ECE358: Computer Networks

Winter 2018

Project 3: Encapsulation and Network Utilities

Date of submission: Mar 28, 2019

Submitted by:
Student ID: 20558988
Student name Jing, Lun
Waterloo Email address: l7jing@edu.uwaterloo.ca

Marks received:

Marked by:

Table of Contents

Question 1. Protocol Header Analysis

f8

```
485b 397e 6fc6 14da e974 0821 0800 4500
0054 0000 4000 4001 c777 8161 380b 8161
3864 0800 8cb8 de36 0001 1377 8b52 0000
0000 2573 0a00 0000 0000 1011 1213 1415
1617 1819 1a1b 1c1d 1e1f 2021 2223 2425
2627 2829 2a2b 2c2d 2e2f 3031 3233 3435
```

Ethernet header:

```
485b 397e 6fc6 14da e974 0821 0800
```

485b 397e 6fc6 : Ethernet destination address is 48 5b 39 7e 6f c6 (unicast).
14da e974 0821 : Ethernet source address is 14 da e9 74 08 21 (unicast).
0800   : The payload type is IP (0x0800)

IP header:

```
4500 0054 0000 4000 4001 c777 8161 380b 8161 3864
```

4<u>5</u>: IPv4
4<u>5</u>: IP header is 4*5 = 20 bytes

00: (0 0 0 0 0 0 0 0 in binary)
    <u>0</u> <u>0</u> <u>0</u> 0 0 0 0 0: the diagram has routine precedence.
    0 0 0 <u>0</u> 0 0 0 0: Normal delay.
    0 0 0 0 <u>0</u> 0 0 0: Normal throughput.
    0 0 0 0 0 <u>0</u> 0 0: Normal Reliability.
    0 0 0 0 0 0 <u>0</u> <u>0</u>: Must be zero, reserved for Future Use.

0054: Total length of IP diagram is 84 (0x0054) bytes.

0000: The identification of this diagram is 0x0000 for fragmentation
purpose.

4000: (0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0):
      0 <u>1</u> 0 0 0 0 0 0 0 0 0 0 0 0 0 0: Don't fragment.
      0 1 <u>0</u> 0 0 0 0 0 0 0 0 0 0 0 0 0: More Fragment flag unset.
      0 1 0 <u>0 0 0 0 0 0 0 0 0 0 0 0 0</u>: Fragment offset is 0.

This diagram cannot be fragmented and there is no fragment after this
datagram, and this is the only fragment of the datagram.

40: Time to alive = 64 (0x40) the datagram may exist for at most 64 hops.

01: The Protocol on top is **ICMP.**

C777: The checksum of the datagram.

8161 380b: Source IP address is 129.97.56.11
8161 3864: Destination IP address is 129.97.56.100

ICMP header:

`0800 8cb8 de36 0001`

```
08: message type is echo.
00: code is zero.
8cb8: ICMP header checksum.
De36: Identifier.
0001: Sequence number.
```

Data:

```
                    1377 8b52 0000
   0000 2573 0a00 0000 0000 1011 1213 1415
   1617 1819 1a1b 1c1d 1e1f 2021 2223 2425
   2627 2829 2a2b 2c2d 2e2f 3031 3233 3435
```

f18

```
   0026 f1ff 0800 10bf 4878 4ef4 0800 4500
   003c c9c2 4000 4006 c0e6 8161 44a5 3212
   b7fa da1f 0050 7ee0 c6bd 0000 0000 a002
   16d0 5188 0000 0204 05b4 0402 080a 0008
   0f7f 0000 0000 0103 0307
```

Ethernet header:

`0026 f1ff 0800 10bf 4878 4ef4 0800`

```
0026 f1ff 0800: Ethernet destination address is 00 26 f1 ff 08 00 (unicast).
10bf 4878 4ef4: Ethernet source address is 10 bf 48 78 4e f4 (unicast).
0800          : The payload type is IP (0x0800)
```

IP header:

`4500 003c c9c2 4000 4006 c0e6 8161 44a5 3212 b7fa`

```
45: IPv4
45: IP header is 4*5 = 20 bytes

00: (0 0 0 0 0 0 0 0 in binary)
    0 0 0 0 0 0 0 0: the diagram has routine precedence.
    0 0 0 0 0 0 0 0: Normal delay.
    0 0 0 0 0 0 0 0: Normal throughput.
    0 0 0 0 0 0 0 0: Normal Reliability.
    0 0 0 0 0 0 0 0: Must be zero, reserved for Future Use.

003c: Total length of IP diagram is 60 (0x003c) bytes.
```

C9c2: The identification of this diagram is 0xc9c2 for fragmentation
purpose.

4000: (0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0):
        0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0: Don't fragment.
        0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0: More Fragment flag unset.
        0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0: Fragment offset is 0.

This diagram cannot be fragmented and there is no fragment after this
datagram, and this is the only fragment of the datagram.

40: Time to alive = 64 (0x40) the datagram may exist for at most 64 hops.

06: The Protocol on top is **TCP**.

c0e6: The checksum of the datagram.

8161 44a5: Source IP address is 129.97.68.165
3212 b7fa: Destination IP address is 50.18.183.250

TCP header:

da1f 0050 7ee0 c6bd 0000 0000 a002
16d0 5188 0000 0204 05b4 0402 080a 0008
0f7f 0000 0000 0103 0307

da1f: The source port is 55839.
0050: The Destination port is 80

7ee0 c6bd: The Seq. no. is 2128660157.
0000 0000: The Ack. No. is 0.

a0: Data offset is 40 (10 * 4)bytes. This is the length of the TCP header.

02: (0 0 0 0 0 0 1 0):
        Flags:
                URG: 0 ACK: 0 PSH: 0 RST: 0 SYN: 1 FIN: 0
The number in the synchronize sequence numbers is valid.

16d0: the receiver window size is 5840 (0x16d0) byte.
5188: Checksum of the whole TCP segment.
0000: Urgent pointer (Not used in this segment)

0204 05b4 0402 080a 0008 0f7f 0000 0000 0103 0307: Options data

Data:

        None data for this datagram.

Question 2. arp

The arp command is used to check, add, and remove address resolution protocol for the device.

From the following diagram, it shows all the current arp entries. For each entry, it includes the information of its domain name, IP address, MAC address and corresponding network interface. For example, the first entry's domain name is ecelinux4.uwaterloo.ca, its IP address is 129.97.56.14, with MAC address 60:a4:4c:53:e4:b1, and the interface is enp1s0.

```
l7jing@eceubuntu1:~$ /usr/sbin/arp -a
ecelinux4.uwaterloo.ca (129.97.56.14) at 60:a4:4c:53:e4:b1 [ether] on enp1s0
exsw02-circuitnet.uwaterloo.ca (129.97.56.1) at b4:99:ba:52:2c:00 [ether] on enp1s0
ecelinux.uwaterloo.ca (129.97.56.13) at 00:0a:cd:2a:f9:1e [ether] on enp1s0
eceserv1.uwaterloo.ca (129.97.56.9) at 00:25:90:5d:b6:47 [ether] on enp1s0
? (169.254.169.254) at <incomplete> on enp1s0
ecesvn.uwaterloo.ca (129.97.56.27) at c8:60:00:c8:77:8c [ether] on enp1s0
ecesystem.uwaterloo.ca (129.97.56.7) at 52:54:00:0c:98:ec [ether] on enp1s0
```

Question 3. ifconfig

The ifconfig is used for configuring the IP address and netmask of the network interfaces, it may change, enable or disable the interface.

Here, in the following diagram, the enp0s31f6, enp1s0 and lo are the names of active network interfaces. For each interface, it provides some information, for example, MAC address, internet IP address and broadcast address. It also includes the information on packets transmission, RX is for received packets and TX is for transmitted packets.

```
l7jing@eceubuntu1:/sbin$ ifconfig -a
enp0s31f6: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 30:5a:3a:83:92:bf  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 16  memory 0xf7000000-f7020000

enp1s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 129.97.56.12  netmask 255.255.255.0  broadcast 129.97.56.255
        inet6 fe80::de86:c6d4:cf68:5575  prefixlen 64  scopeid 0x20<link>
        ether 00:0a:cd:2a:fa:f3  txqueuelen 3000  (Ethernet)
        RX packets 66817358  bytes 24385090487 (24.3 GB)
        RX errors 17  dropped 0  overruns 0  frame 0
        TX packets 64075387  bytes 66480306930 (66.4 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 128  memory 0x2fffe00000-2fffe10000

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 25977944  bytes 34237087394 (34.2 GB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 25977944  bytes 34237087394 (34.2 GB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Question 4. netstat

netstate provides the information and statistics of the network interfaces. For example, for the interface enp1s0, the maximum transmission unit (MTU) is 1500 byte. There are 66820043 packets been received without error, and 17 packets been received with error detected. And it also transmitted 25978186 packets successfully. The Flg denotes the flags of the interface.

```
l7jing@eceubuntu1:/$ netstat -in
Kernel Interface table
Iface      MTU     RX-OK RX-ERR RX-DRP RX-OVR     TX-OK TX-ERR TX-DRP TX-OVR Flg
enp0s31f  1500         0      0      0 0              0      0      0      0 BMU
enp1s0    1500 66820043     17      0 0       64079292      0      0      0 BMRU
lo       65536 25978186      0      0 0       25978186      0      0      0 LRU
```

By using netstat -s, it will show the detailed information and statistics for different protocols, including IP, ICMP, TCP etc.

```
Ip:
    Forwarding: 2
    110998423 total packets received
    32 with invalid addresses
    0 forwarded
    9 with unknown protocol
    0 incoming packets discarded
    110998380 incoming packets delivered
    117715710 requests sent out
    128 dropped because of missing route
Icmp:
    130138 ICMP messages received
    432 input ICMP message failed
    ICMP input histogram:
        destination unreachable: 877
        timeout in transit: 1944
        echo requests: 126977
        echo replies: 324
        timestamp request: 7
        address mask request: 9
    128954 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 1062
        echo requests: 908
        echo replies: 126977
        timestamp replies: 7
IcmpMsg:
        InType0: 324
        InType3: 877
        InType8: 126977
        InType11: 1944
        InType13: 7
        InType17: 9
        OutType0: 126977
        OutType3: 1062
        OutType8: 908
        OutType14: 7
Tcp:
    117580 active connection openings
    87042 passive connection openings
    930 failed connection attempts
    8004 connection resets received
    25 connections established
    106676796 segments received
    151631791 segments sent out
    68916 segments retransmitted
    0 bad segments received
    13554 resets sent
Udp:
    4154620 packets received
    2244 packets to unknown port received
    0 packet receive errors
    7175507 packets sent
    0 receive buffer errors
    0 send buffer errors
    IgnoredMulti: 4954

UdpLite:
TcpExt:
    4 SYN cookies sent
    4 SYN cookies received
    2 invalid SYN cookies received
    569 resets received for embryonic SYN_RECV sockets
    557 packets pruned from receive queue because of socket buffer overrun
    36 ICMP packets dropped because they were out-of-window
    84030 TCP sockets finished time wait in fast timer
    39 time wait sockets recycled by time stamp
    91 packetes rejected in established connections because of timestamp
    1845662 delayed acks sent
    737 delayed acks further delayed because of locked socket
    Quick ack mode was activated 7858 times
    70227441 packet headers predicted
    6963039 acknowledgments not containing data payload received
    61712581 predicted acknowledgments
    TCPSackRecovery: 2510
    Detected reordering 978 times using SACK
    Detected reordering 27 times using time stamp
    1 congestion windows fully recovered without slow start
    21 congestion windows partially recovered using Hoe heuristic
    TCPDSACKUndo: 368
    135 congestion windows recovered without slow start after partial ack
    TCPLostRetransmit: 1195
    TCPSackFailures: 5
    12 timeouts in loss state
    55062 fast retransmits
    2002 retransmits in slow start
    TCPTimeouts: 656
    TCPLossProbes: 11457
    TCPLossProbeRecovery: 1498
    TCPSackRecoveryFail: 87
    TCPDSACKOldSent: 8153
    TCPDSACKOfoSent: 5
    TCPDSACKRecv: 8108
    TCPDSACKOfoRecv: 75
    1342 connections reset due to unexpected data
    258 connections reset due to early user close
    193 connections aborted due to timeout
    TCPDSACKIgnoredOld: 7
    TCPDSACKIgnoredNoUndo: 6487
    TCPSpuriousRTOs: 7
    TCPSackShifted: 81947
    TCPSackMerged: 53898
    TCPSackShiftFallback: 70051
    TCPDeferAcceptDrop: 2139
    TCPReqQFullDoCookies: 4
    TCPRcvCoalesce: 1494058
    TCPOFOQueue: 60408
    TCPOFOMerge: 5
    TCPChallengeACK: 2
    TCPSpuriousRtxHostQueues: 6
    TCPAutoCorking: 2990131
    TCPFromZeroWindowAdv: 890
    TCPToZeroWindowAdv: 890
    TCPWantZeroWindowAdv: 17182
```

```
    TCPSynRetrans: 140
    TCPOrigDataSent: 134544751
    TCPHystartTrainDetect: 742
    TCPHystartTrainCwnd: 16535
    TCPHystartDelayDetect: 101
    TCPHystartDelayCwnd: 3736
    TCPACKSkippedSynRecv: 40
    TCPACKSkippedPAWS: 64
    TCPACKSkippedSeq: 176
    TCPWinProbe: 67
    TCPKeepAlive: 15491
IpExt:
    InMcastPkts: 63131
    OutMcastPkts: 8542
    InBcastPkts: 8885
    InOctets: 96166857596
    OutOctets: 116372526904
    InMcastOctets: 2809996
    OutMcastOctets: 339625
    InBcastOctets: 1722386
    InNoECTPkts: 139193216
    InECT1Pkts: 6955
    InECT0Pkts: 212901
    InCEPkts: 1043
```

Using netstat -r will display the routing table for different destination IP addresses. Gateway basically describe the next hop to reach the destination. Genmask is the net mask. Flags represents flags, where U stands for "route is up", G stands for "using gateway". MSS is the maximum segment size for TCP connections over the route. Window is the default window size of the route. Irtt stands for initial round trip time. Iface is the network interface of the route.

```
l7jing@eceubuntu1:/$ netstat -r
Kernel IP routing table
Destination     Gateway         Genmask         Flags   MSS Window  irtt Iface
default         exsw02-circuitn 0.0.0.0         UG        0 0          0 enp1s0
129.97.56.0     0.0.0.0         255.255.255.0   U         0 0          0 enp1s0
link-local      0.0.0.0         255.255.0.0     U         0 0          0 enp1s0
```

Question 5. nslookup

Using nslookup can show the internet IP addresses and domain names for the host in parameter.

The server is the address of our own system's DNS. #53 means we are communicating with it in port 53.
There are three answers for ecelinux.uwaterloo.ca, it means that ecelinux.uwaterloo.ca will distribute the loads into these three different addresses. The Non-authoritative answer means the current server do not have the original source files of the domain's zone.
The only difference between the two requests is the order of the addresses are changed, therefore, the order of the answer should not affect the results.

```
l7jing@eceubuntu1:/$ nslookup ecelinux.uwaterloo.ca
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   ecelinux.uwaterloo.ca
Address: 129.97.56.13
Name:   ecelinux.uwaterloo.ca
Address: 129.97.110.135
Name:   ecelinux.uwaterloo.ca
Address: 129.97.110.136

l7jing@eceubuntu1:/$ nslookup ecelinux.uwaterloo.ca
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
Name:   ecelinux.uwaterloo.ca
Address: 129.97.110.136
Name:   ecelinux.uwaterloo.ca
Address: 129.97.110.135
Name:   ecelinux.uwaterloo.ca
Address: 129.97.56.13
```

We can see that www.mit.edu also distributes the access queries into three different addresses.

```
l7jing@eceubuntu1:/$ nslookup www.mit.edu
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
www.mit.edu     canonical name = www.mit.edu.edgekey.net.
www.mit.edu.edgekey.net canonical name = e9566.dscb.akamaiedge.net.
Name:   e9566.dscb.akamaiedge.net
Address: 23.195.207.111
Name:   e9566.dscb.akamaiedge.net
Address: 2600:140a:0:395::255e
Name:   e9566.dscb.akamaiedge.net
Address: 2600:140a:0:38b::255e
```

Whereas www.gmail.com has domain names mail.google.com and googlmail.l.google.com, and it has two different addresses.

```
l7jing@eceubuntu1:/$ nslookup www.gmail.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
www.gmail.com   canonical name = mail.google.com.
mail.google.com canonical name = googlemail.l.google.com.
Name:   googlemail.l.google.com
Address: 172.217.1.165
Name:   googlemail.l.google.com
Address: 2607:f8b0:400b:809::2005
```

For the www.facebook.com, the domain name is star-mini.c10r.facebook.com, and it has two different addresses.

```
l7jing@eceubuntu1:/$ nslookup www.facebook.com
Server:         127.0.0.53
Address:        127.0.0.53#53

Non-authoritative answer:
www.facebook.com        canonical name = star-mini.c10r.facebook.com.
Name:   star-mini.c10r.facebook.com
Address: 157.240.18.35
Name:   star-mini.c10r.facebook.com
Address: 2a03:2880:f127:283:face:b00c:0:25de
```

Question 6. ping

Using ping can help us to check if network data packets could be distributed on the address provided. Basically means if we can send and receive packets from the destination address.

By using ping -c10 *hostname*, we will send 10 packets to the destination host to check if we can access the host and how long will it take to connect the host. In the first example, we did not transmit any data pack successfully. But we can actually access the www.ualberta.ca via web browser. The reason is that the www.ualberta.ca probable block the ping assessment from outside.

```
l7jing@eceubuntu1:/$ ping -c10 www.ualberta.ca
PING alb-uofa-prod-itm-sc8cds-1708315660.us-east-1.elb.amazonaws.com (52.5.221.80) 56(84) bytes of data.

--- alb-uofa-prod-itm-sc8cds-1708315660.us-east-1.elb.amazonaws.com ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9209ms
```

To connect to www.lemonde.fr, the average time is 5.358 ms. And we can access www.lemonde.fr via web browser.

```
l7jing@eceubuntu1:/$ ping -c10 www.lemonde.fr
PING s2.shared.global.fastly.net (151.101.126.217) 56(84) bytes of data.
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=1 ttl=53 time=5.39 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=2 ttl=53 time=5.42 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=3 ttl=53 time=5.32 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=4 ttl=53 time=5.36 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=5 ttl=53 time=5.38 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=6 ttl=53 time=5.42 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=7 ttl=53 time=5.46 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=8 ttl=53 time=5.35 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=9 ttl=53 time=5.36 ms
64 bytes from 151.101.126.217 (151.101.126.217): icmp_seq=10 ttl=53 time=5.35 ms

--- s2.shared.global.fastly.net ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9014ms
rtt min/avg/max/mdev = 5.326/5.385/5.468/0.106 ms
```

To connect to www.ucla.edu, the average time is 79.540 ms. And we can access www.ucla.edu via web browser.

```
l7jing@eceubuntu1:/$ ping -c10 www.ucla.edu
PING gateway.lb.it.ucla.edu (164.67.228.152) 56(84) bytes of data.
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=1 ttl=40 time=79.4 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=2 ttl=40 time=79.4 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=3 ttl=40 time=79.5 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=4 ttl=40 time=79.5 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=5 ttl=40 time=79.5 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=6 ttl=40 time=79.5 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=7 ttl=40 time=79.4 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=8 ttl=40 time=79.5 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=9 ttl=40 time=79.6 ms
64 bytes from gateway.lb.it.ucla.edu (164.67.228.152): icmp_seq=10 ttl=40 time=79.6 ms

--- gateway.lb.it.ucla.edu ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9014ms
rtt min/avg/max/mdev = 79.411/79.540/79.699/0.343 ms
```

Question 7. traceroute

Using traceroute can find out the intermediate hops that required to transmit a packet from the starting machine to the ending location.

To access the www.uwaterloo.ca, there are 11 intermediate hops

```
l7jing@eceubuntu1:/$ /usr/sbin/traceroute www.uwaterloo.ca
traceroute to www.uwaterloo.ca (129.97.208.23), 30 hops max, 60 byte packets
 1  exsw02-circuitnet.uwaterloo.ca (129.97.56.1)  3.501 ms  3.483 ms  3.462 ms
 2  v490-eng-rt-e2.ns.uwaterloo.ca (172.16.32.193)  4.657 ms  4.644 ms  4.624 ms
 3  te4-3-dist-rt-phy.ns.uwaterloo.ca (172.18.7.21)  0.895 ms  0.877 ms  1.067 ms
 4  * * xe1-0-0-u11-dist-sa-mc-trust.ns.uwaterloo.ca (172.31.0.149)  0.756 ms
 5  te2-12-dist-rt-mc-global.ns.uwaterloo.ca (172.31.0.161)  1.173 ms  1.238 ms  1.336 ms
 6  te2-16-cn-rt-rac.ns.uwaterloo.ca (172.16.31.113)  0.919 ms  0.825 ms  0.808 ms
 7  e1-1-cr-rt-mc.ns.uwaterloo.ca (172.16.16.65)  2.352 ms  2.321 ms  2.379 ms
 8  xe4-0-1-22-cr-sa-bb2.ns.uwaterloo.ca (172.16.16.7)  1.132 ms  1.056 ms  0.969 ms
 9  e1-25-20-cr-rt-mc-area2.ns.uwaterloo.ca (172.16.16.27)  2.463 ms  2.573 ms  1.910 ms
10  wms.uwaterloo.ca (129.97.208.23)  1.295 ms  1.253 ms  1.341 ms
11  wms.uwaterloo.ca (129.97.208.23)  1.433 ms !N  1.384 ms !N  1.368 ms !N
```

To access www.youtube.com, there are 15 intermediate hops.

```
l7jing@eceubuntu1:/$ /usr/sbin/traceroute www.youtube.com
traceroute to www.youtube.com (172.217.1.14), 30 hops max, 60 byte packets
 1  exsw02-circuitnet.uwaterloo.ca (129.97.56.1)  3.538 ms  3.521 ms  3.501 ms
 2  v490-eng-rt-e2.ns.uwaterloo.ca (172.16.32.193)  3.935 ms  3.921 ms  3.900 ms
 3  te4-3-dist-rt-mc.ns.uwaterloo.ca (172.18.7.17)  0.879 ms  0.960 ms  1.026 ms
 4  xe1-0-0-u10-dist-sa-mc-trust.ns.uwaterloo.ca (172.31.0.145)  0.428 ms  0.433 ms  0.405 ms
 5  te2-12-dist-rt-mc-global.ns.uwaterloo.ca (172.31.0.161)  1.253 ms  1.524 ms  1.706 ms
 6  te2-16-cn-rt-rac.ns.uwaterloo.ca (172.16.31.113)  0.827 ms  0.804 ms  0.795 ms
 7  te-0-0-2-1-ext-rt-mc.ns.uwaterloo.ca (172.16.31.229)  1.307 ms  1.420 ms  1.363 ms
 8  unallocated-static.rogers.com (72.142.108.181)  1.676 ms  3.948 ms  3.958 ms
 9  24.156.146.189 (24.156.146.189)  9.013 ms  8.970 ms  8.854 ms
10  9044-cgw01.wlfdle.rmgt.net.rogers.com (209.148.230.45)  4.949 ms  5.403 ms  4.992 ms
11  209.148.233.38 (209.148.233.38)  5.538 ms 209.148.230.6 (209.148.230.6)  5.520 ms  5.480 ms
12  72.14.209.126 (72.14.209.126)  5.908 ms 72.14.222.87 (72.14.222.87)  4.847 ms  5.018 ms
13  108.170.250.241 (108.170.250.241)  6.007 ms * *
14  216.239.42.60 (216.239.42.60)  6.311 ms  6.252 ms 216.239.42.158 (216.239.42.158)  5.962 ms
15  iad23s25-in-f14.1e100.net (172.217.1.14)  5.950 ms 74.125.244.147 (74.125.244.147)  5.947 ms  6.018 ms
```

To access www.nytimes.com, since we track maximum 30 hops, it seems that we did not reach www.nytimes.com within 30 hops, and starts to lose information from the 12th hop.

```
l7jing@eceubuntu1:/$ /usr/sbin/traceroute www.nytimes.com
traceroute to www.nytimes.com (151.101.125.164), 30 hops max, 60 byte packets
 1  exsw02-circuitnet.uwaterloo.ca (129.97.56.1)  3.561 ms  3.530 ms  3.510 ms
 2  v490-eng-rt-e2.ns.uwaterloo.ca (172.16.32.193)  3.891 ms  4.017 ms  4.006 ms
 3  te4-3-dist-rt-mc.ns.uwaterloo.ca (172.18.7.17)  0.509 ms  0.767 ms  0.876 ms
 4  xe1-0-0-u10-dist-sa-mc-trust.ns.uwaterloo.ca (172.31.0.145)  0.465 ms  0.437 ms  0.424 ms
 5  te2-12-dist-rt-mc-global.ns.uwaterloo.ca (172.31.0.161)  0.979 ms  1.063 ms  1.155 ms
 6  te2-16-cn-rt-mc.ns.uwaterloo.ca (172.16.31.117)  0.737 ms  0.774 ms  0.826 ms
 7  te0-0-2-0-ext-rt-mc.ns.uwaterloo.ca (172.16.32.149)  1.464 ms  1.581 ms  1.532 ms
 8  unallocated-static.rogers.com (72.142.108.181)  1.464 ms  2.274 ms  2.423 ms
 9  24.156.146.189 (24.156.146.189)  6.026 ms  5.977 ms  5.852 ms
10  9044-cgw01.wlfdle.rmgt.net.rogers.com (209.148.230.45)  5.078 ms  5.438 ms  5.386 ms
11  209.148.233.38 (209.148.233.38)  5.753 ms 209.148.235.34 (209.148.235.34)  5.345 ms 209.148.233.38 (209.148.233.38)  5.638 ms
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * *
18  * * *
19  * * *
20  * * *
21  * * *
22  * * *
23  * * *
24  * * *
25  * * *
26  * * *
27  * * *
28  * * *
29  * * *
30  * * *
```