



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**

**Disciplina: Redes de Computadores I**

**Gabarito AP2 - 2º semestre de 2014.**

**Aluno:** \_\_\_\_\_

**Assinatura:** \_\_\_\_\_

Observações:

1. Prova sem consulta e sem uso de máquina de calcular.
2. Use caneta para preencher o seu nome e assinar nas folhas de questões e nas folhas de respostas.
3. Você pode usar lápis para responder as questões.
4. Ao final da prova devolva as folhas de questões e as de respostas.
5. Todas as respostas devem ser transcritas nas folhas de respostas. As respostas nas folhas de questões não serão corrigidas.

1. O UDP (*User Datagram Protocol*) também é chamado de protocolo de transporte mínimo. Por quê? Ele é realmente necessário na Internet? Justifique suas respostas e cite dois protocolos da camada de aplicação que o utilizam. (1,0 ponto)

Resposta: O UDP é chamado de protocolo de transporte mínimo porque só oferece os serviços mínimos da camada de transporte que são: multiplexação, demultiplexação e verificação de integridade (0,4 pontos). O UDP é necessário por possuir menor latência se comparado ao TCP, não manter estados no transmissor e receptor e por permitir que um transmissor possa enviar dados tão rápido quanto desejado e possível. Isso porque o UDP não exige estabelecimento de conexão, não emprega mecanismos de controle de fluxo e congestionamento e seu cabeçalho tem tamanho reduzido (0,4 pontos). São exemplos de protocolos da camada de aplicação que o utilizam o DNS (*Domain Name System*), o NFS (*Network File System*), o SNMP (*Simple Management Network Protocol*), entre outros (0,1 pontos por cada protocolo citado corretamente – até 0,2 pontos).

2. Cite a solução adotada pelas técnicas *Go-Back-N* e retransmissão seletiva para aumentar a utilização do meio de transmissão quando comparadas aos protocolos para-e-espera (*stop and wait*). Diferencie essas duas técnicas (1,0 ponto)

Resposta: A solução é permitir que um transmissor envie um conjunto de segmentos consecutivamente sem que antes seja necessário receber a confirmação de recepção dos segmentos anteriores dentro deste conjunto. Essa

técnica é chamada de paralelismo (*pipelining*) (0,2 pontos). A técnica *Go-Back-N* emprega ACKs cumulativos e o transmissor possui apenas um temporizador para o segmento mais antigo ainda não reconhecido. Dessa forma, se o temporizador estourar, todos os pacotes ainda não reconhecidos devem ser retransmitidos (0,4 pontos). Por outro lado, a retransmissão seletiva reconhece os pacotes individualmente, ou seja, para cada pacote recebido, o receptor envia um ACK. Além disso, o transmissor possui um temporizador para cada pacote ainda não reconhecido, ou seja, se o temporizador estourar é necessário retransmitir apenas o pacote correspondente (0,4 pontos).

3. Descreva sucintamente o mecanismo de abertura de conexão usado pelo TCP. (1,0 ponto)

Resposta: O mecanismo de abertura de conexão do TCP é composto por 3 etapas, por isso a denominação de *three-way handshake* (0,1 ponto). Na primeira etapa, o cliente TCP envia para o servidor TCP um segmento de controle SYN, especificando o número de sequência inicial no sentido cliente-servidor, entre outros parâmetros de configuração. Não são enviados dados neste segmento (0,3 pontos). Na segunda etapa, após receber o SYN, o servidor envia para o cliente um segmento de controle SYN+ACK sinalizando que aceita abrir uma conexão e que alocou espaço em seu *buffer*. Nesse segmento, é especificado, por exemplo, o número de sequência inicial no sentido servidor-cliente (0,3 pontos). Na terceira etapa, após receber o SYN+ACK, o cliente envia um segmento de controle ACK, confirmando, assim, a abertura da conexão. Este último segmento pode conter dados (0,3 pontos).

4. Descreva e diferencie os mecanismos de controle de fluxo e de congestionamento do TCP. (2,0 pontos)

Resposta: O objetivo do controle de fluxo é não sobrecarregar o receptor com mais dados do que ele pode receber. Por outro lado, o controle de congestionamento tem como objetivo inferir um congestionamento e não sobrecarregar ainda mais a rede nessa situação. Essa é a principal diferença entre os dois mecanismos (0,4 pontos). O controle de fluxo funciona da seguinte forma. O receptor anuncia o espaço livre em seu buffer para o transmissor através do campo janela de recepção (*RcvWindow*) presente no cabeçalho de cada segmento enviado. Ao receber essa informação, o transmissor limita sua janela de transmissão, ou seja, a quantidade de dados ainda não reconhecidos, ao tamanho informado no campo. Dessa forma, o receptor não é afogado pelo transmissor (0,8 pontos). O controle de congestionamento também limita a quantidade de dados ainda não reconhecidos, porém, de acordo com o nível de congestionamento da rede. Para tanto, perdas são inferidas através da recepção de três ACKs duplicados e pelo estouro do temporizador de espera de um ACK. Nos

dois casos, a medida tomada após o evento de perda é reduzir a janela de congestionamento pela metade ou para 1 segmento e redefinir o valor do `ssthresh`. Também são definidas diferentes formas de crescimento da janela em função do valor atual da janela de congestionamento: partida lenta (crescimento exponencial) e prevenção de congestionamento (crescimento linear) (0,8 pontos).

5. No TCP, um reconhecimento perdido, enviado pelo receptor, não necessariamente exige uma retransmissão por parte do transmissor. Diga se essa afirmação é VERDADEIRA ou FALSA e justifique sua resposta. (1,5 pontos)

Resposta: A afirmativa é VERDADEIRA (0,5 pontos) porque o TCP usa ACKs cumulativos, ou seja, o  $ACK(n)$  reconhece todos os bytes até e inclusive o byte  $n$  (0,5 pontos). Assim, se um  $ACK(m)$  for perdido, mas um  $ACK(n)$ , onde  $n > m$ , for recebido todos os bytes até  $n$  estarão reconhecidos (0,5 pontos).

6. Sobre o UDP e o TCP, considere as seguintes afirmações:
- Apenas o TCP estabelece uma conexão, transmite segmentos e encerra a conexão.
  - Tanto UDP quanto TCP fornecem à camada de aplicação um serviço não-confiável de transmissão de segmentos.
  - Tanto UDP quanto TCP utilizam números de sequência para garantir que os segmentos são entregues ordenadamente.
  - Nem UDP e nem TCP empregam reconhecimentos positivos (ACKs), retransmissão e temporizadores durante uma conexão.
  - Tanto UDP quanto TCP utilizam o Internet Protocol (IP) como protocolo subjacente para a entrega dos datagramas.

Diga se cada uma das afirmativas é VERDADEIRA ou FALSA. Justifique suas respostas. (1,5 pontos)

Resposta: A afirmativa “a” é VERDADEIRA (0,3 pontos). A afirmativa “b” é FALSA (0,1 pontos), o TCP fornece um serviço confiável para a camada de aplicação (0,2 pontos). A afirmativa “c” é FALSA (0,1 pontos), somente o TCP utiliza números de sequência e garantem entrega ordenada de segmentos (0,2 pontos). A afirmativa “d” é FALSA (0,1 pontos), o TCP emprega ACKs, retransmissão e temporizadores para garantir a entrega confiável dos segmentos (0,2 pontos). A afirmativa “e” é VERDADEIRA (0,3 pontos).

7. Sobre números de sequência e de reconhecimento do TCP:
- Suponha que um sistema final A envia para um sistema final B um arquivo de 10 GB em uma conexão TCP. Nessa situação, se o número de sequência para um segmento dessa conexão for  $n$ , então o número de sequência para o

segmento subsequente será necessariamente  $n+1$ . Diga se essa afirmativa é VERDADEIRA ou FALSA e justifique sua resposta explicando como são definidos os números de sequência e de reconhecimento (1,0 ponto).

Resposta: A afirmativa é FALSA (0,2 pontos), uma vez que os números de sequência são orientados a byte, ou seja, os números de sequência identificam o primeiro byte de dados de um segmento, uma vez que o TCP considera que dados a serem enviados são um fluxo (0,5 pontos). Por outro lado, os números de reconhecimento são definidos pelo número de sequência do próximo byte de dados esperado pelo receptor TCP (0,2 pontos).

- b. Um sistema final A envia para um sistema final B quatro segmentos, um logo após o outro, em uma mesma conexão TCP. Os dois primeiros têm tamanho igual a 50 bytes e o terceiro e quarto igual a 120 bytes. Assuma que o terceiro segmento é perdido e os demais são recebidos por B. Defina possíveis números de sequência dos segmentos enviados de A para B e dos reconhecimentos enviados de B para A (1,0 ponto).

Resposta: Supondo que o número de sequência do primeiro segmento enviado por A é  $n$ , os números de sequência dos segmentos seguintes serão  $n+50$ ,  $n+100$  e  $n+220$ . Por exemplo, se  $n = 0$ , logo o número de sequência dos segmentos enviados por A será 0, 50, 100 e 220. Ao receber o primeiro segmento, B enviará um ACK com número de reconhecimento  $n+50$ , que é o próximo byte esperado por ele. Ao receber o segundo segmento, B enviará um ACK com número de reconhecimento  $n+100$ , que é o próximo byte esperado por ele. Ao receber o quarto segmento, B enviará novamente um ACK com número de reconhecimento  $n+100$  que continua sendo o próximo byte esperado por B. Se  $n=0$ , por exemplo, os três ACKs enviados por B têm número de reconhecimento igual a 50, 100 e 100, respectivamente.