

Lab 1

Objectives:

- Get familiar with the basic networking tools: ping, traceroute, ifconfig, netstat, nslookup
- Gain insights into evaluating network performance and understanding network topology

Prerequisites:

- Week 1 Lectures
- Relevant Parts of Chapter 1 of the textbook
- [Introduction to Tools of the Trade](#)
- [runping.sh](#)
- [plot.sh](#)
- Basic understanding of Linux. A good resource is [here](#) but there are several other resources online

Marks: 10 marks

- You can only attend your allocated lab slot. The tutors will not allow students of other labs to attend their lab slot, and they will ask you to leave the lab.
- This lab comprises of a number of exercises. Please note that not all the exercises for this lab are marked. However, you have to submit a report containing answers for all of the lab 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.

Deadline:

Midnight on the Sunday 3rd of March. 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 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. For example, if you submit your lab work 2 days late and your score on the lab is 8, then your final mark will be $8 - 3.2$ (40% penalty) = 4.8.

Submission Instructions:

Submit a PDF document **Lab1.pdf** with answers to all questions for all exercises. To include all supporting documents, create a tar archive of all the files called **Lab1.tar**. Submit the archive using give. You can submit from a lab machine or ssh into the CSE login server. Instructions to ssh into CSE login servers are [here](#) .

1. Type “tar -cvf Lab1.tar filenames” e.g. tar -cvf Lab1.tar Lab1.pdf output.txt
 2. When you are ready to submit, at the bash prompt type 3331
 3. Next, type: give cs3331 Lab1 Lab1.tar
- The submission system will also accept a PDF called Lab1.pdf. So, if you only have a single PDF file, you can submit the PDF directly (no need to tar).
 - Please make sure that the tar archive is not corrupted. You can untar (use tar -xvf Lab1.tar) the created archive to check that all the files are intact.
 - Max file size for submission is **3MB** .
 - Whether you are COMP3331 or COMP9331 use the above command to submit your lab.

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:

Please note that the instructions provided herein assume that you are running the exercises on a Linux machine (similar to the CSE lab machines). These commands (and the scripts provided) may not work as prescribed on other OSes (Windows, OS X, etc.). We strongly recommend that you run these experiments on CSE machines. If you are running from off-campus, you can ssh into a CSE server. We will be unable to help you diagnose any issues that may arise with OSes other than Linux.

Exercise 1: nslookup

Use the nslookup command from the "Tools of the Trade" and answer the following questions:

1. Which is the IP address of the Google site (www.google.com)? In your opinion, what is the reason of having several IP addresses as an output?
2. Find out name of the IP address 127.0.0.1. What is special about this IP address?

Exercise 2: Use ping to test host reachability

Are the following hosts reachable from your machine by using ping:

- www.cse.unsw.edu.au
- www.getfittest.com.au
- www.mit.edu
- www.intel.com.au
- www.tpg.com.au

- www.hola.hp
- www.amazon.com
- www.tsinghua.edu.cn
- www.kremlin.ru
- 8.8.8.8

If you observe that some hosts are not reachable, then can you explain why? Check if the addresses unreachable by the ping command are reachable from the Web browser.

Exercise 3: Use traceroute to understand network topology

Note: Include all traceroute outputs in your report.

1. Run traceroute on your machine to www.columbia.edu . How many routers are there between your workstation and www.columbia.edu ? How many routers along the path are part of the UNSW network? Between which two routers do packets cross the Pacific Ocean? Hint: compare the round trip times from your machine to the routers using ping.
2. Run traceroute from your machine to the following destinations:
(i) www.ucla.edu (ii) www.u-tokyo.ac.jp and (iii) www.lancaster.ac.uk . At which router do the paths from your machine to these three destinations diverge? Find out further details about this router. (HINT: You can find out more about a router by running the whois command: whois router-IP-address). Is the number of hops on each path proportional the physical distance? HINT: You can find out geographical location of a server using the following tool - <http://www.yougetsignal.com/tools/network-location/>
3. Several servers distributed around the world provide a web interface from which you can perform a traceroute to any other host in the Internet. Here are two examples:
(i) <http://www.speedtest.com.sg/tr.php> and (ii) <https://www.telstra.net/cgi-bin/trace> . Run traceroute from both these servers towards your machine and in the reverse direction (i.e. from your machine to these servers). You may also try other traceroute servers from the list at www.traceroute.org . What are the IP addresses of the two servers that you have chosen. Does the reverse path go through the same routers as the forward path? If you observe common routers between the forward and the reverse path, do you also observe the same IP addresses? Why or why not?

Exercise 4: Use ping to gain insights into network performance

Note: Include all graphs in your report.

We now use the ping utility to investigate network delay and its implications on network performance. In particular, we will analyze the dependency of packet size and delay.

There is a shell script, [runping.sh](#) , provided that you can use instead of running many pings with different packet sizes by hand. After downloading this script on your machine make sure you can execute it. If not, you will have to execute the following command in the command line: `chmod u+x runping.sh`. To run the ping traces you may use the `runping.sh` script as follows:
`./runping.sh www.abc.net`(or whatever other destination you want to ping). It will automatically run ping for different packet sizes and with 50 ping packets per size. Note, since a ping is sent once per second, this script will take a few minutes to finish. Basically, this script only executes the commands:

```
$ ping -s 22 -c 50 -i 1 www.abc.net > www.abc.net-p50
```

...

```
$ ping -s 1472 -c 50 -i 1 www.abc.net > www.abc.net-p1500
```

and writes the output of the pings to the corresponding files.

Use this script for the following destinations:

(i) www.uq.edu.au (ii) www.nus.edu.sg and (iii) www.tu-berlin.de

In other words, execute the following commands

```
$ ./runping.sh www.uq.edu.au
```

```
$ ./runping.sh www.nus.edu.sg
```

```
$ ./runping.sh www.tu-berlin.de
```

In case you notice one of the hosts above is not responsive, select the following alternate destinations:

(i) within Australia (www.uow.edu.au , www.flinders.edu.au , www.uws.edu.au) (ii) Asia
(www.tsinghua.edu.cn , www.sutd.edu.sg , www.iit.ac.in) (iii) Europe
(www.epfl.ch , www.aau.dk , www.uio.no)

Note that all delay values reported are in milliseconds (ms) and reflect the round trip time (RTT) between your host and the destinations.

When the runping.sh script is finished for all destinations, you can plot the results using another provided script, [plot.sh](#) , as follows:

```
$ ./plot.sh www.uq.edu.au-p*
```

```
$ ./plot.sh www.nus.edu.sg-p*
```

```
$ ./plot.sh www.tu-berlin.de-p*
```

If you cannot execute plot.sh, then fix the permissions by executing the following command in the command line:

```
$ chmod u+x plot.sh
```

The script plot.sh will produce the following files: destination_delay.pdf, destination_scatter.pdf, and destination_avg.txt for each of the destinations (e.g., for www.nus.edu.sg we have [www.nus.edu.sg_delay.pdf](#) , [www.nus.edu.sg_scatter.pdf](#) and [www.nus.edu.sg_avg.txt](#)).

The graph *destination_delay.pdf* shows how delay varies over time (different colours correspond to different packet sizes), and *destination_scatter.pdf* shows delay vs. packet size as a scatter plot. *destination_avg.txt* contains the average (2nd column) and minimum (3rd column) delay values corresponding to each packet size (1st column).

1. For each of these locations find the (approximate) physical distance from UNSW using Google Maps and compute the shortest possible time T for a packet to reach that location from

UNSW. You should assume that the packet moves (i.e. propagates) at the speed of light, 3×10^8 m/s. Note that the shortest possible time will simply be the distance divided by the propagation speed. Plot a graph where the x-axis represents the distance to each city (i.e. Brisbane, Singapore and Berlin), and the y-axis represents the ratio between the minimum delay (i.e. RTT) as measured by the ping program (select the values for 50 byte packets) and the shortest possible time T to reach that city from UNSW. (Note that the y-values are no smaller than 2 since it takes at least $2 \cdot T$ time for any packet to reach the destination from UNSW and get back). Can you think of at least two reasons why the y-axis values that you plot are greater than 2?

2. Is the delay to the destinations constant or does it vary over time? Explain why.
3. The measured delay (i.e., the delay you can see in the graphs) is composed of propagation delay, transmission delay, processing delay and queuing delay. Which of these delays depend on the packet size and which do not?