Resources / Lab Exercises (/COMP3331/21T1/resources/57481) / Lab Exercise 6: Solutions

Lab Exercise 6: Solutions

Exercise 1: Setting up NS2 simulation (5 marks)

Link to solution for NS2 tcl file (https://webcms3.cse.unsw.edu.au/COMP3331/21T1/resources/57941) and throughput script (https://webcms3.cse.unsw.edu.au/COMP3331/21T1/resources/57942) . **(3.5 marks)**

The throughput behaviour observed in the throughput plot is shown here (https://webcms3.cse.unsw.edu.au/COMP3331/21T1/resources/57507) .

Q1: Why the throughput achieved by flow tcp2 is higher than tcp1 between time span 6 sec to 8 sec? **(0.5 mark)**

Answer: Flow tcp1 is competing with flow tcp4 (FTP4) on link n1 - n2. It is again competing with tcp1 on link n2-n4. Flow tcp2 has less RTT than RTT for tcp1 thus TCP2 gets higher share of the bandwidth on link n2-n4 and eventually higher throughput is recorded at n5.

Q2: Why the throughput for flow tcp1 is fluctuating between time span 0.5 sec to 2 sec? **(0.5 mark)**

Answer: Flow tcp1 is probing for available bandwidth using Slow Start mechanism.

Q3: Why is the maximum throughput achieved by any one flow capped at around 1.5Mbps? **(0.5 mark)**

Answer: We have not permitted sole use of the Bandwidth for any one flow along its path to the destination. During time 0.5 - 2.0 sec, tcp1 is the only flow active but it is still in the SS phase and have to compete with other flows starting at 2.0 sec before it can discover the maximum bandwidth (2.5Mbps).

Exercise 2: Understanding IP Fragmentation (2.5 Marks)

Question 1: **(0.5 Marks)**

MTU for Ethernet is 1500 bytes. Ping with default values produces 84 bytes IP datagram (48 bytes ICMP payload + 8 bytes ICMP time stamp + 8 bytes ICMP header + 20 bytes IP header). In our example both 2000 and 3500 bytes payload for ICMP would cause fragmentation. The initiating host would cause the fragmentation. For data size of 2000, IP gets a total payload of 2008 bytes (1992 bytes data + 8 bytes ICMP time stamp + 8 bytes ICMP header). It creates two fragments (No 16 and 17 in the capture file), of total length 1500 bytes and 548 bytes. Note that you have to remove the filter "ICMP" in order to see packet No 16.

Question 2: **(0.5 Marks)**

Yes. The reply also get fragmented in three IPv4 datagrams. We cannot be sure who fragmented the original 3500 bytes reply but as it has to travel the last link to our sender, it is bound to be fragmented as the MTU for the last mile link is 1500 bytes.

Question 3: **(0.5 Marks)**

Packet 39: ID=0x7a7b, length=1500, flag=0x01(MF bit on), offset=0

Packet 40: ID=0x7a7b, length=1500, flag=0x01(MF bit on), offset=1480

Packet 41: ID=0x7a7b, length=568, flag=0, offset=2960

Note that Wireshark reports the fragment offsets of data in Bytes while the actual frag offset field in IP header contains different values (1480/8=185)

Question 4: **(0.5 Marks)**

We cannot be sure if the fragmentation of fragments has happened for our echo request from 192.168.1.103 towards 8.8.8.8 as we are only capturing packets at 192.168.1.103. No fragmentation of fragments has occurred for the reply from 8.8.8.8 for data size of 3500. Recall that reassembly is only done at the destination. As 192.168.1.103 is the destination for the reply and we have received only three fragments, fragmentation of fragments has not be done.

Question 5: **(0.5 Marks)**

If we are using a reliable transport protocol such as TCP, the sender has to eventually retransmit all the fragments again. In our example, the fragments would be incomplete and we will not see any reply as the receiver would discard it.

Exercise 3: Understanding the Impact of Network Dynamics on Routing (2.5 marks)

Question 1. **(0.5 Marks)**

Node 0 is sending packets to node 5 and Node 2 is transmitting to Node 5. The transmitted packets follow the route 0-1-4-5 & 2-3-5 respectively and the routes does not change over time.

Question 2. **(0.5 Marks)**

At time t=1.0, link 1-4 goes down, but the route between node 0 and node 5 does not change. As a result node 0 is unable to reach node 5.

At time t=1.2, link 1-4 goes up again, and the packets that were waiting at node 1 are now forwarded to node 4 and, eventually node 5.

Traffic between Node 2 and 5 remains unaffected.

Question 3. **(0.5 Marks)**

We observe additional control traffic whereby DV protocol is exchanging information to build its forwarding table. Now, when link 1-4 goes down, the DV routing protocol discovers a different route (0-1-2-3-5) and uses it. Once link 1- 4 becomes available again, the routing protocol reverts to the original route (0-1-4-5), since it has a lower cost (in terms of number of hops to destination).

Question 4. **(0.5 Marks)**

This changes the cost of link 1-4 to 3 (the cost of the rest of the links is assumed to be 1, implicitly). We can notice that the flow between Nodes 0 and 5 now uses route 0-1-2-3-5 (total cost=4), because the cost of this route becomes lower that the cost of 0-1-4-5 (total cost=5).

Question 5. **(0.5 Marks)**

Consider the route between Nodes 0-5 and 2-5 in this scenario. Total cost between Nodes 0-5 is 5 through route 0-1-4-5 and the total cost is 6 through route 0-1-2-3-5. Flow follows the total least cost path. However, for traffic between 2-5 we have two routes of equal total cost 2-3-5 and 2-1-4-5. Both routes now have equal cost to the destination. Since the network is now using multipath routing, node 2 will split traffic equally on both these paths.

Resource created 3 months ago (Monday 08 February 2021, 02:49:53 PM), last modified 13 days ago (Thursday 29 April 2021, 09:36:04 AM).

Comments

🔍 (/COMP3331/21T1/forums/search?forum_choice=resource/57940)

(/COMP3331/21T1/forums/resource/57940)

Add a comment

There are no comments yet.