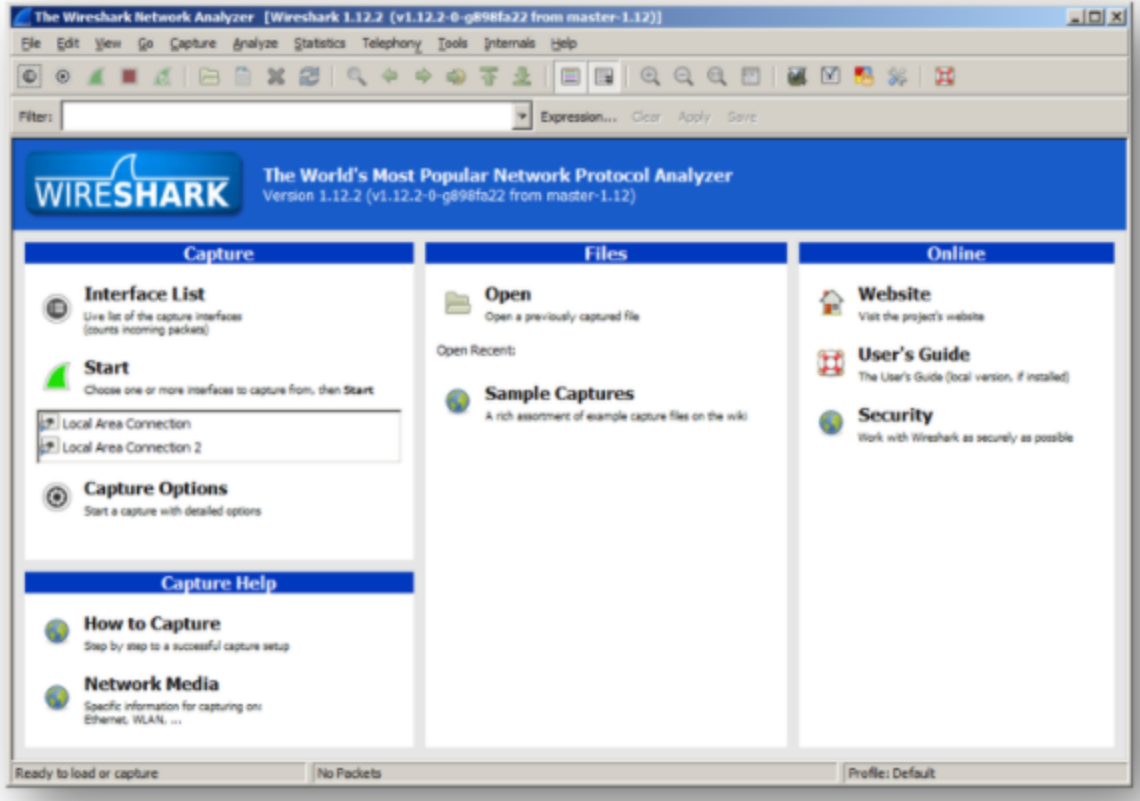
2.8.2 Configuring Wireshark

- Wireshark is free software that can run on practically all modern operating systems. You can download it from https://www.wireshark.org/ or https://www.wireshark.org/download/ for older versions.

- In the following slides, we are going to see how to configure it to use its main features. The version covered is 1.12.2.

2.8.2 Configuring Wireshark

- Here we see Wireshark's main window.



2.8.2 Configuring Wireshark

- Clicking on **Interface List** opens a window with a list of your network cards (wired, wireless, VPNs, virtual interfaces, etc.).

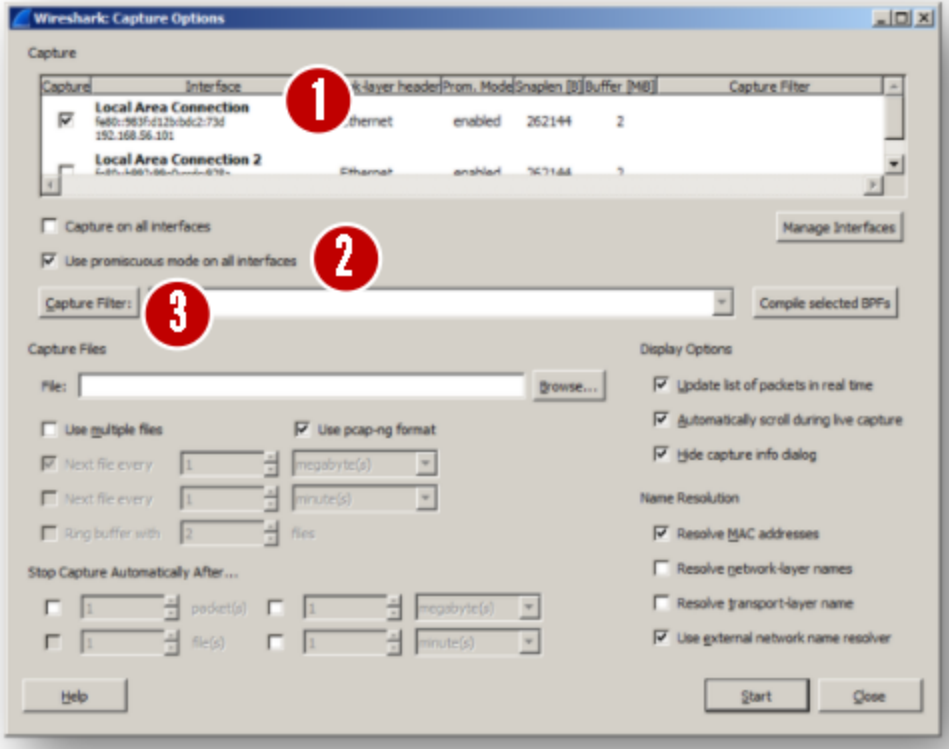
2.8.2 Configuring Wireshark

+ Clicking on **Capture Options** opens...

2.8.2 Configuring Wireshark

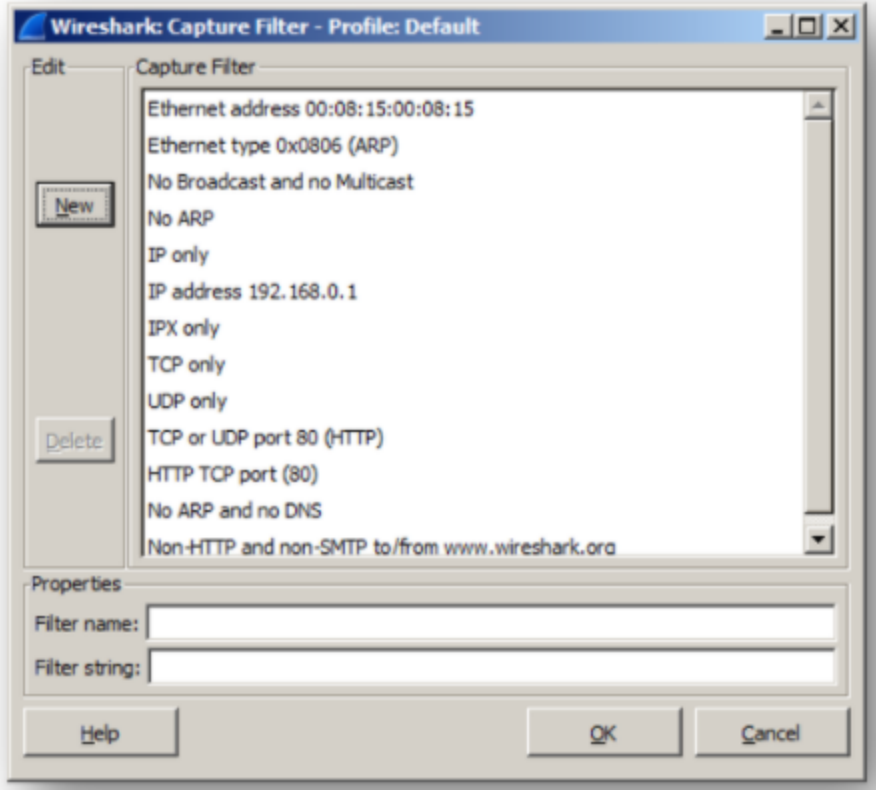
... the **capture options** window, which has a huge impact on your capture session as you can configure:

1. Which interfaces to use during the capture
2. NIC promiscuous mode
3. Capture filtering



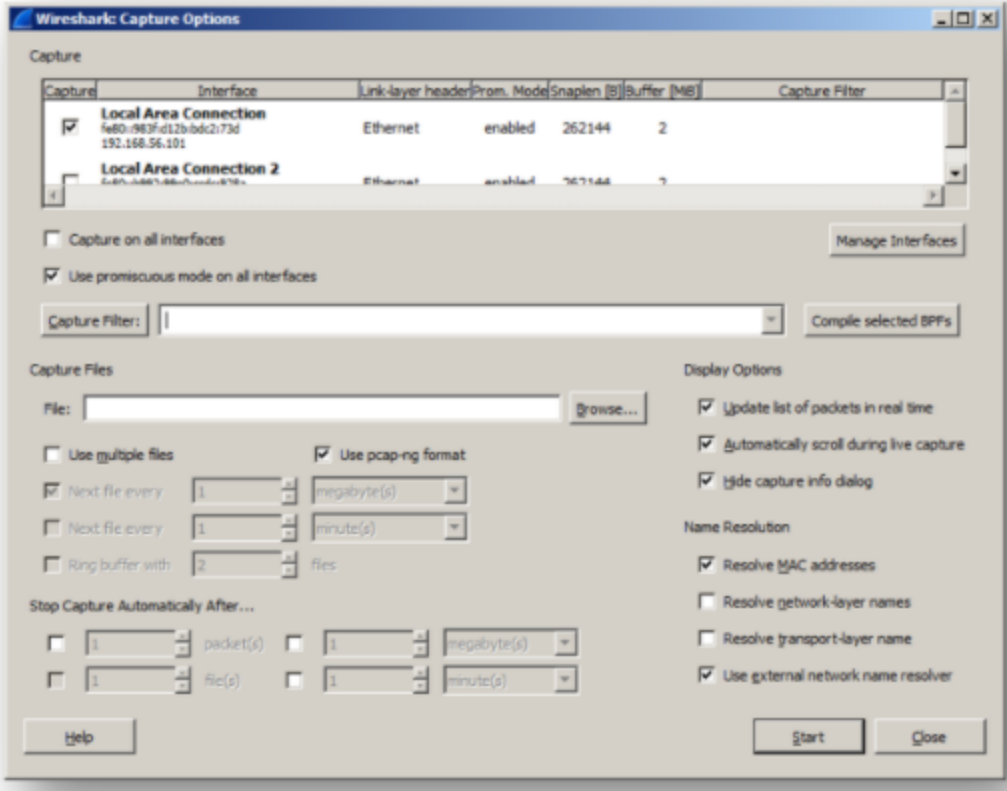
2.8.2 Configuring Wireshark

- **Capture filters** will make Wireshark discard packets that do not match the filter. These filters impact how many packets your computer must process and how big the capture file will be.
- This is very useful to limit captured traffic in high traffic networks.



2.8.2 Configuring Wireshark

- During your first captures, leave the filter blank and just click **start**.

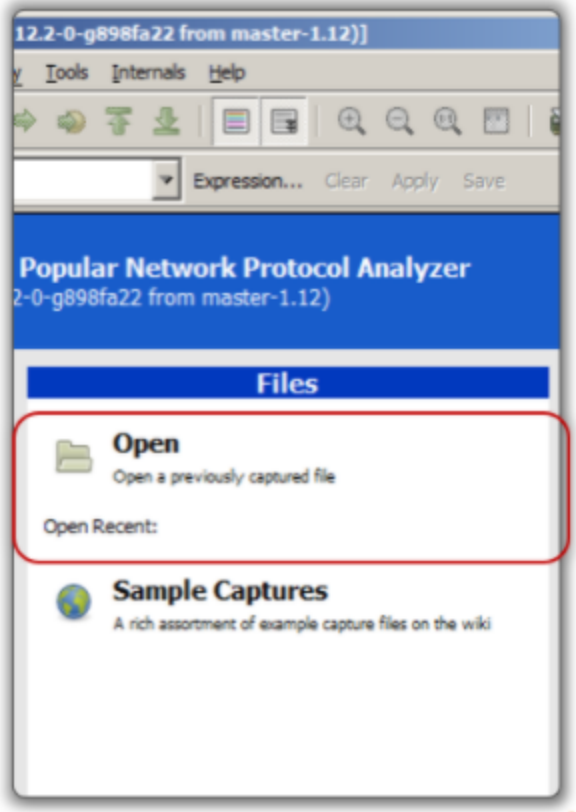


2.8.2 Configuring Wireshark

- To perform these very same operations, you can just select the capture interface and then click on **capture options** or **start** from the main window.

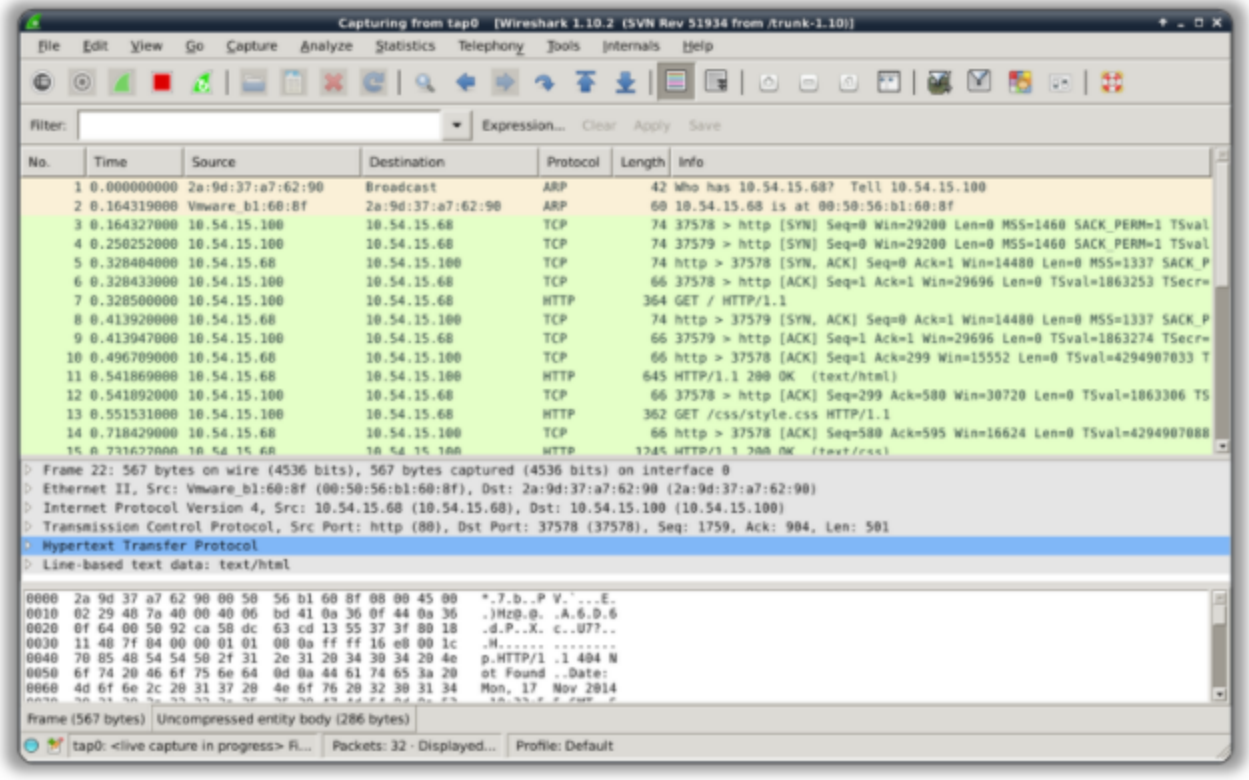
2.8.2 Configuring Wireshark

- PCAP files store an entire capture (from a previous capture session).
- If you already have a PCAP file, you can open it using this button.



2.8.3 The capture window

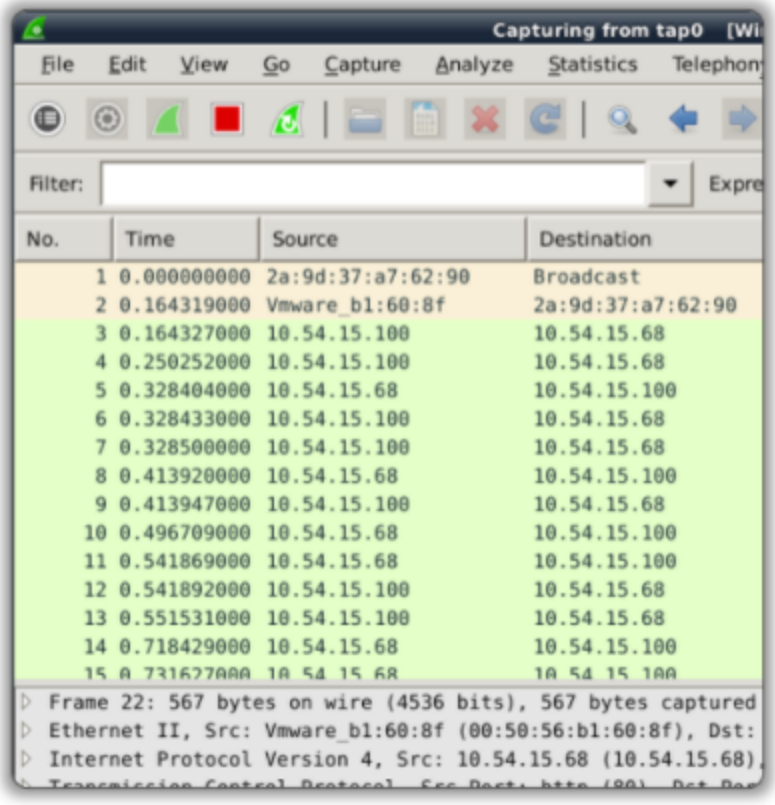
In either case, doing a live capture or opening a previous one, you will see this interface.



2.8.3 The capture window

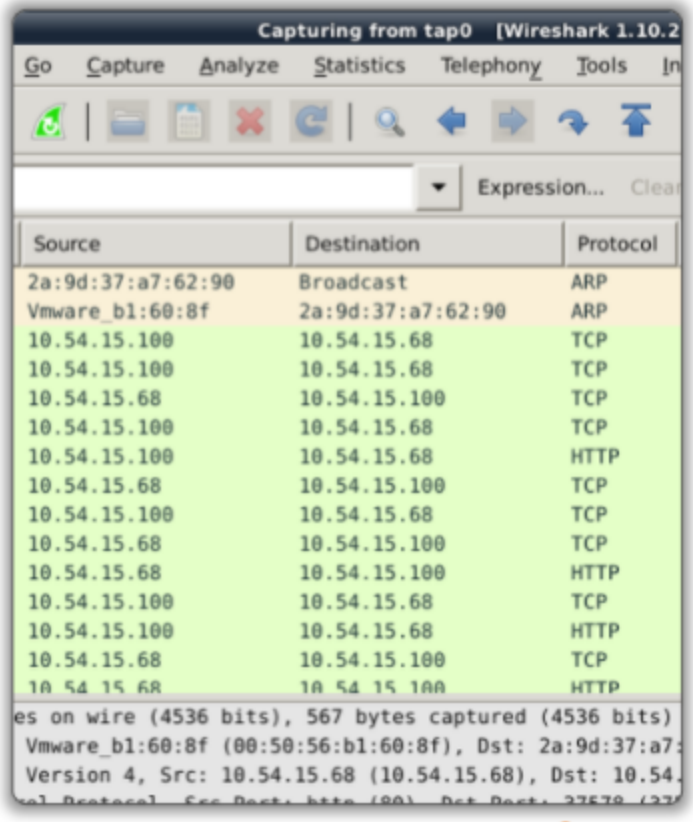
The first two columns of the upper pane contain:

- The **number** of the captured packet.
- The **arrival time** of the packet in seconds. The arrival time is relative to the start of the capture.



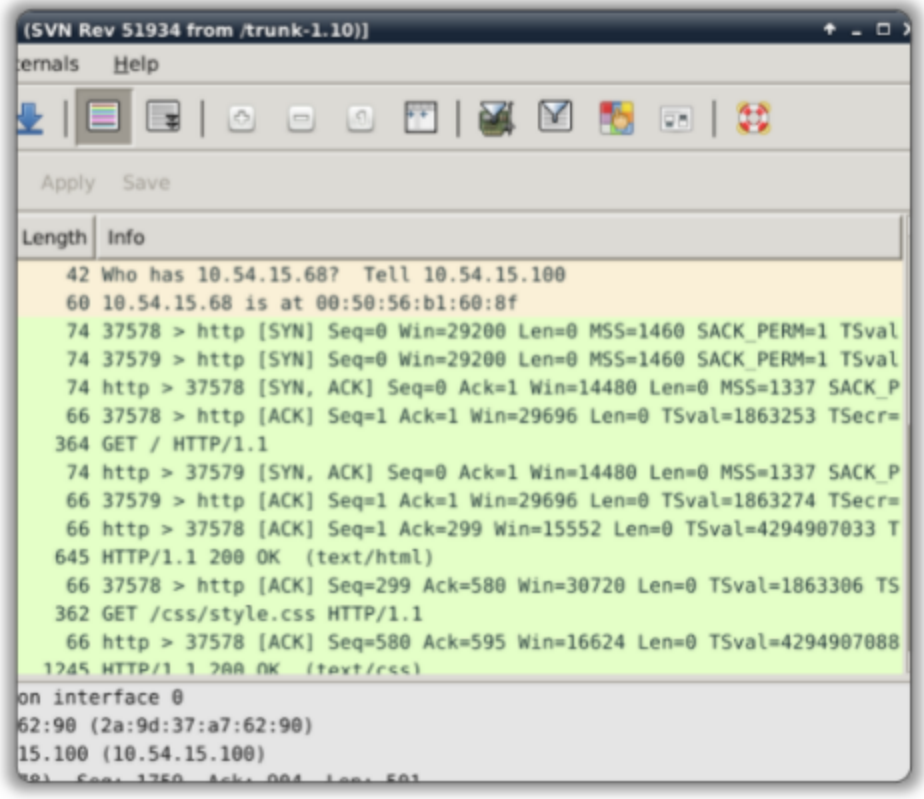
2.8.3 The capture window

- You can then see the **source, destination** and **protocol** columns.
- Note how the source and destination address vary according to the protocol.



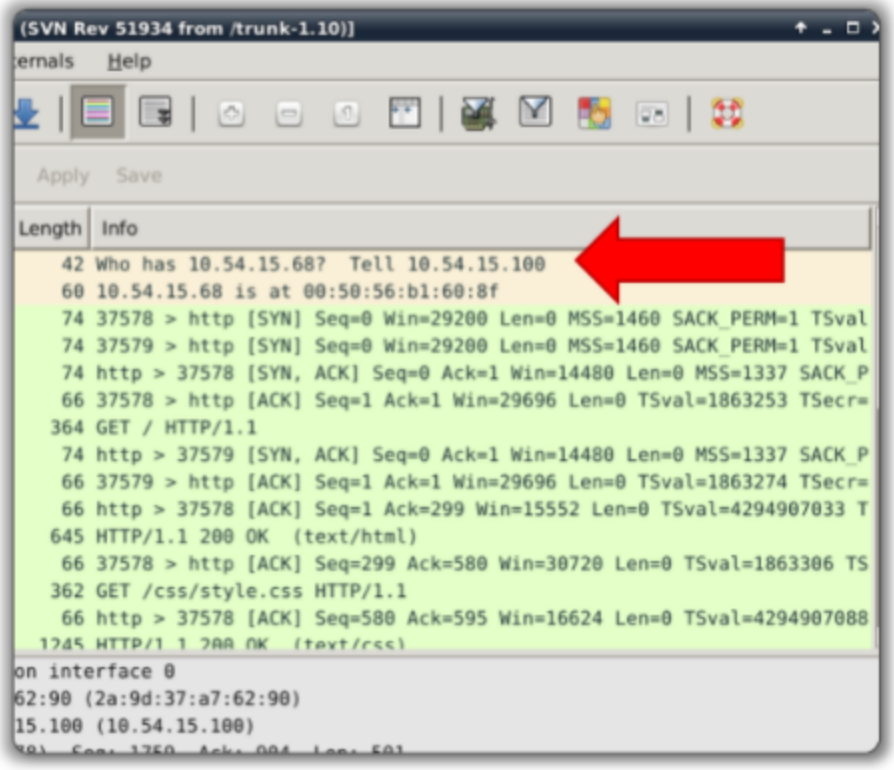
2.8.3 The capture window

- In the last two columns, you can find the **size of the packet** and some related **information**.
- The *info* column is protocol specific.



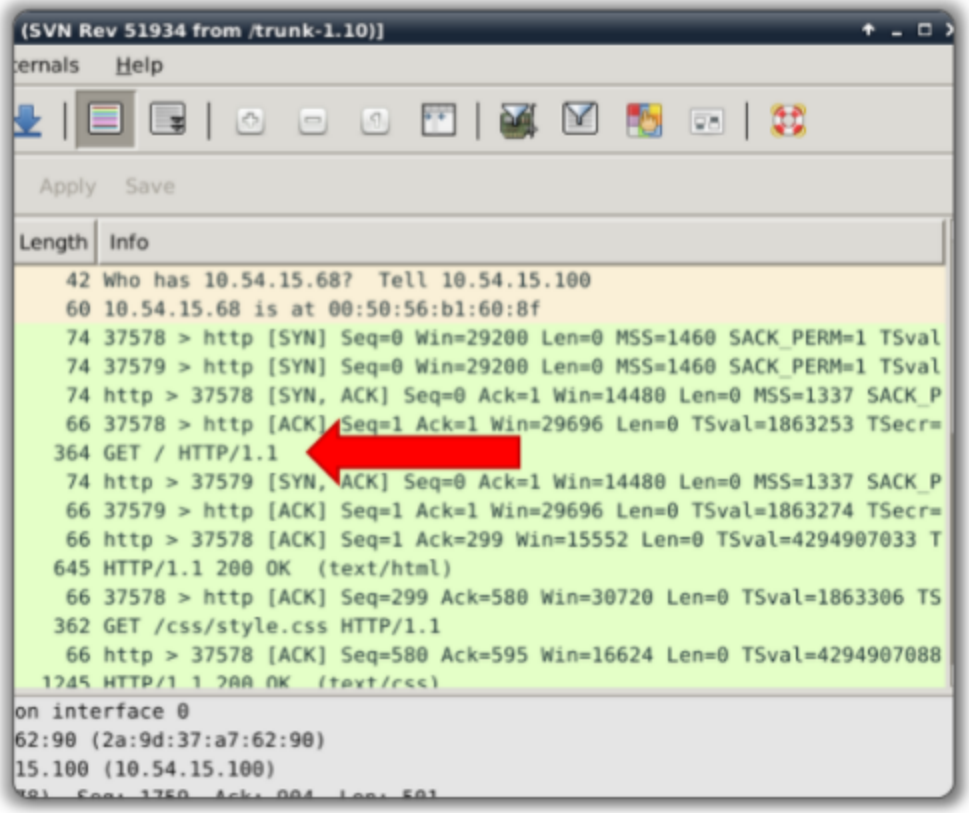
2.8.3 The capture window

- For example, the first two packets are ARP requests and replies.



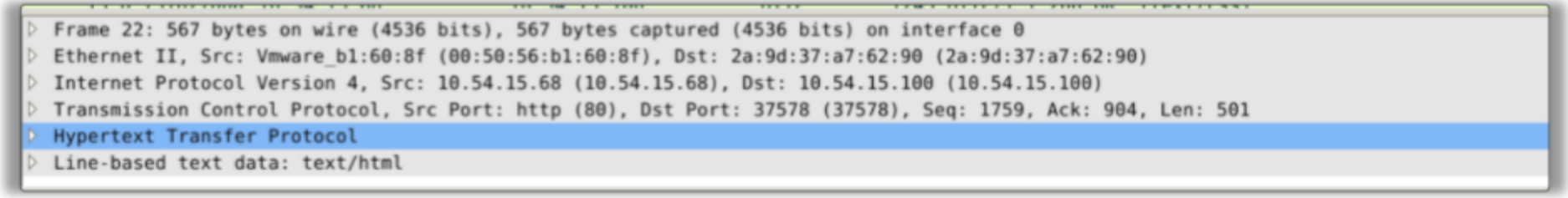
2.8.3 The capture window

- This packet is an HTTP request.



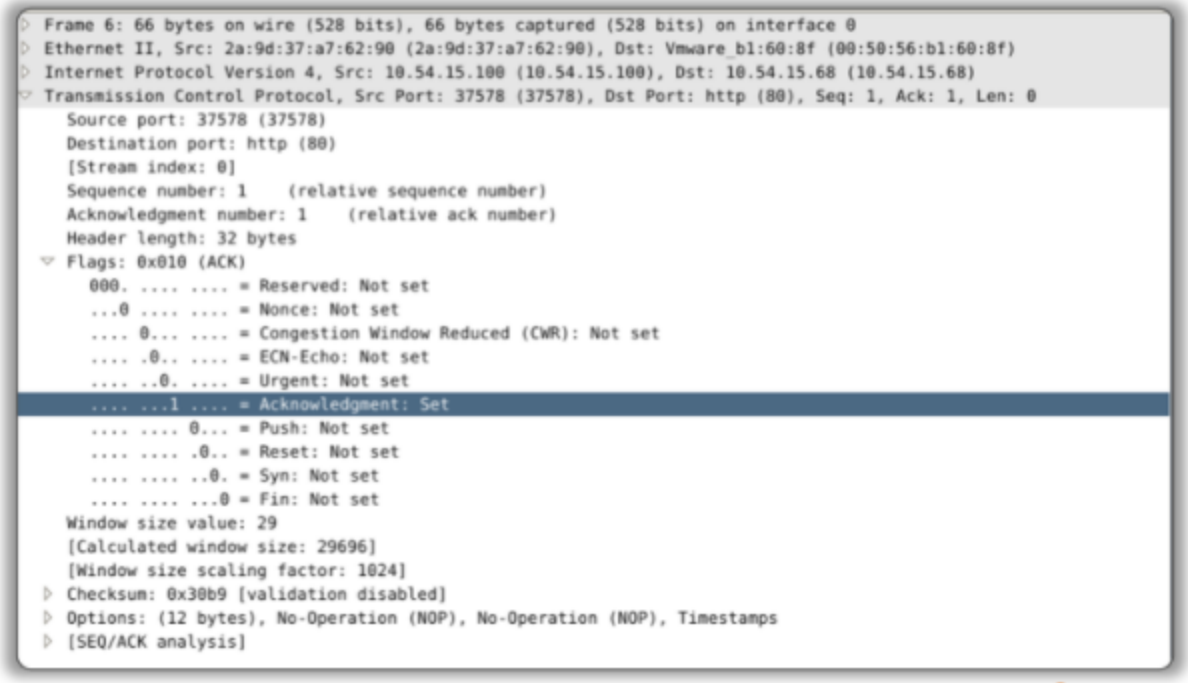
2.8.3 The capture window

- The center pane gives you access to all the protocol layers used by a packet.
- This actually allows you to read the entire packet layer by layer!



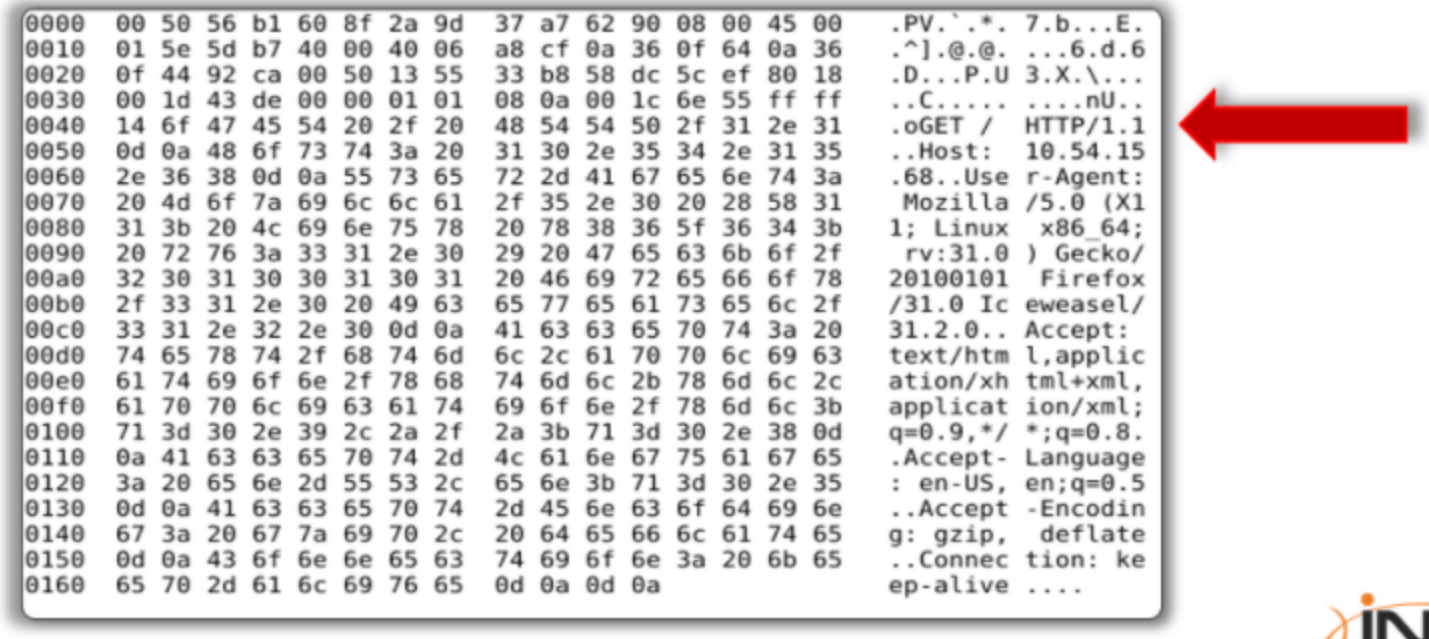
2.8.3 The capture window

- You can drill down to get any information you want about a packet.
- For example, this packet has the *ACK* TCP flag on.



2.8.3 The capture window

- In the bottom pane, you can see the actual packet payload. In this example we see an HTTP GET request.



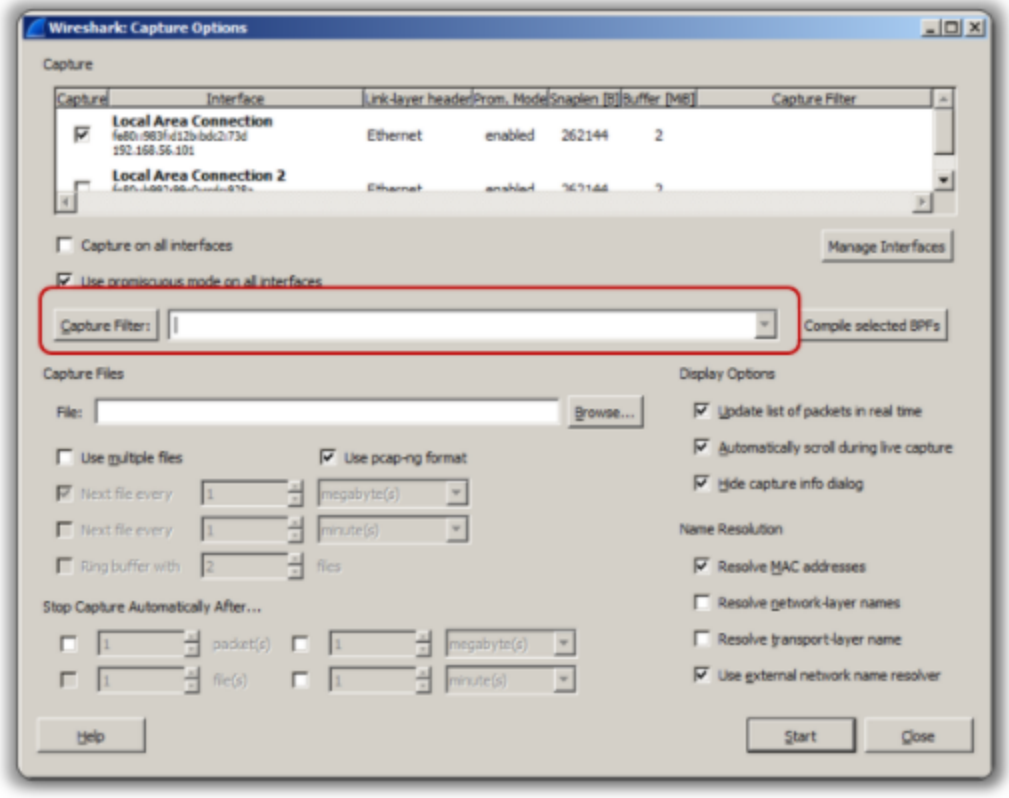
```
0000  00 50 56 b1 60 8f 2a 9d  37 a7 62 90 08 00 45 00   .PV.`.*. 7.b...E.
0010  01 5e 5d b7 40 00 40 06  a8 cf 0a 36 0f 64 0a 36   .^].@.@. ...6.d.6
0020  0f 44 92 ca 00 50 13 55  33 b8 58 dc 5c ef 80 18   .D...P.U 3.X.\...
0030  00 1d 43 de 00 00 01 01  08 0a 00 1c 6e 55 ff ff   ..C..... ....nU..
0040  14 6f 47 45 54 20 2f 20  48 54 54 50 2f 31 2e 31   .oGET /  HTTP/1.1
0050  0d 0a 48 6f 73 74 3a 20  31 30 2e 35 34 2e 31 35   ..Host:  10.54.15
0060  2e 36 38 0d 0a 55 73 65  72 2d 41 67 65 6e 74 3a   .68..Use r-Agent:
0070  20 4d 6f 7a 69 6c 6c 61  2f 35 2e 30 20 28 58 31    Mozilla /5.0 (X1
0080  31 3b 20 4c 69 6e 75 78  20 78 38 36 5f 36 34 3b   1; Linux  x86_64;
0090  20 72 76 3a 33 31 2e 30  29 20 47 65 63 6b 6f 2f    rv:31.0 ) Gecko/
00a0  32 30 31 30 30 31 30 31  20 46 69 72 65 66 6f 78   20100101  Firefox
00b0  2f 33 31 2e 30 20 49 63  65 77 65 61 73 65 6c 2f   /31.0 Ic eweasel/
00c0  33 31 2e 32 2e 30 0d 0a  41 63 63 65 70 74 3a 20   31.2.0.. Accept:
00d0  74 65 78 74 2f 68 74 6d  6c 2c 61 70 70 6c 69 63   text/htm l,applic
00e0  61 74 69 6f 6e 2f 78 68  74 6d 6c 2b 78 6d 6c 2c   ation/xh tml+xml,
00f0  61 70 70 6c 69 63 61 74  69 6f 6e 2f 78 6d 6c 3b   applicat ion/xml;
0100  71 3d 30 2e 39 2c 2a 2f  2a 3b 71 3d 30 2e 38 0d   q=0.9,*/ *;q=0.8.
0110  0a 41 63 63 65 70 74 2d  4c 61 6e 67 75 61 67 65   .Accept- Language
0120  3a 20 65 6e 2d 55 53 2c  65 6e 3b 71 3d 30 2e 35   : en-US, en;q=0.5
0130  0d 0a 41 63 63 65 70 74  2d 45 6e 63 6f 64 69 6e   ..Accept -Encodin
0140  67 3a 20 67 7a 69 70 2c  20 64 65 66 6c 61 74 65   g: gzip,  deflate
0150  0d 0a 43 6f 6e 6e 65 63  74 69 6f 6e 3a 20 6b 65   ..Connec tion: ke
0160  65 70 2d 61 6c 69 76 65  0d 0a 0d 0a               ep-alive ....
```

2.8.4 Filtering

- A traffic capture can be overwhelming, even on a network with just a couple of dozens of nodes.
- Wireshark can **filter** traffic at **capture** or at **display** time.
- Each method has its own pros and cons.

2.8.4.1 Capture Filters

- You can set capture filters **before** starting the capture so that Wireshark will capture only packets matching the filters.



2.8.4.1 Capture Filters

- Here are some basic capture filters.

Syntax	Description
ip	Only packets using IP as layer 3 protocol.
not ip	The opposite of the previous syntax.
tcp port 80	Packets where the source or destination TCP port is 80.
net 192.168.54.0/24	Packets from and to the specified network.
src port 1234	The source port must be 1234; the transport protocol does not matter.
src net 192.168.1.0/24	The source IP address must be in the specified network.
host 192.168.45.65	All the packets from or to the specified host.
host www.examplehost.com	All the packets from or to the specified hostname.

2.8.4.1 Capture Filters

- Capture filters will downsize the amount of traffic gathered.
- The final capture will be **smaller**, and it will contain **only** the **needed traffic**.

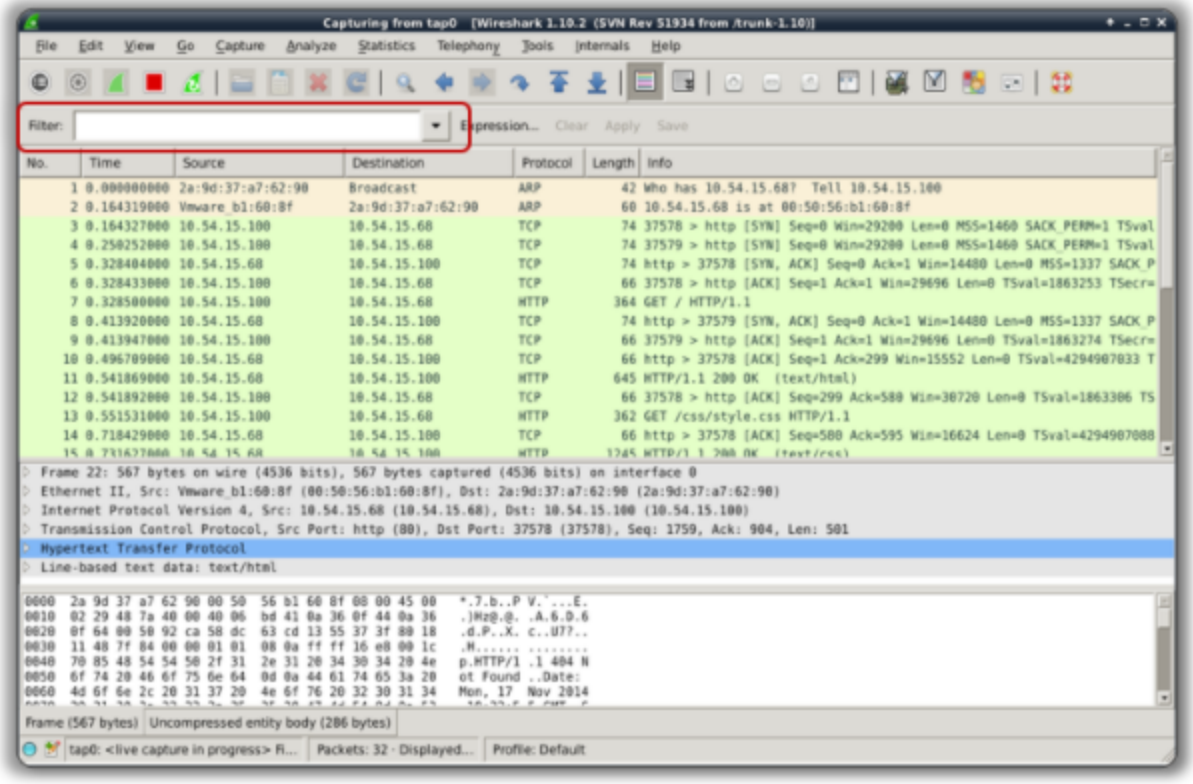
2.8.4.2 Display Filters

- However, capture filters might not catch interesting traffic! Display filters instead allow you to inspect and apply very granular filters to every field of the captured packets. Wireshark then displays only the packets matching the filters.

- You can always remove or fine tune a display filter, something you can't do with the capture filter (you would have to re-start the capture from scratch).

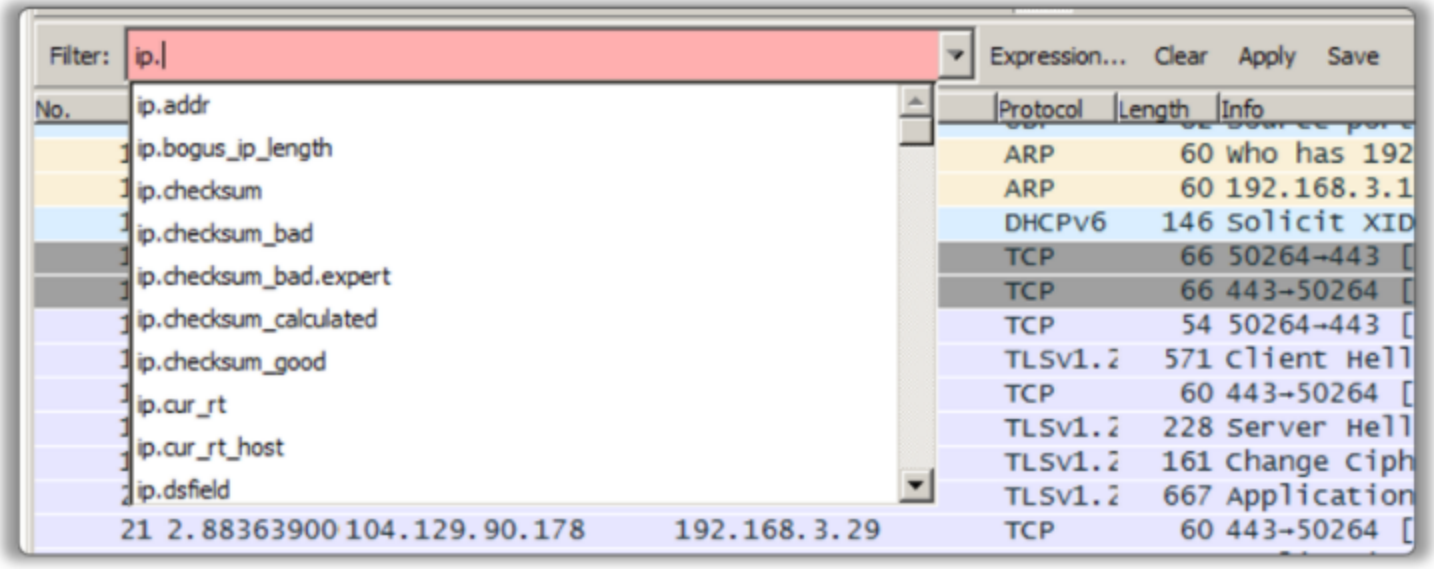
2.8.4.2 Display Filters

- You can use the filter textbox to apply a display filter.



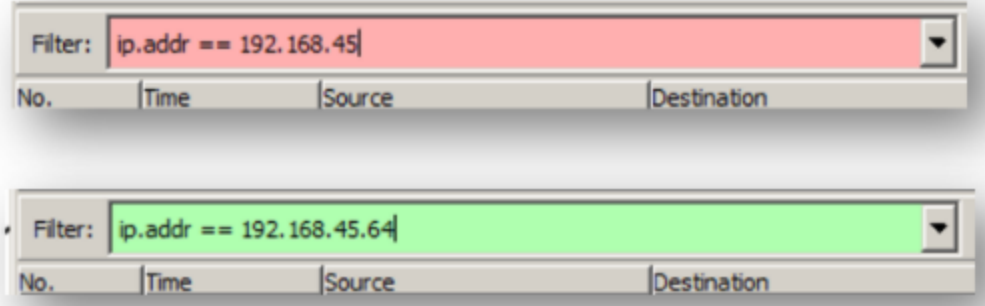
2.8.4.2 Display Filters

- You can start typing a filter and Wireshark will give you valid protocol fields.



2.8.4.2 Display Filters

- The background of the text-box will turn red if the filter is invalid or green when the filter is valid.



2.8.4.2 Display Filters

- A display filter is made by:

Syntax	Description
<Protocolname>	Displays any packet using that protocol.
<Protocolname>[.field]	Displays any packet with the specified field present.
<Protocolname>[.field] [operand value]	Displays any packet whose protocol field matches the operand and value.
<Protocolname>[.field] AND <Protocolname>[.field] [operand value]	You can combine multiple expressions by using logical operators.

2.8.4.2 Display Filters

- Below is an example.

Syntax	Description
ip	Displays IP packets.
ip.addr	Displays IP packets with a populated source or destination address.
ip.addr == 192.168.12.13	Displays IP packets with 192.168.12.13 as source or destination address.
ip.addr == 192.168.12.13 or arp	The above or ARP packets.

2.8.4.2 Display Filters

+ You can find Wireshark display filter reference here.

+ For any other information, please refer to the Wireshark User's Guide.

https://www.wireshark.org/docs/dfref/
https://www.wireshark.org/docs/wsug_html_chunked/

2.8.5 Sample Traffic Captures

+ If you want to practice these topics a little more, you can record some traffic from your computer or you can download a capture from Wireshark website.

http://wiki.wireshark.org/SampleCaptures

References

- Wireshark: https://www.wireshark.org/
- Wireshark Download: https://www.wireshark.org/download/
- Wireshark Display Filter Reference: https://www.wireshark.org/docs/dfref/
- Wireshark User's Guide: https://www.wireshark.org/docs/wsug_html_chunked/
- Wireshark Sample Captures: http://wiki.wireshark.org/SampleCaptures