|  |  |
| --- | --- |
| **Operating Systems & Networking Fundamentals**  Diploma in CSF/IT  Year 1 (2021/22) Semester 2 | Week 13 |
| Practical |
| **Understanding Ethernet Frame Structure using Protocol Analyzer** | |

**OBJECTIVE**

* To learn to use Wireshark protocol analyzer
* To understand Ethernet II frame structure

**INTRODUCTION**

A software protocol analyzer is a software tool that enables a user to capture and analyze the frames that reaches a network interface controller (NIC).

Network administrators use protocol analyzers for network troubleshooting and analysis. Developers use them in developing communication protocols. We will use one, Wireshark, to learn more about protocols. (Wireshark is a free and open source software which you can download and use free of charge.)

Protocol analyzers can filter the frames so that only a subset of the frames is displayed. The criterion of the filter can be defined by the user. The rest of the frames are hidden. (Note: There is also a capture filter which directs the protocol analyzer to capture only certain frames.)

**ACTIVITY**

In this activity we will use the protocol analyzer to capture frames in a LAN with ping traffic.

* 1. Launch the Wireshark program on a PC in the laboratory. A welcome screen lists the available network interfaces on the PC.
  2. Select the “Interface” with network traffic.

[If no network traffic is seen on any of the interfaces, check that the network cable of your laboratory PC is connected to the network i.e. PC’s RJ45 port connected to the wall outlet with a UTP cable]

* 1. Click on ***Capture*** tab and select ***Start*** to start capturing frames on the network. Select ***Stop*** to stop capturing.

The Wireshark window displays the ***Summary***, the ***Detail*** and the ***Hex*** panes or panels as shown in the screenshot below.

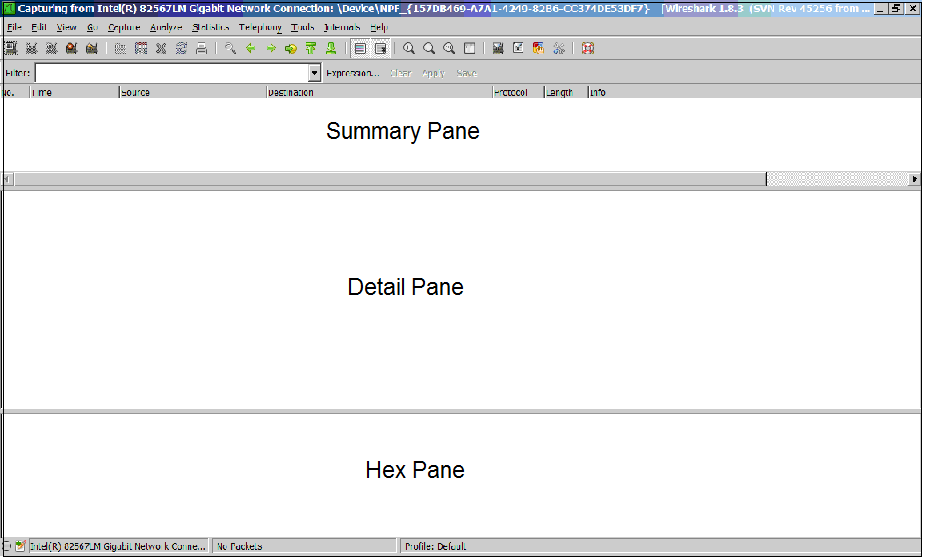


Figure 1: Summary, Detail and Hex Pane

Note: The height of each of the panels can be changed by using the mouse to drag the appropriate boundaries between panels.

* 1. Click on ***Capture*** tab and select ***Start****.*
  2. Generate network traffic by performing a ping to your team member’s laboratory PC.

In the DOS prompt window, type in the Ping command:

ping xxx.xxx.xxx.xxx

where xxx.xxx.xxx.xxx is the IP address of the PC you are trying to communicate with.

If you get a reply stating that:

*Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),*

you can be sure that there is connectivity between the two PCs across the LAN.

Note: If you do not receive the above, turn off the firewall protection, check the IP address and subnet mask settings and try again.

* 1. Click ***Capture*** tab and select ***Stop*** to stop capturing frames. (There is no harm in capturing more frames.)
  2. Save the file with a suitable name that you can remember and retrieve. (You may want to save the file and copy it to your cloud storage e.g. Google Drive or OneDrive.)
  3. Complete the following table by describing the features and information displayed in the Summary, Detail and Hex panels in the appropriate boxes.

|  |  |
| --- | --- |
| **Panel** | **Information Displayed & Feature** |
| **Summary** |  |
| **Detail** |  |
| **Hex** |  |

Table 1: Summary, Detail and Hex Panes

* 1. In the **Summary** panel under the ***Protocol*** and ***Info*** (field) columns, look for and **select** the frame that has the following values: ***Protocol***: ICMP - Echo (ping) request.

(Hint: Click on the header ***Protocol*** all frames containing the same protocol will be together in sequence.

* 1. From the brief (unexpanded) information in the Detail panel of the frame selected in the above step obtain the values for the fields listed in the table below.

|  |  |  |
| --- | --- | --- |
| **Field** | **Value** | **Which Computer’s Address?** |
| Ethernet II Src  (Source MAC Address) | 12 Hex digits | Physical address |
| Ethernet II Dst (Destination MAC\* Address) |  |  |
| Internet ProtocoI, Src (Source IP Address) |  |  |
| Internet ProtocoI, Dst  (Destination IP address) |  |  |

Table 2: MAC and IPv4 Addresses

\*MAC = Media Access Control; MAC address = Hardware / Layer 2 / Data Link address

* 1. Use the command: ***ipconfig* /all** to determine the values of your computer’s configuration that match those in the above table (Step 10).
  2. To identify the header information in an **Ethernet II** frame. Open the Wireshark data file that was saved in Step 7
  3. In the Summary panel, sort the frames according to ***Protocol values***. Select the topmost frame (probably an ***ARP***) and look at the ***Detail*** panel to confirm that the Layer 2 protocol (lowest layer displayed) is ***Ethernet II (MAC)***. If not, choose another frame.

Frame Number of chosen frame: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* 1. Click on the Frame Summary row (topmost row) in the ***Detail*** panel. Notice that all the captured values and their ASCII character equivalents are highlighted as shown in the following screen snapshot.

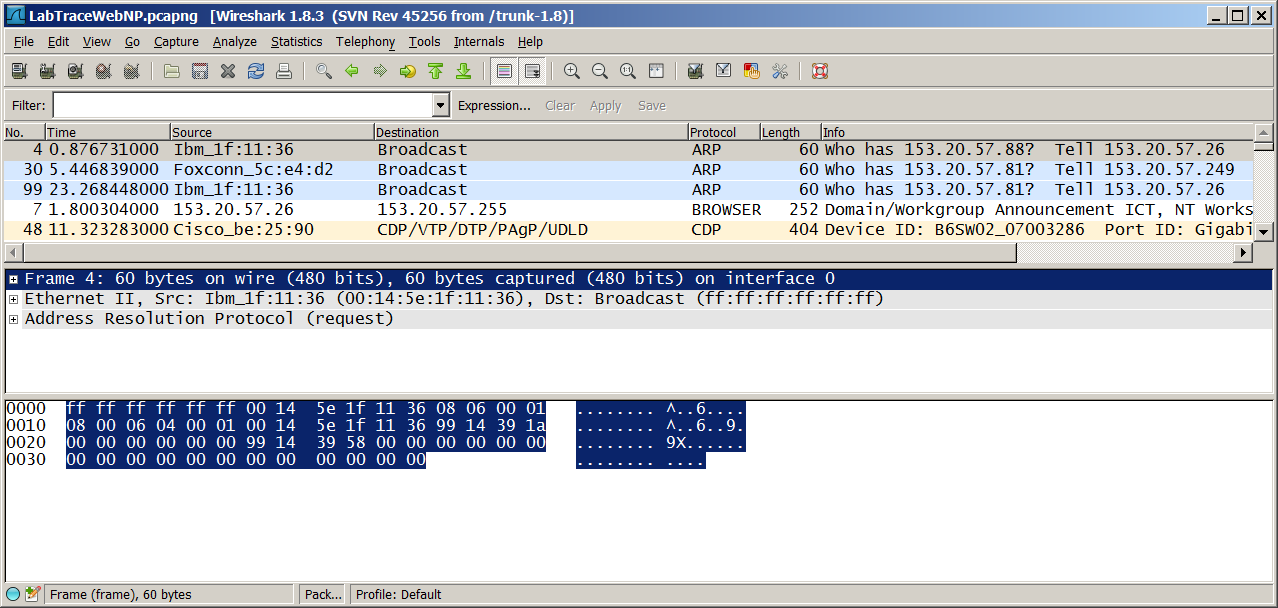


Figure 2: Frame Summary

* 1. In the ***Detail*** panel, click on Ethernet II summary row. Some of the bytes will be highlighted. The front-most portion that is highlighted the Ethernet II header. In the ***Hex*** panel, count the number of bytes of the Ethernet II header.)

The length of the Ethernet II header is \_\_\_\_\_\_\_\_\_\_\_\_\_\_ bytes.

* 1. Expand and display the detailed Ethernet II header information by clicking the plus sign in front of the Ethernet II summary row.
  2. Determine the value of each of the displayed header fields by selecting them in turn and record the results in the Table 3.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name of Ethernet II Header Field\*** | **Start Byte Number (Decimal)** | **End Byte Number (Decimal)** | **Size**  **(Number of Bytes)** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 3 – Ethernet II Header Information (First byte is Byte 0.)

\*NOTES

The Ethernet frame may include a field called Padding. This is the zeroes added to the Ethernet data field to ensure that the **minimum** length of the frame starting from the header to the end of the frame is 64 bytes long. So, the number of bytes in the padding can vary.

The **trailer** portion of an Ethernet frame is a four-byte **frame check sequence** used for error checking (or error control). However, Wireshark does **not** capture these 4 bytes.

**Observations**