Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina: Redes de Computadores I
Gabarito AP3 - 2° semestre de 2015.

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 (0,5 pontos cada item) Suponha que você tenha acabado de criar a empresa "Network UFF" e que gostaria de registrar o domínio netuff.com.br na entidade registradora TLD .com.br. Observações:

- Seu servidor DNS possui nome dns1.netuff.com.br e IP = 212.212.212.1;
- Seu servidor SMTP possui nome smtp.netuff.com.br e IP = 212.212.212.2;
- Seu servidor WWW possui nome www.netuff.com.br e IP = 212.212.212.3.

1.1 Liste quais os registros RR que devem inseridos no servidor TLD.

Resposta:

Desprezando o campo TTL (time to live), temos:

(netuff.com.br, dns1.netuff.com.br, NS)

(dns1.netuff.com.br, 212.212.212.1, A)

1.2 Liste quais os registros RR que devem inseridos no seu servidor DNS.

Resposta:

Desprezando o campo TTL (time to live), temos:

(netuff.com.br, smtp.netuff.com.br, MX)

(netuff.com.br, www.netuff.com.br, CNAME)

(www.netuff.com.br, 212.212.212.3, A)

(smtp.netuff.com.br, 212.212.212.2, A)

1.3 Como navegadores irão obter o endereço IP do seu website?

Resposta:

Digamos, por exemplo, que a consulta foi http://netuff.com.br/index.html

O navegador extrai o nome do hospedeiro (netuff.com.br) e repassa para o lado cliente da aplicação DNS. O cliente DNS envia uma consulta para o servidor de DNS local, que caso tenha em seu cachê o resultado já devolve o endereço. Caso contrário esta consulta é repassada pelo DNS local para um servidor de root. O servidor de root devolve o endereço de um servidor de TLD para "com.br". O servidor TLD retorna o endereço de um servidor de nomes com autoridade para o endereço buscado (dns1.netuff.com.br,212.212.212.1,A). A consulta é enviada para o servidor dns1.netuff.com.br e nele é buscado um registro do tipo "CNAME" com o nome canônico do servidor (netuff.com.br,www.netuff.com.br,CNAME). Obtido o nome canônico é procurado um registro do tipo "A", que contém o endereço solicitado (www.netuff.com.br, 212.212.212.3,A). O DNS local recebe o endereço e, finalmente, este endereço é retornado para o cliente DNS que o repassa ao navegador (212.212.212.3).

1.4 Como um cliente de correio descobre o endereço IP do seu servidor de correio?

Resposta:

Digamos, por exemplo, que o cliente de correio esteja procurando bob@netuff.com.br

O cliente de correio extrai o nome do hospedeiro (netuff.com.br) e repassa para o lado cliente da aplicação DNS. O cliente DNS envia uma consulta para o servidor de DNS local, que caso tenha em seu cachê o resultado já devolve o endereço. Caso contrário esta consulta é repassada pelo DNS local para um servidor de root. O servidor de root devolve o endereço de um servidor de TLD para "com.br". O servidor TLD retorna o endereço de um servidor de nomes com autoridade para o endereço buscado (dns1.netuff.com.br,212.212.212.1,A). A consulta é enviada para o servidor dns1.netuff.com.br e nele é buscado um registro do tipo "MX" com o nome canônico do servidor de e-mail (netuff.com.br, smtp.netuff.com.br,MX). Obtido o nome canônico é procurado um registro do tipo "A", este contém o endereço solicitado (smtp.netuff.com.br,212.212.212.2,A). O DNS local recebe o endereço e, finalmente, este endereço é retornado para o cliente DNS que o repassa ao cliente de correio (212.212.212.2).

2 (1,0 ponto) Comente sobre as vantagens de se utilizar a técnica de conexão de controle "fora da banda" no protocolo FTP.

Resposta:

No FTP, os comandos são enviados e recebidos por uma só conexão persistente que é responsável pela transmissão e recepção dos comandos, bem como pela manutenção dos estados em cada um dos lados da conexão (listas de arquivos, diretórios, diretório corrente, etc.). Na transmissão dos dados, uma conexão específica é criada para este fim e encerrada ao seu término. Durante a transmissão, a manutenção dos estados pode continuar, sem que interfira na tarefa de transmitir arquivos de um lado para o outro da conexão.

3 (1,0 ponto) Comente sobre o desempenho do protocolo HTTP em modo não persistente, persistente sem pipelining e persistente com pipelining.

Resposta:

O HTTP não persistente abre uma conexão TCP a cada requisição e fecha a conexão após o envio de cada resposta. O HTTP persistente mantém a conexão TCP aberta aguardando por novas requisições. O HTTP persistente com *pipelining* é capaz de transmitir os objetos requisitados em "paralelo". Com isso, o HTTP não persistente tende a ter o maior tempo de resposta, pois existe o custo de estabelecimento e liberação de cada conexão TCP. Em relação aos persistentes, a versão com *pipelining* tende a ser a mais eficiente já que não há o intervalo de inatividade entre as requisições dos objetos.

4 (0,5 pontos cada item) Em uma rede que emprega comutação por pacotes, considere o envio de pacotes de um host transmissor para um host receptor sobre uma dada rota com N roteadores.

4.1 Enumere os retardos componentes (delays) do retardo fim-a-fim de um pacote e explique o significado de cada retardo.

Resposta:

- **Retardo de processamento (d_{proc})**: este retardo inclui o tempo para examinar o cabeçalho do pacote e determinar para qual saída será encaminhado. Este retardo também inclui o tempo para a verificação do pacote quanto aos possíveis erros introduzidos durante a transmissão no enlace.
- **Retardo de transmissão (d_{trans})**: este retardo depende do tamanho do pacote (L) e da à taxa de transmissão (R) suportada, e seu valor é L / R.
- **Retardo de propagação (d_{prop})**: é o tempo de propagação de um bit de um extremo do enlace até o outro e depende da velocidade de propagação do sinal no meio utilizado (S) e da distância (D) e seu valor é D/S.
- **Retardo de fila (d_{fila})**: tempo que o pacote aguarda na fila de entrada para ser encaminhado à fila de saída do roteador e o tempo de espera na fila de saída até que este seja o próximo pacote a ser transmitido na interface de saída.

4.2 Qual a expressão do retardo fim-a-fim?

Resposta:

Para N roteadores atravessados da origem até destino, considerando retardos idênticos em cada um dos roteadores, a expressão seria a seguinte:

$$d_{total} = (N+1) * (d_{fila} + d_{proc} + d_{trans} + d_{prop}).$$

Já se considerarmos retardos distintos nos roteadores, a expressão seria a seguinte:

$$d_{total} = \sum_{i=1}^{N+1} \{d_{fila}(i) + d_{proc}(i) + d_{trans}(i) + d_{prop}(i)\}$$

5 (0,5 pontos cada item) Responda verdadeiro ou falso, explicando sua escolha:

5.1 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. O controle de fluxo do TCP, através do campo "RcvWindow", indica para o lado transmissor a sua capacidade de recepção de dados. Portanto, o número de *bytes* não reconhecidos não excederá a capacidade de recepção.

5.2 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 subseqüente e m+1.

Resposta:

Falso. No caso do TCP, o número de sequência do segmento subsequente seria: **m + k**, onde **k** é a quantidade de *bytes* de dados contidos no segmento anterior.

6 (2,0 pontos) Imagine que no estabelecimento de uma conexão TCP, em vez do handshake de três vias, tenha sido empregado o handshake de duas vias. Em outras palavras, a terceira mensagem não é usada. É possível que ocorra algum problema transferência confiável de dados? Forneça um exemplo do problema ou mostre que não existe qualquer problema em se usar o handshake de duas vias.

Resposta:

Se o estabelecimento da conexão se der em duas vias é possível sim que a transferência confiável de dados não ocorra. O ponto crucial do problema é a existência de duplicatas atrasadas. Imagine a situação que cada pacote sofre temporização e é retransmitido duas ou três vezes. Alguns pacotes, tanto de controle como de dados, podem ficar retidos no núcleo da rede e emergir muito tempo depois. Lembremos que quando do estabelecimento da conexão cada lado da conexão começa com número de sequência diferente e ambos devem concordar em relação ao número de sequência escolhido por cada lado. Agora vamos ver como o *handshake* de três vias resolve o problema de pacotes de controle duplicados. Imagine que emerge no hospedeiro 2 (o servidor) uma duplicata do segmento SYN de uma antiga conexão. Essa duplicata chega ao hospedeiro 2 sem o conhecimento do hospedeiro 1 (o cliente). O servidor responde ao cliente com um segmento SYNACK, para verificar se o cliente deseja realmente estabelecer uma nova conexão. O cliente percebe que o número de sequência que está sendo confirmado corresponde a um segmento SYN antigo que ficou retido na rede e rejeita o estabelecimento da conexão. Sem a terceira via do estabelecimento da conexão, isto é o ACK do SYNACK, o servidor poderia receber segmentos de dados duplicados que ficaram retidos na rede, e emergiram após duplicata do segmento de controle SYN duplicado mencionado anteriormente.

7 (0,5 pontos cada item) Os objetivos do controle de fluxo e do controle de congestionamento realizados pelo TCP, não são os mesmos. Responda:

7.1 Qual o objetivo do controle de fluxo?

Resposta:

O controle de fluxo visa impedir que um receptor fique sobrecarregado por estar recebendo pacotes a uma taxa superior à que este possa consumi-los.

7.2 De que forma é realizado o controle de fluxo?

Resposta:

Para implementar o controle de fluxo, o transmissor precisa conhecer o espaço disponível no *buffer* de recepção do lado receptor da comunicação. Este dado é fornecido através do campo "Receive Window", que está presente no cabeçalho do TCP e que indica o espaço, em *bytes*, disponível no *buffer* de recepção. Como o TCP é *full duplex,* o campo "Receive Window" é preenchido, no lado receptor da conexão, todas as vezes que um segmento é enviado para o lado transmissor. Com o mecanismo de controle de fluxo, um problema poderia ser ocasionado quando a janela de recepção disponível chegasse a zero. Nessa situação, o transmissor não enviaria dados para o receptor e caso o receptor não tivesse dados para enviar na conexão, o transmissor não teria como saber que o espaço na janela de recepção foi liberado. O TCP resolve este problema forçando o envio periódico de pacotes, por parte do transmissor, com apenas um *byte* de dados, quando a capacidade da janela de recepção chega a zero.

7.3 Qual o objetivo do controle de congestionamento?

Resposta:

O controle de congestionamento visa proteger a rede de uma carga de pacotes superior a sua capacidade.

7.4 De que forma é realizado o controle de congestionamento?

Resposta:

O controle de congestionamento tem como alvo a infra-estrutura de comunicação da Internet, que interliga os dois hospedeiros, e tem como objetivo evitar um colapso de comunicação no interior da rede IP. O mecanismo de controle de congestionamento no TCP, que é baseado no "Aumento Aditivo, Diminuição Multiplicativa" (AIMD), mantém a conexão TCP em dois estados:

- **Inicio lento (*slow start* – SS):** A conexão TCP está neste estado quando a janela de congestionamento for inferior ao limiar (*threshold)*. A cada confirmação (ACK) recebida em sequência o tamanho da janela de congestionamento é acrescido do tamanho de um MSS, o que resulta na duplicação da janela de congestionamento a cada RTT (*Round Trip Time*).
- **Prevenção de congestionamento (*congestion avoidance* - CA):** A conexão TCP está neste estado quando a janela de congestionamento (CongWin) for igual ou superior ao limiar (*threshold*). A cada confirmação (ACK) recebida em sequência o tamanho da janela de congestionamento é acrescido através da fórmula:

$$CongWin = CongWin + MSS \times MSS/CongWin$$

A aplicação desta fórmula resulta no acréscimo do tamanho de um MSS ao tamanho da janela de congestionamento a cada RTT (*Round Trip Time*). Obviamente, este crescimento na janela de congestionamento é efetuado de forma que não infrinja controle de fluxo.

Outros eventos, além da recepção de um ACK esperado e em sequência, são utilizados neste mecanismo:

- Quando um ACK é recebido em duplicata é incrementado o contador de ACKs em duplicata e quando o terceiro ACK em duplicata é recebido a janela de congestionamento é reduzida à metade e o limiar (*threshold*) também recebe este mesmo valor. Portanto, a conexão entra no estado de prevenção do congestionamento.
- Quando o temporizador expira (*timeout*) o limiar recebe o valor da metade do tamanho da janela de congestionamento e a janela de congestionamento é reduzida para seu tamanho mínimo, ou seja, de um MSS. Portanto, a conexão TCP entra na fase de inicio lento (*Slow Start*).