

Relatório - Socket API (UDP)

Pontifícia Universidade Católica do Paraná

Escola Politécnica - Bacharelado em Engenharia de Software

Disciplina: Conectividade de Sistemas Ciberfísicos

Atividade Prática - Socket API (UDP)

Exercício 1 - Comunicação UDP com Cliente e Servidor

- a) O programa servidor é executado solicitando a porta via entrada padrão. O comando `netstat -a` permite visualizar a porta em escuta.
- b) Em seguida, executa-se o cliente, que envia uma mensagem ao IP e porta informados.
- c) Ao encerrar o servidor e tentar enviar nova mensagem, o cliente continua funcionando (UDP não exige conexão prévia).

Questão 1 - Análise do Comportamento do Cliente e Servidor

- 1.1) O cliente NÃO indica erro se o servidor não estiver iniciado. A mensagem é enviada, pois o UDP é não orientado à conexão.
- 1.2) O cliente NÃO é notificado se enviar mensagem para um servidor inexistente. O pacote é transmitido, mas não há confirmação de recebimento.
- 1.3) O cliente NÃO é notificado se o servidor for encerrado após o envio da primeira mensagem. UDP não mantém estado da conexão.
- 1.4) O servidor NÃO precisa de threads para múltiplos clientes. Como ele trata mensagens uma a uma, o uso de threads seria necessário apenas se desejássemos processar mensagens em paralelo.

Exercício 2 - Argumento de IP no Bind

O código do servidor foi adaptado para aceitar o IP via entrada padrão, além da porta. Isso permite instanciar o servidor em IPs diferentes da mesma máquina (ex: 127.0.0.2, 127.0.0.3).

Dois servidores podem ser lançados na mesma porta se os IPs forem distintos.

Questão 2 - Análise de Envio e Broadcast

- 2.1) Sim, o cliente pode enviar mensagens para servidores diferentes com o mesmo socket. Basta alterar o IP e porta de destino a cada envio.
- 2.2) O cliente NÃO consegue enviar mensagens ao endereço 127.255.255.255, pois o loopback não permite broadcast. O broadcast válido geralmente é feito na rede local (ex: 192.168.0.255).

Relatório - Socket API (UDP)

2.3) O BROADCAST é intrinsecamente não confiável. Não há confirmação de recebimento, pode haver colisões, perda de pacotes e congestionamento na rede. Isso inviabiliza seu uso em cenários que exigem garantias de entrega.