

Programação com Sockets

Marcos Cordeiro Fleury

Departamento de Ciência da Computação,
Universidade de Brasília
Email: marcoscfleury@aluno.unb.br

Resumo—Relatório do trabalho de programação com sockets em Python. O objetivo do trabalho é criar um algoritmo que garanta entrega confiável e em ordem dos dados, controle de fluxo e pipeline ao UDP.

I. RELATÓRIO

Para garantir todos os requisitos pedidos para este trabalho, os códigos UDPClient e UDPServer foram alterados para funcionarem do método go-back-N. Para tal, criou-se uma janela que consiste do número de mensagens que é enviado antes de começar a receber confirmações. Além disso, há um timeout que é o tempo máximo que se esperará antes de se aumentar a janela e também voltar a transmitir as mensagens. Para finalizar o cliente, é verificado se a primeira vez que se tenta receber o acknowledge, se sim a janela é reduzida. Já para o servidor é guardada qual deve ser a próxima mensagem recebida para que não se guarde mensagens que já foram recebidas. No código foi feito um teste em caso de perda de mensagem, a mensagem 5 no código, e um exemplo de atraso maior para verificar como o algoritmo se comporta para estabilizar o fluxo de mensagem. Para finalizar, foi feito um exemplo nos códigos UDPClient2 e UDPServer2 para analisar qual a vazão máxima nos códigos propostos. Na máquina onde ele foi rodado foi alcançada a vazão de 19.000 mensagens por segundo. Os quatro códigos foram enviados em anexo junto com o presente PDF.