CS378 Lab 4: Network Diagnostics
Max. Marks: 37

As we studied in class, the Internet uses multiple protocols at different layers. It is a wonder that such a complex system functions at all! We do, however, sometimes encounter situations where we are not able to use the network as planned. In such situations, we wish to know what went wrong, to rectify the problem or decide if we should switch to another ISP etc.

Network administrators also want to monitor their networks to ensure that it is functioning properly. They have to worry not just about link, node, or protocol failures, but also must be on the lookout for security attacks.

Unfortunately, the Internet was not designed to give high priority to debugging the network, making network measurements, or making the network secure. However, there are some standard tools used by the networking community, a few of which we will study in this lab.

Preliminaries

a) **Capturing terminal output in files:**

As part of this lab, you will run many commands from one or more terminal windows. The commands you type in the terminals and the output which they give should be saved using the following commands.

Option 1 (preferred): **ttyrec -a <roll no>_lab4_terminal_<1/2/..>.txt**

Use a different number after "terminal" in the name for different terminal windows (if you are using more than one terminal window). You will later have to upload all of these files to Moodle.

NOTE: For every terminal window you use for the lab, run a command as given above.

Run command "exit" to stop ttyrec capturing the terminal output in the file.

Make sure to use "-a" to append to the old file. Otherwise if you stop using exit and restart using ttyrec then the file will be overwritten.

Option 2: Instead of ttyrec, you can also use “script”

script -a <roll no>_lab4_terminal_<1/2/..>.txt

Run command “exit” to stop script capturing the terminal output in the file.

Make sure to use “-a” to append to the old file. Otherwise if you stop using exit and restart using script then the file will be overwritten.

b) **Report**

You will also have to write a report which you will later upload to Moodle. This can be an odt, a docx, a pdf, or just a text file.

Name of the report file: <roll no>_lab4_report.<ext> , where <ext> is the appropriate extension.

c) **What to upload**
Finally upload to Moodle <roll no>_lab4.tgz which contains ALL terminal output files as well as the report.

Exercise 1: Investigating MAC and IP addresses of your machine’s own Network Interfaces

Relevant Command: `ifconfig`

Type “ifconfig” in any terminal window. **(Make sure you have started recording using ttyrec or script)**. It will show you a lot of information for each network interface in your machine. You might see a “loopback interface” such as “`lo :″` which is only a virtual interface (not a real network interface), which we will use later in the lab. Identify which of the other interfaces corresponds to the Wired Ethernet interface. (*Hint:* The command “route” will show you the default interface, which likely is the Ethernet interface).

In your report (6 marks), for the Ethernet interface on your lab machine state the following: (**Note:** Mention the Exercise number and the question number in your report).

1. What is the IPv4 address and number of bits used to represent it?
2. What is the IPv6 address (if any) and number of bits used to represent it?
3. What is the hardware address (MAC) and number of bits used to represent it?
4. What is the MTU value?

5. What does MTU mean? What units is it given in?
6. What is the transmit queue length on this interface? Give the units this is measured in.

Exercise 2: Find out if a remote machine is alive and the round trip time (RTT) to it

Relevant command: `ping <dest host>`

You must run Ping to
(i) one other machine in the lab (ask a friend for his IPv4 address),
(ii) www.cse.iitb.ac.in,
(iii) www.<dept>.iitb.ac.in (where <dept> is some non-CSE department such as “ee”, “civil” etc.)

In your report (9 marks), answer the following:

1. Ping uses ICMP. What does ICMP stand for?
2. How does one identify if an IP packet is an ICMP packet or not? (Hint: there must be a relevant field in the IP header)
3. What are the sizes of the ping packets (in bytes) sent out?
4. What is the average measured RTT to each of the three machines given above?
   Note: <CTRL-C> stops ping and gives this value.
5. What are the TTL values in the ping output observed for the 3 machines? Are they the same for all three machines?
   Note: If they are different you don’t have to explain why for this question. We will try to answer the question later.
6. In a short paragraph (in your own words) explain how ping works (what ICMP message does it send, what does it get back, etc.).
7. Try “ping www.google.com”. Do you see any output? If yes, state the RTT of the first 3 pings. If not, speculate what might be happening. (You can browse to find a potential answer or ask a sys-ad if he/she is around).
8. Go to an online server which can “ping” to any machine worldwide, such as ping.eu . What does it say your IP address is? Does this match with your IP address in Exercise 1.1? If they don’t match, suggest why this may be the case (you may browse online for an answer).
9. Click on the “Ping” option (on ping.eu) and ping www.google.com and www.iitd.ernet.in (this is iitd, not iitb). What is the average RTT, and TTL field observed for each of those?

Exercise 3: Finding layer-3 nodes along a full path from source to destination

Relevant command: `traceroute <dest host IP or name>`

In your report (8 (=4x2) marks), answer the following.

1. In a couple of paragraphs (in your own words), explain how traceroute works.
2. How many lines (which begin with a number: 1,2,3, etc.) of output are expected from a traceroute command, and what fields does each line contain?
3. Run traceroute to each of the 3 machines (within iitb) which you pinged in Exercise 2 from your lab machine. Do you see any relationship between the number of hops to each of these machines, and the corresponding TTL value from the ping commands (in Exercise 2 (Q 5))? If so, state what is the relationship and why you saw the particular TTL values from the ping commands.
4. Run traceroute to "www.google.com". If you do get an output, state the number of layer-3 hops on the path and what was the IP address of the last machine observed on the path (this sometimes may be an intermediate router, not the destination host). If a few lines had no meaningful output and only "* * *", then state why this may be occurring.

If you do not get any output, then try running traceroute to www.google.com from ping.eu and answer the same questions as above (IP address of last observed machine on path etc.)

Exercise 4: Find the IP addresses of servers corresponding to different URLs.

Relevant commands: `nslookup, host, dig`

We will study DNS in the last week of CS348. DNS helps get the IP address corresponding to URLs. All the above commands use DNS. You can use any of them to answer the following.

In your report (6 (=3x2)marks), answer the following.

1. What are the IPv4 addresses corresponding to www.mit.edu, www.cs.mit.edu, www.ee.mit.edu, www.ece.rice.edu, www.cs.rice.edu?
2. In your own words describe the relationship (i.e. how similar the IP addresses are) between the IP addresses of www.ece.rice.edu and www.cs.rice.edu. Is this expected? If so, why?
3. What do you notice about the relationship between the IP addresses of www.ee.mit.edu and www.cs.mit.edu? You may have noticed "CNAME" or

"canonical name" in the outputs for these two URLs. What does "canonical name" mean?

Exercise 5: Speed test to measure TCP throughput

Relevant command: `iperf`

This is a classic network bandwidth estimation tool used by network researchers. You can use "man iperf" to find out various options on how to use it. Iperf runs between two machines, an iperf server (on which "iperf -s" is run) and an iperf client (on which "iperf -c <server IP or name> <options>" is run). The client sends messages to the server to either start a TCP connection or a UDP stream. TCP does retransmission of packet losses and congestion control, whereas UDP does not do either (which we will study in CS348). Iperf reports the throughput obtained when data is sent from the client to the server, as well as some other information.

For TCP:
(at server) iperf -s
(at client) iperf -c <server IP or server name>
Note: You may have to use **CTRL-C** at the client machine to stop the iperf experiment

For UDP:
(at server) iperf -s
(at client) iperf -c <server IP or server name> -u -b <UDP data rate> (e.g. -b 10M gives 10Mbps)

In your report (4 (=2x2) marks), answer the following.

1. Start and iperf server in one terminal. In another terminal, run the iperf client (TCP) and give the server name as the IP address of the loopback interface you obtained from the ifconfig command. Stop using CTRL-C and report the Bandwidth (throughput) observed. Now you see the purpose of the loopback interface.
2. Ask a friend in the lab to start an iperf server. Run an iperf client (TCP) on your computer and give your friend's IP address as the server IP address. Report the Bandwidth (throughput) observed. Report the IP address of your friend's machine. Compare the Bandwidth to the answer reported in the previous question and give an explanation as to why one is much higher than the other.

We will skip using the UDP option for this lab.

Exercise 6: Measuring Available Bandwidth without using all the bandwidth

Relevant tool: pathchirp

Available Bandwidth on a network link is defined as the current unused bandwidth on the link. For example, if only 20 Mbps of a 100 Mbps link is currently being used, then the available bandwidth is 80 Mbps. The available bandwidth of a network path is the minimum available bandwidth on all links of the path.

One may want to measure the available bandwidth on a network path to find out if the path is congested or not. Since different links on a path are in different Autonomous Systems (AS), and some ASes do not make public their link utilization information, it is non-trivial for a common user to find out available bandwidth on a path.

One idea is to use a tool such as “iperf” to setup a TCP connection, since TCP is designed to use up the available bandwidth (and possibly even more, as we will later see) on a path. However, using up the entire available bandwidth to measure what it is seems like an overkill.

The question is, can we measure the available bandwidth on a path by only sending out a few data packets. Someone at Rice university built a measurement tool for just this purpose many years ago called “pathchirp”. It is still used today and the pathchirp research paper gets regularly cited by other papers. Pathchirp sends out only a few packets compared to iperf. However, there is no free lunch, and by using only a few packets (compared to a tool such as iperf), pathchirp’s measurements may be less accurate than iperf.

Pathchirp is already installed on your machines. Like iperf, it has a sender (server) and receiver (client).

Instructions for running the code are given below, but can also be found at this link. Pathchirp should be installed in your “Desktop” directory.

- **cd ~/Desktop/pathchirp-2.3.7/Bin**
- **ls ./**
  Let us call the result of this command [subdir]. Examples of [subdir] could be *x86_64*, or *i686, or i386, or sparc* and so on.
- **cd [subdir]**
  All the above commands must be run on both the machine that will send out chirp packets and the machine that will receive the chirp packets.

- On the machine sending out chirp packets run
  ./pathchirp_snd
- On the sender machine receiving chirp packets run
  ./pathchirp_rcv -S [sender machine name or IP address] -t 60
- At the receiver you will observe the output
  Opening file: [resultsfilename]
- After 60 seconds (1 minute) the experiment would have ended. The results will be in the file [resultsfilename] at the receiver in the format [timestamp] [Available bandwidth estimate in Mega bits/sec]
- To view the results run the following at the receiver machine
  more [resultsfilename]
- To rerun the experiment you only need to restart the ./pathchirp_rcv program at the receiver machine.

In your report (4 (=2x2) marks), answer the following.

1. Repeat the experiment of Exercise 5.2 but instead of iperf, make your friend run pathchrip_snd on his/her machine and you run pathchirp_rcv on your machine. Note that iperf will give you one number for Bandwidth whereas pathchirp will keep measuring bandwidth over time. Compare the average of the last 10 measurements (second column of file created with .instbw extension) of pathchirp with that of the results in Exercise 5.2. Are they comparable, at least in order of magnitude?
2. Glance at the pathchirp paper and describe in a paragraph what a “chirp packet train” is. You do not have to say how it is used, but just what it is.

Submission on Moodle:

Upload to Moodle <roll no>_lab4.tgz which contains ALL terminal output files as well as the report.