CS 455: Computer Communications and Networking
Lab - 3: TCP

Grading, submission, and late policy:

- You are expected to complete this lab **on your own** (not with a partner)
- This lab accounts for **3%** of your final grade
- The standard late policy applies - Late penalty for this lab will be 15% for each day. Submissions that are late by 3 days or more will not be accepted
- You will submit your solution via Blackboard

Analyzing TCP with Wireshark

In this lab, we will analyze the behavior of the TCP protocol in detail. We will do so by analyzing a trace of the TCP segments sent and received in transferring a file from your computer to a remote server. We'll study TCP's use of sequence and acknowledgment numbers for providing reliable data transfer.

1. Capturing a bulk TCP transfer from your computer to a remote server

Do the following:

- Start up Firefox browser. Note that the following steps have to be performed on the Firefox browser. Google Chrome might use the "QUIC" protocol instead of HTTP which won't allow you to complete the lab below.
- Go to https://www.ietf.org/archive/id/draft-ietf-quic-transport-17.txt and retrieve an ASCII copy of the new QUIC transport protocol. Store this file somewhere on your computer as .txt file.
- Start Wireshark capture.
- Next, go to http://www.phpathak.com/CS555/TCP-wireshark.html.
- You should see a webpage where you can first choose the txt file of QUIC protocol from your computer, and then click the upload button for uploading the file.
- Once the file is uploaded, you should get a "congratulations, upload successful" message. At this point, you can stop the Wireshark capture.

2. A first look at the captured trace

Before analyzing the behavior of the TCP connection in detail, let's take a high-level view of the trace.

- First, filter the packets displayed in the Wireshark window by entering "tcp" (lowercase, no quotes, and don't forget to press return after entering!) into the display filter specification window towards the top of the Wireshark window.

What you should see is a series of TCP and HTTP messages between your computer and phpathak.com. You should see the initial three-way handshake containing a SYN message. You should see an HTTP POST message. In more recent versions of Wireshark, you'll see "**[TCP segment of a reassembled PDU]**" in the Info column of the Wireshark display to indicate that this TCP

segment contained data that belonged to an upper layer protocol message (in our case here, HTTP). You should also see TCP ACK segments being returned from phpathak.com to your computer.

Answer the following questions, by opening the Wireshark captured packet file. When answering a question you should include a screenshot of the packet(s) within the trace.

1. What are the IP address and TCP port number used by the client computer (source) that is transferring the file to phpathak.com? To answer this question, it's probably easiest to select an HTTP message and explore the details of the TCP packet used to carry this HTTP message, using the "details of the selected packet header window".
2. What is the IP address of phpathak.com? On what port number is it sending and receiving TCP segments for this connection?
3. What are the IP address and TCP port number used by your client computer (source) to transfer the file to phpathak.com?

3. TCP basics

Here is a screenshot of the capture I got on my computer. Observe the SYN, SYN-ACK, and ACK at the top, followed by chunks of data, ack, data, ack, .. packets. Look at the "length" column in the screenshot below.

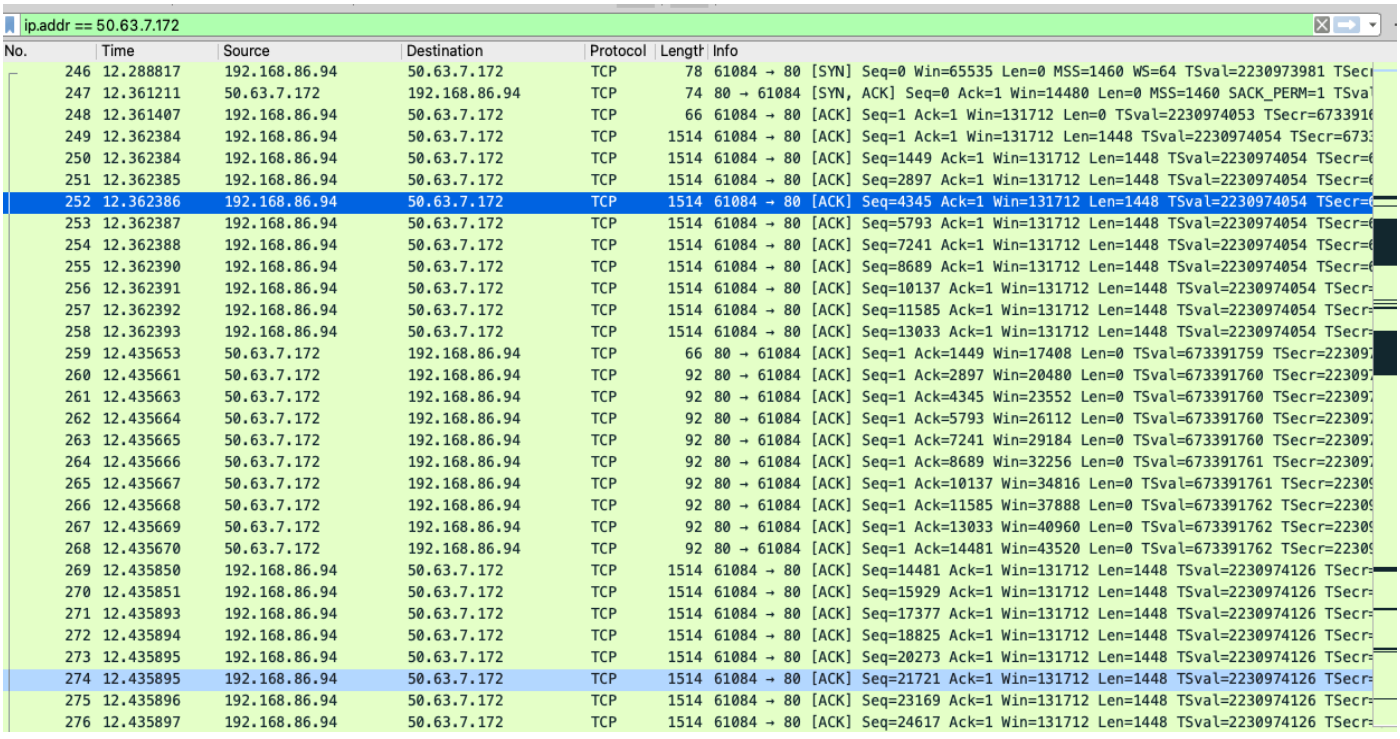

Fig. 1 Example capture for this lab

Based on your track, answer the following questions for the TCP segments.

4. What is the sequence number (raw and relative) of the TCP SYN segment that is used to initiate the TCP connection between the client computer and phpathak.com? What is it in

the segment that identifies the segment as a SYN segment? Include a screenshot of the corresponding packet.

5. What is the sequence number (raw and relative) of the SYNACK segment sent by phpathak.com to the client computer in reply to the SYN? What is the value of the Acknowledgement field in the SYNACK segment? What is it in the segment that identifies the segment as a SYNACK segment? Include a screenshot of the corresponding packet.

6. What is the sequence number of the TCP segment containing the HTTP POST command? The file is uploaded first with many data-ack chunks of packets, followed by the POST command (i.e., the POST command might be later in the capture). Also, keep an eye on [Reassembled PDU in frame: N] Include a screenshot of the corresponding packet.

7. Consider the first 10 TCP segments part of uploading the file (No. 249 onwards in the example screenshot of Fig. 1 given above). At what time was each segment sent? When was the ACK for each segment received? Given the difference between when each TCP segment was sent, and when its acknowledgment was received, what is the RTT value for each of the ten segments? Include a screenshot like Fig. 1 above showing the 10 segments and their corresponding ACKs.

8. What is the length of each of the first 10 TCP segments? What is the length of their corresponding ACK segments?

9. Do you observe any TCP DUP ACKs or retransmitted TCP segments in your capture? Note that this is different for every trace, so it is possible that you might or might not see them. If you do, include a screenshot of such a packet.

What to submit?

- The pcap/pcapng file that you captured using wireshark.
- Prepare your answers in a word/PDF file including the screenshots of packets as asked in the questions above. Submit this file along with the pcap/pcapng capture file on blackboard.