TÉCNICO LISBOA

Computer Networks — 2021/22		**Assignment:**	Lab 2
Configuring & Debugging Networks		**Issued:**	2021-12-02
Network Tools & Configuration		**Submission Due:**	2021-12-05
Authors:	Francisco Chamiça Pereira Prof. Luis D. Pedrosa	**Comments Due:**	2021-12-09
		Version:	1.0

> **Submission note.** You will need to later submit all the answers to this lab's questions in Moodle. As such, we recommend that you take note of each answer during the lab so that later you can just copy them into Moodle. Be sure to also save any screenshots you take, as you can later upload them into your Moodle submission.

1 Networking tools

The goal of this lab is to show you how to use networking tools like `ping`, `traceroute`, `ip`, `wireshark`, and `iperf` to discover a network topology and correctly identify problems in the network.

Using the provided virtual machine:

1. Execute the command `git pull origin master` in `/home/rc/lab-files`.
2. Run the Core Emulator
3. Open `/home/rc/lab-files/lab-net-tools/net-tools-1.imn`.

1.1 `ip`

`ip` is a networking tool that allows you to control interfaces, routes, devices, and tunnels. Use `man ip` to check its manual page, or `ip -h` to print the help menu. You can use it to show all the interfaces with `ip address show`.

Q1 Use the `ip` tool to catalog every host with their respective interfaces and IPs (with corresponding subnet masks), *except* for Carol. *Note: you can create a table in Moodle.*

Q2 Notice that Carol has no IP. Use the `ip` command to give the IP `10.0.2.20/24` to the interface `eth0` of Carol's machine, and take note of the command. After adding the IP, add a default gateway to the machine so it can find the other networks: `ip route add default via 10.0.2.1`. Run the command `ping 10.0.2.20` in alice to check the connectivity with carol. Take a screenshot of the output of the command. Submit both the command and the screenshot as answers to this question. *Note: you can inline an image to your answer in Moodle.*

1.2 `traceroute`, `wireshark`

`wireshark` is a packet sniffer. It allows you to see every packet that is either received or sent by the interfaces in a given host. To open wireshark in a given host's interface, simply right-click the host on `Core` and select `Wireshark`, followed by the interface you want to sniff. To filter the packets that are shown by the interface, you can input filter expressions in the display filter (the horizontal input bar at the top of `wireshark`).

`traceroute` allows you to discover the paths taken by the packets when going from one machine to another. Again, you can access its man page using `man traceroute`, or print its helping menu with `traceroute --help`. If, for example, one wanted to discover the paths taken by packets going from server A to server B, one would run `traceroute B` inside server A.

In order to discover the paths taken by the packets, traceroute uses the Time To Live (TTL) field of the IP protocol to increasingly get closer to the desired destination. First, it sends UDP/IP packets with TTL=1. Because whenever an IP packet is received by a host its TTL is decremented, when these packets arrive at the next host they either:

1. Arrive at the destination with enough TTL to not be dropped, or
2. they arrive with insufficient TTL, and an ICMP `Time to live exceeded in transit` packet is sent back to the source.

In case (2), the source records the IP of the host that sent back the ICMP packet, and increases the TTL of subsequent packets to discover further hops. This process continues until the source no longer receives the ICMP error packets. At that point, `traceroute` outputs the IPs it collected for the entire route from source to destination.

Q3 Open `wireshark` on interface `eth0` of `router1`. Write `udp` in the display filter, and press enter to show only UDP packets. Now run `traceroute` from alice to bob. Use the output shown in `wireshark` to exemplify how `traceroute` works (take at most 4 screenshots of packets to support your explanation). *Note: Each `traceroute` packet associated with a given TTL is sent 3 times.*

Q4 Catalog the path taken by the packets going from the following pairs of sources and destinations: alice->bob, alice->carol, alice->dan, dan->bob, dan->alice. Annotate the host's name of each hop instead of its interface's IP. *Note: your answer should be presented as a table with 2 columns: "Direction" (with, for example, "alice->bob") and "Route" (with, for example, "alice->A->B->C->bob").*

Q5 Notice the strange paths taken by packets between alice and dan. What is going on out of the ordinary? Take at most 3 screenshots of `wireshark` in the interfaces you deem relevant to explain the matter. *Note: support your screenshots with an explanation.*

Q6 Based on **Q5**, when you run `traceroute` on alice, does it find the alice->dan path, or the dan->alice path? How could you find out the opposite path?

1.3 `ping`

`ping` allows you to send ICMP echo requests to another machines. They are a great tool to check connectivity between hosts. To run `ping` between machines A and B, just execute `ping A` whilst in B.

> **Note.** You can filter ICMP packets in `wireshark` by using the `icmp` filter.

Computer Networks — 2021/22		**Assignment:**	Lab 2
Configuring & Debugging Networks		**Issued:**	2021-12-02
Network Tools & Configuration		**Submission Due:**	2021-12-05
Authors:	Francisco Chamiça Pereira Prof. Luis D. Pedrosa	**Comments Due:**	2021-12-09
		Version:	1.0

Q7 Send ICMP requests from alice to dan, and watch the interface `eth1` from `router1` using `wireshark`. Do you see ICMP requests and replies, just requests, or just replies?

Q8 Send ICMP requests from alice to dan, and watch the interface `eth2` from `router1` using `wireshark`. Do you see ICMP requests and replies, just requests, or just replies?

Q9 Explain what you saw in **Q7** and **Q8** with the information retrieved having analyzed the network with `traceroute`.

1.4 `iperf`

Using `iperf`, we can quickly perform performance tests in our network. It needs a server instance running, to which the client instance will connect later on. Then, the client generates traffic and sends it to the server, which then calculates the throughput achieved.

The `iperf` manual page can be accessed by `man iperf`, and its help menu can be seen with `iperf -h`.

To run the server instance, you can run `iperf -s` in a given machine. Then, connect to the server with the `iperf` client using `iperf -c <SERVER_IP> -t 5` (`-t 5` just tells the client to generate traffic for 5 seconds).

Q10 Measure the throughput achieved between alice and bob, and alice and carol. Use the result reported by the server. *Note: your answer should be presented as a table with 2 columns: "Direction" (for example "alice->bob") and "Throughput" (for example "1 kbps").*

2 Fixing a networking issue

Open `/home/rc/lab-files/net-tools/net-tools-2.imn`. Its network topology is identical to `net-tools-1.imn`, except that now there is a malfunctioning link somewhere.

Q11 Discover the malfunctioning link in the network. Which one was it? *Hint: remember ping?*

You can fix the link by stopping the session, right-clicking on top of the link and selecting `Delete`. Next, select the `link tool` option on the left vertical bar and connect the hosts missing the link, thus "replacing the faulty cable". Check if the anomaly was fixed.