

Lab02 - Construindo aplicações distribuídas usando sockets TCP/UDP

Giovanna Borges Bottino - 170011267
Felipe Boccardi Silva Agustini - 180119818

Introdução

O objetivo desse experimento é compreender as características inerentes à construção de aplicações distribuídas, incluindo passagem de parâmetros, envolvendo módulos cliente e servidor usando sockets TCP/UDP.



Características

- Para atender os objetivos do experimento, nós optamos por uma implementação TCP contendo protocolo em modo simplex, diálogo ponto-a-ponto, stop-and-wait e orientado à conexão.
- Decidimos pelo protocolo TCP por ele ser considerado o mais confiável. Enquanto o UDP não garante a entrega, o TCP garante a entrega e a integridade dos dados pacote

Características

- a solução de um servidor nossa maior dificuldade foi o envio de um array de float. Conseguimos solucionar através do envio do endereço de memória do array com o tamanho do tipo float multiplicado pelo tamanho do array.
- Já na solução com múltiplos servidores nossa maior dificuldade foi criar e manter a conexão paralela. Usamos de processos filhos para isso, para quantidade de servidores escolhidos diferentes existe um processo filho relacionado. Por esse motivo, na solução de múltiplos servidores também é possível usar apenas um servidor

LAB1VSLAB2

RPCgen vs Sockets

RPC é o serviço e protocolo que permite o código ser acionado para execução por um aplicativo remoto. Uma implementação de RPC pode ser feita com algum transporte de rede. Já o socket é apenas uma abstração de programação que pode enviar e receber dados com outro dispositivo através de um transporte de rede específico, no nosso caso TCP. Em comparação ao trabalho 1, diferente do RPCgen que precisa ter um arquivo de definição de interface, sockets não precisam e isso facilita no envio de mensagens. Trabalhar com sockets deixa o programador com mais liberdade para achar soluções.



1-10 sockets

| Número de workers | 1 | 2 | 4 | 6 | 8 | 10 |
|-------------------|-------|-------|-------|-------|-------|-------|
| Tempo (s) | 0.002 | 0.003 | 0.003 | 0.003 | 0.004 | 0.004 |

Desvio padrão: 0.000687

Média de tempo de resposta: 0.0032 s

Faixa de erro ± 0.001 s

Podemos ver um aumento do tempo de resposta conforme o aumento do número de workers, porém é um aumento esperado e nada fora do normal.