DI/FCT/UNL
Mestrado Integrado em Engenharia Informática

Confiabilidade de Sistemas Distribuídos
2º Semestre, 2015/2016

Teste de Avaliação nº 1 (9/Abril/2016) T1-A
Componente: Teste da Parte Teórica

PARTE I (Sem Consulta)

Questão 1

Considere as seguintes tipologias de falhas, que podem ocorrer acidentalmente ou por ação de um atacante, como resultado de uma intrusão ao nível dos componentes de um sistema distribuído. Diga quais dessas falhas podem ser toleradas a partir de uma abordagem de replicação daqueles componentes tendo por base um modelo de replicação de máquinas de estados (SMR) e com base no suporte de uma abordagem de um protocolo com as características do PBFT.

Type of failure
Crash failure
Omission failure *Receive omission* *Send omission*
Timing failure
Response failure *Value failure* *State transition failure*
Arbitrary failure

Questão 2

Considere o algoritmo de *Ben-Or* para resolver o consenso probabilístico. Explique o que garante que não é possível decidir um valor diferente em diferentes rondas num protocolo deste tipo.

Questão 3

No algoritmo PBFT, explique como é que uma réplica não secundária descobre que o primário pode ser bizantino.

Questão 4

Uma solução como IPSec,, pode ser usada para estabelecimento de propriedades de segurança num canal, com proteção ao nível rede (IP). A normalização da solução IPSec é constituída basicamente por uma pilha de *subprotocolos* específicos.

Explique quais são esses *subprotocolos*, quais as suas diferenças e que tipo de propriedades de segurança são asseguradas por cada um deles.

PARTE II (Com Consulta)

Questão 5

Explique brevemente mas de forma justificada como é que o protocolo *PAXOS* garante as seguintes propriedades

a) Apenas valores propostos são aprendidos.
b) Apenas um valor é aprendido.

Sugestão: para mostrar que apenas um valor é aprendido comece por mostrar que se uma maioria de *acceptors* aceita uma proposta, nenhum outro valor será aceite.

Questão 6

No protocolo ABD Tolerante a Falhas Bizantinas, porque é que as mensagens têm de incluir um *nonce*? Apresente a sua resposta justificadamente, argumentando sobre a importância de se incluir o *nonce* e sobre a forma como é usado e processado na execução do protocolo.

Questão 7

Considere o seguinte algoritmo:

```
preference ← input
round ← 1
while true do
   send (round, preference) to all processes
   wait to receive n – f (round, *) messages
   if received more than n / 2 (round, v) messages then
     output ← v
     preference ← v
   end
end
```

Para cada uma das propriedades do consenso, indique se o protocolo permite resolver o problema do consenso ou não.

Questão 8

a) No algoritmo PBFT, explique porque é que quando se instala uma nova *view* é necessário propagar as mensagens que não foram completadas na *view* anterior.

b) Considere os aspectos práticos da implementação do protocolo PBFT, no que diz respeito ao uso de assinaturas digitais com chave pública para autenticação de todas as mensagens no protocolo. Imagine que ponderava não incorrer nesse *overhead* na implementação do protocolo, porque poderia proteger as mensagens trocadas entre as réplicas com base em IPSec. Isso seria uma solução viável ?

- Se considera que não indique porquê.
- Se considera que sim indique como proporia que fosse o *deployment* (parametrizações, modos e tipologias de associações de segurança) da solução IPSec na ligação entre as réplicas.