

Resources / Lab Exercises (/COMP3331/18s2/resources/17340)

/ Lab Exercise 6: Routing, Throughput and IP Fragmentation

Lab Exercise 6: Routing, Throughput and IP Fragmentation

There are 7 labs during this course. For each student, the 5 best performing labs will contribute to your final lab mark.

Objectives:

- gain insights into routing dynamics and IP fragmentation
- Set up simulation in NS2 for TCP throughput measurement

Prerequisites and Links:

- Week 7 and 8 Lectures
- Relevant Parts of Chapter 4 and Chapter 5 of the textbook
- Introduction to Tools of the Trade
(<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/17358/edit>)
- Basic understanding of Linux. A good resource is here (<http://www.ee.surrey.ac.uk/Teaching/Unix/>) but there are several other resources online.
- Introduction to ns-2 from Labs 4 and 5.
- tp_routing.tcl (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/17326>)
- exercise2.tcl (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/20938>)
- TCPThroughput.png (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/20939>)
- ip_frag (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/19433>)

Marks: 10 marks.

- This lab requires submission (5 Marks) as well as demonstration (5 Marks)
- This lab comprises of a number of exercises. You have to submit a report containing answers to questions in all exercises.
- We expect the students to go through as much of the lab exercises as they can at home and come to the lab for clarifying any doubts in procedure/specifications.
- Exercise 2 marked with (*) is to be demonstrated in the lab to the tutor for marking purpose.
- Please attend your allocated lab and show/explain the answers of the marked exercises to your tutor.
- Please make sure you **sign the marking form** once the tutor marked your lab. Signing this form implies that you agreed on the mark you received.

Deadline:

Demonstration: You need to attend the lab for demonstration on your allocated lab slot. Please note that there would not be any retake available if you miss your lab slot. This means you will be awarded ZERO for the demonstration component of the lab (5 marks).

Report: *Midnight Friday 21st September 2018* . You can submit as many times as you wish before the deadline. A later submission will override the earlier submission, so make sure you submit the correct file. Do not leave until the last moment to submit, as there may be technical or communications error and you will not have time to rectify it.

Late Report Submission Penalty:

Late penalty will be applied as follows:

- 1 day after deadline: 20% reduction
- 2 days after deadline: 40% reduction
- 3 days after deadline: 60% reduction
- 4 or more days late: NOT accepted

Note that the above penalty is applied to your final mark in report. For example, if you submit your lab work report 2 days late and your score on the lab report is 4, then your final mark will be $4 - 1.6$ (40% penalty) = 2.4.

Submission Instructions:

Submit a PDF document **Lab6.pdf** with answers to all questions for all exercises. To include all supporting files, create a tar archive of all the files called **Lab6.tar**. Submit the archive using give. You can submit from a lab machine or ssh into the CSE login server.

Original Work Only:

You are strongly encouraged to discuss the questions with other students in your lab. However, each student must submit his or her own work. You may need to refer to the material indicated above (particularly Tools of the Trade document) and also conduct your own research to answer the questions.

OS Compatibility:

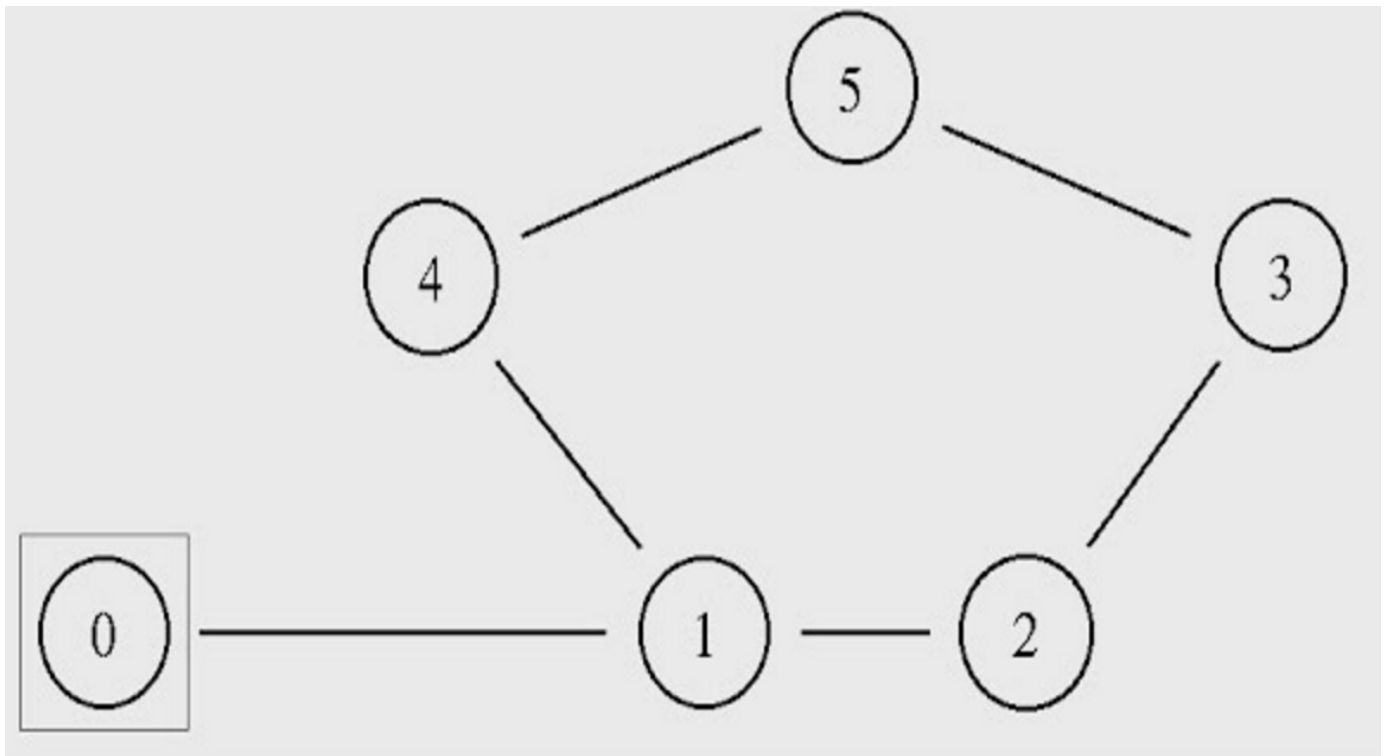
<o:p></o:p> The provided script (for ns-2) have been tested on CSE Linux machines. They may not work on your personal machine even if you have installed ns-2. As such, we suggest that you work on a CSE machine to complete these lab exercises. You can do so by going to a lab in person or via ssh/vlab.

Exercise 1: Understanding the Impact of Network Dynamics on Routing

(include in your report)

In this exercise, we will observe how routing protocols react when network conditions change (e.g., a network link fails) using a ns-2 simulation.

The provided script, tp_routing.tcl (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/17326>) takes no arguments and generates the network topology shown in the figure below.



You can run the simulation with the following command:

```
$ns tp_routing.tcl
```

Step 1: Run the script and observe the NAM window output.

Question 1. Which nodes communicate with which other nodes? Which route do the packets follow? Does it change over time?

Note: You can also answer the above question by examining the simulation setting in the script file.

Step 2: Modify the script by uncommenting the following two lines (line No 84 and 85):

```
$ns rtmodel-at 1.0 down $n1 $n4
$ns rtmodel-at 1.2 up $n1 $n4
```

Step 3: Rerun the simulation and observe the NAM window output.

NOTE: Ignore the NAM syntax warnings on the terminal. These will not affect the simulation.

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?

Step 4: The nodes in the simulation above use a static routing protocol (i.e., preferred routes do not change over time). We are going to change that, so that they use a Distance-Vector routing protocol. Modify the script and uncomment the following line (Line No 16) before the definition of the `finish` procedure.

```
$ns rtproto DV
```

Step 5: Rerun the simulation and observe the NAM window output.

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?

Step 6: Comment the two lines (Lines 84 and 85) that you had added to the script in Step 2 and uncomment the following line (Line 87) instead:

```
$ns cost $n1 $n4 3
```

Step 7: Rerun the simulation and observe the NAM window output.

Question 4: How does this change affect the routing? Explain why.

Step 8: Comment line 87 and Uncomment the following lines (Lines 89 and 90):

```
$ns cost $n1 $n4 2
$ns cost $n3 $n5 3
```

and uncomment the following (Line 29), which is located right after the finish procedure definition:

```
Node set multiPath_ 1
```

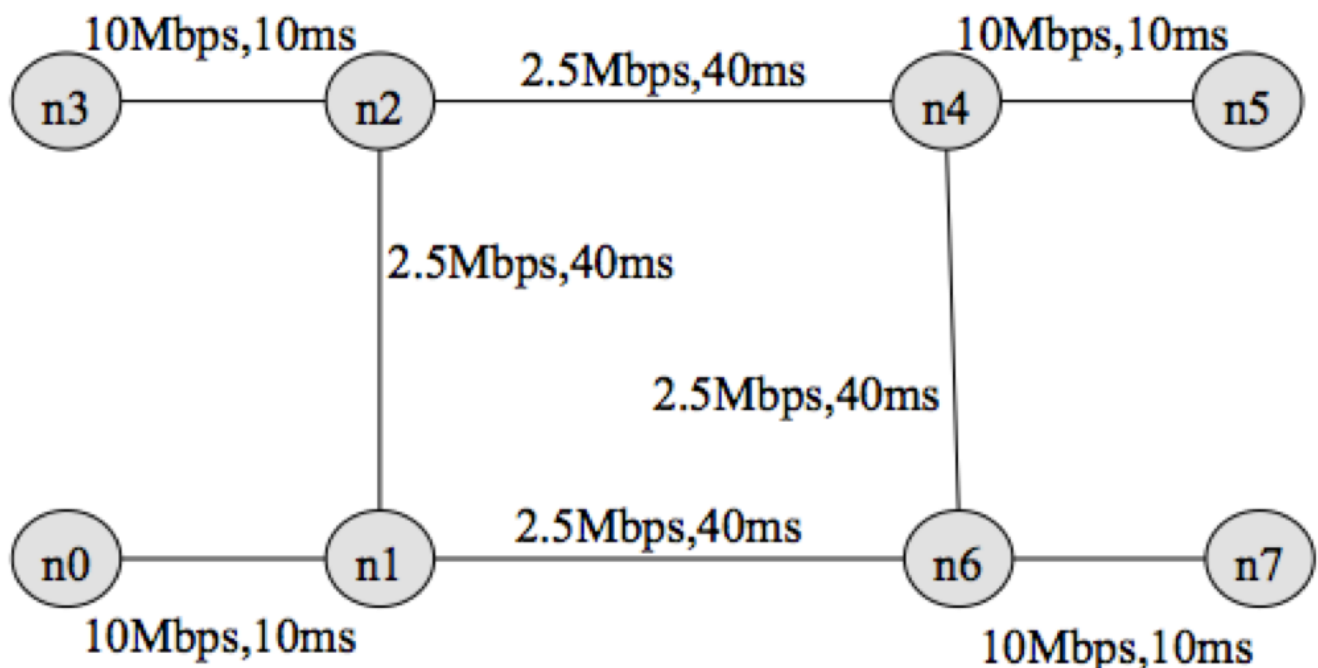
Step 9: Rerun the simulation and observe the NAM window output.

Question 5: Describe what happens and deduce the effect of the line you just uncommented.

(*) Exercise 2: Setting up NS2 simulation for measuring TCP throughput

(Include in your report and demonstrate to your tutor)

Consider the topology shown in the following figure where bandwidth and delay for each link is shown.



You have been provided with a stub tcl file exercise2.tcl

(<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/20938>) . Your task is to complete the stub file so that it runs with ns and produces two trace files tcp1.tr and tcp2.tr and nam.out. Check the animation for the simulation using nam.out file. Next write a script named "throughput.plot" (referenced from within exercise2.tcl in procedure finish()) to plot the throughput received by host n5 for two flows terminating at n5. Uncomment the line (#exec gnuplot throughput.plot &) to execute gnuplot. You have been provided with the throughput plot TCPThroughput.png (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/20939>) produced by gnuplot for comparing your final output.

">>" in the stub file indicates that one (or more) lines need to be added. Remove the ">>" and insert the required code.

Consider the following traffic pattern for your simulation.

FTP/TCP Source n0 -> TCP Sink n5 : start time: 0.5 sec End time: 8.5 sec

FTP/TCP Source n3 -> TCP Sink n5 : start time: 2.0 sec End time: 9.5 sec

FTP/TCP Source n7 -> TCP Sink n0 : start time: 3.0 sec End time: 9.5 sec

FTP/TCP Source n7 -> TCP Sink n3 : start time: 4.0 sec End time: 7.0 sec

You have to submit your completed tcl file (exercise2.tcl) and the script (throughput.plot) for producing the throughput plot. You should be prepared to comment on the throughput behaviour observed in the simulation.

Exercise 3: Understanding IP Fragmentation

(Include in your report)

We will try to find out what happens when IP fragments a datagram by increasing the size of a datagram until fragmentation occurs. You are provided with a Wireshark trace file ip_frag (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/19433>) that contains trace of sending pings with specific payloads to 8.8.8.8. We have used ping with option (-s option on Linux) to set the size of data to be carried in the ICMP echo request message. Note that the default packet size is 64 bytes in Linux (56 bytes data + 8 bytes ICMP header). Also note that Linux implementation for ping also uses 8 bytes of ICMP time stamp option leaving 48 bytes for the user data in the default mode. Once you have send a series of packets with the increasing data sizes, IP will start fragmenting packets that it cannot handle. We have used the following commands to generate this trace file.

Step 1: Ping with default packet size to the target destination as 8.8.8.8

```
ping -c 10 8.8.8.8
```

Step 2: Repeat by sending a set of ICMP requests with data of 2000.

```
ping -s 2000 -c 10 8.8.8.8
```

Step 3: Repeat again with data size set as 3500

```
ping -s 3500 -c 10 8.8.8.8
```

Load this trace file in Wireshark, filter on protocol field ICMP (you may need to clear the filter to see the fragments) and answer the following questions.

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?

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

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?

Question 4: Has fragmentation of fragments occurred when data of size 3500 bytes has been used? Why and why not?

Question 5: What will happen if for our example one fragment of the original datagram from 192.168.1.103 is lost?

Resource created 4 months ago (Monday 16 July 2018, 02:50:39 PM), last modified 2 months ago (Wednesday 19 September 2018, 12:36:55 PM).

Comments

☐ ☐ (/COMP3331/18s2/forums/search?forum_choice=resource/17348) ☐ (/COMP3331/18s2/forums/resource/17348)

Add a comment



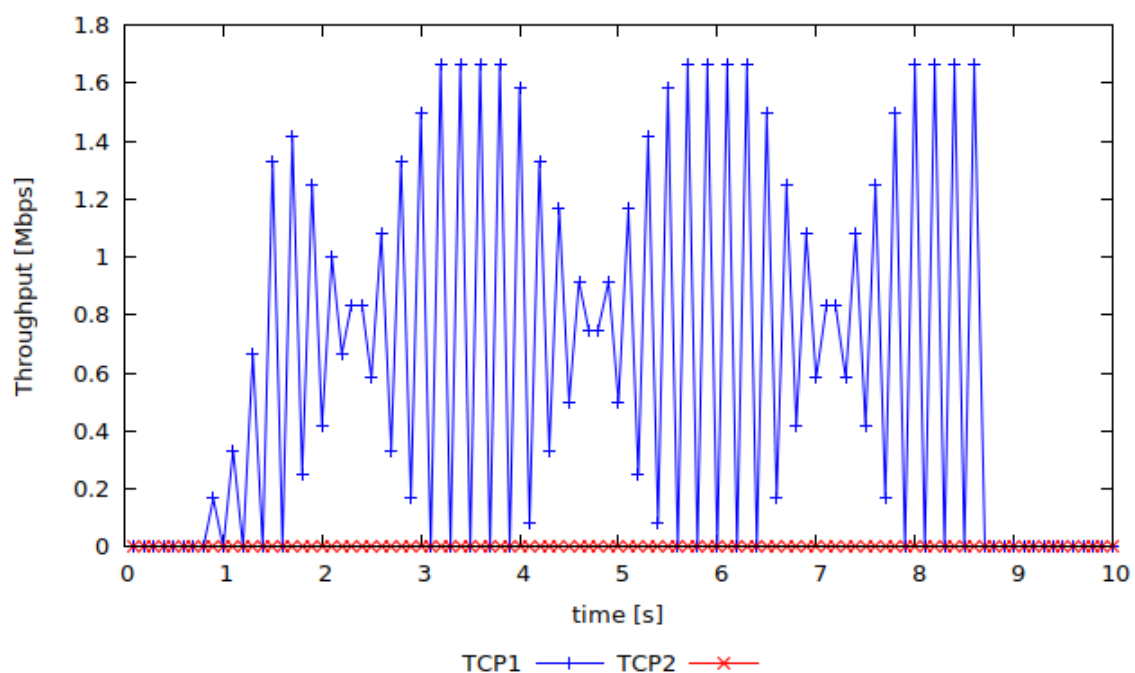
Jack Jiang (/users/z5129432) 2 months ago (Wed Sep 19 2018 19:00:36 GMT+1000 (澳大利亚东部标准时间))

Hello,

I have a question in related to Exercise 2.

I only activated ftp1 by comment out the other lines and got the following graph.

```
# start FTP sessions
$ns at 0.5 "$ftp1 start"
# $ns at 2.0 "$ftp2 start"
# $ns at 3.0 "$ftp3 start"
# $ns at 4.0 "$ftp4 start"
```

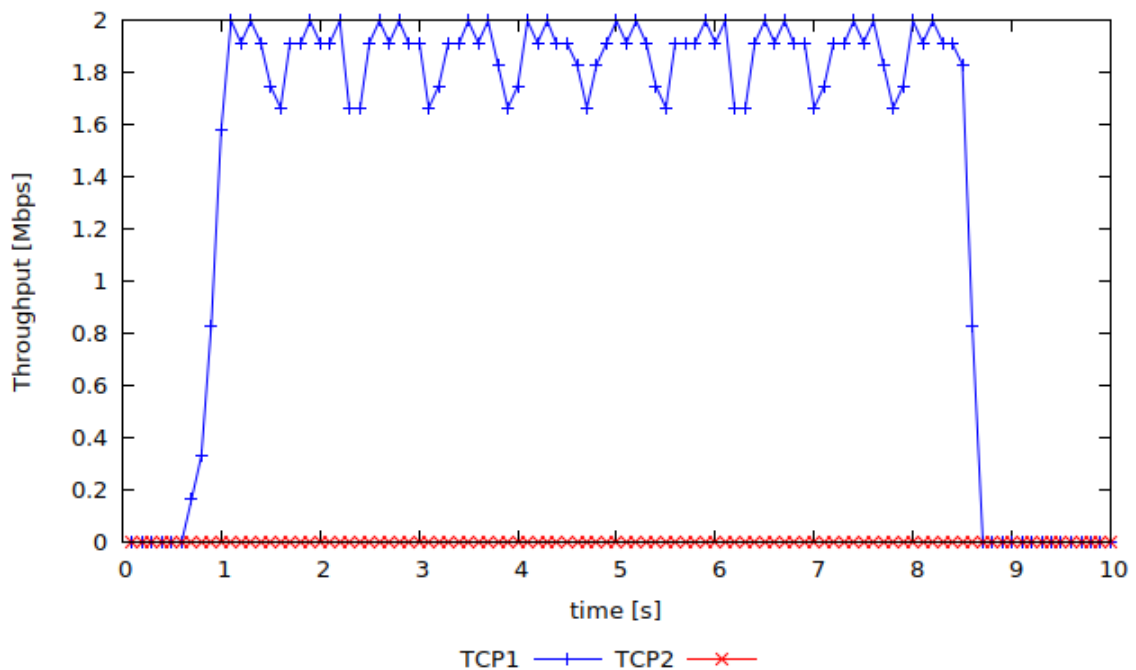


I thought the bottleneck is between n1-n2 or n2-n4, so I expect the throughput will rise to about 2.5Mbps.

However, I have observed that the maximum speed is only 1.6Mbps.

Then I change the delay between n1-n2 and n2-n4 to 10ms and got the following graph.

```
$ns duplex-link $n1 $n2 2.5Mb 10ms DropTail
$ns duplex-link $n2 $n4 2.5Mb 10ms DropTail
```



I find that the throughput increases to about 2Mbps.

It seems that the throughput is bounded by delays, but I don't know why. Can someone explain how does the delay influences the throughput?

Reply



Jie Shang (/users/z5153884) 2 months ago (Wed Sep 19 2018 15:03:41 GMT+1000 (澳大利亚东部标准时间))

Hi Ali,

In exercise 3, I can't get reply when I ping 2000 and 3500, all the packet will get loss. I think the 2000 and 3500 data size can fragmentate into small packets to transfer right? (I used the lab computer)

Reply



Daniel Ho (/users/z3466312) 2 months ago (Wed Sep 19 2018 17:05:07 GMT+1000 (澳大利亚东部标准时间))

There is no need to ping, load the trace file provided (ip_frag) into Wireshark and inspect the packets.

Reply



Jie Shang (/users/z5153884) 2 months ago (Wed Sep 19 2018 19:15:55 GMT+1000 (澳大利亚东部标准时间))

I see, thank you so much!

Reply



Ali Dorri (/users/z5095883) 2 months ago (Wed Sep 19 2018 15:49:43 GMT+1000 (澳大利亚东部标准时间)), last modified 2 months ago (Wed Sep 19 2018 15:49:51 GMT+1000 (澳大利亚东部标准时间))

Do you mean that you run the command to ping with 2000 and 3500? If yes, you do not need to ping anything. The provided wireshark trace file contains the data of different pings. Actually, we just showed you what commands were used to generate the tracefile.

Reply



Jie Shang (/users/z5153884) 2 months ago (Wed Sep 19 2018 19:16:21 GMT+1000 (澳大利亚东部标准时间))

Ok, thank you so much!

Reply



Rafael Barreto Sotomayor (/users/z5192279) 2 months ago (Wed Sep 19 2018 13:41:14 GMT+1000 (澳大利亚东部标准时间))

Hi all - Can someone shed some light on how we can visualise the host/router that fragmented the original datagram using wireshark? (Exercise 3. Question 1)

Thanks...

Reply



Ali Dorri (/users/z5095883) 2 months ago (Wed Sep 19 2018 14:36:15 GMT+1000 (澳大利亚东部标准时间))

Imagine the flow of the packets and see when it is possibly fragmented.

Reply



Rafael Barreto Sotomayor (/users/z5192279) 2 months ago (Wed Sep 19 2018 14:46:39 GMT+1000 (澳大利亚东部标准时间))

I can imagine this, but do I have to provide an IP address of the router that fragmented the datagram? Is it perhaps one of the IPv6 addresses shown on wireshark?

Reply



Ali Dorri (/users/z5095883) 2 months ago (Wed Sep 19 2018 14:52:55 GMT+1000 (澳大利亚东部标准时间))

You do not need to provide any IP addresses, just simply mention whether it is the initializing host or the receiver host that fragments.

Reply



Rafael Barreto Sotomayor (/users/z5192279) 2 months ago (Wed Sep 19 2018 15:03:05 GMT+1000 (澳大利亚东部标准时间))

I see it now... Thanks!

Reply



Peiran Liu (/users/z5139158) 2 months ago (Wed Sep 19 2018 11:54:46 GMT+1000 (澳大利亚东部标准时间))

Hi, I've finished exercise2.tcl, but when I run "ns exercise2.tcl", and it return "using backward compatibility mode".

Can someone tell me how to solve this problem. Thanks!

Reply



Ali Dorri (/users/z5095883) 2 months ago (Wed Sep 19 2018 12:21:56 GMT+1000 (澳大利亚东部标准时间))

Are you running the code on CSE machines? Is this an error that stops your program or just a warning?

Reply



Peiran Liu (/users/z5139158) 2 months ago (Wed Sep 19 2018 12:32:31 GMT+1000 (澳大利亚东部标准时间))

On Vlab. It's a warning, and I cant get any result.

Reply



Ali Dorri (/users/z5095883) 2 months ago (Wed Sep 19 2018 12:38:25 GMT+1000 (澳大利亚东部标准时间)), last modified 2 months ago (Wed Sep 19 2018 12:38:33 GMT+1000 (澳大利亚东部标准时间))

if it is a warning then there shouldn't be any issue caused by that. Maybe other part of your code does not work properly. Does it create the .tr files? is there any other error happening?

Reply



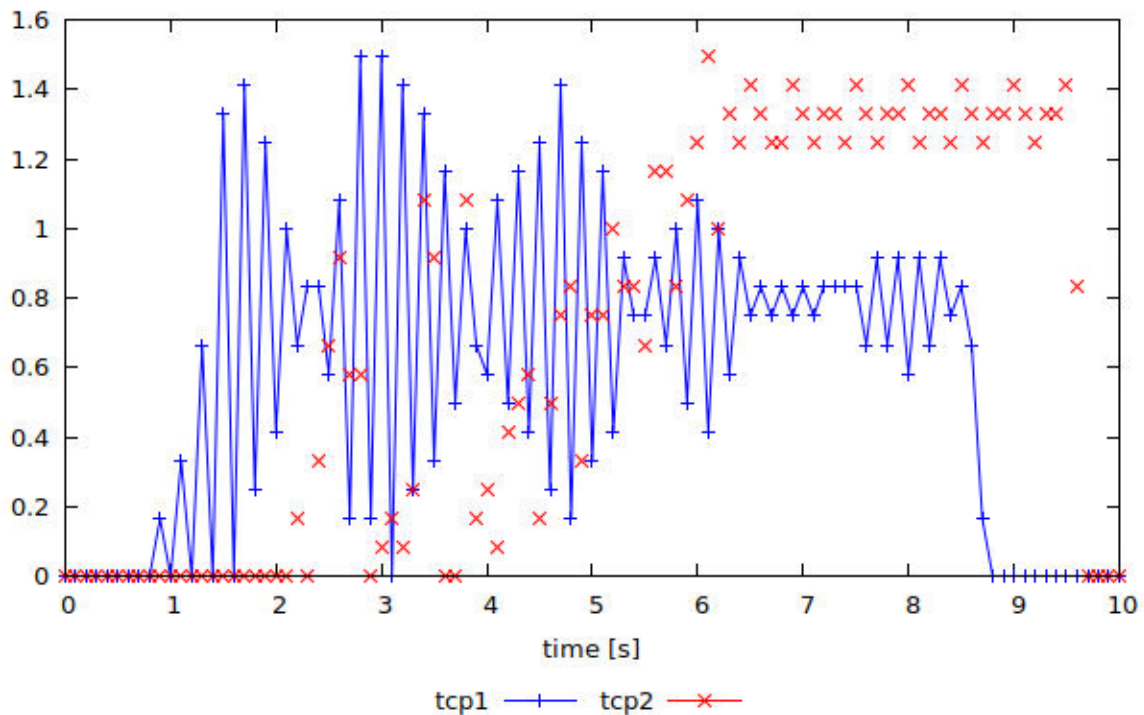
Peiran Liu (/users/z5139158) 2 months ago (Wed Sep 19 2018 12:44:24 GMT+1000 (澳大利亚东部标准时间))

Ok, I've solved this problem, there's some typo in .tcl file. Thanks for your help.

Reply



Zhenqi Wang (/users/z5141545) 2 months ago (Mon Sep 17 2018 21:06:45 GMT+1000 (澳大利亚东部标准时间))



Hi, I've finished exercise 2 but somehow my plot for tcp2 is not connected, and I'm using the same plot command for tcp1.

Can someone help explain what did I do wrong? Thanks!

Reply



Joshua Glover (/users/z5115903) 2 months ago (Tue Sep 18 2018 03:30:23 GMT+1000 (澳大利亚东部标准时间))

Hey Zhenqi,

I had the same problem so maybe we both made the same mistake. Your tcp2.tr file probably has extra lines of output that shouldn't be there, similar to the problem Adith had below. This causes gnuplot to not connect the dots as there is "junk" between the lines it cares about in the tcp2.tr. I fixed it by changing one of the lines relevant to tcp2.tr in my exercise.tcl file. Hope that helps

Reply



Zhenqi Wang (/users/z5141545) 2 months ago (Tue Sep 18 2018 12:02:21 GMT+1000 (澳大利亚东部标准时间))

I'll have a look, thanks!

Reply



Mohamed Al Mouiee (/users/z5114185) 2 months ago (Tue Sep 18 2018 16:02:53 GMT+1000 (澳大利亚东部标准时间))

It's most likely that you're namtrace-all or trace-all for that tcpX.tr file, make sure that your output files are tracing the correct data

Reply



Zhenqi Wang (/users/z5141545) 2 months ago (Tue Sep 18 2018 16:24:30 GMT+1000 (澳大利亚东部标准时间))

Yeah I've realized that and it's working now, thanks.

Reply



Zihao Cheng (/users/z5108506) 2 months ago (Wed Sep 19 2018 12:39:33 GMT+1000 (澳大利亚东部标准时间))

I am facing the same issues, both file 1 and 2 are using trace-all, but seems tcp2.tr output has more lines, could I ask how do u solve this ?
thanks

Reply



Zhenqi Wang (/users/z5141545) 2 months ago (Wed Sep 19 2018 12:46:59 GMT+1000 (澳大利亚东部标准时间))

You shouldn't need trace-all for tcp1 and 2

Reply



Kavitha Narayanan (/users/z5190588) 2 months ago (Mon Sep 17 2018 12:10:09 GMT+1000 (澳大利亚东部标准时间))

Hi,

This is regarding Question 2:

When I execute ns2 simulator, I see the following error;

```
z5190588@vx3:/tmp_amd/reed/export/reed/1/z5190588/Documents/9331/Lab6_1$ ns  
exercise2.tcl
```

```
ns: record: can't read "sink1": no such variable
```

```
while executing
```

```
"$sink1 set bytes_"
```

```
(procedure "record" line 8)
```

```
invoked from within
```

```
"record"
```

I couldn't identify what is wrong.

I have declared sink1 as global inside the 'record' procedure.

Could someone let me know what could be the issue? Thanks.

Reply



Nadeem Ahmed (/users/z3003139) 2 months ago (Mon Sep 17 2018 13:04:06 GMT+1000 (澳大利亚东部标准时间))

Should not be an error if you have defined the sink1 to be global inside record and sink1 does exist in the ns2 main script. Maybe a typo somewhere?

Reply



Kavitha Narayanan (/users/z5190588) 2 months ago (Mon Sep 17 2018 13:38:33 GMT+1000 (澳大利亚东部标准时间))

Thank you so much for your immediate response. I have updated "record" proc to define the variables one in each line rather than listing all of them using comma in a single line and I don't see this error now.

I have another question;

Is tcp1.tr tracing n0 to n5 & n3 to n5 and tcp2.tr n7 to n0, n7 to n3?

Or tcp1.tr tracing n0 to n5 & tcp2.tr tracing n3 to n5?

Thanks for your help.

Reply



Ali Dorri (/users/z5095883) 2 months ago (Mon Sep 17 2018 14:38:46 GMT+1000 (澳大利亚东部标准时间))

tcp1.tr tracing n0 to n5 and tcp2.tr tracing n3 to n5

Reply



Kavitha Narayanan (/users/z5190588) 2 months ago (Mon Sep 17 2018 15:55:57 GMT+1000 (澳大利亚东部标准时间))

Thank you so much for your immediate response.

Reply



Wenxun Peng (/users/z5195349) 2 months ago (Mon Sep 17 2018 11:19:33 GMT+1000 (澳大利亚东部标准时间))

My tcp1.tr file is empty, but tcp2.tr isn't. Is there anything I've missed? And when I write the throughput.plot, I don't know how to start. Is there any information I can refer to ?



```

exercise2.tcl x tcp2.tr x tcp1.tr x
+ 0.5 0 1 tcp 40 ----- 1 0.0 5.0 0 0
- 0.5 0 1 tcp 40 ----- 1 0.0 5.0 0 0
r 0.510032 0 1 tcp 40 ----- 1 0.0 5.0 0 0
+ 0.510032 1 2 tcp 40 ----- 1 0.0 5.0 0 0
- 0.510032 1 2 tcp 40 ----- 1 0.0 5.0 0 0
r 0.55016 1 2 tcp 40 ----- 1 0.0 5.0 0 0
+ 0.55016 2 4 tcp 40 ----- 1 0.0 5.0 0 0
- 0.55016 2 4 tcp 40 ----- 1 0.0 5.0 0 0
r 0.590288 2 4 tcp 40 ----- 1 0.0 5.0 0 0
+ 0.590288 4 5 tcp 40 ----- 1 0.0 5.0 0 0
- 0.590288 4 5 tcp 40 ----- 1 0.0 5.0 0 0
r 0.60032 4 5 tcp 40 ----- 1 0.0 5.0 0 0
+ 0.60032 5 4 ack 40 ----- 1 5.0 0.0 0 1
- 0.60032 5 4 ack 40 ----- 1 5.0 0.0 0 1
r 0.610352 5 4 ack 40 ----- 1 5.0 0.0 0 1

```

Reply



Nadeem Ahmed (/users/z3003139) 2 months ago (Mon Sep 17 2018 12:59:32 GMT+1000 (澳大利亚东部标准时间))

Your tcp2.tr does not contain the right output. This file should have what is being written from the procedure record ().

For throughput.plot look at example scripts given in Lab 5.

Reply



Adith Kumar Sukumar (/users/z5177910) 2 months ago (Sun Sep 16 2018 00:43:33 GMT+1000 (澳大利亚东部标准时间))

For my Q2: I am getting weird data when i am collecting data for TCP2.tr.

```

1 0 0.0
2 0.10000000000000001 0.0
3 0.20000000000000001 0.0
4 0.30000000000000004 0.0
5 0.40000000000000002 0.0
6 + 0.5 0 1 tcp 40 ----- 1 0.0 5.0 0 0
7 - 0.5 0 1 tcp 40 ----- 1 0.0 5.0 0 0
8 0.5 0.0
9 r 0.510032 0 1 tcp 40 ----- 1 0.0 5.0 0 0
10 + 0.510032 1 2 tcp 40 ----- 1 0.0 5.0 0 0
11 - 0.510032 1 2 tcp 40 ----- 1 0.0 5.0 0 0
12 r 0.55016 1 2 tcp 40 ----- 1 0.0 5.0 0 0
13 + 0.55016 2 4 tcp 40 ----- 1 0.0 5.0 0 0
14 - 0.55016 2 4 tcp 40 ----- 1 0.0 5.0 0 0
15 r 0.590288 2 4 tcp 40 ----- 1 0.0 5.0 0 0
16 + 0.590288 4 5 tcp 40 ----- 1 0.0 5.0 0 0
17 - 0.590288 4 5 tcp 40 ----- 1 0.0 5.0 0 0
18 0.59999999999999998 0.0
19 r 0.60032 4 5 tcp 40 ----- 1 0.0 5.0 0 0
20 + 0.60032 5 4 ack 40 ----- 1 5.0 0.0 0 1
21 - 0.60032 5 4 ack 40 ----- 1 5.0 0.0 0 1
22 r 0.610352 5 4 ack 40 ----- 1 5.0 0.0 0 1
23 + 0.610352 4 2 ack 40 ----- 1 5.0 0.0 0 1
24 - 0.610352 4 2 ack 40 ----- 1 5.0 0.0 0 1
25 r 0.65048 4 2 ack 40 ----- 1 5.0 0.0 0 1
26 + 0.65048 2 1 ack 40 ----- 1 5.0 0.0 0 1
27 - 0.65048 2 1 ack 40 ----- 1 5.0 0.0 0 1
28 r 0.690608 2 1 ack 40 ----- 1 5.0 0.0 0 1
29 + 0.690608 1 0 ack 40 ----- 1 5.0 0.0 0 1
30 - 0.690608 1 0 ack 40 ----- 1 5.0 0.0 0 1
31 0.69999999999999996 0.0
32 r 0.70064 1 0 ack 40 ----- 1 5.0 0.0 0 1
33 + 0.70064 0 1 tcp 1040 ----- 1 0.0 5.0 1 2
34 - 0.70064 0 1 tcp 1040 ----- 1 0.0 5.0 1 2
35 + 0.70064 0 1 tcp 1040 ----- 1 0.0 5.0 2 3
36 - 0.701472 0 1 tcp 1040 ----- 1 0.0 5.0 2 3

```

I couldn't figure out the reason for this wrong data and when i tried to plot, it plotted incorrectly. Could you give me reasons as to why the data is skewed.

Thanks.

Reply



Nadeem Ahmed (/users/z3003139) 2 months ago (Sun Sep 16 2018 08:34:12 GMT+1000 (澳大利亚东部标准时间))

Seems like you are writing multiple outputs to your TCP2.tr file. I can see throughput measurements as well as information in NS2 trace file format.

Reply



Zihao Cheng (/users/z5108506) 2 months ago (Wed Sep 19 2018 12:32:16 GMT+1000 (澳大利亚东部标准时间))

I am facing the same issue,

```
#Calculate the bandwidth (in MBit/s) and write it to the files
puts $f1 "$now [expr $bw1/$time*8/1000000]"
puts $f2 "$now [expr $bw2/$time*8/1000000]"
```

this is the only output to file, but why in file2 has so many NS2 trace file? Thanks

Reply



Nadeem Ahmed (/users/z3003139) 2 months ago (Wed Sep 19 2018 13:29:46 GMT+1000 (澳大利亚东部标准时间))

Using "trace-all" somewhere? If yes, change the name of file where this trace information is being written to anything different from file referred by \$f1 and \$f2.

Reply



Jialun Li (/users/z5172023) 2 months ago (Sun Sep 16 2018 01:27:01 GMT+1000 (澳大利亚东部标准时间))

For Q3 how does the three given Ping commands work with the Wireshark trace file ip_frag (<https://webcms3.cse.unsw.edu.au/COMP3331/18s2/resources/19433>), I mean if I just simply run those Ping commands in terminal it just give me like this.

```
z5172023@vx3:/tmp_and/kamen/export/kamen/4/z5172023/Desktop$ ping -c 10 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_req=1 ttl=120 time=1.54 ms
64 bytes from 8.8.8.8: icmp_req=2 ttl=120 time=1.58 ms
64 bytes from 8.8.8.8: icmp_req=3 ttl=120 time=1.59 ms
64 bytes from 8.8.8.8: icmp_req=4 ttl=120 time=1.54 ms
64 bytes from 8.8.8.8: icmp_req=5 ttl=120 time=1.51 ms
64 bytes from 8.8.8.8: icmp_req=6 ttl=120 time=1.55 ms
64 bytes from 8.8.8.8: icmp_req=7 ttl=120 time=1.58 ms
64 bytes from 8.8.8.8: icmp_req=8 ttl=120 time=1.54 ms
64 bytes from 8.8.8.8: icmp_req=9 ttl=120 time=1.52 ms
64 bytes from 8.8.8.8: icmp_req=10 ttl=120 time=1.54 ms

--- 8.8.8.8 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9014ms
rtt min/avg/max/mdev = 1.517/1.554/1.598/0.035 ms
z5172023@vx3:/tmp_and/kamen/export/kamen/4/z5172023/Desktop$ ping -s 2000 -c 10 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 2000(2028) bytes of data.

--- 8.8.8.8 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9210ms

z5172023@vx3:/tmp_and/kamen/export/kamen/4/z5172023/Desktop$ ping -s 3500 -c 10 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 3500(3528) bytes of data.

--- 8.8.8.8 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9212ms
```

Reply



Nadeem Ahmed (/users/z3003139) 2 months ago (Sun Sep 16 2018 08:37:04 GMT+1000 (澳大利亚东部标准时间))

You do not need to run the ping commands (Steps 1 to 3). These are simply illustrating how we captured the trace. Simply load the ip_frag trace in Wireshark and answer the questions.

Reply