Programación Sobre Redes

Teo 14: Introducción a Sockets (UNIX) II

Nicolás Mastropasqua

October 30, 2020

Instituto Industrial Luis A. Huergo

Contenidos

1. Repaso

2. Client-server: Many to one

Repaso

Sockets

- Abstraccion que permite la comunicación entre dos aplicaciones residentes en distintos o mismo host.
- Los sockets pueden trabajar en distintos dominios de direcciones (UNIX domain vs Internet).
- Soportan distintos familias de protocolos de transmisión de datos como SOCK _STREAM o SOCK_DGRAM

Client-server: Many to one

Demo 2 Clients, 1 Server

DEMO

¿Comunicación bloqueante?

Si usamos llamadas bloqueantes podemos esperar indefinidamente por el primer cliente mientras el segundo tiene mucho que enviarnos y nunca se lo pedimos.

Soluciones

- Llamadas no bloqueantes y espera activa.
- Usar la llamada al sistema select().
- Utilizar un proceso (o thread) para atender cada cliente.

Solución 1: Llamadas no bloqueantes

DEMO

Llamadas no bloqueantes

- Se usa la llamada fcntl() para indicar que los sockets del servidor sean no bloqueantes.
- Al intentar recibir mensajes, si se devuelve -1 tenemos que ver si es por un error verdadero o porque no había nada en espera de ser recibido. Usamos la variable errno para discernir esto.
- Este enfoque tiene un problema. Hacemos **busy waiting** ocupando el procesador innecesariamente.

¿Podemos hacerlo mejor?

Threads!

DEMO

Threads

- El servidor se bloquea esperando una conexión (accept)
- Cuando esto ocurre, crea un thread que ejecuta una función para atender al cliente (en este caso, imprimir todo lo que envía)
- Para lo anterior, el thread necesita conocer el nuevo socket creado. Por lo tanto, es un parámetro que debemos enviarle en su creación
- Luego de todo esto, el servidor puede bloquearse para esperar una nueva conexión entrante

Ejercicio

Realizar un programa utilizando los conocimientos anteriores de manera que el servidor soporte la conexión concurrente de *n* clientes. Cada cliente podrá enviar mensajes al servidor, el cual le responderá con el mismo mensaje pero en mayúscula.