Teste-Tipo de Sistemas Distribuídos & Guia de resolução
Capítulos Segurança, Tol.Faltas e Replicação, Nomes, Transações Distribuídas

Baseado em questões de testes e exames de 2013/14

Grupo Segurança

1) Considere dois interlocutores, A e B.
Assuma que, após distribuição através de um canal seguro, A conhece a sua chave privada, KsA, e a chave pública de B, KpB; e vice-versa.
Por vezes, A e B pretendem trocar múltiplas mensagens entre si e para tal seguem o seguinte protocolo:
- O nó (A ou B) que tomar a iniciativa primeiro, gera uma chave simétrica secreta, K.
- Se for o nó A que gerou a chave K, A envia {K}KpB a B; B decifra com KsB e obtém K. (Procedimento inverso para o caso em que B toma primeiro a iniciativa.)
- Sempre que A queira enviar mensagem M a B, ou vice-versa, é enviado {M}K pela rede.

Nas alíneas seguintes assuma que o atacante não consegue quebrar as chaves secretas (KsA, KsB e K).

a) Quando A quer enviar M a B, A poderia optar pela alternativa de enviar {M}KpB. Que desvantagens encontra nesta solução, comparativamente à solução descrita acima?

Referir o melhor desempenho associado ao uso de cifra simétrica para cifrar as mensagens M. Justificar pelo facto de cifra assimétrica ser substancialmente mais lenta que cifra simétrica.
Ver secção 11.3 do livro.

b) O protocolo indicado na alínea 1) é passível de ataque *man in the middle*, em que um intruso T consegue levar A e B a trocarem mensagens cifradas sem saberem que T consegue ter acesso ao respetivo conteúdo em claro.

i) [1,0v] Descreva o procedimento que T deverá levar a cabo para concretizar este ataque. Ilustre com um exemplo.

Referir ataque em que T leva a cabo os passos iniciais do protocolo, levando A a aceitar uma chave de sessão K pensando estar a iniciar sessão com B; complementarmente, T repete o mesmo procedimento com B, levando B a pensar que estabeleceu chave de sessão K' com A.
Exemplificar como, a partir desta situação, T pode agir como *man in the middle*: sempre que A envia mensagem M cifrada com K, T intercepta e decifra M, tendo assim acesso aos dados que A pensava serem confidenciais; posteriormente, T cifra M com K' e envia para B. E vice-versa.
Outras variantes deste ataque são também respostas corretas.
Ver slides sobre "man-in-the-middle".

ii) Proponha uma correcção ao protocolo descrito acima que previna este ataque.

Referir que a solução passa por assegurar a autenticidade e integridade das mensagem inicial que transporta a chave K simétrica. Completar com uma solução concreta para tal: quem gera e envia a chave K, anexa uma assinatura digital da mensagem que transporta K. Apresentar expressão formal que define em que consiste tal assinatura digital.
Opcionalmente, pode também abordar-se o problema da frescura nesta resposta (ver alíneas seguintes).
Ver secção 11.4.1 do livro.

c) O protocolo na alínea 1) é também vulnerável a ataques de repetição (*replay*).

i) Descreva de que forma um atacante T pode concretizar esse ataque. Ilustre com um exemplo em que B é um servidor de contas bancárias e permite aos clientes debitarem/creditarem as suas contas.

Apresentar ataque em que T escuta troca de mensagens legítima entre A e B e posteriormente repete as mensagens antigas para daí obter algum proveito. Um exemplo simples: A é cliente de um serviço B. T observa (e guarda) troca de mensagens que A troca com B para invocar uma operação do serviço de B. No futuro, T pretende repetir a operação invocada por A no passado, pois a execução dessa operação traz benefício a T. Para tal, T limita-se a re-injectar na rede as mensagens emitidas por A na sessão antiga. B não tem forma de duvidar da frescura das mensagens, logo é levado a executar ilegitimamente a operação.
Ver secção 2.4.3 do livro (subsecção "Threats to communication channels").

ii) [1,0v] Proponha uma correcção ao protocolo que previna este ataque.

Apresentar uma solução que permita a B confirmar que a mensagem inicial que transporta K, emitida por A, é fresca.
Várias soluções possíveis, desde o uso de marcas temporal na mensagem com K (e.g., como o protocolo Kerberos faz) à introdução de uma etapa adicional em que B responde à mensagem inicial de A com desafio baseado em nonce (e.g., como o protocolo Needham-Schroeder de chave simétrica faz).
Ver secções 11.6.1 e 11.6.2 do livro para conferir exemplos dos protocolos Needham-Schroeder e Kerberos.

2) Para cada mensagem M recebida por A ou B, o receptor pretende ter forma de garantir confidencialidade, integridade e não repúdio da mensagem.
   a) [0,9v] Apresente detalhadamente uma solução que garanta esses 3 requisitos.

Citar assinatura digital de chave pública como solução. Apresentar formalmente a expressão que define a assinatura.
Ver secção 11.4.1 do livro.

b) [0,9v] Apresente agora uma solução mais eficiente que não ofereça a garantia de não repúdio.

Citar MAC como solução. Apresentar formalmente a expressão que define um MAC.
Ver secção 11.4.2 do livro

3) Uma forma de A ter tomado conhecimento da chave pública de B foi através de um certificado digital de chave pública X509.
   a) Indique os campos principais de um certificado X509 e respectivo significado.

Enumerar os campos principais de certificado X509 e seu significado.
Ver secção 11.4.4 (e 11.2.3) do livro.

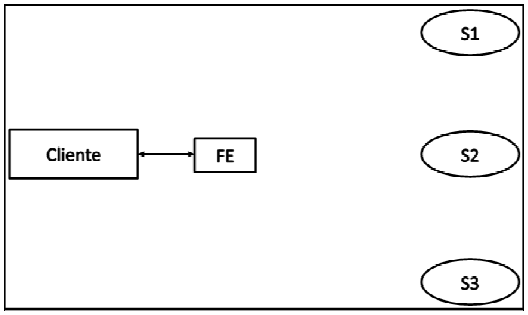
b) [0,9v] A obteve o certificado da chave pública de B a partir de um site desconhecido na Web. Pode A confiar na chave que vem no certificado? Se sim, que passos deve A efectuar para confirmar que a chave é realmente a chave pública actual de B? Se não, justifique.

Referir que sim, pois a autenticidade e integridade do certificado não depende do canal pelo qual foi obtido.
Descrever os passos de validação do certificado: i) validar assinatura do certificado, usando chave pública da CA emissora do certificado (caso seja desconhecida, é necessário descobrir essa chave pública através da hierarquia da PKI); ii) validar período de validade; iii) consultar lista de certificados revogados.
Ver secção 11.4.4 (e 11.2.3) do livro.

Grupo Tolerância a Faltas e Replicação

1) Considere um sistema replicado em que um cliente (C) interactua através de um Front-End (FE) com um conjunto de três servidores (S1, S2, S3). As operações sobre os servidores são apenas de leitura (R) e escrita (W). O protocolo usado é o de p*rimary-backup.* Assuma que os servidores são de **falha silenciosa**, que o sistema é síncrono, que a rede não tem falhas permanentes e garante uma ordem FIFO das mensagens.

a) Considere que o cliente efectua uma escrita na variável A (W(A,10)) e depois lê o valor da mesma variável R(A). Complete o diagrama indicando as operações que são realizadas, descrevendo-as através de setas com a legenda das operações (W(A, 10), R(A)). Escreva as legendas que achar necessárias, indique todas as mensagens trocadas entre cliente e os servidores, e entre estes.



Para cada pedido (W(A,10) e R(A)), apresentar as mensagens pedido-resposta entre FE e S1 (servidor primário); após receção de cada pedido, acrescentar a mensagem de atualização (*update*) para os secundários. Complementarmente, apresentar as mensagens de prova de vida periódicas entre primário e secundários.
Aspetos opcionais: - A propagação dos updates e provas de vida pode ser feita de diferentes formas – primário envia para os dois secundários, ou em cadeia (S1 envia a S2; S2 propaga para S3). - As operações de leitura-apenas (R(A)) não têm de ser propagadas aos secundários; embora não tenha sido estudada nas teóricas, esta otimização pode ser considerada na resposta.
Ver secção 18.3.1 do livro e slides "Replicação Passiva".

b) Explique quantas falhas de servidores pode tolerar este sistema. Justifique.

Referir grau de replicação do protocolo de replicação passiva estudado nas teóricas (N=f+1, em que N é o grau de replicação e f é o número de falhas silenciosas toleradas). Em consequência, concluir que f=2 neste sistema.
Ver secção 18.3.1 do livro e slides "Replicação Passiva".

c) Escreva uma expressão que calcule o tempo máximo de indisponibilidade de sistema, considerando que o cliente utiliza um servidor de nomes (tipo UDDI) para conhecer o endereço do servidor? Defina todos parâmetros os presentes na expressão (ex.: t_reg_uddi - tempo de registo no UDDI, etc.)

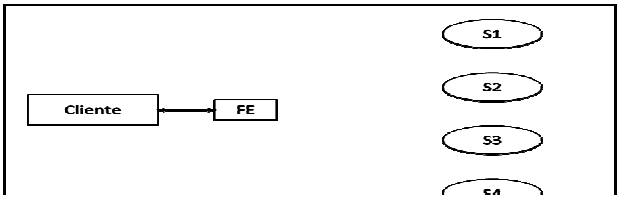
Apresentar expressão do tempo de recuperação no pior caso.
Assumindo P=período das provas de vida e tmax=tempo máximo de entrega de mensagem, o tempo de recuperação no pior dos casos é composto pela soma de: - tempo máximo para o secundário detetar a falha do primário (P+tmax) - tempo máximo para o secundário se registar no UDDI como novo primário - tempo máximo para o cliente consultar o UDDI e descobrir endereço do novo secundário - tempo máximo para o cliente reenviar o pedido (tmax)
Ver slides "Replicação Passiva".

2) Considere agora que a **rede é assíncrona** e que utiliza um protocolo *quorum consensus* com quóruns de maioria.

a) O que significa o pressuposto que o sistema é assíncrono e o que muda em relação ao sistema da pergunta 1?

Apresentar definição de sistema assíncrono.
Referir que sistema assíncrono não é aceitável para o protocolo da pergunta 1, que pressupõe sistema síncrono.
Ver slides "Replicação Passiva" e "Replicação Activa"; ver pdf "Quorum Consensus Replication".

b) Pressupondo que executa a operação (W(A, 10), R(A)). Complete o diagrama com todas as interacções entre o cliente e os servidores, admitindo que S2 não recebe a



mensagem de write e S3 falha depois aquando da execução de R(A).

Apresentar mensagens trocadas entre FE e os múltiplos servidores seguindo o protocolo apresentado nas aulas. O exemplo esperado na resposta é semelhante ao exemplo (com diagrama de mensagens) apresentado na secção "Replicação Activa" dos slides.
Ver slides "Replicação Activa" e pdf "Quorum Consensus Replication".

c) Quantas falhas silenciosas dos servidores tolera este sistema? Justifique.

Referir que o grau de replicação deste protocolo com quóruns de maioria é N=2f+1.
Consequentemente, concluir que neste caso f=1.
Ver pdf "Quorum Consensus Replication".

d) Considere que os servidores S1 e S2 estão ligados a um *backbone* de rede com elevada banda passante, e os restantes têm ligações idênticas e mais lentas à rede (para exemplo a ligação é 3 vezes mais rápida). Como poderia optimizar o funcionamento do sistema para acelerar as estruturas, passando a usar um **protocolo de quóruns com pesos variáveis**?

i) Explique como definiria os pesos de cada servidor.

Apresentar proposta que aumenta o peso de S1 e S2 de forma a que a soma de peso(S1)+peso(S2) > peso(todos os servidores)/2. Desta forma permite-se que, quando S1 e S2 não estão em falha, um pedido possa ser completado apenas com as respostas destes 2 servidores mais rápidos.
Por exemplo, peso(S1)=peso(S2)=2 e peso(S3)=peso(S4)=1.
Ver pdf "Quorum Consensus Replication".

ii) Justifique com um exemplo qual a melhoria que esta proposta apresenta em relação ao quórum de maioria.

Dar exemplo de pedido (escrita ou leitura) que é respondido por S1 e S2 (muito mais rapidamente que a resposta de S3 e S4). Como as respostas de S1 e S2 são suficientes para completar um quórum, o FE retorna mais rapidamente ao cliente.
Ver pdf "Quorum Consensus Replication".

3) Considere agora que os servidores são de **falha arbitrária**. Em que difere este tipo de falha da falha silenciosa? Apresente um exemplo para o cenário da pergunta 1.

Referir que, em caso de falha arbitrária, os servidores em falha podem continuar a responder a pedidos ou a emitir mensagens para o exterior.
Vários exemplos são possíveis em que falha arbitrária perverte o comportamento do sistema. Dois exemplos: servidor primário deixa de responder ao FE mas continua a enviar provas de vida; servidor primário passa a responder erroneamente ao FE mas continua a enviar provas de vida.
Ver secções 2.4.2 e 18.3.1 do livro e slides "Replicação Passiva".

Grupo Nomes

1. Considere que, num determinado país com 5 milhões de habitantes, existia um sistema de identificação nacional de cidadãos que consistia em:
   - Cada cidadão era identificado por um número cidadão que consistia em 140 dígitos decimais.
   - Cada região tinha um serviço de nomes regional, que mantinha um directório com os identificadores dos cidadãos registados nessa região, que associava cada número de cidadão a diversos atributos do cidadão.
   - Quando uma nova pessoa nascia, os pais levavam o recém-nascido aos serviços de identificação, onde era gerado aleatoriamente um identificador para o novo cidadão. Esse novo identificador era registado no directório local.

   a. [1v] Como caracteriza o número de cidadão deste país quanto a: âmbito, pureza, heteronegeidade ? Justifique.

Referir que o número de cidadão é global, puro, homogéneo.
Justificar cada classificação aplicando a definição teórica ao contexto particular do exercício. Exemplo: É global porque tem o mesmo significado em qualquer parte do país.
Ver secção 13.1.1 do livro e slides "Propriedades dos nomes".

   b. Suponha que o governo deste país pondera, em vez desta solução, optar por uma solução hierárquica em que cada número de cidadão era dado por um código de região seguido por um número único de âmbito regional. Que vantagem encontra nessa solução?
   Indique qual/quais das propriedades do nome mudariam em relação à sua resposta à alínea anterior.

Referir a maior eficiência na resolução de nomes. Justificar com o facto do nome passar a ser impuro.
Ver secção 13.1.1 do livro.

2. Considere o serviço de nomes DNS.
   a. "O DNS não garante consistência forte das associações (nome DNS, IP)." Ilustre esta afirmação referindo dois exemplos de mecanismos do DNS.

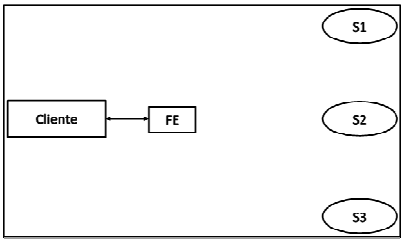
Afirmar que sincronização entre primário e secundário em DNS não é imediata; também a sincronização das caches não é imediata.
Concluir que ambos os fatores levam a inconsistências temporárias na resposta a pedidos de resolução de nomes que tenham mudado recentemente.
Ver secção 13.2 do livro (em particular, "Discussion of the DNS").

   b. "Para resolver um nome DNS em modo iterativo, o cliente contacta sempre todos os servidores os servidores dos domínios desse nome, a começar pelo servidor de raiz." Esta afirmação é verdadeira? Justifique.

Explicar que, caso o cliente ou o servidor local contactado pelo cliente conheçam o endereço de um servidor de um dos sub-domínios do nome a resolver, o DNS evita contactar todos os servidores de domínio desde a raiz.
Concluir que a afirmação é falsa.
Ver secção 13.2.2 e 13.2.3 do livro.

Grupo Transações Distribuídas

1) Considere um programa cliente que quer executar operações de leitura e escrita sobre servidores distintos e garantindo transações atómicas distribuídas, usando 2-phase commit (2PC).



a) Adicione ao diagrama seguinte um componente indispensável para poder executar as transacções distribuídas.

Adicionar processo "coordenador" ao diagrama.
Ver secção 17.2.1 do livro.

b) Represente no diagrama acima, através de setas, todas as interacções que desencadeia a execução da operação *openTransaction.*

Adicionar pedido openTransaction do FE ao Coordenador; resposta ao pedido contém o identificador global da transação distribuída.
Ver secção 17.2.1 do livro.

c) Represente no diagrama acima, através de setas, todas as interacções que desencadeia a execução da transação distribuída composta pela escrita da variável X no servidor S1 e pela leitura da variável Y no servidor S2. Considere só a representação até esse momento; a evolução subsequente e, em particular o 2PC, não fazem parte desta representação.

Para cada invocação (escrita de X sobre S1 e leitura de Y sobre S2): - adicionar mensagem de FE para S1/S2 com pedido da operação (que deve levar identificador da operação, argumentos, id. global da transação distribuída); - adicionar mensagem "join" que S1/S2 enviam ao Coordenador; - adicionar mensagem de resposta ao pedido, retornada de S1/S2 ao FE.
Como o enunciado indica, não se pedia para apresentar as mensagens trocadas durante o protocolo 2PC, que só se iniciará quando o FE efetuou todas as invocações de operações da transação e solicita *closeTransaction* ao Coordenador.
Ver secção 17.2.1 do livro.

d) Depois desta operação o servidor S1 falha silenciosamente.

i) Como será detetado?

Referir que, após a execução da transação ter completado, o FE enviará pedido *closeTransaction* ao Coordenador. Esse pedido causará o início do protocolo 2PC por parte do Coordenador.
Explicar que o Coordenador começará por enviar mensagem canCommit a cada participante e esperará pelos respetivos votos. Concluir que, caso um dos participantes esteja em falha silenciosa (neste caso, S1), o voto desse participante não chegará ao Coordenador em tempo útil, o que leva o Coordenador a *suspeitar* que S1 está em falha.
Opcionalmente, referir que, em sistema assíncrono, Coordenador apenas suspeita da falha de S1 mas nunca tem certeza de que detetou uma falha.
Ver secções 17.3. e 17.3.1 do livro.

ii) O que irá suceder à transacção global nesse caso?

Explicar que a não receção de voto de um dos participantes antes de timeout leva o coordenador a enviar decisão de abortar a transação a todos os participantes.
Ver secção 17.3.1 do livro.

e) Considere agora uma execução alternativa em que a transacção decorre normalmente até ao cliente fazer closeTransaction e o servidor S1 falha depois de ter respondido ao *canCommmit* (*prepare)* com *yes* (*ready)*. O que irá suceder à transação global? Justifique.

Explicar que, neste caso, o Coordenador decide ordenar a todos os participantes que confirmem a transação.
Referir que, quando S1 recuperar, eventualmente receberá a decisão de confirmar (reenviada pelo Coordenador) e confirmará localmente a transação.
Ver secção 17.3.1 do livro e slides "Tolerância a faltas no 2PC" e "Recuperação depois de falta de paragem".

f) Considere agora uma execução alternativa em que a transação decorre normalmente até o cliente fazer closeTransaction e o servidor S1 falha depois de ter efetuado o *commit* local mas antes do *acknowledge* ao coordenador.

i) Como será detectado?

Explicar que, como coordenador não recebe acknowledge deste participante ao fim de timeout após envio da decisão de confirmar, suspeita da falha silenciosa do participante.
Ver secção 17.3.1 do livro e slides "Tolerância a faltas no 2PC" e "Recuperação depois de falta de paragem".

ii) O que irá suceder à transacção global?

Dizer que a transação global já terá sido confirmada nos restantes participantes.
Complementar explicando que, assim que o participante em falha recuperar, este também receberá a decisão e confirmará a sua parte da transação distribuída.
Ver secção 17.3.1 do livro e slides "Tolerância a faltas no 2PC" e "Recuperação depois de falta de paragem".