**Exercise1  
Question1**

The IP address of **gaia.cs.umass.edu** is **128.119.245.12**, also the **port number** is **80**.

The IP address of the **client computer** is **192.168.1.102**, also the **port number** is **1161**.

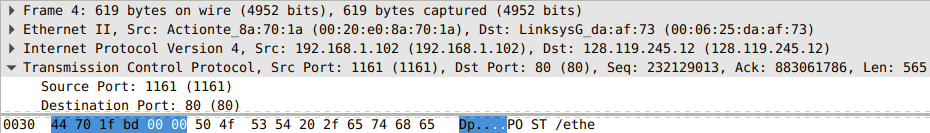
We can also find the result from two pictures given below.





**Question2**

We can find the keyword POST in frame 4. Therefore, we can get the **sequence number**, which is **232129013**. Here is the picture result of this question.



**Question3**

Based on the formula, we can get the following table.

EstimatedRTT = (1-a)\*EstimatedRTT + a\*SampleRTT

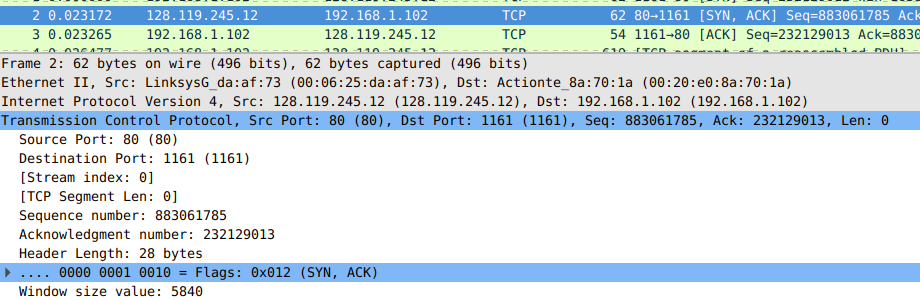
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Index | Sequence Number | Time sent | Time ACK received | RTT | EstimatedRTT |
| 1 | 232129013 | 0.026477 | 0.053937 | 0.027460 | 0.027460 |
| 2 | 232129578 | 0.041737 | 0.077294 | 0.035557 | 0.028472 |
| 3 | 232131038 | 0.054026 | 0.124085 | 0.070059 | 0.033670 |
| 4 | 232132498 | 0.054690 | 0.169118 | 0.114428 | 0.043765 |
| 5 | 232133958 | 0.077405 | 0.217299 | 0.139894 | 0.055781 |
| 6 | 232135418 | 0.078157 | 0.267802 | 0.189645 | 0.072514 |

**Question4**

The length of the first **TCP segment** is **565**. From the **second segment** to the **sixth segment**, the length of these **5 segments** are the same which is **1460**.

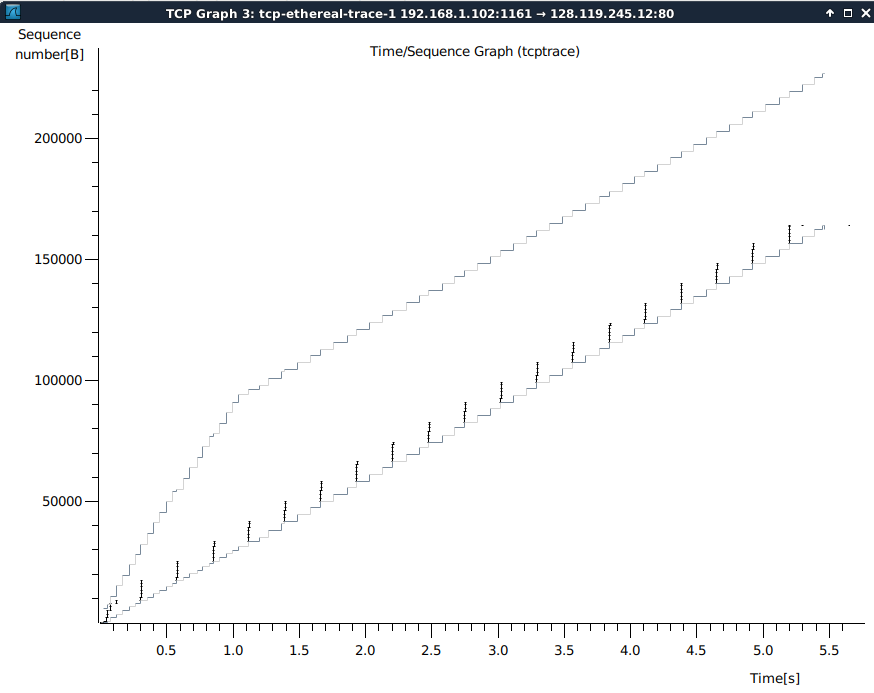
**Question5**

The **minimum amount of buffer space** is **5840**. **No**, it doesn’t. This is because the window size is **increasing** (never decreased). Here is the picture result of question 5.



**Question6**

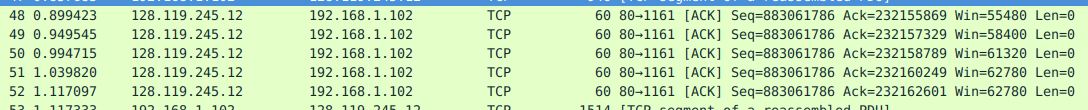
No, there aren’t. This is because the **sequence numbers are incremental**. We can also find this result in the Time/Sequence Graph printed by Wireshark. From the graph given below, the **sequence numbers are increasing** over time. Here is the picture result of this question.

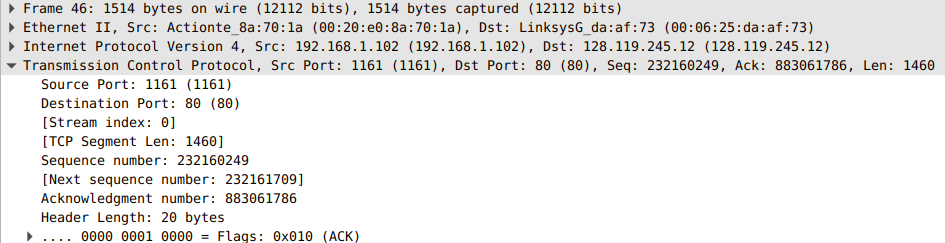


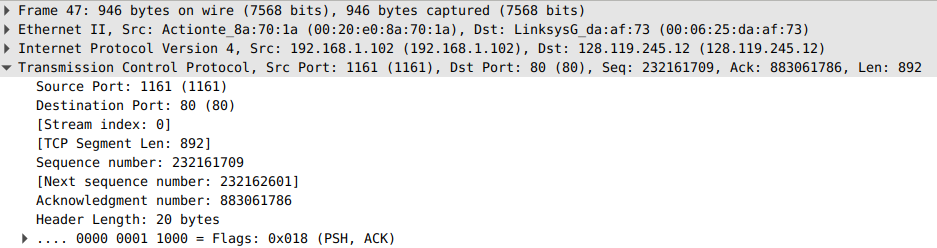
**Question7**

The receiver typically **acknowledges 1460 bytes**.

**Number 46**, which sequence number is **232160249**, **didn’t receive** an ACK with number **232161709**. We can also find this result in the first picture, which means the receiver didn’t get the ACK. Furthermore, **number 46 and number 47** **share one** **single ACK** which number is **232162601**. Here are the pictures of the question7.







**Question8**

The last sequence number is 232293103. The first sequence number is 232129013. By using these 2 sequence numbers, we can get the transferred bytes are 232293103 – 232129013 = **164090** bytes. We can also find the last ACK time is 5.455830s. Also, the first-time sending message which is 0.026477s. We can get the time which is 5.455830 - 0.026477 = **5.429353s**. Therefore, we can calculate the **throughput** which is **164090 / 5.429353s = 30222.754 Bytes/s**.

**Exercise2  
Question1**

We can get the result from the SYN data packet. Therefore, the **sequence number** is **2818463618**.



**Question2**

The **sequence number** is **1247095790**. The value of the **Acknowledgement field is 1**. When the server receives an SYN, then it returns an SYNACK to confirm the connection. To show this in the flags, it sets the Acknowledgement and SYN field as value 1. This is due to the Three-way Handshake.



**Question3**

The **sequence number is 2818463619**. The value of the **Acknowledgement field is 1**.

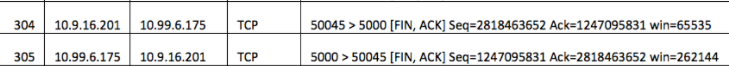
**Yes, it contains data.** By checking the value of ACK and Sequence number, we can find the data is . Therefore, it contains data, which amount is 33 bytes.





**Question4**

**Both server and client did the active close.** This is because after the client sent the FIN message, the server also sent the FIN message. The FIN segment is the last-sending on both the server-side and the client-side. This means both the server and the client want to close the connection. Therefore, this is a **simultaneous close**.



**Question5**

**Client ----sending to ----> Server**: 2818463652 – 2818463619 = **33 Bytes**

**Server ----sending to ----> Client**: 1247095831 – 1247095791 = **40 Bytes**

By doing the subtraction, we can find the relationship that if we minus the initial sequence number from the final ACK, the value of the subtraction would equal the number of bytes which have been transferred in this process plus one SYN and one FIN.

**Final ACK = initial Sequence number + Transferred bytes + one SYN + one FIN**