



Questões do livro - 04 (Teórica)

QUESTÃO 01

Considere um servidor simples que responde a requisições dos clientes sem acessar outros servidores

a) Explique porque normalmente não é possível estabelecer um limite do tempo que se leva para que tal servidor envie a resposta ao cliente

b) O que precisaria ser feito para fazer com que o servidor seja capaz de executar as requisições em um tempo limite? Isso é uma opção prática?

R =

Variabilidade da Carga de Trabalho:

- Se muitos clientes enviarem requisições ao mesmo tempo, o servidor pode ficar sobrecarregado, aumentando o tempo necessário para processar cada requisição.

Problemas de Rede:

- pacotes levam para ir e voltar entre o servidor e o cliente, pode variar devido a congestionamento, problemas de roteamento, ou falhas na rede.

Recursos do servidor:

- Se o servidor não tiver recursos suficientes (CPU, memória..)

QUESTÃO 02

Cite cinco tipos de recurso de hardware e cinco tipos de recursos de dados ou de software que possam ser compartilhados com sucesso. Dê exemplos práticos de seu compartilhamento em sistemas distribuídos.

R =

- **Hardware:** Servidores, Impressoras, Redes de comunicação
- **Software:** Bases de dados, APIs e Serviços Web, Sistema de arquivos distribuídos

QUESTÃO 03

Como os relógios de dois computadores ligados por uma rede local podem ser sincronizados sem referência a uma fonte de hora externa? Quais fatores limitam a precisão do procedimento que você descreveu? Como os relógios de um grande número de computadores conectados pela Internet poderiam ser sincronizados? Discuta a precisão desse procedimento

R =

- Podem ser sincronizados usando o protocolo **NTP** (Network Time Protocol). O cliente envia uma solicitação de tempo ao servidor, que responde com o tempo atual. O cliente ajusta seu relógio com base nessa resposta, levando em consideração o tempo de ida e volta da comunicação.
- Fatores que limitam a precisão incluem a latência e a variabilidade da rede, além do tempo de processamento nos computadores

QUESTÃO 04

Considere as estratégias de implementação de MMOG (massively multiplayer online games) discutidas na Seção 1.2.2. Em particular, quais vantagens você vê em adotar a estratégia de servidor único para representar o estado do jogo para vários jogadores? Quais problemas você consegue identificar e como eles poderiam ser resolvidos?

R =

- Um servidor único facilita que todos os jogadores vejam e interajam com o mesmo estado do jogo, reduzindo problemas de dessincronização
- Ter um ponto central de controle permite uma gestão mais fácil de eventos globais, atualizações e manutenção
- Não há necessidade de lidar com a complexidade adicional de replicação de dados e sincronização entre múltiplos servidores.

QUESTÃO 05

No contexto da comunicação Cliente-Servidor sobre UDP, explique como é possível identificar e mascarar a perda de mensagens em mensagens duplicadas

R =

- **Identificação de Perda de Mensagens:** Usar números de sequência, confirmações (ACKs) e temporizadores para detectar e retransmitir mensagens perdidas.
- **Máscara de Mensagens Duplicadas:** Use números de sequência únicos, timestamps ou identificadores únicos para identificar e descartar mensagens duplicadas.

1. Número de Sequência:

- **Descrição:** Adicione um número de sequência a cada mensagem que o cliente envia para o servidor. O servidor pode usar esse número para identificar pacotes perdidos.
- **Funcionamento:** Quando o servidor recebe uma mensagem, ele verifica o número de sequência. Se o número de sequência estiver fora da ordem esperada ou ausente, o servidor pode determinar que uma mensagem foi perdida. Para garantir a entrega, o servidor pode solicitar que o cliente reenvie as mensagens com números de sequência ausentes.

2. Confirmação (ACK):

- **Descrição:** O servidor pode enviar uma mensagem de confirmação (ACK) para o cliente após receber uma mensagem.
- **Funcionamento:** O cliente aguarda o recebimento do ACK antes de considerar a mensagem como entregue. Se o cliente não receber um ACK dentro de um tempo determinado (timeout), ele pode reenviar a mensagem.

3. Temporizadores e Retransmissão:

- **Descrição:** Use temporizadores para detectar quando uma confirmação não é recebida dentro de um prazo específico.
- **Funcionamento:** Após enviar uma mensagem, o cliente inicia um temporizador. Se o temporizador expirar antes que um ACK seja recebido, o cliente reenvia a mensagem. Este método ajuda a lidar com a perda de pacotes, mas pode levar a retransmissões desnecessárias em redes com alta latência.