Departamento de Informática

Faculdade de Ciências e Tecnologia

UNIVERSIDADE NOVA DE LISBOA

Licenciatura em Engenharia Informática
Sistemas Distribuídos I – 1ª chamada, 23 de Julho de 2004
2º Semestre, 2004/2005

Exame sem consulta com a duração de 2h30min

Aluno nº _____Nome: _____

1. Considere o serviço implementao pelo programa *RMI registry*.
   a. O serviço implementado por este processo é um serviço de nomes ou um serviço de directório? Justifique a sua resposta.

   b. Sabendo que o servidor *RMI registry* funciona como um servidor RMI normal, com excepção do processo de ligação (*binding*) que explora o facto de se conhecer a interface do serviço e a porta em que aguarda conexões, indique se seria possível eliminar a restrição de apenas permitir o registo de objectos remotos que executam na mesma máquina. Justifique a sua resposta.

   c. Considere um ambiente típico de Java RMI em que existe um computador no qual executa o RMI registry e um objecto remoto (servidor) RMI, cada um em sua máquina virtual (e executando em diferentes processos). O objecto remoto está registado no RMI registry e os clientes, para aceder ao objecto remoto, obtém a referência remota a partir do *RMI registry*. Suponha que o processo em que executa o RMI registry termina (por exemplo, executando "killall –9 rmiregistry"). Indique, justificadamente, quais as consequências deste facto para o objecto remoto e para o processo em que o mesmo executa.

d. Considerando as operações mais comuns disponíveis no serviço *RMI registry* (lookup, bind/rebind, list), indique se seria aceitável implementar o acesso a este serviço usando uma semântica "pelo menos uma vez". Justifique.

e. Supondo que pretende implementar a semântica "pelo menos uma vez" usando um protocolo não orientado à conexão (*connectionless* – ex. UDP), qual o protocolo de invocação preferível: request, request/reply ou request/reply/acknowledge? Justifique a resposta.

2. Considere o contexto de um sistema distribuído de ficheiros semelhante ao proposto no trabalho prático, composto por um servidor de espaço de nomes e vários servidores de conteúdos. Supondo que:
   - o servidor de espaço de nomes (SN) tem um par de chaves assimétricas. Os clientes conhecem a chave pública do servidor de espaço de nomes;
   - o servidor de espaço de nomes possui, para cada utilizador um par, nome/password;
   - o servidor de espaço de nomes partilha um chave secreta (simétrica) com cada um dos servidores de conteúdos.

   a. Apresente um protocolo que permita a um cliente (C) executar uma operação num servidor de conteúdos (SC) de forma segura, i.e., garantindo a integridade e secretismo da operação a efectuar e a autenticação mútua entre o cliente e o servidor de conteúdos. Na sua solução, ignore os possíveis ataque por "replaying". O protocolo deve ter, no máximo, 4 mensagens, duas trocadas entre o cliente e o servidor de espaço de nomes e duas trocadas entre o cliente e o servidor de conteúdos. O protocolo deve minimizar a computação necessária. Explique como seriam garantidas as propriedades indicadas. Use as notações sintéticas de descrição de protocolos criptográfico que aprendeu nas aulas.

      NOTA: Caso não consiga resolver o problema assumindo que os elementos do sistema apenas conhecem as chaves indicadas, indique explicitamente as chaves adicionais que assume serem conhecidas para cada um dos elementos do sistema.

CLT -> SN:

SN -> CLT:

CLT -> SC:

SC -> CLT:

Definição dos símbolos usados (complete):
M – mensagem a enviar do cliente para o servidor de conteúdos codificando a operação a efectuar
Res – mensagem a enviar do servidor de conteúdos para o cliente com a resposta à operação efectuada
Knc – chave secreta partilhada entre o servidor de espaço de nomes e o servidor de conteúdos

O que garante o secretismo e a integridade das mensagens?

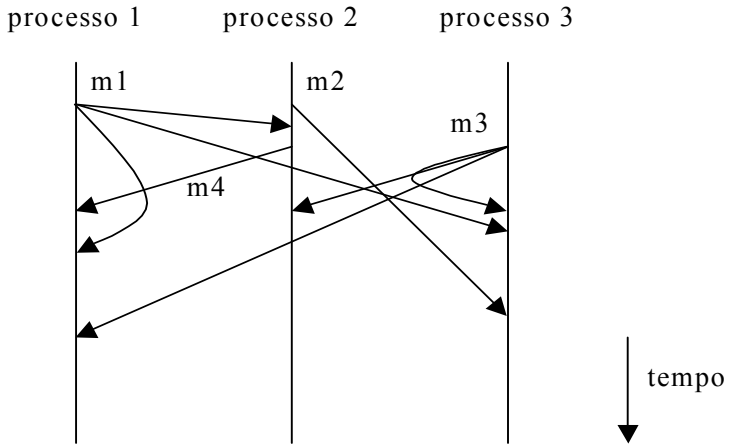
O que garante a autenticação mútua entre o cliente e o servidor de conteúdos?

b. Supondo que a interface de acesso ao serviço de ficheiros era baseada apenas em operações idempotentes, seria importante evitar os ataques por "replaying"? Justifique a sua resposta.

SIM porque / NÃO porque (risque o que não interessar)

c. Discuta de que forma seria possível alterar o seu protocolo para evitar ataques por "replaying".

3. Considere o seguinte esquema temporal representando a propagação de mensagens num sistema composto por três processos. As setas indicam a propagação de mensagens entre os vários processos. O início de uma seta indica o envio de uma mensagem. O fim de uma seta representa a recepção de uma mensagem num processo. Suponha que os processos comunicam usando um sistema de *middleware* capaz de atrasar a entrega das mensagens recebidas para garantir a sua entrega por uma dada ordem.



a. Com base na informação anterior, indique se cada uma das seguintes afirmações é **[V]erdadeira** ou **[F]alsa**. **Nota: as respostas erradas descontam.**

i. Supondo que os três processos formam um grupo, m1 e m3 são mensagem multi-ponto (ou multicast).
ii. Supondo que no processo 2, m1 é entregue antes da emissão de m4, a seguinte ordem de entrega no processo 1 respeita a ordem causal: m4, m1, m3.
iii. As mensagens m1 e m3 são entregues por ordem causal se forem entregues pela seguinte ordem relativa: processo 1: m1 antes de m3; processo 2: m1 antes de m3; processo 3: m3 antes de m1.
iv. As mensagens m1 e m3 são entregues por ordem total se forem entregues pela seguinte ordem relativa: processo 1: m1 antes de m3; processo 2: m1 antes de m3; processo 3: m3 antes de m1.
v. Para respeitar a relação *aconteceu-antes*, m2 deve ser entregue antes de m1 no processo 3.

b. Suponha que pretende usar relógios lógicos de Lamport para atribuir estampilhas temporais aos vários eventos. Indique um valor possível da estampilha temporal para cada um dos seguintes eventos:

Envio de m1:

Envio de m2:

Envio de m3:

Envio de m4:

Recepção de m1 no processo 2:

c. Suponha que se pretendem usar relógios lógicos para ordenar totalmente a entrega das mensagens enviadas no sistema, estampilhando cada mensagem com o par ***(r,i)*** em que ***r*** é uma estampilha temporal obtida a partir do relógio lógico local e ***i*** é um inteiro que identifica o processo em que a mensagem foi enviada. Supondo que os relógios lógicos são iniciados com o valor 0, e que o envio da mensagem m1 tem a estampilha (2,1), indique se é possível entregar a mensagem m1 no processo 2 quando a mensagem é recebida? Porquê?

SIM, porque / NÃO, porque (risque o que não interessar)

4. Suponha que pretende desenvolver um sistema de ficheiros distribuído para um ambiente de rede local em que o modo de acesso típico dos ficheiros é o seguinte:
   - Um ficheiro é aberto para escrita em modo *append* (adição ao conteúdo no fim do ficheiro). Em cada abertura, é efectuada uma sequência (potencialmente longa) de operações de escrita de *arrays* de *bytes* de pequenas dimensões. Um ficheiro está aberto para escrita durante um período de tempo muito curto;
   - Quando um ficheiro é aberto para leitura, todo o conteúdo do ficheiro é lido;
   - As escritas são em grande número;
   - As leituras são raras, mas quando existe uma leitura, é comum existirem mais num curto período de tempo (todas acedendo ao ficheiro completo).

   a. Nestas condições, será interessante implementar uma cache no cliente. Justifique a resposta.

SIM, porque / NÃO, porque (risque o que não interessar)

b. Se quisesse implementar uma cache de leitura/escrita no cliente (e independentemente da resposta anterior), qual a melhor granularidade para os objectos guardados na cache do cliente? Justifique.

Cache por blocos / cache do ficheiro completo (risque o que não interessar)