

Programación Distribuida y Tiempo Real

Práctica 2

1) Buscar y utilizar en vagrant una versión más reciente de Ubuntu LTS que la dada en la explicación de práctica. Identificar versión, origen y si tuviera diferencias, explicar brevemente. Mostrar con una captura de pantalla la identificación en VirtualBox la máquina virtual y el inicio de la terminal en funcionamiento. Consulte a un asistente de IA (deje constancia de cuál) si hay alguna versión en particular que puede ser mejor que otra y por qué.

2) Ejecutar los experimentos del ejercicio 4 a), b) y c) de la práctica anterior en dos computadoras físicamente diferentes. Se copia aquí el enunciado para mayor comodidad: comunicar 10^1 , 10^2 , 10^3 , 10^4 , 10^5 y 10^6 bytes desarrollando experimentos que muestre el tiempo que toma en el cliente

a) La función `write(...)` para cada cantidad de datos. Tomando C como ejemplo:

```
t0 = ...
write(...)
t1 = ...
tiempo = t1 - t0;
```

b) La función `read(...)`. Tomando C como ejemplo:

```
t0 = ...
n = read(sockfd, buffer, 255);
t1 = ...
tiempo = t1 - t0;
```

c) Grafique y explique los resultados obtenidos. En particular, identifique si las diferencias de tiempos son proporcionales a las cantidades de datos, por ej: ¿`write(...)` con 1000 bytes toma 10 veces más tiempo que con 100 bytes? Identifique si el tiempo de la función `read()` se mantiene constante, dado que involucra siempre la misma cantidad de datos.

Proveer una descripción mínima de las computadoras usadas (CPU, RAM, OS). Aclare si las computadoras están en la misma red local o en dos redes locales diferentes (en este último caso, es posible que deban modificar al menos un router si ambas computadoras tienen acceso vía NAT). Pueden utilizar las computadoras del aula o sus propias computadoras. Documentar y entregar no solamente el código de los procesos de comunicaciones sino también el propio experimento.

3) Desarrollar los mismos experimentos de comunicaciones que en el caso anterior, con las mismas computadoras, pero ahora con la misma cantidad de datos en un sentido y en el inverso (experimento “ping-pong”). El tiempo total “de ida i vuelta” de los datos dividido por 2 es el que se utiliza para una estimación del tiempo de mensajes en una dirección. Comparar los tiempos del ejercicio anterior con los que se obtienen en este experimento para cada cantidad de bytes.

4) Teniendo en cuenta los experimentos de tiempos realizados, desarrollar scripts para desplegar un ambiente de experimentación de comunicaciones en una computadora con Vagrant para los siguientes escenarios:

a.- Dos máquinas virtuales, cada una con un proceso de comunicaciones.

b.- Una máquina virtual con uno de los procesos de comunicaciones y el otro proceso de comunicaciones en el host.

En todos los casos deberían quedar los resultados disponibles para su posterior análisis. Se debe resolver el problema de “asincronismo” que genera un error si el proceso “cliente” inicia la ejecución antes que el proceso “servidor”. Consultar con un asistente con IA cuál es sería el mejor método experimental para estimar el tiempo de comunicaciones entre dos computadoras diferentes y comente si le parece correcto o puede identificar algún problema en la respuesta del asistente con IA.

5) Explique si los resultados de tiempo de comunicaciones del experimento del ej. anterior se ven afectados por el orden de ejecución y las diferencias de tiempo de ejecución iniciales de los programas utilizados ¿qué sucede si por alguna razón hay una diferencia de 10 segundos en el inicio de la ejecución entre un programa y otro? Sugerencia: incluya un “sleep” de 10 segundos entre la ejecución de diferentes partes de los programas y comente.

Entrega de la práctica (individual o en grupos de dos estudiantes como máximo):

- Se debe entregar un informe de lo realizado para cada ejercicio. Debe tener un formato bien definido identificando materia, trabajo práctico y autor/es. Se debe entregar en formato electrónico con tipo de archivo .pdf, en tamaño de hoja A4.

- Todos los resultados experimentales de tiempo deben presentarse en formato de tabla y en formato gráfico. En el caso del formato gráfico debe explicarse qué es lo más importante que se puede identificar en cada gráfico.

- Para cada programa modificado o generado para resolver los ejercicios, debe explicarse el cambio o la implementación realizada. Si bien el programa fuente puede estar comentado, el cambio o la implementación realizada debe explicarse en el texto del informe (no es aceptable “ver código fuente” en el informe).

- Se debe entregar en formato electrónico tanto el informe como todo el código fuente usado/desarrollado.