Name 1:
Name 2:

# COM-407: TCP/IP NETWORKING

# LAB EXERCISES (TP) 0
# BASIC CONFIGURATION AND IP SUITE:
# PING(6), TRACEROUTE(6), NETSTAT, NSLOOKUP
# With Solutions

October 13, 2016

**Abstract**

In this lab you will practice some networking commands that enable you to obtain information about Internet machines and about the connectivity and the paths between them.

## 1 ORGANIZATION OF THE TP AND USEFUL COMMANDS

### 1.1 TP REPORT

Type your answers in this document. We recommend you use Adobe Reader XI to open this PDF, as other readers (such as SumatraPDF, but also older versions of Adobe!) don't support saving HTML forms. That will be your TP report (one per group). When you finish, save the report and upload it on moodle. Don't forget to write your names on the first page of the report. **See on Moodle for the deadline.**

### 1.2 WIRESHARK

You will be using Wireshark to sniff packets. Since there are a lot of packets generated by the applications running on your machine, you may want to use filters. `http://wiki.wireshark.org/DisplayFilters`

## 2 THE IPV4 INTERNET

Connect to the Internet in IPv4 and disable any IPv6 connectivity. Then, in order to determine the following information:

- the IP address(es) of your machine <my_ip>,
- the netmask <my_netmask>, and
- the default gateway of your machine <my_gateway>.

In MacOS use

```
# ifconfig
# netstat -nr
```

In Linux use

```
# ifconfig
# route -n
```

or in Windows

```
> ipconfig /all
```

**Q1/** List your findings here:

- IP address: 128.178.151.219
- Network Mask: 255.255.255.0
- Default Gateway: 128.178.151.1

**Q2/** Is your IP address public or private? What does the netmask in IPv4 (or the prefix in IPv6) mean?

***Solution.*** *In this case, the IP address is public, which can be confirmed by navigating to the link* `http://www.myipaddress.com` *and confirming that the IP address given in the web page is the same as the one given to the Ethernet adapter. The netmask or prefix is used to distinguish the "network" and the "host" parts of an IP address.*

Now, download Wireshark and install it on your computer. Start it (as administrator) and use the menu `Capture->Interfaces` to start capturing packets on the interface that you use for Internet connectivity.

**Q3/** Do you see any packet captured with destination IP address of your default gateway?

***Solution.*** *No, unless you are pinging your default gateway or communicating directly with it by any mean (DNS, FTP, HTTP, SCP, etc). In IP, communication is done end-to-end thus in general we should not see IP packets with destination IP address any of the intermediate devices, including the default gateway.*

## 2.1 PING

PONG

The ping command uses the ICMP protocol to probe whether a host is up:

```
# ping <hostname>
```

**Q4/** Start a new capture with Wireshark and then ping `www.facebook.com`. Observe the traffic generated by the ping command. Do you see only ICMP packets?. Stop the ping program and start it again after a couple of seconds. Is there a difference from the first captured packets? Explain.

```
128.178.151.139 128.178.15.227 DNS 74 Standard query 0x59d0  A www.google.com
28.178.15.227 128.178.151.139 DNS 154 Standard query response 0x59d0  A 173.194.35.17
      A 173.194.35.18 A 173.194.35.19 A 173.194.35.20 A 173.194.35.16
128.178.151.139 173.194.35.18 ICMP 74 Echo (ping) request  id=0x0001, seq=3/768, ttl=128
173.194.35.18 128.178.151.139 ICMP 74 Echo (ping) reply    id=0x0001, seq=3/768, ttl=52
```

***Solution.*** *First a DNS query is performed, next a ping request is sent to the IP address of facebook.*

*The second time the DNS request is typically not performed. The IP address was cached.*

*Other valid observations: the ARP request for the gateway is not performed either (ARP cache), the sequence numbers continue from where they left off during the first ping, another IP address is used (due to Facebook's load balancing system), etc.*

**Q5/** In a browser open `www.swisscom.ch`. Next, try pinging it. Explain.

***Solution.*** *The server hosting the website is up, yet it is configured not to respond to ping (ICMP is disabled).*

## 2.2 TRACEROUTE AND NETSTAT

**traceroute** is a tool for displaying the route to a destination.

In MacOS and Linux:

```
# traceroute www.facebook.com
```

In Windows:

```
> tracert www.facebook.com
```

**Q6/** Do you see more than one name/IP address at any of the hops? If so, why?

```
tsf-484-wpa-4-040:~ mohiuddi$ traceroute www.facebook.com
traceroute to star-mini.c10r.facebook.com (31.13.64.35), 64 hops max, 52 byte packets
 1  cv-gigado-v484 (128.179.184.1)  72.081 ms  2.642 ms  1.149 ms
 2  c6-ext-v200 (128.178.200.1)  1.126 ms  3.746 ms  1.599 ms
 3  swiel2 (192.33.209.33)  2.093 ms  1.914 ms  1.825 ms
 4  swiel2-10ge-5-3.switch.ch (130.59.36.78)  3.270 ms  2.346 ms  2.210 ms
```

```
 5  swice2-10ge-4-1.switch.ch (130.59.37.65)  2.832 ms  3.245 ms  3.616 ms
 6  swice3-p23.switch.ch (130.59.36.210)  3.708 ms  3.118 ms  4.326 ms
 7  br02.ams1.tfbnw.net (80.249.209.164)  19.682 ms  18.969 ms  18.255 ms
 8  be2.bb01.ams3.tfbnw.net (204.15.20.10)  26.151 ms  27.995 ms  26.756 ms
 9  ae21.bb02.ams2.tfbnw.net (31.13.27.66)  25.323 ms  31.258 ms  26.080 ms
10  ae1.pr02.ams2.tfbnw.net (74.119.79.195)  18.956 ms  18.960 ms  18.236 ms
11  po102.psw01c.amt2.tfbnw.net (157.240.32.17)  18.176 ms  18.973 ms  18.399 ms
12  msw1am.01.amt2.tfbnw.net (173.252.66.217)  18.807 ms
    msw1ac.01.amt2.tfbnw.net (173.252.65.1)  22.155 ms
    msw1al.01.amt2.tfbnw.net (173.252.66.219)  19.376 ms
13  edge-star-mini-shv-01-amt2.facebook.com (31.13.64.35)  20.706 ms  20.008 ms  21.127 ms
```

***Solution.*** *Yes, on hop 12, we see three different IP Addresses. For discovering hops, traceroute sends UDP packets with increasing TTL to the UDP echo port (tracert in Windows uses ICMP echo requests). For each hop, 3 packets are sent. At hop 12, the three packets are sent on 3 different paths because of a load-balancer on the way that is trying to balance loads on three different paths. The three paths are not seen on Windows machine because the load-balancer ignores ICMP traffic.*

**netstat** is a tool for displaying TCP connections, routing table, interfaces and network statistics. Open a web browser, go to `lca.epfl.ch`, and leave the browser open for the moment.

Look at the active TCP connections.

```
# netstat -t -n
```

The `-n` switch prevents name resolving and makes netstat display results faster (but obviously without the names of the hosts).

**Q7/** Identify the TCP connections opened by visiting the `lca.epfl.ch` webpage. Write them down and describe them here. Is there one, or are there several such connections? Why?

***Solution.*** *Several connections are established, as modern browsers load in parallel the HTML document and the graphics (images, sound, etc.).*

# 3 Names in the Internet

*Juliet*: [...]
What's in a name? That which we call a rose
By any other name would smell as sweet.

W.S.

Replace your DNS servers by an inexisting IP address, say `1.2.3.4`. If you configured statically your DNS servers, don't forget to write them down somewhere.

Go to the `Properties` of your Internet connection. Click on Internet Protocol Version 4, `Properties`, choose `Use the following DNS server addresses`, and write `1.2.3.4`

Use the manual configuration in the network settings and set the DNS address to `1.2.3.4`

Switch to root mode using `su` and edit the `/etc/resolv.conf` file. Comment out the lines that begin with `nameserver` (precede them with the `#` character) and add one line `nameserver 1.2.3.4`

**Q8/** Try pinging Facebook and observe the traffic with Wireshark. What happens?

***Solution.*** *A DNS request is sent to the bogus server `1.2.3.4` with no reply back*

**Q9/** Try pinging the IP address of Facebook that you discovered in Sections 2.1 and 2.2. Does it work?

***Solution.*** *Since there is no need to resolve a name, the ping to Facebook's IP address works fine.*

**nslookup** is a command-line tool for querying Domain Name System (DNS) name servers. Run `nslookup` with the address of the Google public DNS server.

```
# nslookup - 8.8.8.8
```

**Q10/** In the > prompt, type `lca.epfl.ch`. Give the IPv4 and IPv6 addresses of `lca.epfl.ch`. Use `set type=A` for IPv4 or `set type=AAAA` for IPv6

```
icsil1noteb147:~ barreto$ nslookup
> set type=A
> lca.epfl.ch
Server: 2001:620:618:10f:1:80b2:f07:1
Address: 2001:620:618:10f:1:80b2:f07:1#53

lca.epfl.ch canonical name = lca1srv2.epfl.ch.
Name: lca1srv2.epfl.ch
Address: 128.178.156.24
> set type=AAAA
> lca.epfl.ch
Server: 2001:620:618:10f:1:80b2:f07:1
Address: 2001:620:618:10f:1:80b2:f07:1#53

lca.epfl.ch canonical name = lca1srv2.epfl.ch.
lca1srv2.epfl.ch has AAAA address 2001:620:618:19c:1:80b2:9c18:1
```

***Solution.*** *IPv4 address:* `128.178.156.24`
*IPv6 address:* `2001:620:618:19c:1:80b2:9c18:1`

**Q11/** Do you recognize the IPv4 address in the IPv6 address, or vice-versa?

***Solution.*** *An IPv4 address,* `128.178.156.24`*.*

*An IPv6 address,* `2001:620:618:19c:1:80b2:9c18:1`*.*

*There is a mapping between IPv4 and IPv6 addresses (IPv4 appears in the IPv6 address:* `80b2:9c18`*).*

Restore now your initial DNS configuration.

Start a capture in `wireshark` and do a traceroute in IPv4 to `www.facebook.com`. Focus on the line:

```
swiel2 (192.33.209.33)  1.219 ms  0.968 ms  0.944 ms
```

**Q12/** Look at the capture and identify the packet in which you see the name `swiel2`. How does this differ from the usual DNS response observed in previous questions? Based on the observed difference, comment on how `traceroute` works.

***Solution.*** *The* `traceroute` *tool sends testing packets to each intermediate router between the host machine and the destination IP address. In return, each intermediate router respond to such testing packets by sending one packet back to the host machine. All packets (which by the way are UDP and ICMP packets) contain only IP addresses. By default the* `traceroute` *tool makes a reverse DNS query for the IP address of each intermediate router, and then it displays the name in the output of the traceroute command. To disable this reverse query (and thus making the command faster), when typing the traceroute command you can use the "*`-n`*" argument in Mac and Linux, or the "*`-d`*" argument in Windows*

**Q13/** Analyze the capture and comment on how `traceroute` find successive hops.

***Solution.*** *By varying the TTL.*

# 4 The IPv6 Internet

Now let's examine the situation when only IPv6 connectivity is present.

Find an access to an IPv6 network and disable IPv4 on your machine. IPv6 access is provided in INF019 via a wireless access point, or on the PCs in the room via a wired connection.

Use wireshark to observe the traffic. On your computer type

```
# ping6 www.facebook.com
```

**Q14/** Describe some differences in the observed traffic compared to the IPv4 case. Write the average RTT you get and compare it with the IPv4 case. Explain the differences if any.

***Solution.*** *IPv6 and IPv4 packets may take different paths to reach the destination host, also at any given moment we could experience congestion in the network, thus RTT may be different. Differences are also in packet length, protocol used, etc.*

Repeat the test with the `traceroute` command from Section 2. Use:

In Linux or MacOS:

```
# traceroute6 www.facebook.com
```

In Windows:

```
> tracert -6 www.facebook.com
```

**Q15/** Write the result. Does the path to Facebook in the IPv6 Internet cross the same routers as in IPv4?

```
icsil1noteb157:~ mohiuddi$ traceroute6 www.facebook.com
traceroute6 to star-mini.c10r.facebook.com (2a03:2880:f11c:8083:face:b00c::25de) from
 2001:620:618:197:1:80b2:97d7:1, 64 hops max, 12 byte packets
 1  cv-ic-dit-v151-ro  0.477 ms  0.281 ms  0.433 ms
 2  cv-gigado-v100  0.388 ms  0.466 ms  0.378 ms
 3  c6-ext-v200  0.511 ms  0.469 ms  0.441 ms
 4  swiel1-10ge-0-0-0-2.switch.ch  1.168 ms  1.156 ms  1.055 ms
 5  swiel2-10ge-5-3.switch.ch  0.898 ms  1.000 ms  0.755 ms
 6  swice2-10ge-4-1.switch.ch  1.671 ms  1.651 ms  1.552 ms
 7  swice3-p23.switch.ch  1.647 ms  1.724 ms  1.712 ms
 8  2001:7f8:1::a500:6762:1  17.286 ms  17.302 ms  17.253 ms
 9  lo0.franco32.fra.seabone.net  30.441 ms  35.908 ms  30.567 ms
10  2001:41a8:600:2::162  17.802 ms  23.524 ms
    2001:41a8:600:2::15e  23.361 ms
11  po111.asw04.fra2.tfbnw.net  18.316 ms
    po114.asw01.fra2.tfbnw.net  19.627 ms
    po121.asw01.fra2.tfbnw.net  19.966 ms
12  po203.psw01c.frt3.tfbnw.net  18.722 ms
    po204.psw01c.frt3.tfbnw.net  23.821 ms
    po201.psw01b.frt3.tfbnw.net  19.562 ms
13  po3.msw1ac.01.frt3.tfbnw.net  19.592 ms
    po2.msw1ai.01.frt3.tfbnw.net  25.719 ms
```

```
    po3.msw1ad.01.frt3.tfbnw.net  28.020 ms
14  edge-star-mini6-shv-01-frt3.facebook.com  24.834 ms  18.878 ms  18.851 ms

tsf-484-wpa-4-040:~ mohiuddi$ traceroute www.facebook.com
traceroute to star-mini.c10r.facebook.com (31.13.64.35), 64 hops max, 52 byte packets
 1  cv-gigado-v484 (128.179.184.1)  72.081 ms  2.642 ms  1.149 ms
 2  c6-ext-v200 (128.178.200.1)  1.126 ms  3.746 ms  1.599 ms
 3  swiel2 (192.33.209.33)  2.093 ms  1.914 ms  1.825 ms
 4  swiel2-10ge-5-3.switch.ch (130.59.36.78)  3.270 ms  2.346 ms  2.210 ms
 5  swice2-10ge-4-1.switch.ch (130.59.37.65)  2.832 ms  3.245 ms  3.616 ms
 6  swice3-p23.switch.ch (130.59.36.210)  3.708 ms  3.118 ms  4.326 ms
 7  br02.ams1.tfbnw.net (80.249.209.164)  19.682 ms  18.969 ms  18.255 ms
 8  be2.bb01.ams3.tfbnw.net (204.15.20.10)  26.151 ms  27.995 ms  26.756 ms
 9  ae21.bb02.ams2.tfbnw.net (31.13.27.66)  25.323 ms  31.258 ms  26.080 ms
10  ae1.pr02.ams2.tfbnw.net (74.119.79.195)  18.956 ms  18.960 ms  18.236 ms
11  po102.psw01c.amt2.tfbnw.net (157.240.32.17)  18.176 ms  18.973 ms  18.399 ms
12  msw1am.01.amt2.tfbnw.net (173.252.66.217)  18.807 ms
    msw1ac.01.amt2.tfbnw.net (173.252.65.1)  22.155 ms
    msw1al.01.amt2.tfbnw.net (173.252.66.219)  19.376 ms
13  edge-star-mini-shv-01-amt2.facebook.com (31.13.64.35)  20.706 ms  20.008 ms  21.127 ms
```

***Solution.*** *There are some routers with the same name in the two cases. It is not impossible that they are dual-stack routers. The path is however not identical!*

*From the IPv6 wireless network provided in INF019 a consecutive series of hops do not respond. A likely theory is the following: we are using `6to4` (i.e., encapsulating IPv6 in IPv4 packets, we will see details later in the course). The ICMPv6 messages are not seen by the IPv4 nodes used for the tunnel (they just see a regular IPv4 packet whose payload happens to be an IPv6 packet). Probably one of the two IPv6 ends of the tunnel artificially introduces a number of "fake" hops in order to account for the legacy IPv4 network. The way this can be done is by simply dropping packets with TTL less than a certain value.*

Now, open the web browser (new window), go to `lca.epfl.ch`.

**Q16/** Do you notice a difference between two versions of `lca.epfl.ch` pages? Can you imagine by which mechanism such a difference may occur ?
*Hint: Which device (default gateway, intermediate routers, the web server, etc) do you think is in charge of displaying the web content for IPv4 if you are connected to an IPV4 network or for IPv6 otherwise?*

***Solution.*** *There is an IPv6 logo at the bottom of the page. The Route Rank widget does not work in IPv6.*

*Who put this logo on the page we received ? The web server did it. In this case the same web server is reached over IPv4 and IPv6 (in other settings they might be different) but the web server itself, when it is contacted by a client, knows on which network (IPv4 or IPv6) the HTTP request arrives (based on sockets, as we will see later in the course). The web server then runs a script that puts the IPv6 logo in the page when the request arrived over IPv6. Intermediate systems are of course not involved in this.*

Look at the active connections.

```
# netstat -t -n
```

**Q17/** Compare the output that is related to `lca.epfl.ch` with the one that you wrote down for IPv4. Comment about it

***Solution.*** *We can see that the transport layer (TCP) connections are the same for IPv4 and IPv6 networks.*

**Q18/** Try pinging `www.swisscom.ch` again. Did it work? Explain.

***Solution.*** *It works. IPv4 and IPv6 configurations are run separately in routers. It is likely that ICMP is not disabled in the IPv6 interface of* `www.swisscom.ch` *website).*

# 5 IPv4 AND IPv6

Let's see what happens when both IPv4 and IPv6 Internet connectivities are present. Stay connected in IPv6, but enable IPv4.
From your computer do a traceroute in IPv4 and IPv6 to `www.switch.ch`

**Q19/** Does it work in both cases?. Write down any difference in the traceroutes

***Solution.*** *Traceroute works in both cases, and they traverse same routers since the name of intermediate routers are the same.*

Now, start a new `Wireshark` capture, open a browser and type `www.switch.ch`.

**Q20/** Check the capture in Wireshark, your connection to the webpage is done in IPv4 or in IPv6? ***Solution.*** *On Mac, it prefers IPv6 if available*

**Q21/** Explain how do you think your machine could decide whether it uses IPv4 or IPv6.

***Solution.*** *It depends on your machine but in general IPv6 is preferred over IPv4 and decision is based on the DNS query. If the target host has an `AAAA` record, your machine tries an IPv6 connection; if not it goes for IPv4. However, some vendors have decision-making algorithms that tracks the latency on the IPv4 or IPv6 network and based on that decide which network they will use.*