ACTIVIDADES DEL TEMA 1

ACTIVIDAD 1

a) (i) un resultado correcto.
b) Como consecuencia de un defecto, se producirá (iii) un error o (iv) un fallo.
c) Como consecuencia de un defecto, se producirá (iii) un error porque la operación % es la división entera y se proporcionan reales.
d) Se puede diseñar tolerante a fallos para reducir al máximo el número de defectos con una programación robusta, pero seguirá habiendo defectos inevitablemente (no todos se pueden eliminar).

ACTIVIDAD 2

1) PARADA: f+1 porque podrían pararse todas las réplicas excepto una, que quedaría como primaria.
2) OMISIÓN DE RECEPCIONES: f+2 porque es necesario que queden, al menos, dos réplicas para asegurar que no se promociona una secundaria a primaria cuando la primaria sigue activa (es una situación equiparable a la planteada en el enunciado).
3) No puede soportarse el modelo de fallos arbitrarios bajo este modelo pasivo porque, si por ejemplo se da un fallo bizantino (se proporciona una respuesta incorrecta no detectable), por la "confianza" de todas las réplicas secundarias en la primaria, no va a haber forma de detectar dicho fallo, que se asumiría como valor correcto. En un modelo activo, en cambio, podrían hacerse votaciones entre los valores de las réplicas para detectar estos fallos.

ACTIVIDAD 3

Si, por ejemplo, se tienen dos nodos del sistema conectados con un cable en mal estado (pelado, quizá) que está generando fallos por omisión, sería difícil aplicar un modelo de parada, porque no podría determinarse exactamente si hay un defecto en el cable, si uno de los nodos tiene problemas o si los dos los tienen. De hecho, los dos componentes afectados pensarían que es el otro el que tiene problemas y se generarían dos sospechas contrarias que no tendrían una sencilla resolución en el modelo de parada. Sería brusco parar ambos componentes y habría que estudiar más a fondo qué convendría hacer.

ACTIVIDAD 4

Partimos de la premisa de que todos los fallos - excepto los de tipo bizantino - van a poder reducirse a fallos de parada en este modelo, por lo que únicamente estos fallos deberán tenerse en cuenta. Para detectar fallos se utilizará, como mecanismo, la sospecha de los fallos.

En el caso de que la réplica que falle (sea primaria o secundaria) tenga un estado de amnesia, cuando se reinicie y reincorpore al sistema lo hará fijando unos datos preestablecidos para este momento, sin tener en cuenta las peticiones atendidas antes de fallar. En el caso de que el estado sea de amnesia parcial, parte de los datos que tenía antes de fallar los recuperará y los

conocerá, y el resto adoptarán un valor predeterminado. Por último, si el estado para el reinicio es de pausa, deberá recuperar exactamente la información que tenía antes de fallar para poder incorporarse.

Si se sospecha que una réplica secundaria está fallando y se aplica parada, para reincorporarla al sistema deberá revisarse primero el estado de reinicio y actuar según lo explicado en el párrafo anterior y, una vez contenga la información que le corresponda, se reincluirá en el sistema con una IP nueva que el resto deberá conocer y reconocer como perteneciente a una réplica secundaria. Durante este proceso de recuperación de la réplica secundaria, el sistema puede mantener su funcionamiento normal.

Si, en cambio, la réplica a recuperar es la primaria, para garantizar la consistencia, una de las réplicas secundarias cuyo estado esté totalmente actualizado deberá asumir el papel de primaria, y el resto de réplicas secundarias deberán asumir que esta ha pasado a ser la primaria. En el momento de reinsertar a la antigua réplica primaria, esta adoptará el papel de secundaria y, cuando contenga la información correspondiente según lo descrito en el párrafo segundo, adoptará una nueva IP que el resto de réplicas deberá conocer y asociar a una nueva réplica secundaria. Cuando cae la réplica primaria, inevitablemente el sistema deja de estar operativo un tiempo, que es menor si una secundaria ocupa el papel de primaria que si la antigua primaria debe hacerlo, dado que la antigua primaria, antes de reinsertarse con una nueva IP, debe revisar su información, su estado.

ACTIVIDAD 5

Si se da un particionado, o bien las particiones funcionan independientemente, o bien la particionada se anula y el resto funcionan.

En el caso de que se anule, la disponibilidad (probabilidad de que un sistema o servicio esté operativo en un determinado instante) se garantiza siempre y cuando quede al menos una réplica de servidor activa, si bien es cierto que se reduce la probabilidad de que el servicio esté operativo en un determinado instante porque si se producen más particiones posteriormente y todas las que se aíslen se descartan, podría llegar a no quedar ninguna réplica de servidor operativa. La calidad del servicio proporcionado bajaría si se saturaran las solicitudes a servidores y estos, por ser menos, no pudieran atenderlas debidamente.

En el caso de que funcionen de forma independiente, la calidad del servicio se vería afectada negativamente porque se podrían generar inconsistencias entre la información almacenada en cada uno de los servidores, ofreciendo distintas respuestas del mismo servicio. No obstante, si el servidor aislado siguiera actuando como servidor del servicio, la disponibilidad sería la misma que antes de darse el particionado.

ACTIVIDAD 6

Los cinco atributos a comparar son: fiabilidad (“*reliability*”), seguridad (“safety”), disponibilidad (“availability”), mantenibilidad (“maintainability”), seguridad (“security”).

- Fiabilidad (“*reliability*”) y seguridad (“safety”): a mayor seguridad, mayor fiabilidad, pues si se reducen las probabilidades de que un sistema falle catastróficamente en un

intervalo de tiempo, habrá mayor probabilidad de que esté operativo en un instante determinado sabiendo que en un instante t anterior lo estaba.

- Fiabilidad ("*reliability*") y disponibilidad ("availability"): a mayor fiabilidad, mayor probabilidad de que el sistema esté disponible en un instante determinado.
- Fiabilidad ("*reliability*") y mantenibilidad ("maintainability"): cuanto mayor sea la fiabilidad, menos ocasiones de recuperar el sistema (de mantenerlo) se darán; asimismo, cuanto mayor sea la fiabilidad, menos componentes del sistema fallarán al mismo tiempo y, por lo tanto, el tiempo necesario para recuperar el sistema cuando se dé un fallo, será menor, aumentando así la mantenibilidad.
- Fiabilidad ("*reliability*") y seguridad ("security"): a mayor seguridad, menos casos de intrusión en el sistema se darán y, consecuentemente, se reducirán las ocasiones en las que una entrada de un individuo no autorizado consigue parar el servicio reduciendo la fiabilidad.
- Seguridad ("safety") y disponibilidad ("availability"): cuanto mayor sea la primera, menos veces fallará el sistema y, por tanto, mayor será la disponibilidad.
- Seguridad ("safety") y mantenibilidad ("maintainability"): cuanto mayor sea la primera, menos veces fallará el sistema y, por tanto, menos veces será necesario recuperar el sistema, si bien es cierto que la probabilidad de la primera no afecta al tiempo necesario de mantenimiento.
- Seguridad ("safety") y seguridad ("security"): cuanto mayor sea la segunda, menos veces se tendrán accesos no autorizados que paren de forma catastrófica el servicio.
- Disponibilidad ("availability") y mantenibilidad ("maintainability"): cuanto mayor sea la primera, menos veces fallará el sistema o algún componente y, por tanto, menos veces será necesario recuperar el sistema, si bien es cierto que la probabilidad de la primera no afecta al tiempo necesario de mantenimiento.
- Disponibilidad ("availability") y seguridad ("security"): cuanta más seguridad presente el sistema, menos ocasiones habrá de parada del servicio o de uno de sus componentes por intrusiones indeseadas y, así, mayor será la disponibilidad.
- Mantenibilidad ("maintainability") y seguridad ("security"): si la segunda es elevada, se darán pocas ocasiones en las que debido a una intrusión en el sistema sea necesario recuperarlo para ponerlo de nuevo en funcionamiento.

Se puede concluir, pues, que todas las características anteriores tienen influencias entre ellas, en mayor o menor medida. En concreto, un aumento de la seguridad ("security"), conllevará la reducción de los casos en los que el sistema es forzado a fallar por un intruso y es necesario recuperarlo, por lo que se tiene un aumento de la fiabilidad, disponibilidad, mantenibilidad y seguridad ("safety"). Además, cuanto más rápido sea recuperar un sistema, mayor será la fiabilidad y disponibilidad del mismo y, recíprocamente, cuanto más fiable y disponible sea un sistema, menos veces será necesario recuperarlo. Por último, la fiabilidad, la disponibilidad y la seguridad ("safety") están íntimamente ligadas y un aumento o disminución en una de ellas tiene las mismas consecuencias sobre las otras dos.

ACTIVIDADES DEL TEMA 2

ACTIVIDAD 1

1.1)

Ventajas:

- Menor coste de comunicaciones (menor carga de la red) porque sin un sistema de pertenencia a grupos, cada nodo únicamente emite un mensaje a la réplica primaria sin tener en cuenta el estado percibido de la primaria por las otras réplicas secundarias.
- Implementación más sencilla de la monitorización.

Inconvenientes:

- Si el detector no es perfecto, no es un buen sistema cuando se tienen más de dos nodos; pueden darse posibles inconsistencias entre todos los nodos sobre los estados de la réplica primaria y, en su caso, sobre la elección de su reemplazo.
- Cada nodo tiene únicamente información sobre el estado de la réplica primaria y no sobre el resto de réplicas secundarias.

1.2)

En el escenario planteado en el enunciado se da, como mucho, completitud débil (siempre y cuando la sincronía sea perfecta y las comunicaciones no fallen), pues se revisa únicamente el estado de la réplica primaria por parte de cada una de las secundarias, de forma que, si fallara alguna de las réplicas secundarias, no se detectaría dicho fallo.

En lo que a precisión se refiere, si las comunicaciones nunca fallan y es puramente sincrónico el sistema (tiempo real), se tiene precisión fuerte, pues ninguna de las réplicas sospechará de la primaria hasta que no obtenga una respuesta.

1.3)

Esto es parecido a un escenario con modelo de caída y enlace.

Si falla una red que contiene una réplica secundaria, se daría una “partición”, aislando del resto de nodos al nodo secundario de la subred que ha fallado. Como el nodo principal no revisa si los mensajes que envía llegan o no a los nodos secundarios, este nodo se perdería y dejaría de participar en el sistema a no ser que se detectara de otro modo su pérdida y se forzara su reinserción, dotándole de una nueva IP y haciendo que el primario enviara a dicha dirección.

1.4)

Nota: se refiere al ejemplo de partida, el del enunciado inicial.

Los algoritmos de elección de líder son también de consenso (todos deciden lo mismo) con la finalidad de elegir un nodo como líder. Si se complementara el mecanismo de sospecha descrito

con un algoritmo de elección de líder, sería necesario que los nodos secundarios se pudieran de acuerdo para elegir el líder.

No es equivalente porque el servicio de pertenencia consiste en saber qué nodos están operativos, funcionando, y en este cada nodo secundario solo sabría si funciona o no el primario, pero no el resto de secundarios. Además, en el escenario actual alguno de los secundarios, por ejemplo, por un fallo en el canal, podría interpretar que el primario ha dejado de funcionar, mientras que el resto no sospecharían esto porque recibirían de forma normal los mensajes y, consecuentemente, las interpretaciones no serían homogéneas, mientras que en un servicio de pertenencia a grupo hay una homogeneidad garantizada con un protocolo de consenso.

ACTIVIDAD 2

No puede implantarse en un sistema puramente asincrónico porque en ellos no se podría distinguir un nodo que realmente ha fallado de uno lento.

Se resuelve creando un sistema de pertenencia a grupos que informe de cambios de vista (proporcionados por el sistema de pertenencia) a todos los nodos. Es muy importante que todos los mensajes estén bien intercalados con los mensajes del sistema. Se incluye, para ello, un algoritmo de consenso para determinar qué nodos están vivos y cuáles no.

No se puede implantar un detector de fallos perfecto en un sistema asincrónico. Puede llegar a implantarse un detector de fallos estableciendo cierta sincronía, necesaria para determinar el tiempo límite de respuesta a partir del cual se considera fallido un nodo. No obstante, dicho detector nunca sería perfecto, pues, por ejemplo, una comunicación más lenta de lo normal con un nodo en un momento dado podría derivar en la interpretación errónea de que se ha dado un fallo.

ACTIVIDAD 3

3.1)

Dado que cuando un proceso sospecha de la caída de algún otro difunde sus sospechas, que son aceptadas por todos los demás, no puede implantarse de este modo un detector de fallos perfecto porque, si el nodo ha sospechado por un falso positivo (por ejemplo, derivado de una sobrecarga en la red que hace más lenta de lo tolerable la comunicación), será interpretado como fallo por todos los nodos sin revisar su estado y, por tanto, el nodo sospechado se tratará como caído aunque no lo estuviera realmente.

3.2)

La completitud, pues en esa situación los procesos que han fallado serán considerados como sospechosos (completitud), pero quizá por el desconocimiento, a priori, del tiempo máximo de envío, se den falsos positivos en las sospechas que consideren a un proceso correcto como fallido por haberse dado una comunicación lenta.

3.3)

Precisión fuerte (ningún proceso es sospechado antes de fallar): no puede darse porque por comunicaciones lentas, por ejemplo, en los sistemas asincrónicos puede sospecharse como fallido un proceso que realmente sea correcto.

Completitud fuerte (llega un momento en que todo proceso fallido es permanentemente sospechado por todo proceso correcto): sí que podría llegar a darse si todos los nodos revisaran el estado del resto, ya que se alcanzaría un instante a partir del cual todos los procesos detectarían el proceso fallido.

ACTIVIDAD 4

Referencia 1)

J. Tang, M. Larrea, S. Arévalo, E. Jiménez: "*Fault-tolerant broadcast in anonymous distributed systems with fair lossy channels*", Technical Report EHU-KAT-IK-06-14 of the University of the Basque Country UPV/EHU, December 2014

Enlace: http://www.sc.ehu.es/acwlaalm/research/EHU-KAT-IK-06-14.pdf [Consulta: febrero, 2016]

En este artículo se plantea la implementación de algoritmos de un broadcast de un sistema tolerante a fallos donde se dé pérdida de mensajes en los canales de comunicación y/o anonimidad de los mensajes recibidos.

En el artículo de Chandra y Toueg se utiliza el identificador del proceso como parte de la información en caso de fallo, pero este artículo extiende el diseño de detectores de fallos a sistemas distribuidos anónimos donde los procesos no tienen identificadores. La clave del trabajo reside en conseguir identificar los procesos sin perder la anonimidad del sistema.

Referencia 2)

K. P. Kihlstrom, L. E. Moser, P. M. Melliar-Smith: "*Byzantine Fault Detectors for Solving Consensus*", The Computer Journal, Vol. 46, No. 1, 2003

Enlace:
http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.10.328&rep=rep1&type=pdf
[Consulta: febrero, 2016]

En este artículo se amplía el trabajo de Chandra y Toueg considerando detectores de fallos no fiables para resolver el consenso en sistemas asíncronos distribuidos sujetos a fallos bizantinos.

Referencia 3)

C. F. Espinosa Gualotuña, C. R. Egas Acosta: "*Selección de un Gateway de respaldo, adaptando el algoritmo Omega en Redes 6LowPAN*", Revista Tecnológica ESPOL – RTE, Vol. 28, N. 5, 179-193, diciembre 2015

Enlace: www.rte.espol.edu.ec/index.php/tecnologica/article/download/433/300 [Consulta: febrero, 2016]

En este artículo se propone una adaptación del algoritmo propuesto por Chandra y Toueg para mejorar la confiabilidad del acceso a internet de una red sensor inalámbrica, con la implementación de un prototipo de red 6LowPAN.

En concreto, se propone un algoritmo a implementar en cada nodo sensor inalámbrico, tomando como parámetros el nivel de batería de los nodos y su identificador, lo que permite la elección de un Gateway de respaldo de entre los posibles nodos candidatos. En el caso de que el nodo Gateway seleccionado en un inicio del funcionamiento de la red se encuentre indisponible, la utilización del algoritmo permite tener una alternativa de selección de un nodo Gateway de respaldo, cuya finalidad es brindar mayor confiabilidad en la conectividad de este tipo de redes con Internet.