

Objectifs : Réaliser vos premières applications client-serveur utilisant le protocole TCP. En particulier, apprendre à réaliser des échanges de base entre un programme serveur et un ou plusieurs programmes clients et comprendre le fonctionnement de TCP.

Il sera important de suivre les étapes suivantes (valables pour tous les TP) :

1. effectuer une réflexion avant réalisation pour :
 - comprendre la structure de l'application (définir les programmes qui la composent et le rôle de chacun).
 - définir un protocole d'application / d'échange.
2. implémenter les programmes de l'application
3. exécuter les programmes sur différentes machines (impératif)
4. réaliser des tests mettant en évidence quelques propriétés de TCP.

Notations et rappel :

Le protocole de transport **TCP** permet de réaliser des communications en mode connecté. Un message envoyé en **TCP** est transféré/acheminé sous forme de flux d'octets. Enfin, **TCP** gère la duplication et la remise dans l'ordre des paquets à leur réception.

1 Premiers échanges

L'objectif de cet exercice est de mettre en oeuvre un simple échange de données entre un programme client et un programme serveur. Il s'agit d'une base avant de traiter des scénarios mettant en évidence le comportement de TCP (exercices suivants).

L'idée est d'écrire :

- un programme serveur qui attend un message, sous forme de chaîne de caractères, envoyé par un client, affiche ce message, renvoie au client la taille du message reçu (en nombre d'octets), termine le traitement de ce client et termine (un seul client est traité par le serveur).
- un programme client qui demande à l'utilisateur de saisir une chaîne de caractères au clavier (taille max 200 octets), envoie cette chaîne au serveur, reçoit un entier de la part du serveur et compare la valeur de cet entier avec la taille du message envoyé. L'objectif est d'obtenir des valeurs identiques pour toute exécution de votre application.

Pour vous aider, un squelette de programmes client et serveur, ainsi qu'un Makefile sont mis à votre disposition sur Moodle. Vous y trouverez des instructions de développement et d'autres détails.

Une fois les deux programmes écrits entièrement, réaliser plusieurs tests d'exécution et s'assurer de l'affichage correct des chaînes de caractères coté serveur. Il est aussi attendu de vous de minimiser le nombre d'octets échangés (réduits à des données utiles -voir instructions dans les squelettes).

Observer et analysez ce qui se produit à l'exécution si :

- vous lancez le serveur, suspendez son exécution avant l'appel à `accept(...)` et lancez le client. Pour suspendre l'exécution, demandez une saisie d'un caractère ou autre au clavier avant l'appel `accept(..)` (rappelez vous de la démonstration faite en cours). Poursuivre l'exécution du serveur une fois avoir analysé le comportement du client.
- vous faites la même chose mais en lançant 7 clients avant de reprendre la suite de l'exécution du serveur. Que se passe-t-il pour l'ensemble des clients avant et après la reprise de l'exécution du serveur.

2 La suite sera rédigée prochainement