


CSC3511: TCP Wireshark

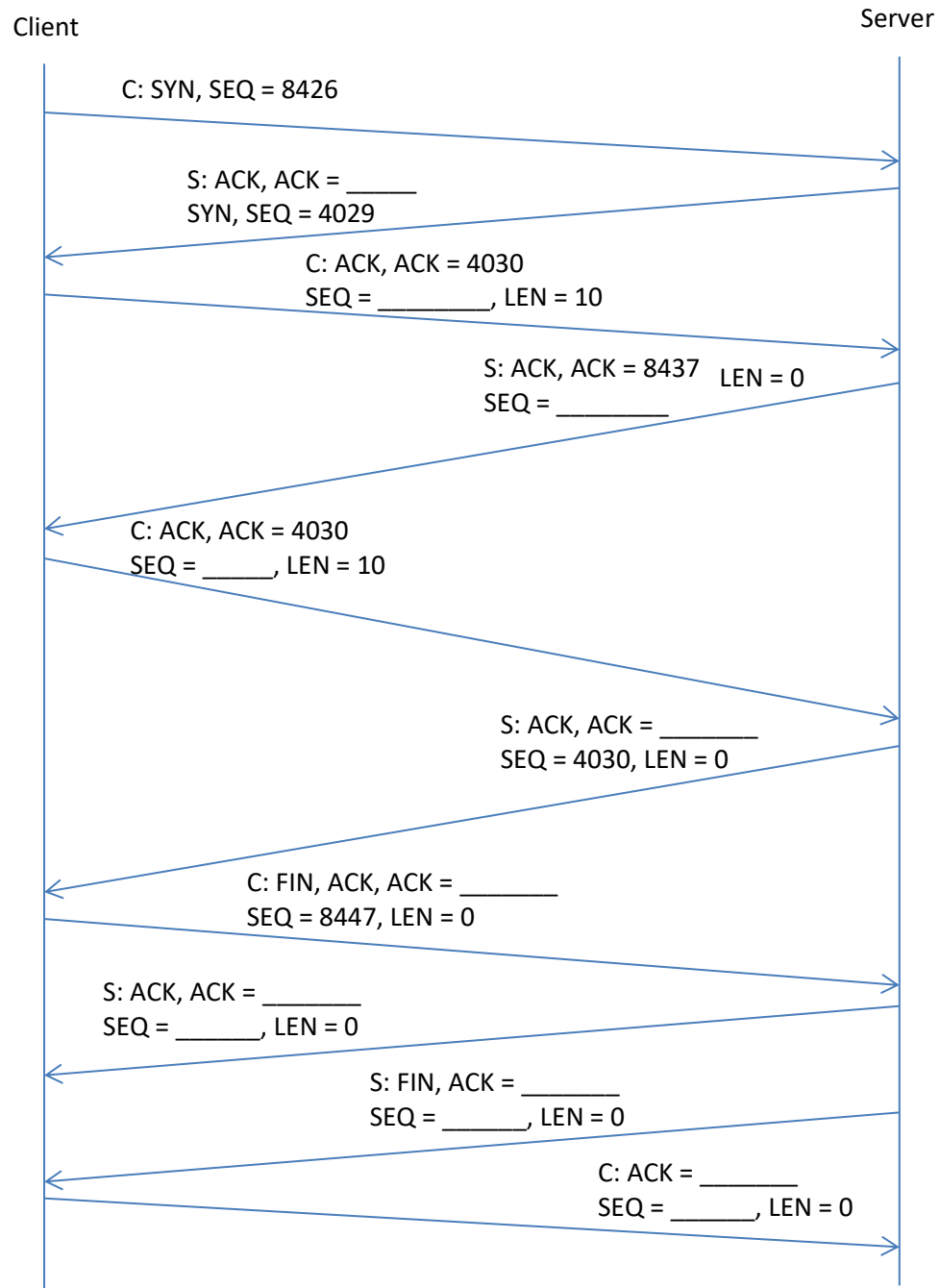
Name: _____

1. Fire up Wireshark on your computer and start capturing. Browse to a webpage (e.g. <http://httpvshttps.com>). Filter with the `http` filter and select the first packet in the request. Turn off auto-scroll . Now turn off the `http` filter by clearing the `http` filter and pressing ENTER. The same packet should be selected, but with other packets around it. If helpful, right-click on the packet and select follow -> TCP Stream to see just the packets that are part of that TCP connection or use a filter like `ip.addr == 45.33.7.16`, using the IP address of the server to only see packets headed for that destination.
 - a. Identify where the TCP source and destination ports appear within the hexadecimal shorthand packet data. Look at a TCP connection to a web server. Write the destination port (on the server) in both decimal and the hexadecimal shorthand for the entire field.
 - b. (if you have time) Identify the TCP sequence number and acknowledgement number in your packet. Write these numbers (in hexadecimal only) here.
 - c. (if you have time) Determine the maximum value of the TCP source port.
 - d. (if you have time) Determine, approximately, the maximum value of the TCP sequence number.

2. SYN and ACK message

- a. Identify the SYN packet sent from the client to the server. Note the sequence number on this packet.
- b. (If you have time) Identify the SYN packet sent to the client in response to the first message from the client. Does the packet have the SYN value you expect?
- c. (If you have time) Identify the second packet from the client to the sender. Does it have the SYN and ACK values you expect?
- d. (If you have time) Identify the SYN and ACK fields within the TCP header. Repeat the above exercises consider the “actual” values instead of those “interpreted” by Wireshark.
- e. (If you have time) Can you see any other TCP packets to the same server?
- f. (If you have time) Explore the other fields in the packet. What questions do you have about them?

3. TCP with stop-and-wait. Fill in the blanks in the following TCP stream (the numbers are above the arrows they describe). No need to fill in the packets after the first FIN at first.



5. TCP with pipelined sliding window. The numbers are above the arrows they describe. Every packet

