Segunda Aula de Sistemas Distribuídos, Questões do Livro

Victor Hugo Vieira Cruz - 20231011090296

2.6 Frequentemente, os computadores usados nos sistemas peer-to-peer são computadores desktop dos escritórios ou das casas dos usuários. Quais são as implicações disso na disponibilidade e na segurança dos objetos de dados compartilhados que eles contêm e até que ponto qualquer vulnerabilidade pode ser superada por meio da replicação?

O uso de computadores desktop em sistemas peer-to-peer traz algumas implicações importantes. Como esses computadores são geralmente de uso pessoal ou de escritório, eles nem sempre estão ligados ou conectados à internet, o que pode prejudicar a disponibilidade dos dados compartilhados. Além disso, esses sistemas podem ser mais vulneráveis a ataques, já que nem sempre contam com boas práticas de segurança ou atualizações em dia. Por outro lado, a replicação dos dados — ou seja, manter várias cópias espalhadas pela rede — ajuda bastante a contornar esses problemas. Mesmo que um computador fique offline ou sofra um ataque, os dados ainda podem estar disponíveis em outros pontos da rede. Ainda assim, a replicação não resolve tudo sozinha: é importante combinar essa estratégia com cuidados de segurança para proteger bem as informações.

2.7 Liste os tipos de recurso local vulneráveis a um ataque de um programa não confiável, cujo download é feito de um site remoto e que é executado em um computador local.

Quando a gente baixa e executa um programa de um site remoto e não confiável, vários recursos do nosso computador podem ficar expostos. Por exemplo:

- Sistema de arquivos: O programa pode fuçar nos seus arquivos, apagar ou alterar coisas importantes.
- Memória: Ele pode acessar dados que estão sendo usados por outros programas e bagunçar tudo.
- Rede: Pode enviar informações pela internet sem você saber, ou até tentar invadir outros sistemas.
- Dispositivos periféricos: Pode ativar coisas como a câmera, microfone ou impressora, colocando sua privacidade em risco.

2.8 Dê exemplos de aplicações em que o uso de código móvel seja vantajoso.

Alguns exemplos em que o uso de código móvel é vantajoso:

- Atualizações de software: Permite que partes do sistema sejam atualizadas diretamente no dispositivo do usuário, sem precisar reinstalar tudo.
- Processamento de dados distribuído: Em vez de transferir grandes volumes de dados pela rede, o código pode ser enviado até onde os dados estão, economizando tempo e recursos.

2.11 Considere um servidor simples que executa pedidos do cliente sem acessar outros servidores. Explique por que geralmente não é possível estabelecer um limite para o tempo gasto por tal servidor para responder ao pedido de um cliente. O que precisaria ser feito para tornar o servidor capaz de executar pedidos dentro de um tempo limitado? Essa é uma opção prática?

Um servidor simples pode ter dificuldade para garantir um tempo fixo de resposta porque vários fatores influenciam isso: a quantidade de pedidos recebidos, o uso de recursos e até possíveis falhas no sistema. Para tentar garantir um tempo de resposta previsível, seria necessário:

- Gerenciar filas de pedidos, evitando sobrecarga.
- Controlar o uso de recursos, como CPU e memória. Apesar de possível, essa abordagem pode exigir mais infraestrutura e planejamento, o que nem sempre é viável em sistemas mais básicos.

2.12 Para cada um dos fatores que contribuem para o tempo gasto na transmissão de uma mensagem entre dois processos por um canal de comunicação, cite medidas necessárias para estabelecer um limite para sua contribuição no tempo total. Por que essas medidas não são tomadas nos sistemas distribuídos de propósito geral atuais?

Alguns fatores influenciam o tempo de transmissão de uma mensagem entre dois processos:

- Latência de transmissão: É o tempo que leva para a mensagem começar a ser enviada até chegar ao destino. Para reduzir esse tempo, seria necessário usar redes mais rápidas e protocolos mais eficientes.
- Largura de banda: Diz respeito à quantidade de dados que podem ser transmitidos por segundo. Isso pode ser otimizado com técnicas como compressão de dados.
- Tempo de processamento: Refere-se ao tempo gasto para processar a mensagem nos dois extremos da comunicação. Melhorar isso envolve otimizar o código e, quando possível, usar máquinas com melhor desempenho. Apesar dessas soluções, elas nem sempre são adotadas em sistemas distribuídos de uso geral porque o custo e a complexidade aumentam bastante. Além disso, esses sistemas geralmente são feitos para serem flexíveis e funcionarem em ambientes muito variados, o que torna difícil garantir tempos fixos de resposta.

2.13 O serviço Network Time Protocol pode ser usado para sincronizar relógios de computador. Explique por que, mesmo com esse serviço, nenhum limite garantido é dado para a diferença entre dois relógios.

O NTP (Network Time Protocol) é uma boa ferramenta para manter os relógios dos computadores relativamente sincronizados, mas não consegue garantir uma diferença mínima exata entre eles. Isso acontece por alguns motivos:

- Variação na latência da rede: O tempo de resposta pode variar por causa do tráfego na rede ou congestionamentos.
- Desvios naturais dos relógios: Cada computador tem seu próprio relógio interno, que pode se adiantar ou atrasar com o tempo.

- Atrasos no processamento: O tempo que o sistema leva para lidar com as mensagens de sincronização também pode influenciar. Por esses motivos, o NTP melhora bastante a sincronização, mas não consegue garantir uma diferença mínima fixa entre os relógios.

2.14 Considere dois serviços de comunicação para uso em sistemas distribuídos assíncronos. No serviço A, as mensagens podem ser perdidas, duplicadas ou retardadas, e somas de verifica-ção se aplicam apenas aos cabeçalhos. No serviço B, as mensagens podem ser perdidas, retardadas ou entregues rápido demais para o destinatário manipulá-las, mas sempre chegam com conteúdo correto. Descreva as classes de falha exibidas para cada serviço. Classifique suas falhas de acordo com seu efeito sobre as propriedades de validade e integridade. O serviço B pode ser descrito como um serviço de comunicação confiável?

- Serviço A: Pode perder, duplicar ou atrasar mensagens. Como a verificação é feita só no cabeçalho, o conteúdo da mensagem pode estar errado. Essas falhas afetam tanto a validade (quando a mensagem se perde ou atrasa) quanto a integridade (quando ela é duplicada ou chega corrompida).
- Serviço B: Pode perder, atrasar ou entregar mensagens rápido demais para o destinatário processar. Mas, ao menos, garante que o conteúdo esteja correto. Essas falhas afetam a validade, mas não a integridade. O serviço B pode ser considerado mais confiável, já que mantém a integridade das mensagens. No entanto, ainda não é perfeito, pois não garante que todas as mensagens cheguem no tempo certo ou sequer sejam entregues.

2.16 Suponha que uma leitura de disco possa, às vezes, ler valores diferentes dos gravados. Cite os tipos de falha exibidos por uma leitura de disco. Sugira como essa falha pode ser mascarada para produzir uma forma de falha benigna diferente. Agora, sugira como se faz para mascarar a falha benigna.

Quando um disco lê valores diferentes do que foi gravado, algumas falhas possíveis são:

- Leitura incorreta: O valor lido não corresponde ao que foi salvo.
- Leitura desatualizada: Pode retornar um valor antigo em vez do mais recente. Para lidar com esse problema de forma segura, pode-se usar técnicas como somas de verificação ou hashes para conferir se os dados lidos estão corretos. Se houver erro, o sistema pode tentar ler de novo ou buscar uma cópia dos dados em outro lugar (por exemplo, em um backup ou em armazenamento redundante). Assim, transforma-se uma falha potencialmente grave em uma falha benigna — ou seja, mais fácil de contornar.

2.3 Defina a propriedade de integridade da comunicação confiável e liste todas as possíveis ameaças à integridade de usuários e de componentes do sistema. Quais medidas podem ser tomadas para garantir a propriedade de integridade diante de cada uma dessas fontes de ameaças?

Integridade na comunicação confiável significa garantir que a mensagem recebida seja exatamente a mesma que foi enviada — sem alterações, perdas ou adições.

A integridade pode ser ameaçada por:

- Interceptação e modificação: Um atacante pode alterar uma mensagem durante a transmissão.
- Injeção de mensagens: O atacante pode enviar mensagens falsas como se fossem legítimas.
- Ataques de replay: Mensagens reais podem ser reenviadas fora de contexto para enganar o sistema. Para evitar esses problemas, é possível usar:
- Criptografia: Protege o conteúdo da mensagem durante a transmissão.
- Assinaturas digitais: Garantem que a mensagem veio de quem diz ter vindo e que não foi alterada.
- Protocolos de autenticação: Ajudam a verificar a identidade de quem está enviando e recebendo a mensagem.

2.18 Descreva as possíveis ocorrências de cada um dos principais tipos de ameaça à segurança (ameaças aos processos, ameaças aos canais de comunicação, negação de serviço) que poderiam ocorrer na Internet.

Na Internet, as principais ameaças à segurança podem ser divididas em três grupos:

- Ameaças aos processos:
    - Malware: Programas maliciosos que causam danos ou roubam informações.
    - Phishing: Tentativas de enganar usuários para que revelem dados confidenciais.
    - Exploração de vulnerabilidades: Ataques que se aproveitam de falhas no software.
- Ameaças aos canais de comunicação:
    - Intercepção de dados: Alguém pode captar as informações transmitidas entre dois pontos, comprometendo a privacidade.
    - Negação de serviço (DoS):
    - Ataques que sobrecarregam um sistema, impedindo que ele funcione corretamente para usuários legítimos. Essas ameaças exigem cuidados com segurança em várias camadas: dos softwares utilizados até a proteção da rede e do comportamento dos usuários.