

1. Justifique todas as respostas.

2. Responda às perguntas 1 e 2 numa folha e às restantes noutra.

1 [3v] - Sobre sistemas distribuídos:

1.1- Indique, justificando, **duas vantagens de distribuição** na arquitetura de aplicações informáticas. ✓

1.2- Diga o que entende por **virtualização de recursos** no contexto de aplicações informáticas. ✓

1.3- Diga sucintamente para que serve um **relógio de Lamport** e como funciona. ✓

2 [7v] - Uma subestação de distribuição de energia elétrica usa um **sistema distribuído** baseado em tecnologia **Ethernet** no controlo das linhas de energia e respetivas proteções. Em particular, as medições de correntes e tensões têm que ser realizadas com uma precisão de $3\mu s$ para que se possam correlacionar devidamente, e a atuação das proteções também tem que ser efetuada com a mesma precisão para evitar sobrecargas assimétricas em qualquer das linhas de energia.

2.1- Para conseguir a sincronização necessária pretende-se usar o protocolo de **sincronização de relógios PTP** (IEEE1588). Diga quais as principais características deste protocolo.

2.2- Sabendo que o protocolo é centralizado e que se consegue uma **exatidão** dos relógios dos *slaves* relativamente ao *master* de $1\mu s$, diga justificando se se pode inferir alguma informação sobre a **precisão** do conjunto dos relógios dos *slaves*. ✓

2.3- O controlo da subestação recebe o estado das linhas de energia através de mensagens enviadas autonomamente por sensores com disparo temporal e período de $200\mu s$. Assumindo a utilização de pilhas IP, que **protocolo de transporte** acha mais adequado para aquelas mensagens? Qual o **modelo de interação** subjacente? Justifique ambas as respostas. ✓

2.4- Considere que os atrasos que afetam as mensagens dos sensores (desde a ordem de transmissão até à receção) variam entre 8 a $12\mu s$, e que a precisão dos relógios é de $1\mu s$. Qual a **latência mínima** (contada a partir do instante de receção esperado) que permite ao controlo da subestação (recetor) **detetar uma omissão** (perda de mensagem) de um sensor? Indique.

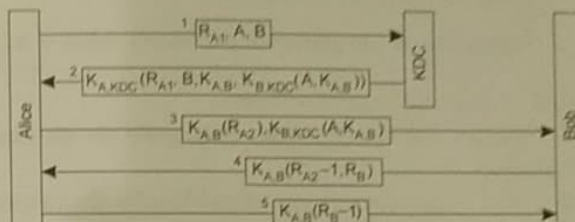
2.5- A subestação contém um **servidor** que faz o registo e classificação de todos os eventos que nela ocorrem. Quais as alternativas que consideraria para a **arquitetura interna** do servidor de modo a garantir um tratamento a atempado de muitos eventos concorrentes? Justifique. ✓

2.6- A subestação comunica com outras subestações para partilha do estado global e potenciais atuações distribuídas na rede de energia. Para isso é criada uma rede de sobreposição (*overlay*) sobre a Internet. Diga sucintamente o que é uma **rede de overlay** e como se poderia implementar.

3 [3v]- Considere o protocolo criptográfico representado na figura abaixo.

3.1- Explique para que serve e dê **duas** vantagens face à solução mais elementar recorrendo a criptografia simétrica.

3.2- Assuma que o *nonce* R_{A1} **não** era usado. Descreva sinteticamente um ataque e em que condições esse ataque poderia ser bem sucedido.



4 [3v]- Considere o algoritmo do **Bully** para a eleição de *leader*.

4.1- Assuma que inicialmente o processo (com identificador) 2 é o processo *leader* e que, além deste, estão presentes os processos 4, 5 e 8. Ilustre a execução **mais simples possível** desse algoritmo, **usando um diagrama temporal**, assumindo que o processo 5 inicia o algoritmo de eleição depois da falha do processo 2. Explique as ações de cada um dos processos.

4.2- Haveria algum problema se o processo 5 não recebesse a mensagem HALT? E se fosse o processo 8 a não receber essa mensagem? Justifique.

5 [1v]- Considere as seguintes propriedades da especificação do problema de **atomic commitment** apresentadas nas aulas teóricas:

AC3 "If some process decides commit, then all processes must have voted commit"

AC4 "If all processes voted commit and there are no failures, then all processes must decide commit"

Para cada uma delas dê **um exemplo** de como a sua omissão da especificação permitiria soluções com resultados indesejáveis, p.ex. que impediriam o seu uso em transações distribuídas. Justifique.

6 [3v]- Considere o algoritmo Paxos.

6.1- Mostre através duma execução, **usando um diagrama temporal**, com 3 processos que, se um *acceptor* não pudesse aceitar mais do que um valor, poderia não ser possível chegar a acordo (i.e. que todos os processos decidissem o mesmo valor). Explique.

6.2- Acabou o MIEEC há alguns anos e agora trabalha numa empresa, e a sua equipa propõe-se usar Paxos num projeto. Um colega seu sugeriu que o pedido de ACCEPT seja enviado **não apenas** aos processos que responderam ao pedido de PREPARE com uma mensagem PROMISE, como descrito nas notas das aulas, **mas a todos** os *acceptors*. Avalie esta proposta **quanto à sua correção** e, se for correta, apresente **uma vantagem e uma desvantagem** (em comparação com a solução apresentada na aula).