

# TD5 - Networks and protocols

## TI602

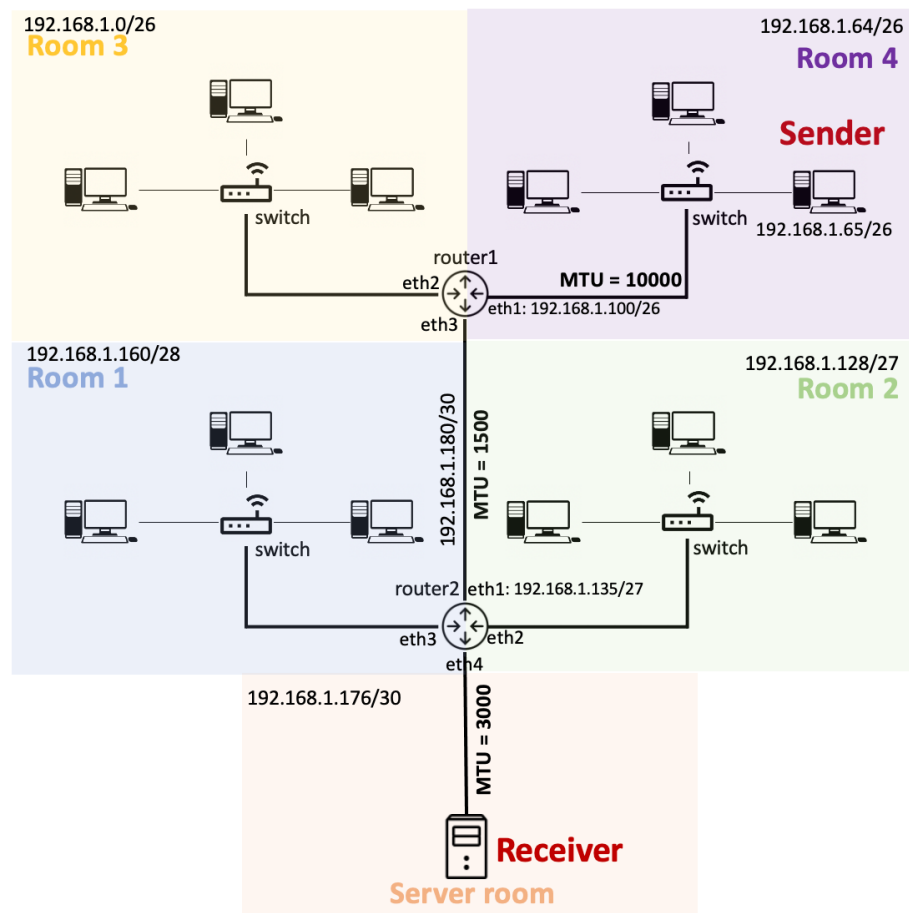
In this tutorial, we are interested in the protocols ARP and UDP.

### Case study

We are interested in an online shooter video game platform where players control characters and compete in 3D space. Characters can interact with the environment and with other characters by attacking each other. This platform runs on a local network infrastructure, each player accesses the game on a machine on which his client application is running. A central server manages the synchronization of the game's views and actions: a client application sends data to the servers at a fairly high frequency on the position of the player's character, movements and the actions executed by the latter (in particular on the other players characters). The server after receiving this information, made a computation based on the actions and movements done (received) and distributes the updates to the other machines so that everyone can see everyone's state almost in real time locally on their machine. In this kind of games, the speed of the messages as well as the optimization of resources is very important in order to guarantee an optimal gaming experience (maximum data transmitted correctly and low latency for a view close to real time).

### Exercise

In the case of our gaming platform, client applications send update messages (character position, movements, actions, etc.) at each 20ms. The messages coming from this application contains 4000 bytes. As our application prioritizes low latency and fluidity in the user experience, UDP is used to ensure the transport of messages.



1. Why UDP is more suitable for our application than TCP?
2. UDP does not guarantee messages delivery, what happens if a message is lost (at user and application level)? is it critical in our case?
3. Describe in detail the process (different steps) of sending an update message from the *sender (client application)* to the *receiver (server application)* at the 4 bottom layers: Transport (UDP), Network (IP and ARP), Data-link (MAC, CSMA/CD) and Physical (encoding/decoding).
4. Decode the ethernet frames bellow (without preamble, delimiter and FCS) and tell, for the ethernet header and the network layer packet inside the frame: what the different fields are, their values, the nature of the protocols used.
5. What is the situation of these frames in the previous process (of question 3).

<p style="text-align: center;"><b>Frame 1</b></p> <p style="text-align: center;">FF FF FF FF FF 08 00 20 02 45 9E 08 06 00 01 08 00 06 04 00 01 08 00 20 02 45 9E C0 A8 01 41 00 00 00 00 00 00 C0 A8 01 64</p>
<p style="text-align: center;"><b>Frame 2</b></p> <p style="text-align: center;">08 00 20 02 45 9E 08 00 20 07 0B 94 08 06 00 01 08 00 06 04 00 02 08 00 20 07 0B 94 C0 A8 01 64 08 00 20 02 45 9E C0 A8 01 41</p>
<p style="text-align: center;"><b>Frame 3</b></p> <p style="text-align: center;">08 00 20 07 0B 94 08 00 20 02 45 9E 08 00 45 00 00 1D 7B BD 00 00 80 11 3A E5 C0 A8 01 41 C0 A8 01 B1 23 82 23 83 00 09 33 A9 01 01</p>