

Assignments with NetGUI-Kathará

Assignment 2: IP datagram and IP address configuration

Computer Networks

GSyC - URJC

Departamento de Teoría de la Señal y Comunicaciones y
Sistemas Telemáticos y Computación

2025

In this assignment, you will learn how to configure the network interfaces of *hosts* and *routers* using two different methods: interactively, through the use of the `ifconfig` or `ip` commands, and statically, using configuration files. In addition, the fields of the IP header will be studied in detail.

1. Fields of the IP Header

Load the file `pcap1.pcap` in Wireshark.

Select the first and only packet, and expand the fields of the IP header in the section where the protocol details of the selected packet are shown.

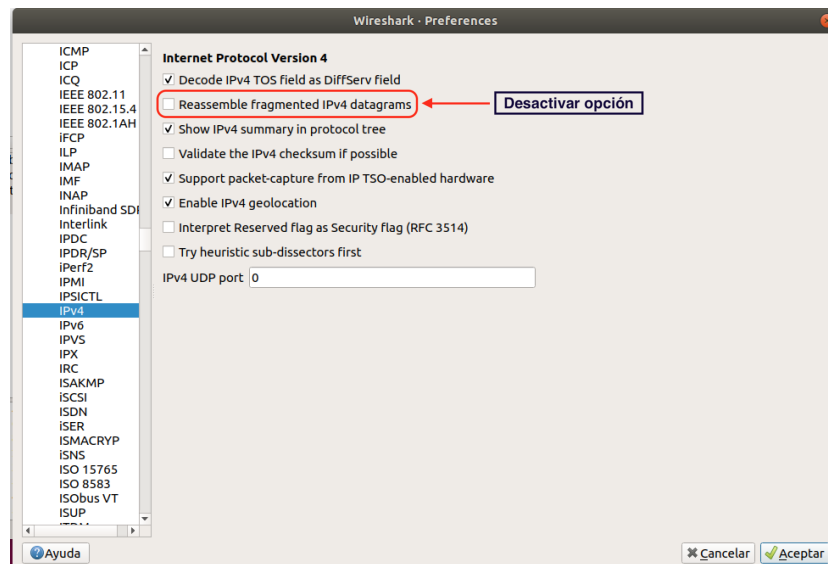
Answer the following questions:

1. What is the source IP address and the destination IP address of the packet?
2. Indicate the value of the TTL field.
3. Knowing that the traffic capture was performed on the destination machine of the packet and that the packet was initially sent from the source machine with $TTL=64$, indicate how many intermediate *routers* the packet passed through.
4. Indicate the protocol field value in the IP header and explain what this field means.

2. IP Fragmentation

Load the file `pcap2.pcap` in Wireshark.

Open the menu `Edit` → `Preferences`, expand the `Protocols` section, and look for the `IPv4` protocol. Disable the option shown in the figure:



The capture shows 3 packets that are 3 fragments of an original IP datagram.

Answer the following questions:

1. How can you tell that the 3 packets belong to the same original datagram?
2. Indicate how much IP data (number of data bytes in the data field of the original IP datagram) is carried in each of the datagrams into which the original datagram has been fragmented. Could the first and second IP datagrams carry more IP data? Why?
3. Indicate how much IP data the original unfragmented IP datagram would contain.
4. Since IP datagrams may arrive out of order, explain how the destination can reorder the fragments and reconstruct the original datagram.
5. Explain how you can tell that the first packet in the capture is the first fragment of a fragmented datagram, rather than a normal unfragmented datagram. Note that Wireshark does not show anything in this first packet that would indicate it is a fragment (unlike the other packets).
6. Explain how you can tell that the last packet in the capture is the final fragment of a fragmented datagram, rather than a normal unfragmented datagram.
7. Now enable the Wireshark option that you previously disabled. Observe how the displayed information changes for the packets:
 - Wireshark now identifies the first packet as a fragment, and also the second, but not the third.
 - Wireshark indicates in the first and second packets that it has reassembled the 3 fragments into the third packet.
 - In the third packet, the header remains as is, but at the end of the IP header, it details the fields of the 3 fragments: [3 IPv4 Fragments (4008 bytes): #1(1480), #2(1480), #3(1048)]
 - The third packet includes in the data section the aggregated data from all 3 fragments.

3. IP Address Configuration

3.1. The ifconfig/ip Command

- Unzip the lab archive 02-lab-IPdatagram1.tgz. Start NetGUI-Kathará and load the folder you just unzipped: 02-lab-IPdatagram1.
- You will see a network scenario like the one in figure 1, where pc1, pc2, and pc3 are three computers, and r1 is a *router*.

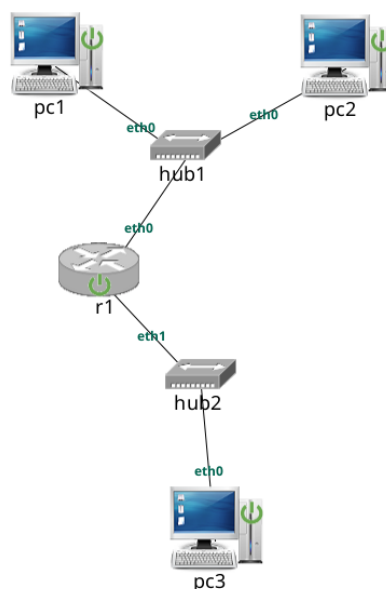


Figura 1: Lab 02-lab-IPdatagram1. Network composed of three PCs and a *router*.

- Start all network devices, including the PCs and the *router*. The *hubs* in NetGUI-Kathará are passive elements and do not need to be started.
1. Check the network configuration on each machine and on the router using the `ifconfig` or `ip` command. Which network interfaces are configured on each, and what IP address is assigned to each interface?
 2. Using the `ifconfig` or `ip` command, assign IP addresses to the network interfaces of the machines and the router as follows:
 - Use 255.255.255.0 as the `netmask` in all cases.
 - Assign an address starting with 151.0.0. to all interfaces connected to `hub1`.
 - Assign an address starting with 152.0.0. to all interfaces connected to `hub2`.

NOTE: Keep in mind that since `r1` is connected to two *hubs*, it will have two IP addresses, one for each interface (`eth0` and `eth1`).

3. Notice that the IP addresses you have configured are shown in the NetGUI interface. Verify the addresses of the interfaces on each virtual machine using `ifconfig` or `ip`. Include in your lab report a screenshot of the NetGUI scenario showing the IP addresses you have configured.
4. Start a traffic capture on `pc2`. To do this, run the following command on `pc2`:

```
pc2:~# tcpdump -i eth0 -s 0 -w /shared/p2-IPdatagram-01.pcap
```

Now you are going to generate traffic as follows: `pc1` will send IP datagrams to `pc2`, and `pc2` will reply. In the previous lab, you used the `arping` tool where you provided the destination machine's Ethernet address as a parameter. In this lab, we will use the `ping` tool, where you provide the destination machine's IP address. This tool will send messages to that IP address, and the destination machine will automatically send a reply for each received message.

To do so, run the following on `pc1`:

```
pc1:~# ping -c 1 151.0.0.Y
```

Where:

- The IP address you must use is the destination IP address of the packets—in this case, the address of `pc2` (replace the `Y` with the actual IP address of the `pc2` machine in your scenario).
- The `-c 1` option tells `ping` to send a single packet to the `pc2` machine, which will automatically respond with a reply message.

Stop the capture by pressing `Ctrl+C` in the `pc2` window.

Analyze the packets shown in the capture. For each packet, indicate:

- Source Ethernet address.
- Destination Ethernet address.
- Type of encapsulated protocol (field `Type`). If the protocol type is IP, also indicate:
 - Source IP address
 - Destination IP address

5. Shut down the `r1` router, and once it is off, start it again. Verify that its IP address configuration has disappeared.

3.2. The `/etc/network/interfaces` File

Unzip the lab file `02-lab-IPdatagram2.tgz`. Start NetGUI-Kathará and load the folder `02-lab-IPdatagram2`. A scenario like the one shown in Figure 2 will appear.

1. How many different networks (groups of interfaces that are adjacent or directly connected to each other) do you think there are in the figure?

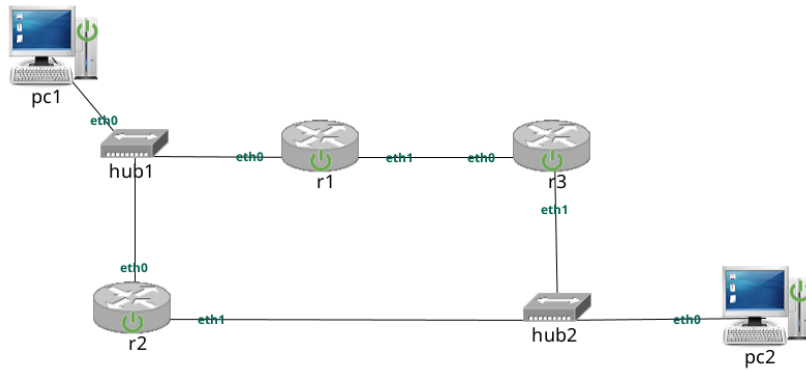


Figura 2: Lab 02-lab-IPdatagram2. Network composed of 2 PCs and 3 routers

2. Start all devices—the PCs and the *routers*. Check that their network interfaces are not configured by running `ifconfig` or `ip`.

3. Edit the `/etc/network/interfaces` file on each machine and add IP addresses as follows:

- Use 255.255.255.0 as the **netmask** in all cases.
- For all interfaces connected to one of the networks, assign an address starting with 201.0.0 ...
- For all interfaces connected to another network, assign an address starting with 202.0.0 ... and another one starting with 203.0.0 ...

4. On each machine, run the necessary command to apply the network interface configuration from the configuration file:

```
systemctl restart networking
```

Check that the interfaces are configured using `ifconfig`. Notice that the IP addresses you have configured are also shown in the NetGUI interface. Include in your lab report a screenshot of the NetGUI scenario showing the IP addresses you have configured.

5. On **r1**, run the necessary command to deactivate the network configuration:

```
systemctl stop networking
```

Check with `ifconfig` how the network interface configuration has been lost on **r1**.

6. Run again the necessary command to enable the network configuration and set up the interfaces according to what is specified in `/etc/network/interfaces`:

```
systemctl start networking
```

7. Modify the IP address of **r3(eth1)** in the `/etc/network/interfaces` file on **r3** to assign it a different IP address from the one you previously assigned, making sure it still belongs to the same subnet as before. Don't forget to run the command to reactivate the network configuration every time you modify the `/etc/network/interfaces` file.

8. The changes you make to the `/etc/network/interfaces` file will persist after rebooting the machines. Verify this by shutting down **r3** and starting it again. Once it has restarted, run `ifconfig` and check that both interfaces on **r3** are correctly configured.

9. Start a traffic capture on **r2**, interface **eth0**. To do this, run the following command on **r2**:

```
r2:~# tcpdump -i eth0 -s 0 -w /shared/p2-IPdatagram-02.pcap
```

Now you are going to generate traffic as follows: **pc1** will send packets to **r1**, and **r1** will respond. To do this, run the following command on **pc1**:

```
pc1:~# ping -c 2 A.B.C.D
```

Where:

- The IP address to use is the destination IP address of the packets—in this case, the IP address of the `eth0` interface on `r1` (replace A.B.C.D with the actual IP address of `r1-eth0` in your scenario).
- The `-c 2` option tells `ping` to send 2 packets to `r1`, and `r1` will automatically respond to each one.

Stop the capture by pressing `Ctrl+C` in the `r2` window.

Analyze the captured packets. For each packet, indicate:

- Source Ethernet address
- Destination Ethernet address
- Type of encapsulated protocol (field `Type`). If the protocol type is IP, also indicate:
 - Source IP address
 - Destination IP address

10. Now try (without capturing traffic) to perform a ping from `pc1` to the IP address of the `eth1` interface on `r1`. What happens? What do you think might be the reason?

4. Assignment Submission

Save the capture files in a folder named `p2` containing `p2-IPdatagram-01.pcap` and `p2-IPdatagram-02.pcap`. Upload the files to the link provided in Aula Virtual, and before the submission deadline, compress the `p2` folder into a file named `p2.zip`, along with another file containing the report in PDF format:

- Report in PDF format.
- `p2.zip`: folder `p2` with the capture files inside.