INSTITUTO POLITÉCNICO NACIONAL

ESCUELA SUPERIOR DE CÓMPUTO

DESARROLLO DE SISTEMAS DISTRIBUIDOS

Profesor:
Coronilla Contreras Ukranio

Protocolo Solicitud Respuesta Confiable versión 3 (Corrección)

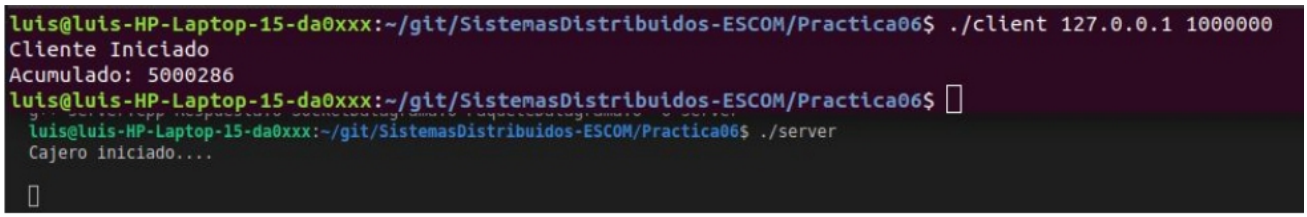
4CM1

EQUIPO 3

Protocolo Solicitud-Respuesta confiable

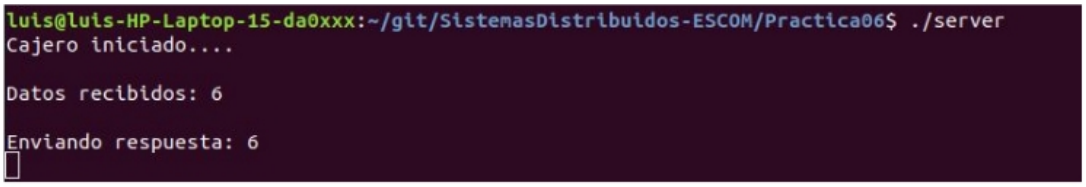
Prueba del primer ejercicio:

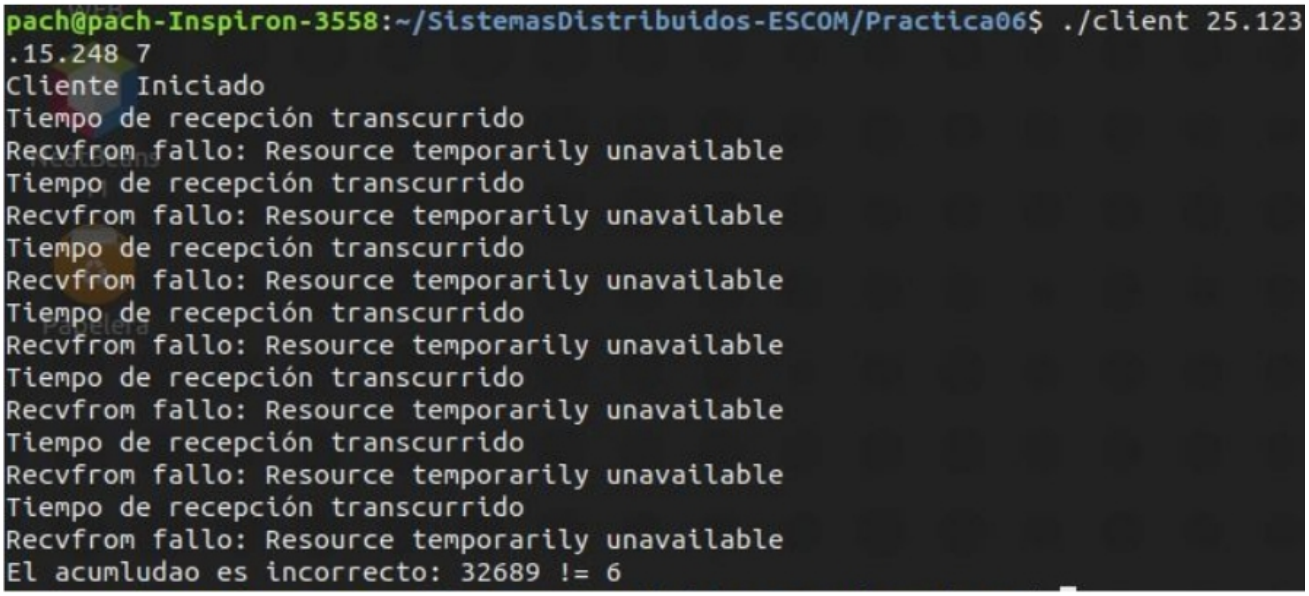
Al probar en un solo equipo funciona, incluso si usamos números grandes como 1000000. Donde el cliente imprime el valor total del estado de la cuenta.



Captura de pantalla mostrando la ejecución del cliente y del servidor en una misma computadora con n igual a 1000000.

Al probar dos equipos, utilizando hamachi, no funciona, incluso si usamos números pequeños como 7, en este caso, probamos con 7, de los cuales el servidor solo recibió 1.

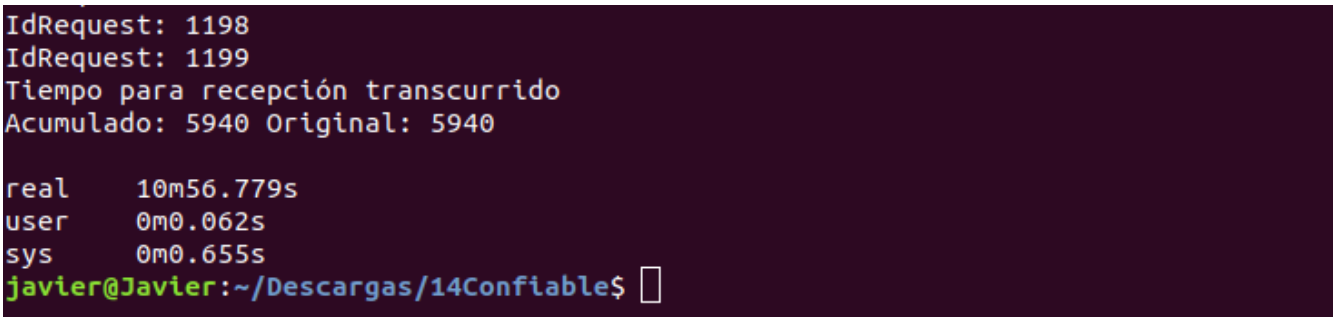




Esto se debe a que al implementamos los tres primitivos (doOperation, getRequest y sendReply) a través de datagramas UDP, así que sufren de los mismos errores de comunicación. Es decir:
• Sufren de fallas de omisión.
• No se garantiza que los mensajes se entreguen en orden del remitente.

Prueba del Segundo ejercicio:

Este error se puede atribuir a que al llegarle un paquete duplicado al servidor, este devuelve un suma distinta a lo que el cliente almacena correctamente.
Para la solución de este problema, se agrego un Idrequest a la clase Solicitud, en la método doOperation, que funciona como un identificador del paquete. En la clase Respuesta, también se hizo un modificación en el método getRequest. Este al ser enviado al servidor, compara que el Id del paquete enviado por el cliente, corresponda ahora con el Id por parte del Servidor. Si estos coinciden, entonces, el número que envía el cliente se devuelve correctamente para que el servidor haga la suma, y ahora, este resultado se envía al cliente.
De este modo, si e servidor recibe un paquete duplicado, este lo descarta, y nuevamente envía la operación que este había realizado anteriormente.



Captura de pantalla del cliente, con n igual a 1200, mostrando la respuesta final verdadera, y la que al final regresó el servidor. Por curiosidad, decidimos ver el tiempo que tardaba el cliente en ejecutar esas solicitudes, y nos dio que fueron casi 11 minutos. 10 minutos con 56 segundos para ser exactos.

Captura de pantalla del servidor, mostrando la respuesta final.