SOCKETS

El kernel se divide en dos subsistemas que son el de ficheros y el de procesos(es una abstracción de la maquina intermediada por el kernel(es el dueño de todos los recuros de un ordenador))(con el mod usuario y portegio(este))se encarga de aislar las zonas de meoria de los otros procesos, **trap**,

Modo de trabajo de unix con los ficheros. OPEN, read, write , close,

Sockets: vamos a crear una abstaccion que representa el extremo de un cable, es decir, un punto final de comunicación, los sockcets pertenecen a los procesos, hicieron un diseño independiente de:

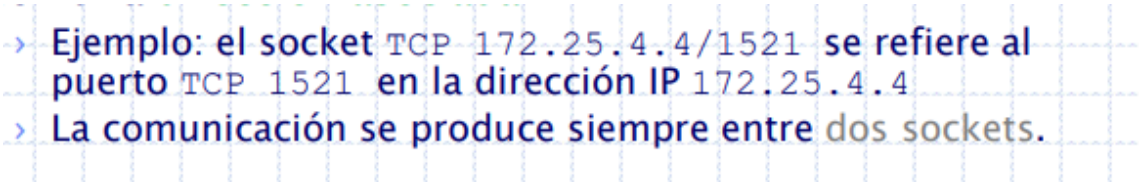
-el proticolo de comunicación, se usa para la torre de protocolos tcp/ip y otros protocolos como una torre osi, el bluethoot,

-son independientes del lenguaje de porgramcion, es decir, que se pueden usar con cualquier lenaguaje, es decir, puedes tener un proceso en Python y otro en ruby, es decir, no necesitas un socket Python para solo los d epython, esto se logra mediante llamadas al sistema,

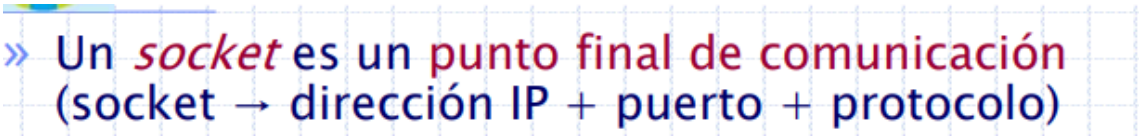
-independientes del sistema operativo, a pesar de estar hecho en el kernel de unix, se puede usar en cualquier so

Un socket te ofrece una interfaz a los servicios en nivel de transporte, (host to host en tcp/ip), un nivel de transporte mueve bits entre procesos, signfiica que tienes un ainterfaz en la que puedes mandar bits a otro proceso, tenemos tres practicas con niveles de transportes distintos,.

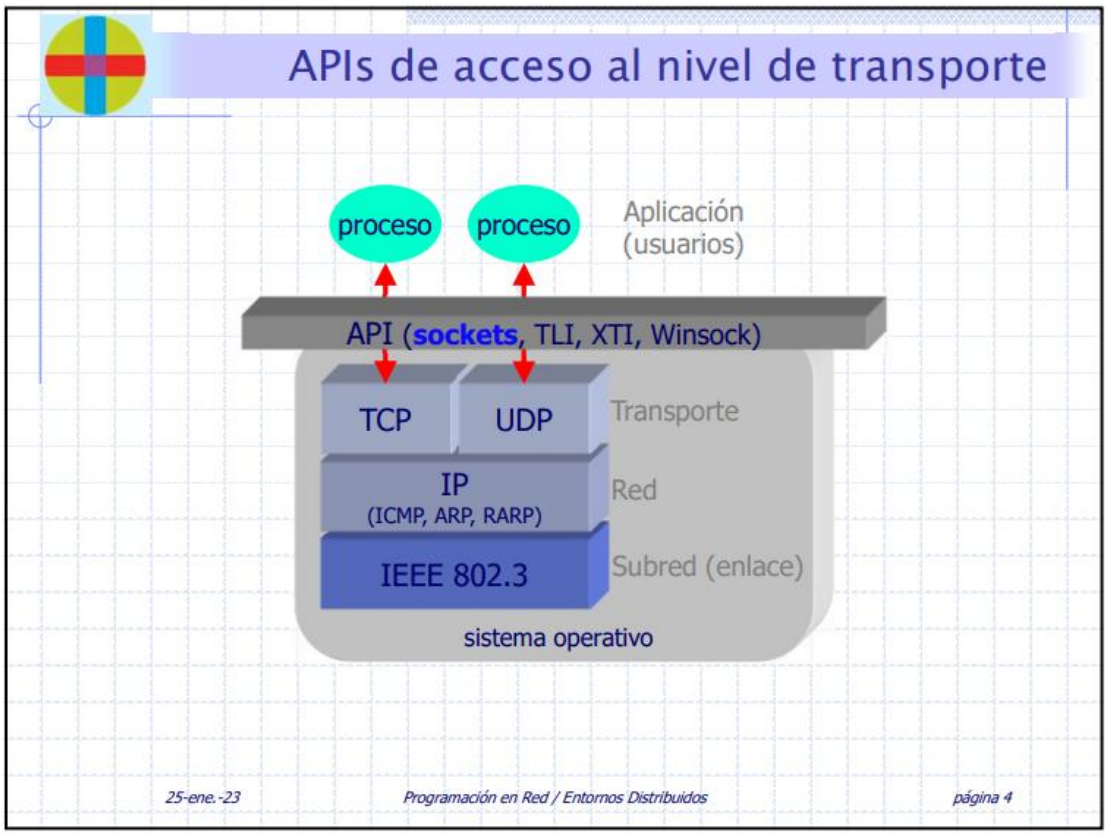
Los scokets Usan direccionamiento **directo,** es decir direcciones en cada extremo de la comunicacion, y **asimetrico**, es decir, tu sabes a quien le tienes que enviar el mensaje pero el destinatario no sabes quien se lo tiene que enviar,



La direccion incluye :



POSIX: interfaz portable(lo puedes utilizar en distintos sistemas unix) en el so unix,



Subred(enlace): ethernet(MAC, dentro de una LAN), host to host, punto a punto

Para poder colocar enlace deajo de ip: una interfaz que posee send packet y receive packet, un loopback(en cualquier maquina que tenga esta interfaz puedes porbar la torre de protocolos),

Diferencia entre subred(homogeno) y red(heterogeneo)

Red: ip(encaminamiento y reenvio), le ofece a la capa de transporte llegar al destino mediante saltos, es decir, mueve bits entre maquinas(interfaces de red), maneja todo lo del nivel inferior,

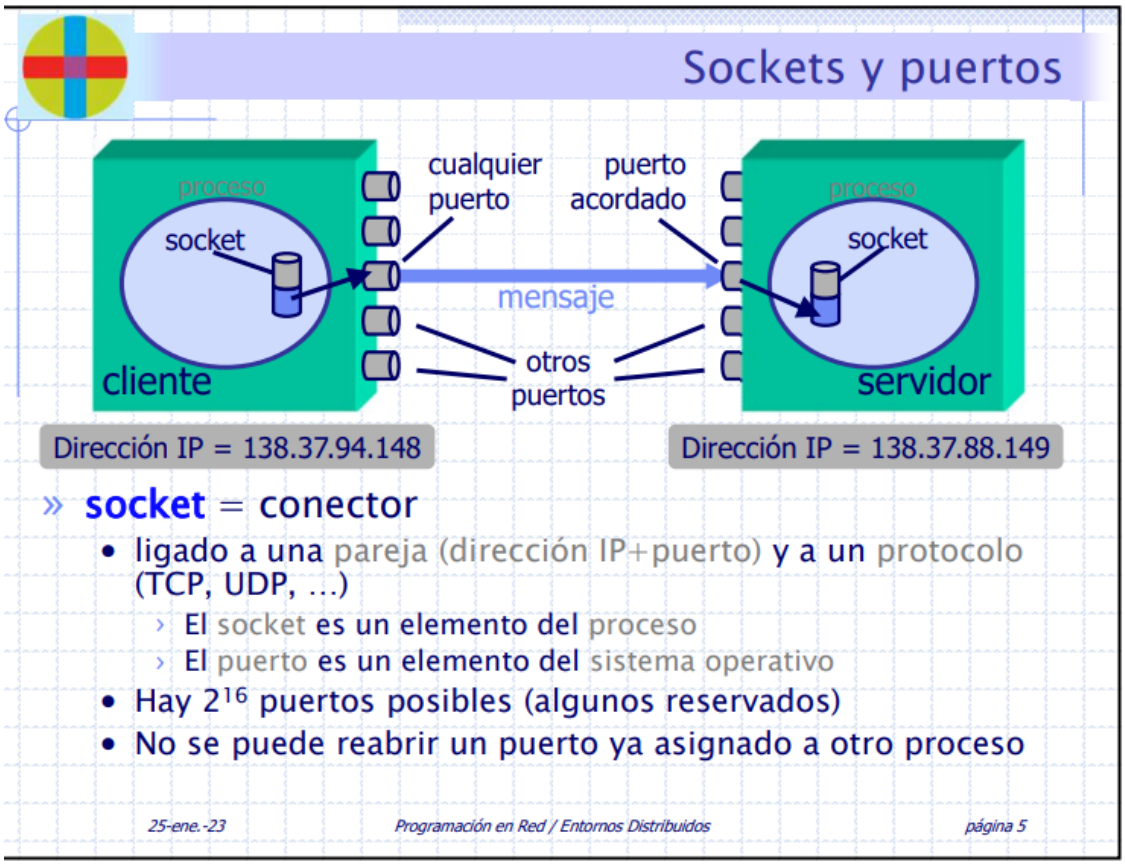
Transporte: udp(es un nivel de transporte(permite comunicación entre procesos), conmutacion de paquetes(lleva los bits dentro d ellos paquetes)), udp es un multiplexor, es decir, que los procesos son capaces de hablar, los puetos disntingyena los varios procesos dentro de una maquina,

Tcp: protocolo conmutación de circuitos, es decir, establece un circuito virtual(un cable), tiene que resolver los porbemas de osi(son 3)

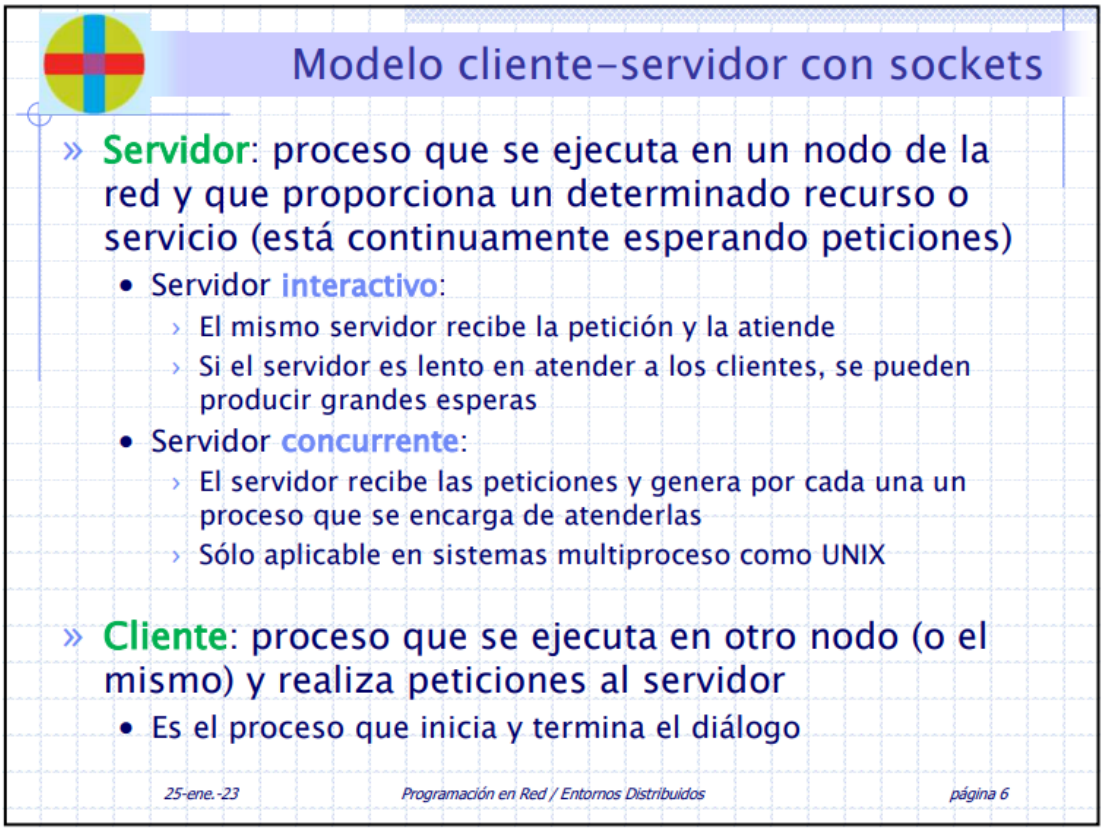
Cuando vas a trabajar tieens que elegir uno entre tcp y udp

Nivel 4(aplicacion): como pasa de un proceso al kenrel(con una llamada al sistema)

API de sockets(api de llamadas al sistema)



Quieres q un proceso hable con otro proceso de otra maquina, para ello utilizamos sockets, no puedes enganchar dos proceso al mismo puerto, solo uno por puerto, tcp deja el puerto reservdo drunte 90s cuando se ha muerto el porcso asciado al puerto,



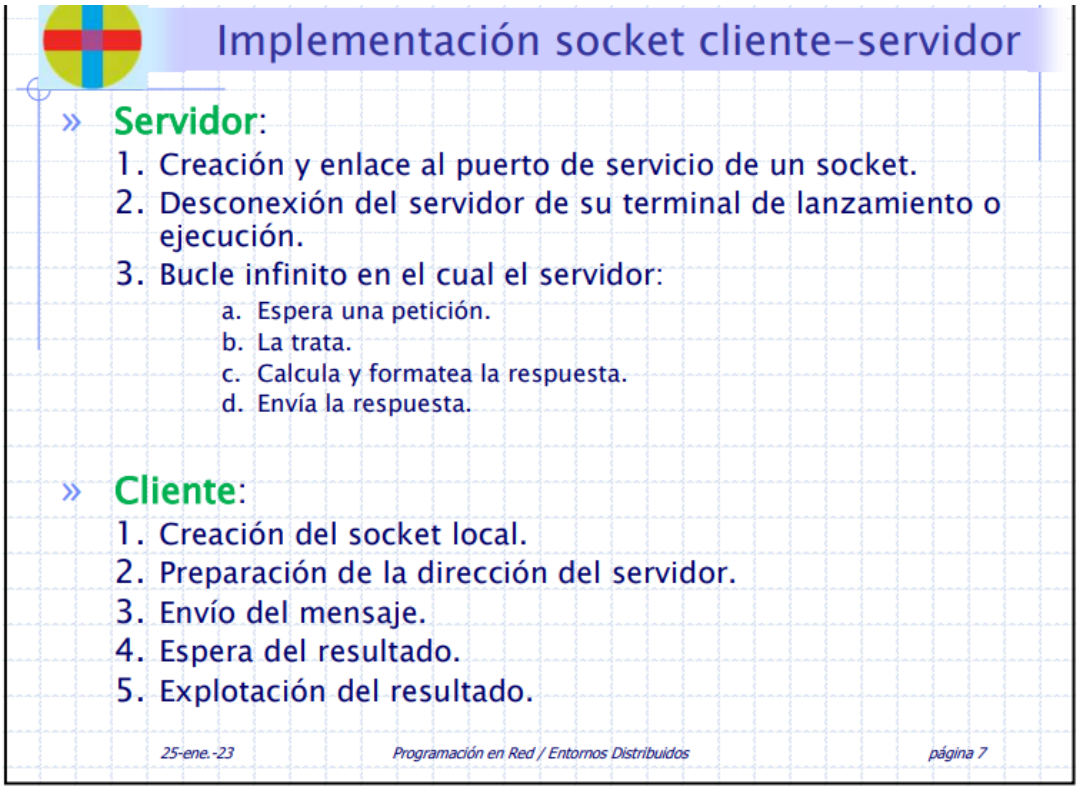
Los dos procesi que se comunican no son iguales, uno es cliente y otr es servidor, el cliente le pide cosas al servidor y el servidor lo responde, mientras espera, el proceso del servidor esta durmiendo

El trabajo del servidor es dar servicio o escuchar peticiones,

Hay 2 tipos de servdiores :

Intercativo vs concurrente: itecativo solo atiende a una petciion, esta pensado para peticiones de corta duracion, concurrente, cea con un fork por cada petición que recibe,

Que hace un cliente y un servidor:



Servdiro: renegar de su padre, x