



CEA510 – Redes de Computadores II

TRABALHO PRÁTICO

O objetivo deste trabalho é praticar programação para comunicação em redes de computadores por meio de Sockets.

1) ORIENTAÇÕES GERAIS

O trabalho deverá ser realizado com **até 2 pessoas**. É permitido (e recomendado) que as duplas troquem experiências sobre as dificuldades do trabalho.

A linguagem de programação e o sistema operacional utilizados são de livre escolha da dupla.

Apresentação e entrega final do relatório: **até 12/12/2018**

Valor: 23 pontos

2) DESENVOLVIMENTO

A dupla deverá escolher uma das duas opções e desenvolver seu trabalho.

- A porta UDP padrão do servidor deve ser o ano de nascimento do integrante mais novo da dupla (ex: 1993)
- O cliente e o servidor devem ter tratamento de erros e exceções. Todos os detalhes, como configuração, mensagens e o que não estiver especificado aqui devem ser definidos e documentados no relatório. Interface, organização, criatividade são valorizadas! Capriche!
- Deem atenção para a qualidade do código produzido: códigos bem comentados, uso adequado de funções, tratamento de erro (emitindo mensagem de erro se necessário), tratamento de exceções (valores fora de faixa, resposta inválida, etc.).

OPÇÃO 1 – Jogo utilizando UDP

Esta opção consiste na implementação de um programa cliente/servidor simples, utilizando a interface de sockets para criar uma versão simplificada de um Jogo qualquer (cartas, tabuleiro, ...). Novamente, a intenção deste trabalho é implementar um serviço de transmissão confiável, utilizando o **UDP**.

Para coordenar o jogo deverá haver um computador que será o servidor central. Os clientes devem se comunicar somente com o servidor. Nele todos os demais computadores que quiserem jogar (clientes) deverão se inscrever através do envio de um apelido e de seu respectivo endereço IP. O servidor armazenará uma lista com os nomes dos usuários. Os computadores clientes deverão acessar o servidor para se inscrever no jogo.

No momento de inscrição no servidor, os clientes também deverão “baixar”/exibir a lista dos jogadores que já estão disponíveis para o jogo. Além disso, a aplicação cliente deverá ter um botão atualizar o status dos outros jogadores.

Regras

Um árbitro (servidor) mantém o controle do jogo. – A quantidade de participantes vai depender do jogo a ser desenvolvido. – Quando um dos participantes começa a partida, é exibida uma mensagem na tela dos demais participantes, informando o início do jogo. Assim, deve-se aguardar um tempo de X segundos para se iniciar a seção do jogo.

Adicionalmente, um timer de X minutos deve ser implementado, caso alguém “durma no jogo”. Além disso, uma mensagem com o vencedor(vencedores) deve ser enviada a cada participante.

A implementação deste jogo pode variar bastante: funcionalidades, interface gráfica sofisticada ou simples, versão para mobile, etc, etc, etc.

OPÇÃO 2 – Transmissão confiável de arquivo utilizando UDP (4 pontos extras)

A aplicação criada deverá funcionar da seguinte maneira:

1. Servidor:

- a. O servidor deve disponibilizar uma lista de todos os arquivos para o cliente que conectar
- b. Criar uma senha para somente o cliente com a senha poder enviar arquivos para o servidor

2. Cliente:

- a. Deve conectar ao servidor de arquivos, receber a lista de arquivos e poder baixar um arquivo.

O objetivo maior deste trabalho é implementar um serviço de transmissão confiável, utilizando o **UDP** como protocolo de transporte. Como esse protocolo não oferece um serviço confiável, será sua função implementar um protocolo de janela deslizante para garantir entrega confiável e em ordem. Soluções sem janela deslizante (só stop-and-wait) também serão aceitas, mas avaliadas com uma penalização de 15%.

O programa cliente solicitará o conteúdo de um arquivo, disponível no servidor de arquivos, implementado na parte 1 do trabalho. O cliente receberá o arquivo e o armazenará no disco.

Será necessário executar experimentos para fins de avaliação do desempenho do protocolo, com e sem falhas. O número de pacotes retransmitidos também deve ser computado para fins de comparação.

Para fins de avaliação você deverá mensurar a largura de banda entre o cliente e o servidor, além do atraso (RTT). Você poderá executar testes variando estes parâmetros. Seu protocolo deve ser avaliado com diferentes tamanhos de janelas, mantendo-se um atraso no link intermediário. Os valores adequados de variação do tamanho da janela devem ser encontrados pelos programadores dos protocolos em cada caso. Para valores menores, pode ser recomendável reduzir o tamanho do arquivo para evitar esperas muito longas.

Além de mostrar o throughput oferecido pelo protocolo, devem também ser apresentados os resultados de execução, simulando uma situação quando o link entre os switches tiver uma taxa de perda de 5% (loss=5).

Para esta parte, a dupla deverá apresentar um relatório descrevendo o funcionamento do seu protocolo e comparando os mecanismos de transmissão no envio de um arquivo de + - 1 MB.

Definição do protocolo:

Neste trabalho, como não teremos TCP para garantir a ordem, vocês serão responsáveis por criar um protocolo confiável sobre UDP usando janela deslizante. Para isso, vocês devem definir quais serão os campos do cabeçalho do seu pacote (que será enviado dentro do pacote UDP), que encapsulará os dados do arquivo.

O protocolo nesse caso será unidirecional, isto é, ele deve ser construído para enviar dados apenas na direção do servidor para o cliente. Mensagens de confirmação podem ser necessárias em diferentes momentos e uma técnica de detecção de erros também deverá ser utilizada, pois erros poderão ocorrer durante os testes.

Recomendo que os cabeçalhos sejam de tamanho fixo, por ser mais simples...

O principal desafio deste trabalho é o tratamento de temporizações (timeouts).

Detalhes de operação:

As mensagens UDP não devem ultrapassar o MTU da rede, isto é, cada mensagem UDP não deve transportar mais que 1460 bytes de dados (isso, mas 20 bytes do cabeçalho UDP e outros 20 do cabeçalho IP completarão os 1500 bytes de dados em um quadro Ethernet).

O relatório a ser apresentado deve conter detalhes como o formato dos cabeçalhos das mensagens, o processo de transmissão do arquivo e de identificação do início e fim da conexão, determinação dos intervalos de temporização, etc.

Informações úteis:

As métricas de comparação entre os mecanismos de transmissão devem ser: tempo total de transmissão e número de pacotes retransmitidos.

Certifique-se de gerar a assinatura md5 do arquivo original e do arquivo escrito pelo receptor para se certificar de que seu protocolo funciona corretamente.

3)TESTES

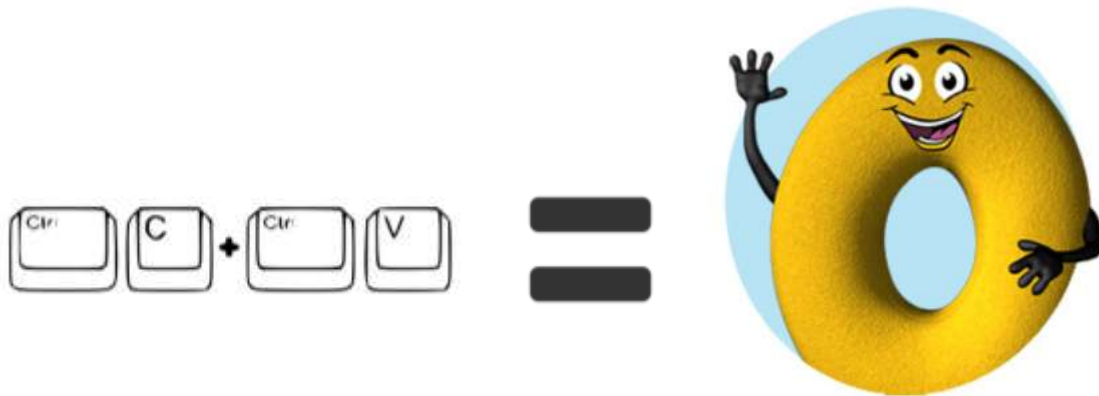
Descrever de forma detalhada os testes realizados, resultados obtidos e análise dos resultados; Adicionalmente, deverá ser incluído prints de telas do Wireshark confirmando os testes.

4) CONCLUSÃO E BIBLIOGRAFIA

- Conclusão do trabalho enfatizando os principais resultados e observações.
- Bibliografia pesquisada (notas de aula, livros, artigos).

5) ENTREGA E APRESENTAÇÃO

- Deve ser entregue via Moodle um arquivo.zip com o seu sistema, um arquivo leiamme com informações sobre a compilação desse programa, além do relatório final, no formato do SBC (disponibilizado no site).
- Todos os integrantes devem estar aptos a responder dúvidas sobre a implementação do trabalho.



Por fim, mesmo que você não consiga implementar o que se pede, qualquer esforço será recompensado!!!

Bom trabalho!