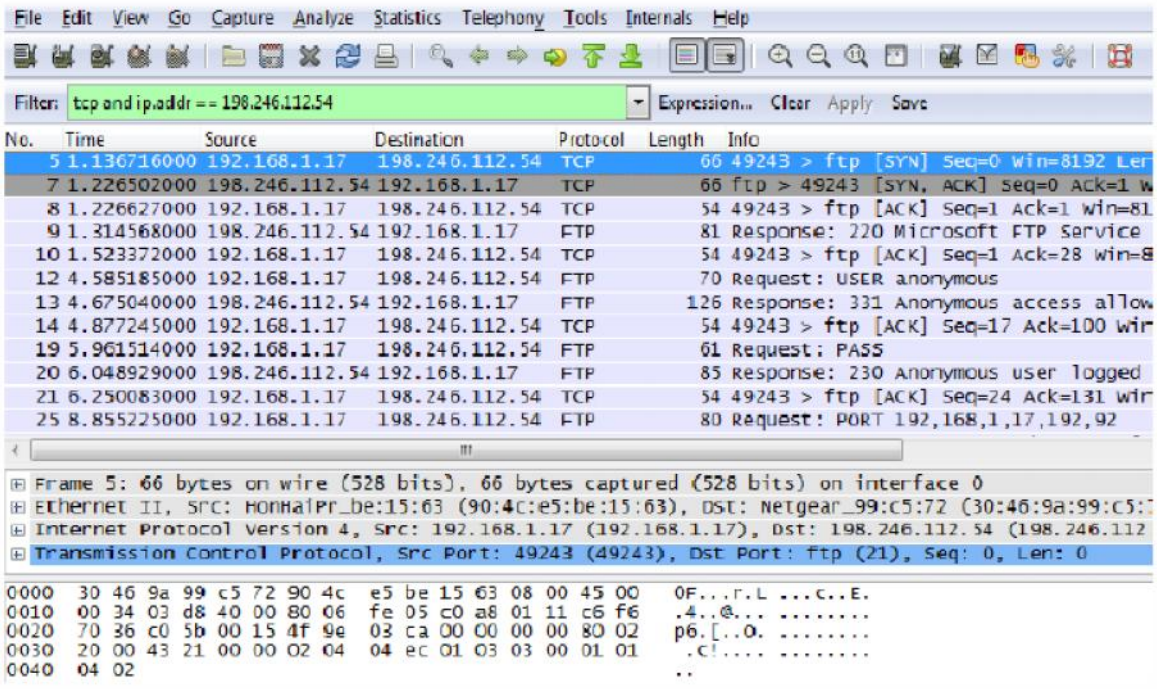
LAB EXERCISE - 5

Tracing FTP with Wireshark

Step s:

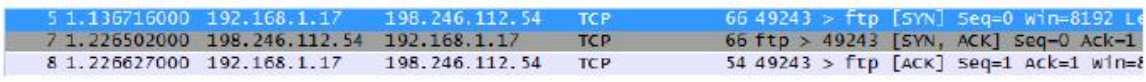
1. Close all unnecessary network traffic, such as the web browser, to limit the amount traffic during the Wireshark capture.
2. Start up your web browser.
3. Start up the Wireshark packet sniffer, (but don't yet begin packet capture).
4. Wait a bit more than one minute, and then begin Wireshark packet capture.
5. Enter the following to your browser ftp.cdc.gov
6. Your browser will display list of files
7. Locate and download the Readme file from that list.
8. Stop the Wireshark capture.
9. View the Wireshark Main Window.

Wireshark captured many packets during the FTP session to ftp.cdc.gov. To limit the amount of data for analysis, type **tcp and ip.addr == 198.246.112.54** in the **Filter: entry** area and click **Apply**. The IP address, 198.246.112.54, is the address for ftp.cdc.gov.



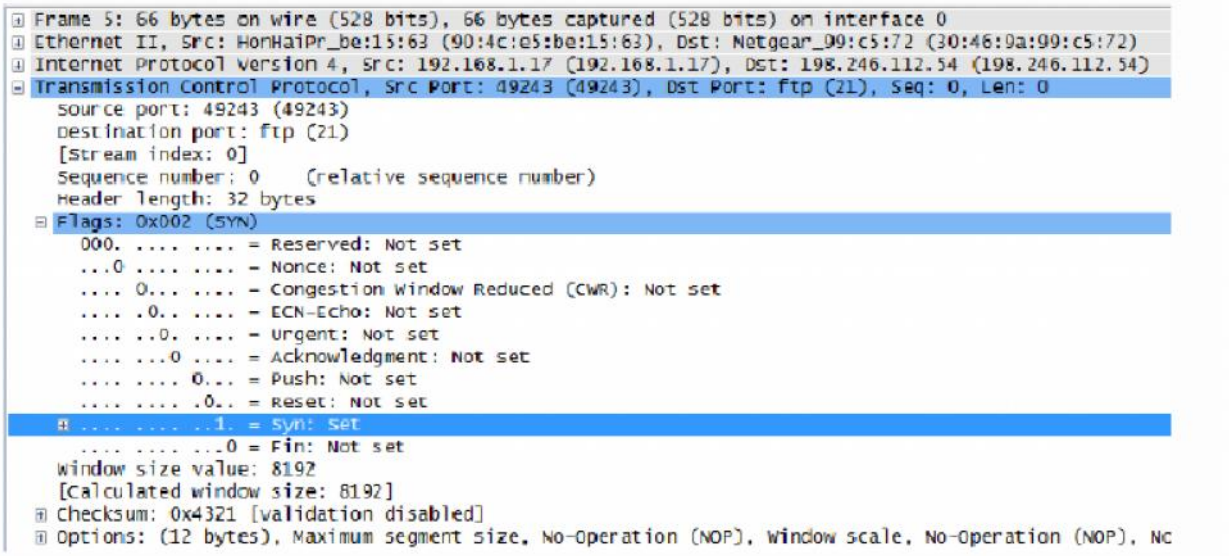
10. Analyze the TCP fields.

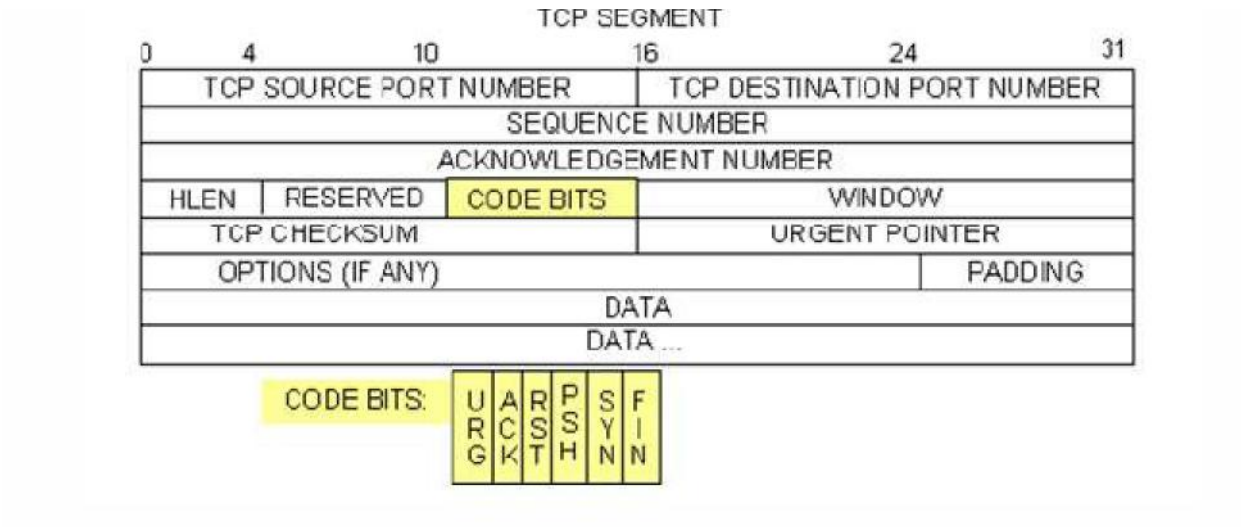
After the TCP filter has been applied, the first three frames in the packet list pane (top section) displays the transport layer protocol TCP creating a reliable session. The sequence of [SYN], [SYN, ACK], and [ACK] illustrates the three-way handshake.



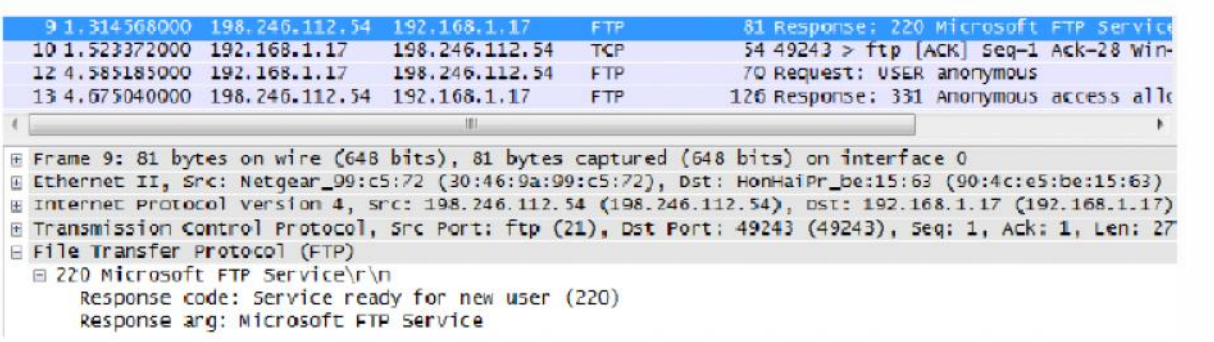
TCP is routinely used during a session to control datagram delivery, verify datagram arrival, and manage window size. For each data exchange between the FTP client and FTP server, a new TCP session is started. At the conclusion of the data transfer, the TCP session is closed. Finally, when the FTP session is finished, TCP performs an orderly shutdown and termination.

In Wireshark, detailed TCP information is available in the packet details pane (middle section). Highlight the first TCP datagram from the host computer, and expand the TCP record. The expanded TCP datagram appears similar to the packet detail pane shown below.



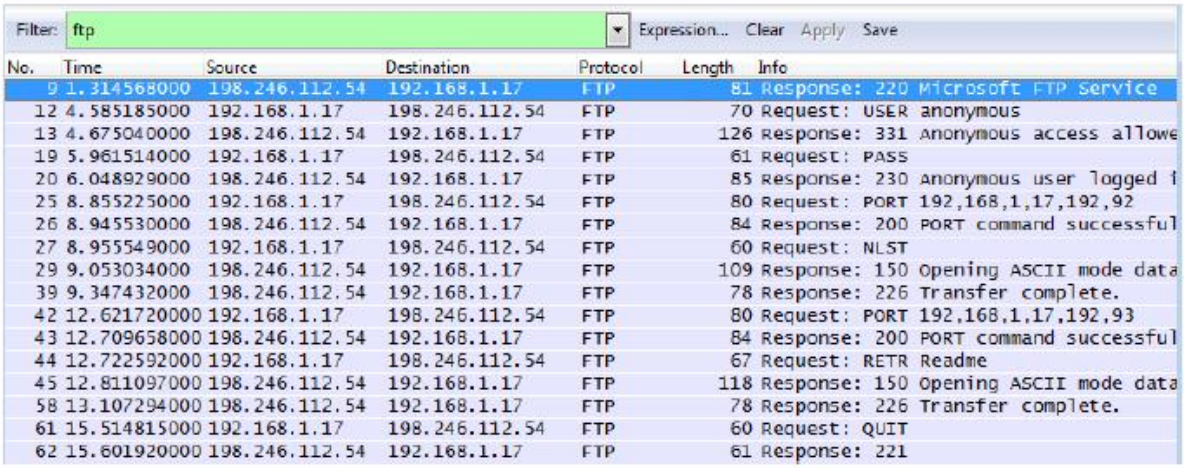


After a TCP session is established, FTP traffic can occur between the PC and FTP server. The FTP client and server communicate between each other, unaware that TCP has control and management over the session. When the FTP server sends a Response: 220 to the FTP client, the TCP session on the FTP client sends an acknowledgment to the TCP session on the server. This sequence is visible in the Wireshark capture below.



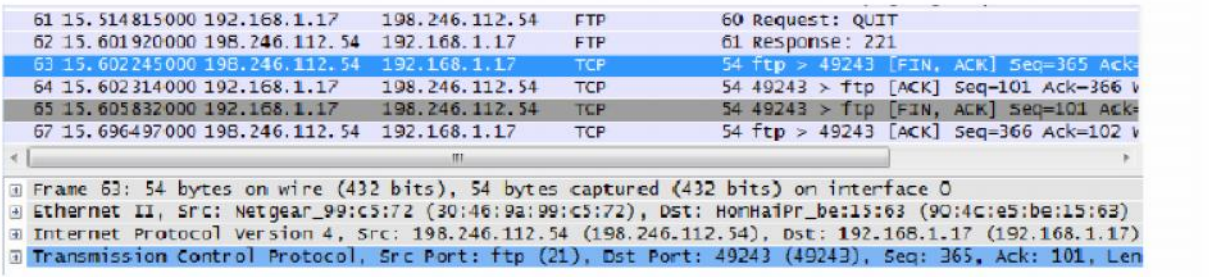
When the FTP session has finished, the FTP client sends a command to —quitll. The FTP server acknowledges the FTP termination with a Response: 221 Goodbye. At this time, the FTP server TCP session sends a TCP datagram to the FTP client, announcing the termination of the TCP session. The FTP client TCP session acknowledges receipt of the termination datagram, then sends its own TCP session termination. When the originator of the TCP termination, FTP server, receives a duplicate termination, an ACK datagram is sent to acknowledge the termination and the TCP session is closed. This sequence is visible in the diagram and capture below.

By applying an **ftp** filter, the entire sequence of the FTP traffic can be examined in Wireshark. Notice the sequence of the events during this FTP session. **The username anonymous was used to retrieve the Readme file**. After the file transfer completed, the user ended the FTP session.



Apply the TCP filter again in Wireshark to examine the termination of the TCP session. Four packets are transmitted for the termination of the TCP session. Because TCP connection is full-duplex, each direction must terminate independently. Examine the source and destination addresses.

Example, If the FTP server has no more data to send in the stream; it sends a segment with the FIN flag set in frame 63. The PC sends an ACK to acknowledge the receipt of the FIN to terminate the session from the server to the client in frame 64. In frame 65, the PC sends a FIN to the FTP server to terminate the TCP session. The FTP server responds with an ACK to acknowledge the FIN from the PC in frame 67. Now the TCP session terminated between the FTP server and PC.


61 15.514815000 192.168.1.17 198.246.112.54 FTP 60 Request: QUIT
62 15.601920000 198.246.112.54 192.168.1.17 FTP 61 Response: 221
63 15.602245000 198.246.112.54 192.168.1.17 TCP 54 ftp > 49243 [FIN, ACK] Seq=365 Ack=
64 15.602314000 192.168.1.17 198.246.112.54 TCP 54 49243 > ftp [ACK] Seq=101 Ack=366 W
65 15.605832000 192.168.1.17 198.246.112.54 TCP 54 49243 > ftp [FIN, ACK] Seq=101 Ack=
67 15.696497000 198.246.112.54 192.168.1.17 TCP 54 ftp > 49243 [ACK] Seq=366 Ack=102 W
Frame 63: 54 bytes on wire (432 bits), 54 bytes captured (432 bits) on interface 0
Ethernet II, Src: Netgear_99:c5:72 (30:46:9a:99:c5:72), Dst: HonHaiPr_be:15:63 (90:4c:e5:be:15:63)
Internet Protocol Version 4, Src: 198.246.112.54 (198.246.112.54), Dst: 192.168.1.17 (192.168.1.17)
Transmission Control Protocol, Src Port: ftp (21), Dst Port: 49243 (49243), Seq: 365, Ack: 101, Len