TSR: Segon parcial

Aquest examen comprèn 10 preguntes d'opció múltiple. En cada cas solament una resposta és correcta. Ha de respondre's en una fulla a part. Les respostes correctes aporten 1 punt a la qualificació d'aquesta prova. Les incorrectes redueixen la qualificació 0.33 punts.

1. Per a desplegar adequadament un servei replicat, hem d'assegurar que:

A	Les seues rèpliques se situen en diferents ordinadors que no depenen d'una mateixa font de fallades.
B	El codi de les seues rèpliques s'ha escrit en JavaScript.
C	Les seues rèpliques usen ZeroMQ com el seu *middleware* de comunicacions.
D	El desplegament es fa amb les eines de Docker.

2. Considerant l'acoblament en serveis distribuïts, hauríem d'assegurar que els components mostren:

A	Acoblament feble perquè implica que els components són fiables.
B	Acoblament fort perquè implica que els components utilitzen algorismes descentralitzats.
C	Acoblament fort perquè implica que els components estan replicats.
D	Acoblament feble perquè implica que els missatges entre components seran xicotets i que la sincronització entre components serà molt relaxada.

3. Quin dels següents models de servei "*cloud*" es preocupa per la gestió de l'elasticitat dels serveis desplegats pels seus clients?

A	SaaS.
B	IaaS.
C	PaaS.
D	Tots ells.

4. Quina de les següents afirmacions sobre seguretat en sistemes distribuïts és FALSA?

A	Els sistemes distribuïts són potencialment vulnerables a causa de la seua necessitat de comunicació entre nodes.
B	Els atacs de denegació de servei consisteixen a "inundar" les instàncies servidores amb més sol·licituds de les que es poden gestionar.
C	Els sistemes distribuïts són potencialment vulnerables perquè no poden proporcionar mecanismes de control físic d'accés a tots els seus nodes i canals.
D	Els sistemes distribuïts tenen una "base informàtica de confiança" (TCB) fàcilment identificable.

5. **Siga aquesta execució en un sistema amb tres processos (P1, P2 i P3):**
W1(x)1, R2(x)1, W2(y)3, W1(y)2, R1(y)3, R3(y)2, R3(y)3, R3(x)1,…
…on P2 no accepta el valor 2 de "y" escrit per P1 en la quarta acció de la traça. El model de consistència de memòria més fort suportat per aquesta execució és:

A	FIFO.
B	Seqüència.
C	"Cache".
D	Causal.

6. **Siga aquesta execució en un sistema amb tres processos (P1, P2 i P3):**
W1(x)1, R2(x)1, W2(y)3, R1(y)3, W1(y)2, R2(y)2, R3(x)1, R3(y)3, R3(y)2,…
El model de consistència de memòria més fort suportat per aquesta execució és:

A	FIFO.
B	Seqüencial.
C	"Cache".
D	Estricte.

7. **El component "mongod" en MongoDB és un…:**

A	Proxy mantingut pels processos clients per a interactuar amb els servidors de configuració de MongoDB.
B	Equilibrador de càrrega que redirigeix les peticions clients al servidor "mongos" apropiat. Aquest gestiona localment una instància de la base de dades MongoDB.
C	Un mòdul JavaScript que ha de ser inclòs en els programes NodeJS per a accedir a la base MongoDB.
D	El servidor que localment gestiona una instància de la base de dades MongoDB.

8. **Si som desenvolupadors, per a reduir les possibles vulnerabilitats dels serveis distribuïts que escriguem... quina és la millor de les següents alternatives?**

A	Comprovar periòdicament que les nostres eines de desenvolupament, les biblioteques i middleware utilitzats i els nodes on es realitzen els desplegaments hagen rebut les seues actualitzacions relacionades amb seguretat.
B	Desplegar els nostres serveis amb docker-compose.
C	Assignar privilegis d'administrador a tots els usuaris dels nostres serveis.
D	Confiar en la reputació del sistema operatiu utilitzat en els nodes amfitrions, comprovant que tots els nodes utilitzen el millor sistema en aquest aspecte.

9. Considerant aquest programa...

```
var cluster = require('cluster');
var http = require('http');
var numCPUs = require('os').cpus().length;
function processRequest(req,resp) {
  // Some code for processing an HTTP request and generate its response.
}
if (cluster.isMaster) {
  for (var i=0; i < numCPUs; i++) cluster.fork();
  cluster.on('exit', function(worker) {
    cluster.fork();
  });
} else {
  http.createServer(processRequest).listen(8000);
}
```

Quines afirmacions són certes?

A	Si un procés treballador mor, el mestre en genera un altre.
B	Aquest programa crea tant processos treballadors com a nuclis de processador hi haja en la màquina.
C	Els processos treballadors utilitzen la funció "processRequest" per a gestionar cada petició HTTP rebuda.
D	Totes les anteriors.

10. Suposem que un programador ha escrit un component broker en NodeJS que escolta en múltiples ports: 8000 (missatges clients), 8001 (connexió per a modificar la configuració del broker) i 8002 (missatges cap als treballadors). Quan els clients i els treballadors se situen en altres nodes, es recomana assignar el seu port 8000 al port 80 de l'amfitrió i el seu port 8002 al 82 de l'amfitrió. El programa del broker es diu Broker.js i aquest fitxer està en la mateixa carpeta que aquest Dockerfile:

```
FROM zmq-devel
COPY ./Broker.js /Broker.js
CMD node /Broker.js
```

Després d'usar l'ordre "docker build –t myBroker ." en la carpeta, s'ha intentat executar el broker, però és incapaç d'interactuar amb alguns clients i treballadors locals.
Per a corregir el problema, caldrà:

A	Afegir aquesta línia al Dockerfile: PORT 8000 8001 8002
B	Afegir aquestes línies al Dockerfile: PORT 80:8000 82:8002 PORT 8001
C	Afegir aquesta línia al Dockerfile: EXPOSE 8000 8001 8002
D	No es necessita fer res en el Dockerfile. El problema està relacionat amb els arguments i opcions utilitzats en l'ordre "docker run myBroker".