



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação

Disciplina: Redes de Computadores I

AP2 - 2º semestre de 2011.

Gabarito

1. Tanto o TCP como o UDP utilizam o complemento de 1 para suas somas de verificação (*checksums*). **(1,0 ponto cada item)**

1. Suponha que você tenha os seguintes bytes: 01011100 e 01010110. Qual o é complemento de um da soma desses dois bytes?

RESPOSTA:

Somando os dois bytes temos 10110010. o complemento de 1, é 01001101.

2. Suponha que você tenha os seguintes bytes: 11011010 e 00110110. Qual o é complemento de um da soma desses dois bytes?

RESPOSTA:

Somando os dois bytes temos 00010001; o complemento de 1, é 11101110.

3. Para os bytes do item 1 acima, dê um exemplo em que um bit é invertido em cada um dos dois bytes e, mesmo assim, o complemento de um não se altera.

RESPOSTA:

Primeiro byte = 01011110 ; segundo byte = 01010100.

2. Suponha que um servidor Web executa no hospedeiro C e escuta na porta 80. Suponha que esse servidor Web usa conexões persistentes e que está nesse momento, recebendo requisições de dois hospedeiros A e B diferentes. Explique: **(1,5 pontos cada item)**

1. Essas requisições são direcionadas para o mesmo *socket* no hospedeiro C?

RESPOSTA:

Não. Para cada conexão persistente, o servidor de Web cria um *socket* de conexão separado. Cada *socket* de conexão é identificado pela 4-tupla <endereço IP de origem, número de porta de origem, endereço IP de destino, número de porta de destino>. Quando hospedeiro C recebe um datagrama IP, ele examina esses quatro campos nos cabeçalhos das camadas de rede e de transporte (datagrama/segmento) para determinar para qual *socket* deve passar a carga útil (*payload*) do segmento TCP.

2. Se as requisições são direcionadas para *sockets* diferentes, podem esses diferentes *sockets* ter o mesmo número de porta 80?

RESPOSTA:

Sim. Como vimos na anterior, cada *socket* de conexão é identificado por quatro informações: <endereço IP origem, número da porta de origem, endereço IP destino e número da porta destino>. Quando o hospedeiro C recebe um datagrama IP, este examina estes quatro campos no datagrama/segmento para determinar para qual *socket* o conteúdo de informação do segmento TCP deve ser repassado. Assim, as requisições de A e B vão para *sockets* diferentes. O identificador para ambos *sockets* tem 80 como porta de destino, porém, os identificadores para estes *sockets* têm valores diferentes para o endereço IP de origem. Diferentemente do que ocorre no UDP, quando a camada de transporte repassa o conteúdo de informações de um segmento de TCP para o processo da aplicação, não especifica o IP de origem, pois isto é implicitamente especificado pelo identificador do *socket*.

3. Considere os protocolos Retorne a N e Repetição Seletiva. Suponha que o número de sequência usado por ambos tenha k bits. Qual é o maior tamanho de janela para que cada um desses protocolos, de modo que transferência de dados confiável não falhe. Explique sua resposta. **(2,0 pontos cada protocolo)**

RESPOSTA:

De modo a garantir que o protocolo Retorne a N não falhe, é necessário determinar o tamanho adequado da janela do transmissor, já que a janela do receptor nesse protocolo tem tamanho igual a 1. Considere em um primeiro momento, que o tamanho da janela do transmissor é 2^k . O transmissor então envia 2^k pacotes com números de sequência 0, 1, ..., $(2^k - 1)$. Considere que todos são recebidos e confirmados isoladamente (ou cumulativamente). O transmissor então, descarta as cópias dos pacotes confirmados, e envia novos pacotes com números de sequência 0, 1, ..., $(2^k - 1)$. Todos eles são recebidos e confirmados, porém todas as confirmações são perdidas. Nesse caso o pacote esperado na janela do receptor, tem número de sequência 0. Após o esgotamento da temporização, o transmissor, retransmite o pacote com número de sequência 0, isto é, uma duplicata, que é aceita pelo receptor como sendo um novo pacote: o protocolo falha! (note que todos os demais pacotes contidos na janela do transmissor, também são retransmitidos e recebidos pelo receptor como pacotes novos). Para garantir que o protocolo não falhe, é necessário que o tamanho da janela do transmissor, tenha no máximo, tamanho de $(2^k - 1)$. Por exemplo, se $k = 3$, os 8 números de sequência usados pelo protocolo, são: 0, 1, ..., 7. Para que o protocolo não falhe, o tamanho máximo da janela do transmissor é $(2^k - 1) = (2^3 - 1) = 7$. Nesse caso inicialmente o transmissor envia os pacotes 0, 1, 2, 3, 4, 5 e 6. Todos são recebidos e confirmados e o receptor passa a esperar o pacote

com número de sequência 7. O transmissor então envia os pacotes 7, 0, 1, 2, 3, 4, e 5, todos são recebidos e confirmados, mas todas as confirmações são perdidas. O receptor nesse momento aguarda o pacote 6. Após o esgotamento da temporização, o transmissor retransmite os pacotes 7, 0, 1, 2, 3, 4, e 5 que não são aceitos pelo receptor, pois ele aguarda o pacote com número de sequência 6.

De modo a garantir que o protocolo Repetição Seletiva não falhe, é necessário determinar a faixa de números de sequência suficiente grande, para que em qualquer instante de tempo, possa-se garantir que as janelas do transmissor e do receptor, contenham números de sequência, tais que duplicatas não sejam aceitas no receptor. Suponha que as janelas do transmissor e do receptor tenham tamanho $(2^k - 1)$. Considere então que o transmissor envia os pacotes com números de sequência de 0, 1, ... $(2^k - 2)$. A janela do receptor permite que pacotes de 0 até $(2^k - 2)$ sejam aceitos. Todos os pacotes são recebidos e confirmados pelo receptor, que passa a aguardar na sua janela os pacotes com números de sequência de $(2^k - 1)$, 0, 1, ..., $(2^k - 3)$. Porém, todas as confirmações enviadas pelo receptor são perdidas. Após o esgotamento da temporização, o transmissor retransmite o pacote com número de sequência 0, que é aceita pelo receptor: o protocolo falha!. A essência do problema é que, depois que o receptor avançou a janela, a nova faixa de números de sequência, tem intercessão com a faixa de números de sequência da janela anterior. Para garantir que o protocolo não falhe, é necessário que o tamanho da janela do receptor, tenha no máximo tamanho de $(2^k / 2)$. Por exemplo se $k = 3$, os 8 números de sequência usados pelo protocolo, são: 0, 1, ..., 7. Para que o protocolo não falhe, o tamanho máximo da janela do transmissor é $(2^k / 2) = (2^3 / 2) = 4$. Nesse caso inicialmente o transmissor envia os pacotes 0, 1, 2 e 3. Todos são recebidos e confirmados e o receptor passa a esperar o pacote com número de sequência 4, 5, 6 e 7. O transmissor então envia os pacotes 4, 5, 6 e 7, todos são recebidos e confirmados, mas todas as confirmações são perdidas. O receptor nesse momento aguarda os pacotes 0, 1, 2 e 3. Após o esgotamento da temporização, o transmissor retransmite os pacotes 4, 5, 6 e 7 que não são aceitos pelo receptor, pois ele aguarda pacotes com números de sequência 0, 1, 2 e 3.