Exercise 1: Setting up NS2 simulation for measuring TCP throughput

Question 1: Why the throughput achieved by flow tcp2 is higher than tcp1 between time span 6 sec to 8 sec?

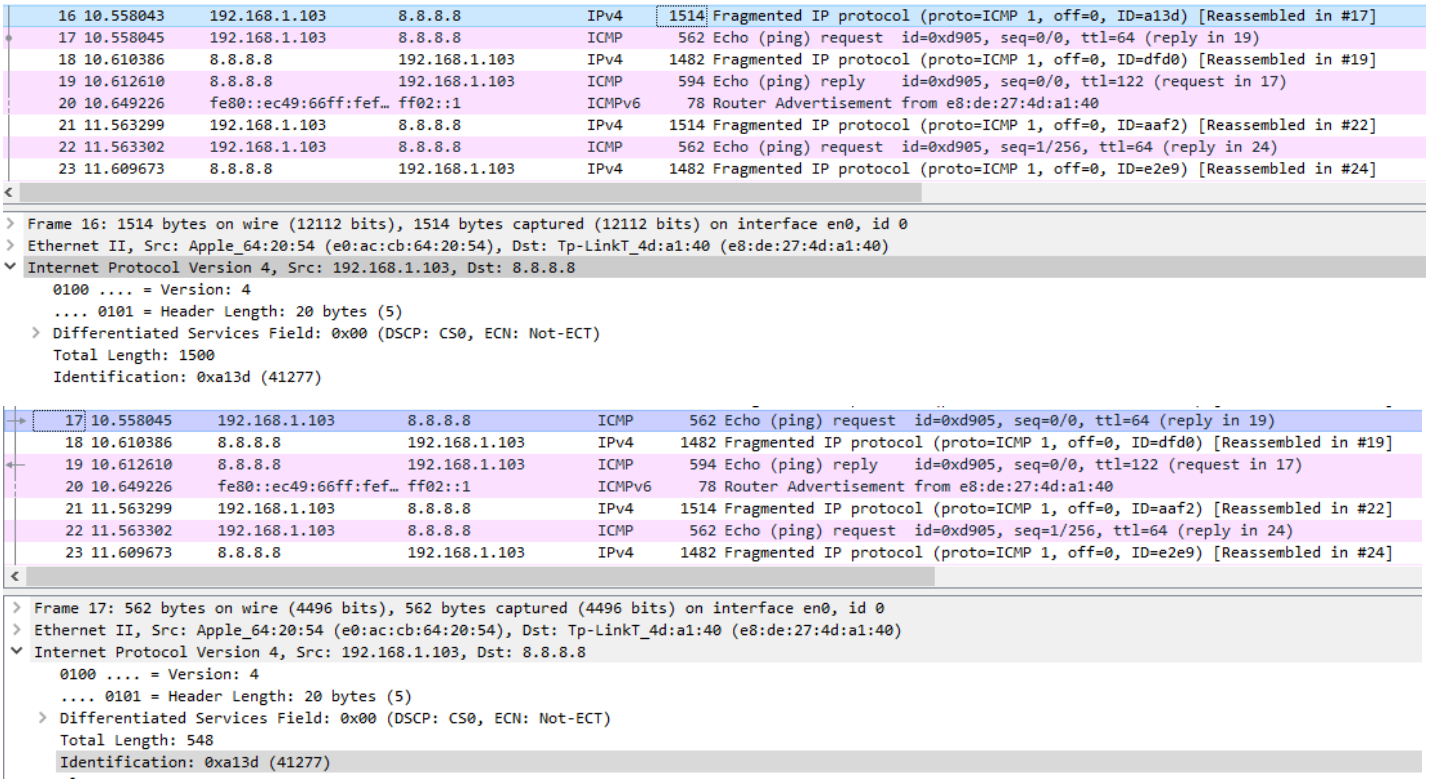
This is because tcp1 is competing with tcp2 on n2 to n4 link and competing with tcp4 on n1 to n2 link while tcp2 is only competing with tcp1 on n2 to n4 link. In this way, tcp1 have two competitor at same time while tcp2 have one. We ignore the competing happen in 10Mbps link since it have higher bandwidth therefore competing on this link won't affect the throughput.

Question 2: Why the throughput for flow tcp1 is fluctuating between time span 0.5 sec to 2 sec?
Between 0.5 sec to 2 sec, tcp1 is at slow start phase which increase its window size exponentially and is not stable. Tcp1 may meet lost event during slow start phase and have its window size reset to 1.

Question 3: Why is the maximum throughput achieved by any one flow capped at around 1.5Mbps?
This is because there are at least two TCP flow competing same link, and TCP is fair which means their throughput will fluctuating around 1.25Mbps. Therefore the maximum throughput achieved will be capped at around 1.5Mbps.

Exercise 2: Understanding IP Fragmentation

Question 1: Which data size has caused fragmentation and why? Which host/router has fragmented the original datagram? How many fragments have been created when data size is specified as 2000?
Since the MTU is 1500 bytes, the 2000 bytes data size ping and 3000 bytes data size ping caused fragmentation which is step 2 and step 3.
Host has fragmented the original datagram, the application layer on the host will check if the packet is larger than MTU and the host will help fragment the datagram.



Two fragments have been created when data size is specified as 2000.

Question 2: Did the reply from the destination 8.8.8.8. for 3500-byte data size also get fragmented? Why and why not?

```
39 19.395869    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
40 19.395870    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
41 19.395871    192.168.1.103    8.8.8.8          ICMP     582 Echo (ping) request  id
42 19.459151    8.8.8.8          192.168.1.103    IPv4    1482 Fragmented IP protocol
43 19.460862    8.8.8.8          192.168.1.103    IPv4    1482 Fragmented IP protocol
44 19.460869    8.8.8.8          192.168.1.103    ICMP     646 Echo (ping) reply    id
```

The reply gets fragmented cause its size also greater than the MTU.

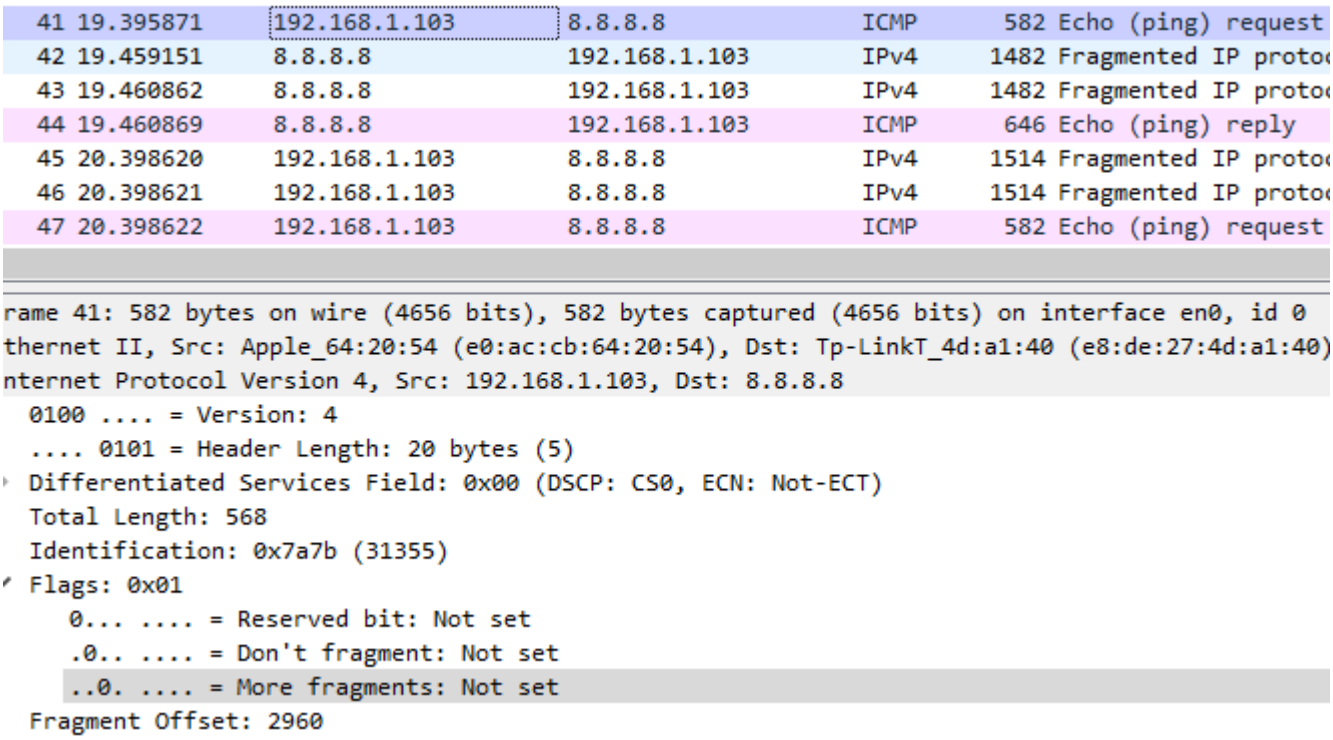
Question 3: Give the ID, length, flag and offset values for all the fragments of the first packet sent by 192.168.1.103 with data size of 3500 bytes?

```
39 19.395869    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
40 19.395870    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
41 19.395871    192.168.1.103    8.8.8.8          ICMP     582 Echo (ping) request  id
42 19.459151    8.8.8.8          192.168.1.103    IPv4    1482 Fragmented IP protocol
43 19.460862    8.8.8.8          192.168.1.103    IPv4    1482 Fragmented IP protocol
44 19.460869    8.8.8.8          192.168.1.103    ICMP     646 Echo (ping) reply    id
45 20.398620    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
46 20.398621    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
47 20.398622    192.168.1.103    8.8.8.8          ICMP     582 Echo (ping) request  id
```

```
> Frame 39: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface en0, id 0
> Ethernet II, Src: Apple_64:20:54 (e0:ac:cb:64:20:54), Dst: Tp-LinkT_4d:a1:40 (e8:de:27:4d:a1:40)
v Internet Protocol Version 4, Src: 192.168.1.103, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0x7a7b (31355)
  v Flags: 0x20, More fragments
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..1. .... = More fragments: Set
    Fragment Offset: 0
```

```
40 19.395870    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
41 19.395871    192.168.1.103    8.8.8.8          ICMP     582 Echo (ping) request  id
42 19.459151    8.8.8.8          192.168.1.103    IPv4    1482 Fragmented IP protocol
43 19.460862    8.8.8.8          192.168.1.103    IPv4    1482 Fragmented IP protocol
44 19.460869    8.8.8.8          192.168.1.103    ICMP     646 Echo (ping) reply    id
45 20.398620    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
46 20.398621    192.168.1.103    8.8.8.8          IPv4    1514 Fragmented IP protocol
47 20.398622    192.168.1.103    8.8.8.8          ICMP     582 Echo (ping) request  id
```

```
> Frame 40: 1514 bytes on wire (12112 bits), 1514 bytes captured (12112 bits) on interface en0, id 0
> Ethernet II, Src: Apple_64:20:54 (e0:ac:cb:64:20:54), Dst: Tp-LinkT_4d:a1:40 (e8:de:27:4d:a1:40)
v Internet Protocol Version 4, Src: 192.168.1.103, Dst: 8.8.8.8
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  > Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 1500
    Identification: 0x7a7b (31355)
  v Flags: 0x20, More fragments
      0... .... = Reserved bit: Not set
      .0.. .... = Don't fragment: Not set
      ..1. .... = More fragments: Set
    Fragment Offset: 1480
```



For first fragments, id is 31355, length is 1500, more fragments flag is 1, other flag is 0 and the offset value is 0.
For second fragments, id is 31355, length is 1500, more fragments flag is 1, other flag is 0 and the offset value is 1480.
For third fragments, id is 31355, length is 568, more fragments flag is 0, other flag is 0 and the offset value is 2960.

Question 4: Has fragmentation of fragments occurred when data of size 3500 bytes has been used? Why and why not?
All the 3500 bytes datagrams arrived in three fragments, therefore there is no fragmentation of fragments occurred.

Question 5: What will happen if for our example one fragment of the original datagram from 192.168.1.103 is lost?
The sender will retransmit all of the fragments if any of them is lost.

Exercise 3: Understanding the Impact of Network Dynamics on Routing

Question 1: Which nodes communicate with which other nodes? Which route do the packets follow? Does it change over time?
Node 0 is communicating with node 5 and node 2 is communicating with node 5. The route for node 0 is 0->1->4->5. The route for node 2 is 2->3->5. And this doesn't change over time.

Question 2: What happens at time 1.0 and at time 1.2? Does the route between the communicating nodes change as a result of that?
Between time 1.0 and 1.2, the link between node 1 and node 4 is down. The route is cut off at this time. All packets arrive at node 1 get dropped since the link is down.

Question 3: Did you observe any additional traffic as compared to Step 3 above? How does the network react to the changes that take place at time 1.0 and time 1.2 now?
There is more small packet size communication between all the nodes at the beginning and the time link is down. When the link is down, the nodes communication between each other and find new valid path to the destination which is route 0->1->2->3->5.

Question 4: How does this change affect the routing? Explain why.
The network avoids use the link between node 1 and node 4, and user route 0->1->2->3->5. The command "$ns

cost $n1 $n4 3" set the cost of the link between node 1 and node 4. Compare to other route, which have less cost, the network after use Distance-Vector routing protocol avoids use link 1->4.

Question 5: Describe what happens and deduce the effect of the line you just uncommented.
From node 0 to node 5, there is a least cost route 0->1->4->5. But form node 2 to node 5, both 2->3->5 and 2->1->4->5 has same cost after making changes. Therefore it chose to use both routes when "Node set multiPath_ 1" is uncommented. This line of code means allow the node use all of the different routes to the destination simultaneously