



Pentru utilizarea paradigmei client-server, exista un flow de evenimente, si anume:

Pentru server

Creare socket `int sockfd = socket(domain, type, protocol)`

- sockfd: socket descriptor, integer
- domain: integer, reprezinta domeniul de comunicare AF_INET (pt protocolul IPV4) sau AF_INET6 (pentru IPV6)
- type: tipul comunicarii
 - SOCK_STREAM: TCP
 - SOCK_DGRAM: UDP
- protocol: valoarea protocolului pentru IP, care este 0.

Setsockopt `int setsockopt(int sockfd, int level, int optname, const void *optval, socklen_t optlen);`

- ajuta la manipularea optiunilor pentru socketul referit prin file descriptorul sockfd. Acest apel este strict optional, dar poate fi de folos daca doriti sa reutilizati adresa si portul. Previne erori precum "address already in use"

Bind `int bind(int sockfd, const struct sockaddr *addr, socklen_t addrlen);`

- dupa crearea socketului, functia bind fixeaza adresa si numarul portului socketului creat intr-o structura de date customizata, numita addr.

Listen `int listen(int sockfd, int backlog);`

- aceasta functie pune serverul intr-un mod pasiv, unde asteapta conexiunea cu clientul. Backlog-ul defineste lungimea maxima a cozii de asteptare pentru care sockfd poate creste. Daca se primeste un request de conexiune atunci cand coada este full, clientul primeste o eroare cu indicatia ECONNREFUSED.

Accept `int new_socket= accept(int sockfd, struct sockaddr *addr, socklen_t *addrlen);`

- extrage prima cerere de conexiune din coada de asteptare pentru socketul care asculta (sockfd), creeaza o noua conexiune cu socketul si returneaza un file descriptor care refera catre socketul cu care s-a realizat conexiunea. In acest moment, conexiunea este stabilita intre server si client si sunt pregatiti sa transfere date.

Pentru client

Conectarea la socket - la fel ca la server

Connect `int connect(int sockfd, const struct sockaddr *addr, socklen_t addrlen);`

- acest apel de sistem conecteaza socketul referentiat prin descriptorul sockfd cu adresa specificata de addr. In addr avem specificate adresa serverului si portul

Exercitiu:

Sa se scrie un server TCP iterativ care functioneaza astfel: atunci cand un client se conecteaza si trimite una din key-urile cunoscute de server atunci se va executa comanda asociata. Pentru orice altceva se va trimite raspunsul "Unknown key" si se inchide conexiunea.

Exemplu fisier:

1111:/bin/ls:

2222:/bin/pwd:

3333:/bin/ps: