

Sockets

Mode non connecté

- Utilise le protocole UDP (User Datagram Protocol)
- Analogue à une communication par courrier
- Nécessite l'adresse de destination à chaque envoi, et aucun accusé de réception n'est donné

Sockets

Etapes Serveur Mode non connecté

- Création : `socket()`
- Association : `bind()`
- Réception / Lecture : `recvfrom()` / `rcvmsg()`
- Envoi / Ecriture : `sendto()` / `sendmsg()`
- Fermeture : `closesocket()`

Sockets

Etapes Client Mode non connecté

- Création : `socket()`
- Envoi / Ecriture : `sendto()` / `sendmsg()`
- Réception / Lecture : `recvfrom()` / `recvmsg()`
- Fermeture : `closesocket()`

Sockets

sendto()

- `int sendto(int sockfd, char *msg, int len, int flags, struct sockaddr *to, int tolen);`
 - `sockfd` : identifiant de socket retourné par `socket()`
 - `msg` : données à envoyer
 - `len` : taille des données à envoyer (en octets)
 - `flags` : 0
 - `to` : adresse de destination
 - `tolen` : taille de la structure `sockaddr` (`sizeof (struct sockaddr)`)
 - La fonction `sendto()` retourne la taille des données envoyées (en octets). Sinon, elle retourne la valeur -1
- Exemple :

```
struct sockaddr_in serv_addr;  
serv_addr.sin_family = AF_INET;  
serv_addr.sin_port = htons(3333);  
serv_addr.sin_addr.s_addr = inet_addr("127.0.0.1");  
sendto(id_sock, "Bonjour", 7, 0, (struct sockaddr *)&serv_addr, sizeof(struct sockaddr));
```

Sockets

recvfrom()

- `int recvfrom(int sockfd, char *msg, int len, int flags, struct sockaddr *from, int *fromlen);`
 - `sockfd` : identifiant de socket retourné par `socket()`
 - `msg` : données à recevoir
 - `len` : taille des données à recevoir (en octets)
 - `flags` : 0
 - `from` : adresse de l'émetteur
 - `fromlen` : taille de la structure `sockaddr` (`sizeof (struct sockaddr)`)
 - La fonction `recvfrom()` retourne la taille des données reçues (en octets). Sinon, elle retourne la valeur -1
- Exemple :

```
char  msg[100];
struct sockaddr_in  from_addr;
int  addr_size = sizeof(struct sockaddr);
recvfrom(id_sock, msg, 100, 0, (struct sockaddr *)&from_addr, &addr_size);
```