Experiment-3

3.1 Aim: Study of packet sniffer tools wireshark:- a. Observer performance inpromiscuous as well as non-promiscuous mode. b. Show the packets can be traced based on different filters

3.2 Course Outcome:

Study and describe the system security, malicious softwares and the Network layer security, Transport layer security and application layer security.

3.3 Lab Objective:

- Observe Wireshark's behavior in promiscuous and non-promiscuous modes.
- Apply filters to capture and analyze network traffic.
- Analyze captured packets to identify specific network events.

3.4 Requirement:

Wireshark: Packet analyzer
Operating system: Windows, Linux.

3.5 Theory: **Wireshark** is one of the most widely used packet analysis tools for network troubleshooting, security analysis, and protocol development. It allows users to capture and inspect data packets traveling through a network interface. In this study, we will focus on two primary aspects of Wireshark usage:

a. Observer Performance in Promiscuous vs Non-Promiscuous Mode

Wireshark can operate in two different modes depending on how network traffic is captured:

1. **Promiscuous Mode:**
    - **Definition:** When the network interface card (NIC) operates in promiscuous mode, it receives all packets from the network, not just those addressed to the machine. In this mode, Wireshark captures all traffic that passes through the network segment, regardless of destination.
    - **Use Case:** Promiscuous mode is commonly used in network diagnostics, monitoring, and security testing. This allows the user to see all network activity on a given network, even traffic not intended for the host.
    - **Performance Impact:** While promiscuous mode provides complete visibility of network traffic, it can cause a large amount of data to be captured. This can put more load on the system processing the packets and could potentially slow down the network interface if there is high traffic volume. Wireshark will also show

packets that are not typically visible in non-promiscuous mode, providing a fuller network analysis.

2. **Non-Promiscuous Mode:**
    - **Definition:** In non-promiscuous mode, the NIC only captures packets that are specifically addressed to the machine running Wireshark. Any broadcast or multicast packets that the host is part of (or part of a network it is connected to) will also be captured, but any other traffic not intended for the device is ignored.
    - **Use Case:** Non-promiscuous mode is typically used when only the network traffic specifically directed to or originating from the host is of interest. This reduces the amount of captured traffic and thus lowers the load on the system running Wireshark.
    - **Performance Impact:** The performance of Wireshark is better in non-promiscuous mode because it is capturing fewer packets, focusing only on the relevant traffic. As a result, the amount of data being processed is reduced, which can be useful when capturing specific traffic related to the host system.

Comparing Performance:

Mode	**Visibility**	**Performance Load**	**Use Case**
Promiscuous Mode	Captures all packets	Higher Load	Network troubleshooting, full traffic analysis
Non-Promiscuous Mode	Captures only relevant packets	Lower Load	Host-specific traffic analysis

3.6 Procedure : Tracing Packets Based on Different Filters

Wireshark provides powerful filtering options that allow you to trace specific packets based on criteria such as protocol, IP address, port number, etc. Filters help isolate the traffic of interest, making it easier to analyze network behavior or troubleshoot specific issues.

Types of Filters in Wireshark:

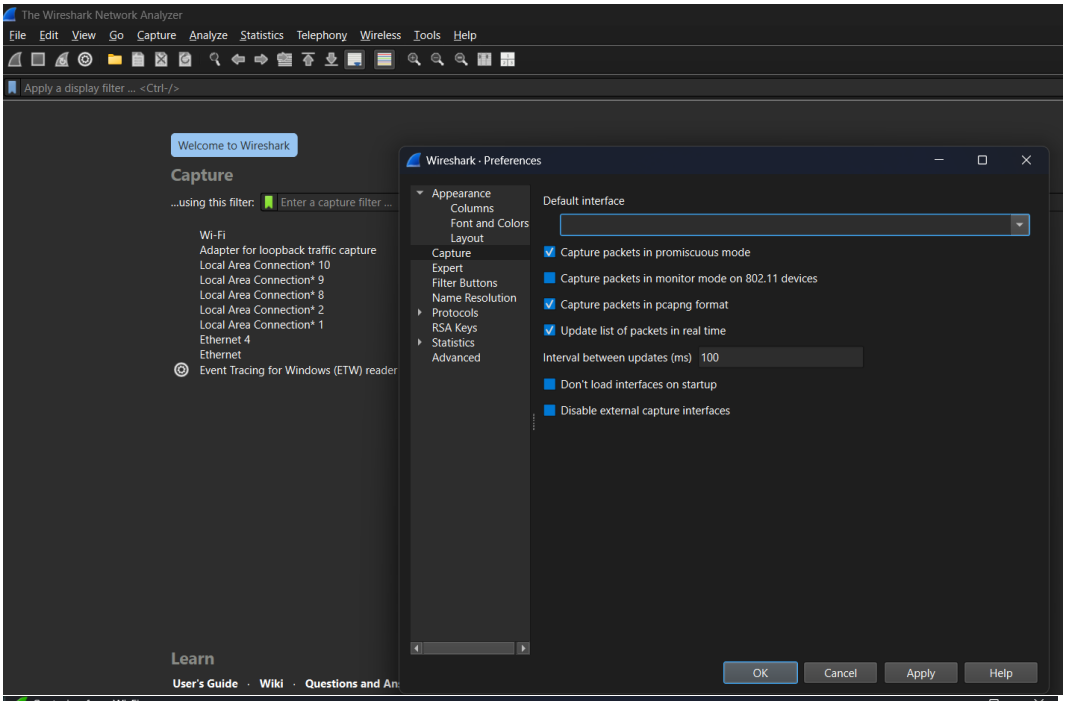
1. **Display Filters:**
    - **Purpose:** Display filters are used to refine which packets are shown in the capture window.
    - **Common Display Filters:**
        - **IP address filter:** To capture packets from or to a specific IP address.
            - Example: ip.addr == 192.168.1.1
        - **Protocol filter:** To capture packets of a specific protocol (e.g., TCP, UDP, HTTP).
            - Example: tcp
        - **Port filter:** To filter packets by port number.
            - Example: tcp.port == 80
        - **HTTP Filter:** To display only HTTP traffic.
            - Example: http
        - **Packet length filter:** To capture packets of a certain size.
            - Example: frame.len > 1500
2. **Capture Filters:**
    - **Purpose:** Capture filters are set before capturing data to limit the packets that Wireshark will capture from the network interface.
    - **Common Capture Filters:**
        - **Capture based on IP address:**
            - Example: host 192.168.1.1
        - **Capture based on protocol:**
            - Example: tcp
        - **Capture based on port:**
            - Example: port 80
        - **Capture based on network subnet:**
            - Example: net 192.168.1.0/24

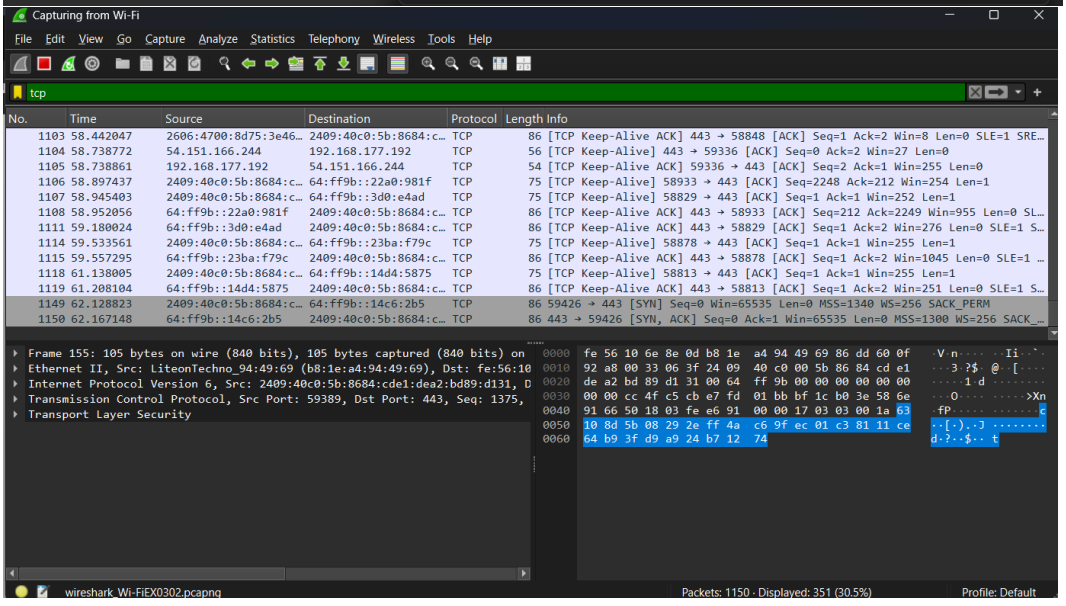
Example of Filtering Traffic in Wireshark:

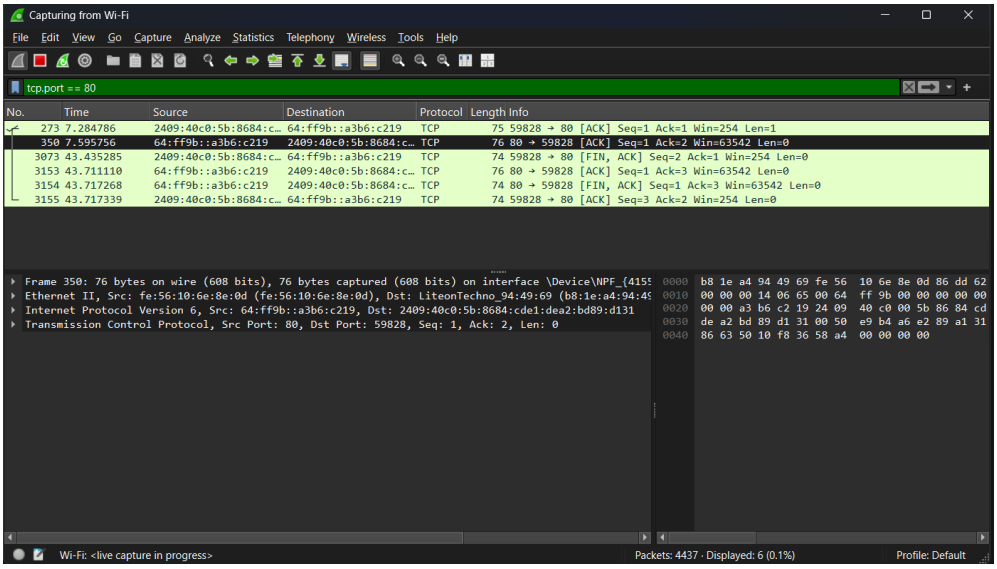
1. **Filter All HTTP Traffic:**
    - Apply the display filter http to view only HTTP traffic. This can be useful when you're investigating web-related issues.
2. **Capture Only Traffic to/from Specific Host:**
    - Use the capture filter host 192.168.1.1 to capture only the packets to and from the host with IP address 192.168.1.1.
3. **Filter by Port (e.g., Port 80):**
    - For HTTP traffic, use the filter tcp.port == 80 to isolate all traffic communicating over port 80.

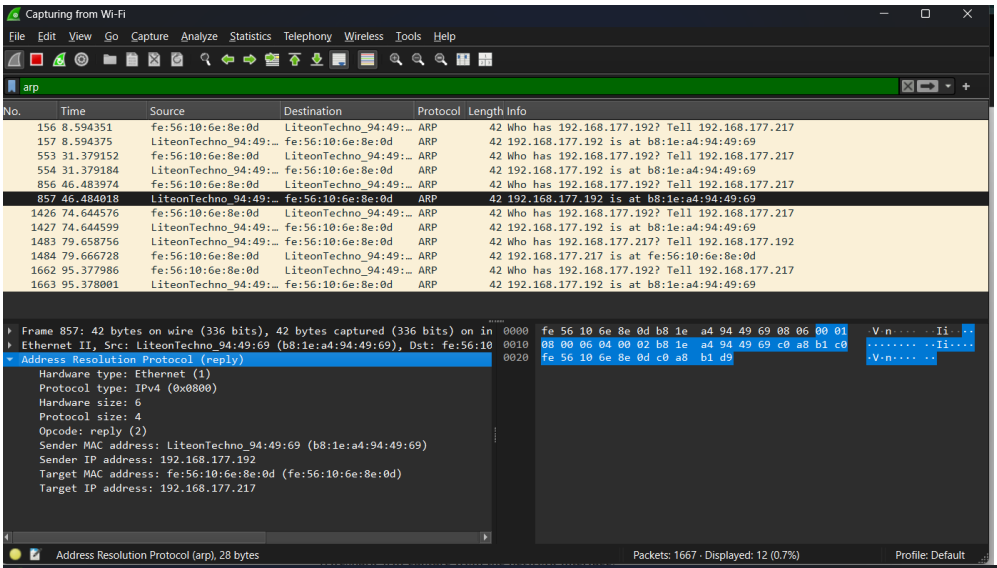
4. **Filter by Protocol (e.g., TCP):**
    - Use the filter tcp to view all TCP traffic. This is helpful when analyzing connection issues or packet behavior on TCP connections.

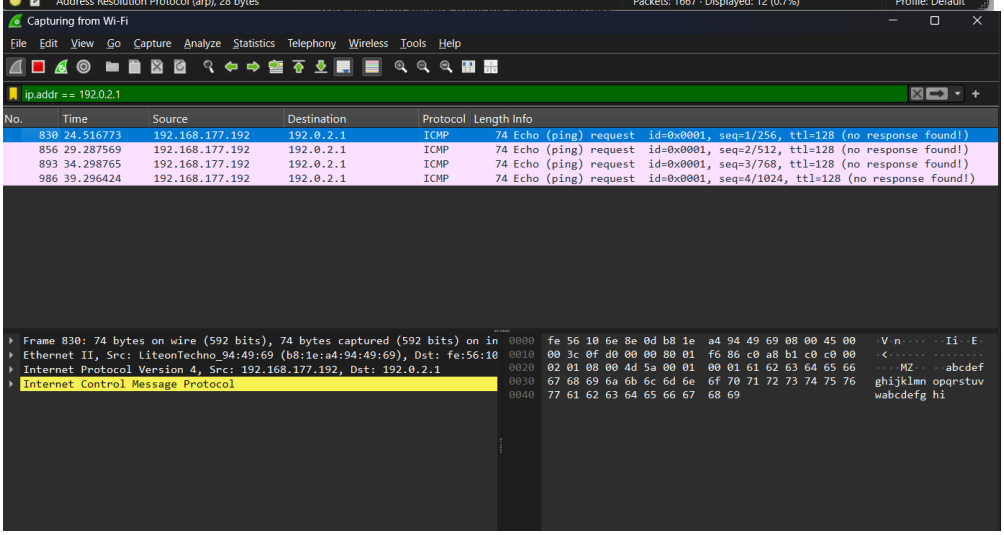
3.7 Result :

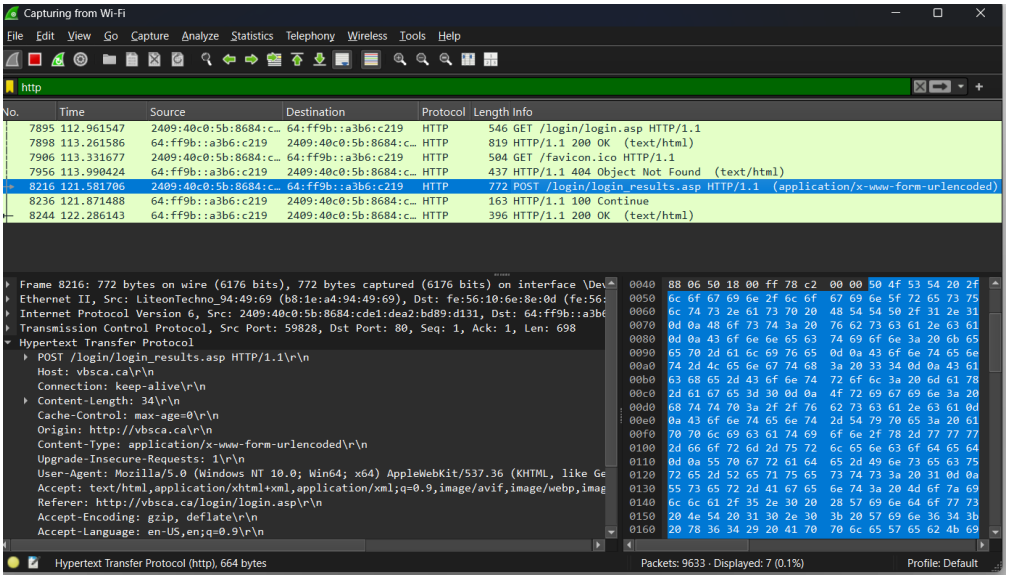


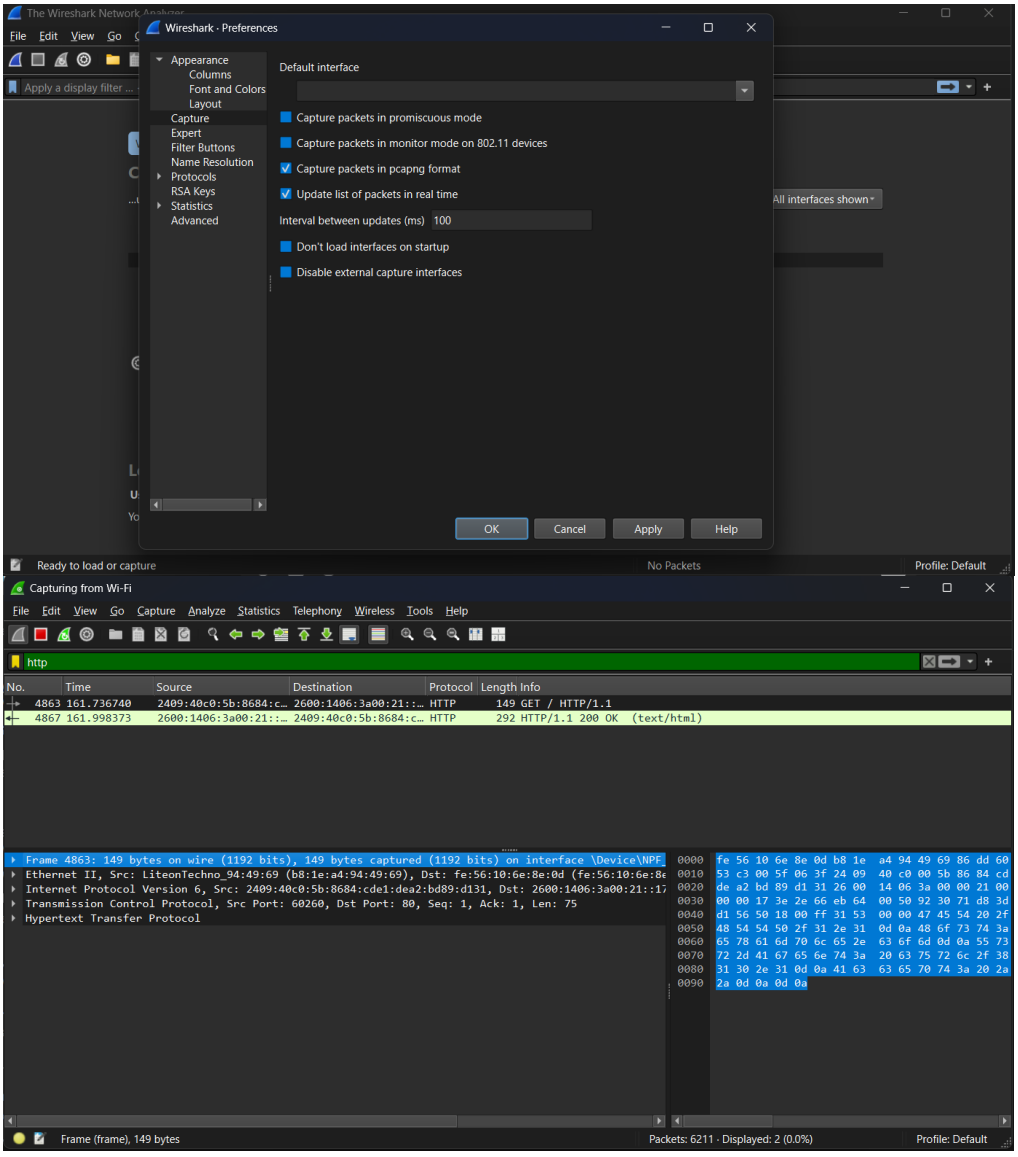


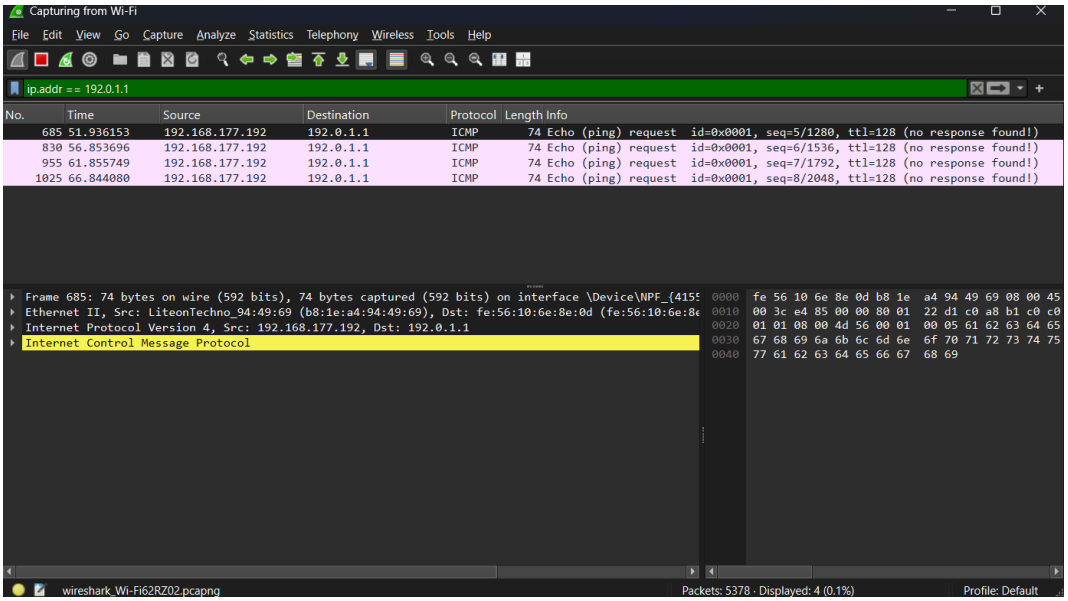

Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp.port == 80
No. Time Source Destination Protocol Length Info
273 7.284786 2409:40c0:5b:8684:c… 64:ff9b::a3b6:c219 TCP 75 59828 → 80 [ACK] Seq=1 Ack=1 Win=254 Len=1
350 7.595756 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… TCP 76 80 → 59828 [ACK] Seq=1 Ack=2 Win=63542 Len=0
3073 43.435285 2409:40c0:5b:8684:c… 64:ff9b::a3b6:c219 TCP 74 59828 → 80 [FIN, ACK] Seq=2 Ack=1 Win=254 Len=0
3153 43.711110 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… TCP 76 80 → 59828 [ACK] Seq=1 Ack=3 Win=63542 Len=0
3154 43.717268 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… TCP 74 80 → 59828 [FIN, ACK] Seq=1 Ack=3 Win=63542 Len=0
3155 43.717339 2409:40c0:5b:8684:c… 64:ff9b::a3b6:c219 TCP 74 59828 → 80 [ACK] Seq=3 Ack=2 Win=254 Len=0
Frame 350: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface \Device\NPF_{4155
Ethernet II, Src: fe:56:10:6e:8e:0d (fe:56:10:6e:8e:0d), Dst: LiteonTechno_94:49:69 (b8:1e:a4:94:49
Internet Protocol Version 6, Src: 64:ff9b::a3b6:c219, Dst: 2409:40c0:5b:8684:cde1:dea2:bd89:d131
Transmission Control Protocol, Src Port: 80, Dst Port: 59828, Seq: 1, Ack: 2, Len: 0
0000 b8 1e a4 94 49 69 fe 56 10 6e 8e 0d 86 dd 62
0010 00 00 00 14 06 65 00 64 ff 9b 00 00 00 00 00
0020 00 00 a3 b6 c2 19 24 09 40 c0 00 5b 86 84 cd
0030 de a2 bd 89 d1 31 00 50 e9 b4 a6 e2 89 a1 31
0040 86 63 50 10 f8 36 58 a4 00 00 00 00
Wi-Fi: <live capture in progress>
Packets: 4437 · Displayed: 6 (0.1%)
Profile: Default


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
arp
No. Time Source Destination Protocol Length Info
156 8.594351 fe:56:10:6e:8e:0d LiteonTechno_94:49:… ARP 42 Who has 192.168.177.192? Tell 192.168.177.217
157 8.594375 LiteonTechno_94:49:… fe:56:10:6e:8e:0d ARP 42 192.168.177.192 is at b8:1e:a4:94:49:69
553 31.379152 fe:56:10:6e:8e:0d LiteonTechno_94:49:… ARP 42 Who has 192.168.177.192? Tell 192.168.177.217
554 31.379184 LiteonTechno_94:49:… fe:56:10:6e:8e:0d ARP 42 192.168.177.192 is at b8:1e:a4:94:49:69
856 46.483974 fe:56:10:6e:8e:0d LiteonTechno_94:49:… ARP 42 Who has 192.168.177.192? Tell 192.168.177.217
857 46.484018 LiteonTechno_94:49:… fe:56:10:6e:8e:0d ARP 42 192.168.177.192 is at b8:1e:a4:94:49:69
1426 74.644576 fe:56:10:6e:8e:0d LiteonTechno_94:49:… ARP 42 Who has 192.168.177.192? Tell 192.168.177.217
1427 74.644599 LiteonTechno_94:49:… fe:56:10:6e:8e:0d ARP 42 192.168.177.192 is at b8:1e:a4:94:49:69
1483 79.658756 LiteonTechno_94:49:… fe:56:10:6e:8e:0d ARP 42 Who has 192.168.177.217? Tell 192.168.177.192
1484 79.666728 fe:56:10:6e:8e:0d LiteonTechno_94:49:… ARP 42 192.168.177.217 is at fe:56:10:6e:8e:0d
1662 95.377986 fe:56:10:6e:8e:0d LiteonTechno_94:49:… ARP 42 Who has 192.168.177.192? Tell 192.168.177.217
1663 95.378001 LiteonTechno_94:49:… fe:56:10:6e:8e:0d ARP 42 192.168.177.192 is at b8:1e:a4:94:49:69
Frame 857: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on in
Ethernet II, Src: LiteonTechno_94:49:69 (b8:1e:a4:94:49:69), Dst: fe:56:10
Address Resolution Protocol (reply)
Hardware type: Ethernet (1)
Protocol type: IPv4 (0x0800)
Hardware size: 6
Protocol size: 4
Opcode: reply (2)
Sender MAC address: LiteonTechno_94:49:69 (b8:1e:a4:94:49:69)
Sender IP address: 192.168.177.192
Target MAC address: fe:56:10:6e:8e:0d (fe:56:10:6e:8e:0d)
Target IP address: 192.168.177.217
0000 fe 56 10 6e 8e 0d b8 1e a4 94 49 69 08 06 00 01
0010 08 00 06 04 00 02 b8 1e a4 94 49 69 c0 a8 b1 c0
0020 fe 56 10 6e 8e 0d c0 a8 b1 d9
Address Resolution Protocol (arp), 28 bytes
Packets: 1667 · Displayed: 12 (0.7%)
Profile: Default


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
ip.addr == 192.0.2.1
No. Time Source Destination Protocol Length Info
830 24.516773 192.168.177.192 192.0.2.1 ICMP 74 Echo (ping) request id=0x0001, seq=1/256, ttl=128 (no response found!)
856 29.287569 192.168.177.192 192.0.2.1 ICMP 74 Echo (ping) request id=0x0001, seq=2/512, ttl=128 (no response found!)
893 34.298765 192.168.177.192 192.0.2.1 ICMP 74 Echo (ping) request id=0x0001, seq=3/768, ttl=128 (no response found!)
986 39.296424 192.168.177.192 192.0.2.1 ICMP 74 Echo (ping) request id=0x0001, seq=4/1024, ttl=128 (no response found!)
Frame 830: 74 bytes on wire (592 bits), 74 bytes captured (592 bits) on in
Ethernet II, Src: LiteonTechno_94:49:69 (b8:1e:a4:94:49:69), Dst: fe:56:10
Internet Protocol Version 4, Src: 192.168.177.192, Dst: 192.0.2.1
Internet Control Message Protocol
0000 fe 56 10 6e 8e 0d b8 1e a4 94 49 69 08 00 45 00
0010 00 3c 0f d0 00 00 80 01 f6 86 c0 a8 b1 c0 c0 00
0020 02 01 08 00 4d 5a 00 01 00 01 61 62 63 64 65 66
0030 67 68 69 6a 6b 6c 6d 6e 6f 70 71 72 73 74 75 76
0040 77 61 62 63 64 65 66 67 68 69


Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
No. Time Source Destination Protocol Length Info
7895 112.961547 2409:40c0:5b:8684:c… 64:ff9b::a3b6:c219 HTTP 546 GET /login/login.asp HTTP/1.1
7898 113.261586 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… HTTP 819 HTTP/1.1 200 OK (text/html)
7906 113.331677 2409:40c0:5b:8684:c… 64:ff9b::a3b6:c219 HTTP 504 GET /favicon.ico HTTP/1.1
7956 113.990424 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… HTTP 437 HTTP/1.1 404 Object Not Found (text/html)
8216 121.581706 2409:40c0:5b:8684:c… 64:ff9b::a3b6:c219 HTTP 772 POST /login/login_results.asp HTTP/1.1 (application/x-www-form-urlencoded)
8236 121.871488 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… HTTP 163 HTTP/1.1 100 Continue
8244 122.286143 64:ff9b::a3b6:c219 2409:40c0:5b:8684:c… HTTP 396 HTTP/1.1 200 OK (text/html)
Frame 8216: 772 bytes on wire (6176 bits), 772 bytes captured (6176 bits) on interface \Dev
Ethernet II, Src: LiteonTechno_94:49:69 (b8:1e:a4:94:49:69), Dst: fe:56:10:6e:8e:0d (fe:56:
Internet Protocol Version 6, Src: 2409:40c0:5b:8684:cde1:dea2:bd89:d131, Dst: 64:ff9b::a3b6
Transmission Control Protocol, Src Port: 59828, Dst Port: 80, Seq: 1, Ack: 1, Len: 698
Hypertext Transfer Protocol
POST /login/login_results.asp HTTP/1.1\r\n
Host: vbsca.ca\r\n
Connection: keep-alive\r\n
Content-Length: 34\r\n
Cache-Control: max-age=0\r\n
Origin: http://vbsca.ca\r\n
Content-Type: application/x-www-form-urlencoded\r\n
Upgrade-Insecure-Requests: 1\r\n
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Ge
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,imag
Referer: http://vbsca.ca/login/login.asp\r\n
Accept-Encoding: gzip, deflate\r\n
Accept-Language: en-US,en;q=0.9\r\n
0040 88 06 50 18 00 ff 78 c2 00 00 50 4f 53 54 20 2f
0050 6c 6f 67 69 6e 2f 6c 6f 67 69 6e 5f 72 65 73 75
0060 6c 74 73 2e 61 73 70 20 48 54 54 50 2f 31 2e 31
0070 0d 0a 48 6f 73 74 3a 20 76 62 73 63 61 2e 63 61
0080 0d 0a 43 6f 6e 6e 65 63 74 69 6f 6e 3a 20 6b 65
0090 65 70 2d 61 6c 69 76 65 0d 0a 43 6f 6e 74 65 6e
00a0 74 2d 4c 65 6e 67 74 68 3a 20 33 34 0d 0a 43 61
00b0 63 68 65 2d 43 6f 6e 74 72 6f 6c 3a 20 6d 61 78
00c0 2d 61 67 65 3d 30 0d 0a 4f 72 69 67 69 6e 3a 20
00d0 68 74 74 70 3a 2f 2f 76 62 73 63 61 2e 63 61 0d
00e0 0a 43 6f 6e 74 65 6e 74 2d 54 79 70 65 3a 20 61
00f0 70 70 6c 69 63 61 74 69 6f 6e 2f 78 2d 77 77 77
0100 2d 66 6f 72 6d 2d 75 72 6c 65 6e 63 6f 64 65 64
0110 0d 0a 55 70 67 72 61 64 65 2d 49 6e 73 65 63 75
0120 72 65 2d 52 65 71 75 65 73 74 73 3a 20 31 0d 0a
0130 55 73 65 72 2d 41 67 65 6e 74 3a 20 4d 6f 7a 69
0140 6c 6c 61 2f 35 2e 30 20 28 57 69 6e 64 6f 77 73
0150 20 4e 54 20 31 30 2e 30 3b 20 57 69 6e 36 34 3b
0160 20 78 36 34 29 20 41 70 70 6c 65 57 65 62 4b 69
Hypertext Transfer Protocol (http), 664 bytes
Packets: 9633 · Displayed: 7 (0.1%)
Profile: Default


Wireshark · Preferences
Appearance
Columns
Font and Colors
Layout
Capture
Expert
Filter Buttons
Name Resolution
Protocols
RSA Keys
Statistics
Advanced
Default interface
Capture packets in promiscuous mode
Capture packets in monitor mode on 802.11 devices
Capture packets in pcapng format
Update list of packets in real time
Interval between updates (ms) 100
Don't load interfaces on startup
Disable external capture interfaces
OK
Cancel
Apply
Help
Ready to load or capture
No Packets
Profile: Default
Capturing from Wi-Fi
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
http
No. Time Source Destination Protocol Length Info
4863 161.736740 2409:40c0:5b:8684:c… 2600:1406:3a00:21::… HTTP 149 GET / HTTP/1.1
4867 161.998373 2600:1406:3a00:21::… 2409:40c0:5b:8684:c… HTTP 292 HTTP/1.1 200 OK (text/html)
Frame 4863: 149 bytes on wire (1192 bits), 149 bytes captured (1192 bits) on interface \Device\NPF
Ethernet II, Src: LiteonTechno_94:49:69 (b8:1e:a4:94:49:69), Dst: fe:56:10:6e:8e:0d (fe:56:10:6e:8e
Internet Protocol Version 6, Src: 2409:40c0:5b:8684:cde1:dea2:bd89:d131, Dst: 2600:1406:3a00:21::17
Transmission Control Protocol, Src Port: 60260, Dst Port: 80, Seq: 1, Ack: 1, Len: 75
Hypertext Transfer Protocol
0000 fe 56 10 6e 8e 0d b8 1e a4 94 49 69 86 dd 60
0010 53 c3 00 5f 06 3f 24 09 40 c0 00 5b 86 84 cd
0020 de a2 bd 89 d1 31 26 00 14 06 3a 00 00 21 00
0030 00 00 17 3e 2e 66 eb 64 00 50 92 30 71 d8 3d
0040 d1 56 50 18 00 ff 31 53 00 00 47 45 54 20 2f
0050 48 54 54 50 2f 31 2e 31 0d 0a 48 6f 73 74 3a
0060 65 78 61 6d 70 6c 65 2e 63 6f 6d 0d 0a 55 73
0070 72 2d 41 67 65 6e 74 3a 20 63 75 72 6c 2f 38
0080 31 30 2e 31 0d 0a 41 63 63 65 70 74 3a 20 2a
0090 2a 0d 0a 0d 0a
Frame (frame), 149 bytes
Packets: 6211 · Displayed: 2 (0.0%)
Profile: Default



3.8 Conclusion

In this practical, we used Wireshark to capture and analyze network packets. We observed its behaviour in promiscuous and non-promiscuous modes and applied filters to trace specific traffic, enhancing understanding of network monitoring and security analysis.