**Chapter 5 [30 points]**

**5.1 What is meant by a control plane that is based on logically centralized control? In such cases, are the data place and the control plane implemented within the same device or tin separate devices? Explain.**

* **A control plane that is based on logically centralized control in a SDN controller, means that it can see information about all the networks, including the network architecture and the hosts. The forwarding plane is the data, and is stored on a different device then the control plane. This allows for better forwarding polices and networking. The data plane consists of network switches that execute “match plus action” rules in the flow tables. The control plane is made of servers and software’s that are determined in the switches flow tables. Here we can see how both live on separate devices.**

**5.2 Compare and contrast link-state and distance-vector routing algorithm.**

* **Link-state routing algorithms use globally verifiable information about the network to calculate the least-cost path between source and destination. This is seen in the Operate Shortest Path First protocol and will calculate a path that uses the least bandwidth using the Dijkstra algorithm. Distance vector routing will use the path with the lowest amount of hops between nodes and is present in the RIP protocol using the Bellman-Ford algorithm. Both our significant in regards to selecting the best routing measure while RIP is slow and uses much less memory, OSPF is fast but memory intensive.**

**5.3 Why are different inter-AS and intra-AS protocols used in the Internet?**

* **Intra-AS protocols refer to routing within the Autonomous System and uses an interior gateway protocol such as RIP or OSPF. This allows for routing between nodes within the same AS. Inter-AS protocols use external gateway protocols like BGP to create routes to other AS outside of each other’s networks. This is used differently on the internet for nodes within your AS, or for nodes without your AS, and will route accordingly for internal or external routes. Intra-AS is more performance oriented while Inter-AS is more policy oriented.**

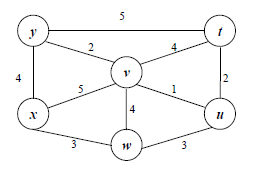
**5.4 Describe the main role of the communication layer, the network-wide state-management layer, and the network-control application layer in an SDN controller.**

* **In a software defined network controller, the communication layer is what communicates between the SDN controller and the controlled switches. This is where SNMP can reside to communicate. The Network-Wide State Management layer creates a distributed database of network link states, switches and services. This is where host info, switch info, link-states, and statistics are held. The Network-Control Application layer is what hosts your abstractions to your API’s this is where your Restful API’s are stored to be called upon.**

**5.5 Briefly explain two modes to convey information in the SNMP protocol.**

* **One mode to convey information in the SNMP protocol is the traditional request-response mode. This is where a managing server will send a request to an agent who will then send a response. The other mode sends unsolicited messages known as trap messages. The agent will send a message to the server even if they didn’t request it. This is done when certain triggers are performed to alert the managing server of a situation. An example why an agent would notify the managing server would be too many incoming packets, or links going up or down.**

**5.6 Consider the following network. With the indicated link costs, use Dijkstra’s shortest-path algorithm to compute the shortest path from *x* to all network nodes. Show how the algorithm works by computing a table similar to the textbook example. In cases when several candidate nodes have the same minimal costs, choose a node according to non-decreasing alphabetical order.**

****

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **STEP** | **N’** | **D(t),p(t)** | **D(u),p(u)** | **D(v),p(v)** | **D(w),p(w)** | **D(y),p(y)** |
| **0** | **x** | **∞** | **∞** | **5,x** | **3,x** | **4,x** |
| **1** | **xw** | **∞** | **6,w** | **5,x** |  | **4,x** |
| **2** | **xwu** | **4,x** |  | **5,x** |  | **4,x** |
| **3** | **xwuv** | **9,t** |  |  |  | **4,x** |
| **4** | **xwuvy** | **8,u** |  |  |  |  |
| **5** | **xwuvyt** |  |  |  |  |  |

**5.7 Consider the count-to-infinity problem in the distance vector routing. Will the problem occur if we decrease the cost of a link? Why?**

* **The count-to-infinity problem is caused because the Bellman-Ford algorithm can’t prevent loops. A loop can occur if an interface goes down or two-routers send updates at the same time. If you decrease the cost of a link it will not cause a loop there for not creating the count-to-infinity problem. This is because connecting two nodes with a link is equivalent to decreasing the link infinite to a finite weight.**

**Chapter 6 [30 points]**

**6.1 Suppose the information content of a packet is the bit pattern 1010 0000 1001 1101 and an even parity scheme is being used. What would the value of the field containing the parity bits be for the case of a two-dimensional parity scheme? Your answer should be such that a minimum-length checksum field is used.**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **1** | **0** | **1** | **0** | **0** |
| **0** | **0** | **0** | **0** | **0** |
| **1** | **0** | **0** | **1** | **0** |
| **1** | **1** | **0** | **1** | **1** |
| **1** | **1** | **1** | **0** | **0** |

**6.2 Refer to the Cyclic Redundancy Check (CRC) codes discussed in the textbook and follow the conventions used in the textbook. Consider the 5-bit generator, G=10101, and suppose that D has the value 1010101000. What is the value of R? Show the process how you obtain the solution.**

**D= 1010101000, G =10101**

**R= (D/G) = (1010101000 / 10101)**

**6.3 Consider two nodes A and B uses the slotted ALOHA protocol to contend for a channel. Suppose node *pA* and *pB* are A and B’s retransmission probabilities. Provide a formula for node A’s average throughput. What is the total efficiency of the protocol with these two nodes?**

* **Node A’s Average throughput = (pA(1-pB))**

**Total Efficiency = ((pA(1-pB)) + (pB (1-pA)))**

**6.4 A disadvantage of the content of approach for LANs, such as CSMA/CD, is the capacity wasted due to multiple stations attempting to access the channel at the same time. Suppose that time is divided into discrete slots, with each of N stations attempting to transmit with probability *p* during each slot. What fraction of slots is wasted due to multiple simultaneous transmission attempts?**

* **The fraction of slots wasted due to multiple simultaneous transmission attempts is directly equal to the probity that there are more than 2 transmission attempts in any given slot.**

**Probability of 2 or More Attempts = (1 – (Pr(No Attempts)) – (Pr(Only 1 Attempt)))**

**6.5 In CSMA/CD, after the 4th collision, what is the probability that a node chooses k=4? The result k=4 corresponds to a delay of how many seconds on a 100 Mbps Ethernet?**

|  |  |
| --- | --- |
| **Collision = 4th**  **(2^4) = 16.** | **Delay Formula = k\*512\*bit-time**  **Bit-time = (1/100mbps) = 0.01 microseconds.**  **So Delay = 4\*512\*.01mircoseconds** |
| **Probability = 1/16.** | **Delay = 20.48 microseconds.** |

**6.6 Why is an ARP query sent within a broadcast frame? Why is an ARP response sent within a frame with a specific destination MAC address?**

* **An ARP query is sent within a broadcast frame because to send an IP datagram you need to know both the IP address and MAC address. In this situation the ARP requester does not know the MAC address, so it is sent within the broadcast frame. The broadcast frame is sent to all IP addresses till a match is found. If it was not sent in the broadcast frame it would not find the matching IP address to find the corresponding MAC address. The matching IP address then returns an ARP response with a frame with the destination MAC address so that further communication doesn’t involve a broadcast throughout the network again.**

**6.7 Why the Ethernet is called a multiple access protocol? Why it is referred to as an unreliable and connectionless approach?**

* **Ethernet is a multiple access protocol because each node on the network has access to send messages simultaneously. This of course can lead to collisions on your network.**

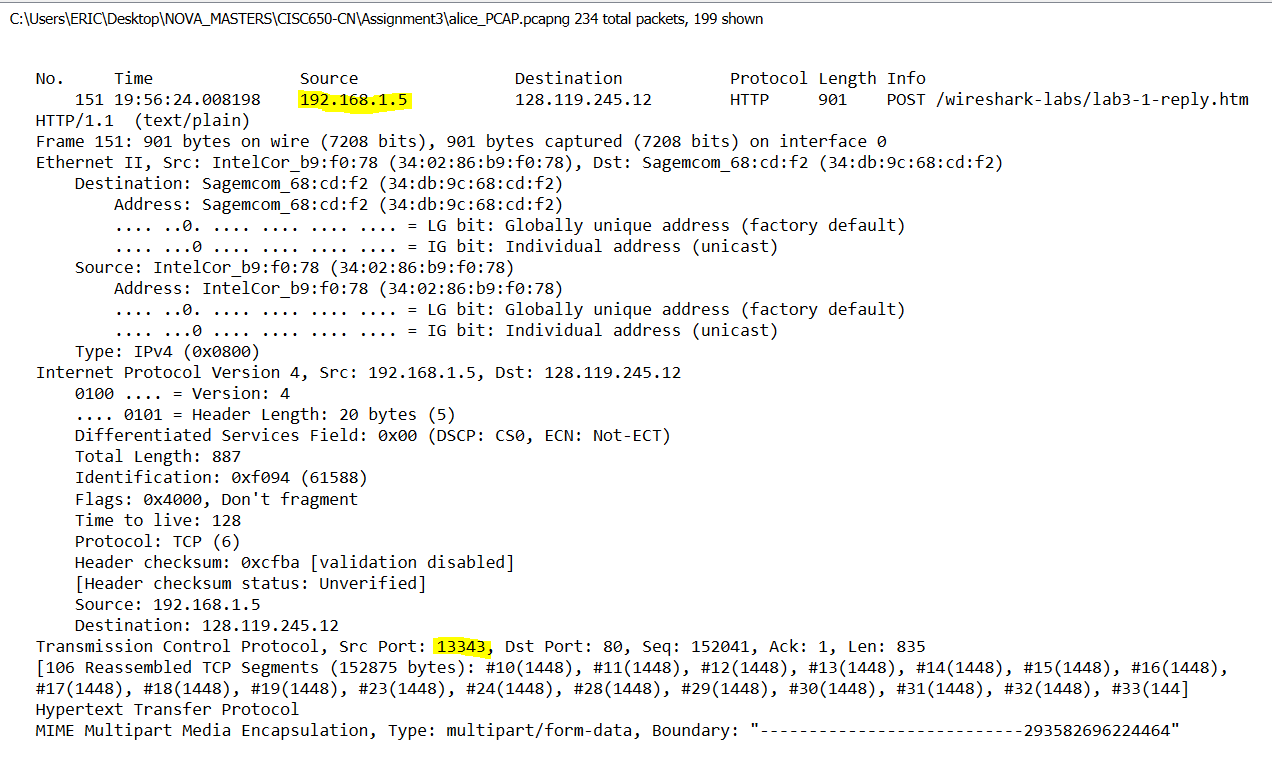
**It is considered unreliable and connectionless because there are no acknowledgments when a packet is received. Its connectionless because it does not establish a connection to the receiver and will send the packets at will. It’s unreliable in the fact that if a receiver gets the packet it will not respond whether the packet has been accepted or dropped. This leaves the sender with a sense of unreliability if the packet got delivered or not.**

**Part 3. Practical assignment [40 points]**

**Please finish questions 1 – 12 listed in file “Wireshark\_TCP.pdf”. Questions 13 – 14 are optional.**

1. What is the IP address and TCP port number used by the client computer (source)that is transferring the file to gaia.cs.umass.edu?

* Source Address = 192.168.1.5
* Source Port =13343

(Highlighted in packet print out below.)

2. What is the IP address of gaia.cs.umass.edu? On what port number is it sending

and receiving TCP segments for this connection?

- IP Address: 128.119.245.12

- Sending and receiving on port 80.

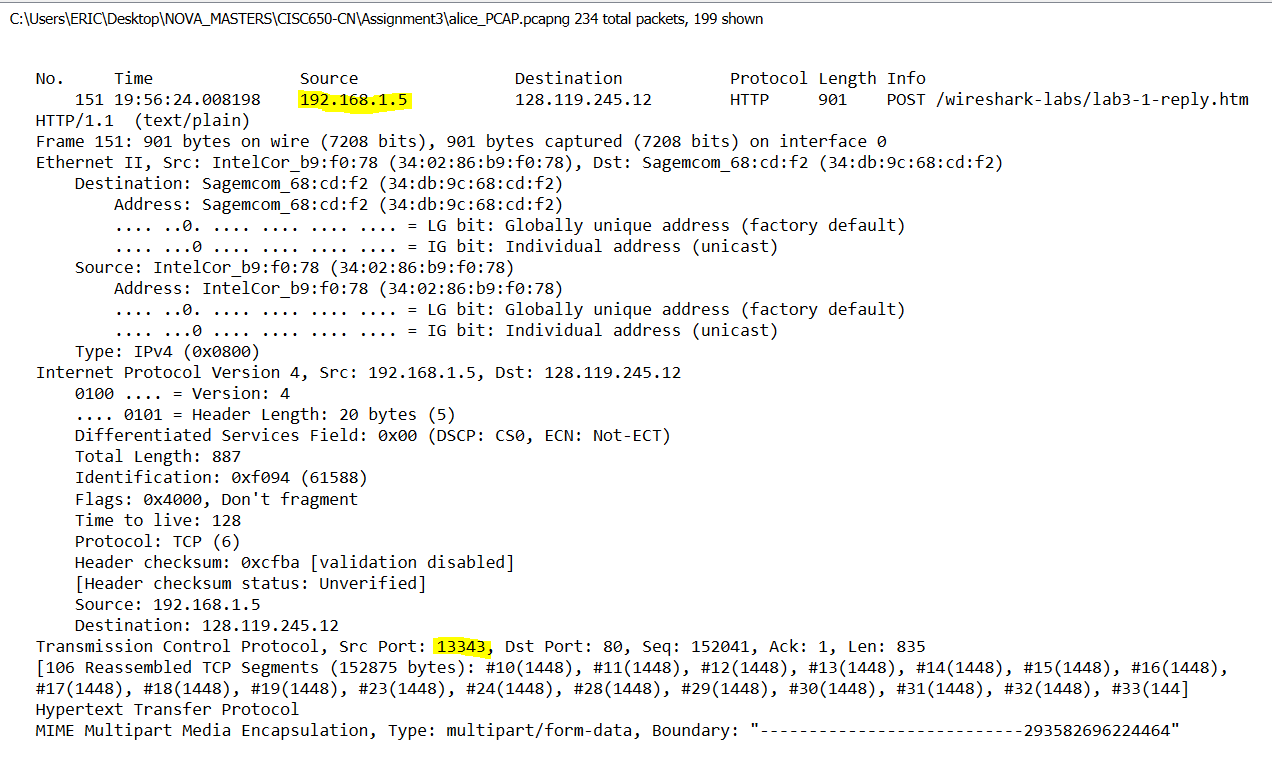
(Shown below.)



3. What is the IP address and TCP port number used by your client computer

(source) to transfer the file to gaia.cs.umass.edu?

* Source Address = 192.168.1.5
* Source Port =13343

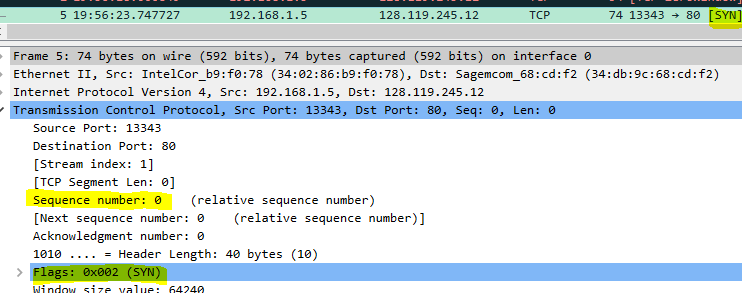
(Highlighted in packet print out below.)

4. What is the sequence number of the TCP SYN segment that is used to initiate the

TCP connection between the client computer and gaia.cs.umass.edu? What is it in the segment that identifies the segment as a SYN segment?

* Sequence number = 0.
* Can be identified with Flag set to 0x002 which indicates SYN.

(Highlights below show sequence number and Flag.)



5. What is the sequence number of the SYNACK segment sent by gaia.cs.umass.edu

to the client computer in reply to the SYN? What is the value of the

Acknowledgement field in the SYNACK segment? How did gaia.cs.umass.edu

determine that value? What is it in the segment that identifies the segment as a

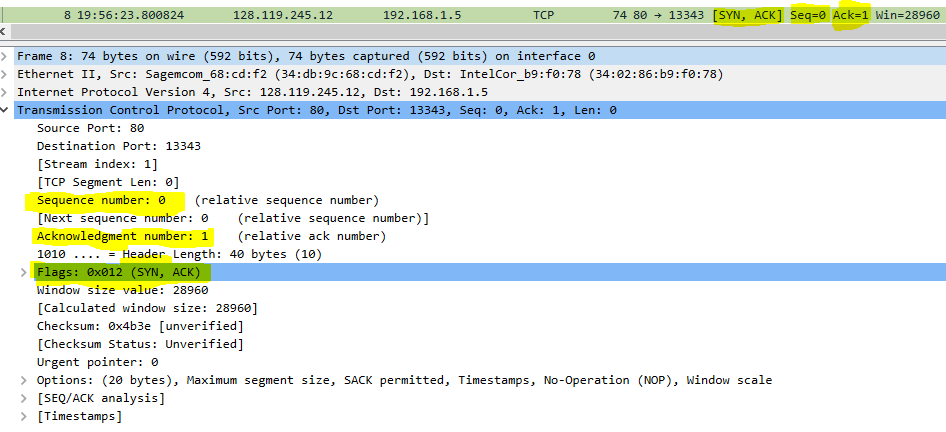
SYNACK segment?

- Sequence number = 0.

- Acknowledgement = 1. (This is because the ACK will be one up from the original senders SYN packets sequence number. In this case our original sender’s sequence number was 0, so we are replying a SYN/ACK of 1.)

- Flag is set as 0x012 to represent SYN/ACK.

(Highlighted Below.)



6. What is the sequence number of the TCP segment containing the HTTP POST

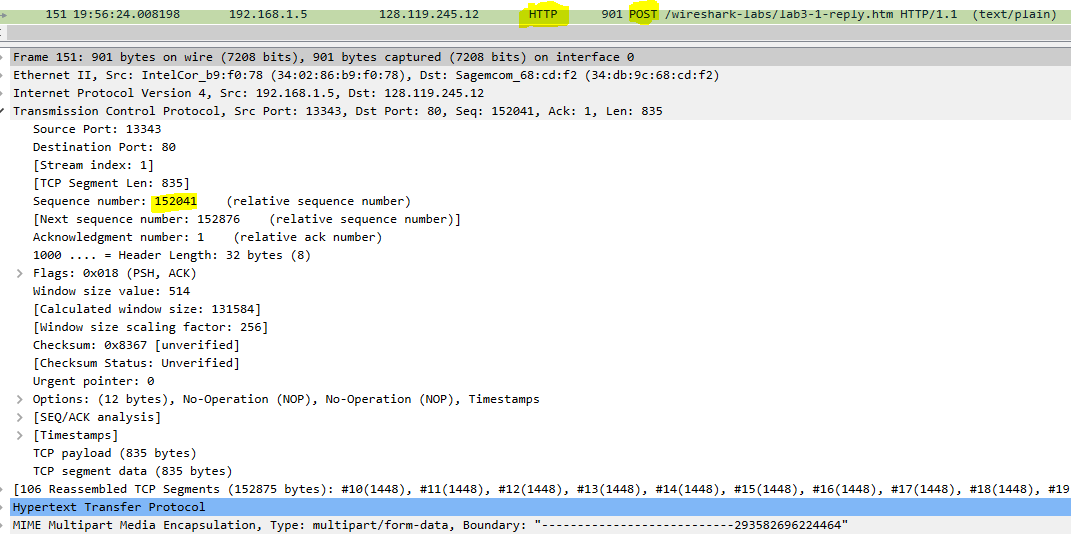
command? Note that in order to find the POST command, you’ll need to dig into

the packet content field at the bottom of the Wireshark window, looking for a

segment with a “POST” within its DATA field.

* Sequence Number = 152041.

(HTTP POST and sequence shown below.)



7. Consider the TCP segment containing the HTTP POST as the first segment in the

TCP connection. What are the sequence numbers of the first six segments in the TCP connection (including the segment containing the HTTP POST)? 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

acknowledgement was received, what is the RTT value for each of the six

segments? What is the EstimatedRTT value (see Section 3.5.3, page 242 in

text) after the receipt of each ACK? Assume that the value of the

EstimatedRTT is equal to the measured RTT for the first segment, and then is

computed using the EstimatedRTT equation on page 242 for all subsequent

segments.

*Note:* Wireshark has a nice feature that allows you to plot the RTT for

each of the TCP segments sent. Select a TCP segment in the “listing of

captured packets” window that is being sent from the client to the

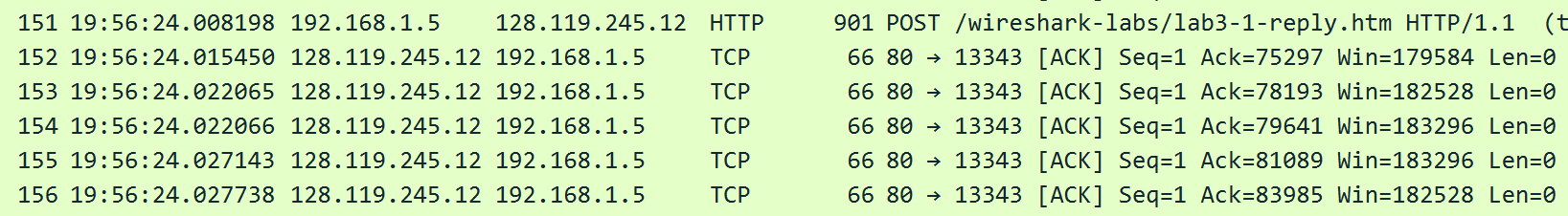
gaia.cs.umass.edu server. Then select: *Statistics->TCP Stream Graph-*

*>Round Trip Time Graph.*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Sequence #* | *Time Sent* | *ACK Received Time in seconds* | *RTT Value* | *Estimated RTT*  *(Estimated using .08080200)* |
| *152041* | *19:56:24.008198* | *No Ack for HTTP POST\** | *N/A* | *N/A* |
| *1* | *19:56:24.015450* | *.08080200* | *19:56:24.096252* | *19:56:24.080802* |
| *1* | *19:56:24.022065* | *.08741400* | *19:56:24.109479‬* | *19:56:24.01092675* |
| *1* | *19:56:24.022066* | *.08741400* | *19:56:24.109480* | *19:56:24.01092675* |
| *1* | *19:56:24.027143* | *.09249100* | *19:56:24.119633* | *19:56:24.011561375* |
| *1* | *19:56:24.027738* | *.08663500* | *19:56:24.114373* | *19:56:24.084998375* |

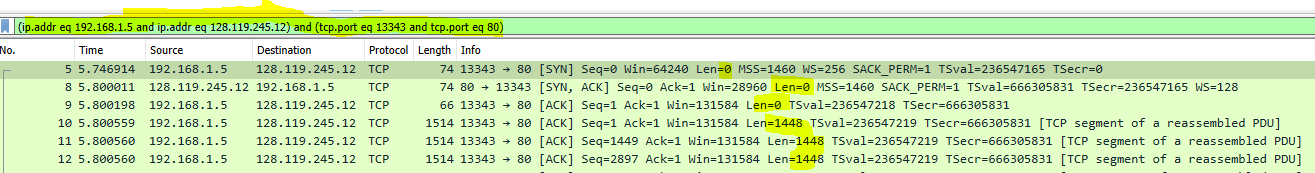
*\*Verified no ACK on HTTP using wireshark command*

*(tcp.ack==152042) displayed nothing.*



8. What is the length of each of the first six TCP segments?

|  |  |
| --- | --- |
| Sequence # | Length |
| 5 | 0 |
| 8 | 0 |
| 9 | 0 |
| 10 | 1448 |
| 11 | 1448 |
| 13 | 1448 |



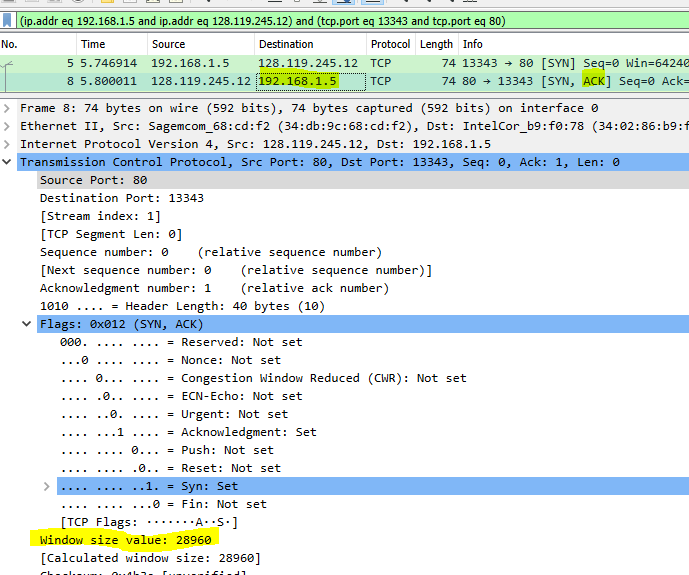
9. What is the minimum amount of available buffer space advertised at the received

for the entire trace? Does the lack of receiver buffer space ever throttle the

sender?

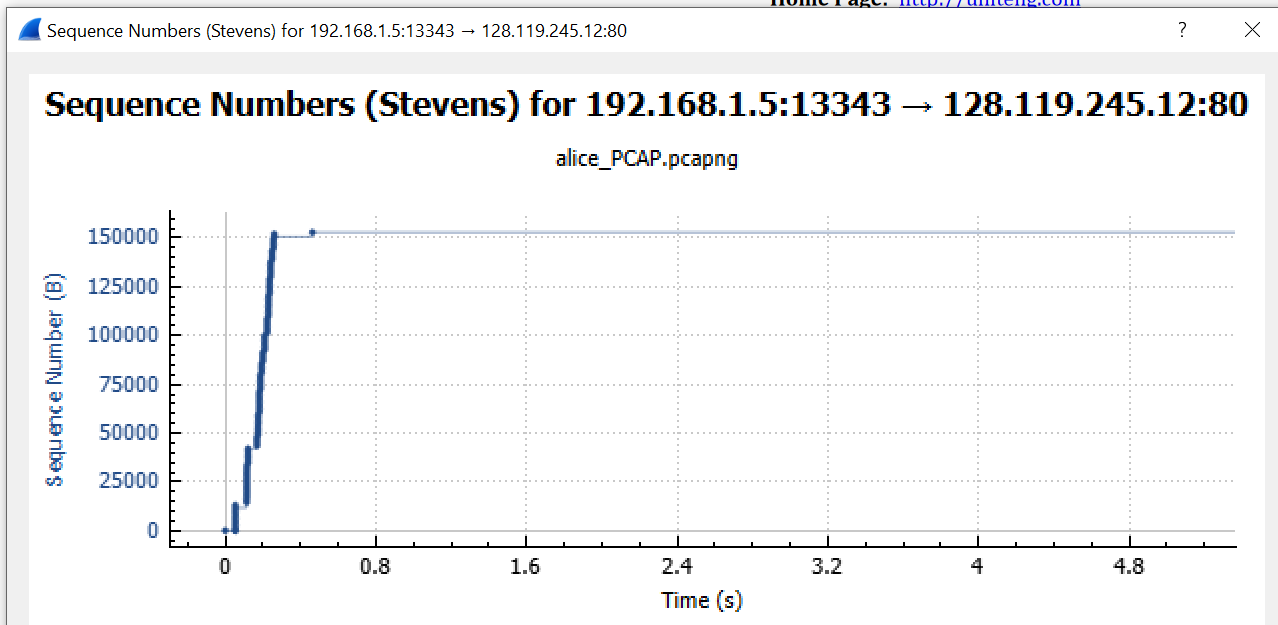
* Minimum buffer space advertised = 28960. (Indicated from first ACK of server)

This does not throttle due to lack of receiver buffer space.



10. Are there any retransmitted segments in the trace file? What did you check for (in

the trace) in order to answer this question?

* There are no retransmitted packets this is seen in the Stevens Sequence number graph because no segments are increasing monotonically.
* 

11. How much data does the receiver typically acknowledge in an ACK? Can you

identify cases where the receiver is ACKing every other received segment (see

Table 3.2 on page 250 in the text).

* The Data the receiver typically acknowledges in an ACK is the difference between the ACK numbers of two consecutive ACKS. An example of where the receiver is ACKing every other received segment starts on segment 152.



12. What is the throughput (bytes transferred per unit time) for the TCP connection?

Explain how you calculated this value.

Alice.txt size on disk = 155,648 bytes

Download time = Difference between time at first TCP segment(segment 5) and last ACK(segment 217) = 5.48336 seconds.

Throughput = (155,648 / 5.48336) =28,385.5155 Bytes per second.





\*\*Optional\*\*

Answer the following questions for the TCP segments the packet trace *tcp-etherealtrace-*

*1* in http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip

13. Use the *Time-Sequence-Graph(Stevens*) plotting tool to view the sequence

number versus time plot of segments being sent from the client to the

gaia.cs.umass.edu server. Can you identify where TCP’s slowstart phase begins

and ends, and where congestion avoidance takes over? Comment on ways in

which the measured data differs from the idealized behavior of TCP that we’ve

studied in the text.

* The slow start appears to begin around .06 seconds and ends around .12 seconds. It looks like congesting avoidance takes place at about .13 seconds because it cuts down the amount being sent. This is different than the idealized behavior because it is only using a fraction of the window size instead of the idealized ½ to 1/3 size.

