



## Meta 2 - Sistema de Messaging

PL2 / Prof. Vasco Pereira

Dário Félix - Nº 2018275530

Vadilson Jacob - Nº 2015270539

Coimbra, 25 de abril de 2021

### ❖ Compilação e Execução

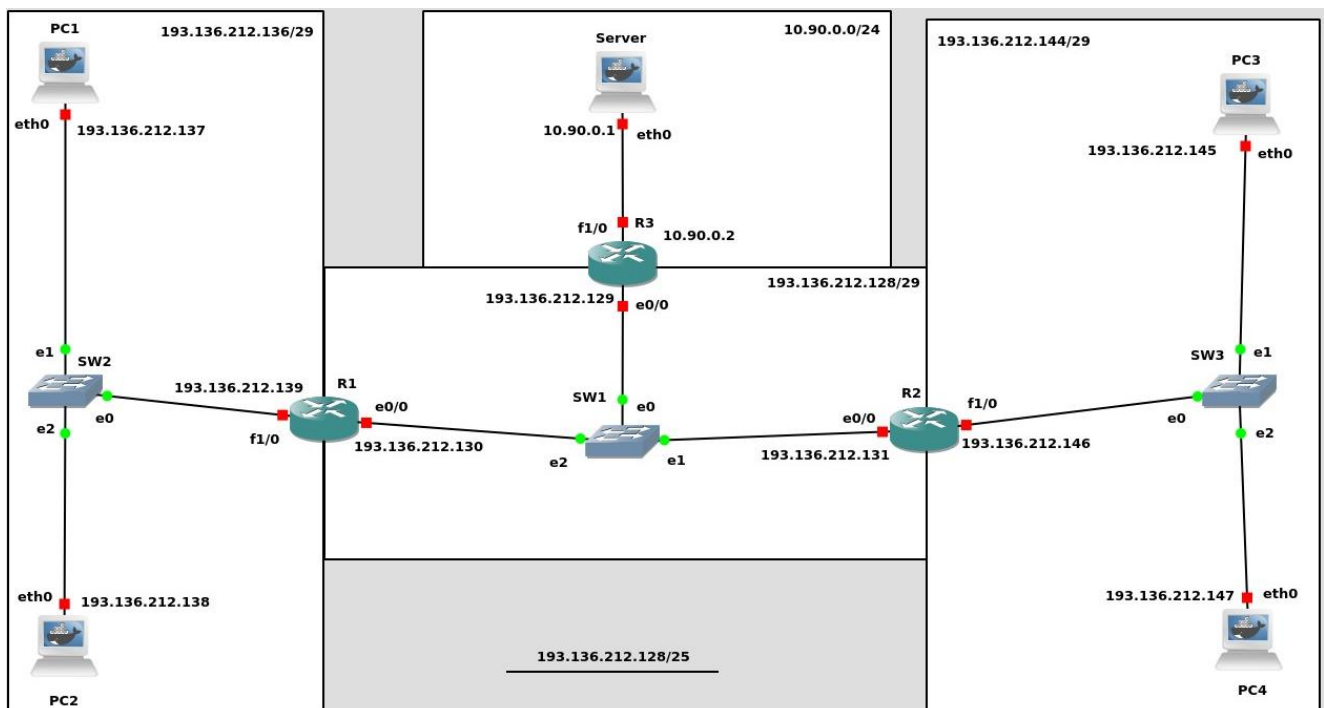
make

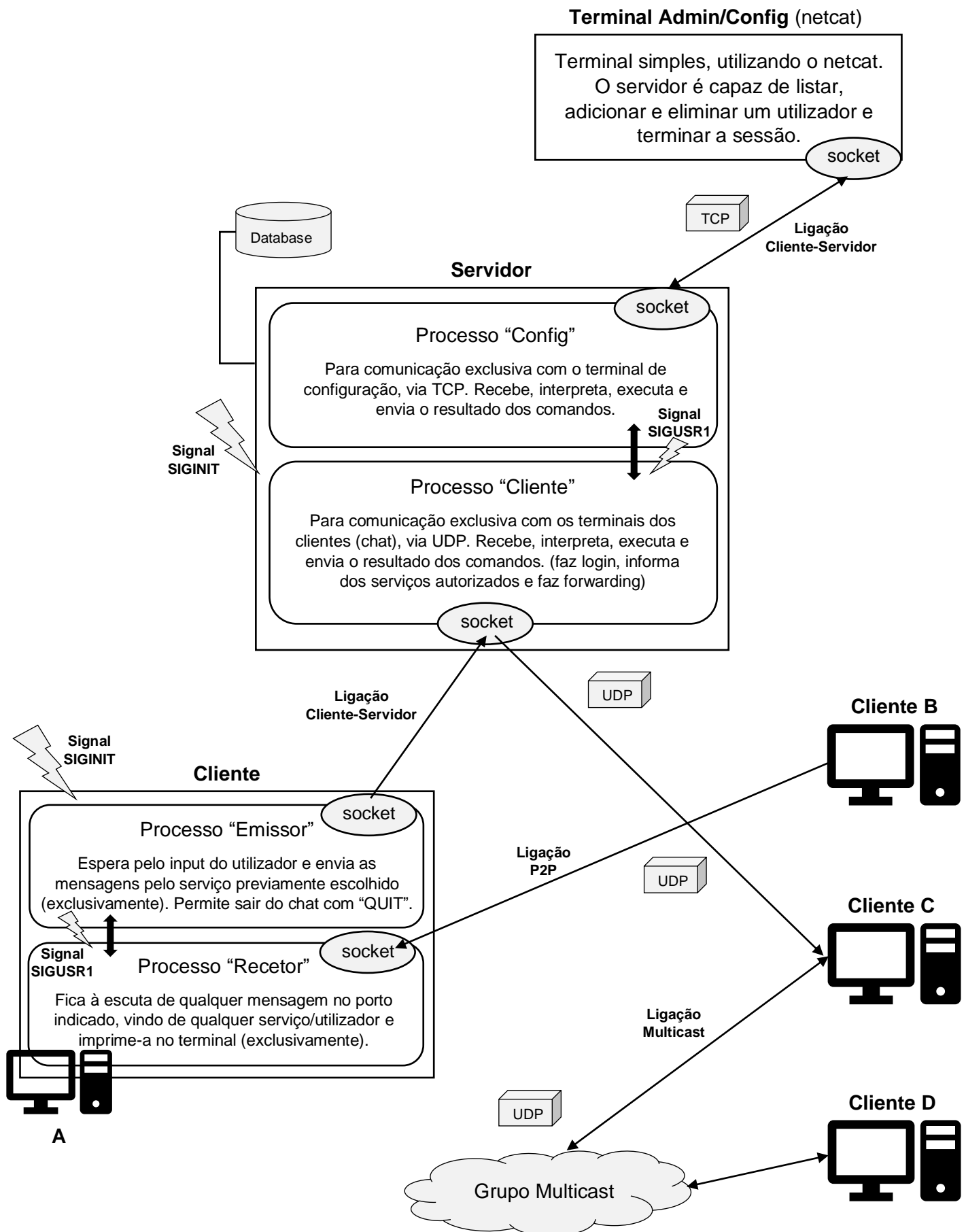
./servidor.out {porto clientes} {porto config} {ficheiro de registos}

./cliente.out {ip servidor} {porto clientes servidor}

nc -v {ip servidor} {porto config servidor}

### ❖ Esquema GNS3



❖ **Arquitetura do Software**

## ❖ Modo de Funcionamento (opções tomadas)

Para a comunicação UDP entre o Servidor e os Clientes, e os Clientes entre si, foi desenvolvido um protocolo com a seguinte estrutura:

chave:valor { ; chave:valor }

No total foram desenvolvidas as seguintes mensagens/pacotes predefinidos:

type:login ; username:\*\*\*\* ; password:\*\*\*\*

type:status ; logged:0 ; msg:\*\*\*\*

type:erro ; msg:\*\*\*\*

type:get\_servico ; username:\*\*\*\*

type:servico\_autorizado ; p2p:1 ; cliente-servidor:0 ; multicast:1

type:servico\_escolhido ; servico:p2p ; id:\*\*\*\*

type:p2p ; ip:\*\*\*\*

type:multicast ; ip:\*\*\*\*

type:enviar\_msg ; id\_fonte:\*\*\*\* ; id\_destino:\*\*\*\* ; msg:\*\*\*\*

Para a comunicação TCP entre o Servidor e o Terminal de Configuração, foram implementados os comandos descritos no enunciado:

LIST

ADD <user-id> <ip> <password> <cliente-servidor> <p2p> <grupo>

DEL <user-id>

QUIT

Para guardar os registos dos Clientes no lado do Servidor, foram usados ficheiros em bytes: esta opção permitiu guardar as estruturas (registo de cada utilizador) de uma forma rápida e consistente, mapeando diretamente a estrutura em memória para o ficheiro. Como o tamanho da estrutura é fixa, no momento da leitura, pode-se saltar registo em registo dentro do ficheiro. Assim os dados não são guardados em memória, apenas são consultados ou editados quando pedido. Isto permite partilhar os dados pelos 2 processos do Servidor e manter os dados persistentes.

O Servidor tem em memória um array de estruturas (ID do grupo + IP multicast do grupo), com um tamanho estático. Estes dados são apenas mantidos em execução, ou seja, a lista é inicializada sempre que o Servidor recomeça. Quando um Cliente solicitar um grupo multicast, o Servidor verifica se já se encontra registado. Caso se verifique isso, envia o IP multicast associado, caso contrário cria um novo grupo e guarda esse novo registo (ID + IP) no array.

Tanto o Servidor como o Cliente tratam o sinal siginit, permitindo terminar todos os processos de forma ordeira, fechando os sockets (file descriptor), o ficheiro se estiver em utilização (file pointer), e também liberta a memória dinâmica alocada.