



## SECOND TERM

### Assessable Activity

Computer Systems  
CFGS DAW

Álvaro Maceda

[a.macedaarranz@edu.gva.es](mailto:a.macedaarranz@edu.gva.es)

2022/2023

Version:230314.1528


#### License




**Attribution - NonCommercial - ShareAlike (by-nc-sa):** No commercial use of the original work or any derivative works is permitted, distribution of which must be under a license equal to that governing the original work.

## Nomenclature

Throughout this unit different symbols will be used to distinguish important elements within the content. These symbols are:

 Important

 Attention

 Interesting

## SECOND TERM: ASSESSABLE ACTIVITY

You must write all the commands as text. **All the screenshots and images attached will be removed before correcting the exercise.** We will only keep the network diagram.

### 1. EXERCISE 1

You have available the IP addresses from [192.168.123.32](#) to [192.168.123.64](#).

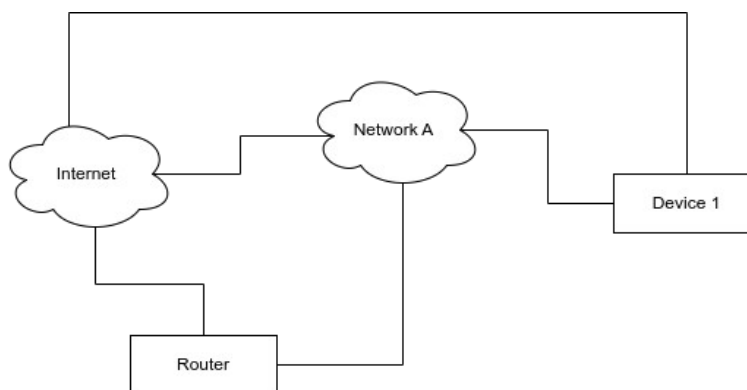
You must design two networks:

1. Network A will have 2 devices connected
2. Network B will have 4 devices connected
3. There is a router with Internet access, and one router for each network.

Design an addressing scheme for the networks to work.

#### Deliverables

- Draw the network topology (you can use, for example, <https://app.diagrams.net/>). This is an (incorrect) example of the type of diagram you must draw:



- Indicate the network address for each network.
- Indicate the IP address for each device (remember that the routes will have two addresses)  
The Internet address for the corresponding router is [213.214.215.216](#). Fill a table like this for each network:

#### Network X

Device	IP
Device 1	
...	
Router	

Gateway: (network's gateway IP address)

## 2. EXERCISE 2

### 2.1 Part 1

Create a Docker image named `exercise3-server` based on Ubuntu 22.04. The image:

- Will run an `openssh-server` at port 33
- There should be two users: `user1` (password `user1server`) and `user2` (password `user2server`)

#### Hints

To configure the SSH server you can add a `.conf` file with the directives needed in the directory `/etc/ssh/ssd_config.d` (for example, `/etc/ssh/sshd_config.d/port.conf`)

To create a user with a password, you can use the option `-p "$(openssl passwd -1 <user password here>)"` for the corresponding command.

To start the SSH server you will need to create the directory `/run/sshd` in the container and launch the command `/usr/sbin/sshd -D -o ListenAddress=0.0.0.0` when the container starts.

To develop the image, you launch a basic Ubuntu container, test the commands there and, once working, add them to the image's creation file.

#### Deliverables:

- Content of the files needed to create the image
- Command to build the image

### 2.2 Part 2

Run two containers at the same time. The first one will be named `ex3-server` and will be created from the previous image. The second one will be named `ex3-client`, and will be created from the Ubuntu 22.04 image.

- The containers will have Internet access and will be in an isolated network.
- The `ex3-server` SSH server would be also accessible at port 22 in your host.

#### Hints

To connect to an SSH server in a non-standard port, you can use the `-p <port>` parameter for the `ssh` command.

If you need some network commands in your containers, you can install `iproute2`, `net-tools` and/or `iputils-ping`

#### Deliverables:

- Command(s) to launch the containers
- Command(s) to test the SSH service from `ex3-client`. You will need to install `ssh` in the container (you don't need to create an image for the client)
- Command(s) to test the SSH service from your `localhost`

