

1. **(1,5 pontos)** Um processo que executa no hospedeiro **A** está associado à porta **p**, e um processo que executa no hospedeiro **B** está associado à porta **q**. É possível que datagramas UDP oriundos desses dois processos sejam entregues no mesmo *socket*, que tem número de porta **r**, existente no hospedeiro **C**? Explique.

**Resposta:**

Sim, ambos os segmentos serão direcionados para o mesmo *socket* no hospedeiro C. Para cada segmento recebido, na interface do *socket*, o sistema operacional disponibilizará para o processo o endereço IP de forma a determinar as origens dos segmentos individuais.

2. **(1,5 ponto)** Descreva por que um desenvolvedor de uma aplicação distribuída escolhe executar sua aplicação sobre UDP em vez de executá-la sobre TCP.

**Resposta:**

A escolha está baseada nas seguintes características do UDP:

- **Não há estabelecimento de conexão:** No TCP existe o *three way handshake* para que a transferência de dados tenha seu início e esta etapa produz atraso.
- **Não há estado de conexão:** O TCP mantém o estado de conexão. Esse estado inclui *buffers* de envio e recepção, parâmetros de controle de congestionamento e parâmetros numéricos de sequência e de reconhecimento, desta forma esta manutenção do estado tem custos de processamento e espaço alocado.
- **Pequeno overhead no cabeçalho do pacote:** O segmento TCP tem 20 bytes, enquanto o do UDP tem somente 8 bytes.
- **Taxa de envio não regulada:** No TCP existem os mecanismos de controle de congestionamento e de fluxo que regulam a taxa de envio de segmentos, sem a interferência da aplicação transmissora. Este controle na taxa de envio de segmentos pode ser crítica para aplicações que requisitam uma taxa de transmissão mínima e que são tolerantes a perda de pacotes até um certo nível.

3. **(1,5 pontos)** É possível que uma aplicação que executa sobre UDP possa se beneficiar da transferência de dados confiável? Se sua resposta é afirmativa, explique como isso é feito.

**Resposta:**

Sim, desde que a transferência de dados confiável seja implementada na própria aplicação, caso contrário não seria possível, pois o UDP não implementa a transferência de dados confiável.

4. **(1,5 pontos)** Considere a transferência do hospedeiro A para o hospedeiro B, de um arquivo (muito grande) com  $L$  bytes. Assuma que o Tamanho Máximo do Segmento (*Maximum Segment Size* - MSS) é de 1460 bytes. Qual o máximo valor de  $L$  de modo a que os faixa de números de sequência não seja esgotada? Lembre-se que o campo para número de sequência no TCP tem quatro bytes.

**Resposta:**

Os números de sequência no TCP não são incrementados de um a cada segmento e sim do número de bytes dos dados enviados. Assim o tamanho do MSS é irrelevante – o tamanho máximo do arquivo que pode ser enviado de A para B é simplesmente o número de bytes que podem ser representados por  $2^{32}$ . Isto é aproximadamente 4 Gbytes.

5. **(1,0 ponto cada item)** Responda verdadeiro ou falso, EXPLICANDO sua escolha:

- a) Suponha que o hospedeiro A esteja enviando para o hospedeiro B um arquivo grande por meio de uma conexão TCP. O número de bytes não reconhecidos que o hospedeiro A envia não pode exceder o tamanho do *buffer* de recepção do hospedeiro B.

**Resposta:**

**Verdadeiro.** Isto é garantido pelo mecanismo de controle de fluxo em relação à janela de recepção, que sempre menor ou igual ao buffer de recepção em B. Portanto o hospedeiro A nunca terá um número de bytes enviados e não reconhecidos que seja superior ao buffer de recepção do hospedeiro B.

- b) Suponha que o hospedeiro A esteja enviando para o hospedeiro B um arquivo grande por meio de uma conexão TCP. Se o número de sequência para um segmento transmitido nessa conexão é  $m$ , então o número de sequência para o segmento subsequente é  $m+1$ .

**Resposta:**

**Falso.** O próximo número de segmento será  $m$  acrescido da quantidade de bytes de dados da aplicação enviados no segmento anterior.

- c) Imagine que o hospedeiro A envie ao hospedeiro B, por uma conexão TCP, um segmento contendo 4 bytes de dados e com número de sequência 38. Nesse mesmo segmento, o número contido no campo de confirmação é obrigatoriamente 42.

**Resposta:**

**Falso.** O valor inicial para os números de sequência e de confirmação, nos dois extremos da conexão são definidos durante o estabelecimento

da conexão TCP (durante o *three way handshake*), e cada uma das partes é livre para escolher o número de seqüência inicial que desejar.

- d) Considere o controle de congestionamento no TCP. Quando um temporizador expira no transmissor, o limiar (*threshold*) é ajustado para a metade do seu valor anterior.

**Resposta:**

Falso. O limiar é ajustado à metade do tamanho da janela de congestionamento quando da ocorrência da expiração do temporizador.