Computer Networking

COMP 177 | Fall 2020 | University of the Pacific | Jeff Shafer

Introducing Wireshark

Recap

Past Topics

- An overview of computer networking

Today's Topic

- Introduce Wireshark
  - General architecture
  - GUI

Why are we learning about Wireshark?

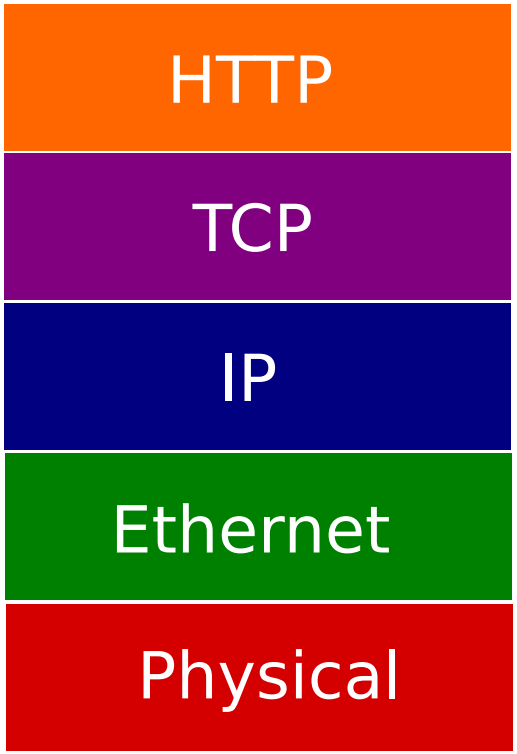
- To *understand how protocols* work, it is helpful to
  - Observe the sequence of packets communicated between network nodes
  - Study the packet details and how protocols work in practice
  - Cause the protocol to do a specific action and check out the result
- *Wireshark* is a free tool that provides such services
  - Supports all major operating systems

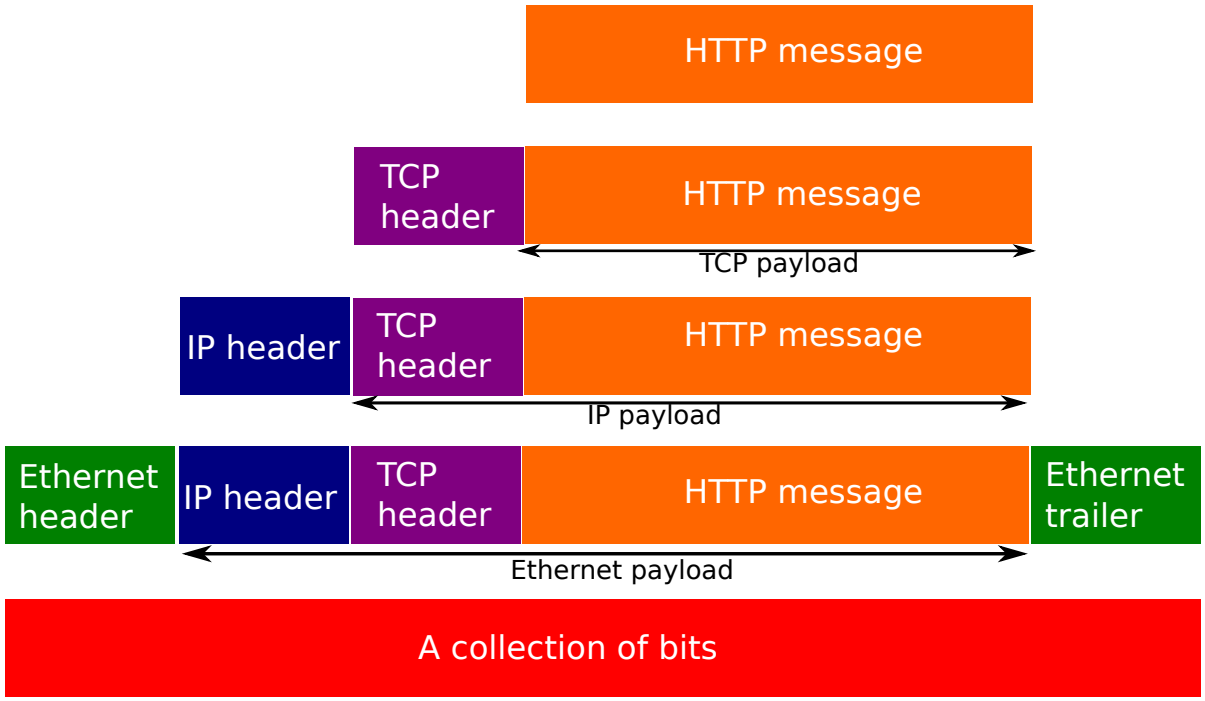
https://www.wireshark.org/

Reading & Parsing Packets

- Wireshark is capable of reading a packet and parsing it into
  - Different protocols headers
  - Different fields in each protocol header
  - … *plus* reporting some *meta-data* about the fields
    - This does not appear directly in the packet header but is based on Wireshark analysis of this packet or even a sequence of packets

Reading & Parsing Packets

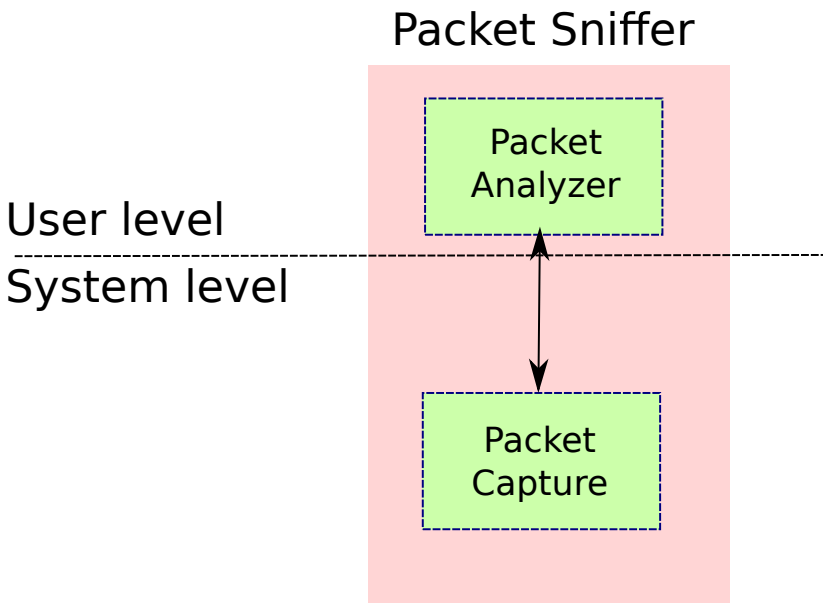




Reading Packets

- Two major ways to obtain packets for analysis:
  - Wireshark can read a collection of *already-captured packets in a file*.
    - File with suffix: `.cap`, `.pcap`, `.pcapng`
    - Such a file may include packets from a remote network
  - Wireshark can capture packets from a *given network interface*
    - This is called *packet sniffing*
    - A packet sniffer collects copies of the sent/received packets, parses, and reports them to the user
    - A packet sniffer acts passively, i.e.,
      - It does not send packets to other machines by itself
      - It does not *spoof* the packets
    - There are many other packet sniffers beside Wireshark
      - tshark, tcpdump, dumpcap, ettercap, etc.

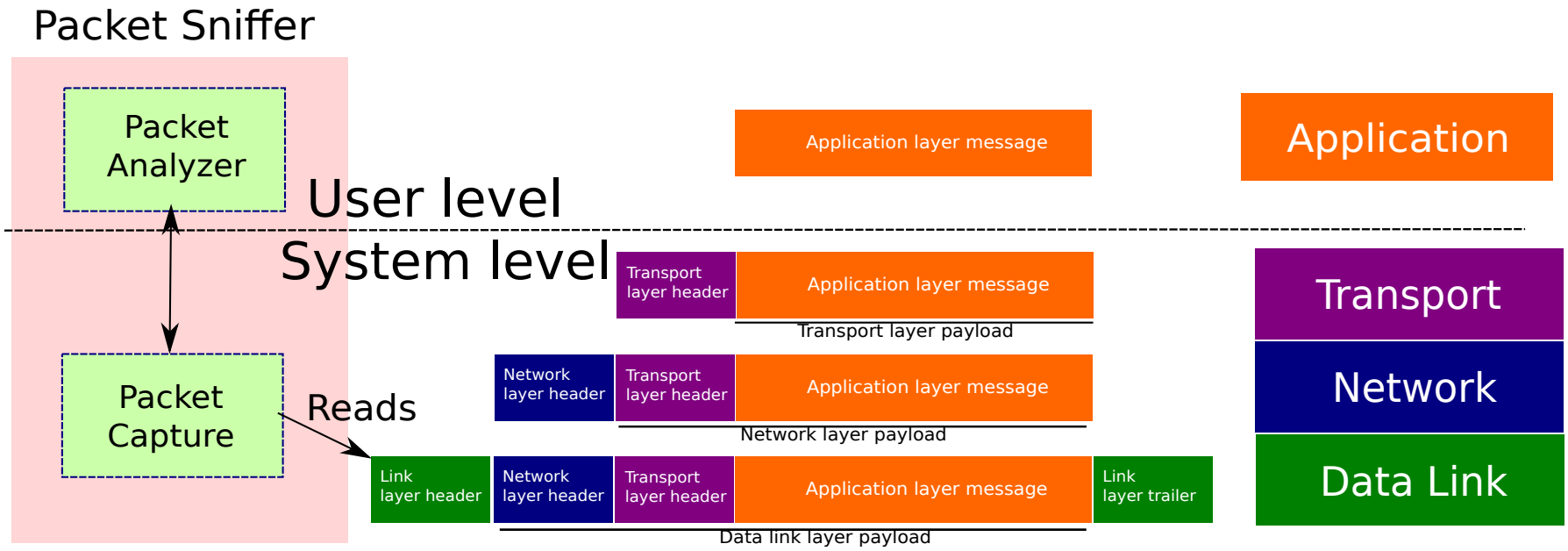
Packet Sniffer Architecture



- A packet sniffer consists of two major components:
  - Packet capture library
    - Runs in system level (operating system)
    - Reads the packets from network interface of the machine
  - Packet analyzer library
    - Runs in user level
    - Parses and reports the packets to the user

Packet Capturing

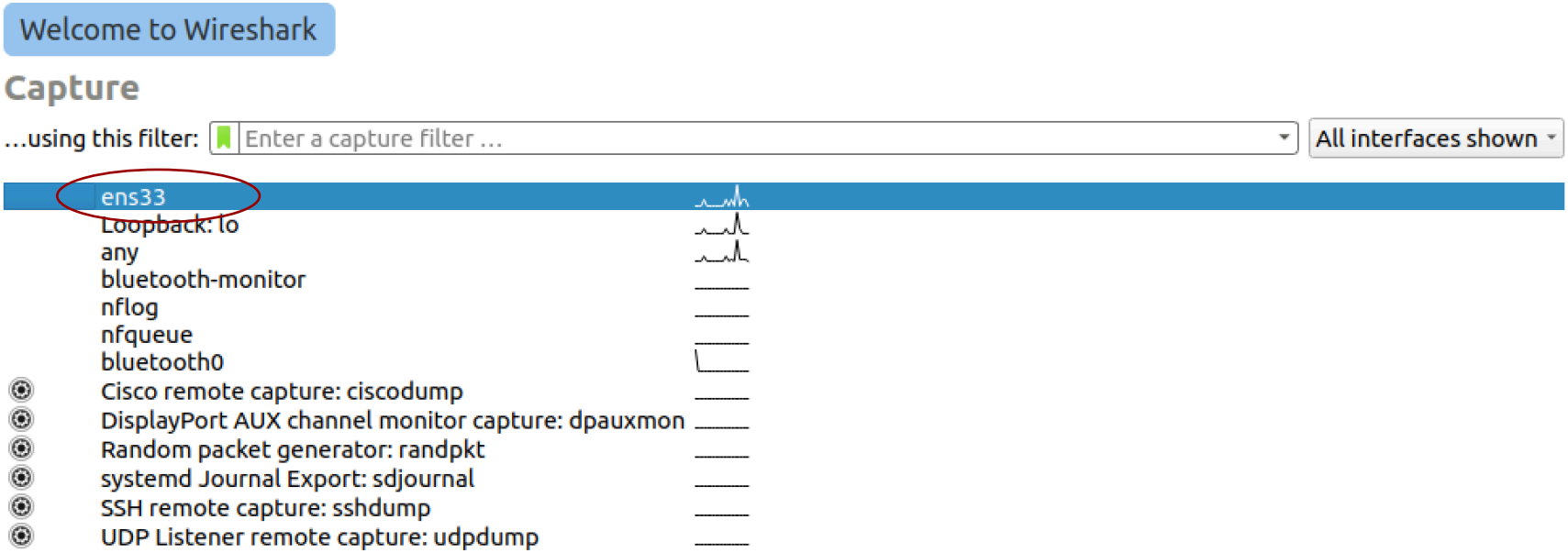
- To capture a packet with all encapsulated headers, it suffices to get a copy of the data link layer frame

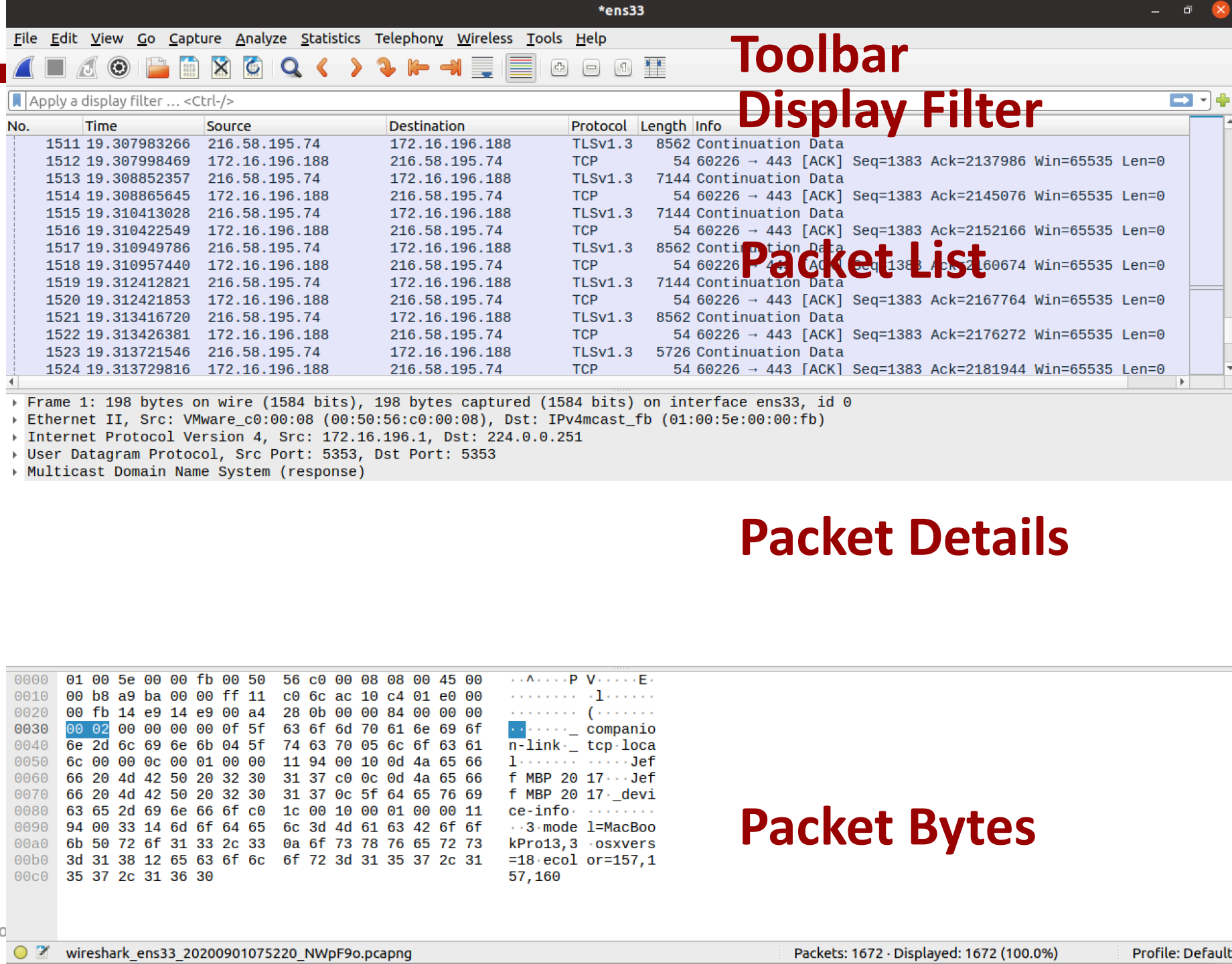


Running Wireshark (Linux)

- Install
  - `$ sudo apt-get install wireshark`
- Configure so non-root users have permission to capture packets (insert your Linux username into the second command)
  - ```
    $ sudo dpkg-reconfigure wireshark-common
    $ sudo usermod -a -G wireshark USERNAME
    ```
- Run Wireshark
  - `$ wireshark`
- Next, interfaces are listed. Selecting one of the interfaces will start capturing packets on that interface.
    ```

Running Wireshark




*ens33
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Toolbar
Apply a display filter ... <Ctrl-/>
Display Filter
No. Time Source Destination Protocol Length Info
1511 19.307983266 216.58.195.74 172.16.196.188 TLSv1.3 8562 Continuation Data
1512 19.307998469 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2137986 Win=65535 Len=0
1513 19.308852357 216.58.195.74 172.16.196.188 TLSv1.3 7144 Continuation Data
1514 19.308865645 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2145076 Win=65535 Len=0
1515 19.310413028 216.58.195.74 172.16.196.188 TLSv1.3 7144 Continuation Data
1516 19.310422549 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2152166 Win=65535 Len=0
1517 19.310949786 216.58.195.74 172.16.196.188 TLSv1.3 8562 Continuation Data
1518 19.310957440 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2160674 Win=65535 Len=0
1519 19.312412821 216.58.195.74 172.16.196.188 TLSv1.3 7144 Continuation Data
1520 19.312421853 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2167764 Win=65535 Len=0
1521 19.313416720 216.58.195.74 172.16.196.188 TLSv1.3 8562 Continuation Data
1522 19.313426381 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2176272 Win=65535 Len=0
1523 19.313721546 216.58.195.74 172.16.196.188 TLSv1.3 5726 Continuation Data
1524 19.313729816 172.16.196.188 216.58.195.74 TCP 54 60226 → 443 [ACK] Seq=1383 Ack=2181944 Win=65535 Len=0
Packet List
Frame 1: 198 bytes on wire (1584 bits), 198 bytes captured (1584 bits) on interface ens33, id 0
Ethernet II, Src: VMware_c0:00:08 (00:50:56:c0:00:08), Dst: IPv4mcast_fb (01:00:5e:00:00:fb)
Internet Protocol Version 4, Src: 172.16.196.1, Dst: 224.0.0.251
User Datagram Protocol, Src Port: 5353, Dst Port: 5353
Multicast Domain Name System (response)
Packet Details
0000 01 00 5e 00 00 fb 00 50 56 c0 00 08 08 00 45 00
0010 00 b8 a9 ba 00 00 ff 11 c0 6c ac 10 c4 01 e0 00
0020 00 fb 14 e9 14 e9 00 a4 28 0b 00 00 84 00 00 00
0030 00 02 00 00 00 00 0f 5f 63 6f 6d 70 61 6e 69 6f
0040 6e 2d 6c 69 6e 6b 04 5f 74 63 70 05 6c 6f 63 61
0050 6c 00 00 0c 00 01 00 00 11 94 00 10 0d 4a 65 66
0060 66 20 4d 42 50 20 32 30 31 37 c0 0c 0d 4a 65 66
0070 66 20 4d 42 50 20 32 30 31 37 0c 5f 64 65 76 69
0080 63 65 2d 69 6e 66 6f c0 1c 00 10 00 01 00 00 11
0090 94 00 33 14 6d 6f 64 65 6c 3d 4d 61 63 42 6f 6f
00a0 6b 50 72 6f 31 33 2c 33 0a 6f 73 78 76 65 72 73
00b0 3d 31 38 12 65 63 6f 6c 6f 72 3d 31 35 37 2c 31
00c0 35 37 2c 31 36 30
Packet Bytes
wireshark_ens33_20200901075220_NWpF9o.pcapng
Packets: 1672 · Displayed: 1672 (100.0%)
Profile: Default

Wrapup

Recap

- Today we discussed
  - Why we need to learn about a packet sniffer
  - The architecture of packet sniffers
  - Wireshark and its GUI

Next Class

- Start discussing data link layer protocols
  - Ethernet (802.3)
  - WiFi (802.11)

Class Activity

CA.1 – Introducing Wireshark

Due tonight at 11:59pm