

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- O que é um **relógio de Lamport**? Existe *drift* nestes relógios?



1.2- Explique a diferença entre **Propagação Epidémica** e **Multicasting de Aplicação**.



2 [7v] - Um sistema de limpeza de grandes superfícies é composto por uma **equipa de robôs** autónomos cooperantes. A cooperação é efetuada através de um **computador central** que recebe informação dos robôs sobre as áreas já cobertas e faz a respetiva fusão para representar o **estado global** atual do processo de limpeza. Os robôs **comunicam** com o computador central e escolhem as suas áreas de intervenção consultando o estado do processo de limpeza.

2.1- Qual **modelo de interação** (Cliente-Servidor, Publicador-Subscritor, Memória Partilhada...) será mais adequado para esta aplicação e porquê?



2.2- A rede apresenta atrasos variáveis causados por variações na carga de comunicação. Qual é o método mais comum para **estimar o atraso de rede online**? Explique como funciona e faça um esboço.



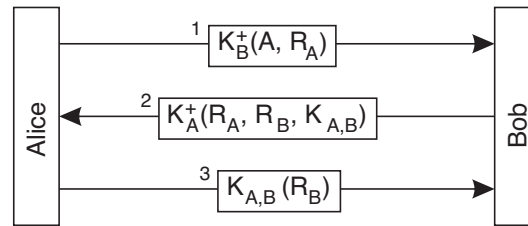
2.3- Para sincronizar os relógios dos robôs é usado **NTP** com a base de tempo instalada no computador central. Considerando que o relógio de cada robô apresenta um **drift-rate** máximo de 1ppm (1 parte por milhão), que a sincronização se faz a cada 10min e que o jitter do atraso de rede é de 5ms, qual a melhor exatidão e precisão que se pode conseguir? Mostre os cálculos.



2.4- Explique se faria sentido usar *Fault-Tolerant Average* (FTA) para melhorar a precisão da sincronização de relógio?



3 [3v] - Considere o **protocolo de autenticação com chave pública** básico apresentado na aula teórica:



3.1- Explique como é que o **Bob autentica a Alice**.

3.2- O que são **certificados digitais**? Explique como podem ser usados no protocolo representado na figura acima.

4 [2v]- Considere a seguinte afirmação:

Uma forma de reduzir a probabilidade de bloqueio na execução do protocolo *2-phase commit*, quando o coordenador falha, é os participantes elegerem entre si um novo coordenador.

Diga, justificando, se é verdadeira ou falsa.

5 [3v]- Considere *primary-backup replication*.

5.1- Este método de replicação pode ser com ou sem **bloqueio**. Explique a diferença entre estas 2 implementações e, para cada uma delas, diga, justificando, uma vantagem (em relação à outra).

Dica: Use diagramas temporais para o ajudarem na explicação.

5.2- Considere a seguinte afirmação:

O uso de *view-synchronous communication* na implementação de *primary-backup replication*, permite ter a vantagem de replicação sem bloqueio sem a sua desvantagem.

Diga, justificando, se é verdadeira ou falsa.

6 [2v]- Considere a seguinte afirmação:

A implementação de *state-based conflict-free replicated data-types*, também designados por *convergent replicated data-types*, exige o recurso a comunicação multicast fiável.

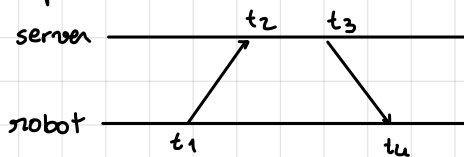
Diga, justificando, se é verdadeira ou falsa.

1.1 Lamport clocks are logical clocks that count events instead of actual time. Therefore, they are not affected by any drift, as they are not incremented using a regular time base.

1.2 Application multicasting consists on building an overlay network whose nodes are the members of the multicast group. Then, a spanning tree is built on the overlay network so that any node wanting to communicate with the group only needs to send the message to the root of the tree. On epidemic propagation there is also an overlay network but there are no defined paths: nodes send the message to some of the neighbours, which in turn send to their neighbours, etc. This method takes a longer time but is more robust to node and link crashes.

2. 2.1 Shared memory since all robots all interact with a global state in order to know where to clean next based on what has already been cleaned.

2.2 Round-trip delay: enviar mensagem com timestamp e ver o timestamp da recepção da resposta



$$\text{delay} = \frac{(t_4 - t_1) - (t_3 - t_2)}{2}$$

Para obter maior precisão, o lado do servidor pode também colocar timestamps na recepção e no envio.

2.3 $p(t) = 1 \text{ ppm}$ (drift rate) $= \left| \frac{c_p(t+\Delta t) - c_p(t)}{\Delta t} - 1 \right|$

sincronização a cada 10 minutos

jitter = 5 ms

precisão = máximo offset = 5 ms + 2 × $\frac{600 \text{ ns}}{1000000}$ = 5 ms + 1.2 = 6.2 ms

um atrasou e o outro adiantou

exatidão = $\frac{600 \text{ ns}}{1000000} + 5 \text{ ms} = 5,6 \text{ ms}$

2.4 Só faz sentido usar FTA quando a referência é virtual, ou seja, é calculando usando as médias dos clocks de cada nó. Isso só acontece quando a sincronização é distribuída. Como neste caso a sincronização é centralizada, não faz sentido usar FTA.

3. 3.1 O Bob envia um desafio encriptado com a chave pública da Alice. Como apenas ela tem acesso à sua chave privada, mais ninguém consegue descriptar a mensagem. Assim, ao receber o desafio de volta e encriptado com a chave partilhada que enviou anteriormente, o Bob sabe que só pode estar a comunicar com a Alice.

3.2 Certificados digitais garantem que uma certa chave pública é de uma determinada entidade. Assim, a Alice sabe que K_B^+ é mesmo do Bob e não de um atacante a fazer-se passar pelo Bob.


4. Falso porque caso o coordenador recupere, ficariam dois coordenadores. Consoante a fase do protocolo em que ocorre o bloqueio, os participantes devem agir de formas diferentes. Se for na fase 1, devem todos abortar. Se for na fase 2, então têm de executar um protocolo de terminação que passa por averiguar se alguém sabe o resultado da votação ou se alguém votou abort. Nestes casos, o sistema consegue avançar. Caso contrário, tem de aguardar pela resposta do coordenador.

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- Identifique e comente a diferença entre **Multiprocessador** e **Sistema Distribuído**

1.2- Refira, justificando, algumas **propriedades dos canais de comunicação** que são relevantes para os sistemas distribuídos.


1.3- Diga sucintamente o que entende por **disseminação epidémica** e dê um exemplo de uma técnica baseada nesse princípio. 


2 [7v] - Um sistema de vigilância móvel usa quatro drones que comunicam entre si com WiFi, em modo ad-hoc e multi-hop. A topologia é em linha, sendo o drone da frente munido de uma câmara (sensor) e os restantes três fazendo reencaminhamento (relays) para um nodo no lado oposto da linha designado por Estação Base (EB). Os vários drones recebem comandos de movimento da EB e esta recebe um stream de vídeo online do drone sensor. As ligações são não-fiáveis, apresentando perdas frequentes.


2.1- Os comandos de movimento são enviados com uma semântica de estado, ou seja, são mensagens periódicas que indicam a velocidade vetorial que cada drone deve aplicar. Indique, justificando, qual o **protocolo de transporte mais adequado**, TCP/IP ou UDP/IP, para as comunicações entre a EB e cada um dos drones. 

2.2- Na sequência da alínea anterior indique, justificando, qual o **paradigma de interação mais adequado**, se Cliente-Servidor ou Publicador-Subscritor. 

2.3- Na topologia em linha, cada ligação tem um atraso de rede que varia entre 0,5ms e 5ms e cada drone repetidor (relay) tem um atraso de reencaminhamento entre 1ms e 2ms. Caracterize o **atraso total** desde que um pacote é enviado pela EB até que é recebido pelo drone sensor.

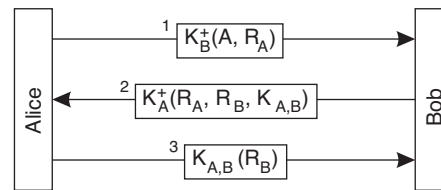
2.4- A EB contém um servidor de tempo, e.g., do protocolo NTP, usado para sincronizar os relógios dos drones. Considerando que esses relógios não têm drift, qual a **melhor precisão que se pode garantir** em cada drone? E será necessário fazer ressincronização periódica? 

2.5- Para diminuir a interferência mútua entre os drones, as transmissões são organizadas em slots consecutivas de forma semelhante a TDMA. Contudo, em vez de se usar um relógio global para determinar o início de cada slot, os drones usam as transmissões uns dos outros para manter a respetiva ordem no ciclo TDMA. Em cada drone é possível implementar um relógio global com base na contagem das slots. Esse relógio é **físico** ou **lógico**? Justifique. 

2.6- A rede TDMA em linha é uma rede overlay, implementada sobre WiFi com MAC do tipo CSMA/CA. Tendo em conta as características da rede referidas na alínea 2.3, indique, justificando, a **duração mínima das slots** que permite **garantir ausência** de interferência mútua. Discuta se, relaxando esta garantia, ou seja, usando uma abordagem estocástica ao atraso da rede, **seria** aumentar a eficiência do protocolo fazendo mais transmissões por slot. 

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

3.1- Diga para que serve, explicando o propósito de cada uma das mensagens representadas.



3.2- Tipicamente um canal de comunicação seguro garante pelo menos uma de 3 propriedades. Admitindo que destas se pode dispensar a confidencialidade, seria possível simplificar este protocolo? Justifique.

4 [1v]- Enumere 2 métricas para caracterizar um sistema tolerante a falhas. Dê um exemplo que ilustre que estas métricas são independentes. I.e que um sistema A pode ser mais tolerante a falhas do que o sistema B, usando uma destas métricas, mas ser menos tolerante a falhas usando a outra. Justifique.

Dica: Use uma figura para o ajudar na explicação.

5 [1v]- Considere o algoritmo do **convite (invitation)** para a eleição de *leader*.

Assuma um estado em que há 2 grupos no sistema cujos membros são os processos (cujos identificadores são) $\{3, 4\}$ e $\{2, 5\}$. Ilustre a execução **mais simples possível** desse algoritmo, **usando um diagrama temporal**, assumindo que o processo 4, *leader* do primeiro grupo, descobriu que o processo 2 é o *leader* do segundo grupo. Explique as ações de cada um dos processos.

6 [2v]- Considere uma execução do algoritmo *two-phase commit* em que todos os participantes votam *commit* na primeira fase e em que o coordenador falha, de modo que nenhum dos participantes recebe qualquer mensagem da segunda fase.

Se os participantes conseguirem entrar em contacto uns com os outros, podem decidir? Em caso afirmativo, diga justificando qual deverá ser a decisão. Em caso negativo, explique porquê.

7 [3v]- Considere a implementação de *State Machine Replication* com Paxos.

7.1- Assuma que se usa uma "janela" de 4 operações (em relação à primeira operação ainda pendente). Explique como o *leader* deverá proceder quando recebe "simultaneamente" 3 pedidos de clientes, e tomou conhecimento da aceitação de todos os pedidos anteriores, excepto os 2 imediatamente anteriores.

7.2- Assuma que é eleito um novo *leader* após a falha do *leader* "anterior". Considere a seguinte afirmação:

O novo leader pode não ter que executar uma nova instância de Paxos para cada uma das operações cuja decisão desconhece.

Diga, justificando, se é verdadeira ou falsa.

1. 1.1 Um sistema distribuído consiste em vários computadores a trabalhar em conjunto, que podem estar geograficamente distantes e cuja comunicação não pode ser considerada instantânea. Num multiprocessador também há vários processos a acontecer em simultâneo mas há recursos físicos partilhados, tais como memória.

1.2 connection-based / connectionless: if the connection needs to be established before starting the data transfer (relevant for the initiation of communication and management of shared resources)
 reliable / unreliable: if the channel ensures messages are not lost / duplicated (relevant to maintain consistency between sender and receiver)
 ensures order: if the channel guarantees messages are delivered in the order they were sent (relevant for consistency)
 message / stream based: if data is sent in packets or in a flow (relevant for data interpretation)
 flow control: if the channel prevents fast senders from overflowing slow receivers
 number of end points: if the communication is between just two nodes (unicast), a group of nodes (multicast) or all nodes in the network (broadcast)

1.3 Epidemic propagation is a type of multicast communication where nodes send the message to (some) of their neighbours in a lazy way. There are no defined paths, so the propagation may take a long time but is very robust to node/link crashes and is highly scalable. One example is the anti-entropy protocol, where nodes regularly exchange info with random neighbours.

2. drone^① ↔ drone^② ↔ drone^③ ↔ drone^④ ↔ base station

2.1 UDP porque se se usasse TCP a baixa fiabilidade do canal obrigaria a muitas retransmissões. Assim, as mensagens seriam enviadas com atraso e já não seriam úteis quando fossem recebidas.

2.2 Publisher-subscriber porque assim a base envia mensagens sempre que precisa em vez de as nodes terem de pedir (mensagens frequentes)

2.3 delay varia entre 0.5 ms e 5 ms
 relay tem atraso entre 1 e 2 ms

$$\text{atraso máximo} = 5 + 2 + 5 + 2 + 5 + 2 + 5 = 20 + 6 = 26 \text{ ms}$$

$$\text{atraso mínimo} = 0.5 + 1 + 0.5 + 1 + 0.5 + 1 + 0.5 = 2 + 3 = 5 \text{ ms}$$

2.4	jitter ₁ = $d_{\max} - d_{\min} = 21 \text{ ms}$	$\delta_1 > 21 \text{ ms} \times (1 - 1/2) = 10.25 \text{ ms}$
	jitter ₂ = $5 + 2 + 5 + 2 + 5 - (0.5 + 1 + 0.5 + 1 + 0.5) = 15.5 \text{ ms}$	$\delta_2 > 7.75 \text{ ms}$
	jitter ₃ = $5 + 2 + 5 - (0.5 + 1 + 0.5) = 10 \text{ ms}$	$\delta_3 > 5 \text{ ms}$
	jitter ₄ = $5 - 0.5 = 4.5 \text{ ms}$	$\delta_4 > 2.25 \text{ ms}$

como não há drift não é preciso fazer sincronizações periódicas

2.5 Lógico porque é implementado com base em eventos e não com base temporal

2.6 Para garantir a ausência de interferência, os slots têm de ter a duração das transmissões mais o atraso de rede. Assim, cada slot tem de ter o tempo de transmissão de dados + $5ms$.
A abordagem estocástica aumenta a eficiência.

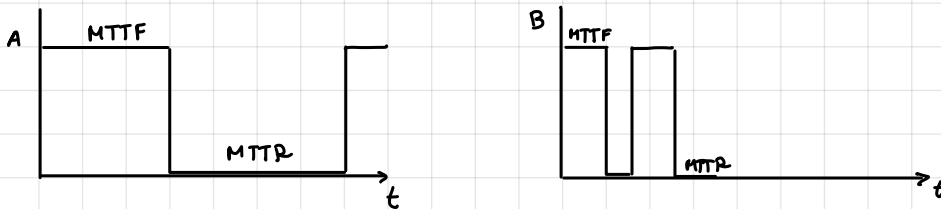
- 3.3.1
- 1) Alice envia uma mensagem encriptada com a chave pública do Bob com a sua identidade e um desafio. Para Bob responder ao desafio tem de ter a sua chave privada para conseguir desencriptar a mensagem, autenticando-se assim.
 - 2) Bob responde ao desafio da Alice e envia-lhe um desafio também, em conjunto com uma chave partilhada. Tudo vai encriptado com a chave pública da Alice, pelo que só ela será capaz de desencriptar a mensagem.
 - 3) Alice envia o desafio de volta ao Bob, encriptado com a chave partilhada. Assim, Alice está autenticada perante Bob.

3.2 Retirando a chave partilhada, a confidencialidade não seria garantida e o protocolo seria simplificado. No entanto, a integridade também não seria garantida, uma vez que um Chuck poderia adulterar as mensagens e a Alice e o Bob nunca reparariam. No entanto a autenticação não seria posta em causa. Usam K_A em vez de $K_{A,B}$.

4. Reliability e availability

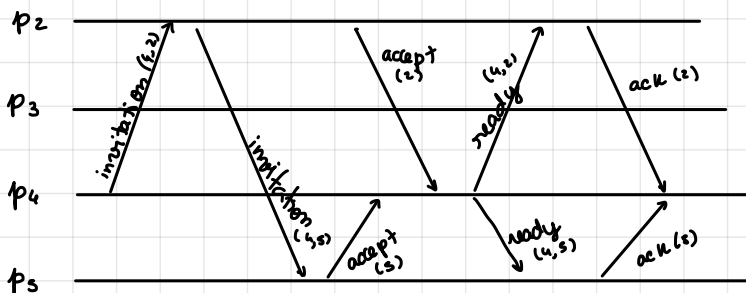
Reliability mede a probabilidade de um sistema não ter falhado até ao tempo t . Availability mede a probabilidade de um sistema funcionar no tempo t , mesmo que tenha falhado anteriormente (pode já ter sido reparado).

Reliability avaliada por mean time to failure.



A é mais reliable mas menos available

5. {3, 44} {2, 54}
↑ ↑
leader leader



6. Os participantes ao entrarem em contacto uns com os outros, ficam a saber se alguém votou abort. Se alguém tiver votado abort, então é certo que a decisão do coordenador seria abort. Também acontece que algum participante sabe a decisão do coordenador. Caso todos tenham votado commit e não saibam a decisão, têm de esperar pela recuperação do coordenador.

7. 7.1 janela de 4 operações
· aceite
· ?
· ?
} simultaneo

janela anda 2 para a frente
processa os 2 que foram aceites
pode começar paxos para mais dois

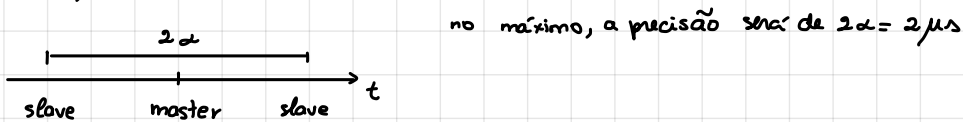
7.2 Verdade, pode mandar uma mensagem prepare única para todas as operações que não conhece.

Teste 23 de novembro de 2017

- 1.1 Partilha de recursos, melhoria de performance, melhor availability e reliability, escalabilidade
- 1.2 Consiste na emulação de um sistema, o que permite por exemplo manter acesso a legacy systems.
- 1.3 Relógio que conta eventos em vez de tempo. Permite ordenação de eventos.

2. $\delta = 3\mu s$

2.2 $\alpha = 1\mu s$



2.3 período = $200\mu s$

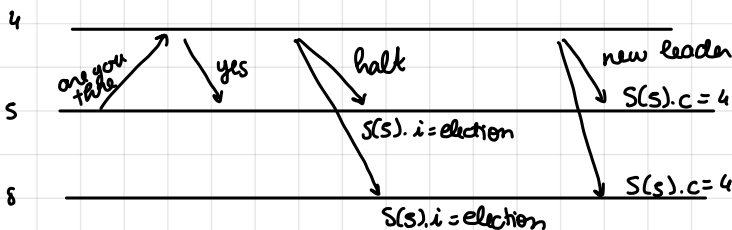
TCP porque não se pode perder mensagens nem receber duplicados
cliente- servidor porque toda a comunicação é triggered pelo cliente

2.4 atrasos 8 a $12\mu s$
precisão = $1\mu s$ $13\mu s$

3. 3.1 permite a comunicação entre a Alice e o Bob usando uma shared key gerada pelo KDC, pelo que não têm que decorar uma
→ menos recursos de memória necessários
→ mais escalável

- 3.2
 - Chuck sabe $K_{A,KDC}$
 - consegue enviar mensagem 1 ao KDC e aprende $K_{A,B}$
 - consegue ter acesso a todas as mensagens entre A e B

4. 4.1 2, 4, 5, 8



- 4.2 halt faz o processo passar para election
5 não recebe → não passa para election → não guarda id 4 → continua normal e com $S(5).c = 2$
mas 8 tem $S(8).c = 4$ → assertion 1 violada
mesma coisa se fosse o 8