

Exercício Aula 06

Professor: Felipe Lobo

Aluno: VALERIA ALEXANDRA GEVARA PARRA 2019047300

1. Qual o objetivo de cada camada da pilha de camadas?

Entre cada par de camadas adjacentes existe uma interface. A interface define as operações e os serviços que a camada inferior tem a oferecer a camada que se encontra acima dela. Um conjunto de camadas e protocolos é chamado arquitetura de rede. Não é necessário que as interfaces de todas as máquinas de uma rede sejam iguais, desde que cada uma delas possa usar todos os protocolos de maneira correta. Uma lista de protocolos usados por um determinado sistema, um protocolo por camada, é chamada pilha de protocolos. Todas as camadas precisam de um mecanismo para identificar os transmissores e os receptores.

2. Como funciona a transferência de dados de uma camada x de uma máquina para uma camada x de outra máquina?

Na realidade, os dados não são transferidos diretamente da camada n de uma máquina para a camada n de outra máquina, ou seja, cada camada transfere os dados e as informações de controle para a camada imediatamente abaixo dela, até ser alcançada a camada mais baixa. Abaixo da camada 1 encontra-se o meio físico através do qual se dá a comunicação propriamente dita a comunicação virtual é mostrada por linhas pontilhadas e a comunicação física por linhas contínuas em alguns sistemas, os dados são transferidos em apenas um sentido; em outros, os dados trafegam em ambos os sentidos.

3. Exalte algumas questões de projetos relacionadas às camadas.

- O protocolo também deve definir a quantos canais lógicos corresponde a conexão e quais são suas prioridades. Muitas redes fornecem pelo menos dois canais lógicos por conexão, um para dados normais e um para dados urgentes. Questões de Projeto relacionadas.
- Muitos códigos de detecção e correção de erros são conhecidos, mas as partes envolvidas na conexão devem chegar a um consenso quanto ao que está sendo usado.
- O receptor deve ter algum meio para informar ao transmissor quais mensagens foram recebidas corretamente e quais não foram.
- Nem todos os canais de comunicação preservam a ordem das mensagens enviadas a eles.
- O protocolo deve permitir explicitamente ao receptor remontar de forma adequada os fragmentos recebidos.
- Uma solução óbvia é numerar os fragmentos, mas essa solução ainda deixa aberta a questão do que deve ser feito com os fragmentos que chegarem fora de ordem. Questões de Projeto relacionadas às camadas
- Uma questão que afeta cada nível é como impedir que um transmissor rápido envie uma quantidade excessiva de dados a um receptor mais lento.
- Várias soluções foram propostas e serão discutidas a seguir. Esse questão é conhecida como controle de fluxo

4. Defina serviços orientados a conexões e serviços sem conexão.

- As camadas podem oferecer dois tipos diferentes de serviços as camadas situadas acima delas: serviços orientados a conexões e serviços sem conexões.
- O serviço orientado a conexões se baseia no sistema telefônico. Para falar com alguém, você tira o fone do gancho, discar o número, fala e, em seguida, desliga.
- Da mesma forma, para utilizar um serviço de rede orientado a conexões, primeiro o usuário

do serviço estabelece uma conexão, utiliza a conexão, e depois libera a conexão

- Cada serviço pode ser caracterizado por uma qualidade de serviço. Alguns serviços são confiáveis, no sentido de nunca perderem dados. Em geral, um serviço confiável é implementado para que o receptor confirme o recebimento de cada mensagem, de modo que o transmissor se certifique de que ela chegou.
- Uma situação típica em que um serviço orientado a conexões confiável é apropriado é a transferência de arquivos.
- Um exemplo em que não se tem a necessidade de estabelecer uma conexão para enviar uma única mensagem curta com a necessidade de uma confiabilidade é o serviço de datagramas com confirmação;
- Funciona semelhante a enviar uma carta registrada e receber um aviso de recebimento. Quando recebe o aviso, o transmissor fica certo de que a mensagem foi entregue ao destinatário.

5. O que são primitivas de serviços? Dê um exemplo de utilização de primitivas visando estabelecer uma conexão entre servidor e cliente.

Um serviço é especificado formalmente por um conjunto de primitivas (operações) disponíveis para que um processo do usuário acesse o serviço.

- Essas primitivas informam ao serviço que ele deve executar alguma ação ou relatar uma ação executada por uma entidade par.
- Se a pilha de protocolos estiver localizada no sistema operacional, como ocorre com frequência, as primitivas serão normalmente chamadas do sistema
- As primitivas para um serviço orientado a conexões são diferentes das que são oferecidas em um serviço sem conexões. Como um exemplo mínimo das primitivas de serviço que poderiam ser fornecidas para implementar um fluxo de bytes confiável em um sistema cliente/servidor.
- Primeiro, o servidor executa LISTEN para indicar que está preparado para aceitar conexões de entrada. Depois de executar a primitiva, o processo servidor fica bloqueado até surgir uma solicitação de conexão.
- Em seguida, o processo cliente executa CONNECT para estabelecer uma conexão com o servidor. A chamada de CONNECT precisa especificar a quem se conectar; assim, ela poderia ter um parâmetro fornecendo o endereço do servidor
- Em geral, o sistema operacional envia então um pacote ao par solicitando que ele se conecte. Quando o pacote chega ao servidor, ele é processado pelo SO do servidor.
- Quando o sistema observa que o pacote está solicitando uma conexão, ele verifica se existe um ouvinte. Nesse caso, ele realiza duas ações: desbloqueia o ouvinte e envia de volta uma confirmação (2). A chegada dessa confirmação libera o cliente. Nesse momento, o cliente e o servidor estão em execução e tem uma conexão estabelecida entre eles.
- A próxima etapa é a execução de RECEIVE pelo servidor, a fim de se preparar para aceitar a primeira solicitação. Normalmente, o servidor faz isso imediatamente após ser liberado de LISTEN, antes da confirmação poder retornar ao cliente. A chamada de RECEIVE bloqueia o servidor.
- Depois, o cliente executa SEND para transmitir sua solicitação (3), seguida pela execução de RECEIVE para receber a resposta.
- A chegada do pacote de solicitação a máquina servidora desbloqueia o processo servidor,

para que ele possa processar a solicitação. Depois de terminar o trabalho, ele utiliza SEND para enviar a resposta ao cliente (4). A chegada desse pacote desbloqueia o cliente, que pode agora examinar a resposta