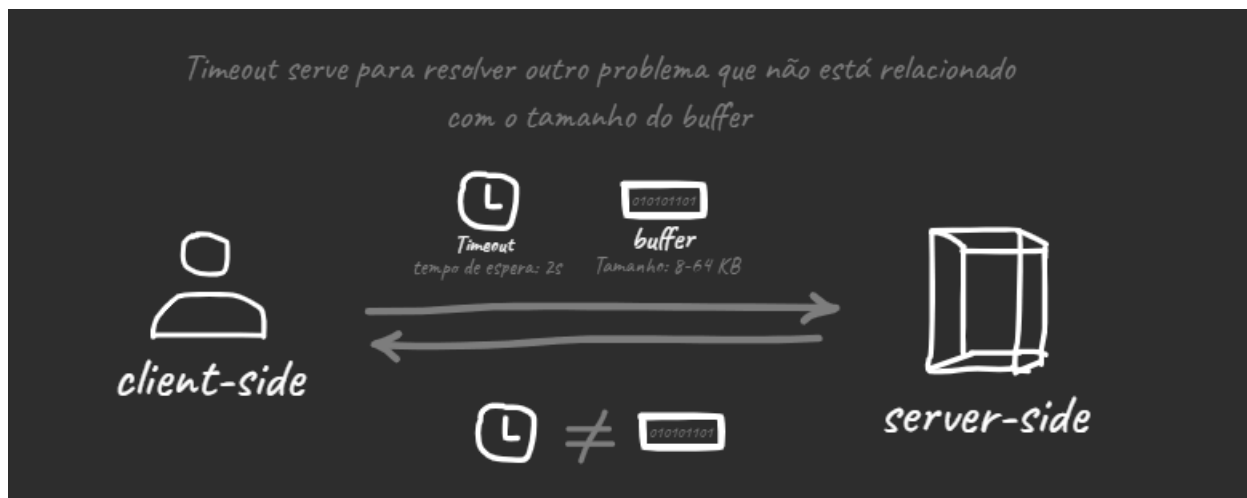


Lista de Exercícios Avaliativa sobre Sockets UDP e TCP

Para entender o motivo dos programas que utilizam tanto timeout como também o envio do tamanho do arquivo tem dificuldades ao transferir arquivos grandes, é importante entender como cada uma dessas soluções funciona, a implementação de timeout serve para evitar que o programa fique esperando sem um tempo definido por dados depois do final de uma transmissão. Ele define o tempo limite para operações de leitura ou escrita sobre o socket e isso evita que o sistema trave por tempo indefinido se o processo de comunicação não ser feito corretamente até o final.

Já o envio do tamanho do arquivo permite que o receptor saiba a quantidade de dados que será transmitida ainda antes do início da operação, dessa forma o receptor sabe quando termina a operação.

No caso do timeout, o problema de transferência de grandes arquivos contorna essa solução da seguinte forma: O timeout só controla o tempo de espera porém não aumenta o limite do tamanho do buffer. Se você tentar transferir um arquivo maior do que o buffer permite, o problema continuará existindo mesmo que o timeout esteja configurado corretamente.



Em relação à ideia de enviar o tamanho, a biblioteca socket tem uma limitação que é o tamanho do buffer e a quantidade de dados depende do tamanho dele. Se o programa tentar transmitir um arquivo inteiro ou grandes blocos de dados de uma só vez, apenas uma parte será enviada corretamente. O restante do arquivo pode ser descartado ou perdido.

Fazer a transferência de grandes volumes de dados em uma única operação pode sobrecarregar o sistema, tanto no lado do cliente quanto no do servidor. Isso aumenta a chance de perder pacotes. Embora o timeout e envio do tamanho do arquivo dê certo na transferências de arquivos menores, as transferências de

grande escala exigem soluções mais avançadas. Dividir o arquivo em pedaços menores e ajustar os tempos de espera. Dessa forma, a comunicação fica muito mais confiável e o risco de perder dados no caminho é menor do que enviar tudo em grandes blocos.

