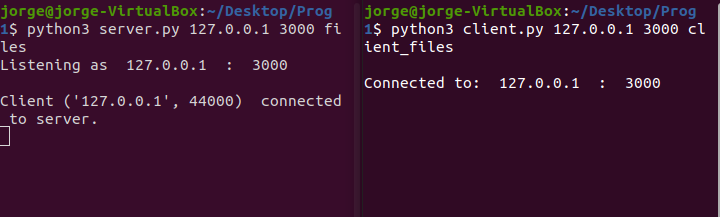
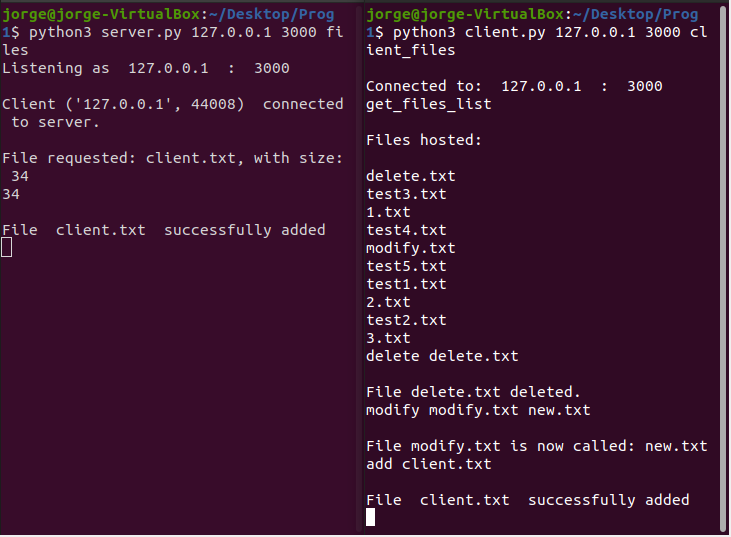
**Output files and Performance evaluation results:**

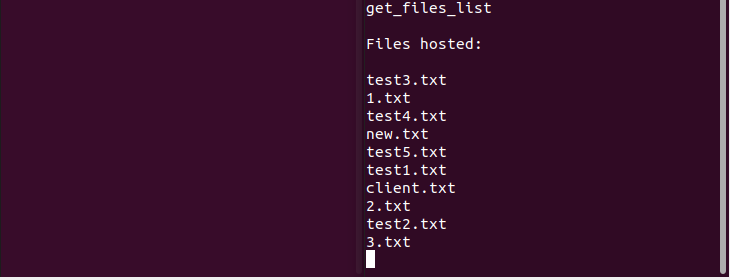
1. Connecting with client to the server and ensure requirements and 1 file transfer:

* Connection: Client – Server Connection:

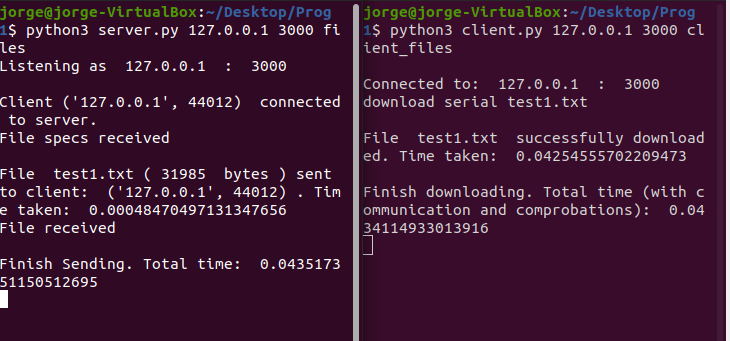


* Commands: get\_files\_list (Client gets the list of files that the server contains), delete (client removes one file from the server), add (client sends a file into the server) and modify (modify the name of a file from the server).

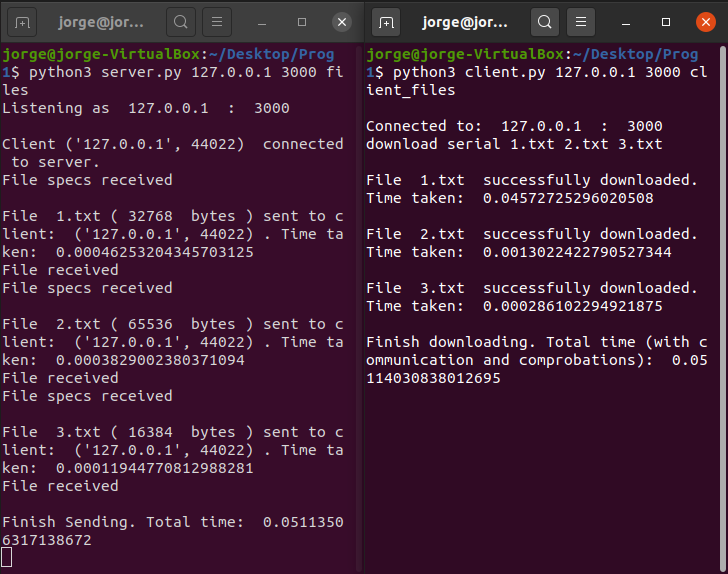


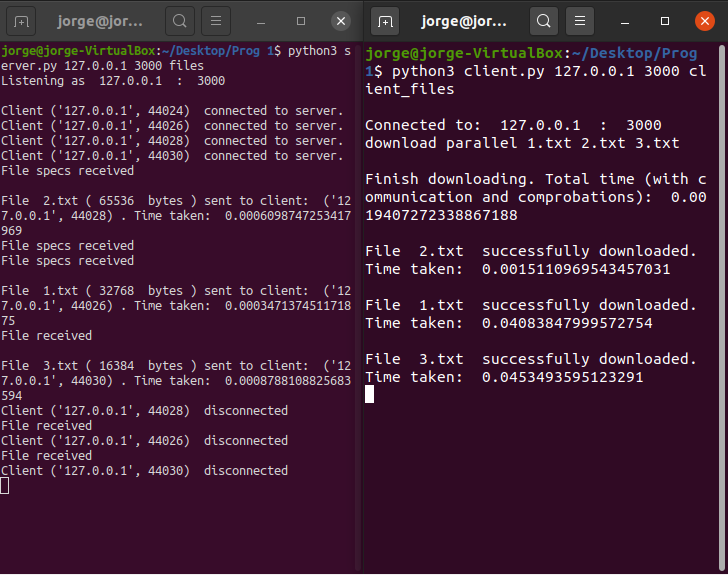


* One File download: client download a file from the server.

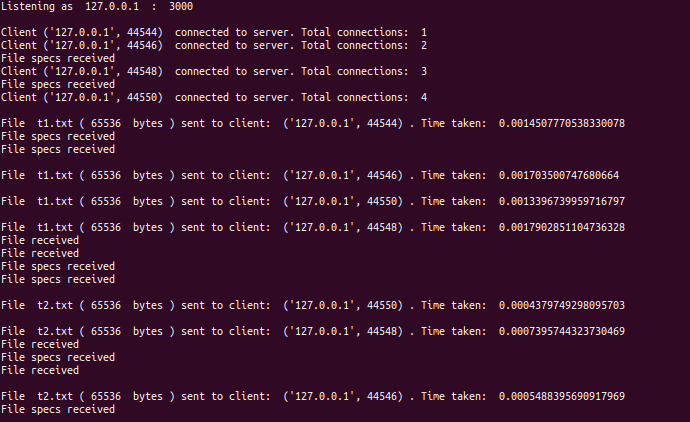


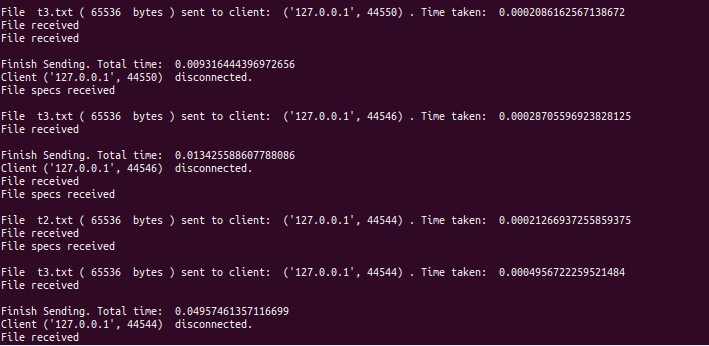
* Multiple File download: client download several files either in serial or parallel. For the parallel download, each file creates a new connection to the server which gets disconnected after the file transfer.





1. Four clients are connected to the server and downloading files at the same time:





1. Measuring the transfer speed when varying the number of concurrent clients:

A bash script has been created to connect N concurrent clients to the server and download in parallel 10 files each (each file creates a different thread). The size of the files is 64 KB each and the experiment has been repeated thrice for each number of clients (2, 4, 8 and 16). The results can be seen in the following graph:

Gráfico, Gráfico de líneas

Descripción generada automáticamente

Therefore, increasing the number of concurrent clients downloading at the same time, decreases the transfer speed of the downloads.

1. Measuring the transfer speed when varying the file size:

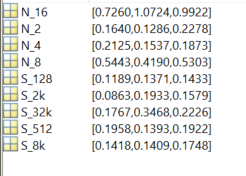
A bash script has been created to connect 4 concurrent clients to the server and download in parallel 10 files each (each file creates a different thread). The size of the files varies (128, 512, 2k, 8k and 32k Bytes) and the experiment has been repeated thrice for each file size. The results can be seen in the following graph:



Although it may seem weird that the transfer speed increases with the file size, it is just a misconception.

Actually, the time measured in the graph is the one needed to receive all the data in the clients. However, this is not the transfer time but the time it took the client to read the data from the “buffer”. The same happens at the server, the times printed are not the transfer times, but the times to write the data in the “buffer”. Therefore, as all the sizes are relatively small, the time taken to read the data is approximately the same for all the files and, when these times get scaled by their file size, it looks as if the speed in increasing.

\*\* As it was not compulsory to create a log for the clients, the output times (in seconds) for both graphs are the following:

 N\_xx: Time taken for the three experiments with xx concurrent clients.

S\_xxx: Time taken for the three experiments with xxx Bytes files.