# Wireshark: The Basics

### Learned:

- You can use Wireshark's filter menu to search for specific protocols such as TCP and UDP to analyze traffic.

- Wireshark lets you open PCAP files to review older captured data, which helps identify traffic patterns and anomalies over time.

- Capturing exact packets is valuable when you detect suspicious behavior these can serve as evidence or clues.

- Each packet includes multiple headers from Layers 1–5 of the OSI model, such as

- Layer 1: Physical bits/bytes on the wire
- Layer 2: Source and destination MAC addresses
- Layer 3: IP addresses for source and destination

### Use Cases

Wireshark is one of the most potent traffic analyzer tools available in the wild. There are multiple purposes for its use:

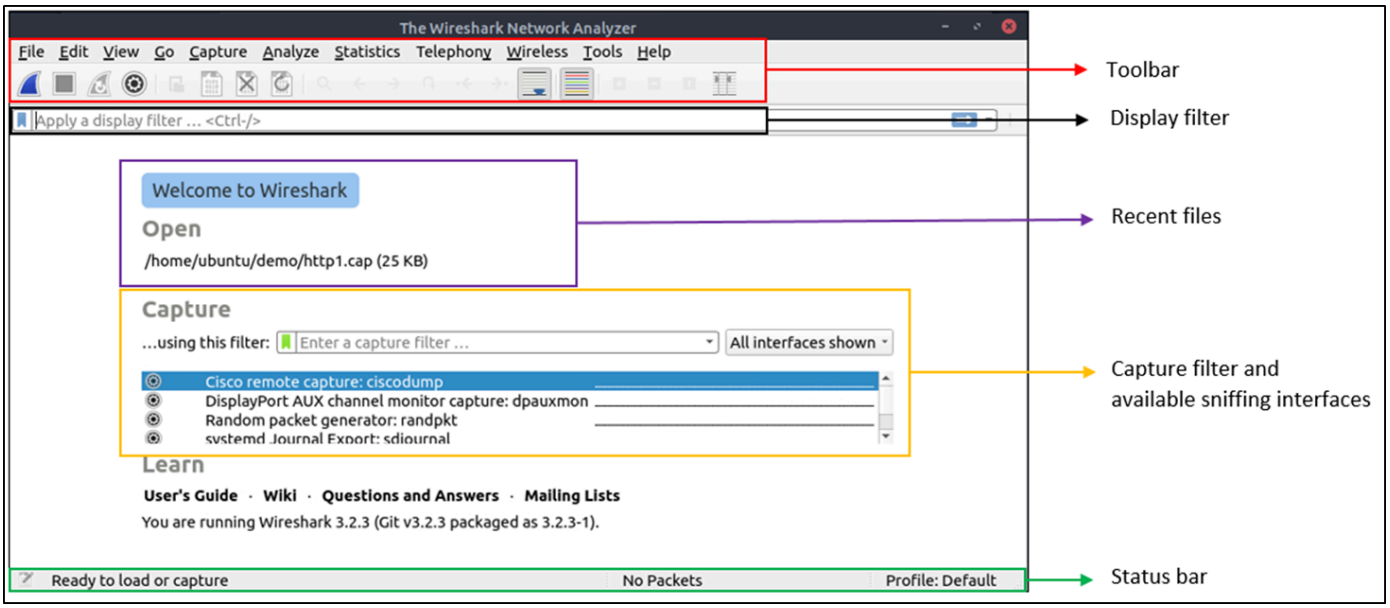
- Detecting and troubleshooting network problems, such as network load failure points and congestion.
- Detecting security anomalies, such as rogue hosts, abnormal port usage, and suspicious traffic.
- Investigating and learning protocol details, such as response codes and payload data.

# Gul and Data

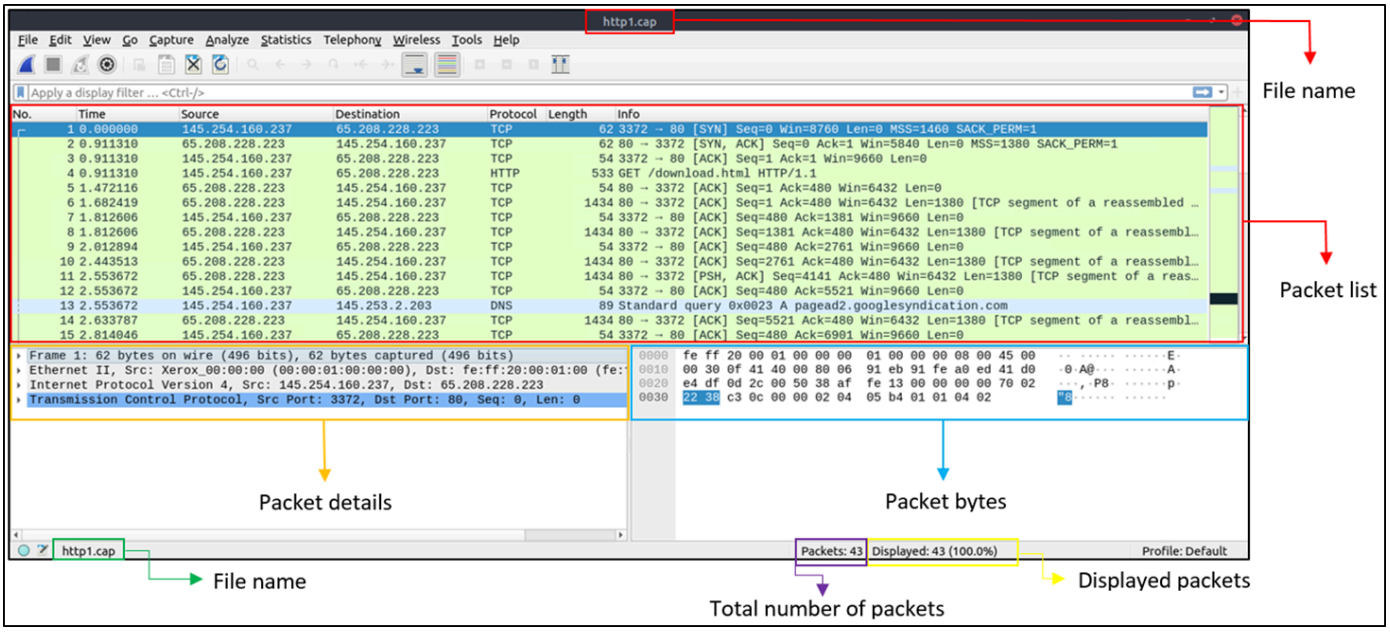
Wireshark GUI opens with a single all-in-one page, which helps users investigate the traffic in multiple ways. At first glance, five sections stand out.

| | |
|---|---|
| **Toolbar** | The main toolbar contains multiple menus and shortcuts for packet sniffing and processing, including filtering, sorting, summarising, exporting and merging. |
| **Display Filter Bar** | The main query and filtering section. |
| **Recent Files** | List of the recently investigated files. You can recall listed files with a double-click. |

| | |
|---|---|
| **Capture Filter and Interfaces** | Capture filters and available sniffing points (network interfaces). The network interface is the connection point between a computer and a network. The software connection (e.g., lo, eth0 and ens33) enables networking hardware. |
| **Status Bar** | Tool status, profile and numeric packet information. |




## Loading PCAP Files




Now, we can see the processed filename, detailed number of packets and packet details. Packet details are shown in three different panes, which allow us to discover them in different formats.

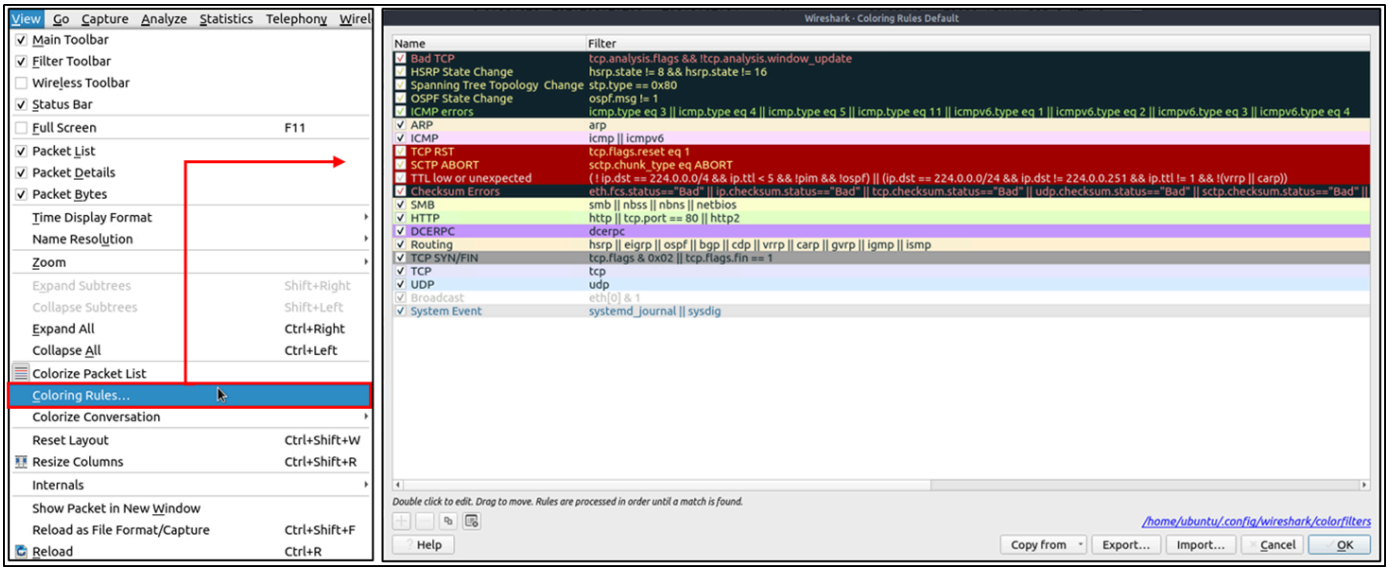
| | |
|---|---|
| Packet List Pane | Summary of each packet (source and destination addresses, protocol, and packet info). You can click on the list to choose a packet for further investigation. Once you select a packet, the details will appear in the other panels. |

| | |
|---|---|
| Packet Details Pane | Detailed protocol breakdown of the selected packet. |
| Packet Bytes Pane | Hex and decoded ASCII representation of the selected packet. It highlights the packet field depending on the clicked section in the details pane. |

## Coloring Packets

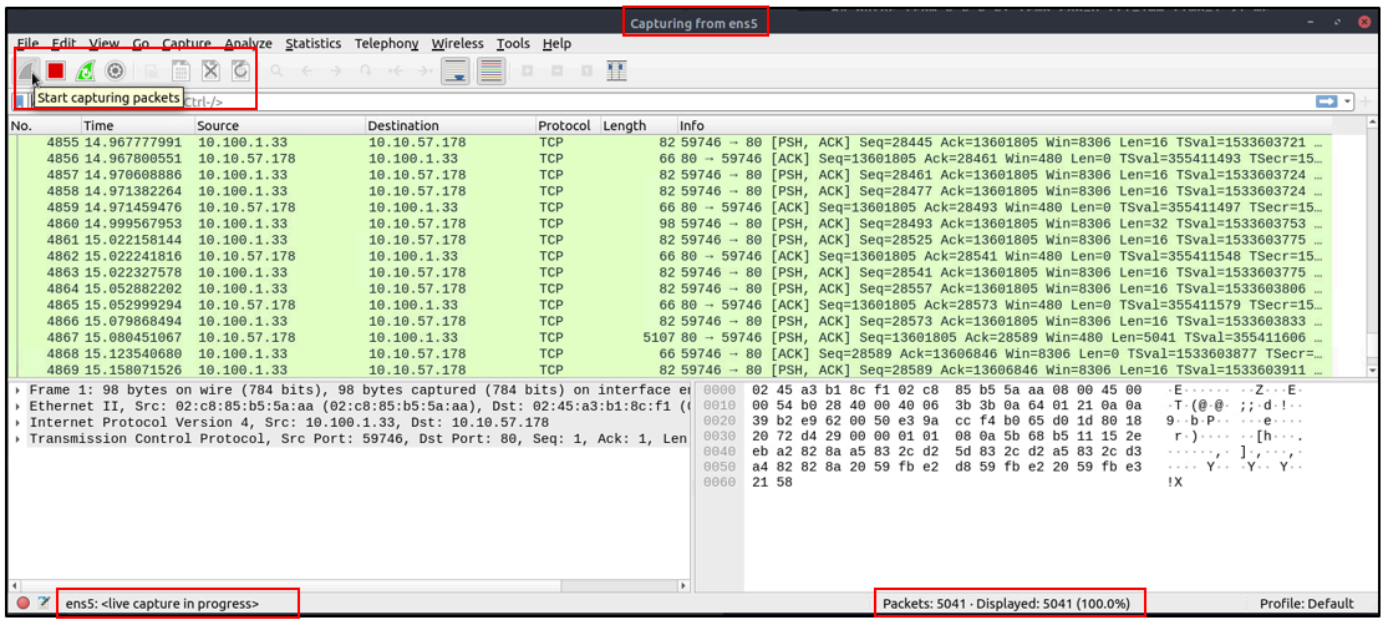
Along with quick packet information, Wireshark also color packets in order of different conditions and the protocol to spot anomalies and protocols in captures quickly. This glance at packet information can help track down exactly what you're looking for during analysis. You can create custom colour rules to spot events of interest by using display filters.

Wireshark has two types of packet colouring methods: temporary rules that are only available during a program session and permanent rules that are saved under the preference file (profile) and available for the next program session. You can use the "right-click menu" or **"View --> Coloring Rules"** menu to create permanent coloring rules. The **"Colorize Packet List"** menu activates/deactivates the coloring rules. Temporary packet coloring is done with the "right-click menu" or **"View --> Conversation Filter"** menu.



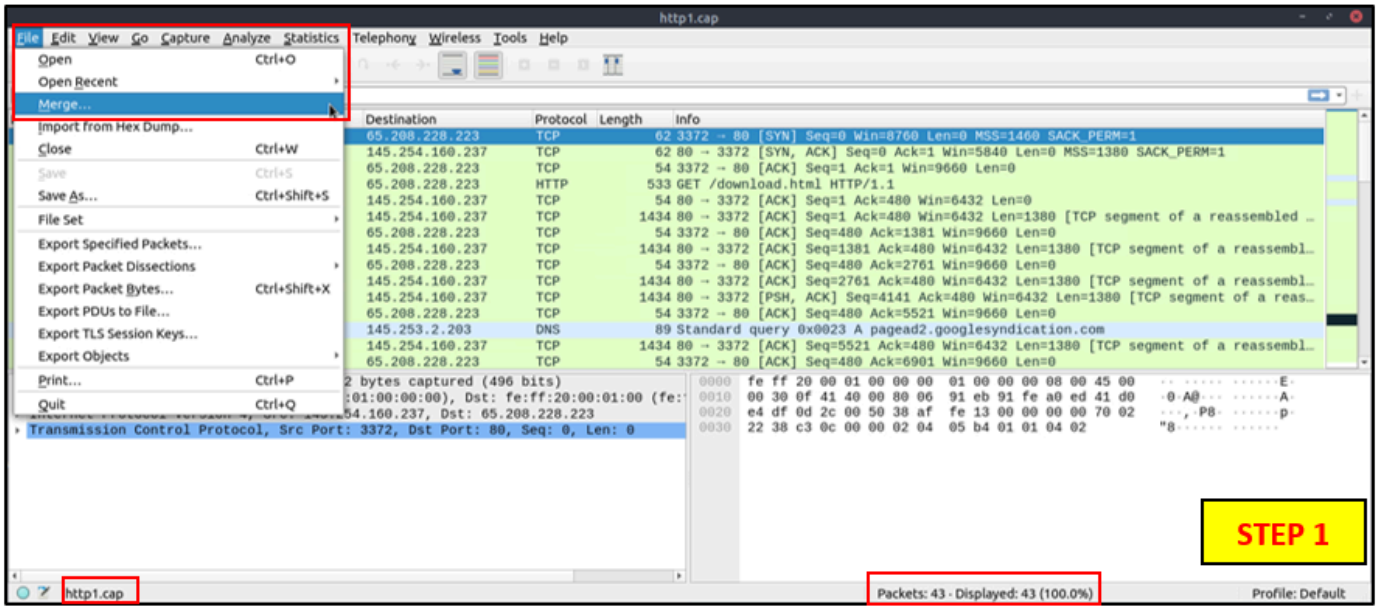

## Traffic Sniffing

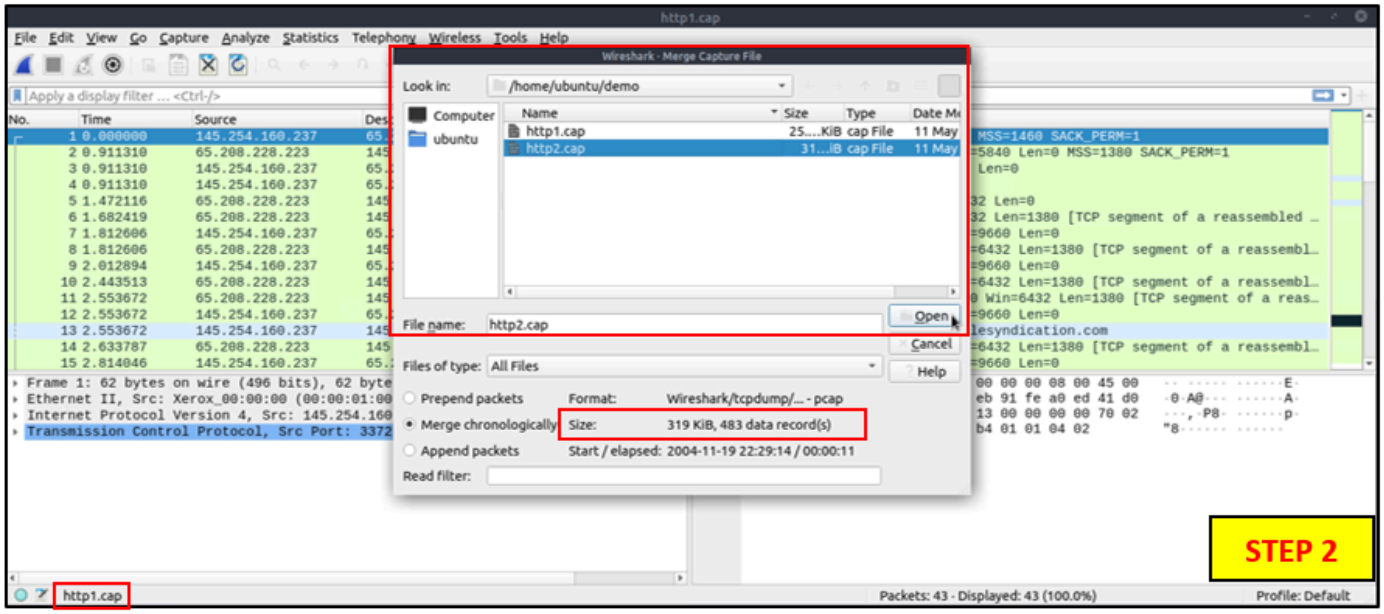
You can use the blue **"shark button"** to start network sniffing (capturing traffic), the red button will stop the sniffing, and the green button will restart the sniffing process. The status bar will also provide the used sniffing interface and the number of collected packets.

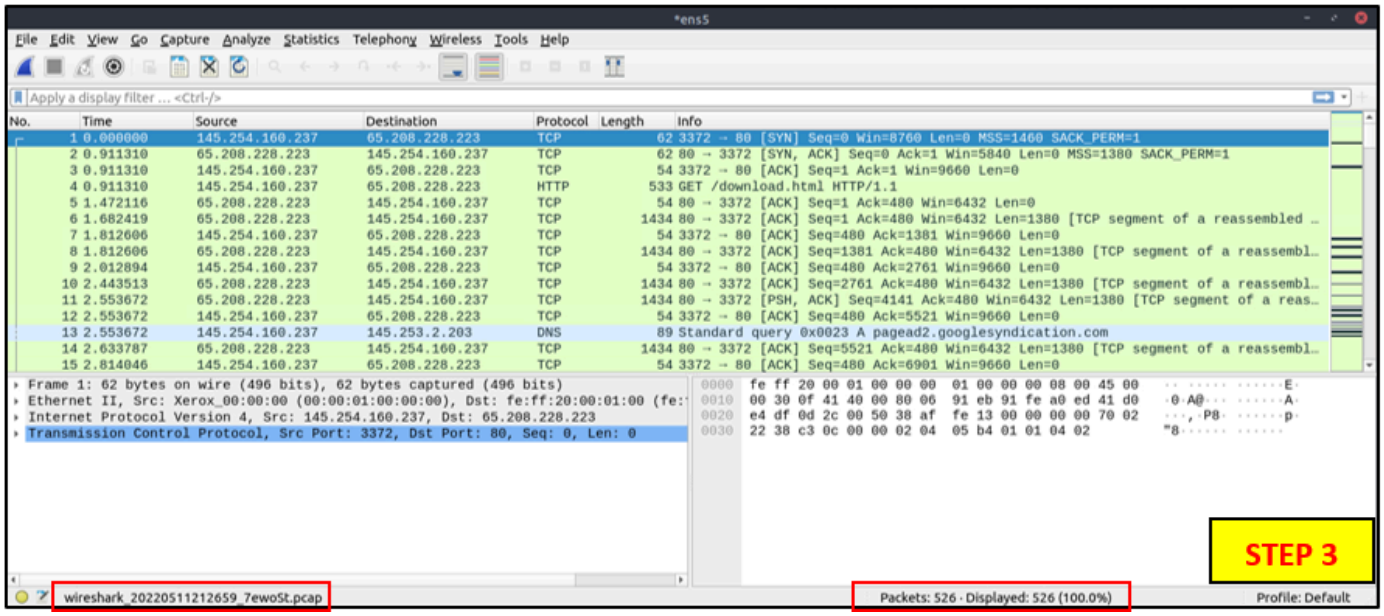



# Merge PCAP Files

Wireshark can combine two pcap files into one single file. You can use the **"File --> Merge"** menu path to merge a pcap with the processed one. When you choose the second file, Wireshark will show the total number of packets in the selected file. Once you click "open", it will merge the existing pcap file with the chosen one and create a new pcap file.

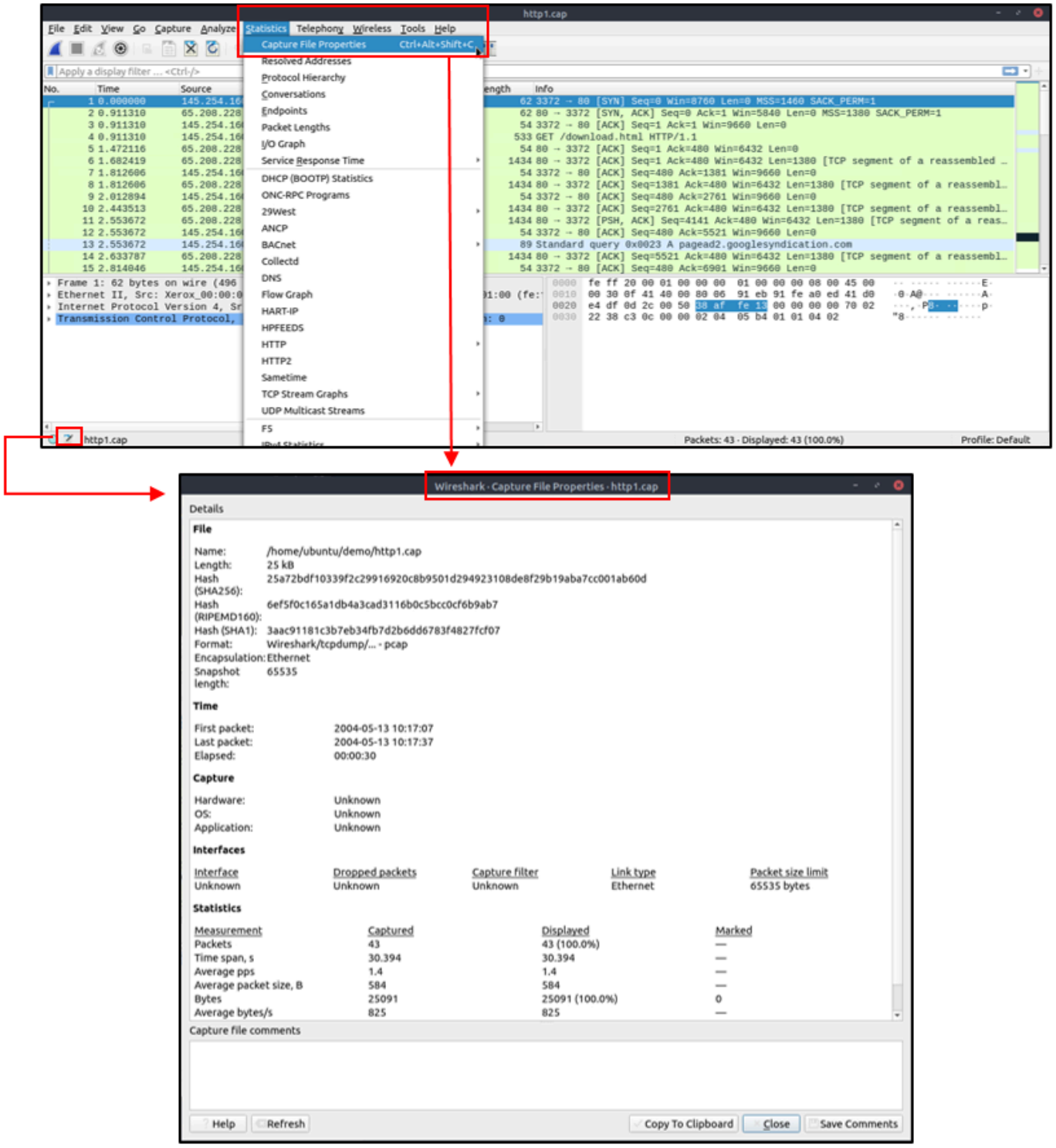


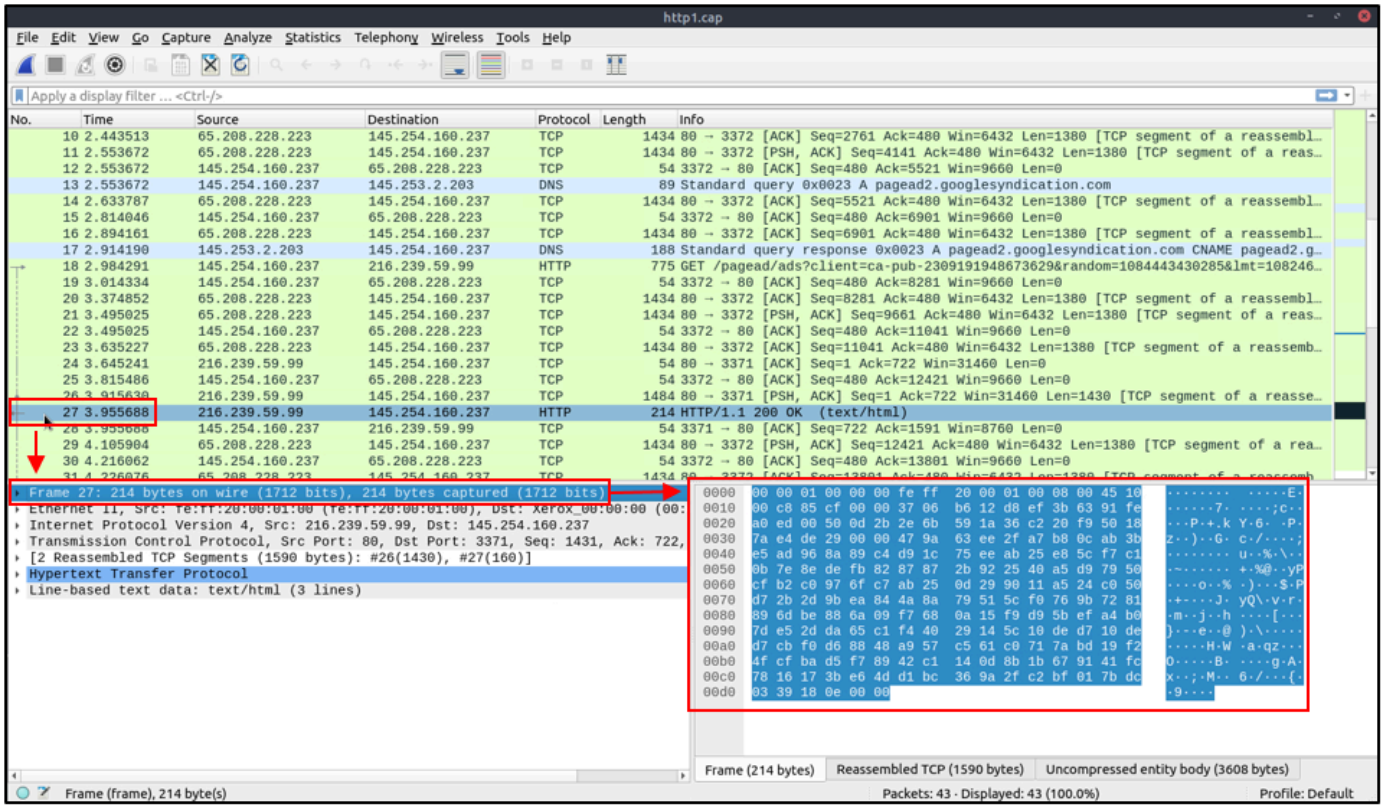

# View File Details

To identify the file, classify and prioritize it. You can view the details by following "**Statistics --> Capture File Properties**" or by clicking the **"pcap icon located on the left bottom"** of the window.

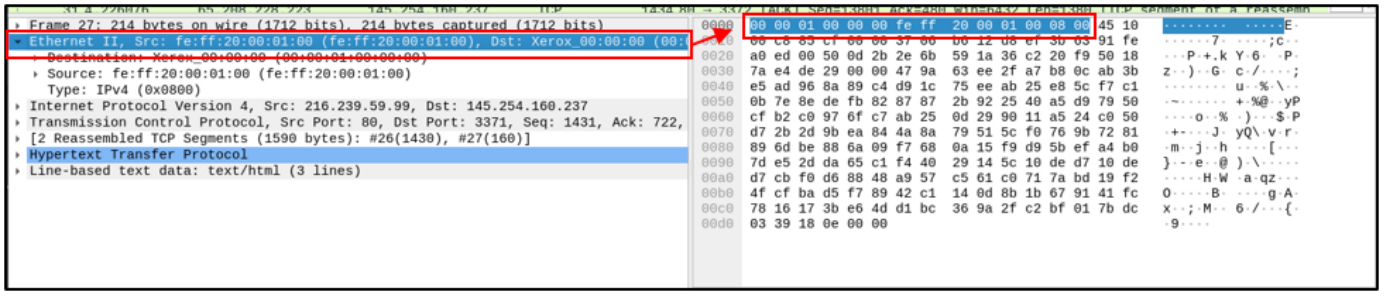



## Packet Details

You can click on a packet in the packet list pane to open its details (double-click will open details in a new window). Packets consist of 5 to 7 layers based on the OSI model.




Each time you click a detail, it will highlight the corresponding part in the packet bytes pane.




**The Frame (Layer 1):** This will show you what frame/packet you are looking at and details specific to the Physical layer of the OSI model.

```
Frame 27: 214 bytes on wire (1712 bits), 214 bytes captured (1712 bits)
    Encapsulation type: Ethernet (1)
    Arrival Time: May 13, 2004 06:17:11.266912000 Eastern Daylight Time
    [Time shift for this packet: 0.000000000 seconds]
    Epoch Time: 1084443431.266912000 seconds
    [Time delta from previous captured frame: 0.040058000 seconds]
    [Time delta from previous displayed frame: 0.040058000 seconds]
    [Time since reference or first frame: 3.955688000 seconds]
    Frame Number: 27
    Frame Length: 214 bytes (1712 bits)
    Capture Length: 214 bytes (1712 bits)
    [Frame is marked: False]
    [Frame is ignored: False]
    [Protocols in frame: eth:ethertype:ip:tcp:http:data-text-lines]
    [Coloring Rule Name: HTTP]
    [Coloring Rule String: http || tcp.port == 80 || http2]
```

**Source [MAC] (Layer 2):** This will show you the source and destination MAC Addresses; from the Data Link layer of the OSI model.

```
Ethernet II, Src: fe:ff:20:00:01:00 (fe:ff:20:00:01:00), Dst: Xerox_00:00:00 (00:00:01:00:00:00)
  > Destination: Xerox_00:00:00 (00:00:01:00:00:00)
  > Source: fe:ff:20:00:01:00 (fe:ff:20:00:01:00)
    Type: IPv4 (0x0800)
```

**Source [IP] (Layer 3):** This will show you the source and destination IPv4 Addresses; from the Network layer of the OSI model.

```
Internet Protocol Version 4, Src: 216.239.59.99, Dst: 145.254.160.237
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x10 (DSCP: Unknown, ECN: Not-ECT)
    Total Length: 200
    Identification: 0x85cf (34255)
  > Flags: 0x0000
    Fragment offset: 0
    Time to live: 55
    Protocol: TCP (6)
    Header checksum: 0xb612 [validation disabled]
    [Header checksum status: Unverified]
    Source: 216.239.59.99
    Destination: 145.254.160.237
```

**Protocol (Layer 4):** This will show you details of the protocol used (UDP/TCP) and source and destination ports; from the Transport layer of the OSI model.

```
Transmission Control Protocol, Src Port: 80, Dst Port: 3371, Seq: 778787098, Ack: 918692089, Len: 160
    Source Port: 80
    Destination Port: 3371
    [Stream index: 1]
    [TCP Segment Len: 160]
    Sequence number: 778787098
    [Next sequence number: 778787258]
    Acknowledgment number: 918692089
    0101 .... = Header Length: 20 bytes (5)
  > Flags: 0x018 (PSH, ACK)
    Window size value: 31460
    [Calculated window size: 31460]
    [Window size scaling factor: -1 (unknown)]
    Checksum: 0xde29 [unverified]
    [Checksum Status: Unverified]
    Urgent pointer: 0
  > [SEQ/ACK analysis]
  > [Timestamps]
    TCP payload (160 bytes)
    TCP segment data (160 bytes)
```

**Protocol Errors:** This continuation of the 4th layer shows specific segments from TCP that needed to be reassembled.

```
[2 Reassembled TCP Segments (1590 bytes): #26(1430), #27(160)]
    [Frame: 26, payload: 0-1429 (1430 bytes)]
    [Frame: 27, payload: 1430-1589 (160 bytes)]
    [Segment count: 2]
    [Reassembled TCP length: 1590]
    [Reassembled TCP Data: 485454502f312e3120323030204f4b0d0a5033503a20706f…]
```

**Application Protocol (Layer 5):** This will show details specific to the protocol used, such as HTTP, FTP, and SMB. From the Application layer of the OSI model.

```
Hypertext Transfer Protocol
  > HTTP/1.1 200 OK\r\n
    P3P: policyref="http://www.googleadservices.com/pagead/p3p.xml", CP="NOI DEV PSA PSD IVA PVD OTP OUR OTR IND OTC"\r\n
    Content-Type: text/html; charset=ISO-8859-1\r\n
    Content-Encoding: gzip\r\n
    Server: CAFE/1.0\r\n
    Cache-control: private, x-gzip-ok=""\r\n
  > Content-length: 1272\r\n
    Date: Thu, 13 May 2004 10:17:14 GMT\r\n
    \r\n
    [HTTP response 1/1]
    [Time since request: 0.971397000 seconds]
    [Request in frame: 18]
    [Request URI [truncated]: http://pagead2.googlesyndication.com/pagead/ads?client=ca-pub-2309191948673629&random=108444
    Content-encoded entity body (gzip): 1272 bytes -> 3608 bytes
    File Data: 3608 bytes
```

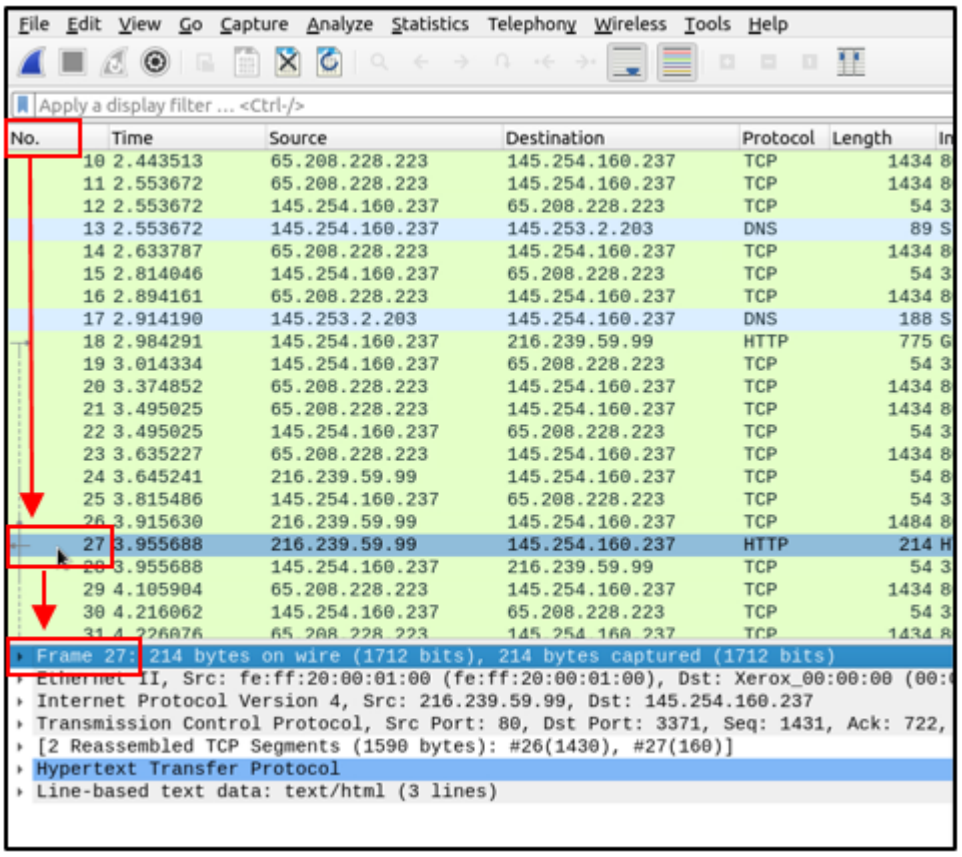
**Application Data:** This extension of the 5th layer can show the application-specific data.

```
Line-based text data: text/html (3 lines)
    <html><head><style><!--\n
     [truncated].ch{cursor:pointer;cursor:hand}a.ad:link { color: #000000 }
     [truncated]function ss(w,id) {window.status = w;return true;}function
```

# Packet Navigation

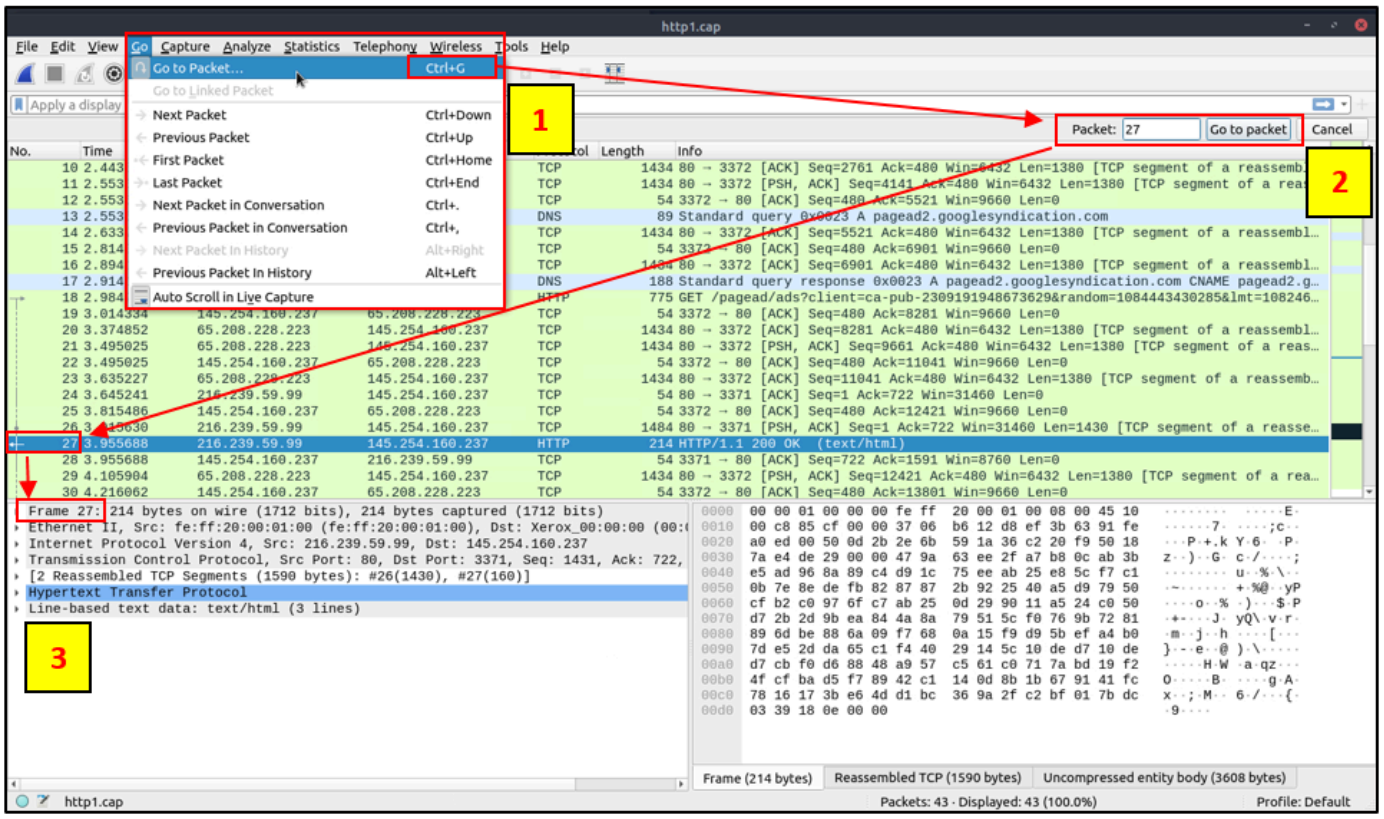
## Packet Numbers

Wireshark calculates the number of investigated packets and assigns a unique number for each packet. This helps the analysis process for big captures and makes it easy to go back to a specific point of an event.




## Go to Packet

Packet numbers do not only help to count the total number of packets or make it easier to find/investigate specific packets. This feature not only navigates between packets up and down; it also provides in-frame packet tracking and finds the next packet in the particular part of the conversation. You can use the **"Go"** menu and toolbar to view specific packets.

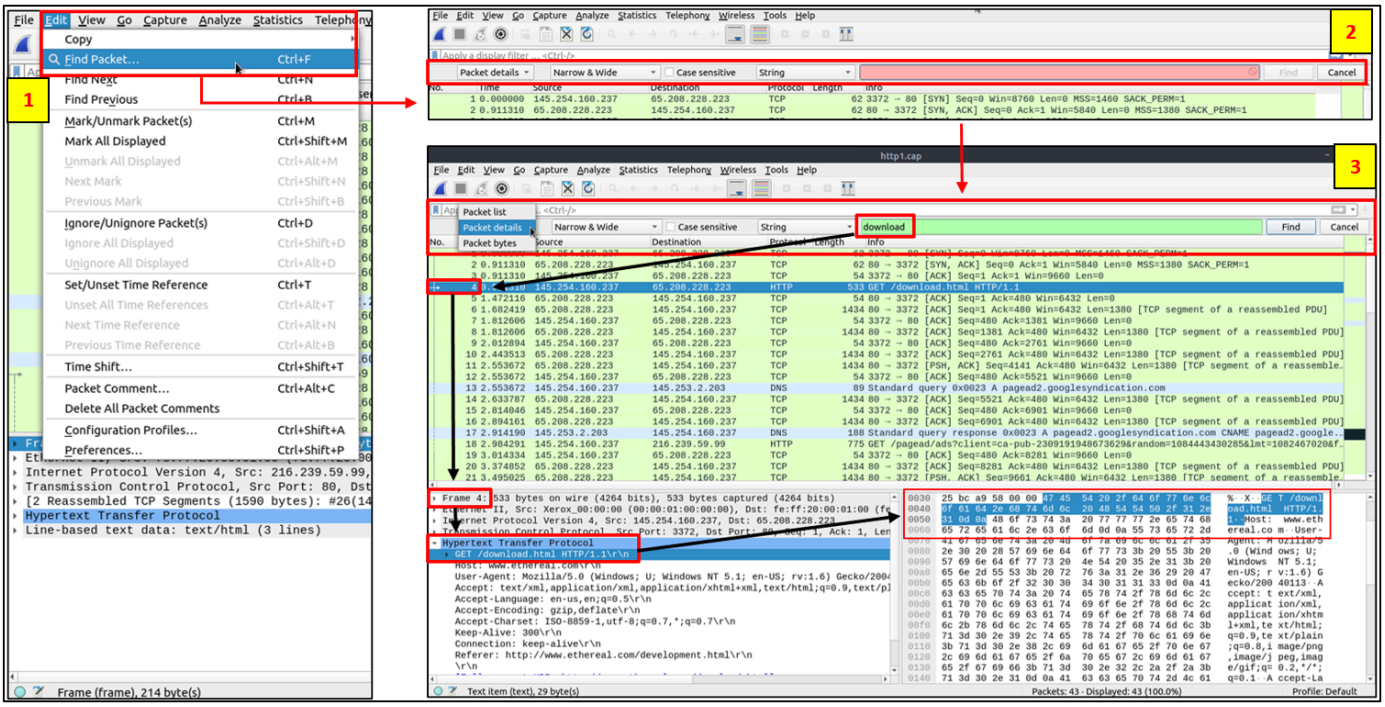



## Find Packets

Apart from packet number, Wireshark can find packets by packet content. You can use the **"Edit --> Find Packet"** menu to make a search inside the packets for a particular event of interest. This helps analysts and administrators to find specific intrusion patterns or failure traces.

There are two crucial points in finding packets. The first is knowing the input type. This functionality accepts four types of inputs (Display filter, Hex, String and Regex). String and regex searches are the most commonly used search types. Searches are case insensitive, but you can set the case sensitivity in your search by clicking the radio button.

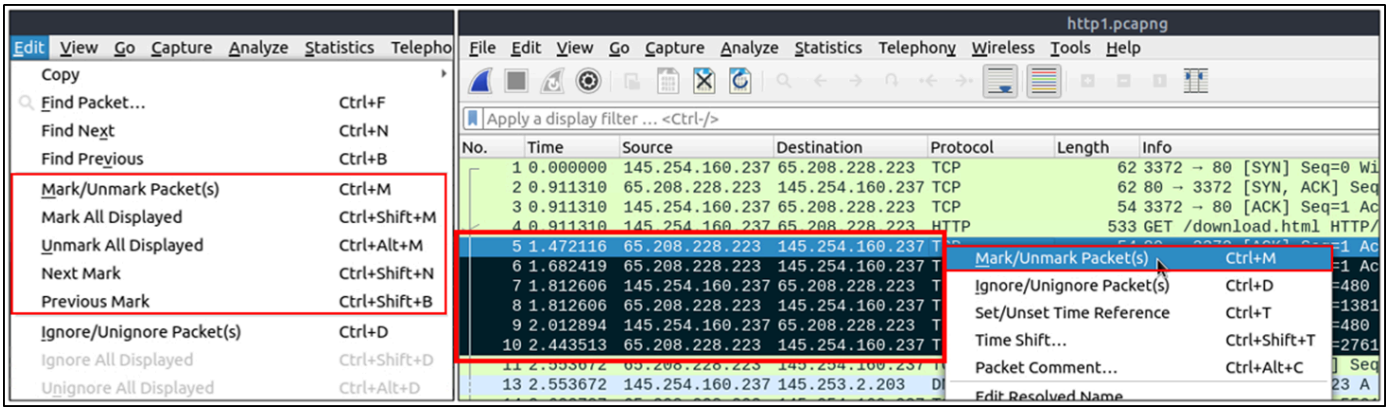
The second point is choosing the search field. You can conduct searches in the three panes (packet list, packet details, and packet bytes), and it is important to know the available information in each pane to find the event of interest. For example, if you try to find the information available in the packet details pane and conduct the search in the packet list pane, Wireshark won't find it even if it exists.




## Mark Packets

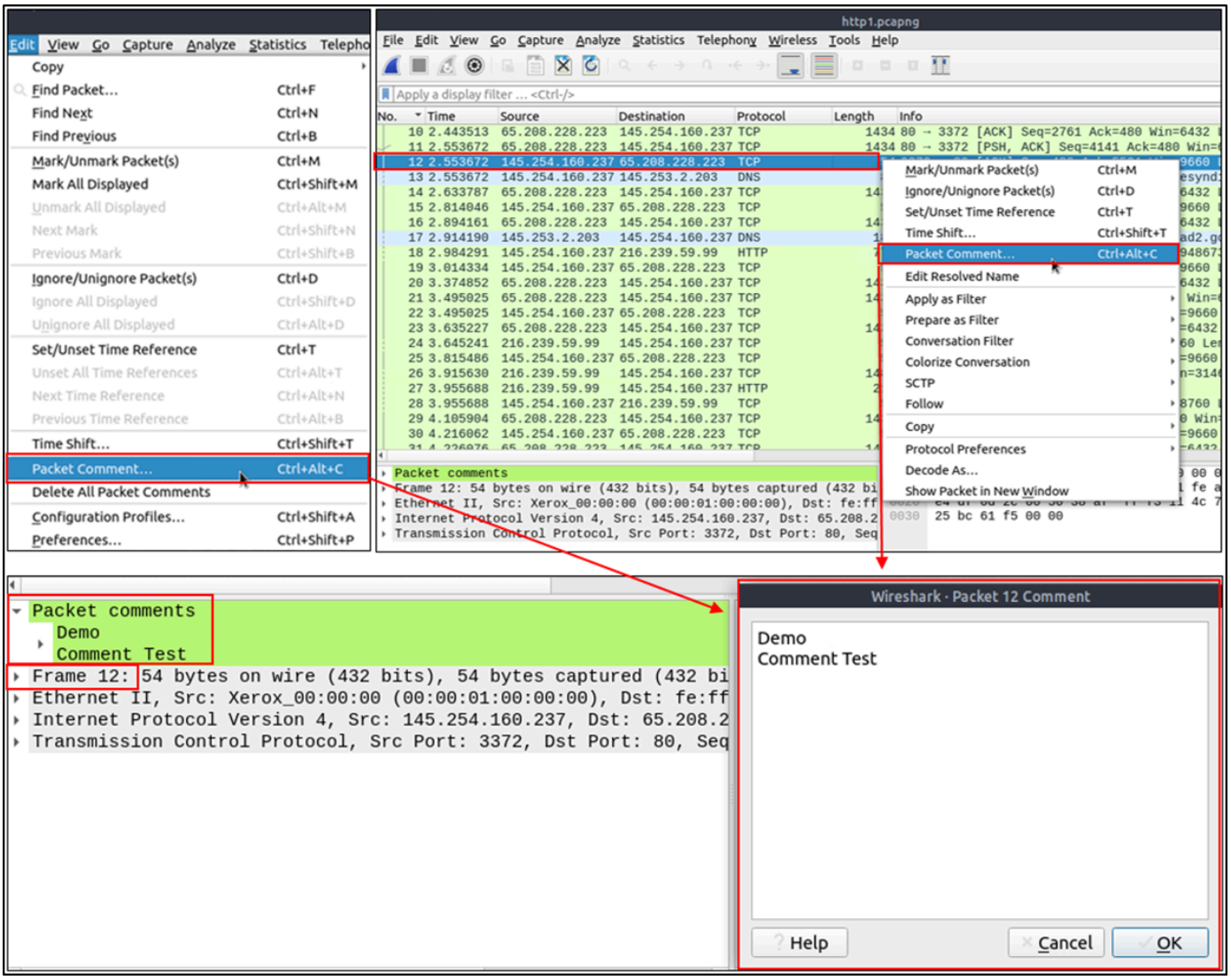
Marking packets is another helpful functionality for analysts. You can find/point to a specific packet for further investigation by marking it. It helps analysts point to an event of interest or export particular packets from the capture. You can use the "Edit" or the "right-click" menu to mark/unmark packets.

Marked packets will be shown in black regardless of the original colour representing the connection type. Note that marked packet information is renewed every file session, so marked packets will be lost after closing the capture file.



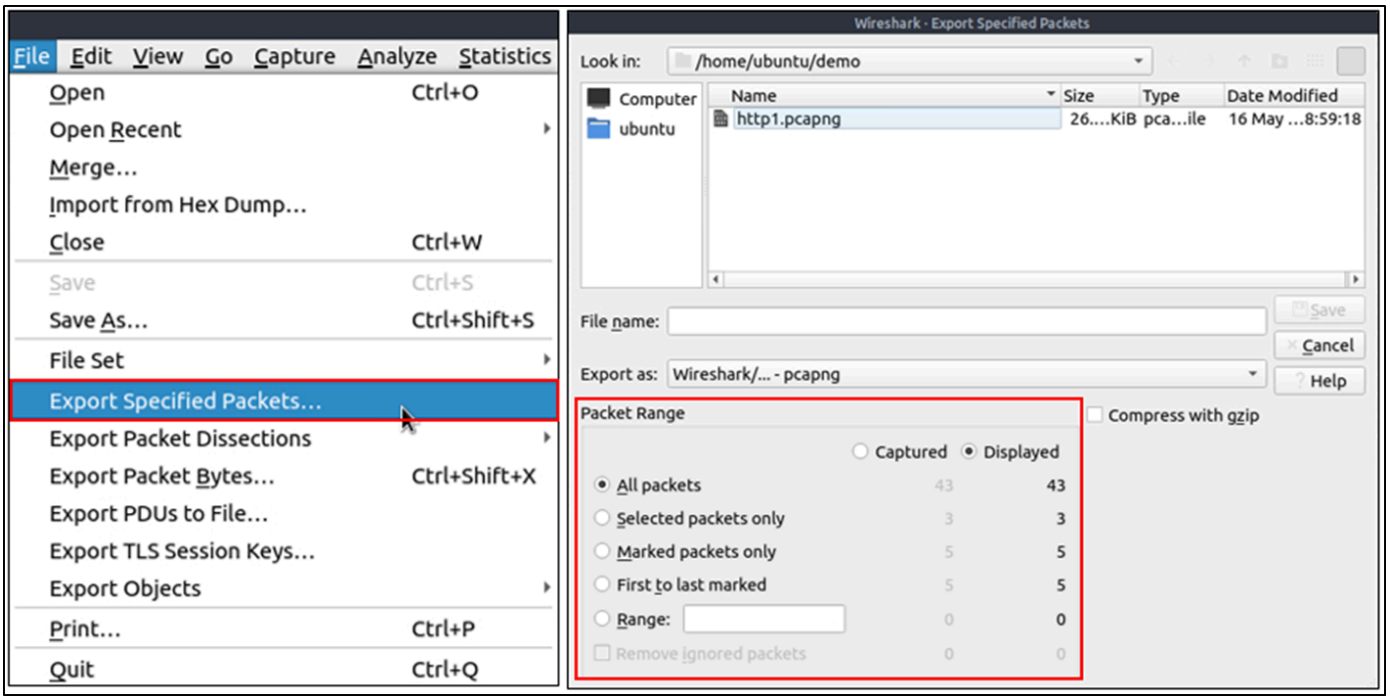

## Packet Comments

Similar to packet marking, commenting is another helpful feature for analysts. You can add comments for particular packets that will help the further investigation or remind and point out important/suspicious points for other layer analysts. Unlike packet marking, the comments can stay within the capture file until the operator removes them.



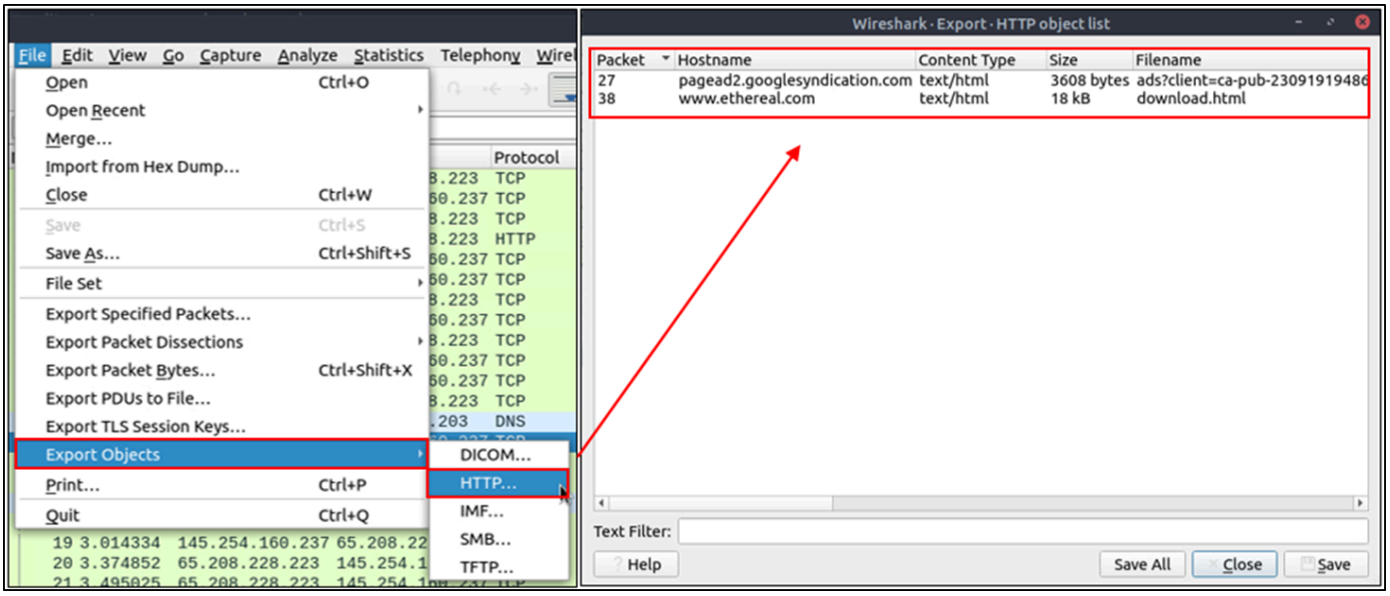

## Export Packets

Capture files can contain thousands of packets in a single file. As mentioned earlier, Wireshark is not an IDS, so sometimes, it is necessary to separate specific packages from the file and dig deeper to resolve an incident. This functionality helps analysts share the only suspicious packages (decided scope). Thus redundant information is not included in the analysis process. You can use the **"File"** menu to export packets.



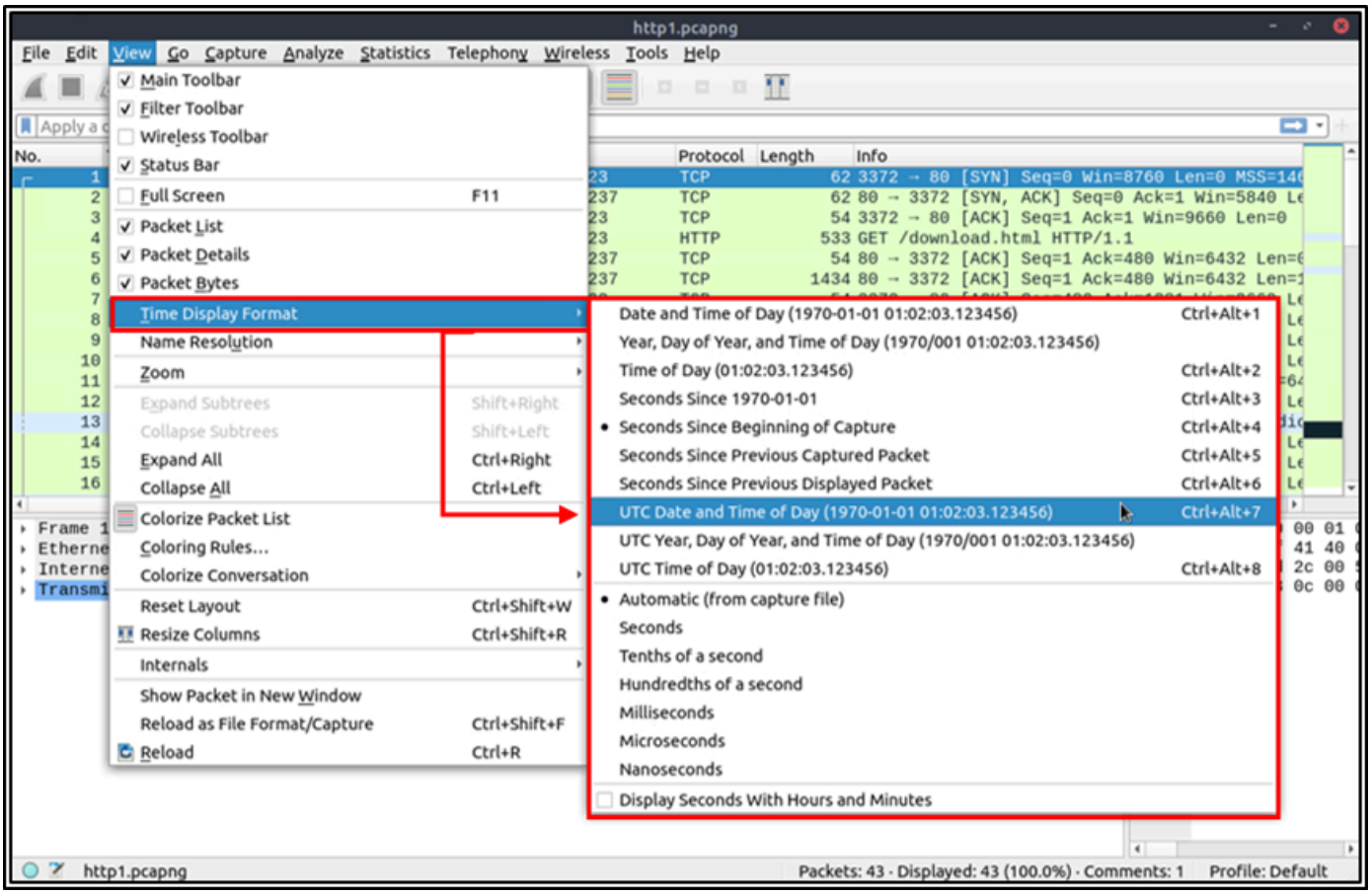

## Export Objects (Files)

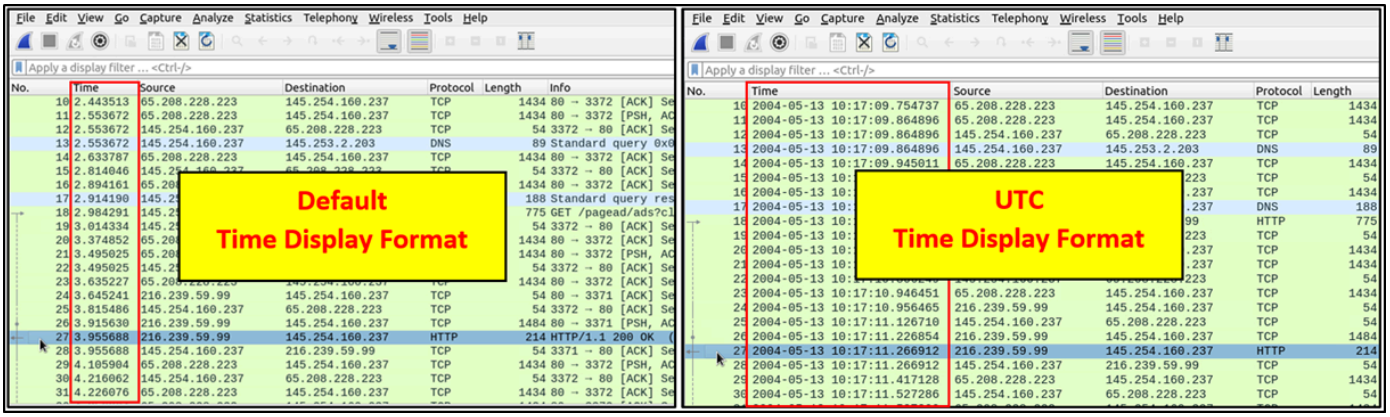
Wireshark can extract files transferred through the wire. For a security analyst, it is vital to discover shared files and save them for further investigation. Exporting objects are available only for selected protocol's streams (DICOM, HTTP, IMF, SMB and TFTP).




## Time Display Format

Wireshark lists the packets as they are captured, so investigating the default flow is not always the best option. By default, Wireshark shows the time in "Seconds Since Beginning of Capture", the common usage is using the UTC Time Display Format for a better view. You can use the "View --> Time Display Format" menu to change the time display format.




## Expert Info

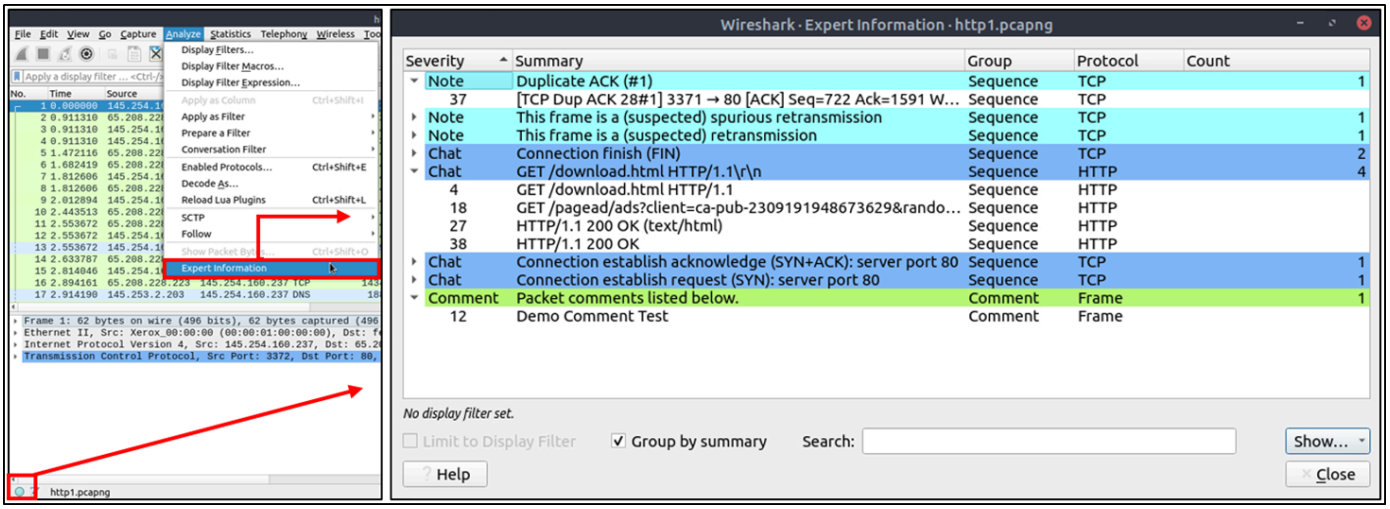
Wireshark also detects specific states of protocols to help analysts easily spot possible anomalies and problems. Note that these are only suggestions, and there is always a chance of having false positives/negatives.

| | | |
|---|---|---|
| **Severity** | **Colour** | **Info** |
| **Chat** | **Blue** | Information on usual workflow. |
| **Note** | **Cyan** | Notable events like application error codes. |
| **Warn** | **Yellow** | Warnings like unusual error codes or problem statements. |
| **Error** | **Red** | Problems like malformed packets. |

Frequently encountered information groups are listed in the table below. You can refer to Wireshark's official documentation for more information on the expert information entries.

| | | | |
|---|---|---|---|
| **Group** | **Info** | **Group** | **Info** |
| **Checksum** | Checksum errors. | **Deprecated** | Deprecated protocol usage. |
| **Comment** | Packet comment detection. | **Malformed** | Malformed packet detection. |

You can use the **"lower left bottom section"** in the status bar or **"Analyse --> Expert Information"** menu to view all available information entries via a dialogue box. It will show the packet number, summary, group protocol and total occurrence.




# Packet Filtering

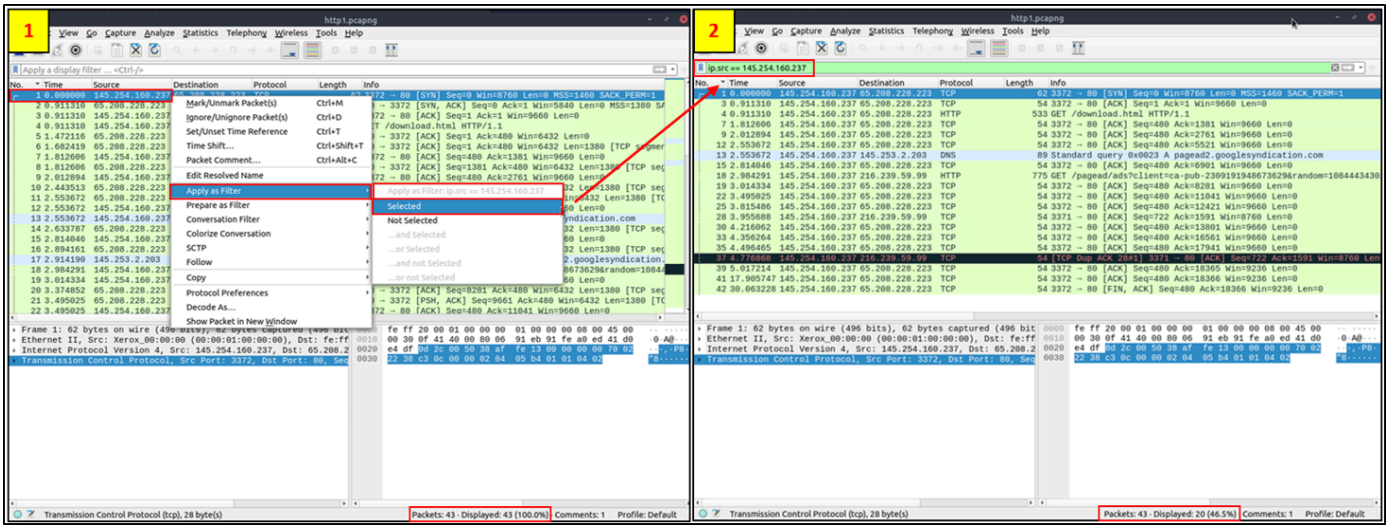
Wireshark has a powerful filter engine that helps analysts to narrow down the traffic and focus on the event of interest. Wireshark has two types of filtering approaches: capture and display filters. Capture filters are used for **"capturing"** only the packets valid for the used filter. Display filters are used for **"viewing"** the packets valid for the used filter.

Filters are specific queries designed for protocols available in Wireshark's official protocol reference. While the filters are only the option to investigate the event of interest, there are two different ways to filter traffic and remove the noise from the capture file. The first one uses queries, and the second uses the right-click menu.

### Apply as Filter

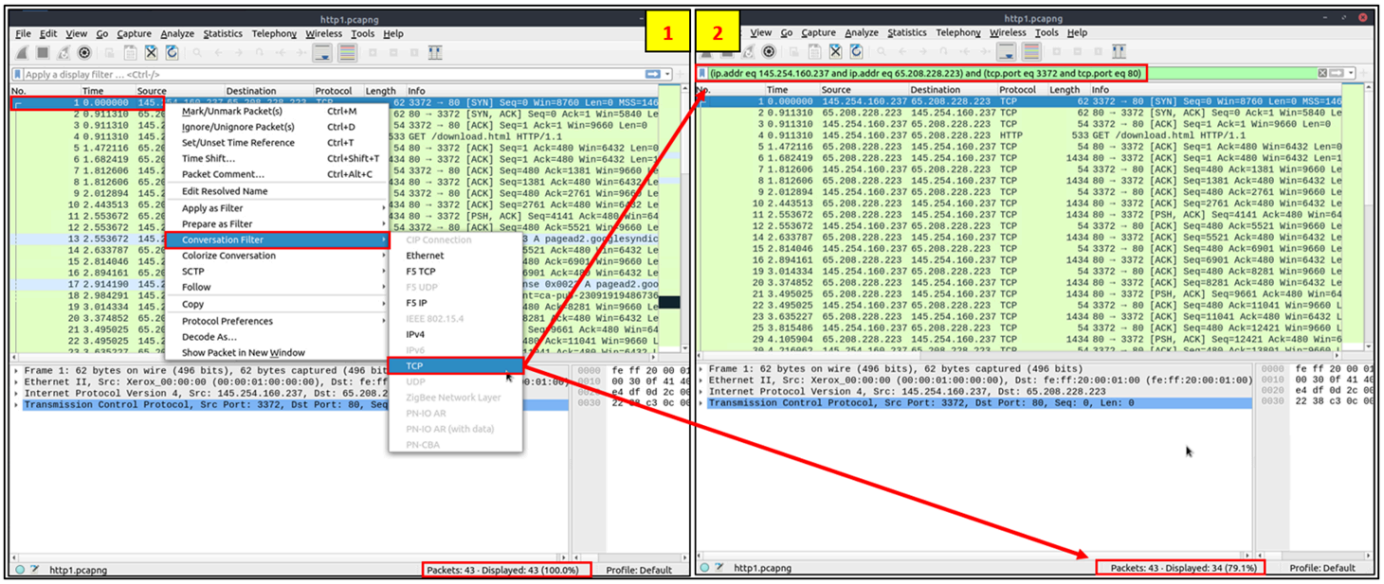
This is the most basic way of filtering traffic. While investigating a capture file, you can click on the field you want to filter and use the "right-click menu" or **"Analyse --> Apply as Filter"** menu to filter the specific value. Once you apply the filter, Wireshark will generate the required filter query, apply it, show

the packets according to your choice, and hide the unselected packets from the packet list pane. Note that the number of total and displayed packets are always shown on the status bar.



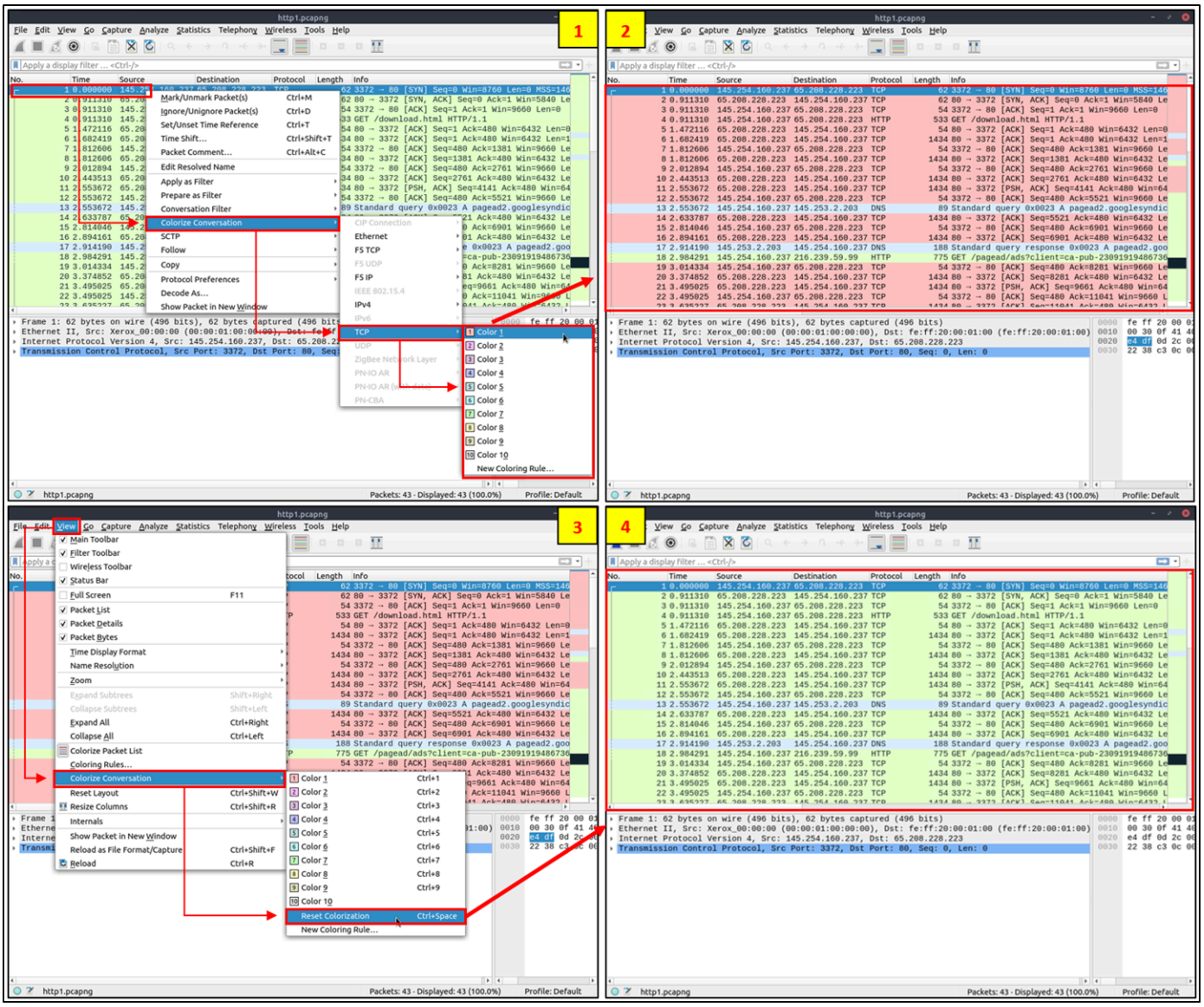

## Conversation Filter

When you use the "Apply as a Filter" option, you will filter only a single entity of the packet. This option is a good way of investigating a particular value in packets. However, suppose you want to investigate a specific packet number and all linked packets by focusing on IP addresses and port numbers. In that case, the "Conversation Filter" option helps you view only the related packets and hide the rest of the packets easily. You can use the"right-click menu" or "**Analyse --> Conversation Filter**" menu to filter conversations.



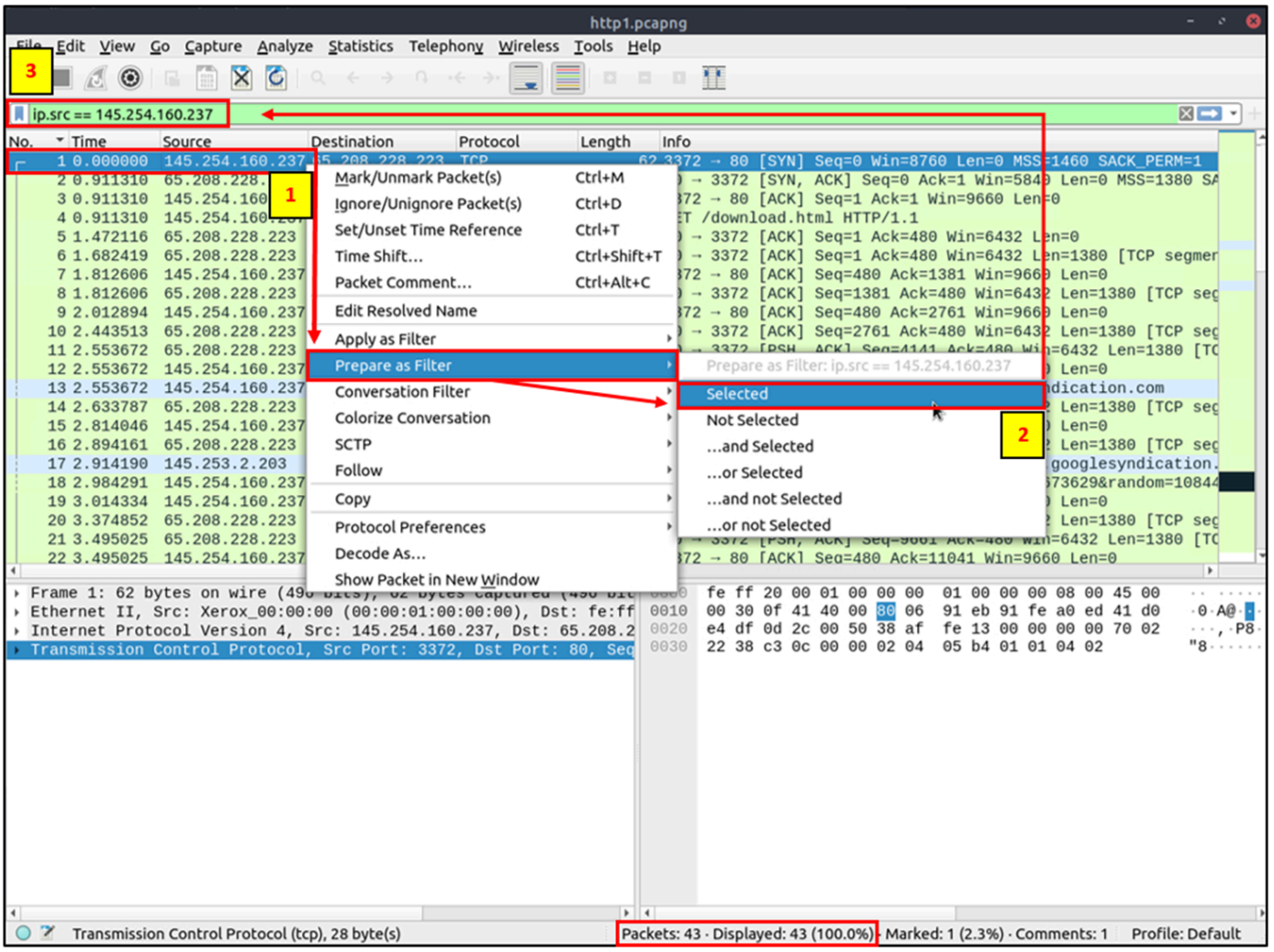

## Colourise Conversation

This option is similar to the "Conversation Filter" with one difference. It highlights the linked packets without applying a display filter and decreasing the number of viewed packets. This option works with the "Colouring Rules" option ad changes the packet colours without considering the previously applied colour rule. You can use the "right-click menu" or **"View --> Colourise Conversation"** menu to colourise a linked packet in a single click. Note that you can use the **"View --> Colourise Conversation --> Reset Colourisation"** menu to undo this operation.



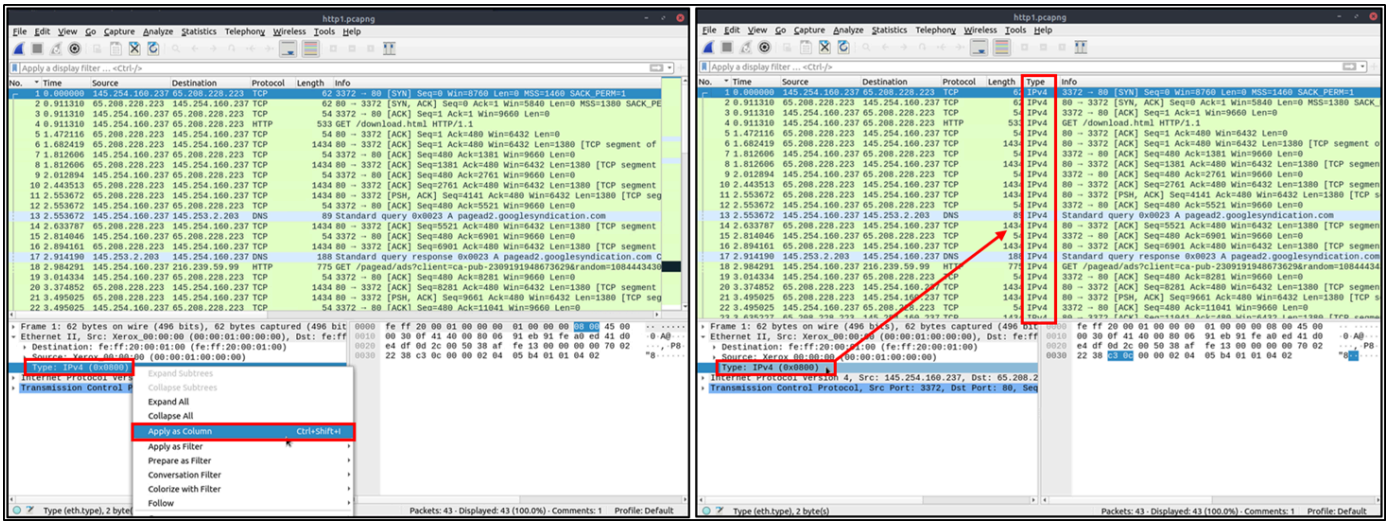

## Prepare as Filter

Similar to "Apply as Filter", this option helps analysts create display filters using the "right-click" menu. However, unlike the previous one, this model doesn't apply the filters after the choice. It adds the required query to the pane and waits for the execution command (enter) or another chosen filtering option by using the **".. and/or.."** from the "right-click menu".




## Apply as Column

By default, the packet list pane provides basic information about each packet. You can use the "right-click menu" or "**Analyse --> Apply as Column**" menu to add columns to the packet list pane. Once you click on a value and apply it as a column, it will be visible on the packet list pane. This function helps analysts examine the appearance of a specific value/field across the available packets in the capture file. You can enable/disable the columns shown in the packet list pane by clicking on the top of the packet list pane.




## Follow Stream

Wireshark displays everything in packet portion size. However, it is possible to reconstruct the streams and view the raw traffic as it is presented at the application level. Following the protocol, streams help analysts recreate the application-level data and understand the event of interest. It is also possible to view the unencrypted protocol data like usernames, passwords and other transferred data.

You can use the"right-click menu" or **"Analyse --> Follow TCP/UDP/HTTP Stream"** menu to follow traffic streams. Streams are shown in a separate dialogue box; packets originating from the server are highlighted with blue, and those originating from the client are highlighted with red.