# CS 585: Computer Networks

**CS 485: Introduction to Computer Networks**
**ECE 440: Introduction to Computer Networks**

*Fall, 2025*

Afsah Anwar, afsah@unm.edu, Farris #2120.
**Office Hours**: Tuesdays & Thursdays: 12:30PM—1:30PM

Muhammad Danish, mdanish@unm.edu, Farris #2145.
**Office Hours**: Mondays & Wednesdays: 10:00AM—11:00AM

## Homework 2

## Guidelines

- Upload on Canvas under Homework 2.
- Late submissions will **NOT** be accepted.
- Students may collaborate among themselves to complete the homeworks/labs.
- Please follow the steps carefully and attach screenshots for each step as evidence.
- For questions 1 and 2, attach screenshots wherever required (not essentially every step).

## Question 1: Getting Wireshark

In order to run Wireshark, you'll need to have access to a computer that supports both Wireshark and the libpcap or WinPCap packet capture library. The libpcap software will be installed for you, if it is not installed within your operating system, when you install Wireshark. See http://www.wireshark.org/download.html for a list of supported operating systems and download sites.

Download and install the Wireshark software:

- Go to http://www.wireshark.org/download.html and download and install the Wireshark binary for your computer.

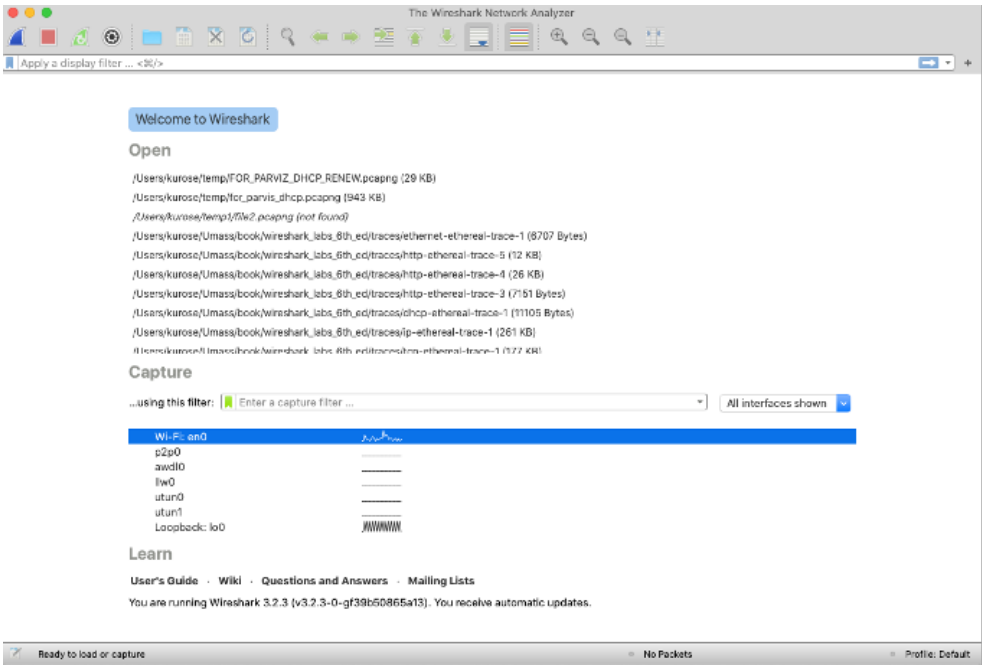


Figure 1: Wireshark - Launch screen

The Wireshark FAQ has a number of helpful hints and interesting tidbits of information, particularly if you have trouble installing or running Wireshark.

## Question 2: Running Wireshark

When you run the Wireshark program, you'll get a startup screen that looks similar to figure 1.

There's not much that's very interesting on this screen. But note that under the Capture section, there is a list of so-called interfaces. For instance, figure 1 lists "Wi-Fi en0" which is the interface for Wi-Fi access. All packets to/from this computer will pass through the Wi-Fi interface, so it's here where we'll want to capture packets.

Click on one of these interfaces to start packet capture, i.e., Wireshark will begin capturing all packets being sent to/from that interface.

The Wireshark interface has five major components:

- The command menus are standard pulldown menus located at the top of the Wireshark window. Of interest to us now are the File and Capture menus. The File menu allows you

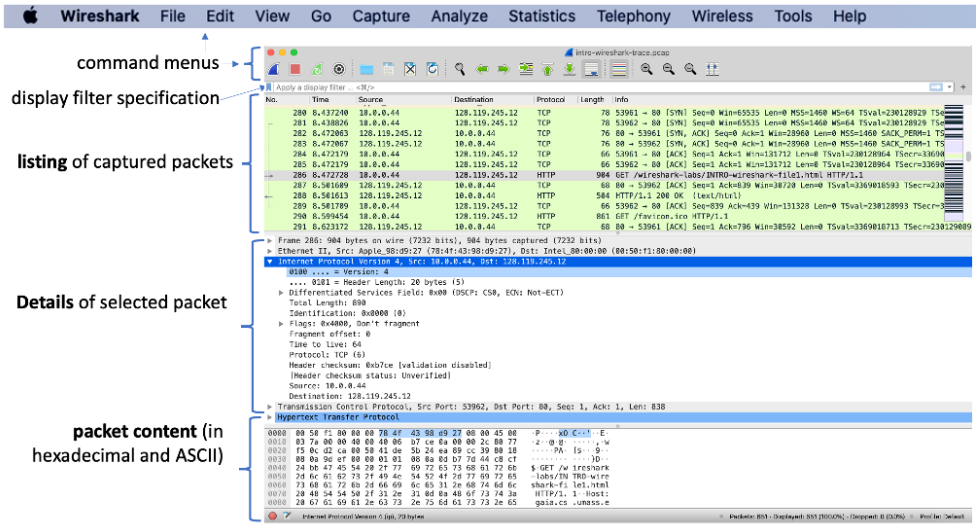


Figure 2: Wireshark screen during capture. It looks very similar when you stop the capture. The only apparent difference will be no details of newer connections being sent to or from your system. You will also see a similar screen when reading a capture file (saved from earlier capture instance).

to save captured packet data or open a file containing previously captured packet data and exit the Wireshark application. The Capture menu allows you to begin packet capture.

- The packet-listing window displays a one-line summary for each packet captured, including the packet number (assigned by Wireshark; note that this is not a packet number contained in any protocol's header), the time at which the packet was captured, the packet's source and destination addresses, the protocol type, and protocol-specific information contained in the packet. The packet listing can be sorted according to any of these categories by clicking on a column name. The protocol type field lists the highest-level protocol that sent or received this packet, i.e., the protocol that is the source or ultimate sink for this packet.

- The packet-header details window provides details about the packet selected (highlighted) in the packet-listing window. (To select a packet in the packet-listing window, place the cursor over the packet's one-line summary in the packet-listing window and click with the left mouse button.). These details include information about the Ethernet frame (assuming the packet was sent/received over an Ethernet interface) and IP datagram that contains this packet. The amount of Ethernet and IP-layer detail displayed can be expanded or minimized by clicking on the plus/minus boxes or right/downward-pointing triangles to the left of the Ethernet frame or IP datagram line in the packet details window. If the

packet has been carried over TCP or UDP, TCP or UDP details will also be displayed, which can similarly be expanded or minimized. Finally, details about the highest-level protocol that sent or received this packet are also provided.

- The packet-contents window displays the entire contents of the captured frame, in both ASCII and hexadecimal format.
- Towards the top of the Wireshark graphical user interface, is the packet display filter field, into which a protocol name or other information can be entered in order to filter the information displayed in the packet-listing window (and hence the packet-header and packet-contents windows). In the example below, we'll use the packet-display filter field to have Wireshark hide (not display) packets except those that correspond to HTTP messages.

## Question 3: Let's Go!

This question allows you to use wireshark to observe what's going on in your system.

Follow the following steps:

1. Launch your VM (if not already).
2. Launch your favourite browser.
3. Clear caches and cookies.
    - Follow the instruction listed by University of Iowa if you are not aware of it: https://its.uiowa.edu
4. Flush DNS Cache: https://kinsta.com/knowledgebase/flush-dns/
    - Flush DNS cache through your browser. Chrome, Brave, Edge, Opera: chrome://net-internals/#dns
    - Mozilla Firefox: about:networking#dns
5. Launch Wireshark.
6. Begin capture.
7. Now access the following urls:
    (a) https://www.x.com/
    (b) https://www.unm.edu/

(c) https://www.cs.unm.edu/

(d) Log in to your canvas. Traverse to the course page. Then log out of Canvas.

(e) Go to https://www.facebook.com/

(f) Go to https://www.usenix.org/

(g) Go to `https://www.iijlab.net/en/members/romain/pdf/romain_sigcomm2017.pdf`.

For each of the URLs, attend to the following questions:

1. What languages (if any) does your browser indicate that it can accept to the server?
2. What is the IP address of your computer? What is the IP address of the URL?
3. What is the status code returned from the server to your browser?
4. Comment on sequence of events that leads to you having access to the website.
5. Comment on the various protocols and port numbers.
6. How many Transport layer protocol packets do you see for every domain/URL listed above. For Canvas, only count until you see the UNM canvas landing page.
7. Comment on how is cookie involved in the interaction.
8. Analyze the difference in accessing a website for the first time vs. second time. (Before you do so, close the browser and restart.) Just choose one for this step.
9. At the end, check the DNS cache to see its size. Did you access all of those? What are the ones that you don't remember accessing?
10. Analyze all the DNS packets. Are they all the same?

Hint: You want want to perform all the access, save it as a PCAP, and then try to answer the questions.

## Question 4: Playing with files.

---

- Clear cookies and DNS as done in the previous question.
- Download `https://www.iijlab.net/en/members/romain/pdf/romain_sigcomm2017.pdf`.

  - Launch your Terminal.
  - Download the file using the link stated above. Use wget (wget <link>) or curl (curl -O <link>).

- Transferring the file:
  - scp /path/to/the/file username@IP:/path/where/you/want/to/save/it

Now, attend to the following:

1. comment on DNS and http requests
2. how many packet were transported during download and upload.
3. Also, analyze the attributes of the datagram.
4. What type of error check is involved.
5. What are your comments on the type of error check involved.

## Question 5: RFC - HTTP 1.1

Other popular RFC is HTTP. Locate RFC 2616. Read the following: Redirection, Client Error, Server Error, Caching, Expiration, and Security Considerations. Now, comment on their necessity.

## Question 6: Short URLs & Redirect Chains

This question explores how short URLs work. Only shorten benign, public URLs (e.g., `https://www.unm.edu/`, `https://www.cs.unm.edu/`, `https://www.usenix.org/`). Follow the steps below.

1. Pick any **two** long targets from Question 3 (e.g., `https://www.unm.edu/` and `https://www.usenix.org/`).
2. Create **two** different short links pointing to those targets using **two different shorteners** (e.g., TinyURL, is.gd, t.ly). Record:

- The short link you created.
- Its long (destination) URL.

3. Clear browser caches and cookies. Flush DNS as in Question 3.
4. Open your browser's Developer Tools → **Network** tab (ensure "Preserve log" is enabled).
5. Visit each short link in your browser **one at a time** while the Developer Tools are open.
6. In Developer Tools, select the first request for the short URL, then:
   - Follow the **redirect chain**.
   - For each hop, note the `Status` code and `Location` header.
   - Take screenshots showing the chain.
7. Identify which domains (shortener, intermediaries, final site) set cookies in the chain, and take a screenshot showing where you observed this.
8. Explain how these cookies are created and what their purposes are.
9. Explain the meaning of each status code observed and the purpose of the `Location` header.