

## Laborator 6 – Retele de calculatoare

### Demoni si superservere

1. Scrieti versiuni ale serverelor TCP pentru serviciile **echo**, **daytime** si **time** care presupun ca socket-ul pe care comunica este duplicat in descriptorii de fisier cu numerele 0, 1 si 2 ai procesului server (altfel spus, nu trebuie sa creeze socket, nici sa-l pregateasca pentru serviciu; pur si simplu presupune ca altcineva a facut acesti pasi si a conectat socket-ul de serviciu la descriptorii de fisier 0, 1 si 2 ai procesului server).

2. Scrieti un superserver **inetd** care accepta cereri de la clienti TCP pentru serviciile de **echo**, **daytime** si **time** folosind versiunile de server de la punctul 1.

*Indicatie:* serverul **inetd** va crea sockets pentru cele trei servicii si ii va inregistra intr-un file descriptor set pe care il va pasa ca parametru apelului *select*. Cand unul dintre descriptorii de fisier devine disponibil pt citire (i.e., un client a facut o cerere), **inetd** apeleaza *accept* si paseaza noul socket creat unui proces copil creat cu *fork* care va executa serviciul corespunzator cererii invocand apelul sistem *exec* (cu variantele sale) caruia ii paseaza ca parametru programul corespunzator dezvoltat la punctul 1.

Vezi si diagrama urmatoare. Pentru a duplica socket-ul creat de *accept* in descriptorii 0, 1, 2 ai procesului copil trebuie apelata functia *dup* (sau *dup2*). Dupa ce socket-ul de serviciu a fost duplicat, procesul copil poate executa folosind functia *exec* programele executabile pe care le-ati scris la punctul 1, care, asa cum s-a specificat, citesc de pe file descriptorul 0 (acum un duplicat al socket-ului de serviciu) cererile clientilor si le scriu raspunsul in file descriptorul 1 (poate fi si 0) care refera si el in acest moment in procesul copil socket-ul de serviciu in urma duplicarii.

