Internet Routes and Measure of Round Trip Times

Lab 4

50.005 Computer System Engineering

Due: 01 Apr 08:30 AM (Week 10)

Natalie Agus
INFORMATION SYSTEMS TECHNOLOGY AND DESIGN

Overview

In this lab exercise, you will learn how to use `ping` and `traceroute` to measure round trip times and find network routes.

Learning objectives

At the end of this lab exercise, you should be able to:

- Understand how the `ping` and `traceroute` utilities work.
- Use the `ping` utility to measure network round trip times.
- Use the `traceroute` utility to find network routes.
- Observe and understand the effects of varying packet sizes on delays experienced.

Preparation

You will need `ping` and `traceroute` to be installed on your OS. Most **Ubuntu** / **MacOS** installations should already include `ping` by default. You can install `traceroute` by running "`sudo apt-get install traceroute`" from the command line.

Submission

- **The total points for this lab is 35**
- Export this handout and fill in your answers in the blanks denoted in **blue**
- Export as pdf and **ZIP** it (not rar, or any other compression algorithm)
- **Upload** to `@csesubmitbot` telegram bot using the command `/submitlab4`
- **CHECK** your submission by using the command `/checksubmission`

Part 1: Measurement of round trip times using ping

The `ping` utility is one of the most widely-used network utilities. It enables you to measure the time that it takes for a packet to travel through the Internet to a remote host and back.

The `ping` utility works by sending a short message, known as an *echo-request*, to a remote host using the Internet Control Message Protocol (ICMP). When a host that supports ICMP receives an echo-request message, it replies by sending an echo-response message back to the originating host.

In the first part of this lab exercise, you will use the `ping` utility to send echo requests to a number of different hosts. In many of the exercises, you will be referring to hosts using their DNS names rather than their IP addresses. For more information about `ping`, you can look up its manual page by running "`man ping`" from the command line.

Round trip times

Use `ping` to send 10 packets to each of the following hosts. Each packet should have a size of 56 bytes, and there should be an interval of 5 seconds between each packet sent.

```
www.csail.mit.edu

www.berkeley.edu

www.usyd.edu.au

www.kyoto-u.ac.jp
```

Note: The size of each packet is 56 bytes by default, but you may observe that the actual size of the packet is larger than 56 bytes. You can look up the manual for `ping` to understand why such a discrepancy exists.

Question 1 [4pt] : For each host, record the percentage of packets sent that resulted in a successful response. Record also the minimum, average, and maximum round trip times for the packets that resulted in a response.

Your answer:

Website	Successful Percentage %	Min RTT	Average RTT	Max RTT
www.csail.mit.edu	100%	3.644	4.632	5.538
www.berkeley.edu	100%	212.822	213.266	213.691
www.usyd.edu.au	100%	95.550	174.051	270.703
www.kyoto-u.ac.jp	100%	3.854	5.210	6.515

Question 2 [4pt]: Describe and explain the differences in the minimum round trip time to each of these hosts.

Your answer: The minimum RTT is usually a function of how far the server is from the requesting client. Hence we can assume that the servers for MIT and Kyoto University are closer to us and hence have a lower RTT. The Sydney University servers are farther away but not as far away as Berkeley servers. It is also possible that the physical servers are not closer, but MIT and Kyoto University are using CDNs to cache content close to us, hence reducing RTT.

Other factors that may affect min RTT is server congestion and bandwidth saturation.

Question 3 [4pt]: Repeat the exercise using packet sizes of 56, 512 and 1024 bytes. Record the minimum, average, and maximum round trip times for each of the packet sizes. Why are the minimum round-trip times to the same hosts different when using 56, 512, and 1024–byte packets?

Your answer: Min RTT increases with packet size because more data has to be sent over the internet, and since the speed is (almost) same across time, the total time taken increases.

Website	Packet Size	Successful Percentage %	Min RTT	Average RTT	Max RTT

www.csail.mit.edu	56	100%	2.951	3.978	5.222
	512	100%	4.070	5.031	6.049
	1024	0%	-	-	-
www.berkeley.edu	56	100%	214.824	220.566	227.444
	512	100%	212.284	213.291	214.758
	1024	0%	-	-	-
www.usyd.edu.au	56	100%	95.150	164.573	200.708
	512	100%	96.446	157.124	197.727
	1024	0%	-	-	-
www.kyoto-u.ac.jp	56	100%	3.691	4.739	5.878
	512	100%	3.825	5.234	6.494
	1024	0%	-	-	-

Unanswered pings

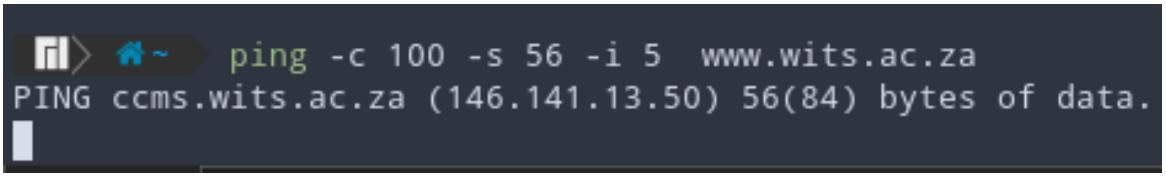
Use `ping` to send 100 packets to the following host. Each packet should have a size of 56 bytes, and there should be an interval of 5 seconds between each packet sent.

`www.wits.ac.za`

Question 4 [8pt]: Record the **percentage** of the packets sent that resulted in a **successful response** for each host. What are some possible **reasons** why you may not have received a response? (Be sure to check the host in a web browser).

You can provide screenshots to your answer.

Your answer: 0%, the host denies all the packets. This might be because the host blocks all non-HTTP requests (as HTTP requests still succeed as we can visit the website from the browser), or atleast blocks all ICMP ECHO requests.



Part 2: Understanding Internet routes using traceroute

The `traceroute` utility is another useful network utility. It enables you to trace the route taken by a packet from your machine to a remote host.

Here is an example of the output produced when `traceroute` is used to trace the route taken by a packet to `www.mit.edu`.

```
traceroute to www.mit.edu (118.215.81.86), 30 hops max, 60 byte pack-
ets

 1  192.168.9.2 (192.168.9.2)  0.221 ms  0.193 ms  0.107 ms
 2  10.12.0.1 (10.12.0.1)  3.363 ms  2.555 ms  3.253 ms
 3  172.16.1.106 (172.16.1.106)  3.072 ms  3.416 ms  3.418 ms
 4  172.16.1.210 (172.16.1.210)  4.977 ms  4.712 ms  4.921 ms
 5  192.168.22.27 (192.168.22.27)  4.806 ms  6.521 ms  6.451 ms
 6  103.24.77.1 (103.24.77.1)  7.172 ms  3.590 ms  3.187 ms
 7  201.210-193-8.qala.com.sg (210.193.8.201)  4.312 ms  9.056 ms 7.870 ms
 8  137.203-211-158.unknown.qala.com.sg (203.211.158.137) 8.904 ms 6.690 ms  6.555 ms
 9  213.203-211-158.unknown.qala.com.sg (203.211.158.213) 7.710 ms 5.423 ms  5.193 ms
10  203.116.10.125 (203.116.10.125)  6.783 ms  6.705 ms  6.440 ms
```

Each line in the output begins with a host on the route from your computer to `www.mit.edu`, followed by the round-trip times for 3 packets sent to that host. For more information about `traceroute`, you can look up its manual page by running "`man traceroute`" from the com- mand line.

Question 5 [5pt]: Explain how `traceroute` discovers a path to a remote host. (*Hint:* The `traceroute` manual will be helpful for answering this question.)

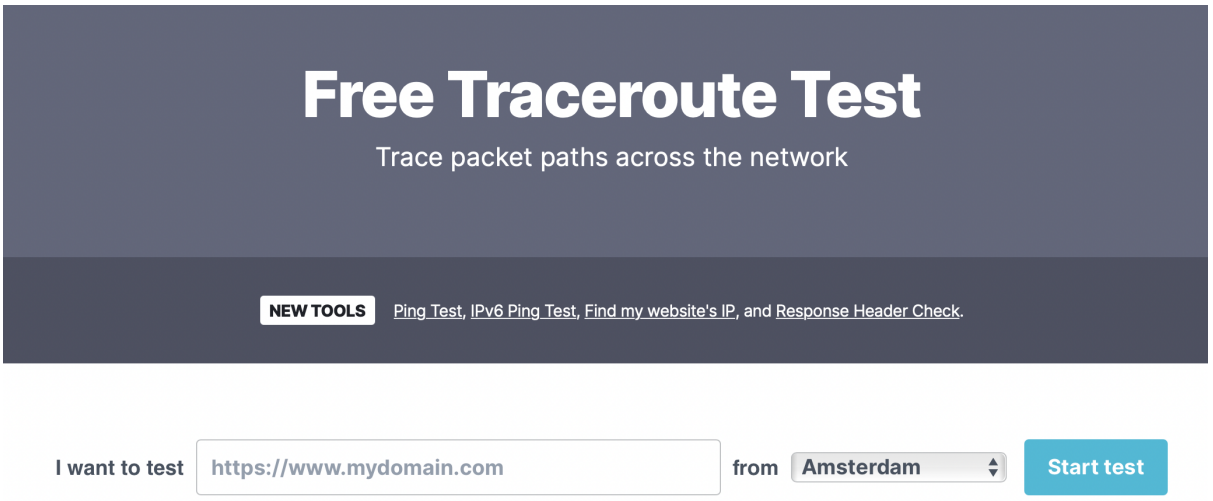
Your answer: The program starts off by sending packets with a TTL value of one and tries to listen for a ICMP TIME_EXCEEDED response. When this response is received, it is added to the list and a new packet with higher TTL is sent out. This continues until we receive a ICMP "Port Unreachable", which means that we have reached the host, or we reach a specified maximum number of hops. Usually there are multiple probe packets sent out simultaneously (default is 3).

Route asymmetries

In this exercise, you will run traceroute in two opposite directions. First, you will run traceroute on a remote host to see the route taken to your network. You will also run traceroute from your computer to see the route taken to that host.

Step 1: Find out your computer's public IP address. (Hint: You can use a website like http://www.whatismypublicip.com/, or search for "what is my ip" using Google's search engine.)

Step 2: Visit https://www.uptrends.com/tools/traceroute in your web browser. Enter your computer's public IP address, select the "from Location" and click "Start Test" to start a traceroute to your computer. Follow the steps shown below for at least three locations namely: New York, Amsterdam, Tokyo.



Step 3: After `traceroute` finishes running, you should be able to view the route taken from `specified location` to your network. Record the IP address of the first hop, which will be used in the next step.

Hop	Time	Time	Time	Host name	IP address
1	<1	<1	<1	gateway.as64425.com	5.182.210.1
2	-	-	-		
3	1	1	1		212.119.24.97
4	2	1	1	ae-10.r25.amstnl02.nl.bb.gin.ntt.net	129.250.2.90
5	12	12	12	ae-6.r20.parsfr04.fr.bb.gin.ntt.net	129.250.4.138
6	14	15	17	ae-2.r21.parsfr04.fr.bb.gin.ntt.net	129.250.3.46
7	90	87	86	ae-13.r24.asbnva02.us.bb.gin.ntt.net	129.250.6.6
8	149	149	149	ae-2.r24.snjsca04.us.bb.gin.ntt.net	129.250.6.237
9	-	-	-		
10	149	149	149	ae-2.r00.mlpsca01.us.bb.gin.ntt.net	129.250.4.101
11	148	148	148		129.250.24.196
12	155	155	155		129.250.130.254
13	148	148	148		198.107.143.162
14	-	-	-		

Step 4: On your computer, run `traceroute` using the IP address recorded in the previous step as the remote destination.

`$ traceroute <ip address from step 3>`

Question 6 [5pt]: Record the output of `traceroute` when run in both directions above. **Paste it as screenshots here or at the end of this document.**

Traceroutes were run on a VPS

My IP: 51.79.177.43

New York IP

Hop	Time	Time	Time	Host name	IP address
1	1	<1	<1		45.58.112.1
2	1	1	1		50.208.232.129
3	1	1	1		96.110.33.241
4	2	2	2		96.110.35.66
5	2	2	1		96.110.36.150
6	2	2	2		96.110.35.129
7	3	2	2		96.110.35.114
8	19	19	18		96.110.38.65
9	19	19	19		96.110.35.17
10	18	18	19		96.110.35.2
11	41	41	42		96.110.37.146
12	41	42	42		96.110.39.73
13	42	42	42		96.110.39.106
14	42	42	42		96.110.36.205
15	42	42	41		96.110.37.225
16	42	41	42		96.110.37.210
17	66	66	66		96.110.39.22
18	65	66	66		96.110.46.37
19	66	65	66		96.110.41.206
20	71	70	70		142.44.208.141
21	68	68	68		178.32.135.159
22	233	233	233		54.36.50.141
23	*	*	*	Request timed out	
24	*	*	*	Request timed out	
25	*	*	*	Request timed out	

```
root@shohamc1:~# traceroute 45.58.112.1
traceroute to 45.58.112.1 (45.58.112.1), 30 hops max, 60 byte packets
 1  10.37.130.2 (10.37.130.2)  0.074 ms  0.042 ms  0.025 ms
 2  10.50.192.252 (10.50.192.252)  0.205 ms 10.50.192.253 (10.50.192.253)  0.214 ms 10.50.192.252 (10.50.192.252)  0.180 ms
 3  10.133.2.100 (10.133.2.100)  0.366 ms  0.468 ms  0.584 ms
 4  10.75.0.10 (10.75.0.10)  0.156 ms 10.75.0.12 (10.75.0.12)  0.140 ms  0.111 ms
 5  10.75.248.2 (10.75.248.2)  0.963 ms 10.75.248.4 (10.75.248.4)  6.469 ms  6.481 ms
 6  sin-gss1-sbb1-nc5.sgp.asia (103.5.15.16)  0.768 ms  0.855 ms  0.803 ms
 7  mrs-mrs1-sbb1-nc5.fr.eu (103.5.15.30)  135.690 ms  135.687 ms  135.710 ms
 8  be107.par-th2-sbb1-nc5.fr.eu (54.36.50.134)  148.390 ms  148.377 ms  145.627 ms
 9  10.200.2.30 (10.200.2.30)  142.890 ms be102.rbx-g1-nc5.fr.eu (94.23.122.146)  146.738 ms  149.329 ms
10  be103.rbx-g4-nc5.fr.eu (54.36.50.229)  151.526 ms lon-thw-sbb1-nc5.uk.eu (54.36.50.240)  152.359 ms  154.708 ms
11  be100-1295.nwk-1-a9.nj.us (192.99.146.127)  223.012 ms lon-drch-sbb1-nc5.uk.eu (54.36.50.230)  149.863 ms  149.851 ms
12  be100-1298.nwk-5-a9.nj.us (192.99.146.133)  229.967 ms  233.327 ms be100-2.nwk-5-a9.nj.us (178.32.135.219)  223.810 ms
```

Amsterdam IP

Hop	Time	Time	Time	Host name	IP address
1	<1	<1	<1		5.182.210.1
2	1	1	1		87.245.246.38
3	15	15	15		87.245.233.109
4	*	*	*	Request timed out	
5	*	*	*	Request timed out	
6	*	*	*	Request timed out	
7	*	*	*	Request timed out	
8	24	24	24		91.121.131.74
9	34	34	34		54.36.50.135
10	*	*	*	Request timed out	
11	174	172	172		103.5.15.17
12	*	*	*	Request timed out	
13	*	*	*	Request timed out	
14	*	*	*	Request timed out	
15	*	*	*	Request timed out	
16	172	172	172		51.79.177.43

```
root@shohamc1:~# traceroute 5.182.210.1
traceroute to 5.182.210.1 (5.182.210.1), 30 hops max, 60 byte packets
 1  10.37.130.2 (10.37.130.2)  0.101 ms  0.049 ms  0.053 ms
 2  10.50.192.253 (10.50.192.253)  0.151 ms 10.50.192.252 (10.50.192.252)  0.173 ms  0.190 ms
 3  10.133.2.102 (10.133.2.102)  0.453 ms  0.543 ms 10.133.2.98 (10.133.2.98)  0.658 ms
 4  10.75.0.8 (10.75.0.8)  0.356 ms 10.75.0.14 (10.75.0.14)  0.334 ms  0.305 ms
 5  10.75.248.4 (10.75.248.4)  0.499 ms 10.75.248.2 (10.75.248.2)  0.937 ms 10.75.248.4 (10.75.248.4)  0.473 ms
 6  * sin-gss1-sbb1-nc5.sgp.asia (103.5.15.16)  0.663 ms  0.555 ms
 7  10.200.0.192 (10.200.0.192)  1.207 ms  1.155 ms sin-sg1-sbb1-nc5.sgp.asia (103.5.15.4)  0.995 ms
 8  33891.sgw.equinix.com (27.111.228.116)  0.598 ms  0.678 ms 10.200.0.192 (10.200.0.192)  1.000 ms
 9  33891.sgw.equinix.com (27.111.228.116)  0.622 ms  0.660 ms *
10  * * ae5-2074.ams10.core-backbone.com (81.95.2.138)  162.735 ms
```

Tokyo IP

Hop	Time	Time	Time	Host name	IP address
1	3	<1	<1		31.204.145.130
2	<1	1	1		109.200.218.1
3	68	67	67		109.200.218.223
4	*	*	*	Request timed out	
5	*	*	*	Request timed out	
6	*	*	*	Request timed out	
7	*	*	*	Request timed out	
8	*	*	*	Request timed out	
9	*	*	*	Request timed out	
10	*	*	*	Request timed out	
11	*	*	*	Request timed out	
12	*	*	*	Request timed out	
13	73	73	73		51.79.177.43

```
root@shohamc1:~# traceroute 31.204.145.130
traceroute to 31.204.145.130 (31.204.145.130), 30 hops max, 60 byte packets
 1  10.37.130.2 (10.37.130.2)  0.085 ms  0.048 ms  0.030 ms
 2  10.50.192.253 (10.50.192.253)  0.189 ms  0.167 ms  0.138 ms
 3  10.133.2.98 (10.133.2.98)  0.473 ms 10.133.2.96 (10.133.2.96)  0.337 ms 10.133.2.102 (10.133.2.102)  0.515 ms
 4  10.75.0.14 (10.75.0.14)  0.156 ms 10.75.0.12 (10.75.0.12)  0.152 ms 10.75.0.8 (10.75.0.8)  0.110 ms
 5  10.75.248.4 (10.75.248.4)  1.779 ms 10.75.248.2 (10.75.248.2)  0.470 ms  0.487 ms
 6  * * sin-gss1-sbb1-nc5.sgp.asia (103.5.15.16)  0.591 ms
 7  10.200.0.192 (10.200.0.192)  0.986 ms sin-sg1-sbb1-nc5.sgp.asia (103.5.15.4)  1.131 ms  1.116 ms
 8  * 10.200.0.192 (10.200.0.192)  0.943 ms *
 9  hosted-by.i3d.net (31.204.145.130)  73.768 ms  73.738 ms  73.887 ms
```

Question 7 [5pt]: Describe anything unusual you might observe about the output. Are the same routers traversed in both directions? If no, why might this be the case?

Your answer: The first few hops of each traceroute is the same, as they are internal network entities.

An usual occurrence is that the same routers are not traversed in both directions. This is immediately obvious as the number of hops needed to reach the host is not the same in all cases. The IP addresses encountered on the traceroute are also different. This might be due to different routes available for the same destination and packets highly unlikely to have the same exact path for the same set of source and destination.

Another unusual occurrence is that traceroutes for Amsterdam and New York are unsuccessful and have a lot of timeouts after the last entry. This might be an anti-DDoS protection by the DNS provider (or a router somewhere) as a possible attack might be to flood the router with UDP packets (default on traceroute) and hence the “real and useful” TCP packets cannot be served in a reasonable amount of time. Hence they might have decided to drop all UDP requests or route them to an unmonitored port, or flat out block the ports (default port used is 33434, incremented by 1 for every subsequent probe).

By specifying the -I flag while running traceroute, we can send ICMP ECHO for probes instead and here we are able to reach the host. (screenshot below)

```
traceroute to 5.182.210.1 (5.182.210.1), 30 hops max, 60 byte packets
 1  10.37.130.2 (10.37.130.2)  0.089 ms  0.058 ms  0.035 ms
 2  10.50.192.253 (10.50.192.253)  0.182 ms  0.194 ms  0.190 ms
 3  10.133.2.102 (10.133.2.102)  0.356 ms  0.478 ms  0.591 ms
 4  10.75.0.14 (10.75.0.14)  0.100 ms  0.136 ms  0.131 ms
 5  10.75.248.4 (10.75.248.4)  0.737 ms  0.748 ms  0.748 ms
 6  sin-gss1-sbb1-nc5.sgp.asia (103.5.15.16)  0.702 ms * *
 7  sin-sg1-sbb1-nc5.sgp.asia (103.5.15.4)  1.061 ms  1.060 ms  0.978 ms
 8  10.200.0.192 (10.200.0.192)  0.801 ms  0.860 ms  0.929 ms
 9  33891.sgw.equinix.com (27.111.228.116)  0.789 ms  0.793 ms  0.787 ms
10  ae5-2074.ams10.core-backbone.com (81.95.2.138)  162.649 ms  162.662 ms  162.657 ms
11  * * *
12  gateway.as64425.com (5.182.210.1)  180.690 ms  180.698 ms  180.692 ms
root@shohamc1:~# traceroute -I 45.58.112.1
traceroute to 45.58.112.1 (45.58.112.1), 30 hops max, 60 byte packets
 1  10.37.130.2 (10.37.130.2)  0.099 ms  0.057 ms  0.055 ms
 2  10.50.192.253 (10.50.192.253)  0.210 ms  0.220 ms  0.222 ms
 3  10.133.2.100 (10.133.2.100)  0.419 ms  0.553 ms  0.738 ms
 4  10.75.0.10 (10.75.0.10)  0.124 ms  0.147 ms  0.147 ms
 5  10.75.248.4 (10.75.248.4)  0.678 ms  0.690 ms  0.687 ms
 6  * * *
 7  mrs-mrs1-sbb1-nc5.fr.eu (103.5.15.30)  135.717 ms  135.748 ms  135.582 ms
 8  be107.par-th2-sbb1-nc5.fr.eu (54.36.50.134)  146.104 ms  146.129 ms  146.125 ms
 9  be103.rbx-g3-nc5.fr.eu (54.36.50.227)  146.793 ms  146.802 ms  146.815 ms
10  lon-thw-sbb1-nc5.uk.eu (54.36.50.240)  152.552 ms  152.565 ms  152.560 ms
11  be100-1295.nwk-1-a9.nj.us (192.99.146.127)  223.133 ms  223.745 ms  224.256 ms
12  be100-2.nwk-5-a9.nj.us (178.32.135.219)  219.288 ms  219.105 ms  219.437 ms
13  * * *
14  * * *
15  * * *
16  45.58.112.1 (45.58.112.1)  230.627 ms  230.654 ms  231.196 ms
```