

## TP n°10

### Les sockets en mode connecté

*Le but de ce TP est de manipuler les sockets en mode connecté (utilisation de TCP).*

## 1 Le *chatbot*

Nous souhaitons écrire un *chatbot* en réseau (un *chatbot* implique une IA que nous ignorerons ici). Le serveur prend son numéro de port d'écoute en argument. Le client prend en arguments l'adresse et le numéro de port du serveur. L'utilisateur du client peut saisir des phrases qui sont envoyées au serveur. Ce dernier envoie une réponse qui est choisie aléatoirement parmi une liste définie. Quand l'utilisateur saisit "Au revoir", le serveur répond également "Au revoir" et les deux applications s'arrêtent.

### Questions

1. Précisez l'ensemble des messages échangés dans l'application.
2. Écrivez les deux applications.
3. Rendez le serveur multi-clients (gestion des clients en parallèle à l'aide de fils). Le fils s'arrête lorsque le client dit "Au revoir".
4. Comment gérer l'arrêt du serveur avec un CTRL+C ? Modifiez votre application.
5. Lors du positionnement d'un signal avec `sigaction`, vous pouvez utiliser l'option `SA_RESTART`. Consultez le man pour avoir de l'aide sur cette option. Pour illustrer son fonctionnement, modifiez le serveur pour qu'il affiche le nombre de clients actuellement connectés lorsqu'il reçoit un `SIGUSR1`.

## 2 Le proxy (UDP et TCP)

Nous souhaitons écrire un proxy. Il se met en attente de requêtes en UDP qui contiennent simplement un numéro de port TCP. Il crée alors un fils qui se met en attente de deux connexions sur ce numéro de port. Une fois les deux connexions établies, le proxy écoute sur la première socket et attend une taille et un message (un tableau de `char`). Il envoie alors la taille et le message dans la deuxième socket. Il écoute ensuite dans la deuxième socket et procède de la même manière. Il continue ainsi jusqu'à ce que la taille envoyée soit 0 ; il envoie alors 0 dans l'autre socket et s'arrête.

Pour tester son fonctionnement, nous développons deux applications qui se contentent d'envoyer la chaîne "Bonjour" et d'attendre cette même chaîne.

### Questions

1. Précisez l'ensemble des messages échangés dans l'application.
2. Écrivez ces deux applications.